

EXHIBIT C

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

In these Infringement Contentions, AGIS Software Development LLC (“AGIS”) contends that at least the following claims of U.S. Patent No. 9,408,055 (the “’055 Patent”) identified below are infringed by the Accused Products (e.g., phones and tablets) which are manufactured, sold, offered for sale, and/or used by Defendants ZTE (TX) Inc., ZTE (USA) Inc., and ZTE Corporation (collectively, “ZTE”).

The Accused Products comprise ZTE products running the Android mobile operating system and manufactured, used, or sold during and after 2011. For example, the Accused Products comprise the following Android-based phones and tablets: Adamant, Agent, Anthem 4G, Aspect, Avail / Merit, Avid 4, Avid 4G, Avid Plus, Avid Plus / Avid 828, Avid Trio / ZFive 2, Awe / Emblem, Axon, Axon 7 Max, Axon 7 mini, Axon Elite, Axon Lux, Axon M, Axon Max, Axon Mini, Axon Pro / Axon, Blade A1, Blade A2, Blade A2 Plus, Blade A2S, Blade A3, Blade A910, Blade C, Blade D Lux, Blade D2, Blade D6, Blade E, Blade Force, Blade G2, Blade L, Blade L3, Blade L5 Plus, Blade Max 3, Blade Max Blue, Blade Q, Blade Q Maxi, Blade Q Mini, Blade Qlux 4G, Blade S6, Blade S6 Lux, Blade S6 Plus, Blade S7, Blade Spark, Blade V580, Blade V6, Blade V7, Blade V7 Lite, Blade V7 Max, Blade V7 Plus, Blade V8, Blade V8 Lite, Blade V8 Mini, Blade V8 Pro, Blade Vantage, Blade Vec 3G, Blade Vec 4G, Blade X, Blade X Max, Blade X3, Blade X5, Blade X9, Blade Z Max, C78, C79, C88, CAPTR II / A210, Chorus, Citrine, Compel, Concord / Midnight, Concord II, CYMBAL LTE, Cymbal LTE (Verizon), Cymbal Z-320, Cymbal LTE, Cymbal-C LTE, Cymbal-G LTE, CYMBAL-T, Cymbal-T LTE, Engage, Engage LT / Engage MT, Essence C70, F160, Fanfare, Fanfare 2, Fanfare 3, Flash, Force, Fury / Director, Geek, Grand Memo II LTE, Grand S Flex, Grand S II, Grand S Pro, Grand S3, Grand X, Grand X 3, Grand X 4, Grand X 4, Grand X Max 2 / Imperial MAX, Grand X Max+, Grand X Quad Lite, Grand XMax, Groove, Hawkeye, Imperial, Imperial II, Jasper LTE, Kis 3 Max, Kis Flex, Majesty, Majesty Pro, Majesty Pro LTE, Majesty Pro Plus LTE, Maven 2, Maven 2 / Sonata 3, Maven 3 / Overture 3, Max, MAX Blue LTE, MAX XL, Max+, Memo, Midnight PRO LTE, MSGM8 II, N919D, Nubia M2, Nubia M2 Lite, Nubia M2 Play, Nubia My Prague, Nubia N1, Nubia N1 lite, Nubia N1, Nubia N2, Nubia Prague S, Nubia X6, Nubia Z11, Nubia Z11 Max, Nubia Z11 mini, Nubia Z11 mini S, Nubia Z17, Nubia Z17 Lite, Nubia Z17 mini, Nubia Z17 miniS, Nubia Z17S, Nubia Z7, Nubia Z7 Max, Nubia Z7 mini, Nubia Z9 Classic, Nubia Z9 Elite, Nubia Z9 Exclusive, Nubia Z9 Max, Nubia Z9 mini, Obsidian, Open, Open C, Open II, Overture, Overture 2 / Maven, Prelude / Avail 2, Prelude+, Prestige, Prestige 2, Q519T, Quartz, R225, Reef, Render, Salute, Savvy, Score M / Score, Small Fresh 4, Small Fresh 5, Solar, Sonata / Radiant, Sonata 2 / Paragon, Sonata 3, Source, Speed, Star 1, Star 2, Tempo, Tempo X, TXXM8 3G, V3 Energy Edition, V3 Extreme Edition, V3 Youth Edition, V5, V870, Valet, Vital / Supreme, Warp, Warp 4G, Warp 7, Warp Elite, Warp Sequent, Warp Sync, Whirl, Z221, Z222 / Z223, Z331, Z431 / Altair, Z432 / Altair, Z667 / Zinger / Prelude 2 / Flame / Whirl 2, Z998 / Unico LTE, ZFive 2 LTE, ZFive L LTE, Zinger, ZMax, Zmax 2, ZMAX 2 (Unlocked), ZMAX Champ LTE, ZMAX Grand / Champ / Avid 916, ZMAX GRAND LTE,

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

ZMax Pro, and any variants thereof. AGIS reserves the right to amend this list of Accused Products as discovery progresses. For example, AGIS reviewed Android-based products from Android-based handset manufacturers, including three ZTE phones (serial numbers 329F75623FA5; 329F7562388B; and 329F75624913) which are available for inspection at ZTE’s request. For example, the Accused Products comprise ZTE products, including but not limited to the phones and tablets as described herein, running the following versions (and all intervening updates and sub-versions) of the Android mobile operating system: Android 2.3, 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, 5.1, 6.0, 7.0, 7.1, 8.0, and 8.1. For example, the Accused Products comprise ZTE products, including but not limited to the phones and tablets as described herein, running any versions of the following Android-based applications and/or software: Android Device Manager, Find My Phone, Find My Device, Google Latitude, Google Plus, Google Hangouts, Google Maps, Google Assistant, Google Search, Google Messages, Android Messenger, Google Allo, Google Duo, Gmail, and Google Chrome. For example, the Accused Products comprise ZTE products, including but not limited to the phones and tablets described herein, participating in any networks and/or services related to the execution and/or use of the Android mobile operating system versions and Android-based applications and/or software described herein.

AGIS does not concede that any claims of the ’055 Patent that are not listed below are not infringed by the identified products. Moreover, the citations to certain documents and other information below are intended to be exemplary only and in no way foreclose AGIS from citing or relying on additional documents, information, source code, and/or testimony at a later time. These contentions are preliminary in nature, and an analysis of ZTE’s products, internal documentation, source code, and/or testimony from relevant witnesses may more fully and accurately describe the infringing features of its accused products. Accordingly, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions once such additional information is made available to AGIS. Furthermore, AGIS reserves the right to supplement, correct, modify, and/or amend these contentions as discovery in this case progresses; in view of the Court’s claim construction order(s); in view of any positions taken by ZTE, including but not limited to positions on claim construction, invalidity, and/or non-infringement; and in connection with the preparation and exchange of expert reports.

| US9408055B2 | ZTE |
|---|--|
| I[P] A method comprising: performing by a first device: | ZTE performs either directly or indirectly, induces others to perform, and/or contributes to the performance of each step of this method as set forth below. The Accused Products meet the claim limitations by providing device-location tracking features such as |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.

Google Maps Share Location

Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps' Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.

See, e.g., <https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/>; <https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html>; <https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html>; <http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html>; <https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Control within reach, even when your device isn't

One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching Find My Device as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit android.com/find or check out the app.

See, e.g., <https://www.blog.google/products/android/google-play-protect/>

Link your phone to Google



You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.

Link your Android phone

Step 1: Update the Google app

- 1. On your phone, go to the Google app page on the Play Store.
- 2. Tap **Update**.



Step 2: Turn on Google Now

- 1. On your phone, open the Google app .
- 2. At the top left, tap Menu  > **Settings** > **Now cards**.
- 3. Turn on **Show cards**.
- 4. Turn on **Show notifications**.

Step 3: Turn on Web & App Activity

- 1. Visit the **Account History** page.
- 2. Make sure the switch is on (green).

Step 4: Sign in to your browser

- 1. On your phone, open the Google app .
- 2. At the top left, tap the Menu .
- 3. At the top left, you'll see the email address you use for the Google app.
- 4. Visit www.google.com  on your computer.
- 5. If you aren't signed in already, click **Sign in** in the top right corner of the page.
- 6. Sign in using the Google Account you use for the Google app.

Step 5: Send information to your phone

- 1. Do one of the searches below, like **note to self**, or **send directions to my phone**.
- 2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE |
| | https://support.google.com/websearch/answer/6128427 |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

What you can do once your phone is linked

Find my phone

You can get the current location of your phone if you can't find it.

1. On your computer's browser, search on www.google.com for **find my phone**.
2. If your phone is turned on and connected to the Internet, you'll see your phone's location.
3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking **Ring**. You can stop the ringing from your phone when you find it.

Tip: You can also find your missing phone using the **Android Device manager** which lets you find your device or remotely ring, lock, or erase it.

Send directions to my phone

Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.

1. On your computer's browser, search on www.google.com for **send directions to my phone**.
2. Enter in your destination.
3. Click **Send directions to your phone**.
4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.

Send a note to my phone

1. On your computer's browser, search on www.google.com for **send a note to my phone**.

2. Type your note in the box.
3. Click **Send note to your phone**.
4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.

Set an alarm

1. On your computer's browser, search on www.google.com for **set an alarm**.

2. Choose the time you want the alarm to go off.
3. Click **Set an alarm on your phone**.
4. An alarm will now be set on your phone's Clock app.

Set a reminder

1. On your computer's browser, search on www.google.com for **set an reminder**.

2. Type what you want to be reminded about, and either when or where you want the reminder to go off.
3. Click **Remind me on my devices**.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://support.google.com/websearch/answer/6128427>

Share your location using Google Maps

You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.

<https://support.google.com/plus/answer/3302509?hl=en&co=GENIE.Platform%3DAndroid&oco=1>

Location

Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.

- **Mode** – Sets the how your current location information is determined.
- **Recent Location Request** – Displays applications and services that have recently requested your location information.
- **Camera** – Checkmark to tag photos or videos with their locations.
- **Google Location History** – Allows you to view and manage your Google location history.

Accounts & sync

Use the Accounts & sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.

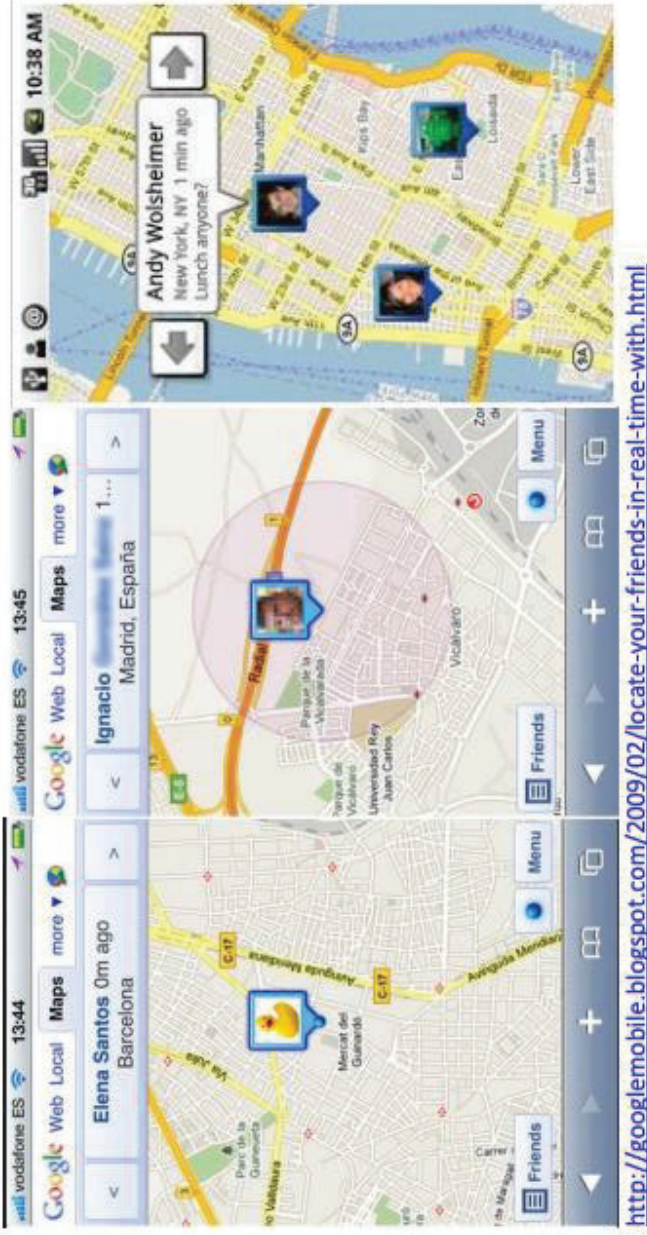
Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch **Add account** to add new account.

.. ...

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.

ZTE makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as the Accused Products described herein as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example ZTE Accused Products that perform each step of this method as set forth below.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE phones
 mob.org * Mobile phones and smartphones catalogue

Sort by: Popularity Date Price

HTC
 LG
 Samsung
 Motorola
 Fly
 Sony-Ericsson
 Apple
 Nokia
 Wobisado
 Vertu
 BenQ-Siemens
 Sagem
 Alcatel
 Philips

All brands

ZTE Rapido
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 4.5 inch.
 Android 4.1.2

ZTE Grand S3
 Mobile phone
 2015 year
 Touchscreen: 1080 x 1920
 5.5 inch.
 Android 4.4

ZTE Geek 2
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE V5s
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE Grand X Quad
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.1

ZTE ZMAX
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.4.2

ZTE Grand Memo Lite
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.3

https://mob.org/phone/zte/page_3/sort_date_down/

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | |
|--------------------|------------|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> |  <p>AT&T TREK™ 2 HD</p> <p>ZTE ZPAD 8</p> <p>AT&T PRIMETIME™</p> <p>https://www.zteusa.com/products/tablets</p> | <p>[1A] obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices.</p> <p>For example, the Accused Products include a Contacts app to access contact information for second users using respective second devices.</p> |
|--------------------|------------|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Phone calls

How to make calls



There are many ways to make a call with your phone, and they're all easy to do.

Calling from the dialer



1. From the home screen, tap .
2. Enter the phone number with the on-screen keypad. Tap  to delete wrong digits.
3. Tap  to place the call.

Tip: To make international calls, press and hold **0+** to enter the "+".

Calling from your contacts


1. From the home screen, tap .
2. Swipe your finger up or down to scroll through the contacts list and tap  next to the contact you want to call.

Tips:

- You can search for a contact by tapping  and entering the contact name.
- You can also access your contacts by tapping  > **Favorites**.



People

You can add contacts on your phone and synchronize them with the contacts in your Google account or other accounts that support contact syncing. To see your contacts, tap  on the home screen. From there, you can tap the tabs on the top to quickly switch to **Groups**, or **Favorites**.

Importing and exporting contacts

You can import/export contacts from/to your SIM card, phone storage, or microSDHC card. This is especially useful when you need to transfer contacts between different devices. You can also quickly share your contacts using Bluetooth, Email, Messaging, etc.

Importing contacts from the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Manage SIM card contacts**.
2. If you have added contact accounts other than the phone, select an account in which to save the contacts.
3. Tap the contacts you want to import one by one, or tap  > **Import all**.

Importing contacts from a microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Import from phone storage**.
2. If you have added contact accounts other than the phone, select an

Calling from your call history

1. From the home screen, tap  > **Call log**.
2. Tap  next to the number you want to call.

Calling from a text message

If a text message contains a phone number that you want to call, you can make the call while viewing the text message.

1. From the home screen, tap .
2. Tap the conversation and then find the message that contains the phone number you need.
3. Tap the number and then tap .

Using speed dial

Press and hold the **1-9** key from the dialer to call the corresponding speed dial number.

The **1** key is reserved for your voicemail.

Assigning a speed dial key

1. From the home screen, tap .
2. In the Phone tab, tap  > **Speed dial setting**.
3. Tap a speed dial key and tap **Set speed dial contact**.
4. Select a contact from the contact list.

Note: You can also set your speed dial keys from the dialer. Press and hold the **2-9** key and then tap **OK**.


3. Select the vCard file(s) in the microSDHC card or the phone storage and tap **OK**.

Note: If the microSDHC card is not installed in the phone, you can import vCard file(s) in the phone storage.

Exporting contacts to the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Export to SIM card**.
2. Select the contacts you want to export and then tap .

Exporting contacts to the microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Export to phone storage**.
2. The phone will prompt you with the name of the vCard file and the directory in which the file will be saved. Tap **OK** to create the file.

Note: If the microSDHC card is not installed in the phone, you can export vCard file(s) into the phone storage.

Sharing contact information

1. From the Contacts screen, tap  > **Import/export** > **Share visible contacts**.
2. Choose how to share the contacts. Options depend on the applications and services installed.

Creating a contact

1. From the Contacts screen, tap  to add a new contact.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

2. Tap the account field near the top of the screen to choose where to save the contact. If a sync account is selected, the contacts will be synced automatically with your account online.
3. Enter the contact name, phone numbers, email addresses, and other information.
4. Tap **DONE** to save the contact.

Adding a contact to Favorites

You can add the contacts you use frequently to Favorites so that you can find them quickly.

1. From the Contacts screen, tap the contact you want to add to **Favorites**.
2. Tap  next to the contact's name.

Searching for a contact

1. Tap the search field above the contacts list.
2. Enter the contact name you want to search for. Matching contacts will be listed.

Joining contacts


As your phone synchronizes with multiple online accounts, you may see duplicate entries for the same contact. You can merge all the separate information of a contact into one entry in the Contacts list.

1. From the home screen, tap .
2. Tap a contact to display the contact's details.
3. Tap  > **Edit** >  > **Join**.

Entering text

You can enter text using the onscreen keyboard. Some apps open it automatically. In others, you open it by tapping where you want to type. You can also enter text by speaking with the Google voice typing feature. Tap  to hide the onscreen keyboard.

Changing input methods

1. When you use the onscreen keyboard to enter text, the icon  appears on the notification bar.
2. Open the notification panel and tap **Choose input method**.
3. Select an input method you need.

Google keyboard

The Google keyboard provides a layout similar to a desktop computer keyboard. Turn the phone sideways and the keyboard will change from portrait to landscape.

To use the landscape keyboard, tap the **Auto-rotate screen** check box in  > **System settings** > **Accessibility**.

Note: The landscape keyboard is not supported in all applications.

4. Tap the contact whose information you want to join with the first entry.
 5. Tap **DONE**.
- The information from the second contact is added to the first, and the second contact is no longer displayed in the contacts list. You can repeat these steps to join another contact to the main contact.

Separating contact information

If contact information from different sources was joined in error, you can separate the information back into individual contacts on your phone.











1. From the home screen, tap .
2. Tap a contact you have merged and want to separate.
3. Tap  > **Edit** >  > **Separate**.
4. Tap **OK** to confirm.

Creating a new group

1. From the Contacts screen, tap .
2. Tap .
3. If you have added contact accounts other than the phone, choose an account for the new group.
4. Enter the group name and tap **DONE**.
5. Tap  and select the contacts you wish to be the group members.
6. Tap .

To send messages to the group members, you can tap a group and then tap  > **Send group message**.



- Tap the alphabetic keys to enter letters. Press and hold the keys to enter associated accented letters or numbers. For example, to enter É, press and hold  and the available accented letters and number 3 appear. Then slide your finger to choose **E**.
- Tap  to use uppercase or lowercase letters. This key also changes to indicate the current case you are using:  for lowercase,  for uppercase, and  when locked in uppercase. Press and hold or double-tap  to lock the keyboard in uppercase.
- Tap  to delete any text you have entered.
- Tap  to select numbers and symbols. You can then tap  to access more.
- Tap  to enter miniature icons.
- Tap  to use Google's networked voice input.
- Press and hold  to change the input language or the Google keyboard settings.

In another example, the Accused products run Android Messages and Google Hangouts which both access

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p>ZTE</p> <p>contact information for second users using respective second devices.</p> <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the `ContactsContract.Data` table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the `ContactsContract.RawContacts` table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the `ContactsContract.Contacts` table represents an aggregate of one or more `RawContacts` presumably describing the same person. When data in or associated with the `RawContacts` table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- `ContactsContract.Groups`, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- `ContactsContract.StatusUpdates`, which contains social status updates including IM availability.
- `ContactsContract.AggregationExceptions`, which is used for manual aggregation and disaggregation of raw contacts
- `ContactsContract.Settings`, which contains visibility and sync settings for accounts and groups.
- `ContactsContract.SyncState`, which contains free-form data maintained on behalf of sync adapters
- `ContactsContract.PhoneLookup`, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a `ContactsContract.Data` row that is linked to the raw contacts_id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for `emilyd@gmail.com` (the raw contact row for Thomas Higginson associated with the Google account `emilyd@gmail.com`) has a home email address of `thigg@gmail.com` and a work email address of `thomas.higginson@gmail.com`, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the `ContactsContract.Data` table. To help manage this, the `ContactsContract.Data` table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

| Task | Action | Data | MIME type | Notes |
|----------------------------|-------------|---|-----------|---|
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

59  /** Show all phone numbers and pick them when clicking */
60  public static final int ACTION_PICK_PHONE = 90;
61
62  /** Show all postal addresses and pick them when clicking */
63  public static final int ACTION_PICK_POSTAL = 100;
64
65  /** Show all postal addresses and pick them when clicking */
66  public static final int ACTION_PICK_EMAIL = 105;
67
68  /** Show all contacts and create a shortcut for the picked contact */
69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110;
70
71  /** Show all phone numbers and create a call shortcut for the picked number */
72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120;
73
74  /** Show all phone numbers and create an SMS shortcut for the picked number */
75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130;
76
77  /** Show all contacts and activate the specified one */
78  public static final int ACTION_VIEW_CONTACT = 140;
79
80  /** Show contacts recommended for joining with a specified target contact */
81  public static final int ACTION_PICK_JOIN = 150;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY  = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




US9408055B2

ZTE

Send & receive text messages in Android Messages





You can send and receive text messages with friends and contacts on Android Messages.

Start a conversation

1. Open the Android Messages app .
2. Tap Compose .
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next .

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329

See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](#) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
2. Tap Menu  > **Create label**.
3. Enter a label name and tap **OK**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
- **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

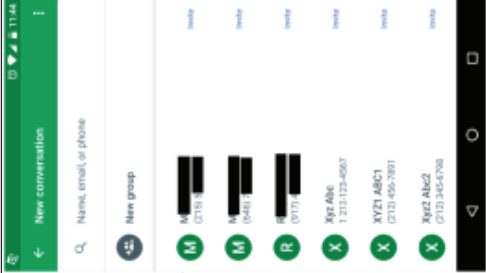
You can send and receive messages with one person or multiple people.

COMPUTER ANDROID IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0



Contact someone

You can call, email, or send text messages to your contacts.

1. Open your device's Contacts app.
2. Tap a contact in the list.
3. Choose an option:
 - Call
 - Email
 - New message

https://support.google.com/nexus/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a conversation

1. Open the Android Messages app
2. Tap Compose
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329
<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

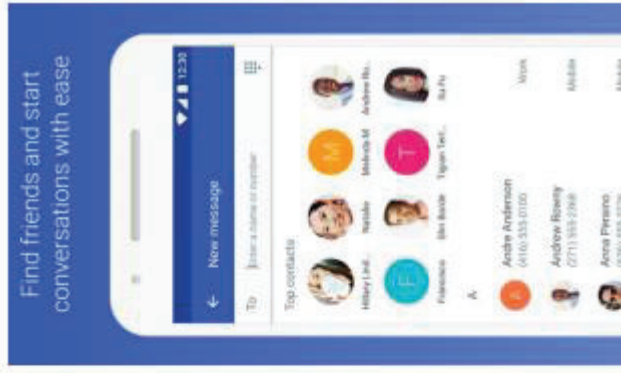


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add + > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

1. Open the Hangouts app.
2. At the bottom, tap Add + > New conversation > New group.
3. Enter and select the names, phone numbers, or email addresses of people in your group.
4. Tap Done.

https://support.google.com/hangouts/answer/3111943?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Contact someone

You can call, email, or send text messages to your contacts.

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Choose an option:
 - Call 
 - Email 
 - New message 

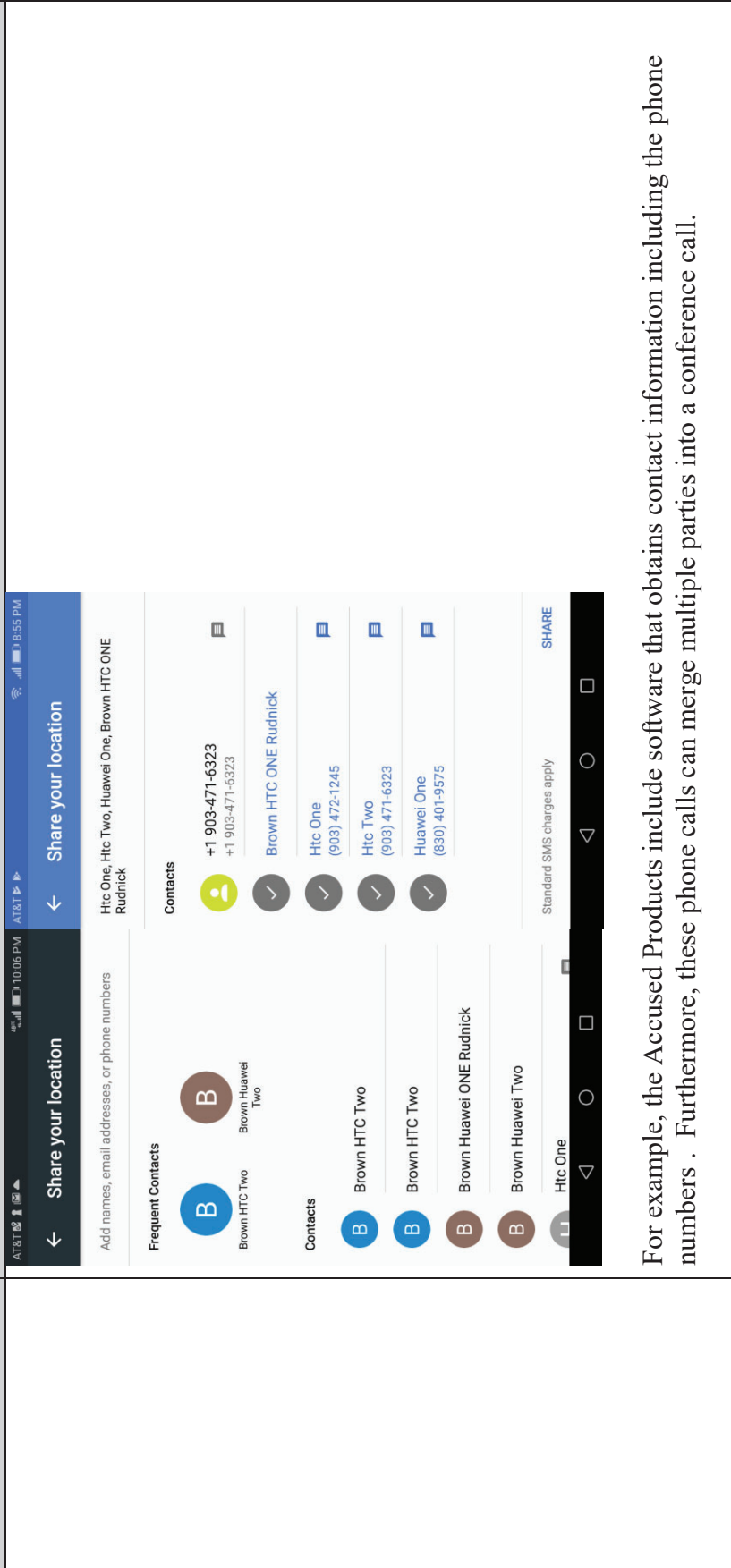
https://support.google.com/hexus/answer/611873?hl=en&ref_topic=6118711

Exemplary Google Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

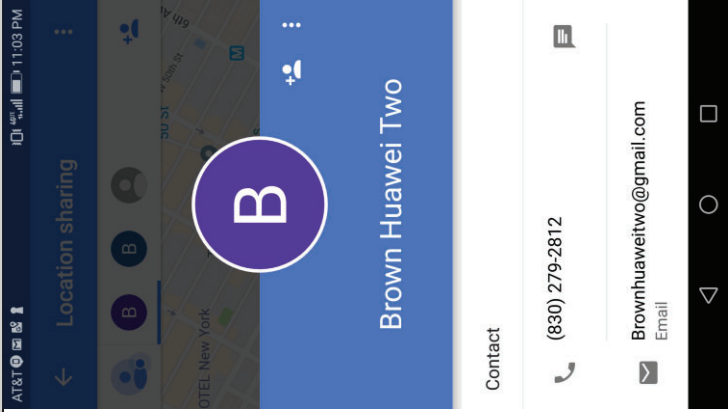
| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p>  |
| <p>[1B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using respective telephone numbers to send, from the first device to the first device to the</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See also, e.g., 1[P]-[1A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p> | <p>ZTE</p> <p>Using Wi-Fi Direct</p> <p>Wi-Fi Direct allows Wi-Fi devices to connect to each other without the need for wireless access points (hotspots).</p> <p>Connecting to another device via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. From the home screen, tap [REDACTED] > System settings > Wi-Fi. 2. If Wi-Fi is off, slide the Wi-Fi switch to the On position. 3. Tap [REDACTED] > Wi-Fi Direct. Your phone will search for other devices enabled with Wi-Fi Direct connections. 4. Tap a device name under Peer Devices to connect with it. The other device will receive a Wi-Fi Direct connection prompt and need to accept the request for connection. Both devices may need to enter a common PIN. If prompted, tap Connect. 5. Once connected, the device is displayed as "Connected." <p>Sending data via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. Open the appropriate application and select the file or item you want to share. 2. Select the option for sharing via Wi-Fi Direct. The method may vary by application and data type. 3. Tap a device the phone has connected with or wait for it to search for new devices and tap one of them. <p>Receiving data via Wi-Fi Direct</p> <p>When an attempt to transfer data via Wi-Fi Direct is received, you can see a notification in the status bar. Tap Accept to start receiving the data. Received files are stored automatically in a dedicated folder (Wi-FiShare, for instance) in the phone storage or microSDHC directory. You can access them with the File Manager app.</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Messaging

You can use Messaging to exchange text messages (SMS) and multimedia messages (MMS).

Message box



Instead of an inbox and outbox, your phone organizes all messages you sent and received into one box, where messages exchanged with the same number are grouped into one message thread on the Messaging screen. You can tap a thread to see the conversation you have had with someone.

Message threads are sorted in chronological order with the latest one on top.

Opening the messaging screen

From the home screen, tap .

The Messaging screen opens, where you can create a new message, search for messages, or open an ongoing message thread.

- Tap  to write a new text or multimedia message.
- Tap  to search for a message with keywords.
- Tap an existing message thread to open the conversation you've had with a certain number.

Sending a message





1. From the messaging screen, tap  at the bottom.
2. Add recipients by one of the following ways:
 - Tap the **To** field and manually enter the recipient's number or the contact name. If the phone presents a few suggestions, tap the one you want to add.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



- Select recipients from your contacts by tapping .
- 3. Tap the **Type message** field and enter the content of your text message.
- 4. If you want to send a multimedia message, tap the paper clip icon  to attach a file or a slideshow to the message.
- 5. Tap  to send your message.

Notes:

- You can also include email addresses as recipients for multimedia messages.
- Do not add any attachment if you want to send a text message. Otherwise you may be charged for a multimedia message.


Replying to a message

Messages you receive are appended to existing threads of the same number. If the new message comes from a new number, a new thread is created.


1. From the Messaging screen, tap the thread that has the message you want to reply to.
2. Type your reply in the text box at the bottom. You can tap the icon  if you want to reply with an MMS.
3. Tap  to send your message.

Forwarding a message

1. From the Messaging screen, tap the thread that has the message you want to forward.
2. Press and hold the message.
3. Tap **Forward** in the menu that opens.

4. Enter a recipient for the message and edit the content if you want.
5. Tap  to send your message.

Changing message settings

The phone's message settings are pre-configured for you to use immediately. To change them, tap  > **Settings** from the Messaging screen.

Storage settings:

- **Delete old messages:** Delete old messages as limits are reached.
- **Text message limit:** Set the maximum number of text messages allowed in a single thread.
- **Multimedia message limit:** Set the maximum number of multimedia messages allowed in a single thread.

Text (SMS) message settings:

- **Manage SIM card messages:** Manage the messages stored on your SIM card.

- **Service Center:** Enables you to view and edit the service center number.

Multimedia (MMS) message settings:

- **Auto-retrieve:** Automatically download multimedia messages.

Display settings:

- **Bubble and background:** Set the appearance of the messaging bubbles and background.

Notification settings:


- **Notifications:** Show message notifications in the status bar.
- **Choose ringtone:** Choose a ringtone for your incoming messages.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Email

From the home screen, tap . You can receive and send emails from your webmail or other accounts using POP3 or IMAP, or access your Exchange ActiveSync account for your corporate email needs.

Setting up the first email account

1. When you open **Email** for the first time, enter your email address and password.
2. Tap **Next** to let the phone retrieve the network parameters automatically.
Note: You can also enter these details manually by tapping **Manual setup** or when automatic setup fails.
3. Follow the on-screen instructions to finish the setup.

Your phone will show the inbox of the email account and start to download email messages.

Checking your emails

Your phone can automatically check for new emails at the interval you set when setting up the account.

You can also check new emails manually by tapping  in any of the email account's boxes. Tap **Load more messages** at the bottom of the email list to download earlier messages.

Responding to an email

You can reply to or forward a message that you receive. You can also delete messages and manage them in other ways.

Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.

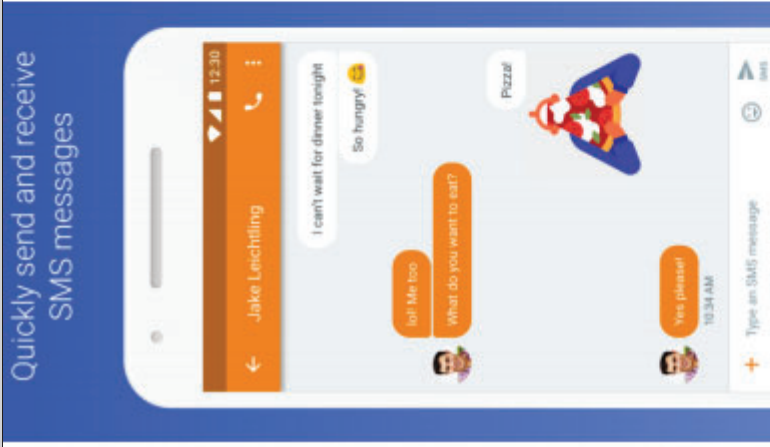
• **Enhanced features:** On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.

<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER **ANDROID** IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app .
2. At the bottom right, tap Add  > New Conversation .
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send .

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

1. Open the Hangouts app .
2. At the bottom right, tap Add .
3. Choose **New SMS**.
4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.
5. Tap the number or contact.
6. Tap Continue .
7. Type your message and tap Send .

<https://support.google.com/hangouts/answer/3441321?hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google Maps Share Location

Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.

Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group

Exemplary Support for Google Maps:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap  > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
|---------------------------|---|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>



Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap **More**  **> Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap **More**  **> Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap **Menu**  **> Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon **> More**  **> Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap **More** .
4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD



Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform.%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list

If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
 2. Tap **Lists**.
 3. Tap on the list you want to follow > More  > **Follow**.
- https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1

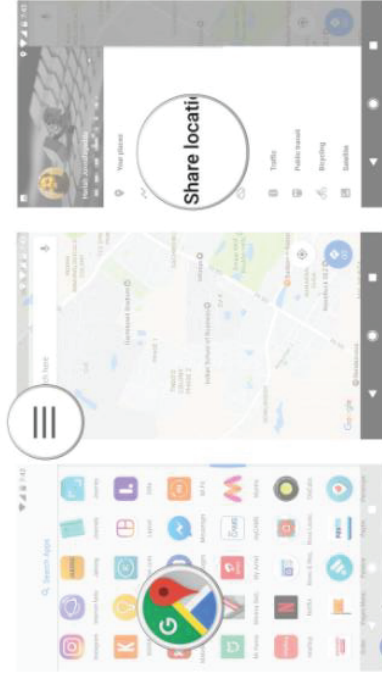
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.

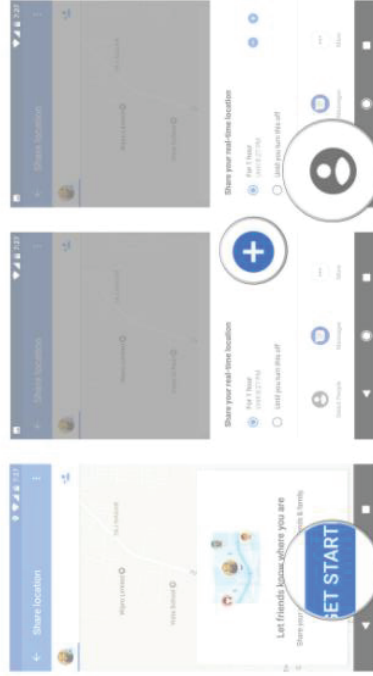


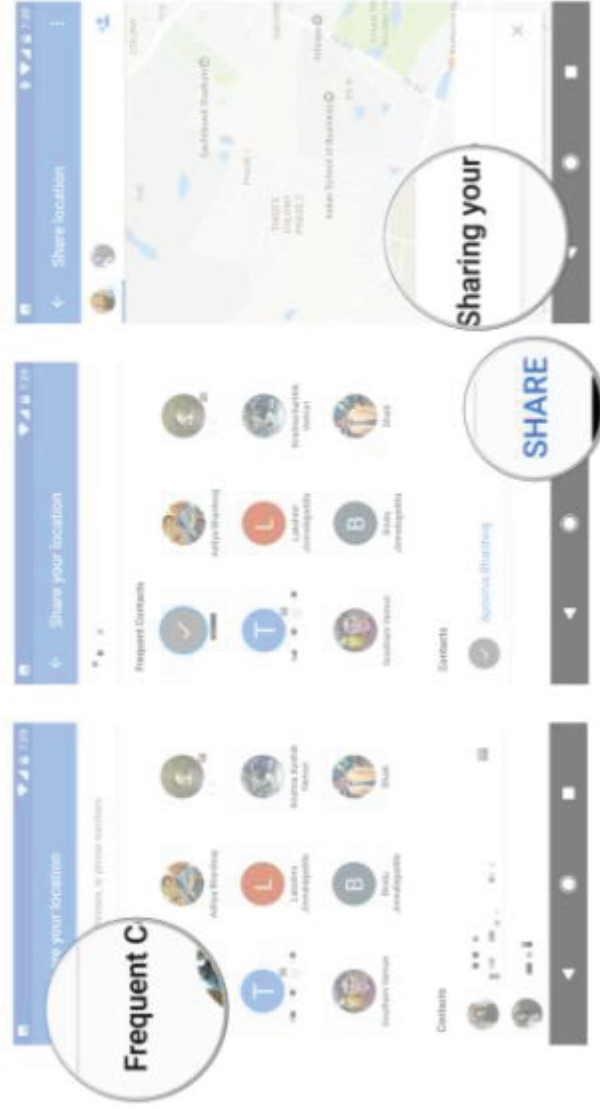
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://www.androidcentral.com/how-share-location-google-maps>

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

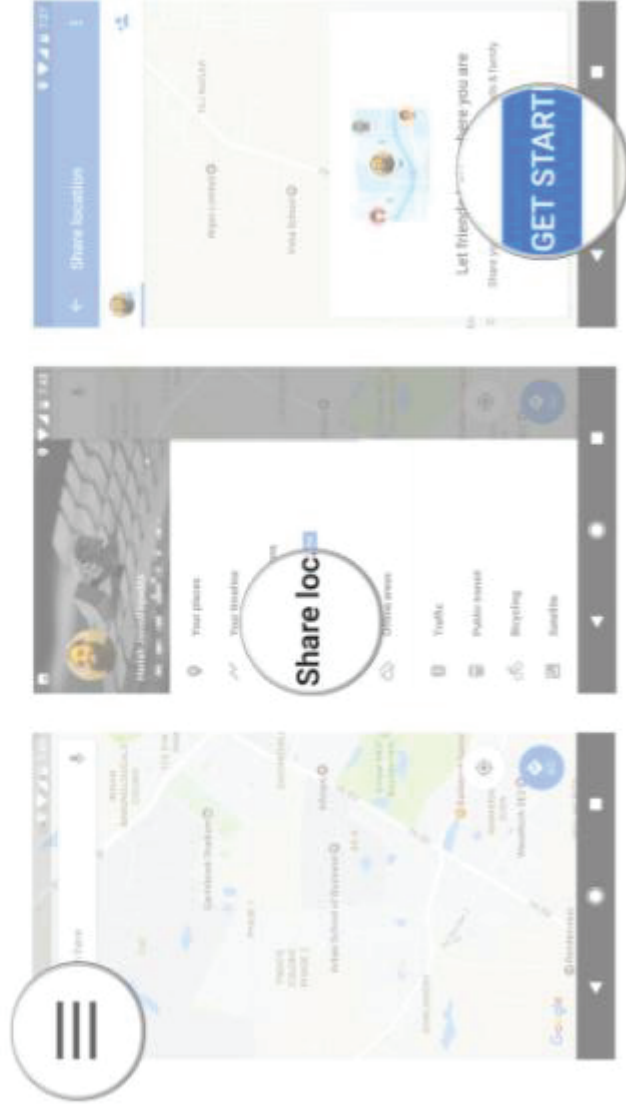
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

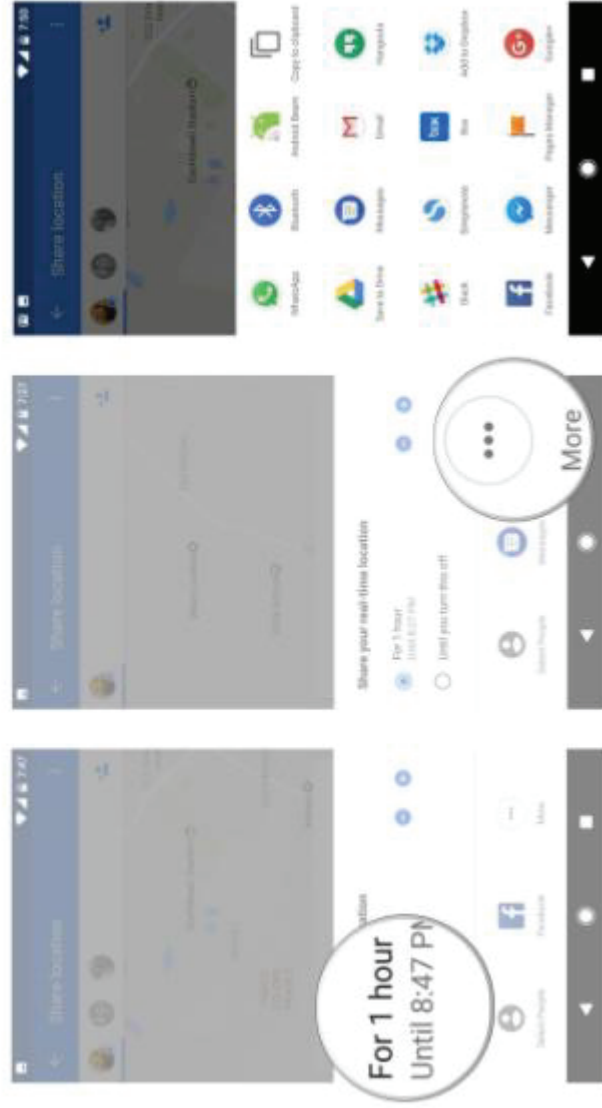
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

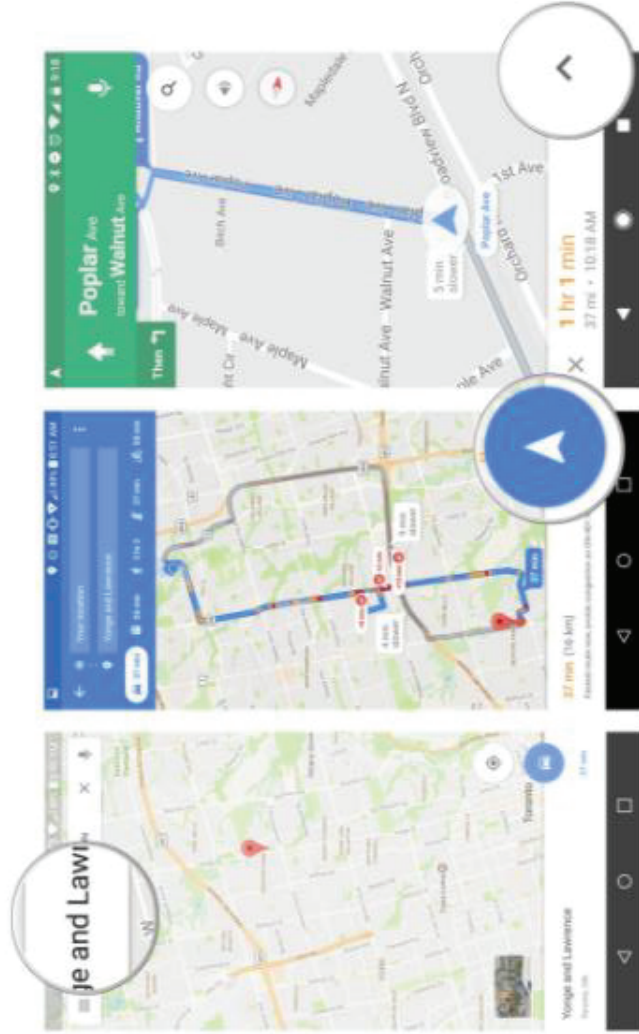
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



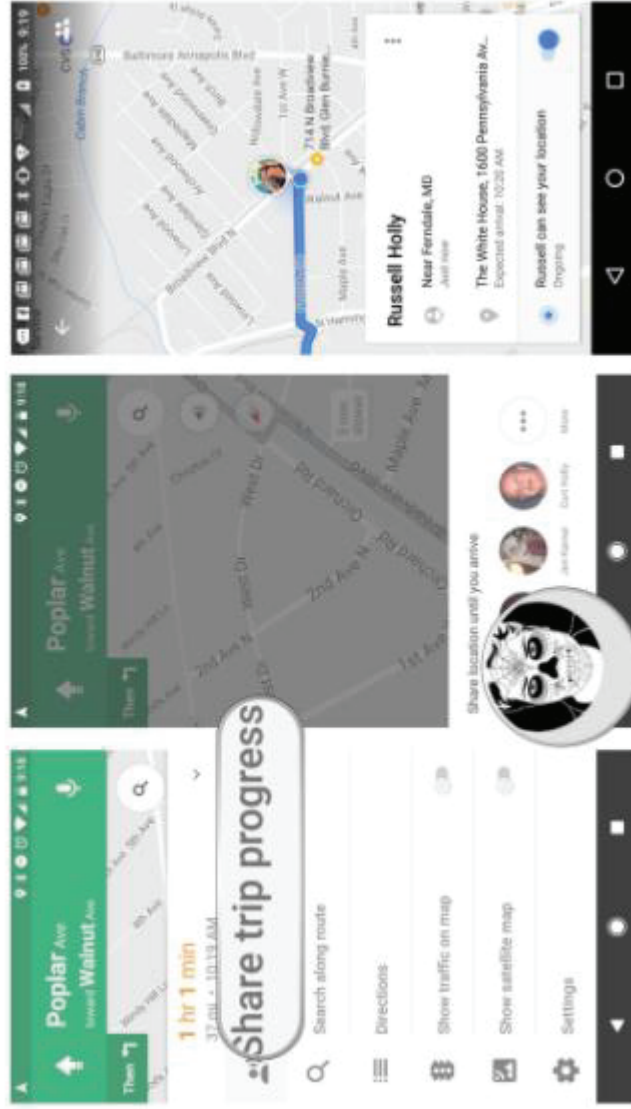
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



You can also stop sharing your location with someone before a trip ends.

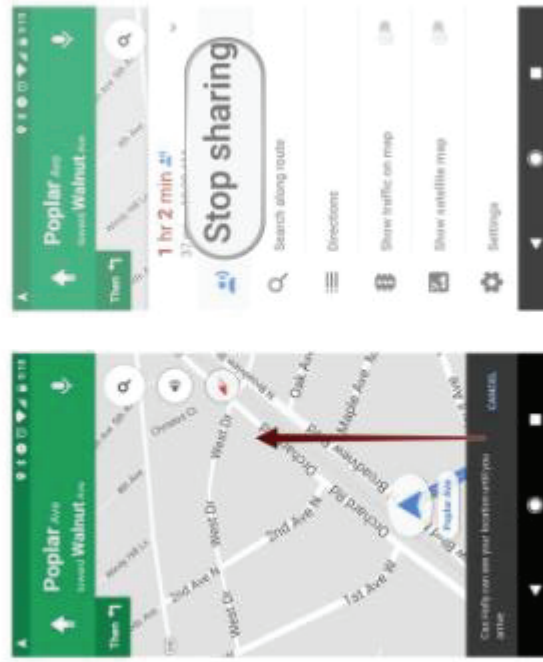
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?

<https://www.androidcentral.com/how-share-location-google-maps>





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

As shown below, a group may also be defined within Google Contacts.





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


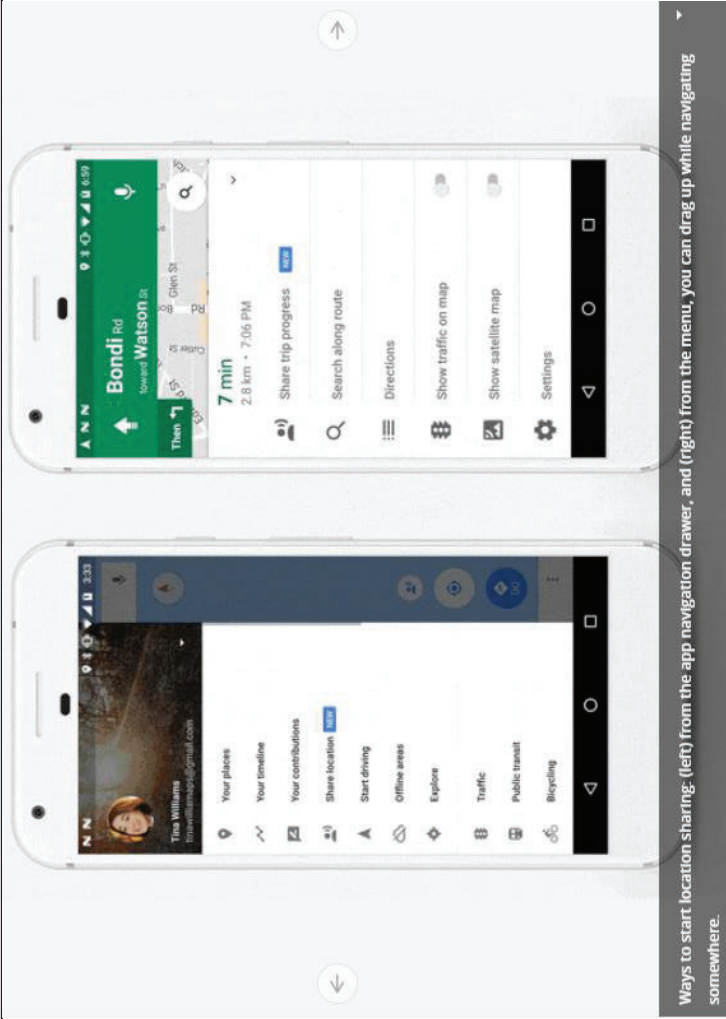
| | |
|-------------|--|
| US9408055B2 | ZTE <h3>Share your contacts</h3> <ol style="list-style-type: none">1. Open your device's Contacts app .2. Tap a contact in the list.3. Tap More  > Share.4. Choose how you want to share the contact. <p>https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

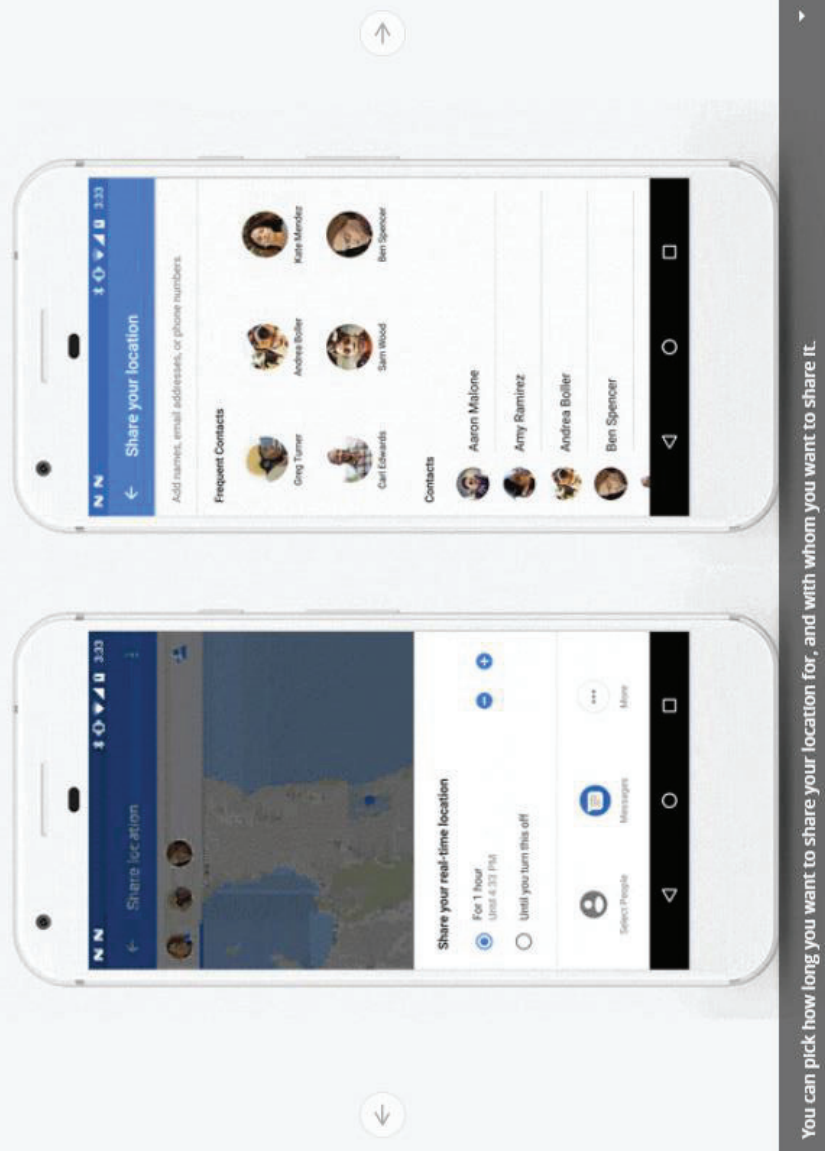


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

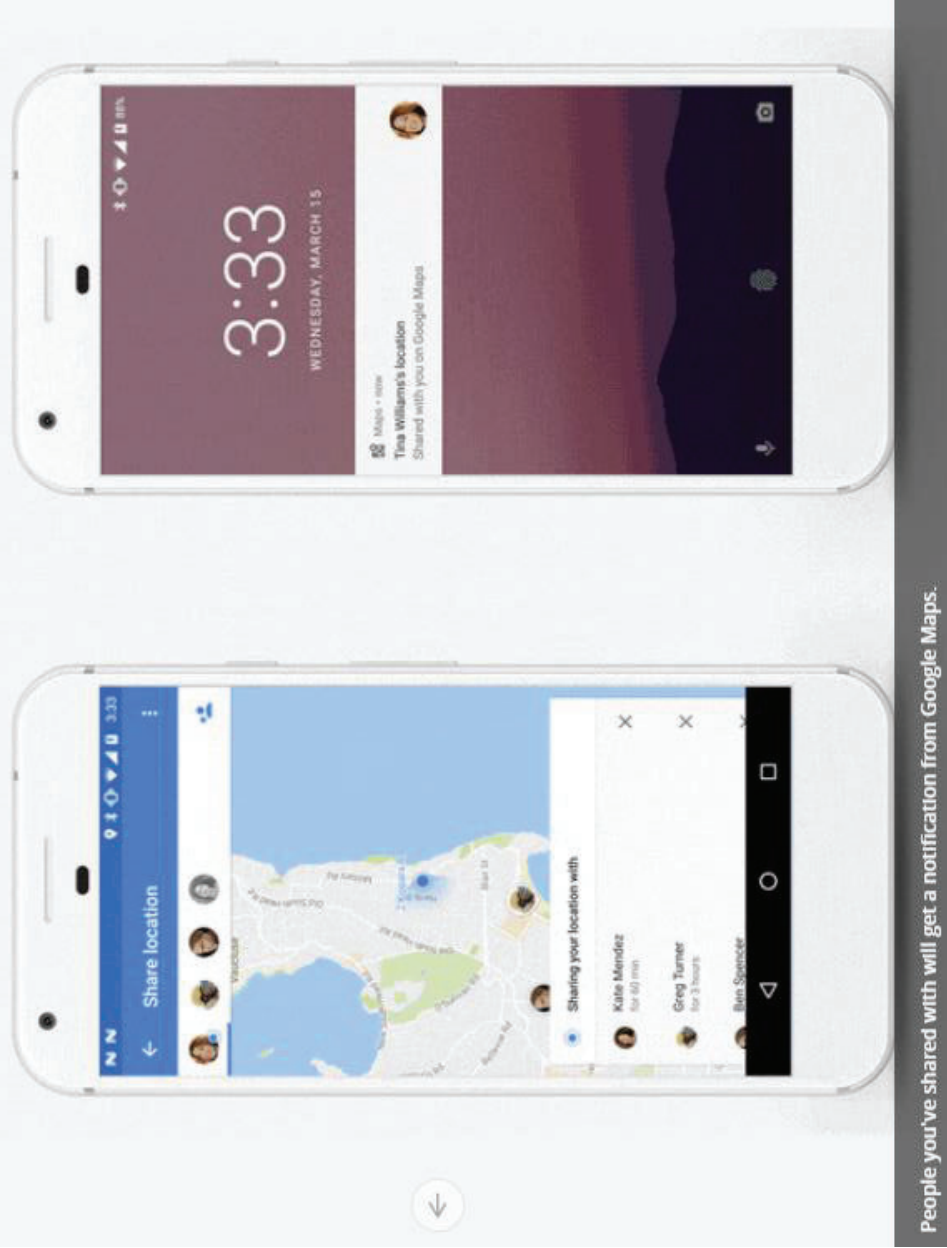


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



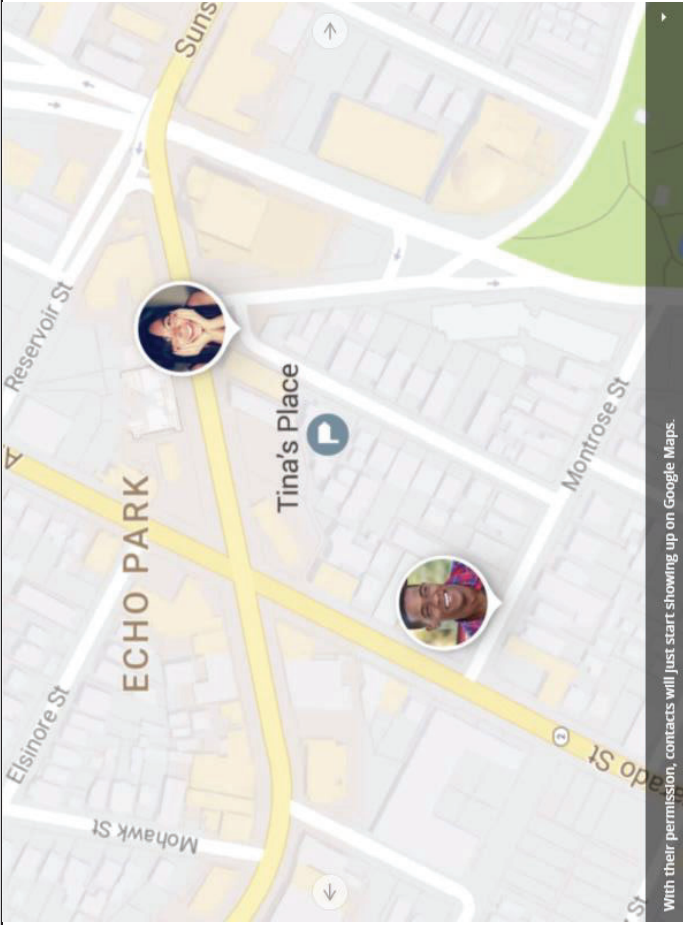
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



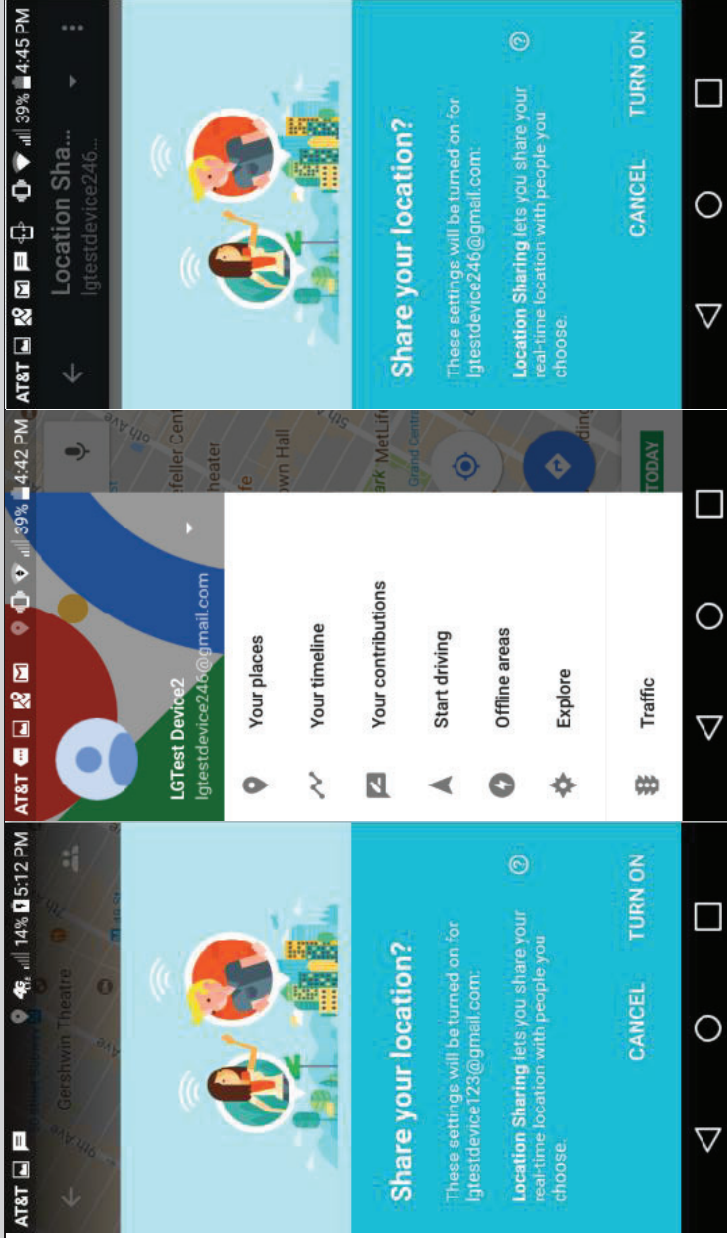
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Contacts Provider

The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.

This guide describes the following:

- The basic provider structure.
- How to retrieve data from the provider.
- How to modify data in the provider.
- How to write a sync adapter for synchronizing data from your server to the Contacts Provider.

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the `ContactsContract.Data` table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the `ContactsContract.RawContacts` table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the `ContactsContract.Contacts` table represents an aggregate of one or more `RawContacts` presumably describing the same person. When data in or associated with the `RawContacts` table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- `ContactsContract.Groups`, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- `ContactsContract.StatusUpdates`, which contains social status updates including IM availability.
- `ContactsContract.AggregationExceptions`, which is used for manual aggregation and disaggregation of raw contacts
- `ContactsContract.Settings`, which contains visibility and sync settings for accounts and groups.
- `ContactsContract.SyncState`, which contains free-form data maintained on behalf of sync adapters
- `ContactsContract.PhoneLookup`, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a `ContactsContract.Data` row that is linked to the raw contact's `_id` value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for `emilyd@gmail.com` (the raw contact row for Thomas Higginson associated with the Google account `emilyd@gmail.com`) has a home email address of `thigg@gmail.com` and a work email address of `thomas.higginson@gmail.com`, the `Contacts Provider` stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the `ContactsContract.Data` table. To help manage this, the `ContactsContract.Data` table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

| Task | Action | Data | MIME type | Notes |
|----------------------------|--------------------------|--|-----------|--|
| Pick a contact from a list | <code>ACTION_PICK</code> | One of: <ul style="list-style-type: none"> <code>Contacts.CONTENT_URI</code>, which displays a list of contacts. <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact. <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact. <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. |

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25  * Parsed form of the intent sent to the Contacts application.
26  */
27  public class ContactsRequest {
28
29      /** Default mode: browse contacts */
30      public static final int ACTION_DEFAULT = 10;
31
32      /** Show all contacts */
33      public static final int ACTION_ALL_CONTACTS = 15;
34
35      /** Show all contacts with phone numbers */
36      public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38      /** Show contents of a specific group */
39      public static final int ACTION_GROUP = 20;
40
41      /** Show all starred contacts */
42      public static final int ACTION_STARRED = 30;
43
44      /** Show frequently contacted contacts */
45      public static final int ACTION_FREQUENT = 40;
46
47      /** Show starred and the frequent */
48      public static final int ACTION_STREQUENT = 50;
49
50      /** Show all contacts and pick them when clicking */
51      public static final int ACTION_PICK_CONTACT = 60;
52
53      /** Show all contacts as well as the option to create a new one */
54      public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56      /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57      public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

59  /** Show all phone numbers and pick them when clicking */
60  public static final int ACTION_PICK_PHONE = 90;
61
62  /** Show all postal addresses and pick them when clicking */
63  public static final int ACTION_PICK_POSTAL = 100;
64
65  /** Show all postal addresses and pick them when clicking */
66  public static final int ACTION_PICK_EMAIL = 105;
67
68  /** Show all contacts and create a shortcut for the picked contact */
69  public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110;
70
71  /** Show all phone numbers and create a call shortcut for the picked number */
72  public static final int ACTION_CREATE_SHORTCUT_CALL = 120;
73
74  /** Show all phone numbers and create an SMS shortcut for the picked number */
75  public static final int ACTION_CREATE_SHORTCUT_SMS = 130;
76
77  /** Show all contacts and activate the specified one */
78  public static final int ACTION_VIEW_CONTACT = 140;
79
80  /** Show contacts recommended for joining with a specified target contact */
81  public static final int ACTION_PICK_JOIN = 150;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactFileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 458 private void configureFragments(boolean fromRequest) { 459 if (fromRequest) { 460 ContactListFilter filter = null; 461 int actionCode = mRequest.getActionCode(); 462 boolean searchMode = mRequest.isSearchMode(); 463 final int tabToOpen; 464 switch (actionCode) { 465 case ContactsRequest.ACTION_ALL_CONTACTS: 466 filter = ContactListFilter.createFilterWithType(467 ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468 tabToOpen = TabState.ALL; 469 break; 470 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471 filter = ContactListFilter.createFilterWithType(472 ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473 tabToOpen = TabState.ALL; 474 break; 475 case ContactsRequest.ACTION_FREQUENT: 476 case ContactsRequest.ACTION_STREQUENT: 477 case ContactsRequest.ACTION_STARRED: 478 tabToOpen = TabState.FAVORITES; 479 break; 480 case ContactsRequest.ACTION_VIEW_CONTACT: 481 tabToOpen = TabState.ALL; 482 break; 483 default: 484 tabToOpen = -1; 485 break; 486 } 487 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

release/src/com/android/contacts/activities/PeopleActivity.java

```

488     if (tabToOpen != -1) {
489         mActionBarAdapter.setCurrentTab(tabToOpen);
490     }
491
492     if (filter != null) {
493         mContactListFilterController.setContactListFilter(filter, false);
494         searchMode = false;
495     }
496
497     if (mRequest.getContactUri() != null) {
498         searchMode = false;
499     }
500
501     mActionBarAdapter.setSearchMode(searchMode);
502     configureContactListFragmentForRequest();
503 }
504
505 configureContactListFragment();
506
507 invalidateOptionsMenuIfNeeded();
508 }

```

https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-
 release/src/com/android/contacts/activities/PeopleActivity.java

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public class ProfileAndContactsLoader extends CursorLoader {
36
37     private boolean mLoadProfile;
38
39     private String[] mProjection;
40
41     private Uri mExtraUri;
42     private String[] mExtraProjection;
43     private String mExtraSelection;
44     private String[] mExtraSelectionArgs;
45     private boolean mMergeExtraContactsAfterPrimary;
46
47     public ProfileAndContactsLoader(Context context) {
48         super(context);
49     }

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID       = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI    = 3;
50         public static final int CONTACT_LOOKUP_KEY   = 4;
51     }
52
53
54     public static class GroupDetailQuery {
55         private static final String[] PROJECTION = new String[] {
56             Data.CONTACT_ID,           // 0
57             Data.PHOTO_URI,            // 1
58             Data.LOOKUP_KEY,           // 2
59             Data.DISPLAY_NAME_PRIMARY, // 3
60             Data.CONTACT_PRESENCE,     // 4
61             Data.CONTACT_STATUS,       // 5
62         };
63
64         public static final int CONTACT_ID           = 0;
65         public static final int CONTACT_PHOTO_URI    = 1;
66         public static final int CONTACT_LOOKUP_KEY    = 2;
67         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
68         public static final int CONTACT_PRESENCE_STATUS = 4;
69         public static final int CONTACT_STATUS        = 5;
70     }
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57     * Ensures that this is a valid group URI. If invalid, then an exception is
58     * thrown. Otherwise, the original URI is returned.
59     */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1->

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

[release/src/com/android/contacts/common/GroupMetaData.java](https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java)

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/**flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/**flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendTextMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57  *
58  * This class serves two purposes:
59  * - Process phone verification SMS messages
60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61  */
62  public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> ignoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1->

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

[release/src/com/android/messaging/receiver/SmsReceiver.java](https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java)

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsuUtil.isAtLeastL_MRI()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

228 @Override
229 public void onReceive(final Context context, final Intent intent) {
230     LogUtil.v(TAG, "SmsReceiver.onReceive " + intent);
231     // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver.
232     if (PhoneUtils.getDefault().isSmsEnabled()) {
233         final String action = intent.getAction();
234         if (OsUtil.isSecondaryUser() &&
235             (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) ||
236              // TODO: update this with the actual constant from Telephony
237               "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) {
238             postNewMessageSecondaryUserNotification();
239         } else if (!OsUtil.isAtLeastKLP()) {
240             deliverSmsIntent(context, intent);
241         }
242     }
243 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

52 * Class that sends chat message via MMS.
53 *
54 * The interface emulates a blocking send similar to making an HTTP request.
55 */
56 public class MmsSender {
57     private static final String TAG = LogUtil.BUGLE_TAG;
58
59     /**
60      * Send an MMS message.
61      *
62      * @param context Context
63      * @param messageUri The unique URI of the message for identifying it during sending
64      * @param sendReq The SendReq PDU of the message
65      * @throws MmsFailureException
66      */
67     public static void sendMms(final Context context, final int subId, final Uri messageUri,
68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException {
69         sendMms(context,
70                 subId,
71                 messageUri,
72                 null /* locationUrl */,
73                 sendReq,
74                 true /* responseImportant */,
75                 sentIntentExtras);
76     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *     it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113 String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? ("", proxy=" " + proxyHost + ":", proxyPort) : ""))
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpParams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/**statusCode*/, "Sending empty PDU");
156     }
157     connection.setOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1->

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[IC] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p> | <p>release/src/android/support/v7/mms/MmsHttpClient.java</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices.</p> <p>For example, the ZTE accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE Mobile Location Service System

2004-01-31




I. Introduction

Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.

The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.

http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html

Send your location

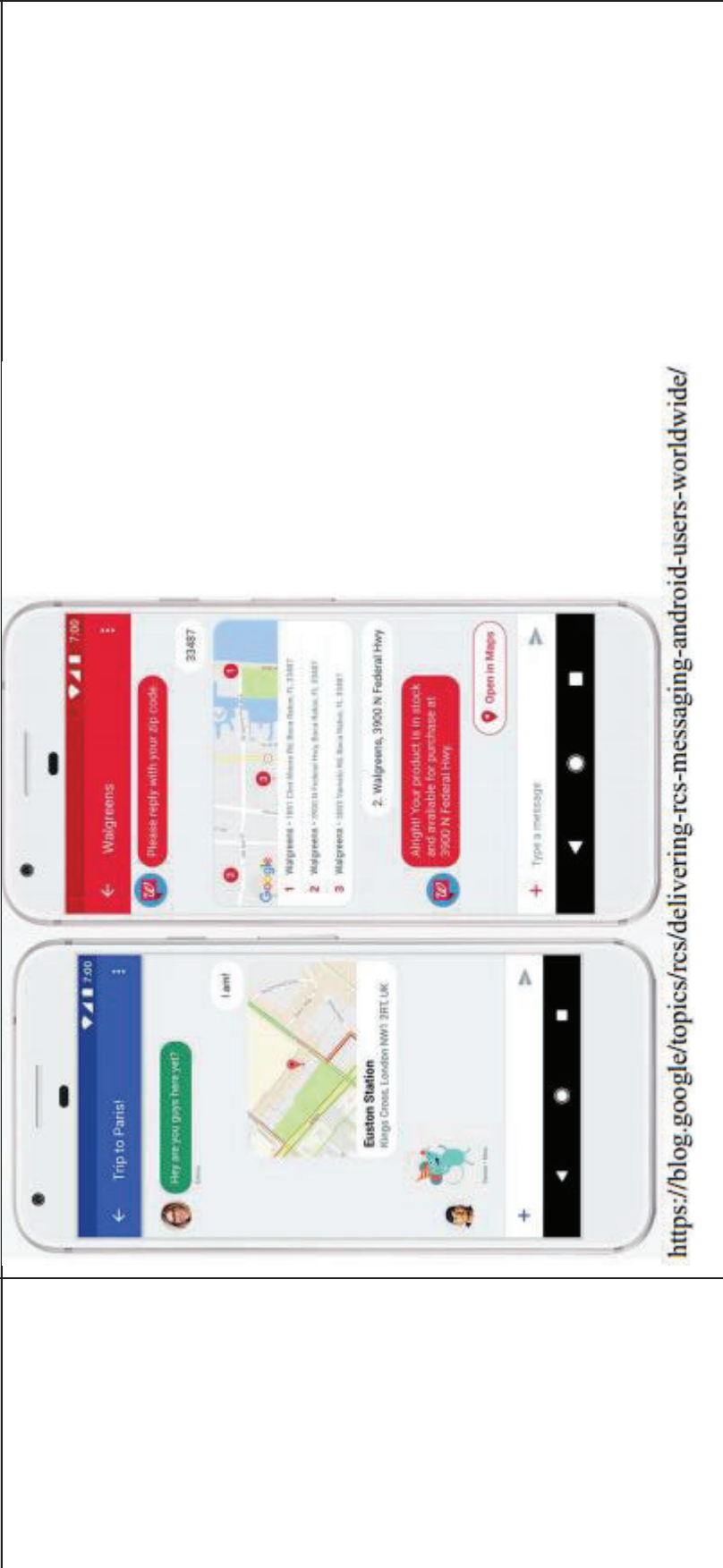
1. Open the Android Messages app .
2. Open or start a conversation.
3. Tap Attach .
4. Tap Location on .
5. To send your location, tap Send .

https://support.google.com/pixelphone/answer/6159880?hl=en&ref_topic=6211804

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share a location or place

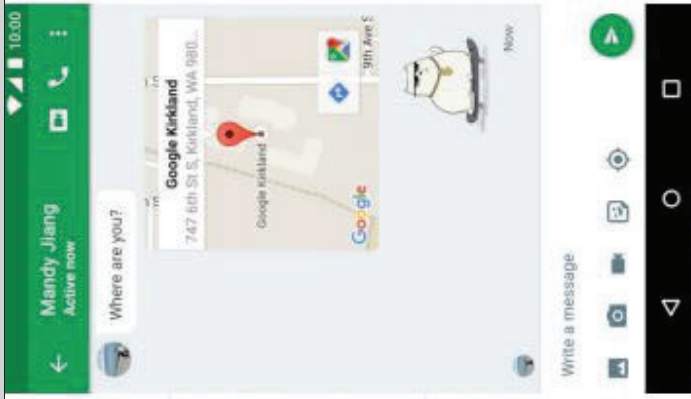
Share your location

1. On your Android phone or tablet, open the Hangouts app.
2. Open a conversation.
3. Tap Location.
4. Tap Select this location > Select.

Share a place

1. On your Android phone or tablet, open the Hangouts app.
2. Open a conversation.
3. Tap Location > Search Q.
4. Type in a location or address.
5. Tap Select.

https://support.google.com/hangouts/answer/3115410?visit_id=l-636271867303650973-2491837168&rd=1&co=GENIE.Platform%3DAndroid&oco=1
<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>



Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.</p> <p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><u>Further regarding Google Maps</u>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><u>Exemplary Support for Google Maps:</u></p> |
|---------------------------|--|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location



1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE




<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD



Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAn

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

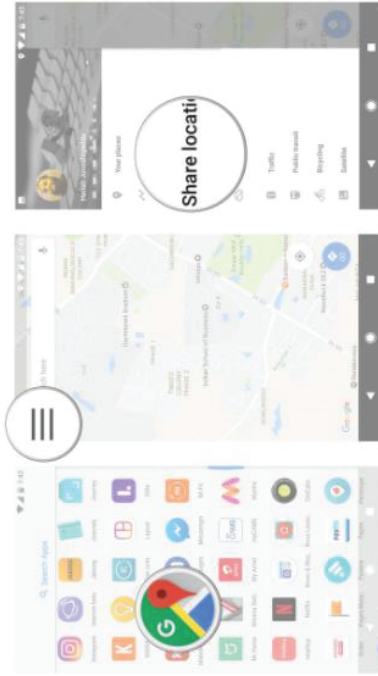
US9408055B2

ZTE

droid&oco=1

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.

6. Tap Select People.

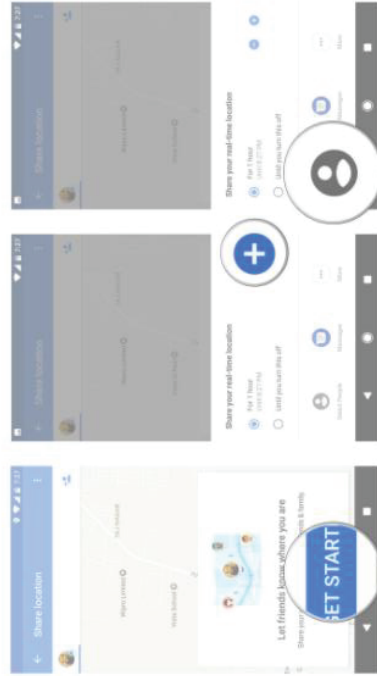


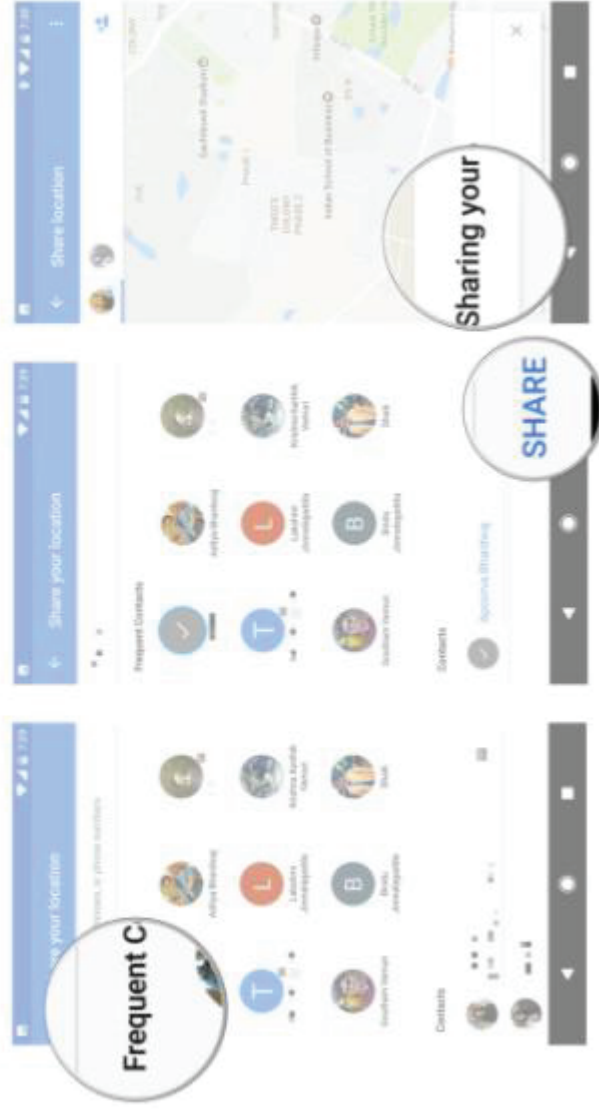
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://www.androidcentral.com/how-share-location-google-maps>

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

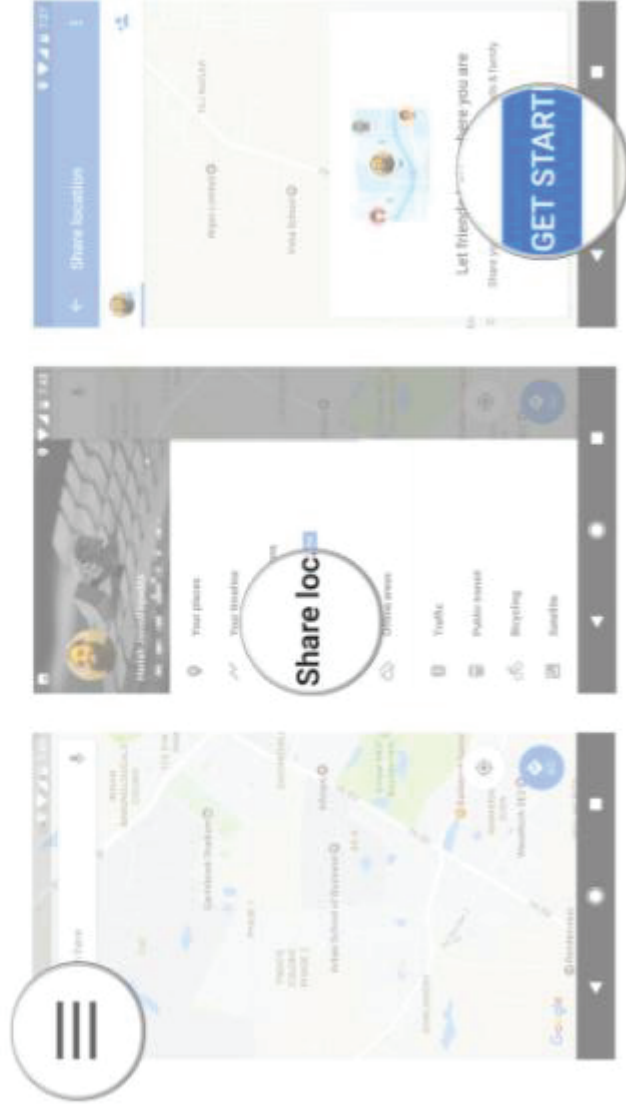
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

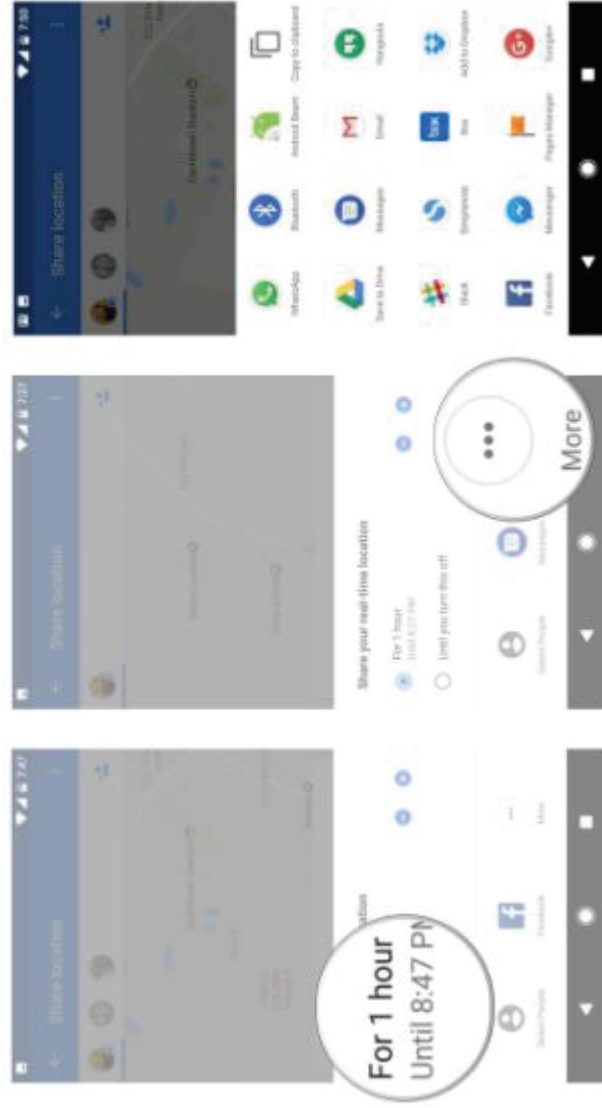
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

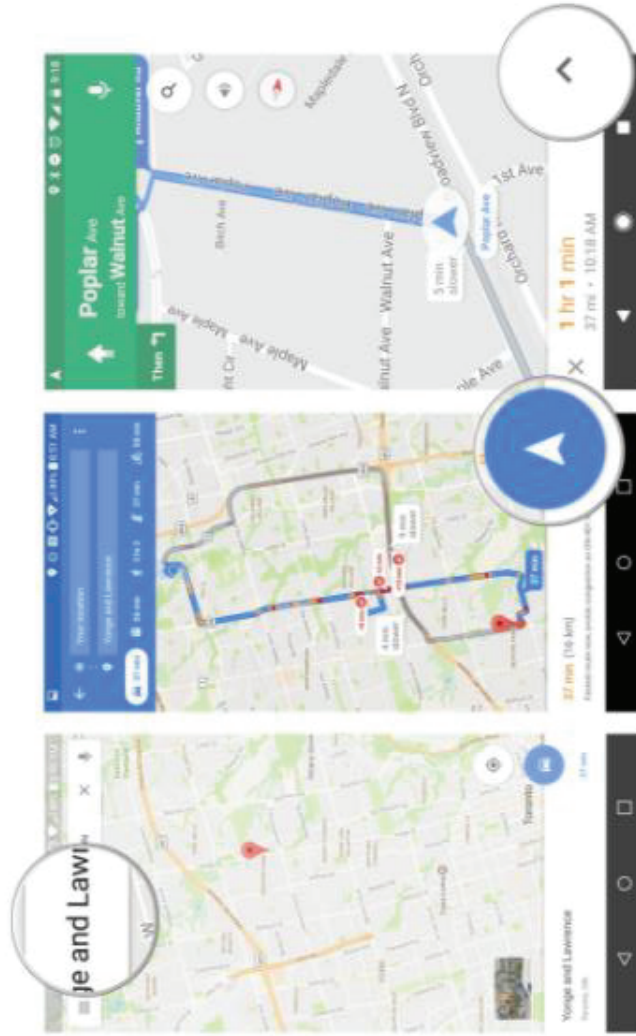
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



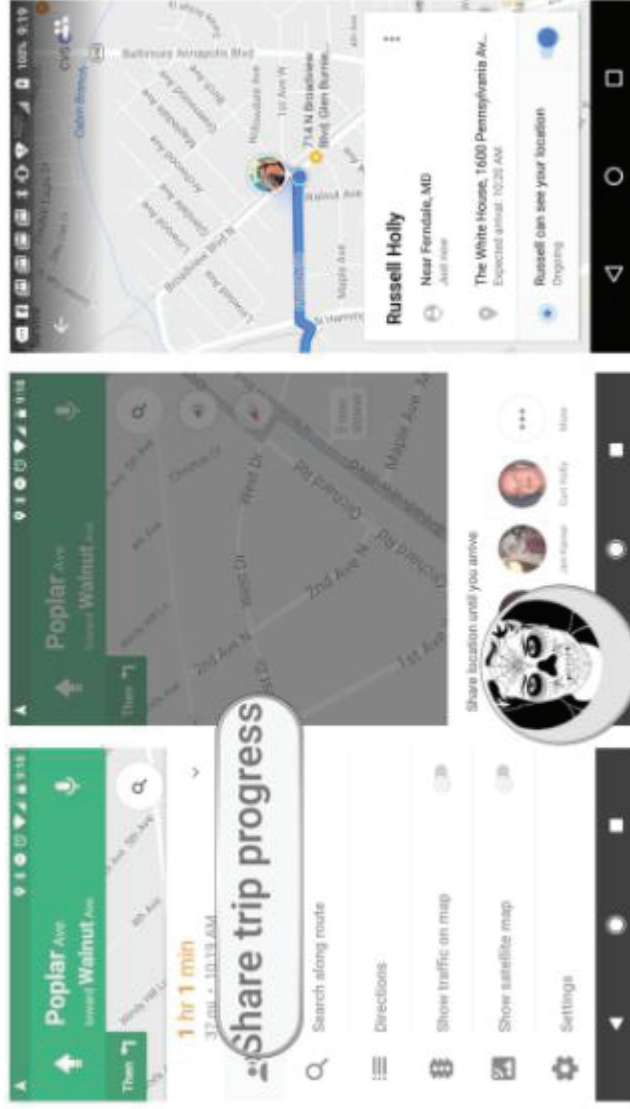
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



You can also stop sharing your location with someone before a trip ends.

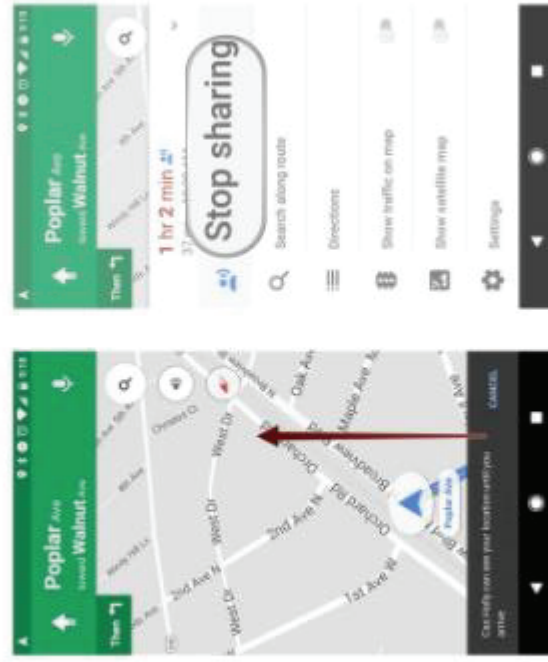
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?

<https://www.androidcentral.com/how-share-location-google-maps>





As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.



https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

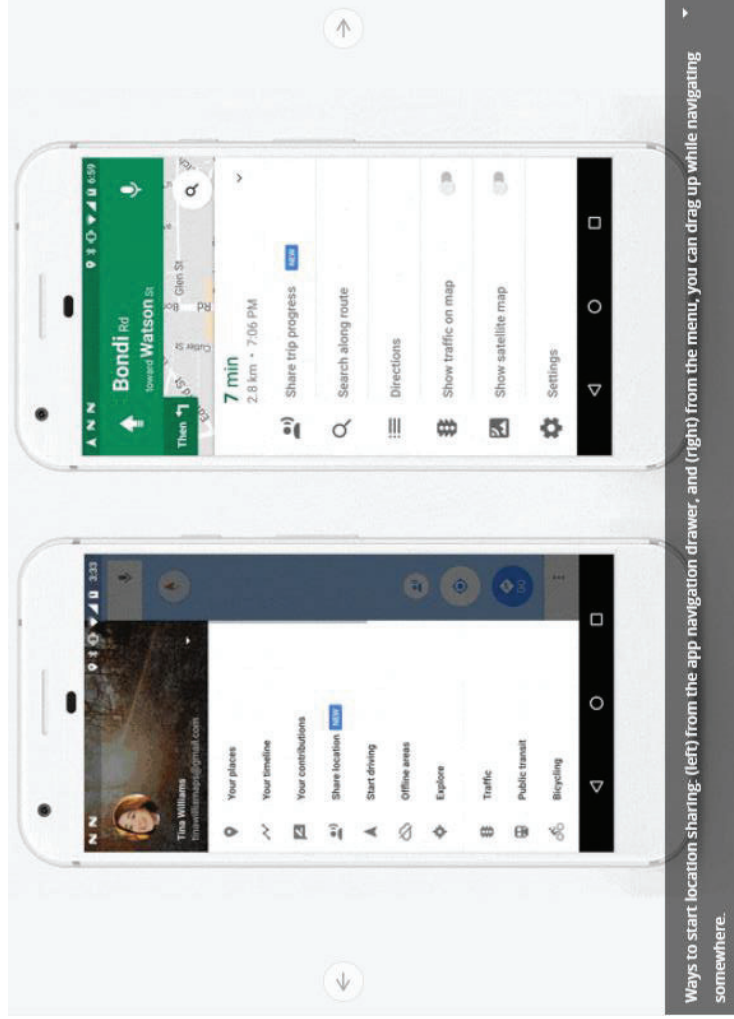
US9408055B2

ZTE

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

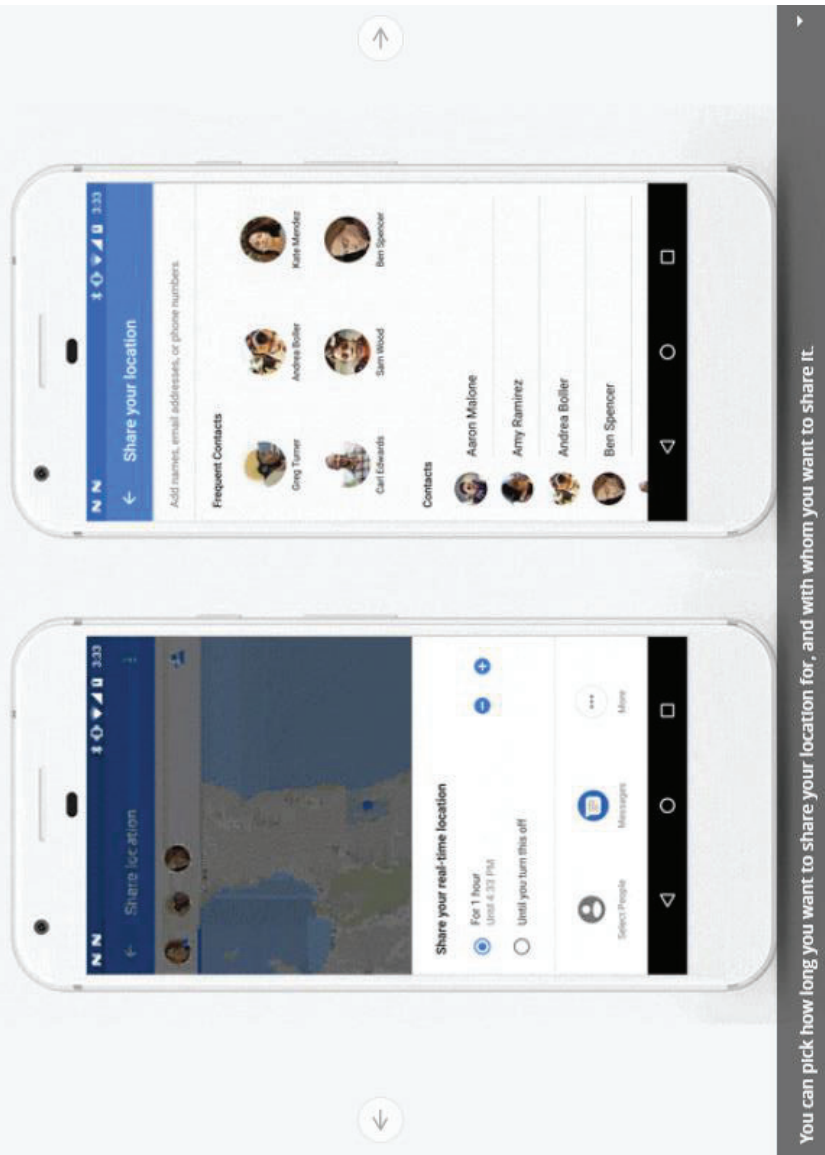


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

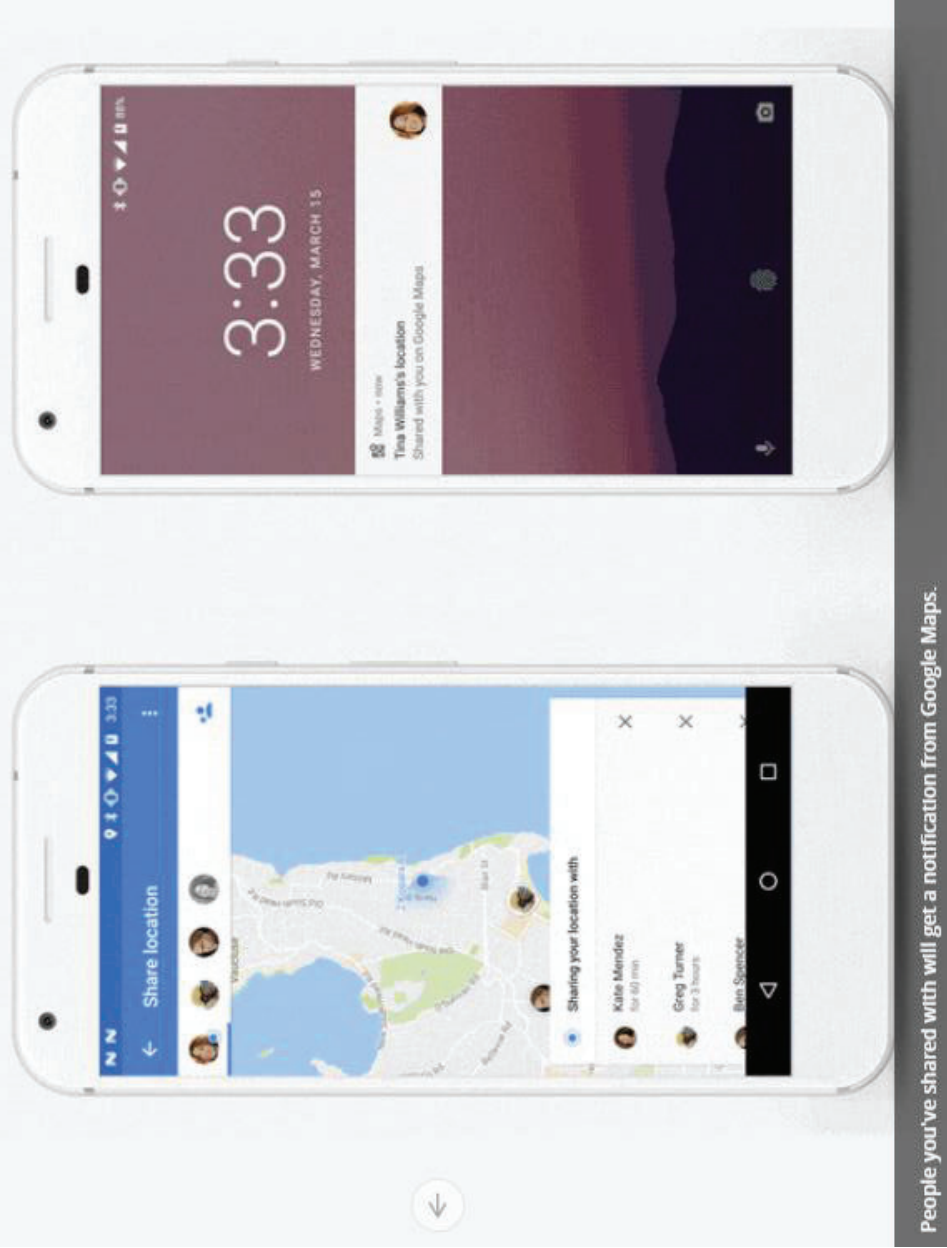


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

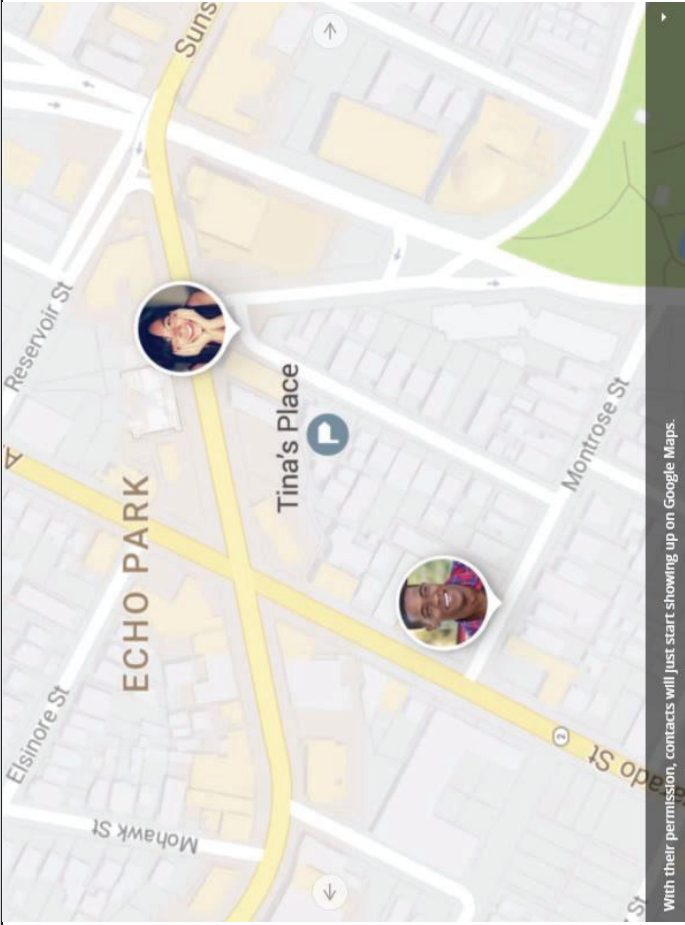


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



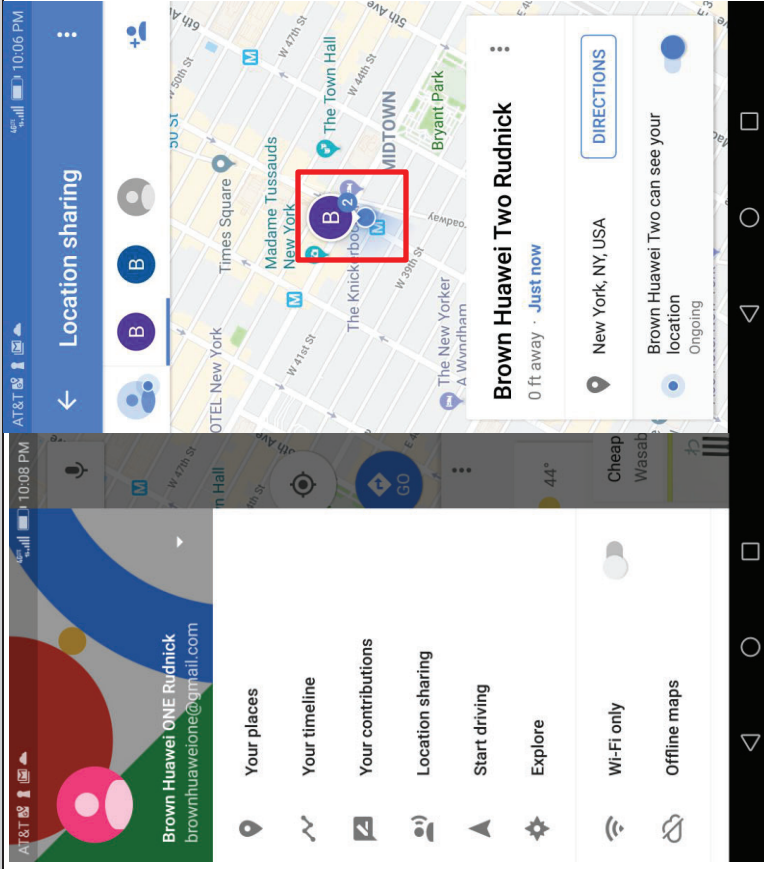
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | ZTE <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the `ContactsContract.Data` table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the `ContactsContract.RawContacts` table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the `ContactsContract.Contacts` table represents an aggregate of one or more `RawContacts` presumably describing the same person. When data in or associated with the `RawContacts` table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- `ContactsContract.Groups`, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- `ContactsContract.StatusUpdates`, which contains social status updates including IM availability.
- `ContactsContract.AggregationExceptions`, which is used for manual aggregation and disaggregation of raw contacts
- `ContactsContract.Settings`, which contains visibility and sync settings for accounts and groups.
- `ContactsContract.SyncState`, which contains free-form data maintained on behalf of sync adapters
- `ContactsContract.PhoneLookup`, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a `ContactsContract.Data` row that is linked to the raw contact's `_id` value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for `emilyd@gmail.com` (the raw contact row for Thomas Higginson associated with the Google account `emilyd@gmail.com`) has a home email address of `thigg@gmail.com` and a work email address of `thomas.higginson@gmail.com`, the `Contacts Provider` stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the `ContactsContract.Data` table. To help manage this, the `ContactsContract.Data` table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

| Task | Action | Data | MIME type | Notes |
|----------------------------|--------------------------|--|-----------|--|
| Pick a contact from a list | <code>ACTION_PICK</code> | One of: <ul style="list-style-type: none"> <code>Contacts.CONTENT_URI</code>, which displays a list of contacts. <code>Phone.CONTENT_URI</code>, which displays a list of phone numbers for a raw contact. <code>StructuredPostal.CONTENT_URI</code>, which displays a list of postal addresses for a raw contact. <code>Email.CONTENT_URI</code>, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. |

<https://developer.android.com/guide/topics/providers/contacts-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/list/ContactsRequest.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactFileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 458 private void configureFragments(boolean fromRequest) { 459 if (fromRequest) { 460 ContactListFilter filter = null; 461 int actionCode = mRequest.getActionCode(); 462 boolean searchMode = mRequest.isSearchMode(); 463 final int tabToOpen; 464 switch (actionCode) { 465 case ContactsRequest.ACTION_ALL_CONTACTS: 466 filter = ContactListFilter.createFilterWithType(467 ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468 tabToOpen = TabState.ALL; 469 break; 470 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471 filter = ContactListFilter.createFilterWithType(472 ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473 tabToOpen = TabState.ALL; 474 break; 475 case ContactsRequest.ACTION_FREQUENT: 476 case ContactsRequest.ACTION_STREQUENT: 477 case ContactsRequest.ACTION_STARRED: 478 tabToOpen = TabState.FAVORITES; 479 break; 480 case ContactsRequest.ACTION_VIEW_CONTACT: 481 tabToOpen = TabState.ALL; 482 break; 483 default: 484 tabToOpen = -1; 485 break; 486 } 487 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

488     if (tabToOpen != -1) {
489         mActionBarAdapter.setCurrentTab(tabToOpen);
490     }
491
492     if (filter != null) {
493         mContactListFilterController.setContactListFilter(filter, false);
494         searchMode = false;
495     }
496
497     if (mRequest.getContactUri() != null) {
498         searchMode = false;
499     }
500
501     mActionBarAdapter.setSearchMode(searchMode);
502     configureContactListFragmentForRequest();
503 }
504
505 configureContactListFragment();
506
507 invalidateOptionsMenuIfNeeded();
508 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,          // 3
43             Data.LOOKUP_KEY,         // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53
54     public static class GroupDetailQuery {
55         private static final String[] PROJECTION = new String[] {
56             Data.CONTACT_ID,           // 0
57             Data.PHOTO_URI,           // 1
58             Data.LOOKUP_KEY,          // 2
59             Data.DISPLAY_NAME_PRIMARY, // 3
60             Data.CONTACT_PRESENCE,    // 4
61             Data.CONTACT_STATUS,      // 5
62         };
63
64         public static final int CONTACT_ID           = 0;
65         public static final int CONTACT_PHOTO_URI   = 1;
66         public static final int CONTACT_LOOKUP_KEY   = 2;
67         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
68         public static final int CONTACT_PRESENCE_STATUS = 4;
69         public static final int CONTACT_STATUS      = 5;
70     }
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

B-115

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-117

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19  * Meta-data for a contact group. We load all groups associated with the contact's
20  * constituent accounts.
21  */
22  public final class GroupMetaData {
23      private String mAccountName;
24      private String mAccountType;
25      private String mDataSet;
26      private Long mGroupId;
27      private String mTitle;
28      private boolean mDefaultGroup;
29      private boolean mFavorites;
30
31      public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32                          String title, boolean defaultGroup, boolean favorites) {
33          this.mAccountName = accountName;
34          this.mAccountType = accountType;
35          this.mDataSet = dataSet;
36          this.mGroupId = groupId;
37          this.mTitle = title;
38          this.mDefaultGroup = defaultGroup;
39          this.mFavorites = favorites;
40      }
41
42      public String getAccountName() {
43          return mAccountName;
44      }
45
46      public String getAccountType() {
47          return mAccountType;
48      }
49
50      public String getDataSet() {
51          return mDataSet;
52      }
53
54      public long getGroupId() {
55          return mGroupId;
56      }
57
58      public String getTitle() {
59          return mTitle;
60      }
61
62      public boolean isDefaultGroup() {
63          return mDefaultGroup;
64      }
65
66      public boolean isFavorites() {
67          return mFavorites;

```

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56  * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57  *
58  * This class serves two purposes:
59  * - Process phone verification SMS messages
60  * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61  */
62  public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> ignoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1->

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsuUtil.isAtLeastL_MRI()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

228 @Override
229 public void onReceive(final Context context, final Intent intent) {
230     LogUtil.v(TAG, "SmsReceiver.onReceive " + intent);
231     // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver.
232     if (PhoneUtils.getDefault().isSmsEnabled()) {
233         final String action = intent.getAction();
234         if (OsUtil.isSecondaryUser() &&
235             (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) ||
236              // TODO: update this with the actual constant from Telephony
237               "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) {
238             postNewMessageSecondaryUserNotification();
239         } else if (!OsUtil.isAtLeastKLP()) {
240             deliverSmsIntent(context, intent);
241         }
242     }
243 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

52 * Class that sends chat message via MMS.
53 *
54 * The interface emulates a blocking send similar to making an HTTP request.
55 */
56 public class MmsSender {
57     private static final String TAG = LogUtil.BUGLE_TAG;
58
59     /**
60      * Send an MMS message.
61      *
62      * @param context Context
63      * @param messageUri The unique URI of the message for identifying it during sending
64      * @param sendReq The SendReq PDU of the message
65      * @throws MmsFailureException
66      */
67     public static void sendMms(final Context context, final int subId, final Uri messageUri,
68                               final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException {
69         sendMms(context,
70                subId,
71                messageUri,
72                null /* LocationUrl */,
73                sendReq,
74                true /* responseImportant */,
75                sentIntentExtras);
76     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97 * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98 *
99 * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *   it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113 String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));
131     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpParams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/**statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178 } else if (METHOD_GET.equals(method)) {
179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180         logHttpHeaders(connection.getRequestProperties());
181     }
182     connection.setRequestMethod(METHOD_GET);
183 }
184 // Get response
185 final int responseCode = connection.getResponseCode();
186 final String responseMessage = connection.getResponseMessage();
187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189     logHttpHeaders(connection.getHeaderFields());
190 }
191 if (responseCode / 100 != 2) {
192     throw new MmsHttpException(responseCode, responseMessage);
193 }
194 final InputStream in = new BufferedInputStream(connection.getInputStream());
195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196 final byte[] buf = new byte[4096];
197 int count = 0;
198 while ((count = in.read(buf)) > 0) {
199     byteOut.write(buf, 0, count);
200 }
201 in.close();
202 final byte[] responseBody = byteOut.toByteArray();
203 Log.d(MmsService.TAG, "HTTP: response size="
204     + (responseBody != null ? responseBody.length : 0));
205 return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1522 219 1585">ZTE</p> <p data-bbox="235 504 308 1585">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <pre data-bbox="357 357 397 1564">public static LocationRequest create ()</pre> <p data-bbox="422 1071 446 1564">Create a location request with default parameters.</p> <p data-bbox="479 462 544 1564">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="560 1470 584 1564">Returns</p> <ul data-bbox="609 1281 633 1543" style="list-style-type: none">• a new location request <p data-bbox="682 304 714 1585">https://developers.google.com/android/reference/com/google/android/location/gms/location/LocationRequest</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><code>public Task<Location> getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task<LocationAvailability> getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|---|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | | | |
| <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|---------|---------------------------------------|----------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1575">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1575">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check <code>isSuccessful()</code> to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p>Public Constructors</p> <p>public MapView (Context context)</p> <p>public MapView (Context context, AttributeSet attrs)</p> <p>public MapView (Context context, AttributeSet attrs, int defStyleAttr)</p> <p>public MapView (Context context, GoogleMapOptions options)</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[1D] transmitting IP-based messages including a location of the first device to the respective second devices;</p> | <p>public void <code>getMapAsync</code> (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void <code>onCreate</code> (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p>locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> <h3 style="text-align: center;">Location Based Service (LBS)</h3> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.


Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. <p>Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:

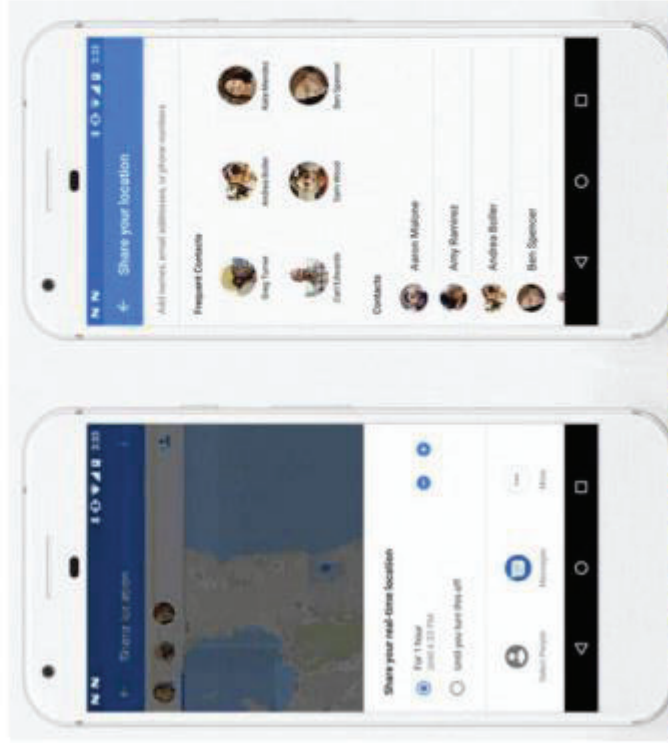
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. If you haven't already, add their Gmail address to your Google Contacts [\[2\]](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap the Menu  > **Share location** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

<https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&hl=en>

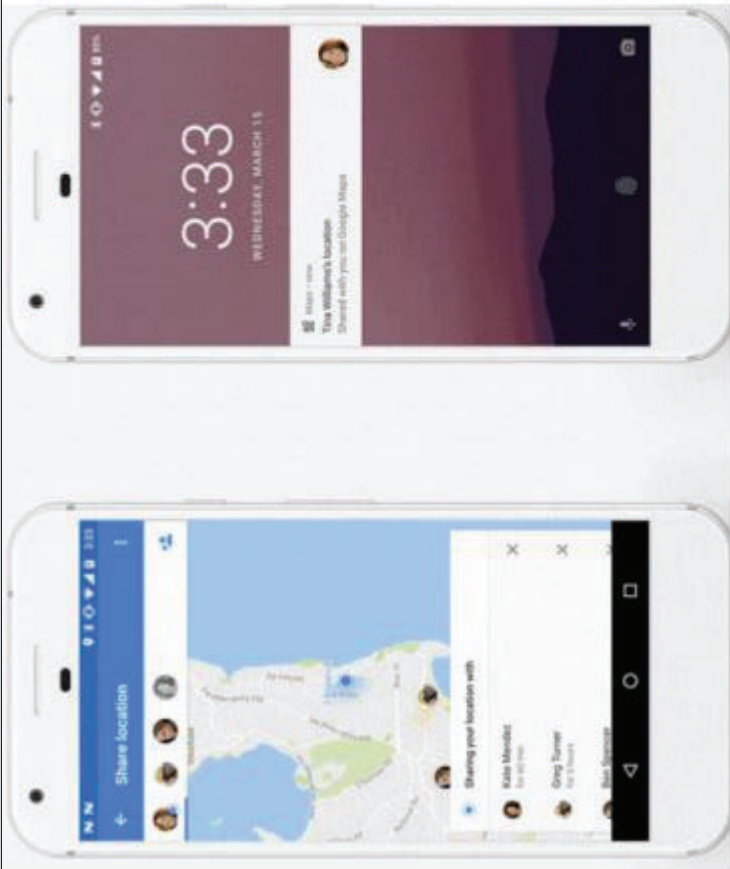


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Below are exemplary methods used by Google applications to obtain, send, and receive locations.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

The Google Play services Location API

The Google Play services Location API is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:

- Determine the device location.
- Listen for location changes.
- Determine the mode of transportation, if the device is moving.
- Create and monitor predefined geographical regions, known as geofences.

The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class Making Your App Location Aware or the Location API Reference. Code examples are included as part of the Google Play services SDK.

<https://developers.google.com/maps/documentation/android-api/location>

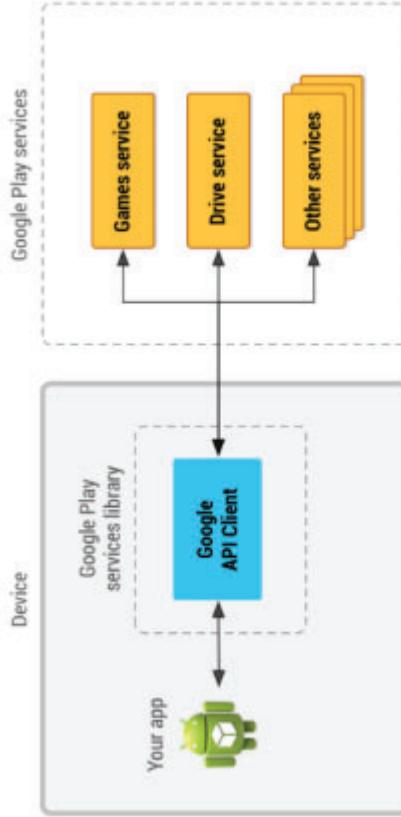


Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.

<https://developers.google.com/android/guides/api-client#Starting>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get the Last Known Location

Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's `getLastLocation()` method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the [Specify App Permissions](#) section of this document.

To request the last known location, call the `getLastLocation()` method, passing it your instance of the `GoogleApiClient` object. Do this in the `onConnected()` callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:

```

public class MainActivity extends AppCompatActivity implements
    ConnectionCallbacks, OnConnectionFailedListener {
    ...
    @Override
    public void onConnected(Bundle connectionHint) {
        mLastLocation = LocationServices.FusedLocationApi.getLastLocation(
            mGoogleApiClient);
        if (mLastLocation != null) {
            mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));
            mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));
        }
    }
}

```

The `getLastLocation()` method returns a `Location` object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.

<https://developer.android.com/training/location/retrieve-current.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[1E] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions</p> | <p>Determining the user's current location</p> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() { var startPos; var geoSuccess = function(position) { startPos = position; document.getElementById('startLat').innerHTML = startPos.coords.latitude; document.getElementById('startLon').innerHTML = startPos.coords.longitude; }; navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p>https://developers.google.com/web/fundamentals/native-hardware/user-location/</p> |
| <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device Manager places symbols on the map and in a list, where those symbols represent the linked devices and their</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device Manager places symbols on the map and in a list, where those symbols represent the linked devices and their</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>corresponding to the respective locations of the second devices;</p> | <p>ZTE</p> <p>positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p>Exemplary Support for Google Maps:</p> <p>Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1596">ZTE</p> <p data-bbox="259 693 300 1228">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1050">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

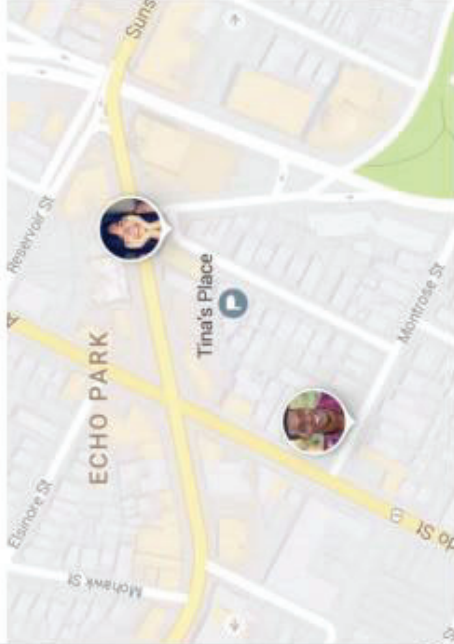
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 🗣️ > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap ⚙️ > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
|---------------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app.
2. Tap the Menu icon > **Share location**.
3. Choose someone.

- To see an updated location, tap on a friend's icon > More > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app.
2. On the map, tap the location icon.
3. At the bottom, tap More >
4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

https://support.google.com/maps/answer/7326816?hl=en&ref_topic=3092425&co=GENIE.Platform%3DAndroid&oc=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap More  > **Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share.**
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap More .
4. To temporarily hide someone, tap **Hide from map.** You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

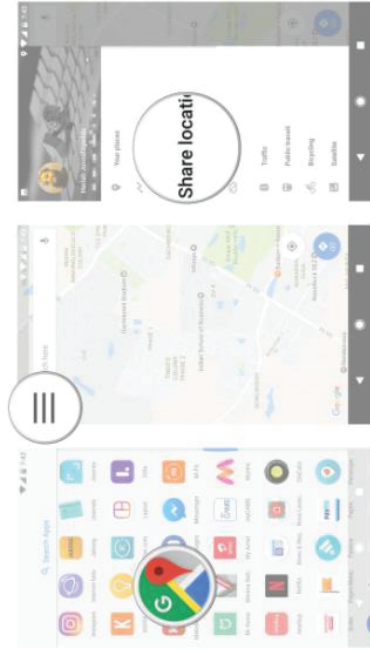
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

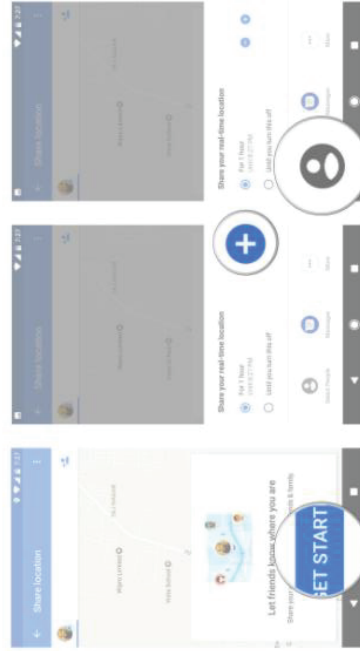
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



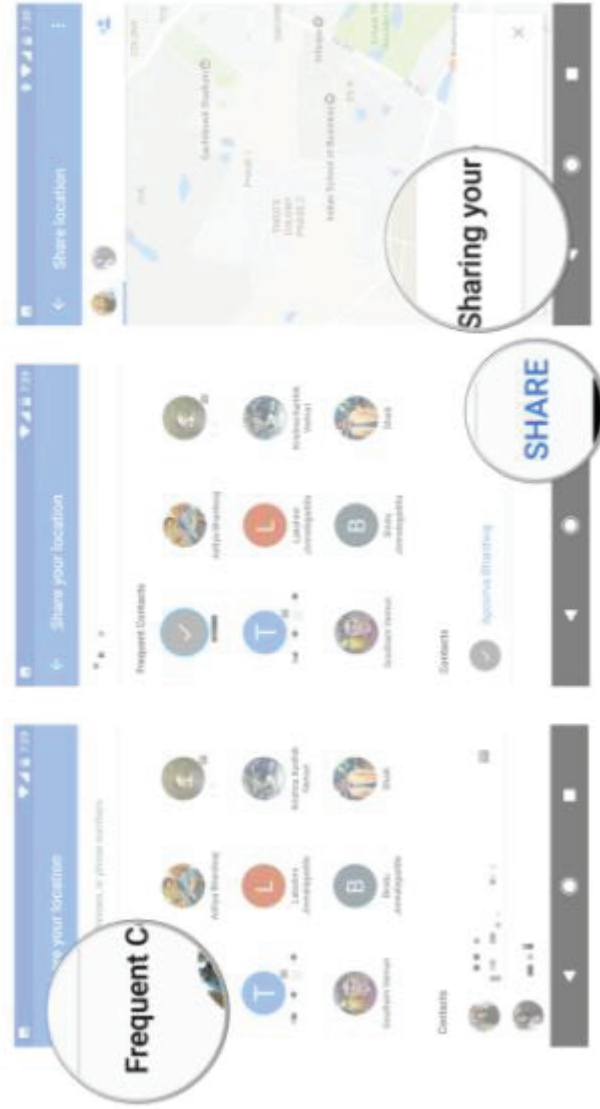
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

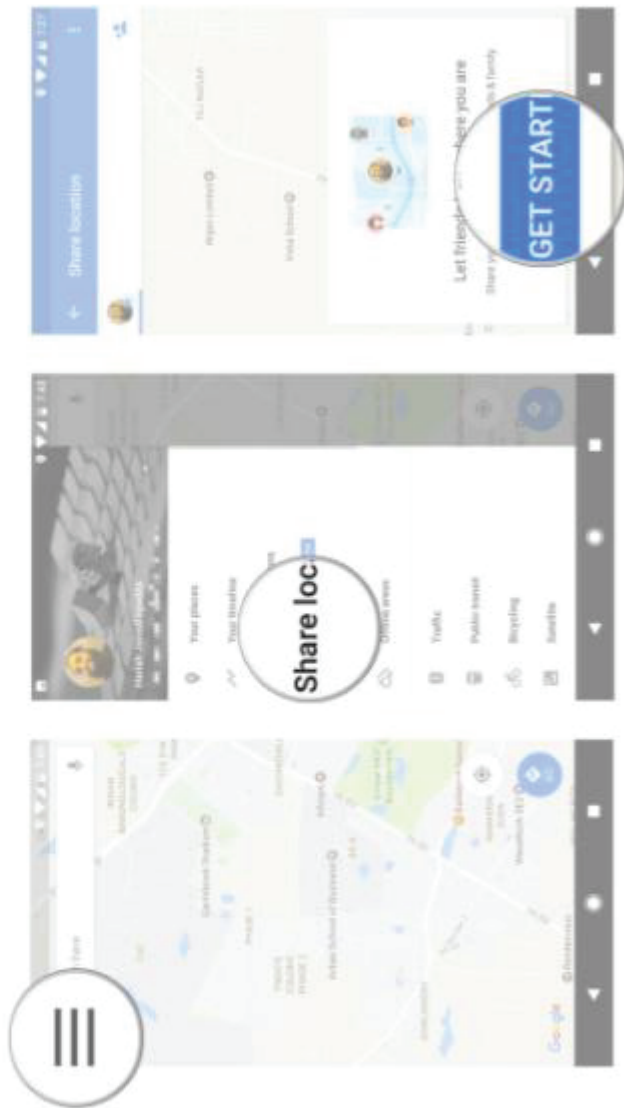
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

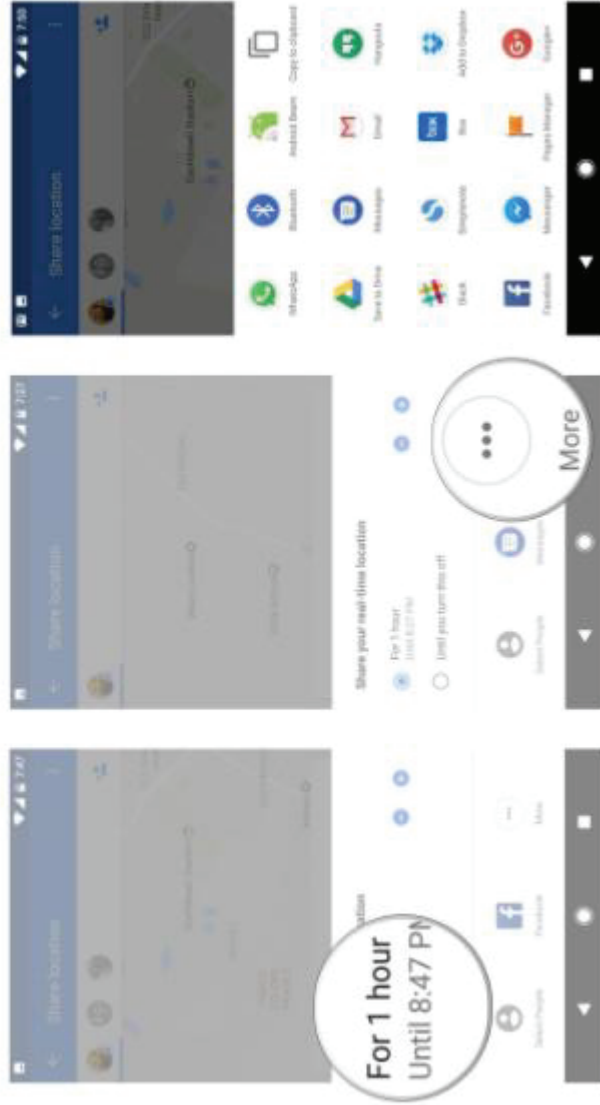
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

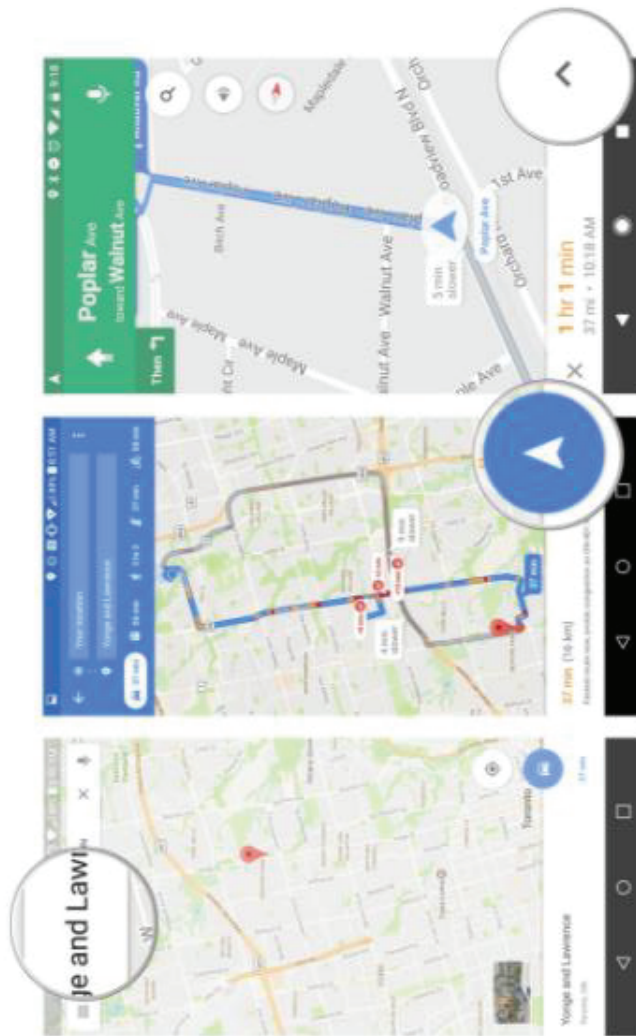
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



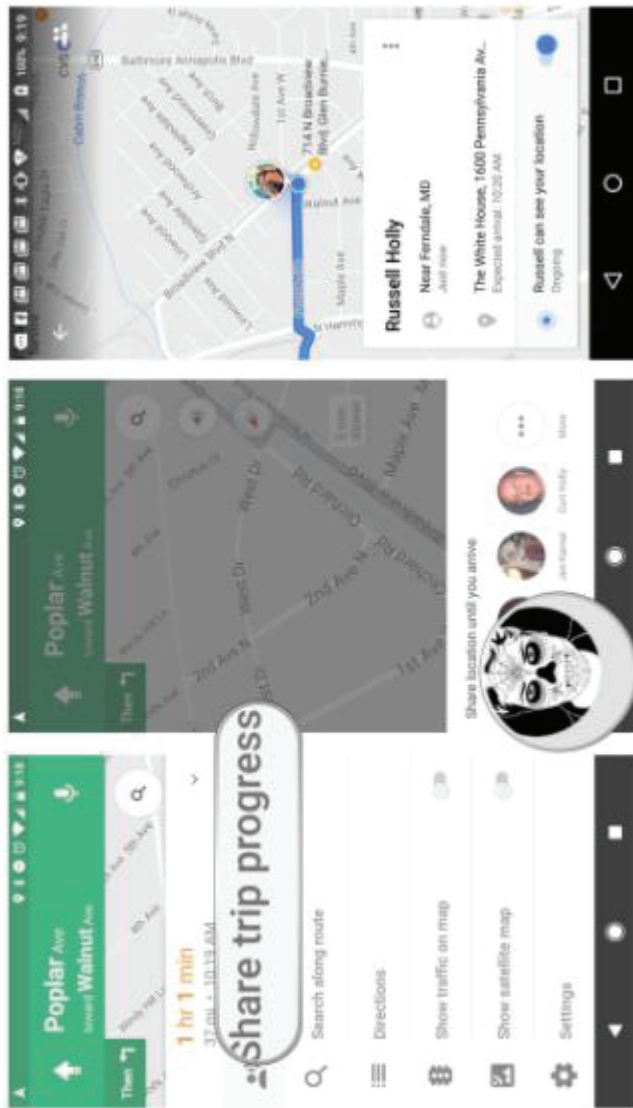
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



You can also stop sharing your location with someone before a trip ends.

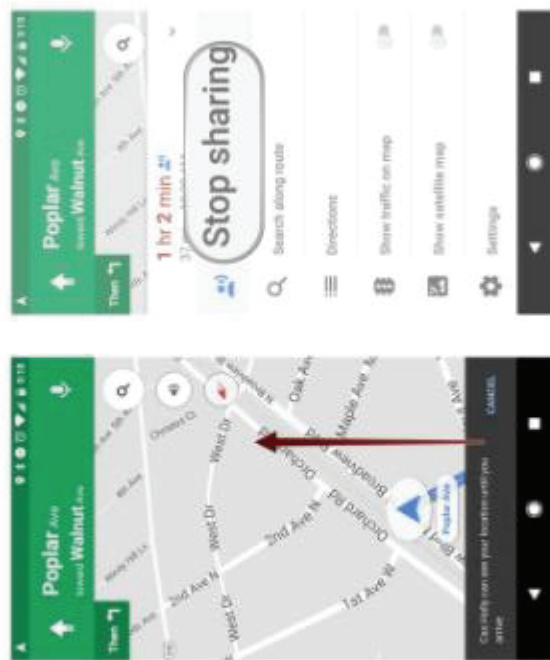
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

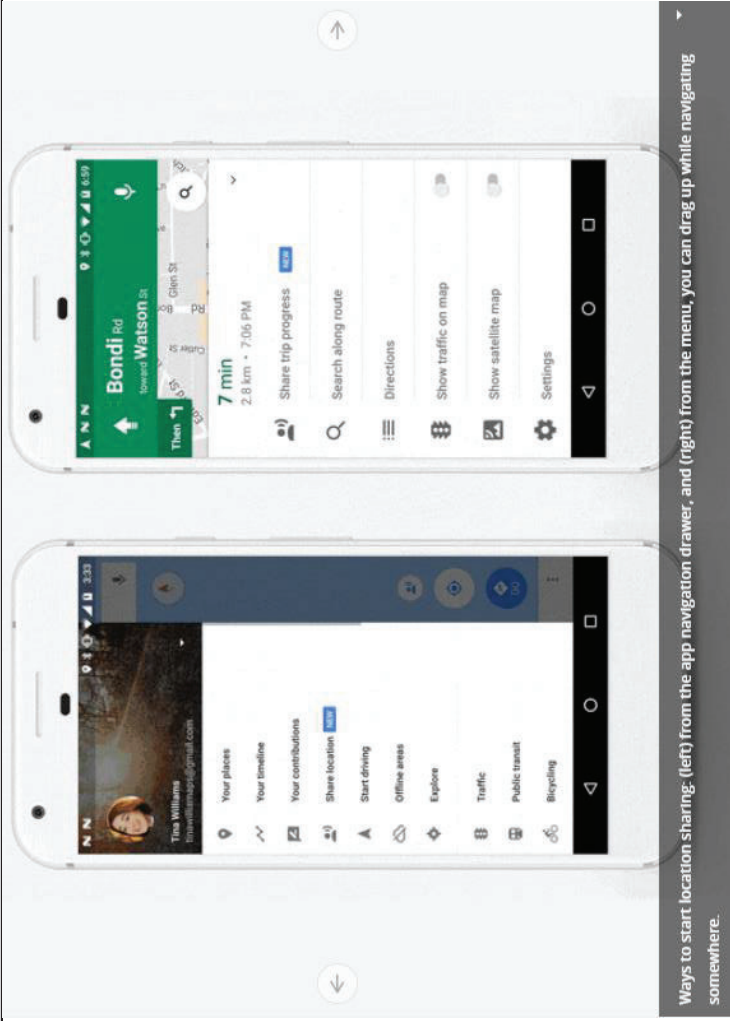
Are you excited that location sharing is back in Google Maps? How often do you use the feature?

<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

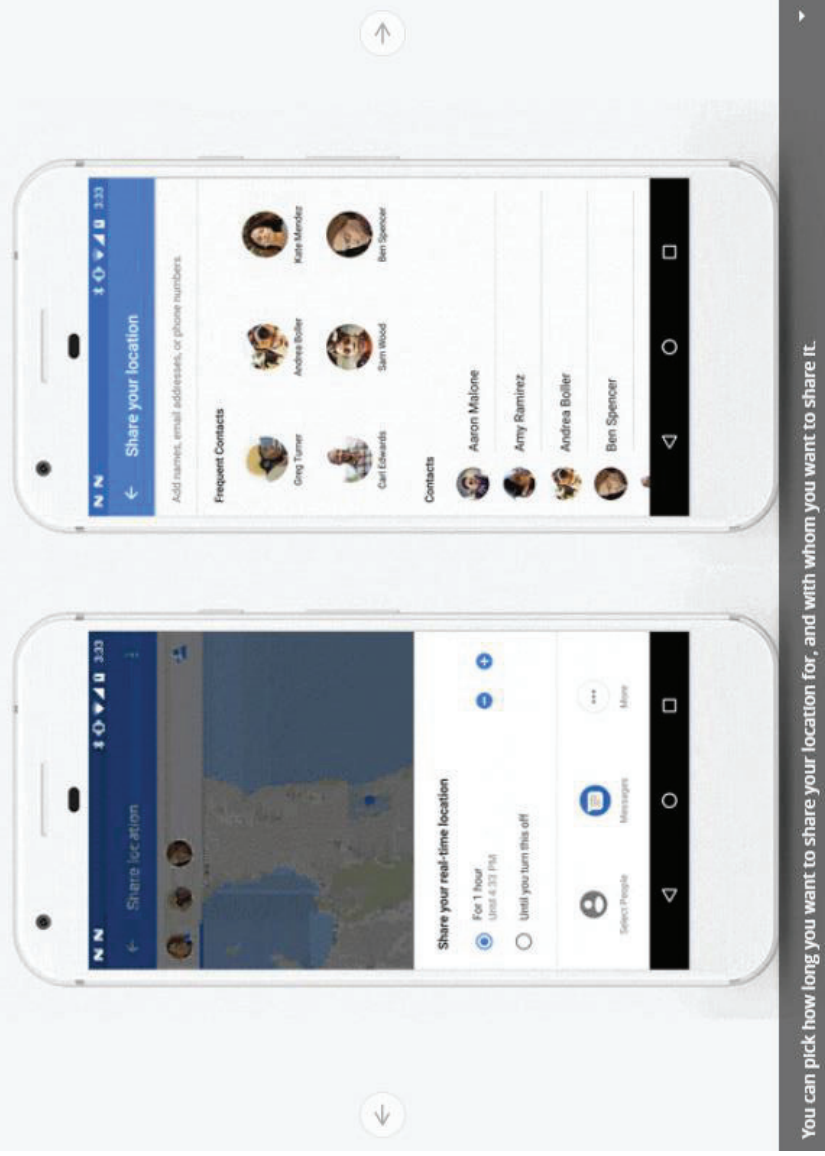


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

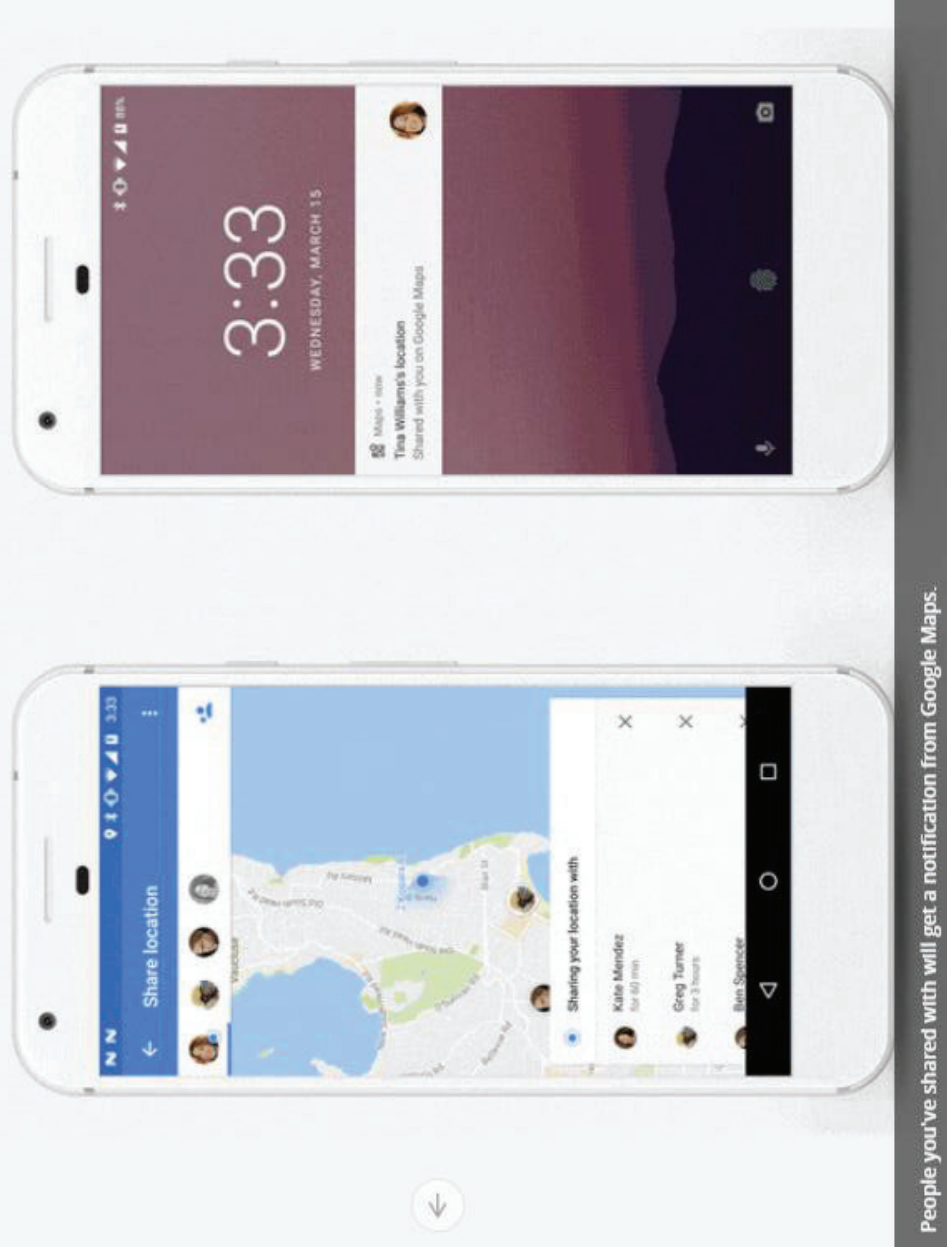


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

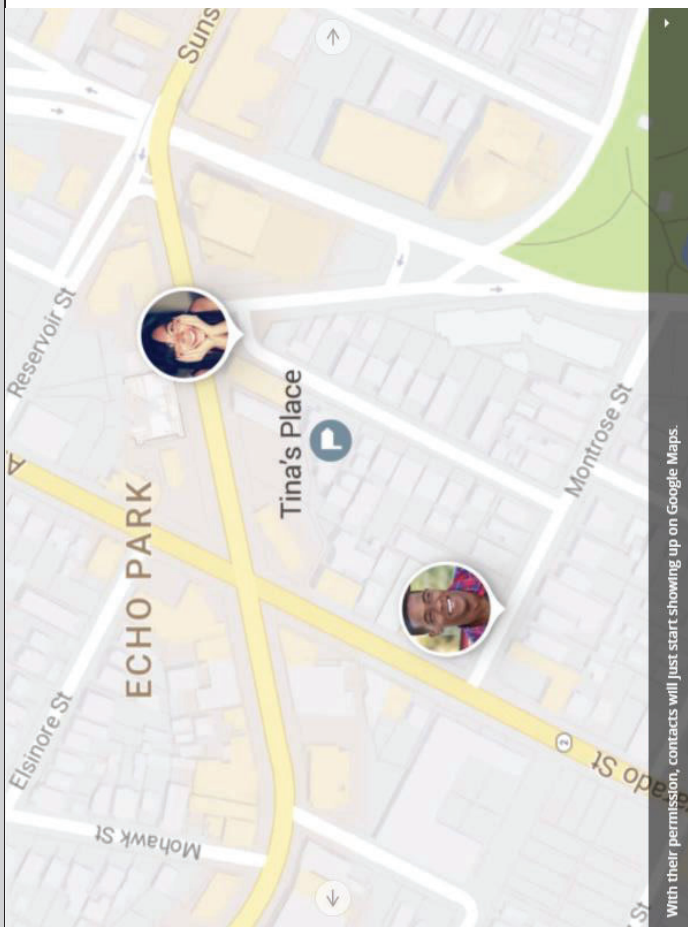


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



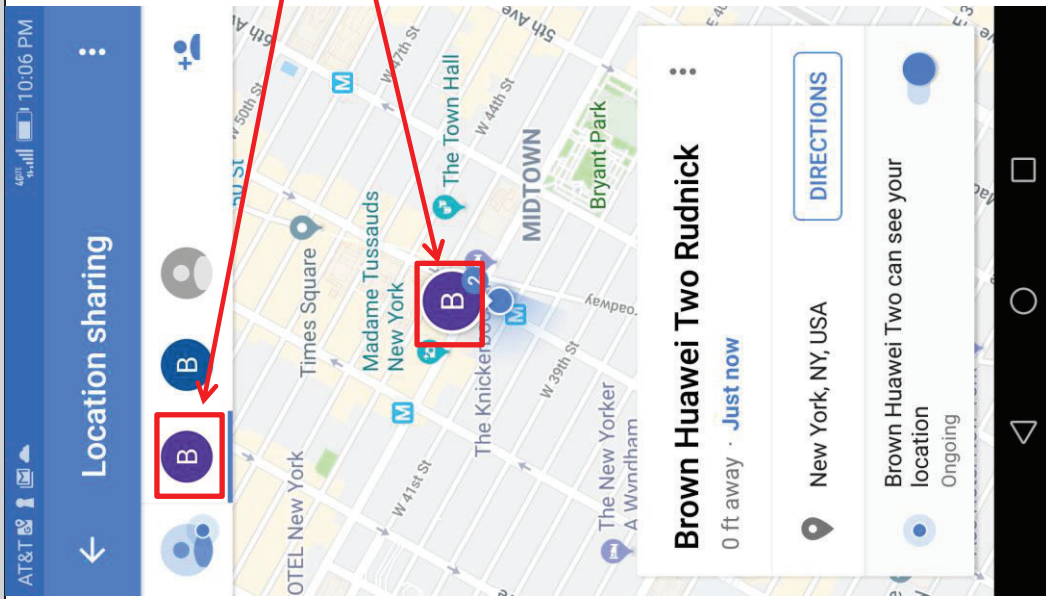
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



B-165

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE |
| | <p><u>Exemplary Source Code:</u></p> <p>The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre>public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p>Returns</p> <ul style="list-style-type: none">• a new location request <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><code>public Task<Location> getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task<LocationAvailability> getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|--------------------|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|---------|---------------------------------------|----------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1575">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1575">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check <code>isSuccessful()</code> to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="243 871 276 1564">public abstract void onLocationChanged (Location location)</p> <p data-bbox="300 1176 332 1575">Called when the location has changed.</p> <p data-bbox="349 1428 381 1554">Parameters</p> <p data-bbox="414 1470 454 1564">location</p> <p data-bbox="414 966 446 1165">The updated location.</p> <p data-bbox="519 294 560 1585">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p data-bbox="665 1281 706 1575">Public Constructors</p> <p data-bbox="763 1165 803 1564">public MapView (Context context)</p> <p data-bbox="852 966 893 1564">public MapView (Context context, AttributeSet attrs)</p> <p data-bbox="941 829 982 1564">public MapView (Context context, AttributeSet attrs, int defStyle)</p> <p data-bbox="1031 850 1071 1564">public MapView (Context context, GoogleMapOptions options)</p> <p data-bbox="1128 420 1169 1585">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>public void getMapAsync (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void onCreate (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[IF] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices;</p> | <p>ZTE</p> <p>display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server.</p> <p><u>Exemplary Support for Google Maps:</u></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 🗣️ > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap ⚙️ > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
|---------------------------|---|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap **More**  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list

If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > **More**  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAn

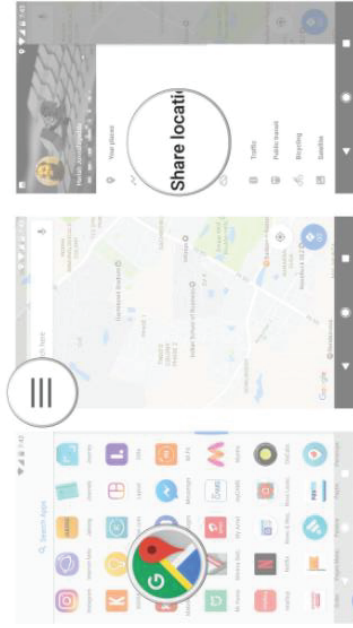
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

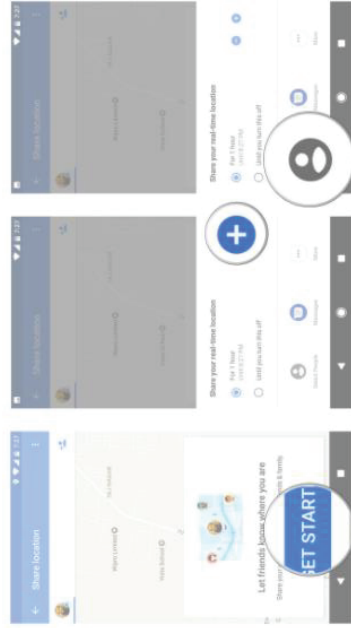
ZTE
droid&oco=1

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



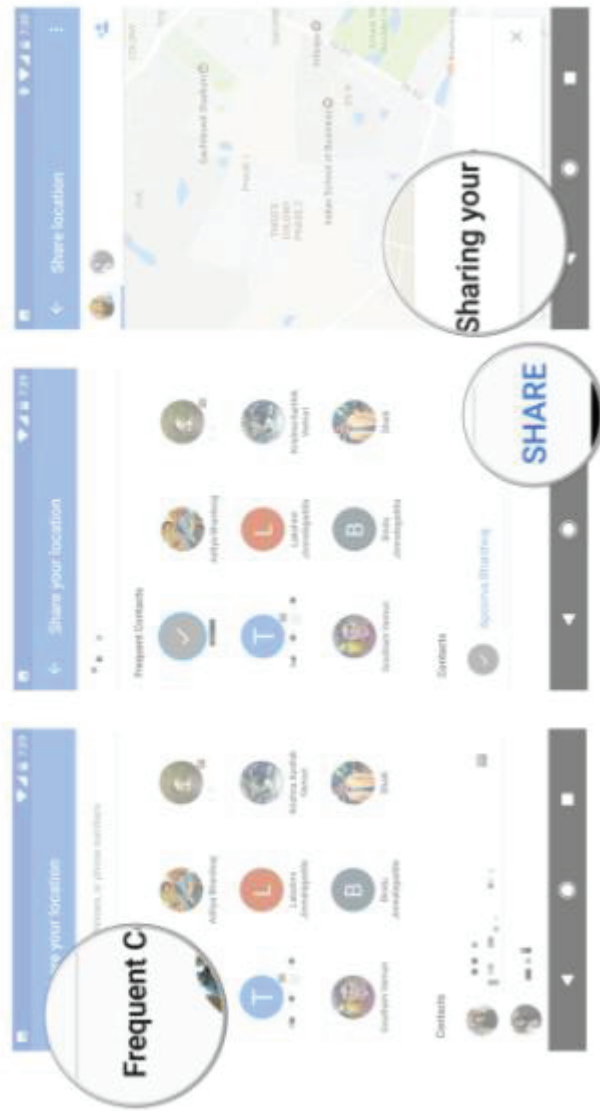
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

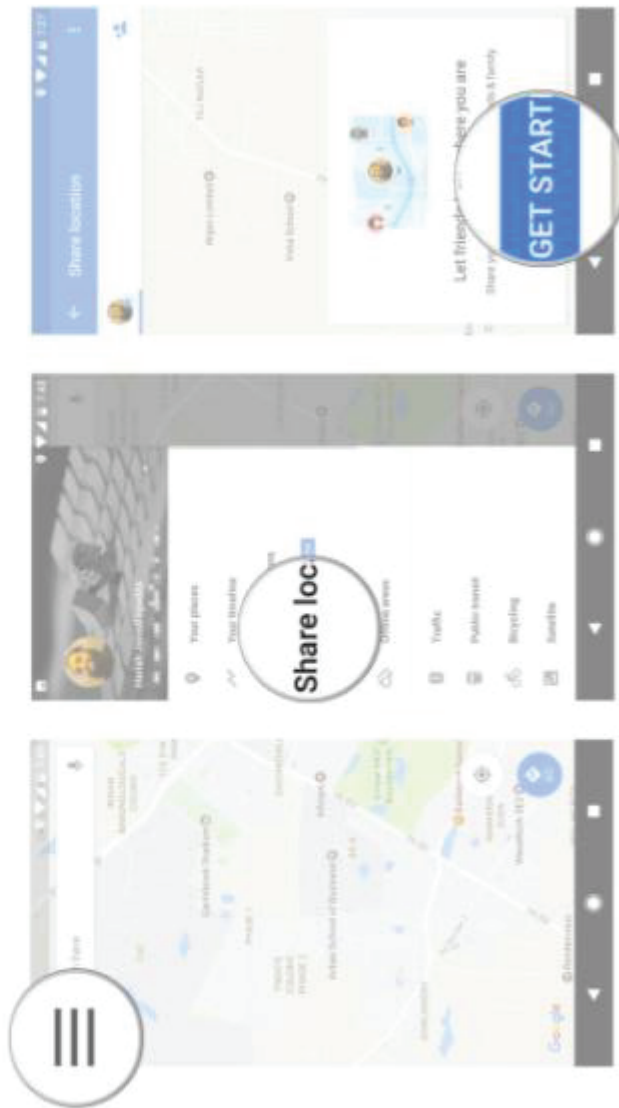
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

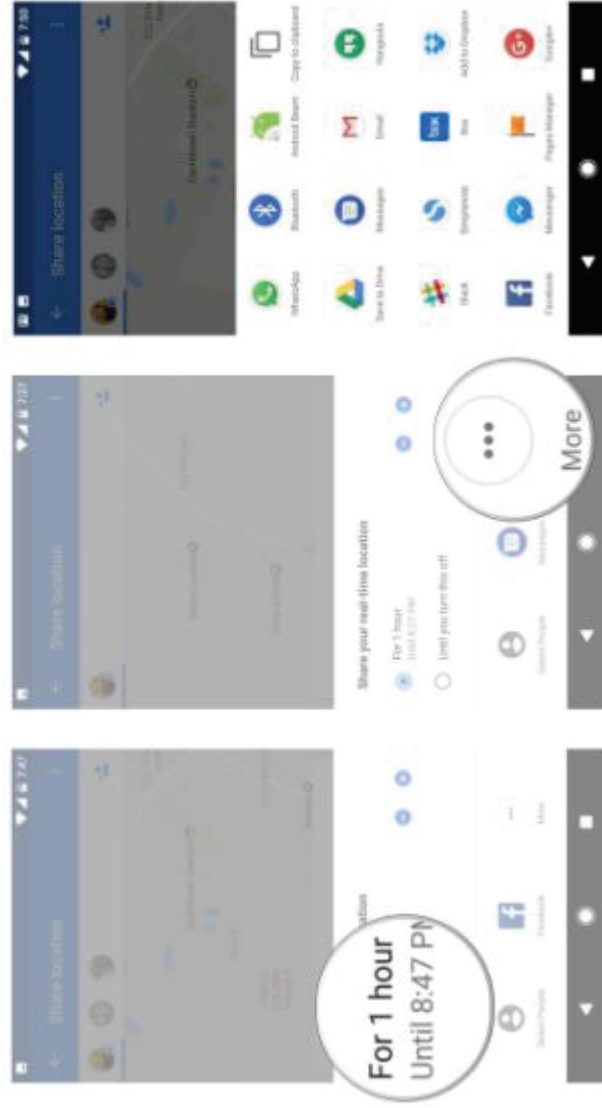
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

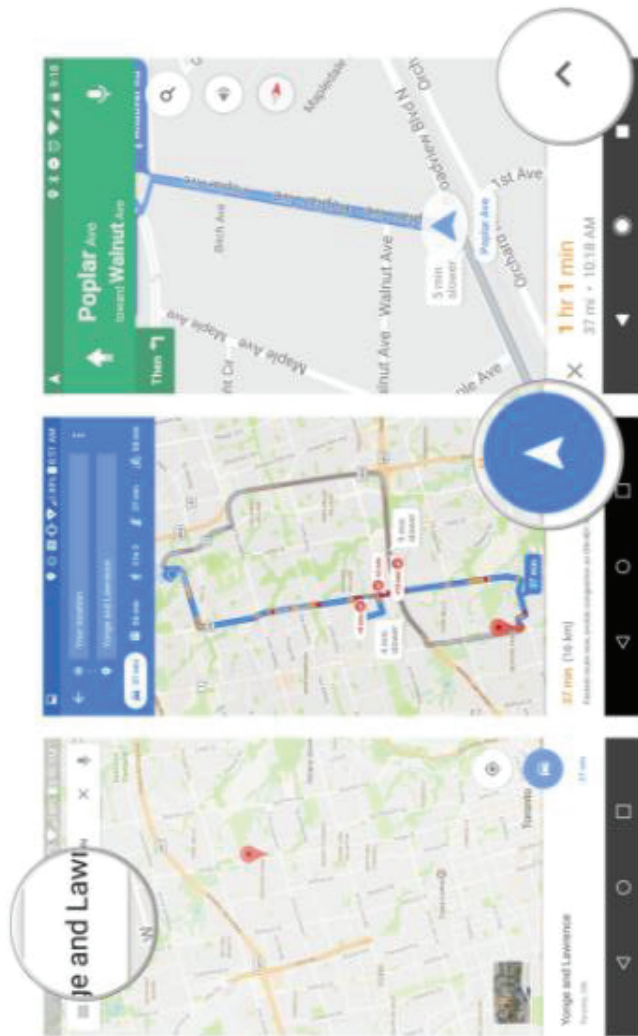
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



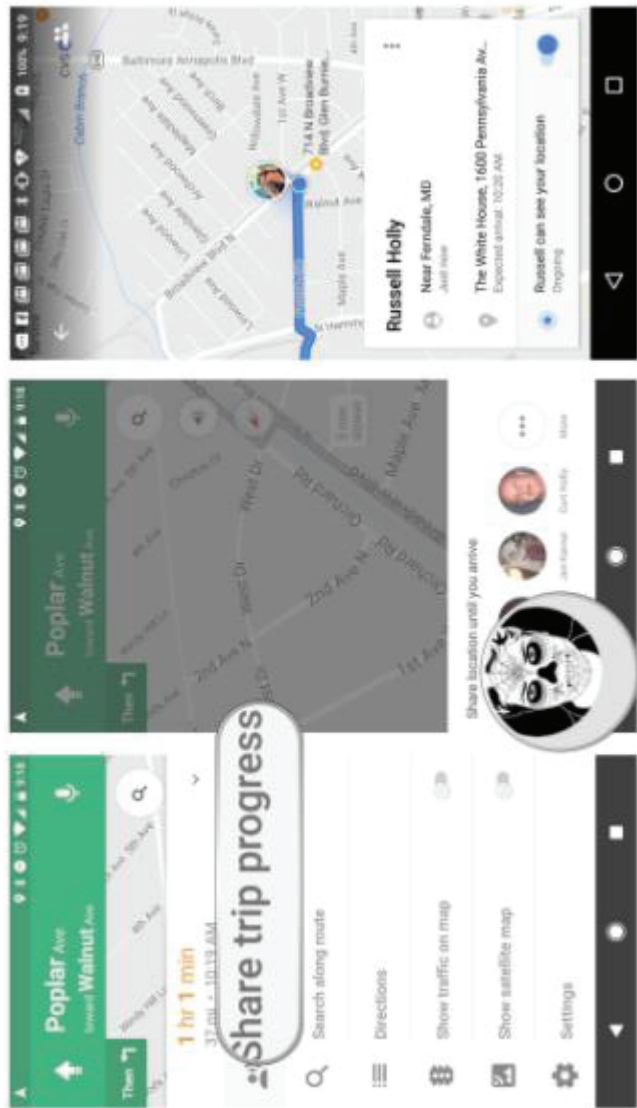
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



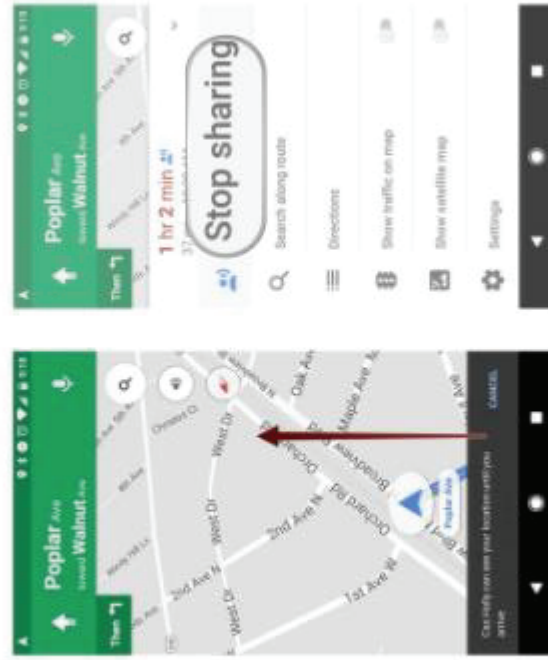
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>



As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

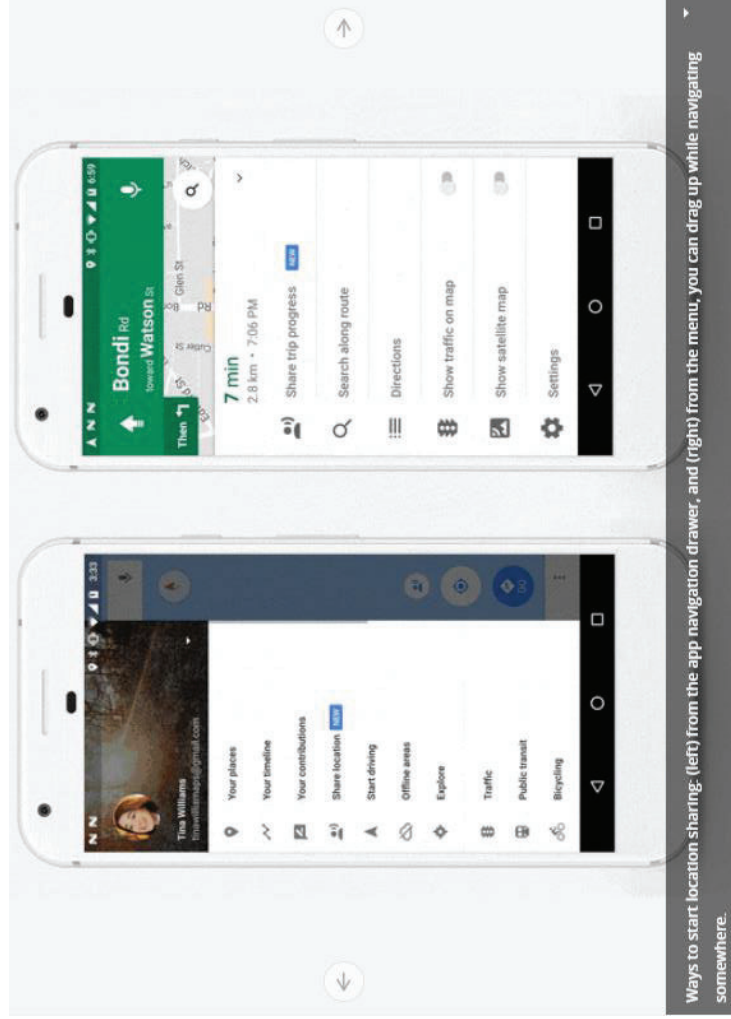
US9408055B2

ZTE

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

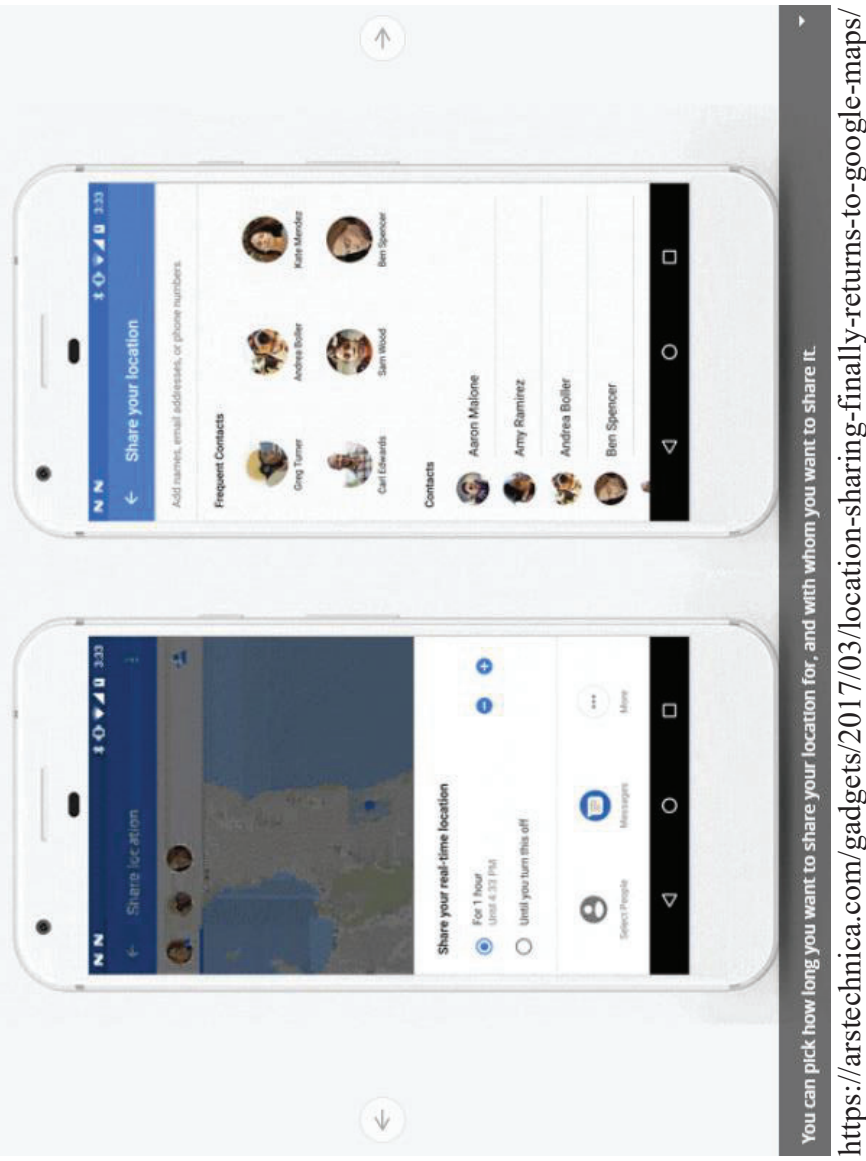
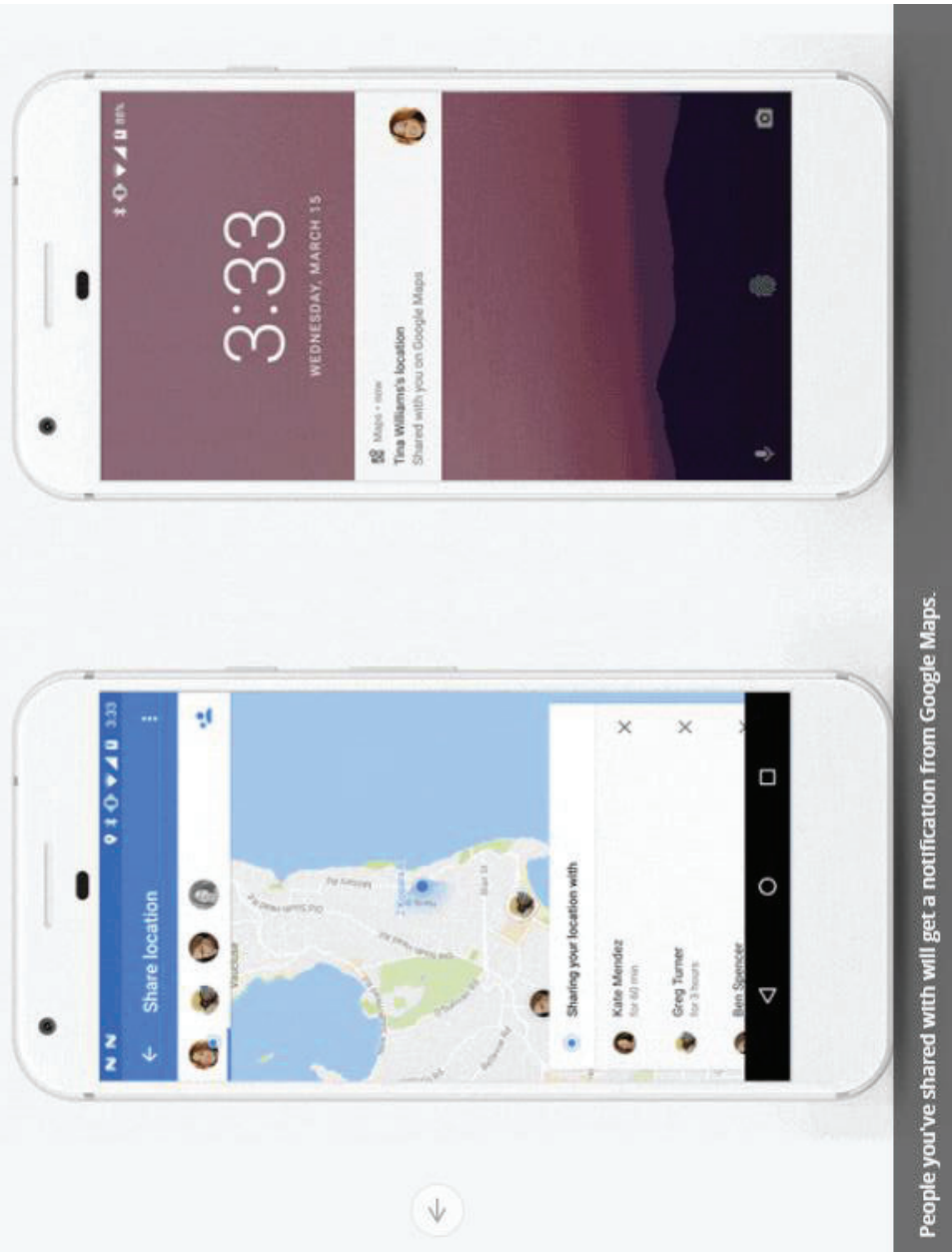


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



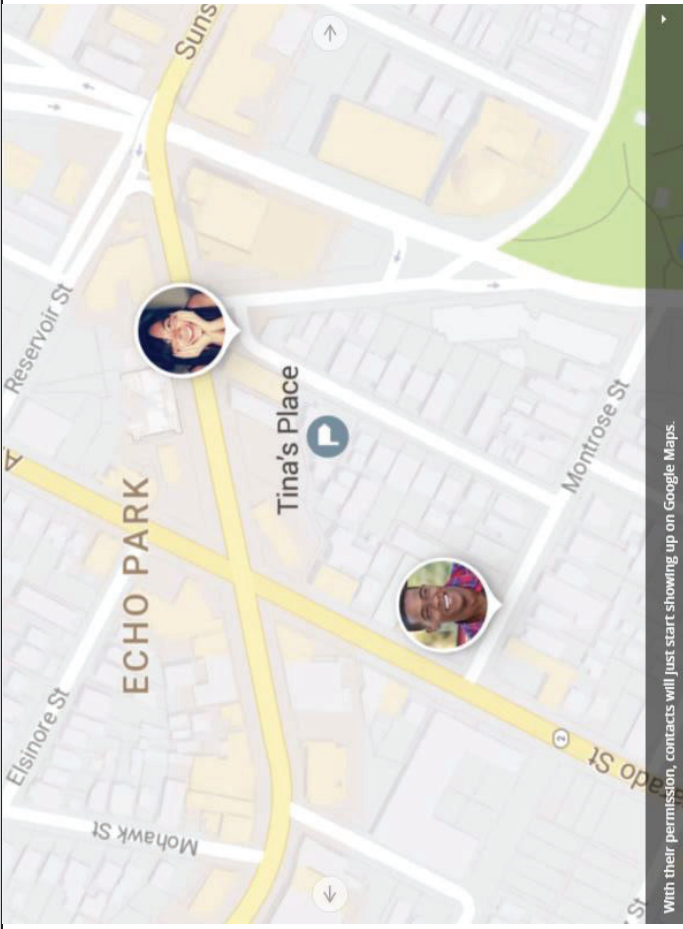
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Stop sharing

1. Open the Google Maps app .
2. Tap the Menu  > **Share location**.
3. Next to the person with whom you want to stop sharing, tap Remove .



Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
2. Set a driving destination. [Learn how to navigate to a place.](#)
3. After you start navigation, tap **More**  > **Share trip progress**.
4. Choose a person from the list.
5. Tap **Share**.
6. Location Sharing will stop when you reach your destination or stop navigating.

• To stop sharing before you arrive, tap **More**  > **Stop sharing**.

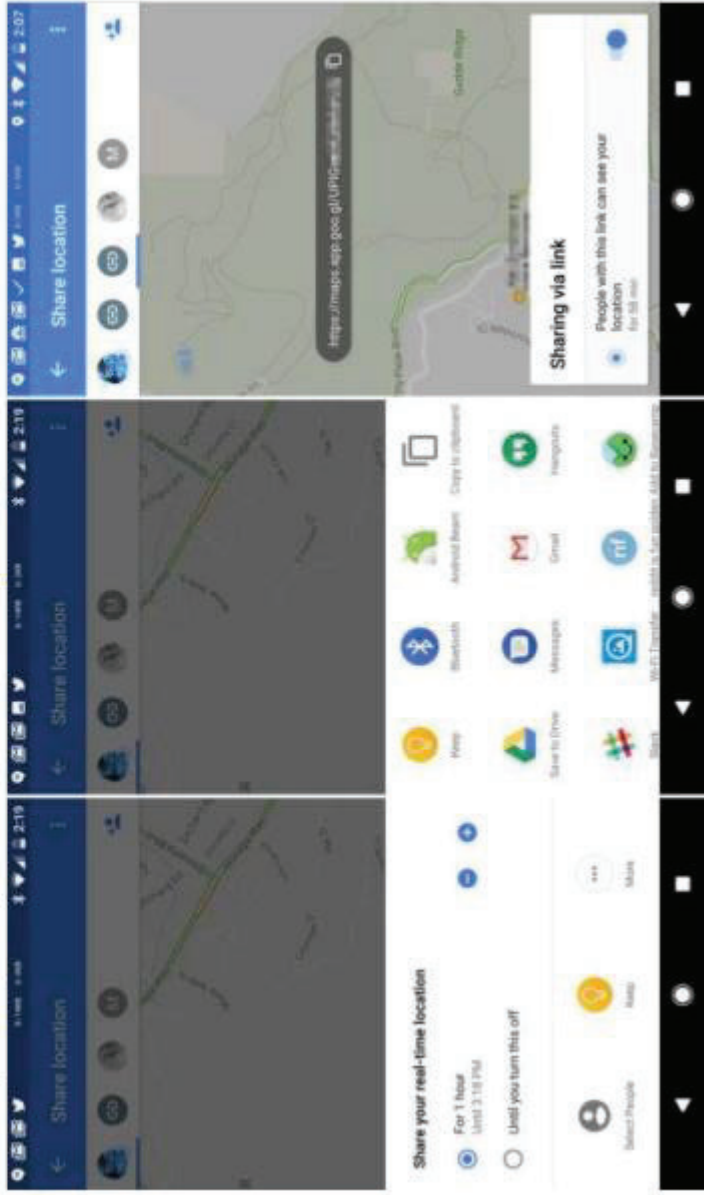
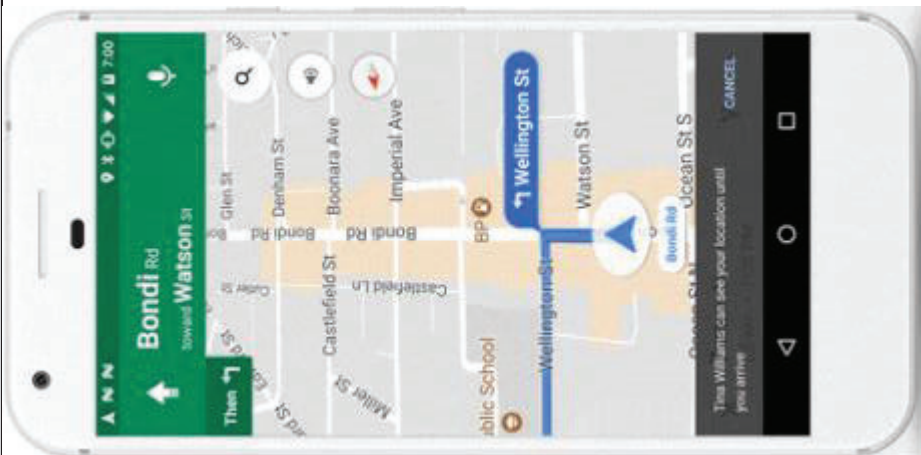


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

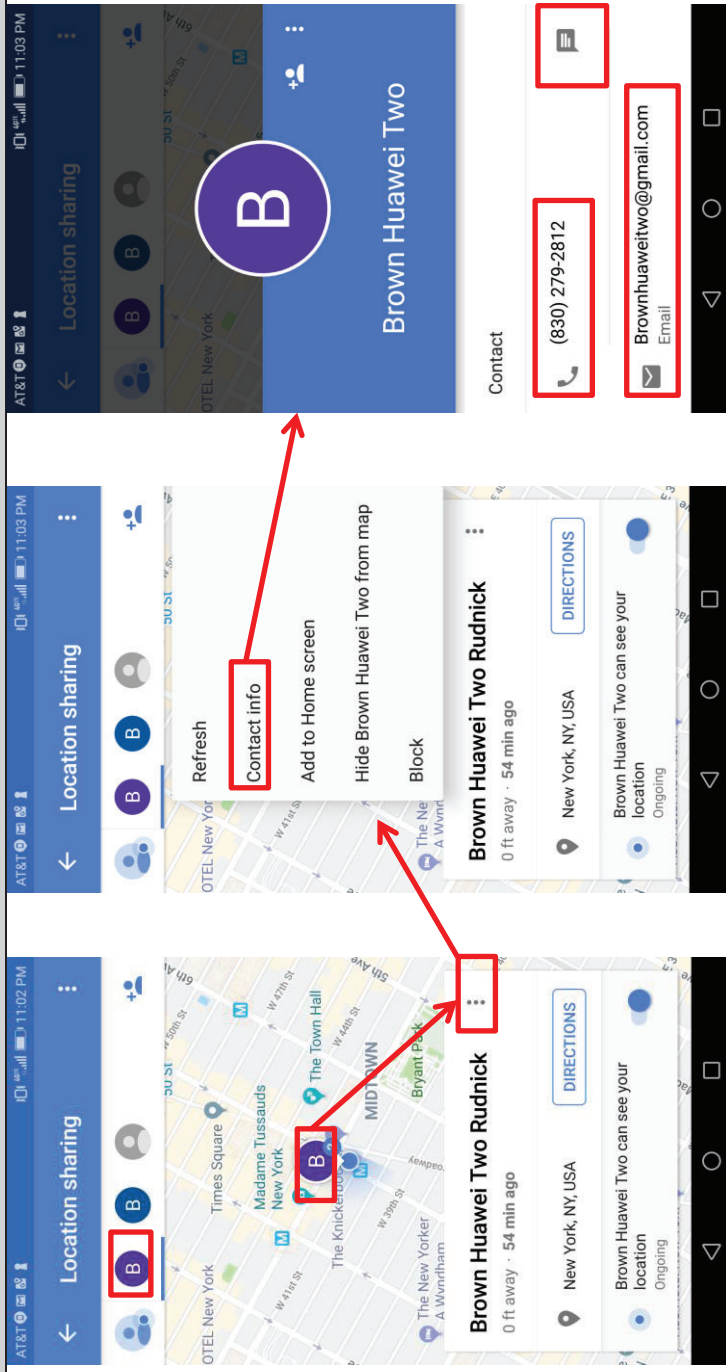


Exemplary Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1520 220 1587">ZTE</p> <p data-bbox="233 506 302 1587">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <pre data-bbox="358 1121 391 1562">public static LocationRequest create ()</pre> <p data-bbox="423 1073 448 1572">Create a location request with default parameters.</p> <p data-bbox="480 464 540 1572">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="565 1472 589 1551">Returns</p> <ul data-bbox="610 1289 634 1551" style="list-style-type: none">• a new location request <p data-bbox="651 306 678 1587">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-------------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><code>public Task<Location> getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task<LocationAvailability> getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|--------------------|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|--------------------|------------|

| | |
|---|--|
| <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> | |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="196 1520 220 1583">ZTE</p> <p data-bbox="233 302 261 1583">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p data-bbox="315 1283 342 1570">Public Constructors</p> <p data-bbox="415 1173 443 1560">public MapView (Context context)</p> <p data-bbox="505 968 532 1560">public MapView (Context context, AttributeSet attrs)</p> <p data-bbox="594 831 621 1560">public MapView (Context context, AttributeSet attrs, int defStyle)</p> <p data-bbox="683 852 711 1560">public MapView (Context context, GoogleMapOptions options)</p> <p data-bbox="745 428 773 1583">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[IG] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p>public void getMapAsync (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void onCreate (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[IG] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.


Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.

Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.

Selection with Markers:

<https://developers.google.com/maps/documentation/android-api/marker>

Queries with Geo Tagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app.
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share. If you don't see this, tap More.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Share your E. T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app.
2. Set a driving destination. Learn how to navigate to a place.
3. After you start navigation, tap More.
4. Choose a person from the list.
5. Tap Share.
6. Location Sharing will stop when you reach your destination or stop navigating.

To stop sharing before you arrive, tap More.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en>

Markers (adding location information to the link associated with the database):

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions() .position(PERTH) .draggable(true));</pre> |
| <p>[1H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anysevice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1596">ZTE</p> <p data-bbox="259 693 300 1228">ZTE Mobile Location Service System</p> <p data-bbox="332 903 365 1050">2004-01-31</p> <p data-bbox="430 1386 462 1564">I. Introduction</p> <p data-bbox="470 357 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 357 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1596">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.


Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.


Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.

Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.

Placing a Marker:

<https://developers.google.com/maps/documentation/android-api/marker>

based on queries with GeoTagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share  if you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E. T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More  > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More  > Stop sharing. |
| <p>2[A]. The method of claim 1, further comprising</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number. See claim 1, which is</p> |

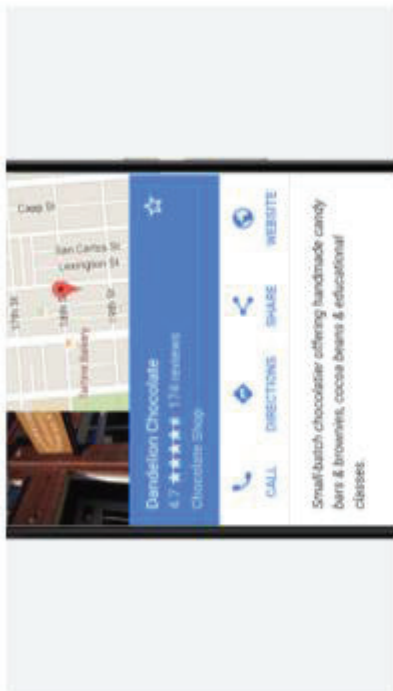
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2 performing, by the first device: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;</p> | <p>ZTE incorporated herein by reference in its entirety. See, e.g., Placing a Marker: https://developers.google.com/maps/documentation/android-api/marker based on queries with GeoTagging database: https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient Sharing a link: https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DDesktop&hl=en Users can use android maps to receive information associated with fixed locations such as restaurants, stadiums, transportation stations (e.g., bus and train stations), etc.</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Place details

Retrieve rich details about a place, including name, address, phone number, website link and more.

<https://developers.google.com/places/android-api/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Business phone numbers

Google is fantastic for tracking down business phone numbers. You can accomplish this in a number of ways, including:

- **type of business plus zip code:** Perhaps you don't know the name of the business you're looking for, but you have something in mind. Type in the business genre, for example, "pizza restaurant", then the zip code. Google will return local listings that include maps, reviews, and contact information (phone numbers, addresses, [website URLs](#), even email addresses if available).
- **type of business plus city:** Just like in the previous example, except you can substitute the name of a city for a zip code, i.e., "Seattle doctors".




<https://www.ifewire.com/google-phone-number-search-3481892>

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app .
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share . If you don't see this, tap More  > Share.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1596">ZTE</p> <p data-bbox="259 693 300 1228">ZTE Mobile Location Service System</p> <p data-bbox="332 903 365 1050">2004-01-31</p> <p data-bbox="430 1386 462 1564">I. Introduction</p> <p data-bbox="470 357 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="682 357 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="812 189 893 1596">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no 14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

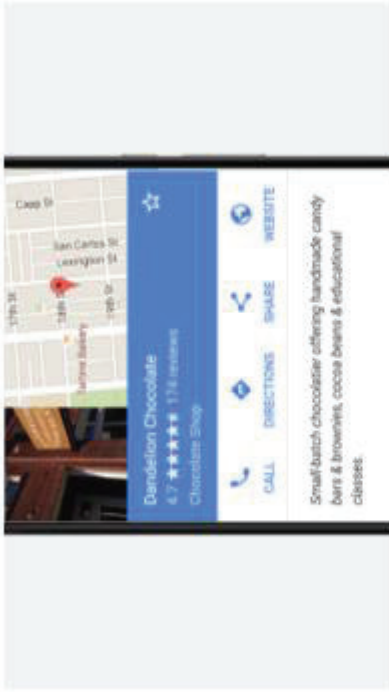
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| <p>[2B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.</p> | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 1 and 2[A], which are incorporated herein by reference in their entireties.</p> <p>A user can search for a fixed location using the maps application, which presents the user with a symbol(s) on the map corresponding to the fixed location's geographical coordinates. The user can then select the symbol(s) to see the fixed location's associated telephone number and select a "soft switch" having a symbol of a telephone handset to call the telephone number associated with the symbol</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Place details

Retrieve rich details about a place, including name, address, phone number, website link and more.

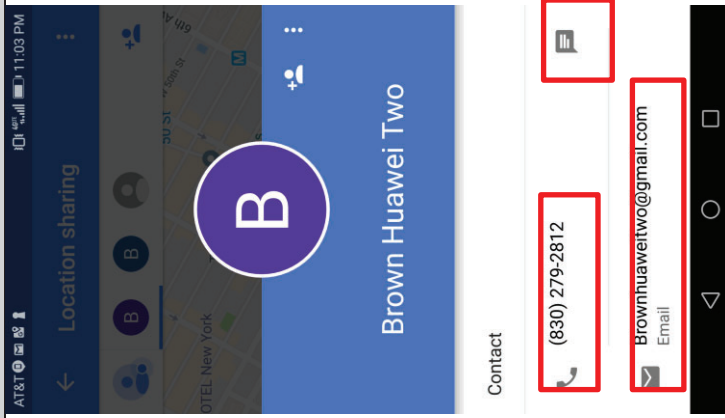
<https://developers.google.com/places/android-api/>

For example, the Accused Products include software that registers touch events with an interactive display, where touching a “call” button starts a phone call. Furthermore, these phone calls can merge multiple parties into a conference call.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



See also:
<https://www.wikihow.com/Conference-Call-on-an-Android>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| <p>3. The method of claim 1 wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p> | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p> <p>The Accused Products include an a TextToSpeech class that allows Android devices to convert any text displayed on the device to speech which is then recited using an electronic voice. The user of a first device sends a text message, image, video, or command to a second device using the Message app. The second device receives the communication and converts all text to speech when, e.g., extToSpeech is enabled.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Android allows you convert your text into voice. Not only you can convert it but it also allows you to speak text in variety of different languages.</p> <p>Android provides TextToSpeech class for this purpose. In order to use this class, you need to instantiate an object of this class and also specify the initListener. Its syntax is given below –</p> <pre>private EditText write; ttobj=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnIn @Override public void onInit(int status) { } });</pre> <p>In this listener, you have to specify the properties for TextToSpeech object , such as its language ,pitch e.t.c. Language can be set by calling setLanguage() method. Its syntax is given below –</p> <pre>ttobj.setLanguage(Locale.UK);</pre> <p>https://www.tutorialspoint.com/android/android_text_to_speech.htm</p> |
| <p>4[A]. The method of claim 1 wherein: the SMS messages include an Internet Protocol (IP) address of the first device;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the SMS messages include an Internet Protocol (IP) address of the first device. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>Android devices can send SMS messages that include an IP address of a device using Google Voice text message services. Google Voice is a voice over IP technology over internet protocol (IP) networks which use IP addresses to route data. Accordingly, the SMS messages include an IP address.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Send & get text messages

You can send text messages for free to U.S. and Canadian numbers using your Google Voice number. Texts sent using Google Voice will use Wi-Fi, or mobile data from your cell phone service plan if you're not connected to Wi-Fi. If you're outside the U.S. and are not using Wi-Fi, your cell phone company might charge you extra roaming fees to send a text.

Google Voice & Hangouts: You can choose to link Google Voice and Hangouts. If you do, you'll have to use Hangouts to send texts, not the Google Voice website or apps. To stop using Hangouts for texts, turn off Google Voice in Hangouts.

ANDROID COMPUTER IPHONE & IPAD





If you haven't yet, download the Google Voice app on your Android device. 

Send a text message

With the Google Voice website and apps, you can text people messages and photos and send texts to groups of people.

If you send a text longer than 160 characters to a non-Google Voice number, it will be sent as multiple messages.

Note: You can't send texts to five- or six-digit "short code" numbers.

1. On your Android device, open the Google Voice app .
2. Open the tab for Messages .
3. At the bottom, tap Add .
4. Enter a contact's name or phone number.
 - To create a group text message, add up to 30 names or phone numbers.
5. Enter your message, and tap Send .

To include an image with your message, tap Select image . If your image is bigger than 2MB, it'll be sent as a smaller file. But GIFs over 2MB won't send <https://support.google.com/voice/answer/1151116?co=GENIE.Platform%3DAndroid&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[4B] and the IP-based responses include respective IP addresses of the second devices.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] the IP-based responses include respective IP addresses of the second devices. See claim 4[A], which is incorporated herein by reference in its entirety.</p> <p>Android devices can reply to SMS texts via, e.g., Google Voice that also include IP address to route the message to its destination device via an IP network.</p> <p>Reply to a text message</p> <ol style="list-style-type: none"> 1. On your Android device, open the Google Voice app . 2. Open the tab for Messages . 3. Tap the text message you want to reply to. 4. At the bottom, enter your message, and tap Send . <p>https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>5. The method of claim 1, further comprising performing by the first device: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing by the first device: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>For example, Android devices include a LocationListener that allows the transmission and receipt of notification when a location of a device has changed.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

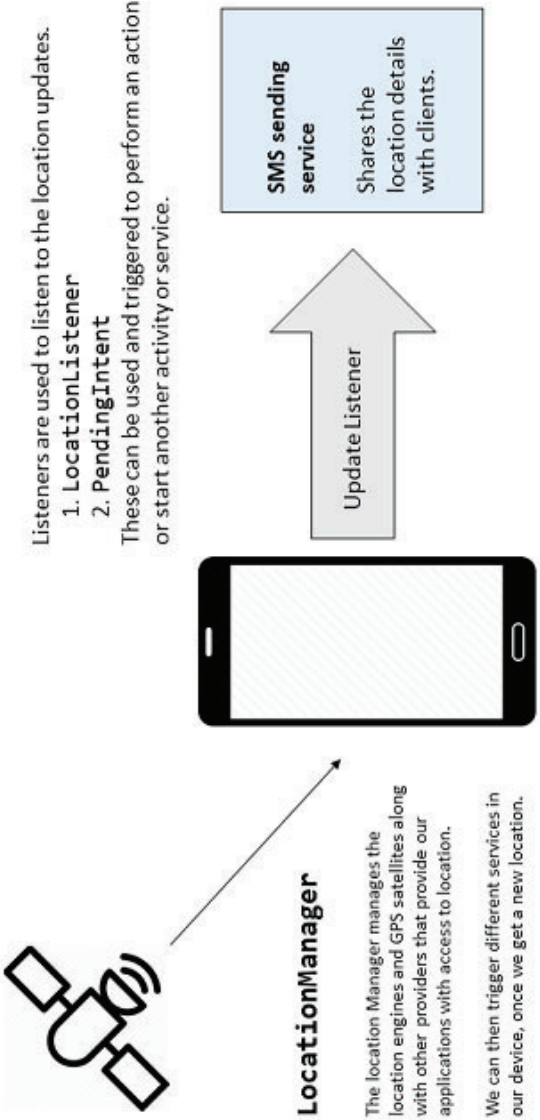
| | |
|--|---|
| <p>US9408055B2</p> <p>previous location of the first device, a passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time.</p> | <p>ZTE</p> <h2>LocationListener</h2> <p>public interface LocationListener android.location.LocationListener</p> <p>Used for receiving notifications from the LocationManager when the location has changed. These methods are called if the LocationListener has been registered with the location manager service using the <code>requestLocationUpdates(String, long, float, LocationListener)</code> method.</p> <h3>Developer Guides</h3> <p>For more information about identifying user location, read the Obtaining User Location developer guide.</p> <p>https://developer.android.com/reference/android/location/LocationListener.html</p> <p>Listeners are used to listen to the location updates.</p> <ol style="list-style-type: none"> 1. LocationListener 2. PendingIntent <p>These can be used and triggered to perform an action or start another activity or service.</p>  <p>LocationManager</p> <p>The location Manager manages the location engines and GPS satellites along with other providers that provide our applications with access to location.</p> <p>We can then trigger different services in our device, once we get a new location.</p> <p>SMS sending service</p> <p>Shares the location details with clients.</p> <p>Update Listener</p> <p>Figure 1: Demonstration of our requirement and work around. Explains what happens and what objects are being used in this demonstration.</p> |
|--|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




| US9408055B2 | ZTE |
|--|--|
| <p>6[A]. The method of claim 1, further comprising, by the first device: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;</p> | <p>https://www.codeproject.com/Articles/1040389/Ok-Android-Broadcast-my-location</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>For example, a first android device can receive second user selection of one or more of the symbols of second devices to join a communication network. For example, an Accused Product joins a communication network by signing-in using a Google Account. Signing-in to a Google Account with credentials (e.g., Gmail address or a registered username) which identify the user as being associated with the Google network of services. A user enters her credentials and transmits the credentials in the form of a request, e.g. an HTTP request. Moreover, each Google Account is associated with groups within certain Google services, e.g. Hangouts, Google+, Google Groups. Thus the Google network itself constitutes a group and additional examples of groups include Groups, Circles, Communities, as well as associations and shares created in Google services such as Android Messages, Contacts, Calendar, Gmail, Drive, Docs, Sheets, Slides, Keep, Photos, Contacts, Allo, Youtube, Waze, Hangouts, Google+, and other Google services or applications.</p> <p>Using Google Maps on an Accused Product, a user joins the Google network (and any associated groups within the Google network) by signing-on with a Google Account, as shown below:</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts [🔗](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap the Menu  > **Share location** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

<https://support.google.com/plus/answer/3302509?hl=en&co=GENIE.Platform%3DAndroid&oco=1>

Instead of selecting multiple contacts, a user can create a group and share to that group, as shown below:

Group contacts

You can organize the people and businesses in Contacts using labels.

Create a group

1. Go to [Google Contacts](#) [🔗](#).
2. On the left under "Labels," click **Create label**. (If you don't see "Labels," go to group contacts in old Contacts.)
3. Type a name, then click **OK**.

Add contacts to a group label

1. To select contacts, check the boxes next to their names.
2. In the top right, click Label . (If you don't see Label , go to group contacts in old Contacts.)
3. Choose the groups you want to add the contacts to. You'll see a checkmark appear next to the groups you chose.

<https://support.google.com/mail/answer/30970?hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Use Gmail to access your Google Account

If you use Gmail, you already have a Google Account. With a Google Account, you have access to free Google products like Drive, Docs, Calendar, and more [\[1\]](#).

To sign in to your Google Account (or any Google product):

1. Go to the sign in page of the product (for Google Accounts it is myaccount.google.com [\[2\]](#)).
2. Enter your Gmail username (everything that appears before '@gmail.com').
3. Enter your password.

https://support.google.com/accounts/answer/76194?hl=en&ref_topic=7189242


Sign in or out of Google+

To sign in, enter your Google username and password. If you're on a public or shared device, don't forget to sign out so no one else can get in to your Google+ account.

Tip: If you don't know your username or password, you can get help signing in to your Google account [\[3\]](#).

Sign in to your Google+ account for the first time

Android app

1. On your Android phone or tablet, open the Google+ app .
2. Tap the account you want to use.

Computer

1. On your computer, open Google+ [\[4\]](#).
2. Enter your username and password and click **Sign in**.

https://support.google.com/plus/answer/1301225?hl=en&ref_topic=3049735

The sign-in process includes a message because a request message (e.g. HTTP[S] request) is sent from the Accused Product, as shown below:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Connect to the Online Service

The example below shows how to connect to a Google server. Since Google uses the industry standard OAuth2 protocol to authenticate requests, the techniques discussed here are broadly applicable. Keep in mind, though, that every server is different, and that every server is different. You may find yourself needing to make minor adjustments to these instructions to account for your specific situation.

The Google APIs require you to supply four values with each request: the API key, the client ID, the client secret, and the auth key. The first three come from the Google API Console website. The last is the string value you obtained by calling `AccountManager.getAuthToken()`. You pass these to the Google Server as part of an HTTP request.

```
URL url = new URL("https://www.googleapis.com/tasks/v1/users/{me}/lists?key=" + your_api_key);
URLConnection conn = (URLConnection) url.openConnection();
conn.setRequestProperty("client_id", your_client_id);
conn.setRequestProperty("client_secret", your_client_secret);
conn.setRequestProperty("Authorization", "OAuth " + token);
```

If the request returns an HTTP error code of 401, then your token has been denied. As mentioned in the last section, the most common reason for this is that the token has expired. The fix is simple: call `AccountManager.invalidateAuthToken()` and repeat the token acquisition dance one more time.

Because expired tokens are such a common occurrence, and fixing them is so easy, many applications just assume the token has expired before even asking for it. If renewing a token is a cheap operation for your server, you might prefer to call `AccountManager.invalidateAuthToken()` before the first call to `AccountManager.getAuthToken()`, and spare yourself the need to request an auth token twice.

<https://developer.android.com/training/id-auth/authenticate.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

2. In the activity's `onActivityResult` method, retrieve the sign-in result with `getSignInResultFromIntent`.

```

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    // Result returned from launching the Intent from GoogleSignInApi.getSignInIntent(...);
    if (requestCode == RC_SIGN_IN) {
        GoogleSignInResult result = Auth.GoogleSignInApi.getSignInResultFromIntent(data);
        handleSignInResult(result);
    }
}
    
```

SignInActivity.java

After you retrieve the sign-in result, you can check if sign-in succeeded with the `isSuccess` method. If sign-in succeeded, you can call the `getSignInAccount` method to get a `GoogleSignInAccount` object that contains information about the signed-in user, such as the user's name.

```

private void handleSignInResult(GoogleSignInResult result) {
    Log.d(TAG, "handleSignInResult:" + result.isSuccess());
    if (result.isSuccess()) {
        // Signed in successfully, show authenticated UI.
        GoogleSignInAccount acct = result.getSignInAccount();
        mStatusTextView.setText(getString(R.string.signed_in_fmt, acct.getDisplayName()));
        updateUI(true);
    } else {
        // Signed out, show unauthenticated UI.
        updateUI(false);
    }
}
    
```

SignInActivity.java

You can also get the user's email address with `getEmail`, the user's Google ID (for client-side use) with `getId`, and an ID token for the user with `getIdToken`. If you need to pass the currently signed-in user to a backend server, send the ID token to your backend server and validate the token on the server.

<https://developers.google.com/identity/sign-in/android/sign-in>

[6B] and receiving user input assigning the one or more

ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 1 and 6[A], which are incorporated herein by

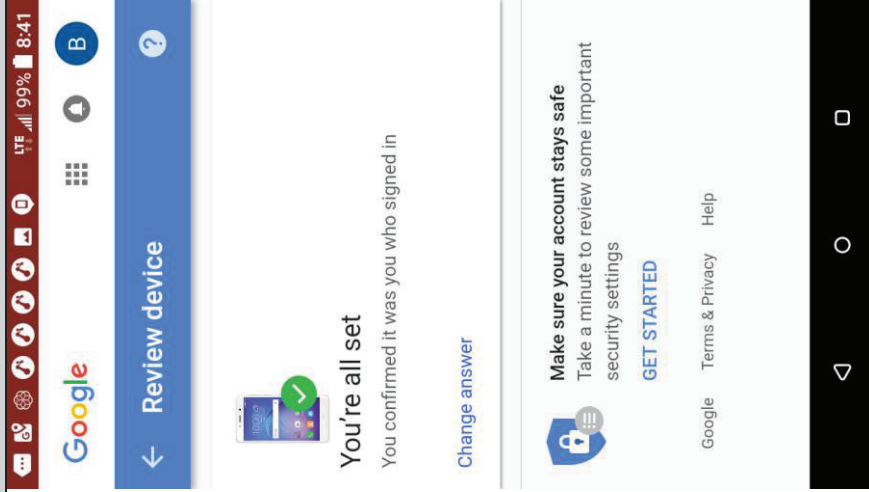
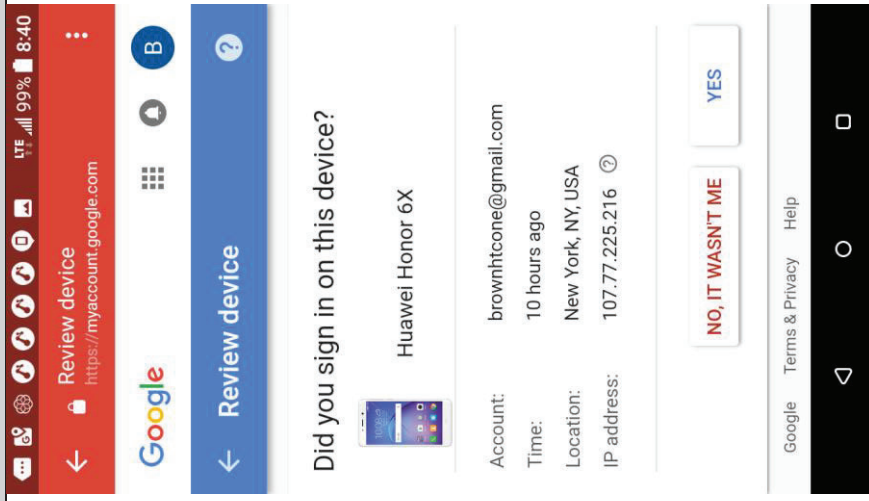
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2 second devices corresponding to the second selected one or more symbols to a sub-net.</p> | <p>ZTE reference in their entireties.</p> <p>For example, an Accused Product is joined to a group (e.g. Google services such as the Play and Play Protect services) using a Google ID (e.g, an email address). Alternatively, any device with a browser or a device-location application can sign on to a Google service. The Google ID corresponds to one or more groups. These groups include, for example, the Google network, “friends” or “families” as defined by association with device-location features and applications (e.g., Find My Device, Hangouts (including Allo and Duo), Google+, Maps, and Chrome.)</p> <p>In establishing these groups, each Accused Product receives messages from other devices, e.g. other Accused Products, related to forming a group.</p> <p>In other examples, a user participates in groups within services of the Google network, e.g. Android Device Manager, Google Maps, Google Contacts, Google+, Google Groups, among others. Using Android Device Manager, a group of devices is associated with the sign-in credentials. Each member of the group is represented by a symbol. Using Google Maps, as integrated with Google Contacts and Messages, a user shares her location with a group by sending her location to a group or multiple contacts in a group message. Groups are also established in many other services across Google’s network, as shown by Google+ and Google Groups. In fact, Google+ included the location-sharing feature prior to its inclusion in Google Maps.</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



e.g., above, the device receives a message indicating that the second device has joined the group, i.e. the google account.

Google Share Location

Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE <p>Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Use circles on Google+

You can use circles to control who you share posts with. Add people to a circle, then share a Collection or a post with that circle so those people can see it.

You can add someone to a circle even if they don't follow you, so they'll be able to see posts you share with that circle, but might not see those posts in their home stream.

When you add someone to a circle, they might get a notification letting them know you added them to one of your circles. They can see anything you share with that circle, including posts you shared with that circle before you added them. If that person adds you to their circles, posts you've shared with them may appear on their Home page.

You can see people you have in your circles by choosing **People** in your Navigation menu.

The people and pages you add to circles are publicly visible by default, but you can change who can see the people and pages you add. Depending on your settings, information about the people or pages you've added to circles may also appear in shared endorsements.

You can add someone to a circle or remove someone from a circle at any time.

Create a circle

1. In the Navigation menu, click or tap **People**.
2. Click or tap **Following**.
3. Click or tap **NEW CIRCLE**.
4. Name your circle, then click or tap **CREATE**.

https://support.google.com/plus/answer/1301225?hl=en&ref_topic=3049735

Find and join Communities

You can connect with people who share your interests by joining a public or private Community in Google+. Stay up to date on what's happening in your Communities by seeing Community posts in your stream and getting notifications.

COMPUTER ANDROID IPHONE & IPAD

<https://support.google.com/plus/answer/6320394>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Find & join a group

You can join a Google Group to have discussions about a topic or communicate with your team, organization, class, or other group. Some groups grant immediate membership, but others need you to request membership first.

Join a group

1. Sign in to Google Groups [G2](#). Learn how to join if you don't have a Google Account.

Use groups to share content

Google Groups makes it easy to share your Google documents, sites, videos, and calendars with multiple people. As you add new members to your groups, they'll automatically gain access to content you previously shared with that group.

For example, if you create a group with the address "marketing-team@your_domain.com" and add five members, you can instantly share a Google document with them, just by sharing the document with the group's address. If you later add another member, that member automatically inherits permission to access the document or any other content you shared with the group. Similarly, if you remove a member from a group, that person no longer has access to any content you shared with the group.

<https://support.google.com/a/answer/167101?hl=en>

In addition to the communication described above with respect to the Android Device Manager and Google Maps applications, messages can be received over messaging applications integrated in the Accused Products, such as Google Messages or Google Hangouts.

Exemplary Support for Google Maps:





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap **More**  **> Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap **More**  **> Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap **Menu**  **> Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon **> More**  **> Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap **More** .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD


Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1

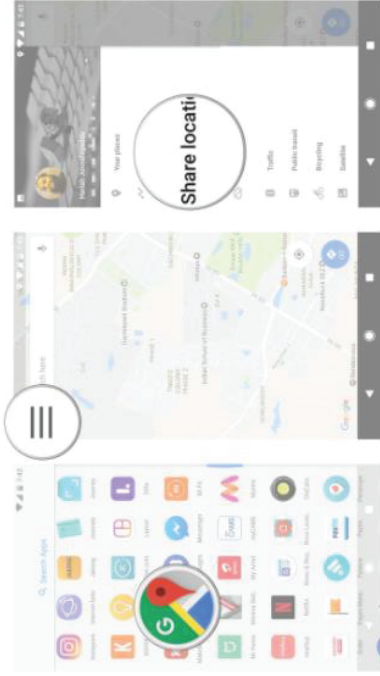
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

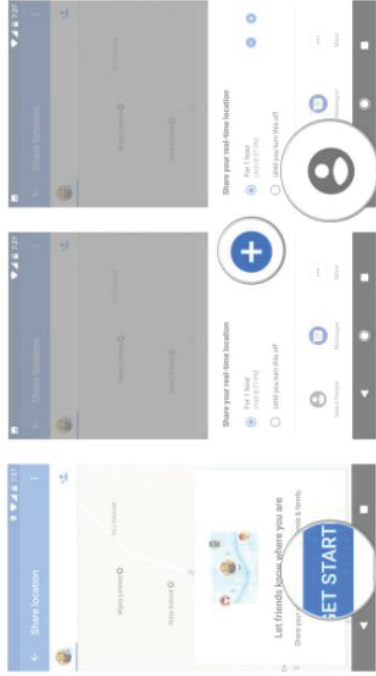
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



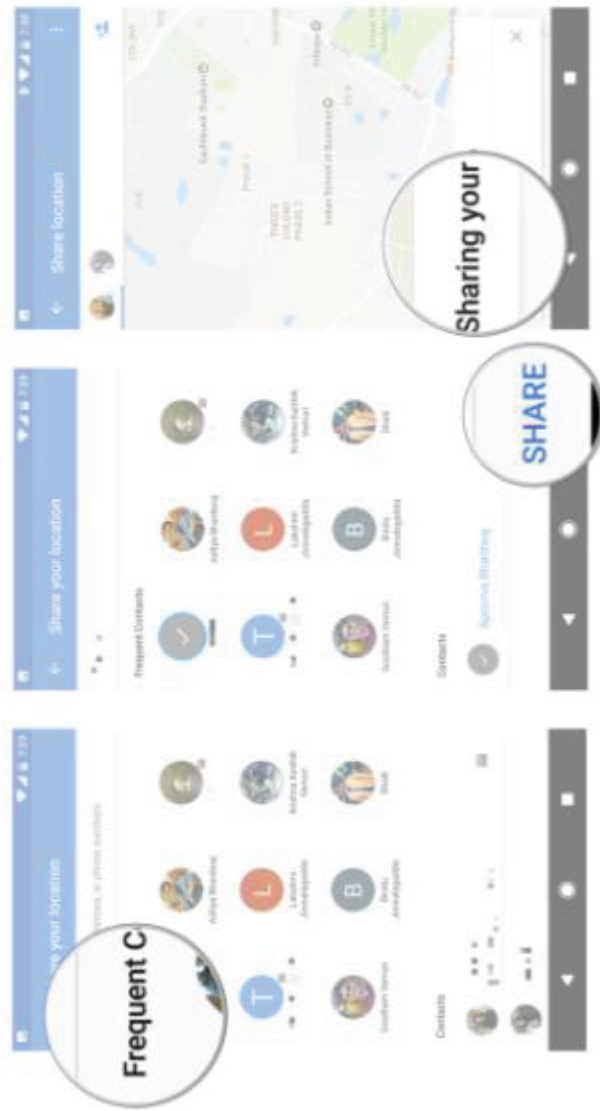
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

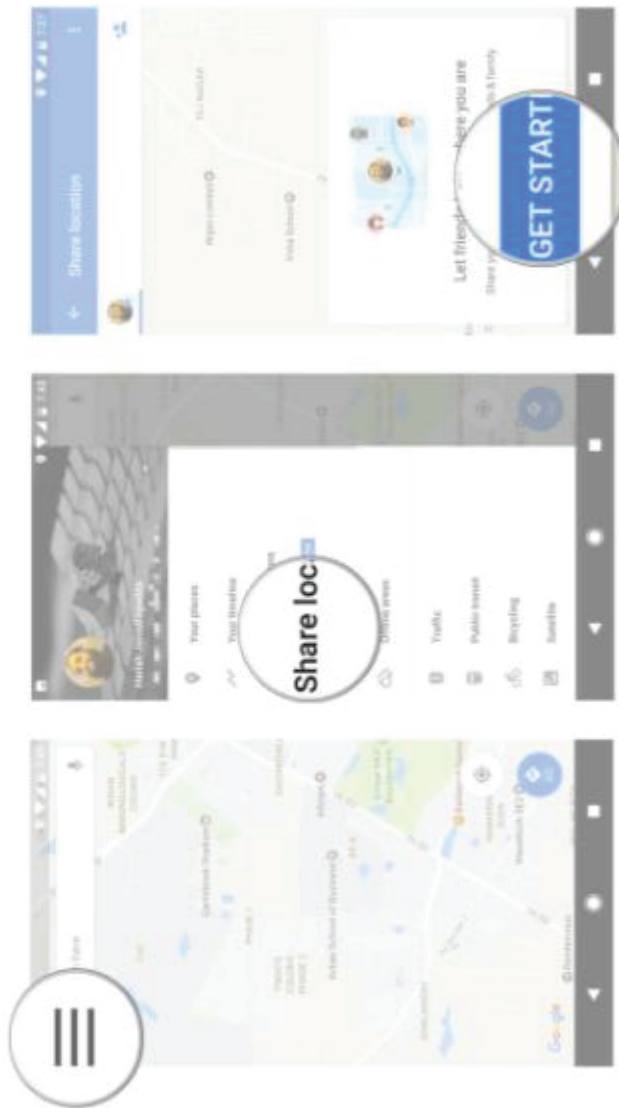
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



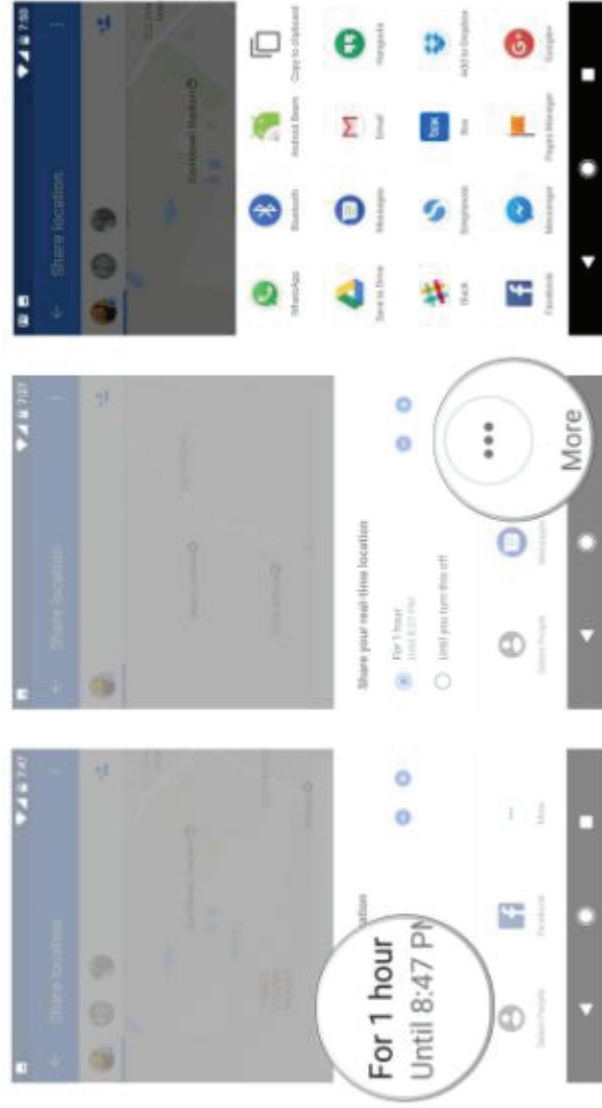
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

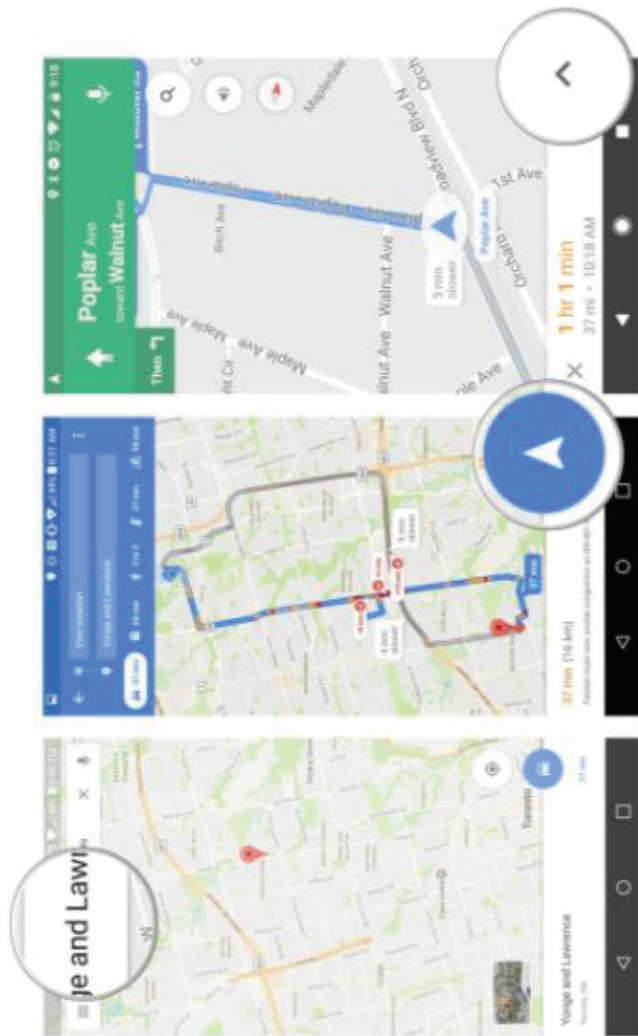
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



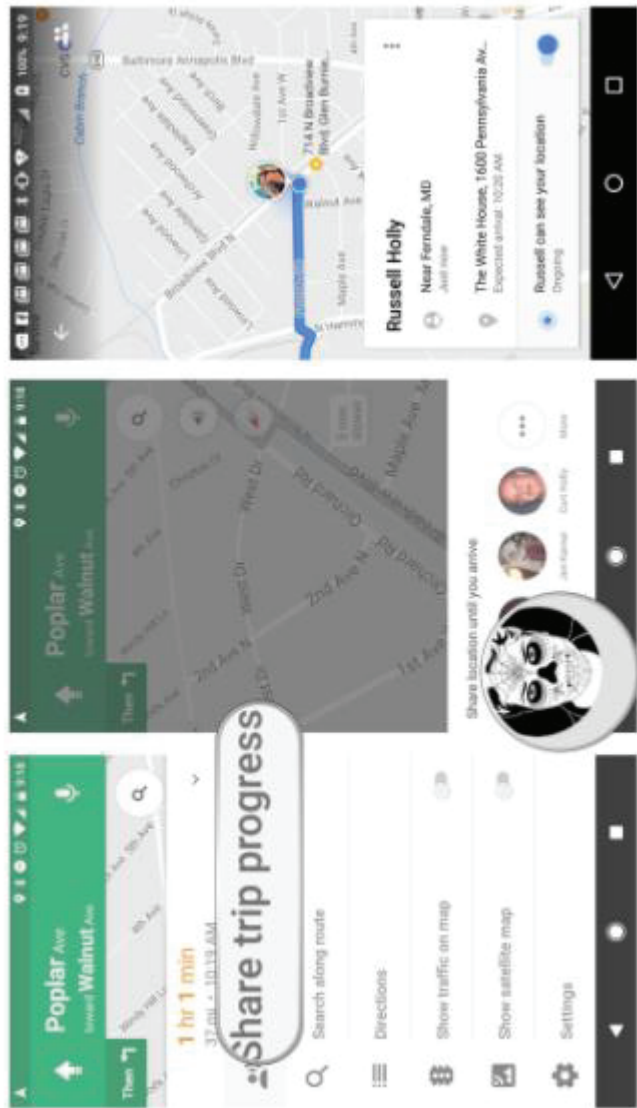
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



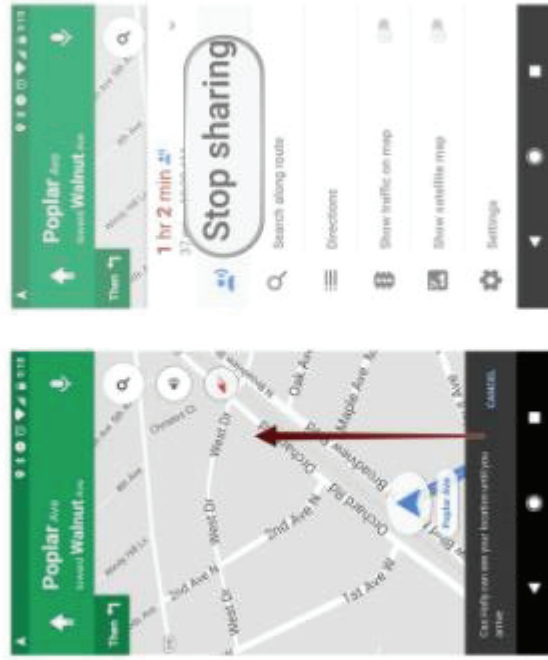
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.

Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.

Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.


Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

As shown below, a group may also be defined within Google Contacts.

See your contacts

1. Open your device's Contacts app .
2. Tap Menu .

- **See contacts by label:** Choose a label from the list.
- **See contacts for another account:** Tap Down arrow  > pick an account.
- **See the contacts for all your accounts:** Choose **All contacts**.

Tip: If you have multiple contacts with the same information, the information will be grouped into one contact.

- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <h3 data-bbox="235 1197 284 1564">Label your contacts</h3> <p data-bbox="300 1113 332 1564">You can group contacts together using labels.</p> <ol data-bbox="349 378 600 1564" style="list-style-type: none"><li data-bbox="349 1155 381 1564">1. Open your device's Contacts app .<li data-bbox="389 1218 422 1564">2. Tap Menu  > Create label.<li data-bbox="430 1218 462 1564">3. Enter a label name and tap OK. <ul data-bbox="479 378 600 1564" style="list-style-type: none"><li data-bbox="479 861 511 1564">• Add one contact to a label: Tap Add contact  > choose a contact.<li data-bbox="519 378 600 1564">• Add multiple contacts to a label: Tap Add contact  > touch and hold a contact > tap the other contacts > tap Add. https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711 <h3 data-bbox="649 1155 698 1564">Share your contacts</h3> <ol data-bbox="714 567 885 1564" style="list-style-type: none"><li data-bbox="714 1113 747 1564">1. Open your device's Contacts app .<li data-bbox="755 1239 787 1564">2. Tap a contact in the list.<li data-bbox="795 1260 828 1564">3. Tap More  > Share.<li data-bbox="844 1050 885 1564">4. Choose how you want to share the contact. <p data-bbox="893 567 933 1585">https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

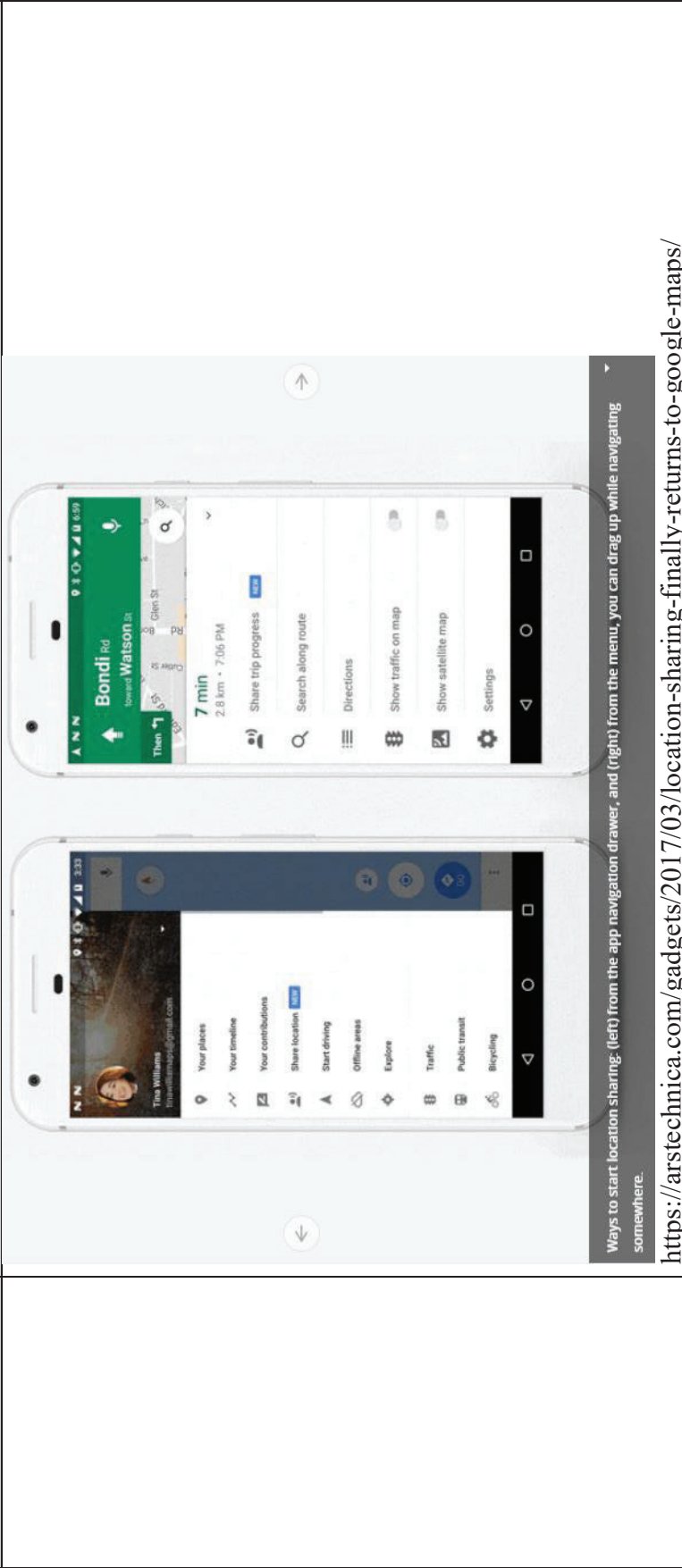


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

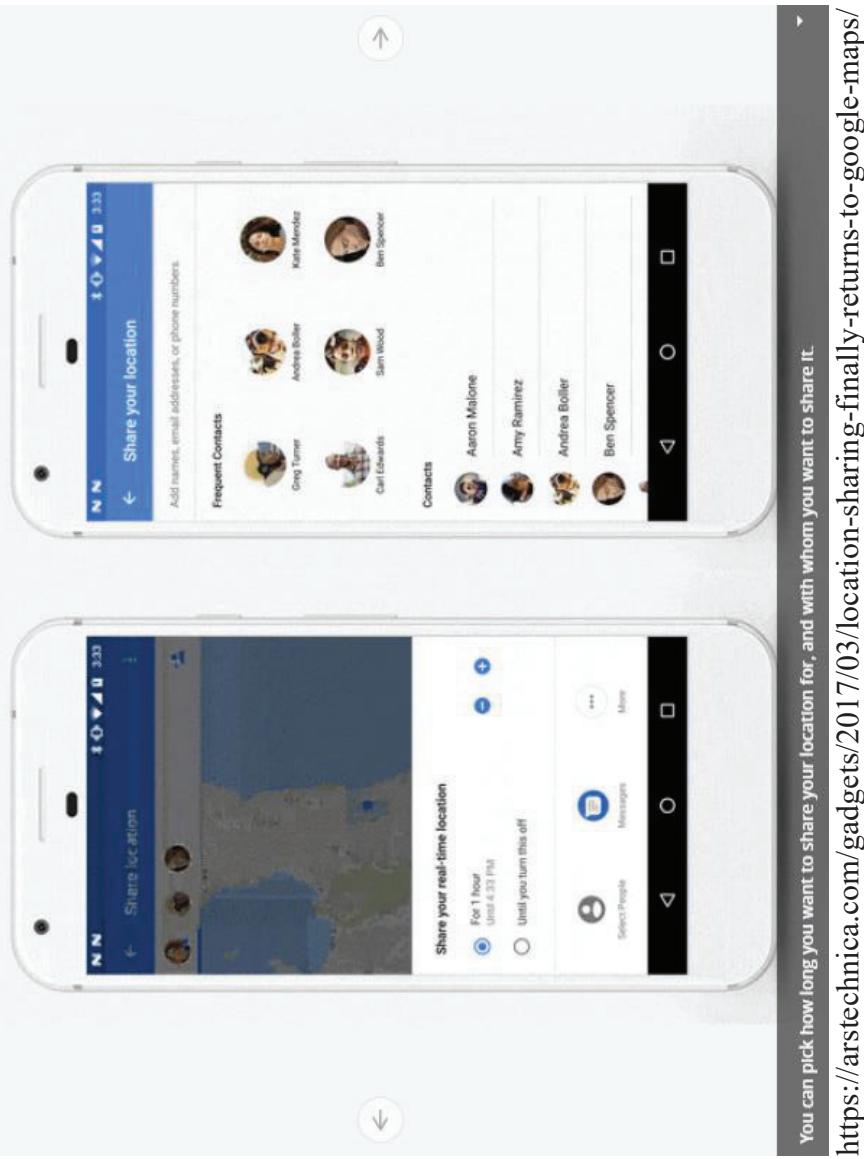
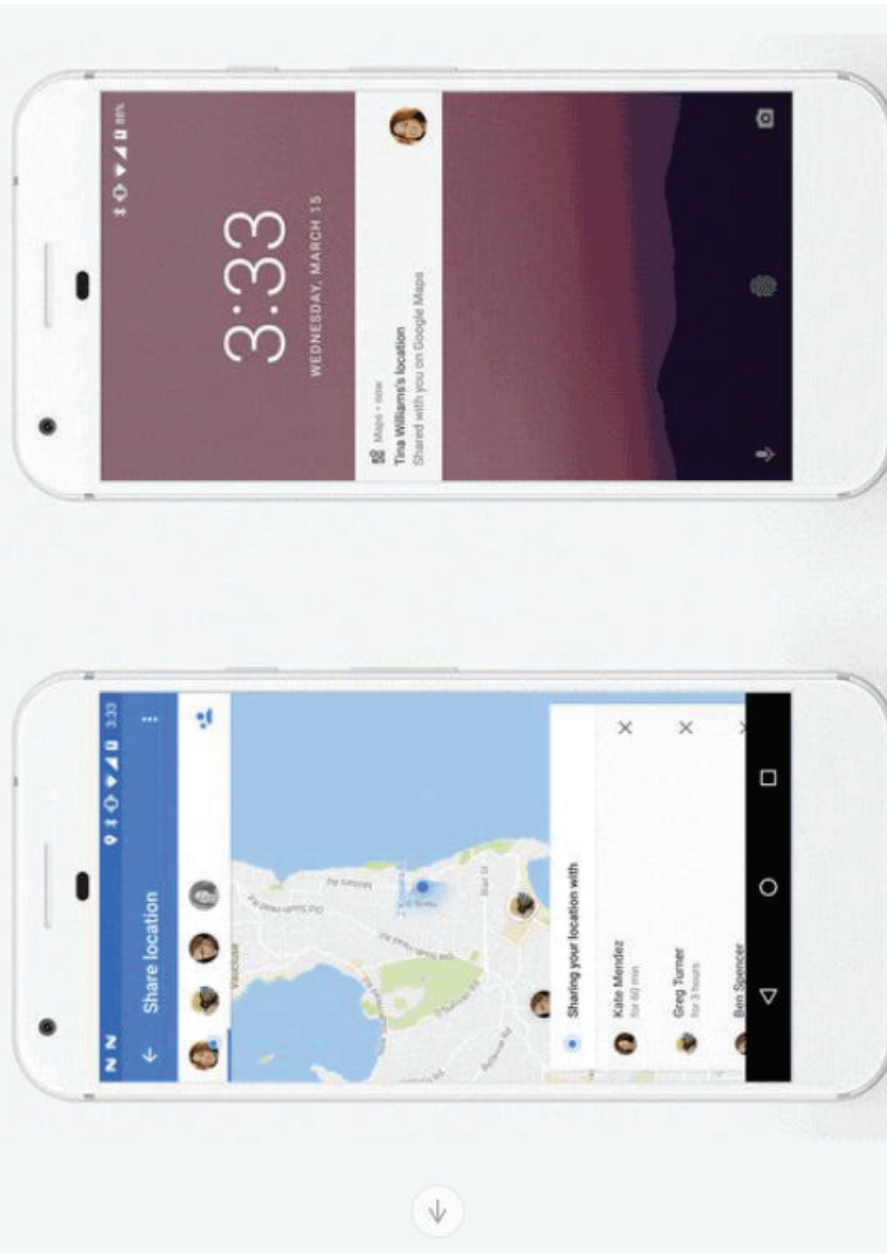


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



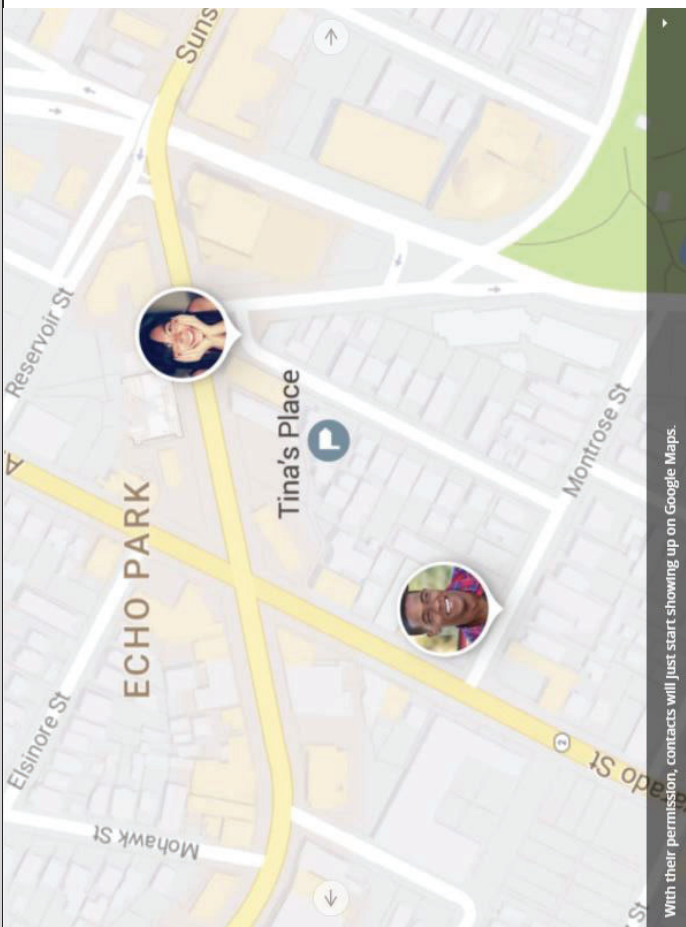
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

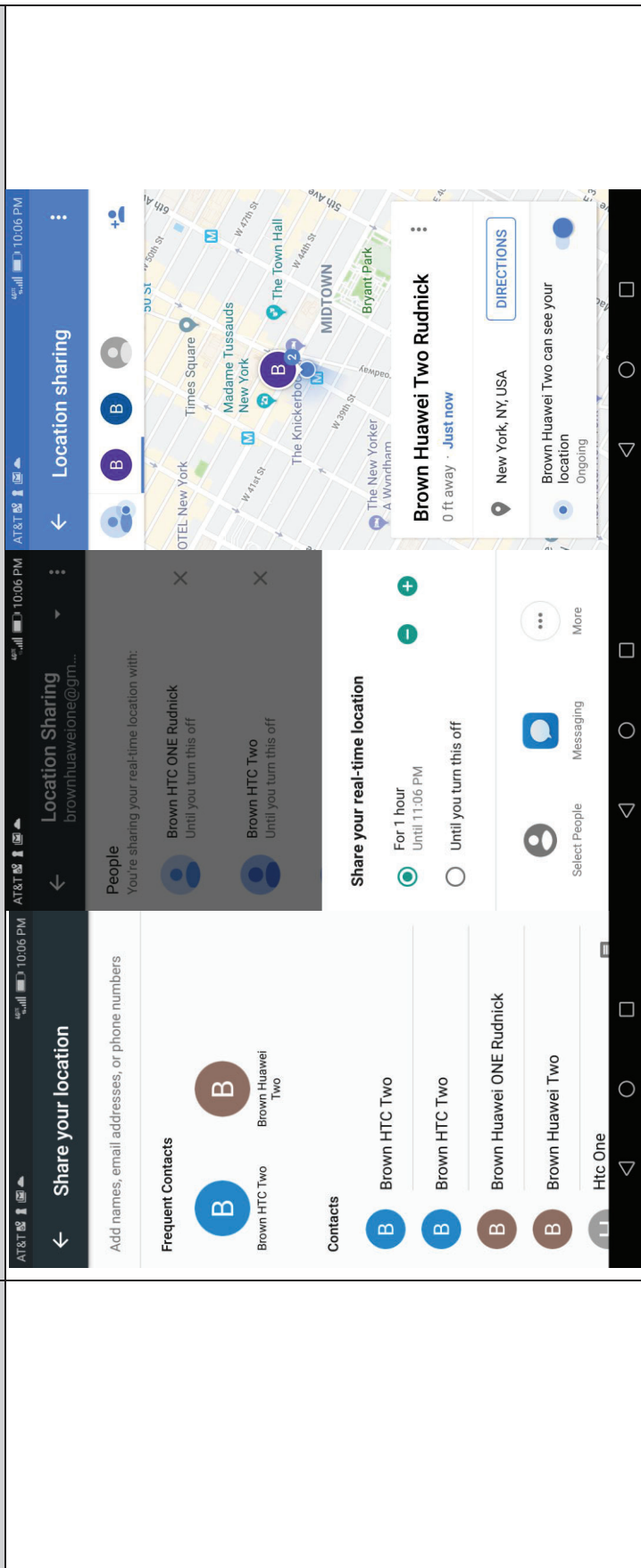


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <h2 data-bbox="235 1081 292 1575">Contacts Provider</h2> <p data-bbox="324 630 592 1575">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="617 1260 641 1575">This guide describes the following:</p> <ul data-bbox="665 724 852 1575" style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p data-bbox="860 609 885 1585">https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contact's _id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|---|-------------|--|-----------|--|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | One of: <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. |
| https://developer.android.com/guide/topics/providers/contacts-provider.html | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

B-276

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1- release/src/com/android/contacts/list/ContactsRequest.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-279

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *   it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113 String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : ""))
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));
131     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpParams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

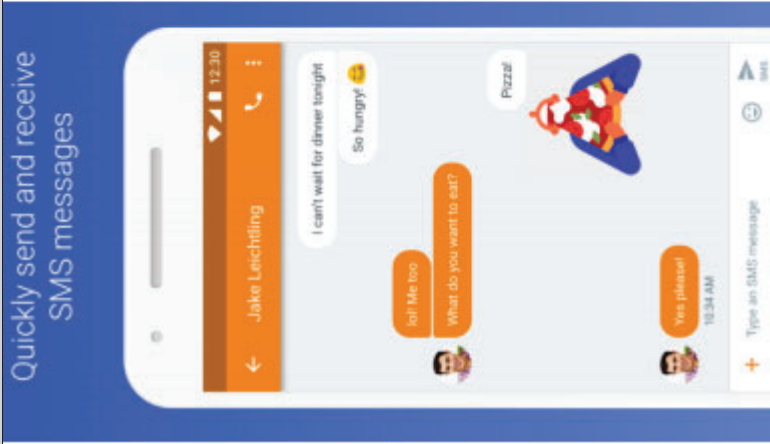
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> <p>7[A]. The method of claim 6, further comprising performing, by the first device: receiving user selection of the sub-net;</p> | <p>ZTE</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving user selection of the sub-net. See claims 1 and 6, which are incorporated herein by reference in their entireties.</p> <p>A user of the Accused Products selects a set of second devices assigned to a group to establish a group messaging session for sharing voice, text, photographs, or video via, e.g., Android's messaging app, Google Hangouts, etc.. For example, the user can select a group name corresponding to the group in order to perform a subsequent action associated with the group such as conducting a conference communication session.</p> <p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages</p> <p>Enhanced features: On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p>https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Google Hangouts

<https://hangouts.google.com/>

Hangouts bring **conversations** to life with photos, emoji, and even **group** video calls for free. Connect with friends across computers, **Android**, and Apple devices. You visited this page.

<https://www.google.com/search?q=android+group+conversation&oq=android+group+con&aqs=chrome.0.0j69i57j0l4.10109j0j1&sourceid=chrome&ie=UTF-8>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER ANDROID IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add ➕ > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send ➤.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

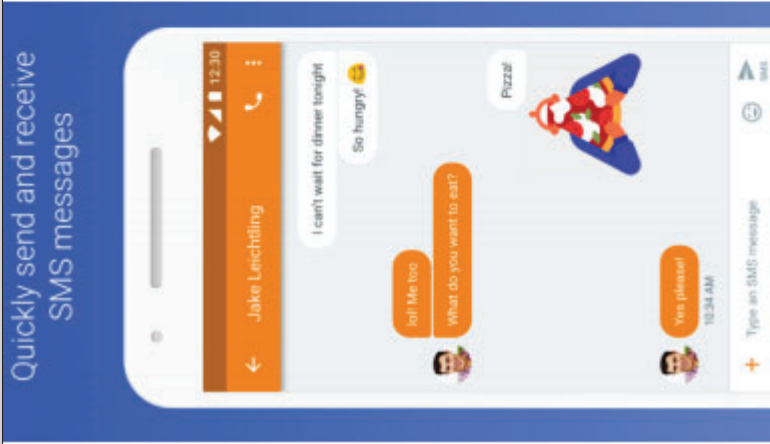
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> <p>[7B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.</p> | <p>ZTE</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 1, 6, and 7[A], which are incorporated herein by reference in their entireties.</p> <p>A user of the Accused Products selects a set of second devices assigned to a group to establish a group messaging session for sharing voice, text, photographs, or video via, e.g., Android's messaging app, Google Hangouts, etc.. For example, the user can select a group name corresponding to the group in order to perform a subsequent action associated with the group such as conducting a conference communication session.</p> <p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages</p> <p>Enhanced features: On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more</p> <p>https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en</p> |
|--|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Google Hangouts

<https://hangouts.google.com/>

Hangouts bring **conversations** to life with photos, emoji, and even **group** video calls for free. Connect with friends across computers, **Android**, and Apple devices. You visited this page.

<https://www.google.com/search?q=android+group+conversation&oq=android+group+con&aqs=chrome.0.0j69i57j0l4.10109j0j1&sourceid=chrome&ie=UTF-8>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER ANDROID IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add ➕ > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send ➤.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|--|
| <p>8. The method of claim 1, wherein the first device is a cellular phone or a personal digital assistant (PDA).</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claim 1, which is incorporated herein by reference in its entirety.</p> <p>Upon information and belief, the Accused Products are forms of cellular devices or PDAs in that the functionality of a PDA has been subsumed into smartphones, tablets, and portable media players having the functionalities of a PDA that include cellular transceivers to enable cellular communications. To the extent that it is necessary, AGIS submits that the Accused Products meet the claim limitation “the first device is a personal digital assistant (PDA) or a cellular phone” under the doctrine of equivalents. For example, U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes ZTE Android phones as smartphones for which U.S. Cellular provides cellular services. https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod1060051</p> |
| <p>9. The method of claim 1, further comprising performing, by the first device: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the at least one second device. See claims 1 and 6-7, which are incorporated herein by reference in their entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Google Maps, a user can select a device’s symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the display specifying an action and, based thereon, initiating a phone call or phone conference with the at least one second device. See claims 1 and 6-7, which are incorporated herein by reference in their entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Google Maps, a user can select a device’s symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


| | |
|---|--|
| <p>US9408055B2</p> <p>conference with the at least one second device.</p> | <p>ZTE</p> <h2>Group contacts</h2> <p>You can organize the people and businesses in Contacts using labels.</p> <h3>Create a group</h3> <ol style="list-style-type: none">1. Go to Google Contacts.2. On the left under "Labels," click Create label. (If you don't see "Labels," go to group contacts in old Contacts.)3. Type a name, then click OK. <p>Add contacts to a group label</p> <ol style="list-style-type: none">1. To select contacts, check the boxes next to their names.2. In the top right, click Label . (If you don't see Label, go to group contacts in old Contacts.)3. Choose the groups you want to add the contacts to. You'll see a checkmark appear next to the groups you chose. <h3>Google Hangouts</h3> <p>https://hangouts.google.com/</p> <p>Hangouts bring conversations to life with photos, emoji, and even group video calls for free. Connect with friends across computers, Android, and Apple devices.</p> <p>You visited this page.</p> <p>https://www.google.com/search?q=android+group+conversation&og=android+group+con&aqs=chrome.0.0j69i57j0l4.10109j0j1&sourceid=chrome&ie=UTF-8</p> |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products



| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <ol style="list-style-type: none"> 1 Phone the first person. 2 After the call connects and you complete a few pleasantries, touch the Add Call icon. <div style="text-align: center;">  </div> <p>The Add Call icon is shown. After touching that icon, or a similar icon, the first person is put on hold.</p> 3 Dial the second person. <p>You can use the dialpad or choose the second person from the phone's address book or the recent calls log.</p> <p>Say your pleasantries and inform the party that the call is about to be merged.</p> 4 Touch the Merge or Merge Calls icon. <div style="text-align: center;">  </div> <p>The two calls are now joined: The touchscreen says <i>Conference Call</i>, and the End Last Call icon appears. Everyone you've dialed can talk to and hear everyone else.</p> 5 Touch the End Call icon to end the conference call. <p>All calls are disconnected.</p> <p>http://www.dummies.com/consumer-electronics/smartphones/droid/how-to-make-a-conference-call-on-an-android-phone/</p> |
| <p>10. The method of claim 3, wherein the video comprises a video clip or a video transmission.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claim 3, which is incorporated herein by reference in its entirety.</p> |

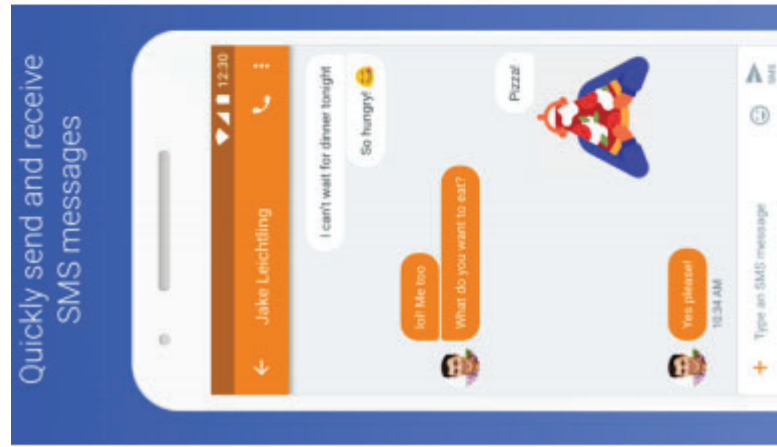
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages

• **Enhanced features:** On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more
<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Google Hangouts https://hangouts.google.com/  Hangouts bring conversations to life with photos, emoji, and even group video calls for free. Connect with friends across computers, Android, and Apple devices. You visited this page. https://www.google.com/search?q=android+group+conversation&oq=android+group+con&aqs=chrome.0.0j69i57j0l4.10109j0j1&sourceid=chrome&ie=UTF-8</p> <ol style="list-style-type: none"> 1. Wake your Android device and unlock the screen. 2. Tap the "Application Menu" button at the bottom of the screen. Then tap the "Camera" icon. 3. Move the camera function slider at the bottom of the window to the "Video Camera" option. 4. Tap the "Resolution" icon at the lower left corner of the viewing window. Then tap the option that corresponds with how you wish to share the video. For example, if you wanted to send the video as an MMS message, tap the "MMS" option. http://smallbusiness.chron.com/shrink-videos-big-android-28400.html |
| <p>11[A]. The method of claim 6, further comprising, by the first device: receiving user selection of the sub-net;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving user selection of the sub-net. See claims 6 and 7, which are incorporated herein by reference in their entirety.</p> |
| <p>[11B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech,</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 3, 6, 7[A], 9, 11[A], which are incorporated herein by reference in their entirety.</p> |

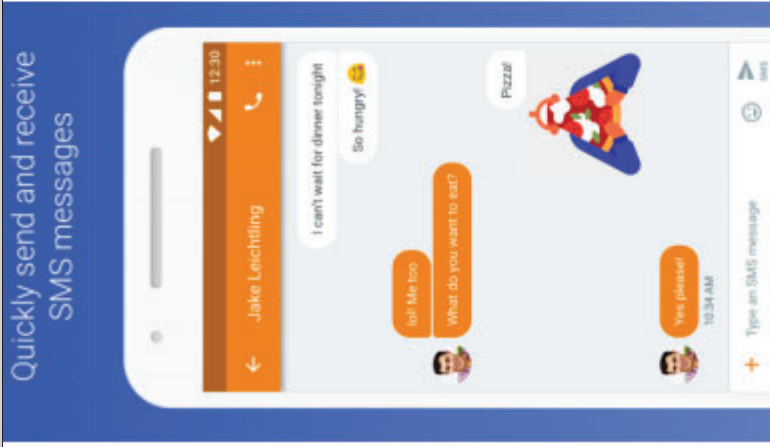
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2 vibrate, or increase sound levels.</p> | <p>ZTE</p> |
| <p>12. The method of claim 1, wherein first device sends the SMS messages and receives the IP-based responses via a cellular communications network.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein first device sends the SMS messages and receives the IP-based responses via a cellular communications network. See claims 1 and 8, which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products are known to use IP-based communication over wireless or data connections. Both Android Messages and Google Hangouts utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact. U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes ZTE Android phones as smartphones for which U.S. Cellular provides cellular communications services to enable the IP-based communications over its cellular network.</p> <p>https://www.uscellular.com/devices/phones/zte-imperial.html?productId=prod2140117</p> <p>Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <p>Enhanced features: On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.</p> <p>https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|---|
| <p>13. The method of claim 1, further comprising performing, by a first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by a first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating a data call with the particular second device. See claims 1[F], 9, and 12, which are incorporated herein by reference in their entirety.</p> |
| <p>14. The method of claim 1, wherein the user input further specifies information associated with the entity, and wherein the method further comprises performing, by the first device: based on the user input, transmitting the user-specified information associated with the entity to the second devices.</p> <p>A user shares a location corresponding to an entity (e.g. restaurant) that is not the device location. A user can search for or navigate to a location and then specify the location with a symbol corresponding to the entity. The user can also send information associated with the entity via the Android app as a comment in an MMS message.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user input further specifies information associated with the entity, and wherein the method further comprises performing, by the first device: based on the user input, transmitting the user-specified information associated with the entity to the second devices.</p> <p>A user shares a location corresponding to an entity (e.g. restaurant) that is not the device location. A user can search for or navigate to a location and then specify the location with a symbol corresponding to the entity. The user can also send information associated with the entity via the Android app as a comment in an MMS message.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




| | |
|---|---|
| <p>US9408055B2</p> <p>input, transmitting the user-specified information associated with the entity to the second devices.</p> | <p>ZTE</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share . If you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>15. The method of claim 14, further comprising, by the performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the entity and, based, thereon, displaying the information associated with the entity.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the entity and, based, thereon, displaying the information associated with the entity. See claims 1[G] and 14, which are incorporated herein by reference in their entirety.</p> <p>Using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share. If you don't see this, tap More > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E. T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More > Stop sharing. <p>https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>16. The method of claim 15, wherein the</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the information comprises a category of the entity. See claims 13-15, which are</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| ZTE | ZTE |
|--|---|
| <p>US9408055B2 information comprises a category of the entity.</p> | <p>incorporated herein by reference in their entirety. For example, the Google Maps app identifies restaurants that are categorized as providing breakfast, lunch, coffee, dinner, drinks, etc.</p> |
| <p>17. The method of claim 15, wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and information associated with the entity.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device uses an Internet Protocol to transmit the user-specified symbol, location, and information associated with the entity. See claims 1, 4, 12, and 15, which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share. If you don't see this, tap More. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E. T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> To stop sharing before you arrive, tap More. <p>https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>18. The method of claim 1, wherein the entity comprises a</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the entity comprises a vehicle, a person, an event, a site, a building, or a facility.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>vehicle, a person, an event, a site, a building, or a facility.</p> | <p>For example, Google Maps allows a user to mark a location on its map and associate with a vehicle, person, event, site, building, or facility. For example, restaurants and train stations can be considered a site, a building, and a facility.</p> <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share . If you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E.T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More  > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More  > Stop sharing. <p>https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|---|
| <p>19[A]. The method of claim 1, wherein the user-specified symbol is a first user-specified symbol, wherein the user-specified symbol corresponds to a second entity other than the first device and the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user-specified symbol is a first user-specified symbol, wherein the user-specified location is a first-user specified location, wherein the entity is a first entity, and wherein the method further comprises performing, by the first device: receiving user-specified information transmitted by a particular second device, the user-specified information including a second user-specified location and a second user-specified symbol corresponding to a second entity other than the first device and the second devices. See claims 1 and 13-15, which are incorporated herein by reference in their entirety.</p> <p>Using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share  if you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E. T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More  > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More  > Stop sharing. <p>https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>[19B] and adding the second user-specified</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] adding the second user-specified symbol to the interactive display at a position on the</p> |

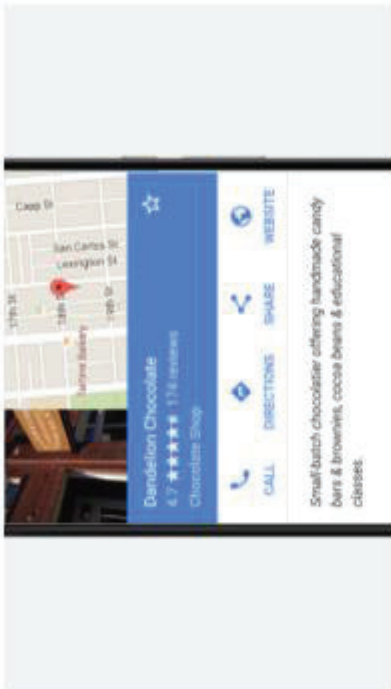
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2 symbol to the interactive display at a position on the interactive map corresponding to the second user-specified location.</p> | <p>ZTE interactive map corresponding to the second user-specified location. See claim 19[A], which is incorporated herein by reference in its entirety. Upon information and belief, when a user received shared map or location, the user's device adds the received shared map or location on its interactive map.</p> |
| <p>20. The method of claim 19, wherein the user-specified information further includes information associated with the second entity, and wherein the method further comprises performing, by the first device: identifying user interaction with the interactive display selecting the second user-specified symbol corresponding to the second entity and, based, thereon, displaying the information associated with the second entity. See claims 2, and 14-15, which are incorporated herein by reference in their entirety. The Accused Devices allow users to receive information associated with fixed locations such as restaurants, stadiums, transportation stations (e.g., bus and train stations), etc.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the user-specified information further includes information associated with the second entity, and wherein the method further comprises performing, by the first device: identifying user interaction with the interactive display selecting the second user-specified symbol corresponding to the second entity and, based, thereon, displaying the information associated with the second entity. See claims 2, and 14-15, which are incorporated herein by reference in their entirety. The Accused Devices allow users to receive information associated with fixed locations such as restaurants, stadiums, transportation stations (e.g., bus and train stations), etc.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Place details

Retrieve rich details about a place, including name, address, phone number, website link and more.

<https://developers.google.com/places/android-api/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Business phone numbers

Google is fantastic for tracking down business phone numbers. You can accomplish this in a number of ways, including:

- **type of business plus zip code:** Perhaps you don't know the name of the business you're looking for, but you have something in mind. Type in the business genre, for example, "pizza restaurant", then the zip code. Google will return local listings that include maps, reviews, and contact information (phone numbers, addresses, [website URLs](#), even email addresses if available).
- **type of business plus city:** Just like in the previous example, except you can substitute the name of a city for a zip code, i.e., "Seattle doctors".




<https://www.ifewire.com/google-phone-number-search-3481892>

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app .
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share . If you don't see this, tap More  > Share.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| <p>21. The method of claim 1, further comprising, by the first device: presenting a symbol corresponding to a facility selected from the group consisting of a hospital, a police station, and a fire station, wherein the symbol corresponding to the facility is positioned on the interactive map at a position corresponding to a location of the facility.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: presenting a symbol corresponding to a facility selected from the group consisting of a hospital, a police station, and a fire station, wherein the symbol corresponding to the facility is positioned on the interactive map at a position corresponding to a location of the facility. See claim 1, which is incorporated herein by reference in its entirety.</p> <p>Upon information and belief, the Accused Products presents a symbol on the map corresponding to a location of a hospital, a police station, and a fire station in response to users to search for any one of those facilities. For example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>22. The method of claim 21, further comprising performing, by the</p> | <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share. If you don't see this, tap More. Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E. T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app. 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More. Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> To stop sharing before you arrive, tap More. Stop sharing. <p>https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en</p> |
| <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility and, based, thereon, displaying information associated with the facility. See claims 15-16, which are incorporated herein by reference in their entirety.</p> | <p>performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility and, based, thereon, displaying information associated with the facility. See claims 15-16, which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|---|
| <p>first device: identifying user interaction with the interactive display selecting the symbol corresponding to the facility and, based, thereon, displaying information associated with the facility.</p> | |
| <p>23[A]. The method of claim 1, further comprising performing, by the first device: receiving a message sent by a particular second device, wherein the message indicates an action to be performed by the first device; wherein the message indicates an action to be performed by the first device;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: receiving a message sent by a particular second device, wherein the message indicates an action to be performed by the first device. See claims 1, 6-7, 9, 12-13, which are incorporated herein by reference in their entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Google Maps, a user can select a device's symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action. For example, a user can send data causing the second device to: ring an audio alert, display a message and lock the screen, erase its contents</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

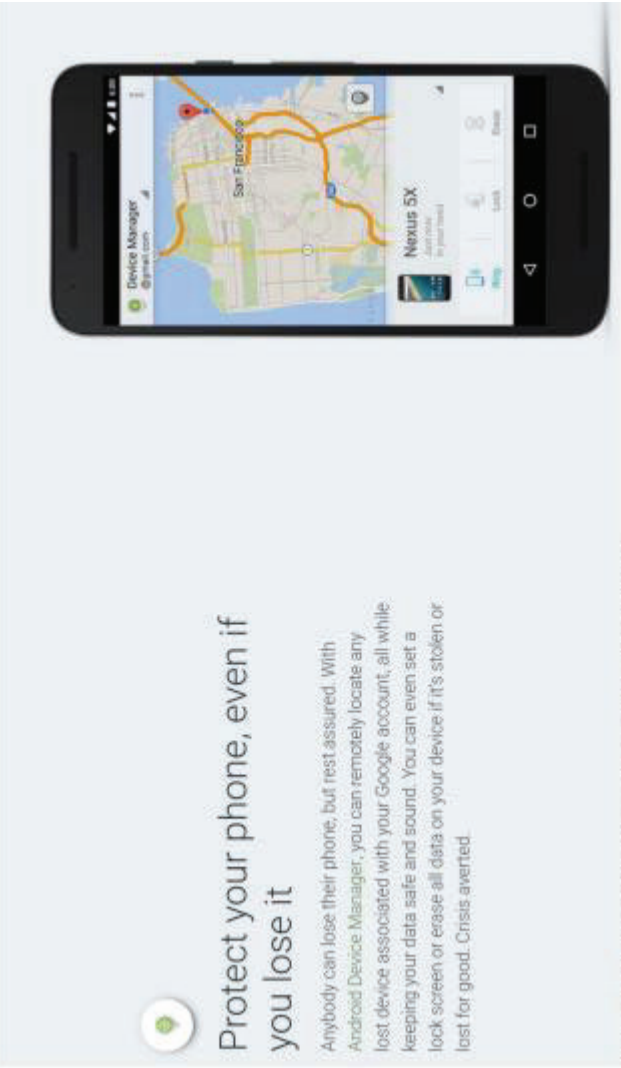
| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p>  <p>Protect your phone, even if you lose it</p> <p>Anybody can lose their phone, but rest assured. With Android Device Manager, you can remotely locate any lost device associated with your Google account, all while keeping your data safe and sound. You can even set a lock screen or erase all data on your device if it's stolen or lost for good. Crisis averted.</p> <p>https://www.android.com/security/overview/</p> |
| <p>[23B] and performing the indicated action.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] performing the indicated action. See claim [23A], which is incorporated herein by reference in its entirety.</p> <p>For example, a user may choose a symbol representing a device and then choose a corresponding action. Using Android Device Manager, a user can select a device's symbol and then effect an action to that device by specifying some action using the interface. Data is sent to the device based on that action. For example, a user can send data causing the second device to: ring an audio alert, display a message and lock the screen, erase its contents.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

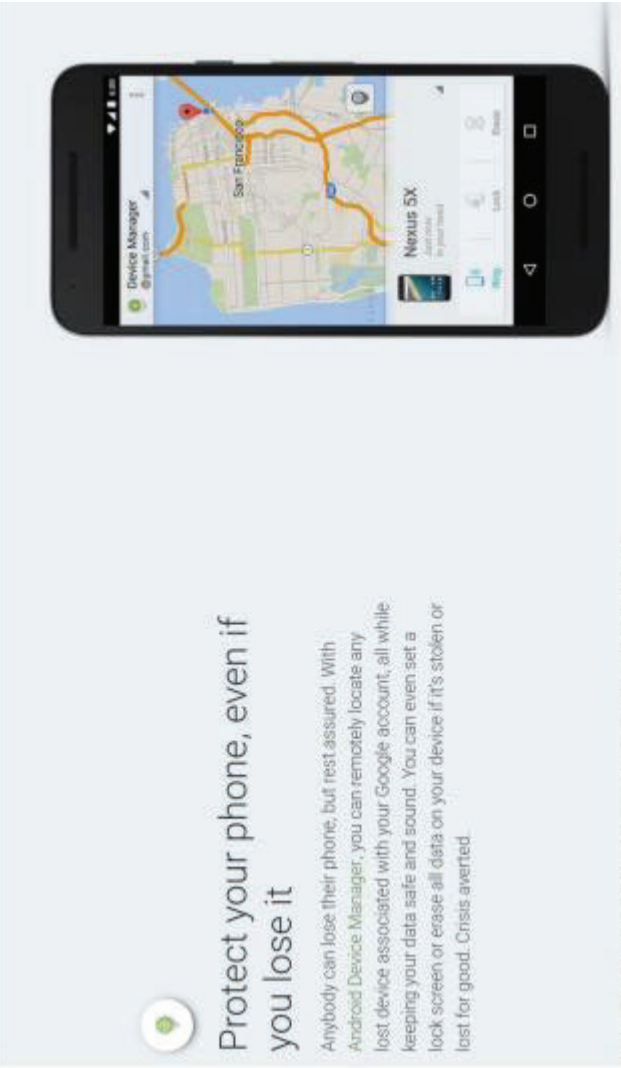
| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>24. The method of claim 23, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech,</p> | <p>Anybody can lose their phone, but rest assured. With Android Device Manager, you can remotely locate any lost device associated with your Google account, all while keeping your data safe and sound. You can even set a lock screen or erase all data on your device if it's stolen or lost for good. Crisis averted.</p>  <p>https://www.android.com/security/overview/</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See claims 1 and 23, which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|---|
| <p>changing sound intensity, and displaying information.</p> | |
| <p>25. The method of claim 1, further comprising, by the first device: remotely controlling a particular second device to perform an action by sending a message to the second device, wherein the message indicates the action to be performed.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of performing, by the first device: remotely controlling a particular second device to perform an action by sending a message to the second device, wherein the message indicates the action to be performed.</p> |
| <p>26. The method of claim 25, wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the indicated action is selected from the group consisting of playing audio, initiating a phone call, vibrating, converting text to speech, changing sound intensity, and displaying information. See claim 24, which is incorporated herein by reference in its entirety.</p> |
| <p>27. The method of</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2 claim 1, further comprising performing, by the first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device.</p> | <p>ZTE performance of performing, by the first device: identifying second user interaction with the interactive display selecting a particular user-selectable symbol corresponding to a particular second device and user interaction with the display specifying an action and, based thereon, initiating voice-over-IP (VOIP) communication with the particular second device. See claim 4, which is incorporated herein by reference in its entirety. Using, e.g., Android's Google Voice app, users can initiate VOIP communications. For example, Google Voice is a voice over IP technology over internet protocol (IP) networks.</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Send & get text messages

You can send text messages for free to U.S. and Canadian numbers using your Google Voice number. Texts sent using Google Voice will use Wi-Fi, or mobile data from your cell phone service plan if you're not connected to Wi-Fi. If you're outside the U.S. and are not using Wi-Fi, your cell phone company might charge you extra roaming fees to send a text.

Google Voice & Hangouts: You can choose to link Google Voice and Hangouts. If you do, you'll have to use Hangouts to send texts, not the Google Voice website or apps. To stop using Hangouts for texts, turn off Google Voice in Hangouts.

ANDROID COMPUTER IPHONE & IPAD





If you haven't yet, download the Google Voice app on your Android device. 

Send a text message

With the Google Voice website and apps, you can text people messages and photos and send texts to groups of people.

If you send a text longer than 160 characters to a non-Google Voice number, it will be sent as multiple messages.

Note: You can't send texts to five- or six-digit "short code" numbers.

1. On your Android device, open the Google Voice app .
2. Open the tab for Messages .
3. At the bottom, tap Add .
4. Enter a contact's name or phone number.
 - To create a group text message, add up to 30 names or phone numbers.
5. Enter your message, and tap Send .

To include an image with your message, tap Select image . If your image is bigger than 2MB, it'll be sent as a smaller file. But GIFs over 2MB won't send <https://support.google.com/voice/answer/115116?co=GENIE.Platform%3DAndroid&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2 28[P]. A system comprising: a first device programmed to perform operations comprising:</p> | <p>ZTE ZTE infringes either directly or indirectly, induces others to infringe, and/or contributes to the infringement of this system as set forth below. See claim 1[P], which is incorporated herein by reference in its entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><u>Google Maps Share Location</u></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps’ Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p>See, e.g., https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/; https://googleblog.blogspot.com/2009/02/see-where-your-friends-</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

[are-with-google.html](https://google.com/pixelphone/answer/6160491); <https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html>; <http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html>; <https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html>

Control within reach, even when your device isn't

One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching Find My Device as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit android.com/find or check out the app.

See, e.g., <https://www.blog.google/products/android/google-play-protect/>

Find your device using Android Device Manager

If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.

Before you can use Android Device Manager to locate your device: Your device's location access need to be turned on and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.

Tip: if you've linked your phone to Google, you can locate or ring it by searching for **find my phone** on google.com.

<https://support.google.com/pixelphone/answer/6160491>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Link your phone to Google

You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.

Link your Android phone

Step 1: Update the Google app

1. On your phone, go to the Google app page on the Play Store.
2. Tap **Update**.

Step 2: Turn on Google Now

1. On your phone, open the Google app .
2. At the top left, tap Menu  > **Settings** > **Now cards**.
3. Turn on **Show cards**.
4. Turn on **Show notifications**.

Step 3: Turn on Web & App Activity

1. Visit the **Account History** page.
2. Make sure the switch is on (green).

Step 4: Sign in to your browser

1. On your phone, open the Google app .
2. At the top left, tap the Menu .
3. At the top left, you'll see the email address you use for the Google app.
4. Visit www.google.com  on your computer.
5. If you aren't signed in already, click **Sign in** in the top right corner of the page.
6. Sign in using the Google Account you use for the Google app.

Step 5: Send information to your phone

1. Do one of the searches below, like **note** to **self**, or **send directions** to **my phone**.
2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

What you can do once your phone is linked

Find my phone

You can get the current location of your phone if you cant find it.

1. On your computer's browser, search on www.google.com for **find my phone**.
2. If your phone is turned on and connected to the Internet, you'll see your phone's location.
3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking **Ring**. You can stop the ringing from your phone when you find it.

Tip: You can also find your missing phone using the **Android Device manager** which lets you find your device or remotely ring, lock, or erase it.

Send directions to my phone

Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.

1. On your computer's browser, search on www.google.com for **send directions to my phone**.
2. Enter in your destination.
3. Click **Send directions to your phone**.
4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.

Send a note to my phone

1. On your computer's browser, search on www.google.com for **send a note to my phone**.
2. Type your note in the box.
3. Click **Send note to your phone**.
4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.

Set an alarm

1. On your computer's browser, search on www.google.com for **set an alarm**.
2. Choose the time you want the alarm to go off.
3. Click **Set an alarm on your phone**.
4. An alarm will now be set on your phone's Clock app.

Set a reminder

1. On your computer's browser, search on www.google.com for **set an reminder**.
2. Type what you want to be reminded about, and either when or where you want the reminder to go off.
3. Click **Remind me on my devices**.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your location using Google Maps

You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.

<https://support.google.com/plus/answer/3302509?hl=en&co=GENIE.Platform%3DAndroid&oco=1>

Location

Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.

- **Mode** – Sets the how your current location information is determined.
- **Recent Location Request** – Displays applications and services that have recently requested your location information.
- **Camera** – Checkmark to tag photos or videos with their locations.
- **Google Location History** – Allows you to view and manage your Google location history.

Accounts & sync

Use the Accounts & sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.

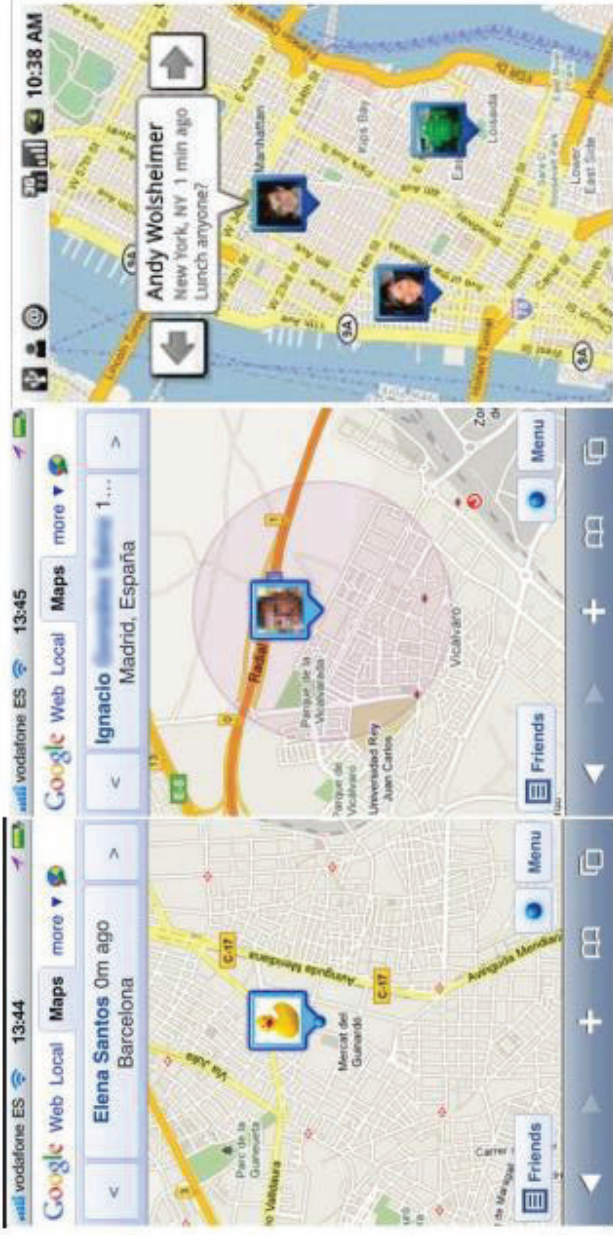
Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch **Add account** to add new account.

.. ..

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html>

Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.

ZTE makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as ZTE phones and ZTE tablets, as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example ZTE Android devices that perform each step of this method as set forth below.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE phones
 mob.org * Mobile phones and smartphones catalogue

Sort by: Popularity Date Price

HTC
 LG
 Samsung
 Motorola
 Fly
 Sony-Ericsson
 Apple
 Nokia
 Wobisado
 Vertu
 BenQ-Siemens
 Sagem
 Alcatel
 Philips

All brands

ZTE Rapido
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 4.5 inch.
 Android 4.1.2

ZTE Grand S3
 Mobile phone
 2015 year
 Touchscreen: 1080 x 1920
 5.5 inch.
 Android 4.4

ZTE Geek 2
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE V5s
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE Grand X Quad
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.1

ZTE ZMAX
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.4.2

ZTE Grand Memo Lite
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.3

https://mob.org/phone/zte/page_3/sort_date_down/

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p>  <p>https://www.zteusa.com/products/tablets</p> |
| <p>[28A] obtaining contact information of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p> | <p>ZTE makes, uses, sells, and otherwise provides this second device by making, using, selling, and importing Android phones and tablets as well as by providing its servers for use with Android devices to enable features such as Find My Device.</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of a first device programmed to perform operations comprising: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claim 1[A] and 28[P], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Phone calls

How to make calls



There are many ways to make a call with your phone, and they're all easy to do.

Calling from the dialer



1. From the home screen, tap .
2. Enter the phone number with the on-screen keypad. Tap  to delete wrong digits.
3. Tap  to place the call.

Tip: To make international calls, press and hold **0+** to enter the "+".

Calling from your contacts

1. From the home screen, tap .
2. Swipe your finger up or down to scroll through the contacts list and tap  next to the contact you want to call.

Tips:

- You can search for a contact by tapping  and entering the contact name.
- You can also access your contacts by tapping  > **Favorites**.



People

You can add contacts on your phone and synchronize them with the contacts in your Google account or other accounts that support contact syncing.

To see your contacts, tap  on the home screen. From there, you can tap the tabs on the top to quickly switch to **Groups**, or **Favorites**.

Importing and exporting contacts

You can import/export contacts from/to your SIM card, phone storage, or microSDHC card. This is especially useful when you need to transfer contacts between different devices. You can also quickly share your contacts using Bluetooth, Email, Messaging, etc.



Importing contacts from the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Manage SIM card contacts**.
2. If you have added contact accounts other than the phone, select an account in which to save the contacts.
3. Tap the contacts you want to import one by one, or tap  > **Import all**.

Importing contacts from a microSDHC card or phone storage



1. From the Contacts screen, tap  > **Import/export** > **Import from phone storage**.
2. If you have added contact accounts other than the phone, select an

Calling from your call history

1. From the home screen, tap  > **Call log**.
2. Tap  next to the number you want to call.

Calling from a text message

If a text message contains a phone number that you want to call, you can make the call while viewing the text message.

1. From the home screen, tap .
2. Tap the conversation and then find the message that contains the phone number you need.
3. Tap the number and then tap .

Using speed dial

Press and hold the **1-9** key from the dialer to call the corresponding speed dial number.

The **1** key is reserved for your voicemail.

Assigning a speed dial key

1. From the home screen, tap .
2. In the Phone tab, tap  > **Speed dial setting**.
3. Tap a speed dial key and tap **Set speed dial contact**.
4. Select a contact from the contact list.

Note: You can also set your speed dial keys from the dialer. Press and hold the **2-9** key and then tap **OK**.


3. Select the vCard file(s) in the microSDHC card or the phone storage and tap **OK**.

Note: If the microSDHC card is not installed in the phone, you can import vCard file(s) in the phone storage.

Exporting contacts to the SIM card


1. From the Contacts screen, tap  > **Import/export** > **Export to SIM card**.
2. Select the contacts you want to export and then tap .

Exporting contacts to the microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Export to phone storage**.
2. The phone will prompt you with the name of the vCard file and the directory in which the file will be saved. Tap **OK** to create the file.

Note: If the microSDHC card is not installed in the phone, you can export vCard file(s) into the phone storage.

Sharing contact information

1. From the Contacts screen, tap  > **Import/export** > **Share visible contacts**.
2. Choose how to share the contacts. Options depend on the applications and services installed.

Creating a contact

1. From the Contacts screen, tap  to add a new contact.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

2. Tap the account field near the top of the screen to choose where to save the contact. If a sync account is selected, the contacts will be synced automatically with your account online.
3. Enter the contact name, phone numbers, email addresses, and other information.
4. Tap **DONE** to save the contact.

Adding a contact to Favorites

You can add the contacts you use frequently to Favorites so that you can find them quickly.

1. From the Contacts screen, tap the contact you want to add to **Favorites**.
2. Tap  next to the contact's name.

Searching for a contact

1. Tap the search field above the contacts list.
2. Enter the contact name you want to search for. Matching contacts will be listed.

Joining contacts


As your phone synchronizes with multiple online accounts, you may see duplicate entries for the same contact. You can merge all the separate information of a contact into one entry in the Contacts list.

1. From the home screen, tap .
2. Tap a contact to display the contact's details.
3. Tap  > **Edit** >  > **Join**.

Entering text

You can enter text using the onscreen keyboard. Some apps open it automatically. In others, you open it by tapping where you want to type. You can also enter text by speaking with the Google voice typing feature. Tap  to hide the onscreen keyboard.

Changing input methods

1. When you use the onscreen keyboard to enter text, the icon  appears on the notification bar.
2. Open the notification panel and tap **Choose input method**.
3. Select an input method you need.

Google keyboard

The Google keyboard provides a layout similar to a desktop computer keyboard. Turn the phone sideways and the keyboard will change from portrait to landscape.

To use the landscape keyboard, tap the **Auto-rotate screen** check box in  > **System settings** > **Accessibility**.

Note: The landscape keyboard is not supported in all applications.

4. Tap the contact whose information you want to join with the first entry.
5. Tap **DONE**.

The information from the second contact is added to the first, and the second contact is no longer displayed in the contacts list.




You can repeat these steps to join another contact to the main contact.

Separating contact information

If contact information from different sources was joined in error, you can separate the information back into individual contacts on your phone.

1. From the home screen, tap .
2. Tap a contact you have merged and want to separate.
3. Tap  > **Edit** >  > **Separate**.
4. Tap **OK** to confirm.

Creating a new group

1. From the Contacts screen, tap .
2. Tap .
3. If you have added contact accounts other than the phone, choose an account for the new group.
4. Enter the group name and tap **DONE**.
5. Tap  and select the contacts you wish to be the group members.
6. Tap .

To send messages to the group members, you can tap a group and then tap  > **Send group message**.















- Tap the alphabetic keys to enter letters. Press and hold the keys to enter associated accented letters or numbers. For example, to enter É, press and hold  and the available accented letters and number 3 appear. Then slide your finger to choose **E**.
- Tap  to use uppercase or lowercase letters. This key also changes to indicate the current case you are using:  for lowercase,  for uppercase, and  when locked in uppercase. Press and hold or double-tap  to lock the keyboard in uppercase.
- Tap  to delete any text you have entered.
- Tap  to select numbers and symbols. You can then tap  to access more.
- Tap  to enter miniature icons.
- Tap  to use Google's networked voice input.
- Press and hold  to change the input language or the Google keyboard settings.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p>ZTE</p> <p>In another example, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.</p> <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider-provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contacts_id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/provider-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|--|-------------|---|-----------|---|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID       = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI    = 3;
50         public static final int CONTACT_LOOKUP_KEY   = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI    = 1;
65         public static final int CONTACT_LOOKUP_KEY    = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS        = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57     * Ensures that this is a valid group URI. If invalid, then an exception is
58     * thrown. Otherwise, the original URI is returned.
59     */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




US9408055B2

ZTE

Send & receive text messages in Android Messages





You can send and receive text messages with friends and contacts on Android Messages.

Start a conversation

1. Open the Android Messages app .
2. Tap Compose .
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next .

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329

See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](#) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
2. Tap Menu  > **Create label**.
3. Enter a label name and tap **OK**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
- **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER ANDROID IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

Contact someone

You can call, email, or send text messages to your contacts.

1. Open your device's Contacts app.
2. Tap a contact in the list.
3. Choose an option:
 - Call
 - Email
 - New message

https://support.google.com/nexus/answer/6118731?hl=en&ref_topic=6118711

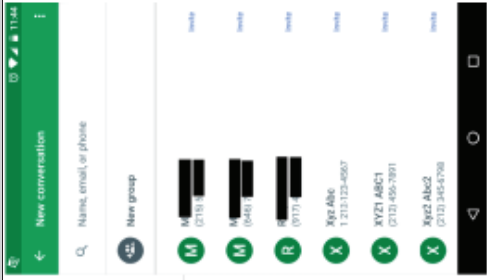


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Start a conversation

1. Open the Android Messages app
2. Tap Compose
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329
<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add + > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

1. Open the Hangouts app.
2. At the bottom, tap Add + > New conversation > New group.
3. Enter and select the names, phone numbers, or email addresses of people in your group.
4. Tap Done.

https://support.google.com/hangouts/answer/3111943?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products





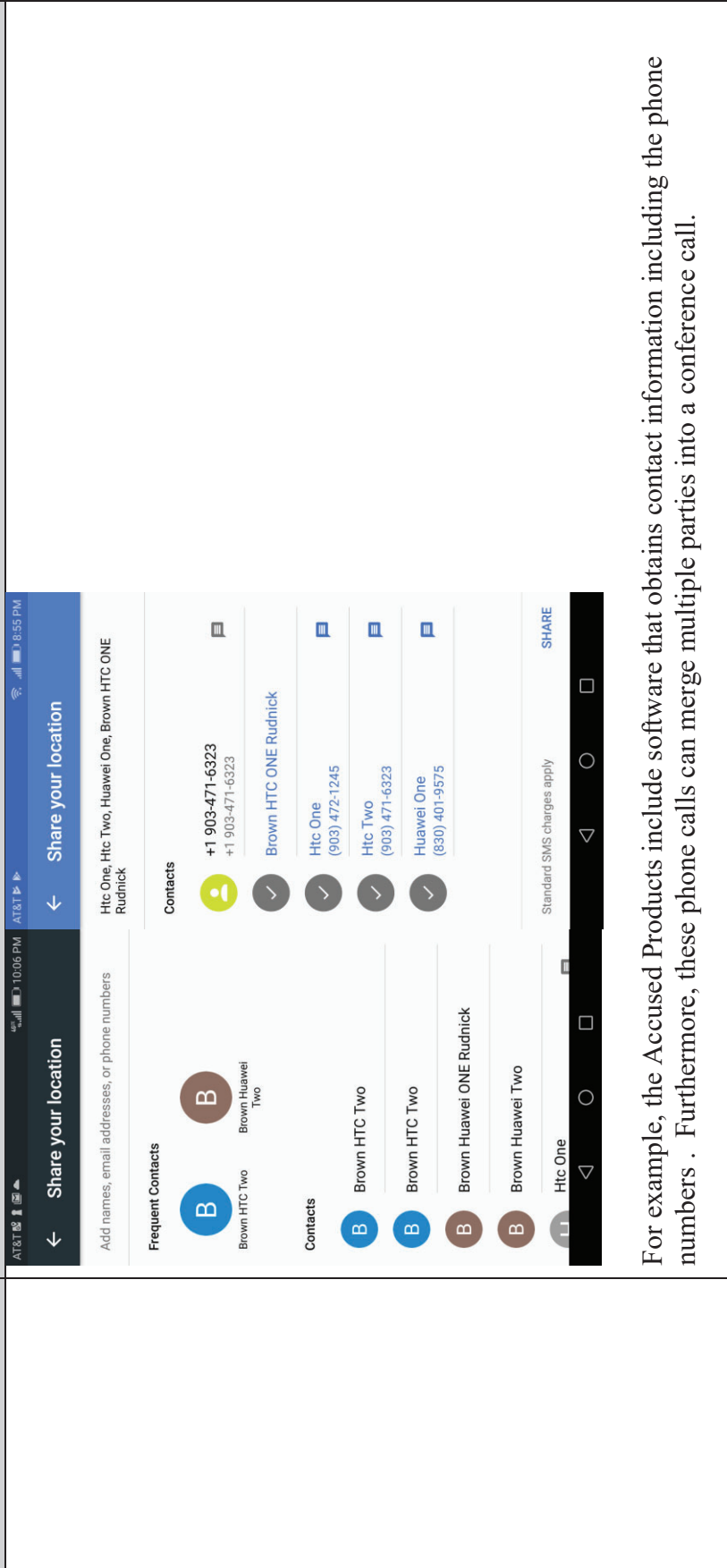
| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <h3 data-bbox="235 1186 276 1554">Contact someone</h3> <p data-bbox="300 903 332 1554">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="357 1113 625 1554" style="list-style-type: none">1. Open your device's Contacts app .2. Tap a contact in the list.3. Choose an option:<ul data-bbox="495 1291 625 1522" style="list-style-type: none">• Call • Email • New message  <p data-bbox="641 630 673 1585">https://support.google.com/hexus/answer/611873?hl=en&ref_topic=6118711</p> <p data-bbox="706 1071 738 1585"><u>Exemplary Google Maps Screenshots:</u></p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

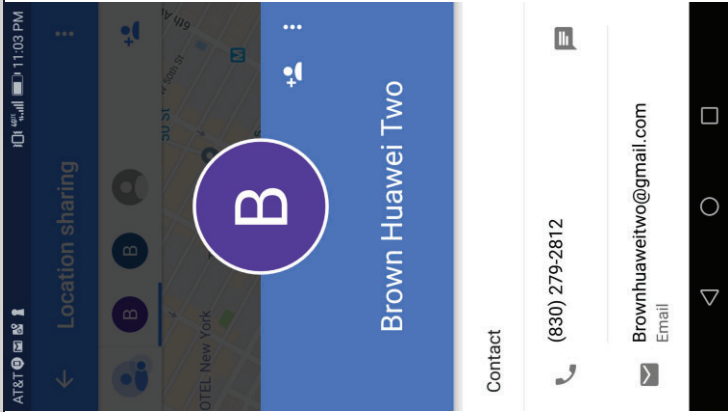
| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p>  |
| <p>[28B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claim 1[B], which is incorporated herein by reference in its entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claim 1[B], which is incorporated herein by reference in its entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products



| | |
|--|--|
| <p>US9408055B2</p> <p>the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p> | <p>ZTE</p> <p>SMS message, with additional information, to a contact.</p> <p>Using Wi-Fi Direct</p> <p>Wi-Fi Direct allows Wi-Fi devices to connect to each other without the need for wireless access points (hotspots).</p> <p>Connecting to another device via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > System settings > Wi-Fi. 2. If Wi-Fi is off, slide the Wi-Fi switch to the On position. 3. Tap  > Wi-Fi Direct. Your phone will search for other devices enabled with Wi-Fi Direct connections. 4. Tap a device name under Peer Devices to connect with it. The other device will receive a Wi-Fi Direct connection prompt and need to accept the request for connection. Both devices may need to enter a common PIN. If prompted, tap Connect. 5. Once connected, the device is displayed as "Connected." <p>Sending data via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. Open the appropriate application and select the file or item you want to share. 2. Select the option for sharing via Wi-Fi Direct. The method may vary by application and data type. 3. Tap a device the phone has connected with or wait for it to search for new devices and tap one of them. <p>Receiving data via Wi-Fi Direct</p> <p>When an attempt to transfer data via Wi-Fi Direct is received, you can see a notification in the status bar. Tap Accept to start receiving the data. Received files are stored automatically in a dedicated folder (<i>WiFiShare</i>, for instance) in the phone storage or microSDHC directory. You can access them with the File Manager app.</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Messaging

You can use Messaging to exchange text messages (SMS) and multimedia messages (MMS).

Message box



Instead of an inbox and outbox, your phone organizes all messages you sent and received into one box, where messages exchanged with the same number are grouped into one message thread on the Messaging screen. You can tap a thread to see the conversation you have had with someone.

Message threads are sorted in chronological order with the latest one on top.

Opening the messaging screen

From the home screen, tap .

The Messaging screen opens, where you can create a new message, search for messages, or open an ongoing message thread.

- Tap  to write a new text or multimedia message.
- Tap  to search for a message with keywords.
- Tap an existing message thread to open the conversation you've had with a certain number.

Sending a message





1. From the messaging screen, tap  at the bottom.
2. Add recipients by one of the following ways:
 - Tap the **To** field and manually enter the recipient's number or the contact name. If the phone presents a few suggestions, tap the one you want to add.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



- Select recipients from your contacts by tapping .
- 3. Tap the **Type message** field and enter the content of your text message.
- 4. If you want to send a multimedia message, tap the paper clip icon  to attach a file or a slideshow to the message.
- 5. Tap  to send your message.

Notes:

- You can also include email addresses as recipients for multimedia messages.
- Do not add any attachment if you want to send a text message. Otherwise you may be charged for a multimedia message.


Replying to a message

Messages you receive are appended to existing threads of the same number. If the new message comes from a new number, a new thread is created.


1. From the Messaging screen, tap the thread that has the message you want to reply to.
2. Type your reply in the text box at the bottom. You can tap the icon  if you want to reply with an MMS.
3. Tap  to send your message.

Forwarding a message

1. From the Messaging screen, tap the thread that has the message you want to forward.
2. Press and hold the message.
3. Tap **Forward** in the menu that opens.

4. Enter a recipient for the message and edit the content if you want.
5. Tap  to send your message.

Changing message settings

The phone's message settings are pre-configured for you to use immediately. To change them, tap  > **Settings** from the Messaging screen.

Storage settings:

- **Delete old messages:** Delete old messages as limits are reached.
- **Text message limit:** Set the maximum number of text messages allowed in a single thread.
- **Multimedia message limit:** Set the maximum number of multimedia messages allowed in a single thread.

Text (SMS) message settings:

- **Manage SIM card messages:** Manage the messages stored on your SIM card.
- **Service Center:** Enables you to view and edit the service center number.
- **Multimedia (MMS) message settings:**
 - **Auto-retrieve:** Automatically download multimedia messages.

Display settings:

- **Bubble and background:** Set the appearance of the messaging bubbles and background.

Notification settings:


- **Notifications:** Show message notifications in the status bar.
- **Choose ringtone:** Choose a ringtone for your incoming messages.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Email

From the home screen, tap . You can receive and send emails from your webmail or other accounts using POP3 or IMAP, or access your Exchange ActiveSync account for your corporate email needs.

Setting up the first email account

1. When you open **Email** for the first time, enter your email address and password.
2. Tap **Next** to let the phone retrieve the network parameters automatically.
Note: You can also enter these details manually by tapping **Manual setup** or when automatic setup fails.
3. Follow the on-screen instructions to finish the setup.

Your phone will show the inbox of the email account and start to download email messages.

Checking your emails

Your phone can automatically check for new emails at the interval you set when setting up the account.

You can also check new emails manually by tapping  in any of the email account's boxes. Tap **Load more messages** at the bottom of the email list to download earlier messages.

Responding to an email

You can reply to or forward a message that you receive. You can also delete messages and manage them in other ways.

Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.

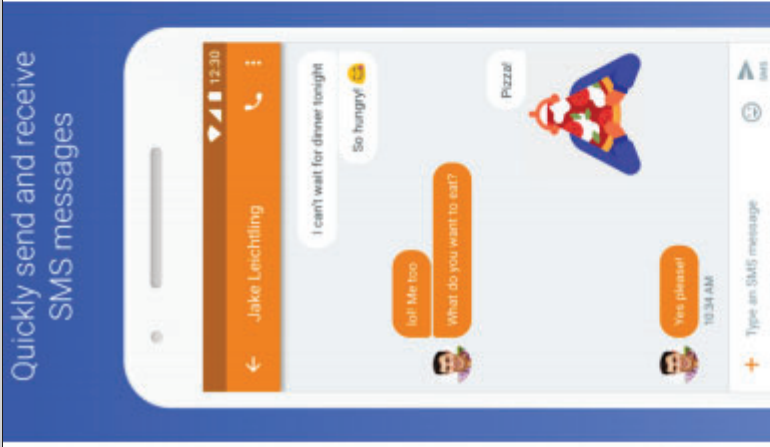
• **Enhanced features:** On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.

<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER **ANDROID** IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap **Add** > **New Conversation**.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap **Send**.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

1. Open the Hangouts app.
2. At the bottom right, tap **Add**.
3. Choose **New SMS**.
4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.
5. Tap the number or contact.
6. Tap **Continue**.
7. Type your message and tap **Send**.

<https://support.google.com/hangouts/answer/3441321?hl=en>

Google Maps Share Location

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p><u>Exemplary Support for Google Maps:</u></p> |
|---------------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. <p>Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap  > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap **More**  **> Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap **More**  **> Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap **Menu**  **> Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon **> More**  **> Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap **More**  **> Hide from map.**
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD

Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%63DAn droid&oco=1

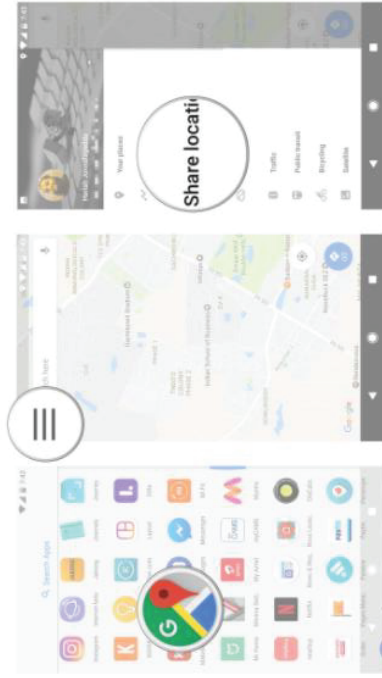
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

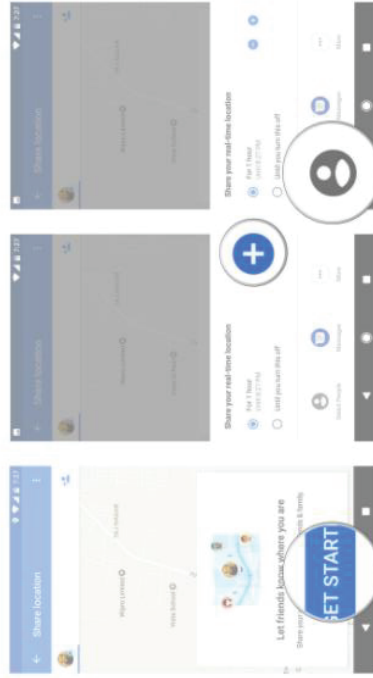
ZTE

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



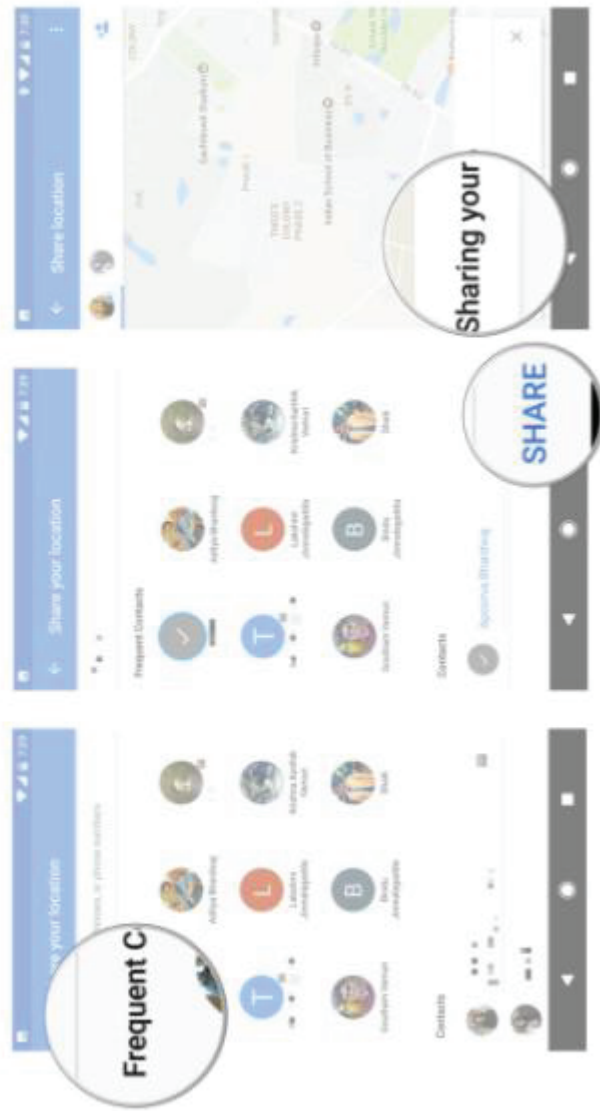
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

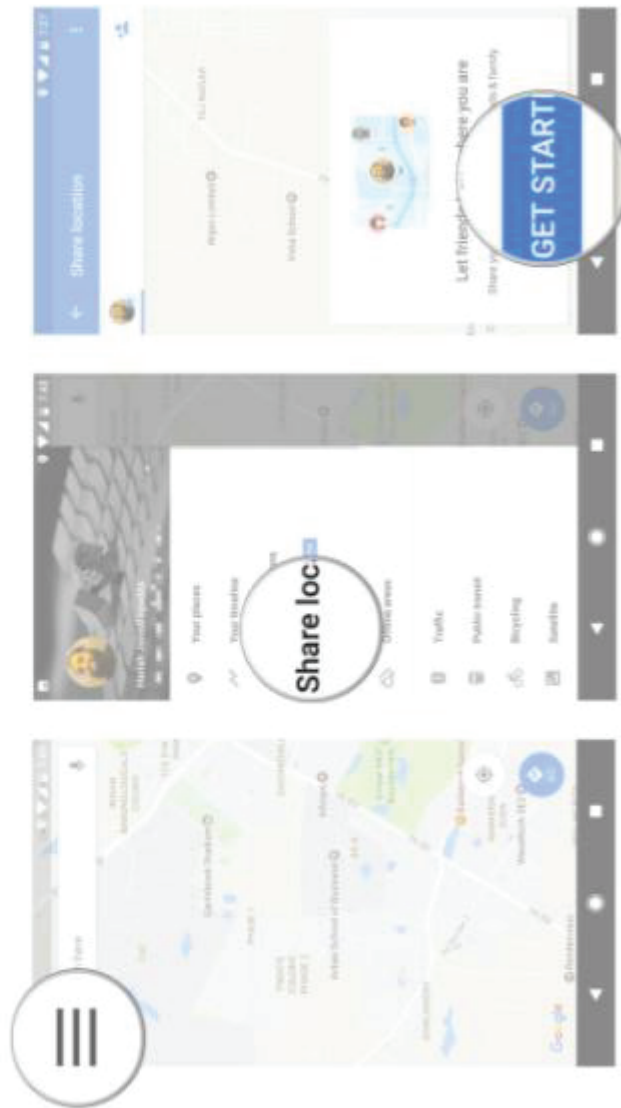
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

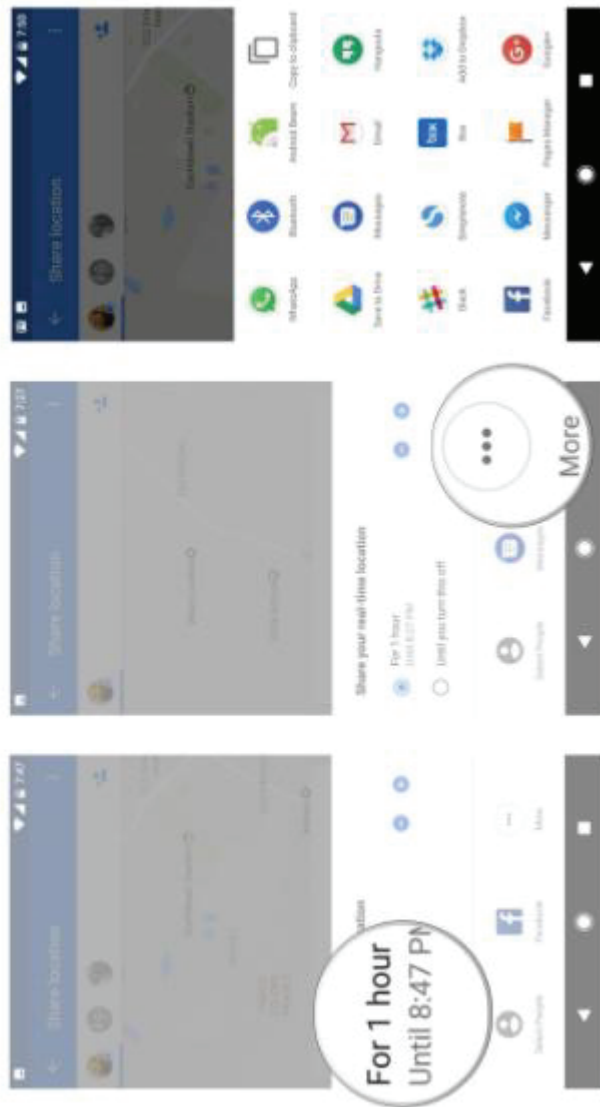
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

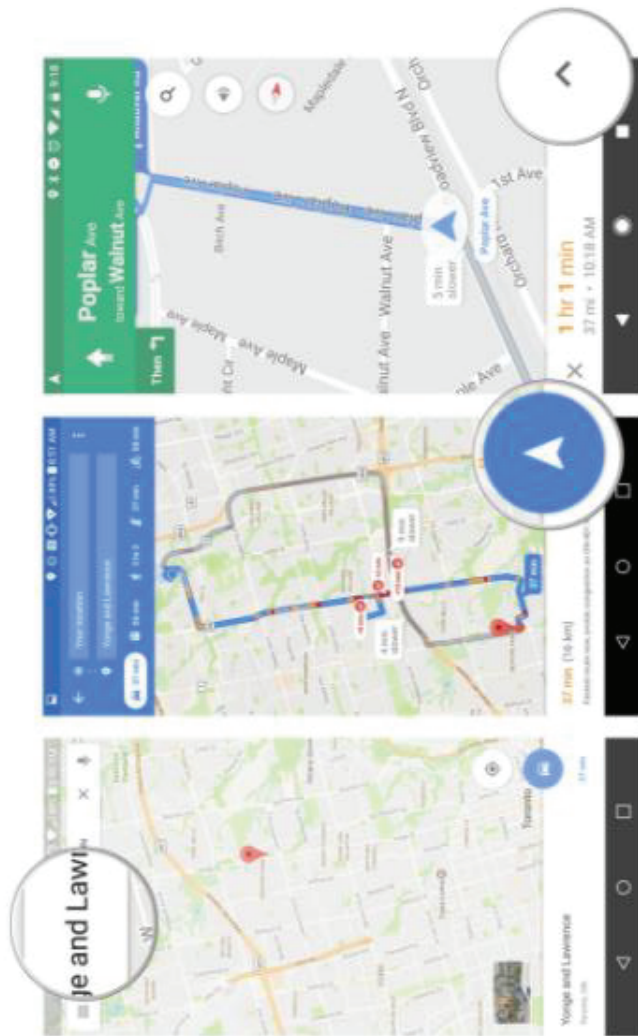
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



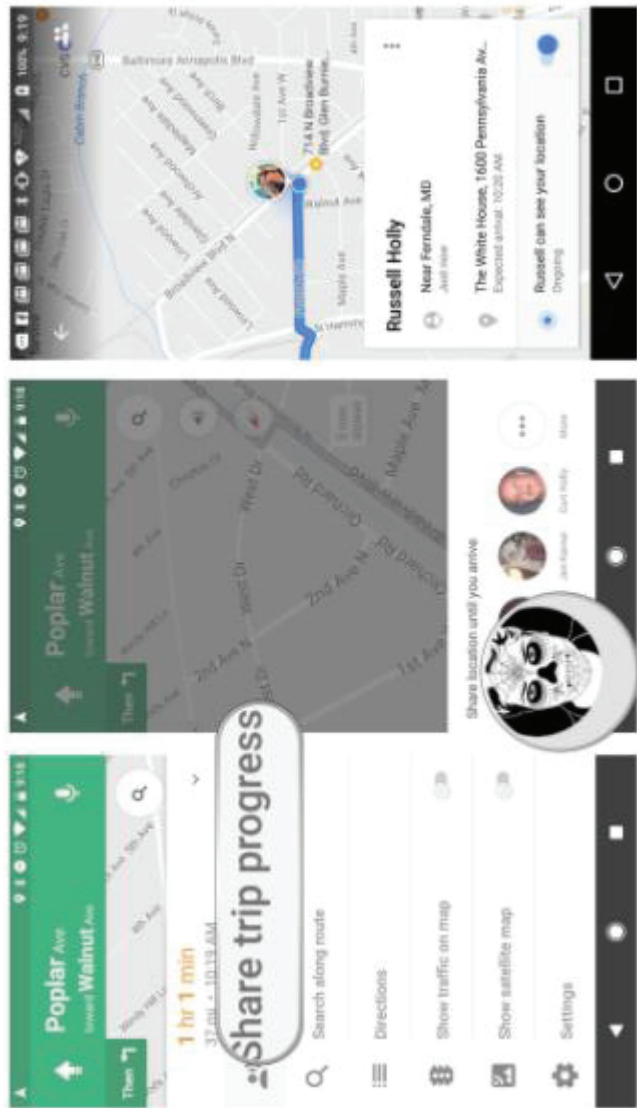
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



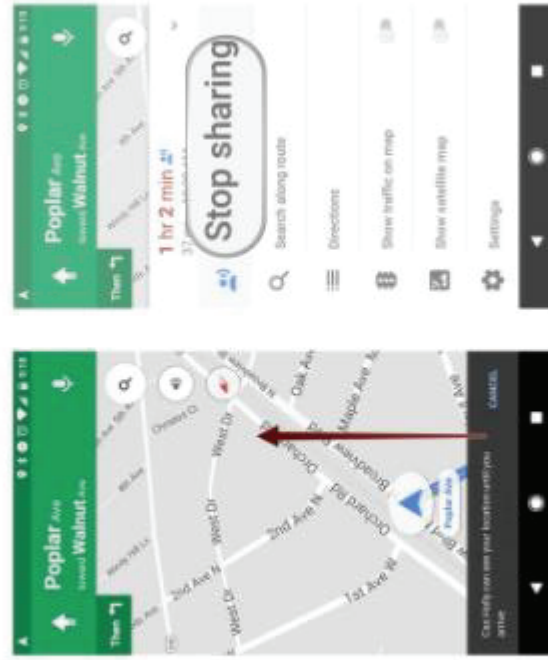
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>





As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

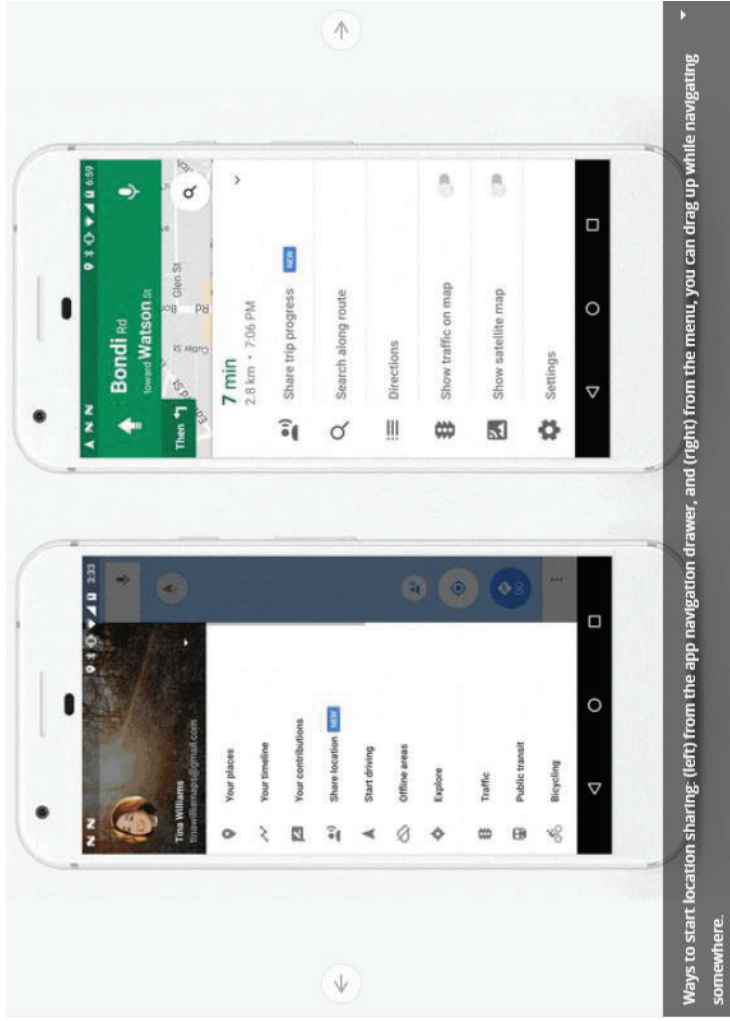
1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

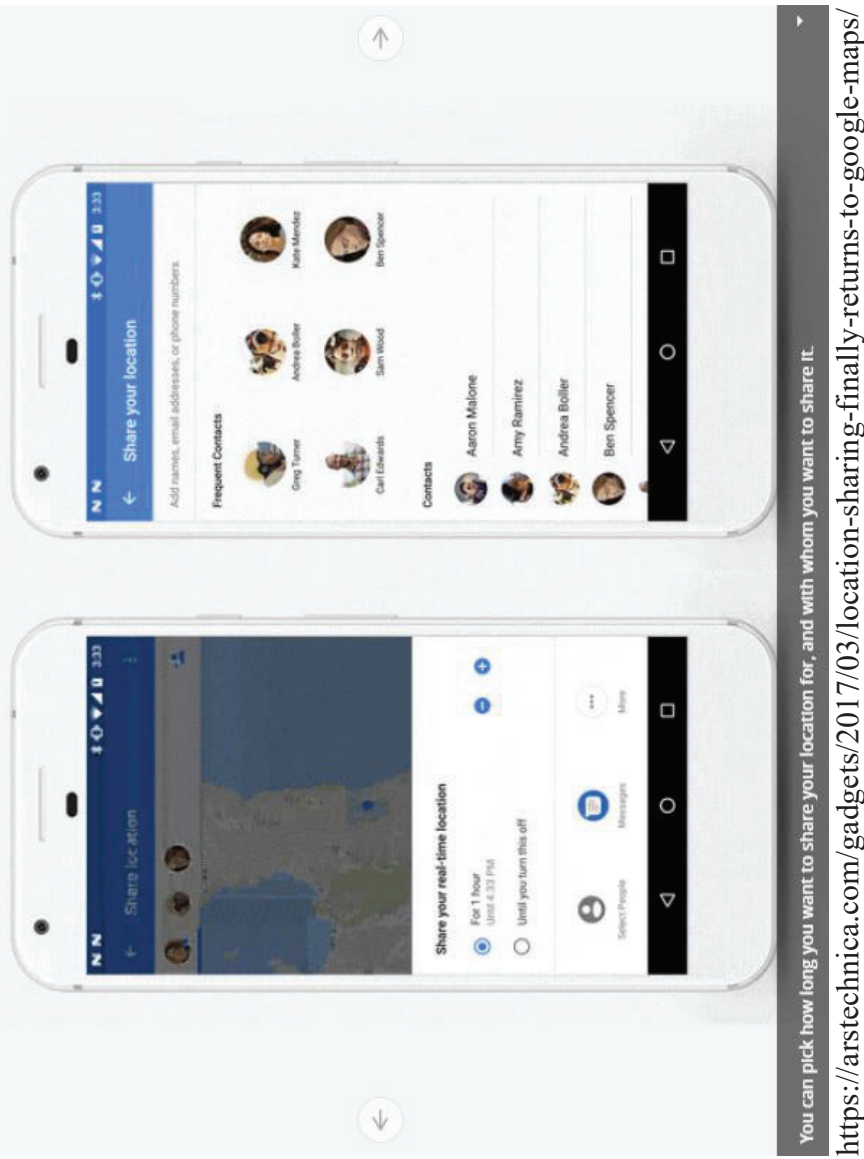
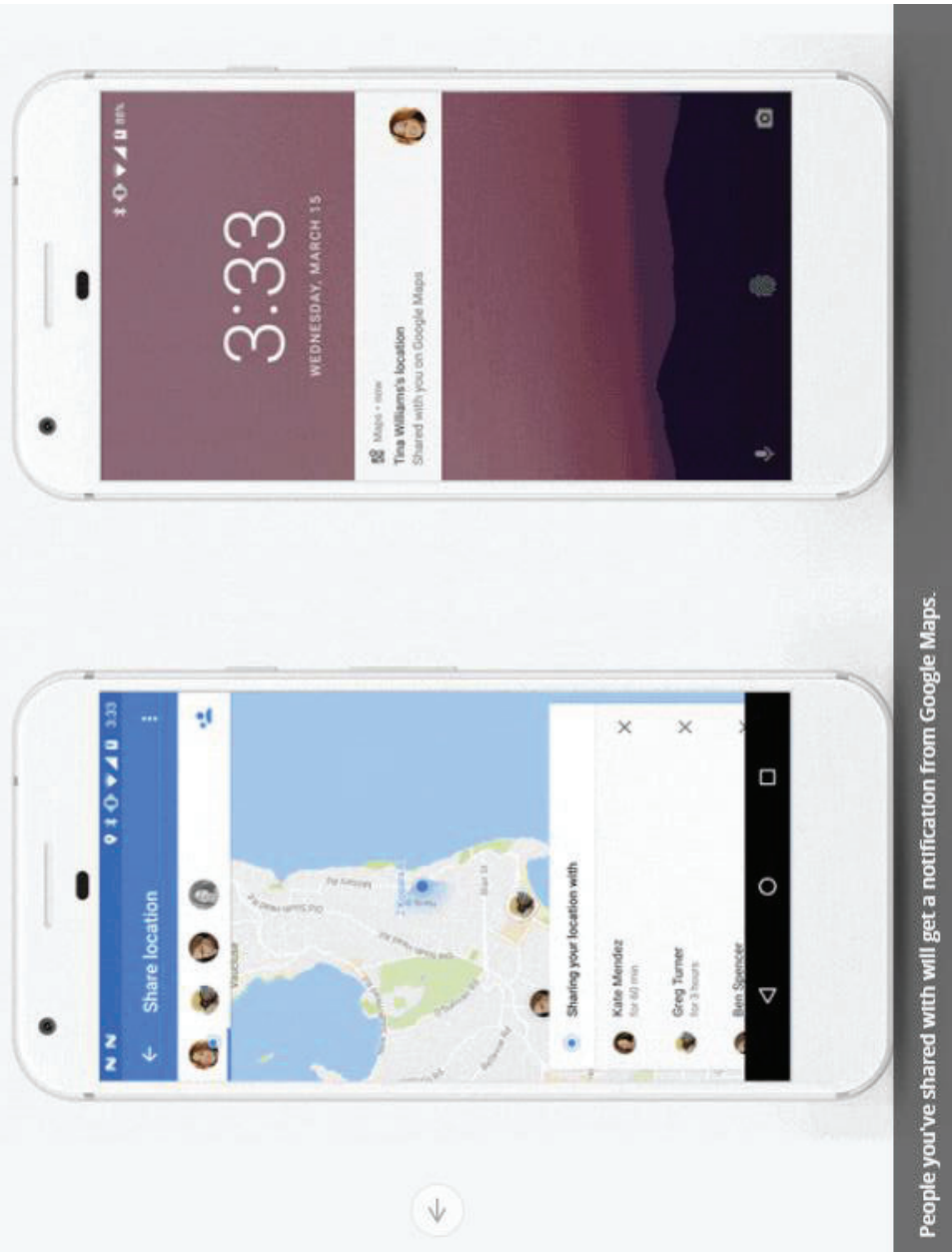


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

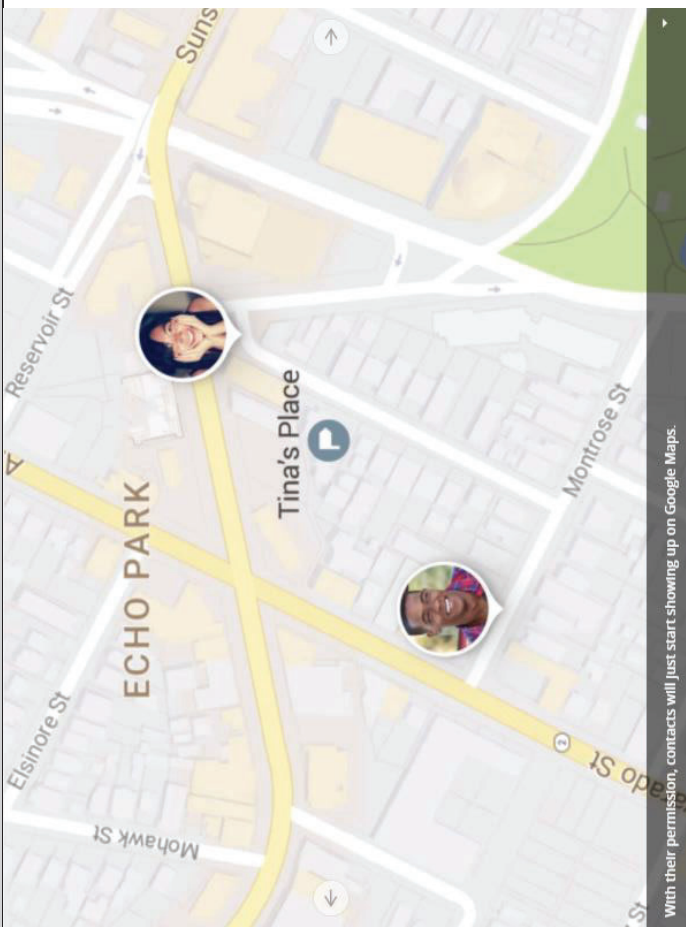


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



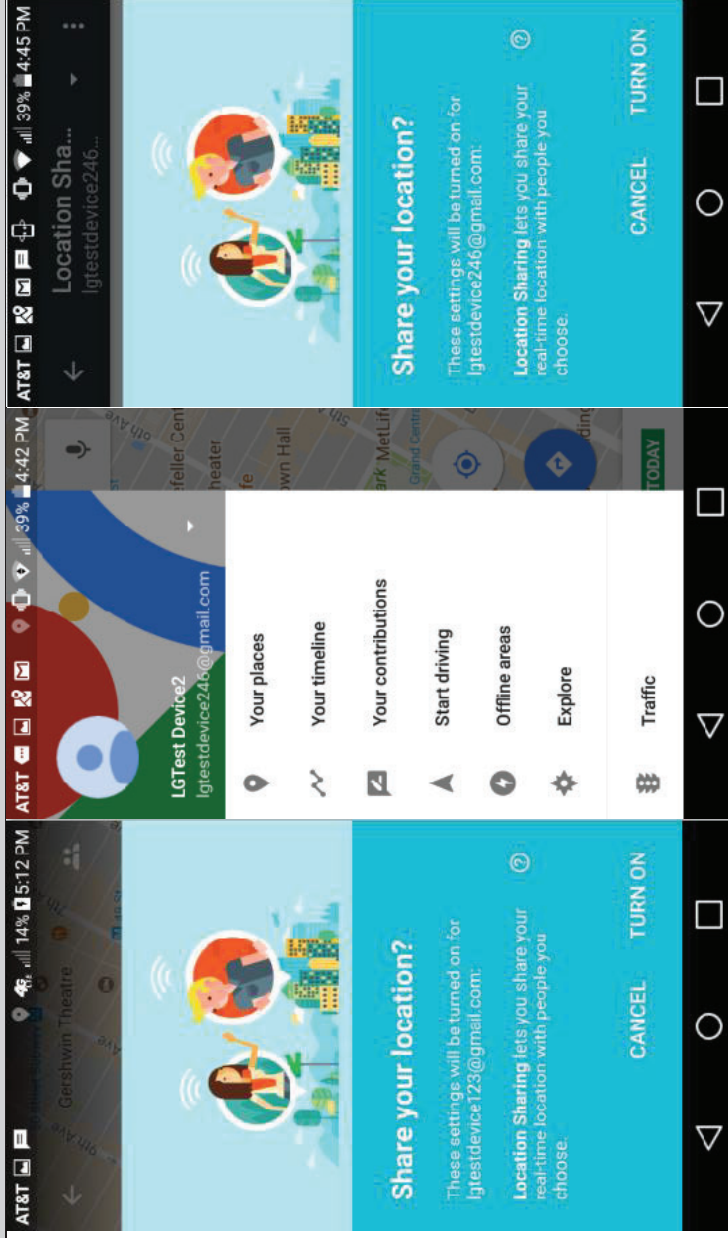
With their permission, contacts will just start showing up on Google Maps.
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | ZTE <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the `ContactsContract.Data` table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the `ContactsContract.RawContacts` table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the `ContactsContract.Contacts` table represents an aggregate of one or more `RawContacts` presumably describing the same person. When data in or associated with the `RawContacts` table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- `ContactsContract.Groups`, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- `ContactsContract.StatusUpdates`, which contains social status updates including IM availability.
- `ContactsContract.AggregationExceptions`, which is used for manual aggregation and disaggregation of raw contacts
- `ContactsContract.Settings`, which contains visibility and sync settings for accounts and groups.
- `ContactsContract.SyncState`, which contains free-form data maintained on behalf of sync adapters
- `ContactsContract.PhoneLookup`, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a `ContactsContract.Data` row that is linked to the raw contact's `_id` value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for `emilyd@gmail.com` (the raw contact row for Thomas Higginson associated with the Google account `emilyd@gmail.com`) has a home email address of `thigg@gmail.com` and a work email address of `thomas.higginson@gmail.com`, the `Contacts Provider` stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the `ContactsContract.Data` table. To help manage this, the `ContactsContract.Data` table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | | |
|--|-------------|---|-----------|---|--|
| Task | Action | Data | MIME type | Notes | |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> | |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 458 private void configureFragments(boolean fromRequest) { 459 if (fromRequest) { 460 ContactListFilter filter = null; 461 int actionCode = mRequest.getActionCode(); 462 boolean searchMode = mRequest.isSearchMode(); 463 final int tabToOpen; 464 switch (actionCode) { 465 case ContactsRequest.ACTION_ALL_CONTACTS: 466 filter = ContactListFilter.createFilterWithType(467 ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468 tabToOpen = TabState.ALL; 469 break; 470 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471 filter = ContactListFilter.createFilterWithType(472 ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473 tabToOpen = TabState.ALL; 474 break; 475 case ContactsRequest.ACTION_FREQUENT: 476 case ContactsRequest.ACTION_STREQUENT: 477 case ContactsRequest.ACTION_STARRED: 478 tabToOpen = TabState.FAVORITES; 479 break; 480 case ContactsRequest.ACTION_VIEW_CONTACT: 481 tabToOpen = TabState.ALL; 482 break; 483 default: 484 tabToOpen = -1; 485 break; 486 } 487 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[28C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices; information of the respective second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claim 1[C], which is incorporated herein by reference in its entirety.</p> <p>For example, the ZTE accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE Mobile Location Service System

2004-01-31




I. Introduction

Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.

The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.

http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html

Send your location

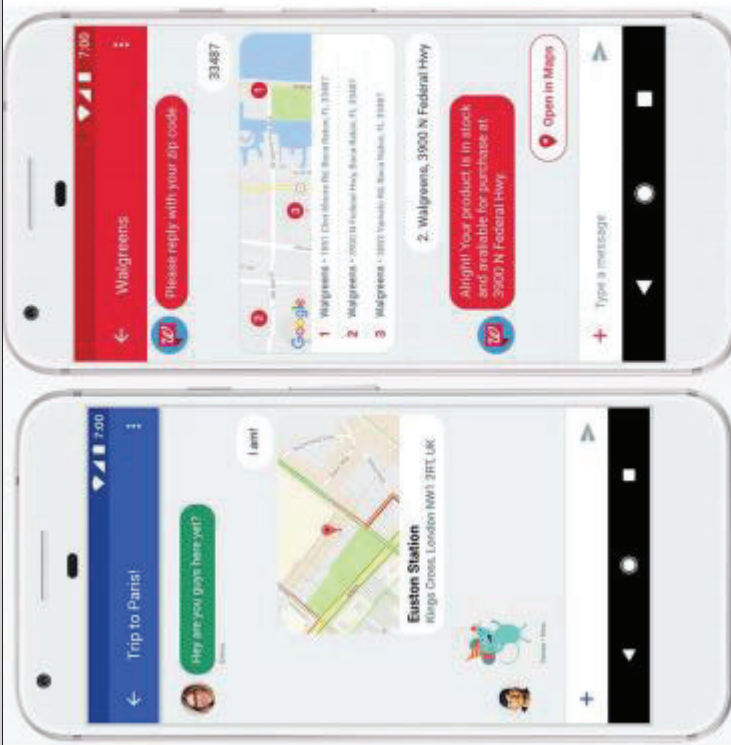
1. Open the Android Messages app .
2. Open or start a conversation.
3. Tap Attach .
4. Tap Location on .
5. To send your location, tap Send .

https://support.google.com/pixelphone/answer/6159880?hl=en&ref_topic=6211804

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://blog.google/topics/rcs/delivering-rcs-messaging-android-users-worldwide/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share a location or place

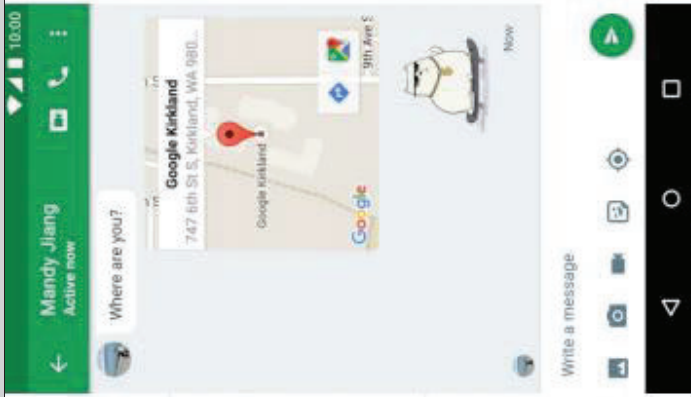
Share your location

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location.
- 4. Tap Select this location > Select.

Share a place

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location > Search Q.
- 4. Type in a location or address.
- 5. Tap Select.

https://support.google.com/hangouts/answer/3115410?visit_id=l-636271867303650973-2491837168&rd=1&co=GENIE.Platform%3DAndroid&oco=1
<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>



Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.

The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p><u>Further regarding Google Maps</u>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p><u>Exemplary Support for Google Maps:</u></p> | |





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD



Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1

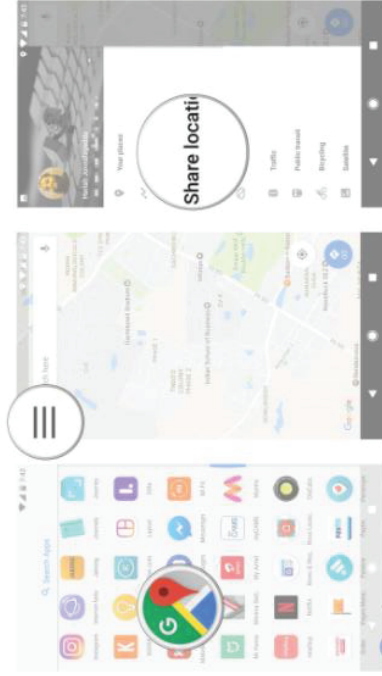
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

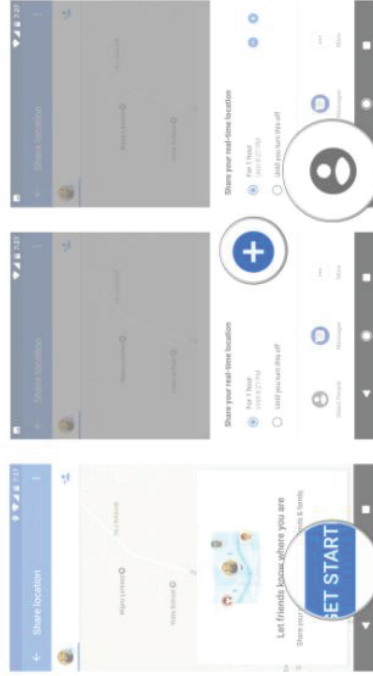
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



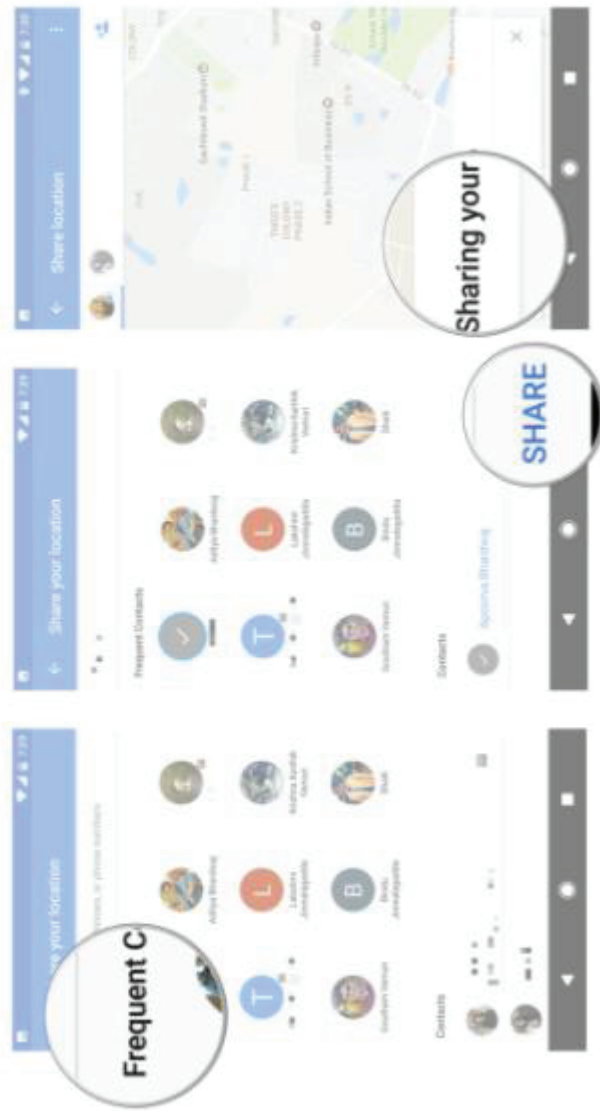
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

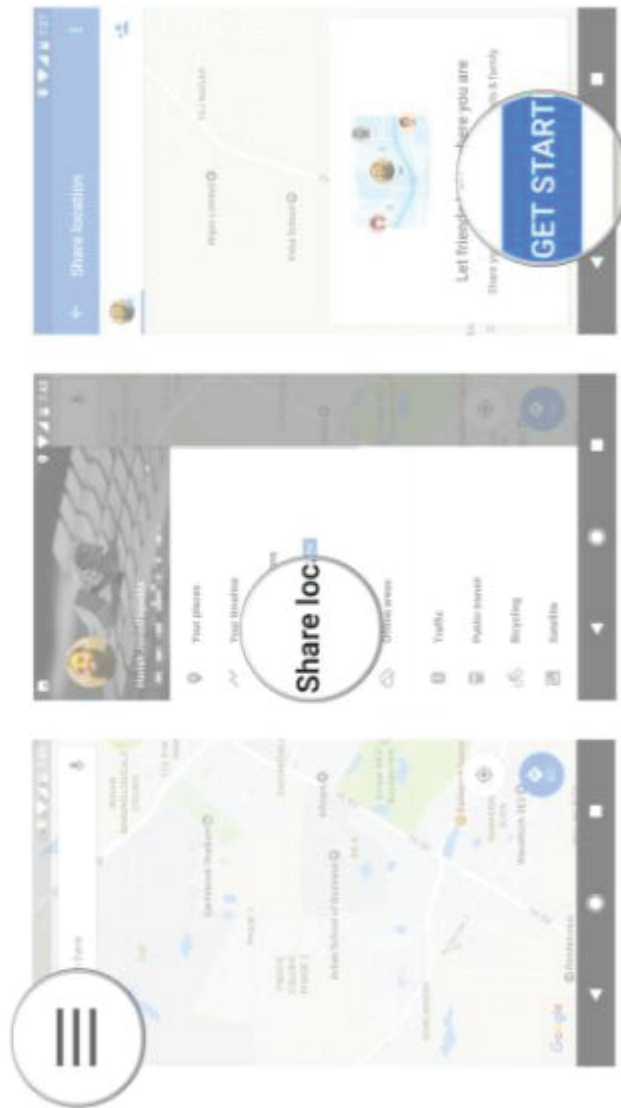
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



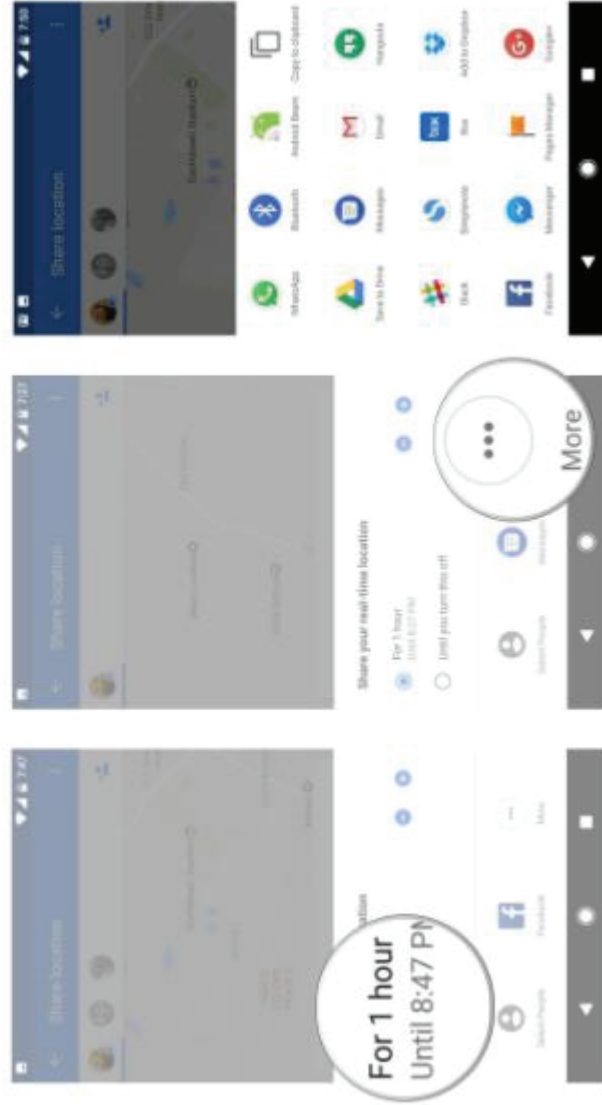
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

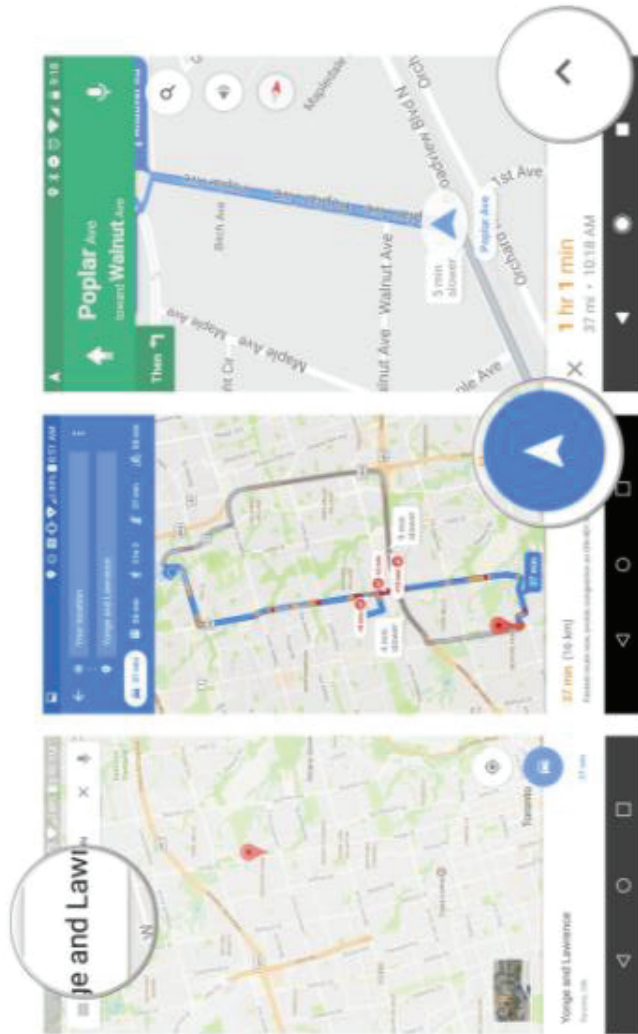
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



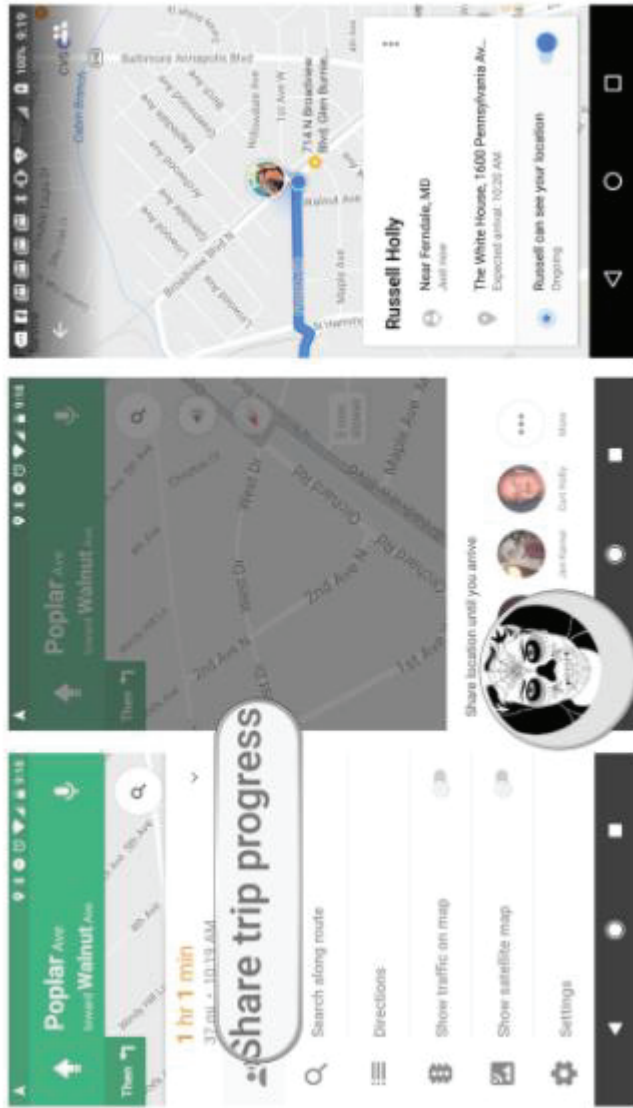
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



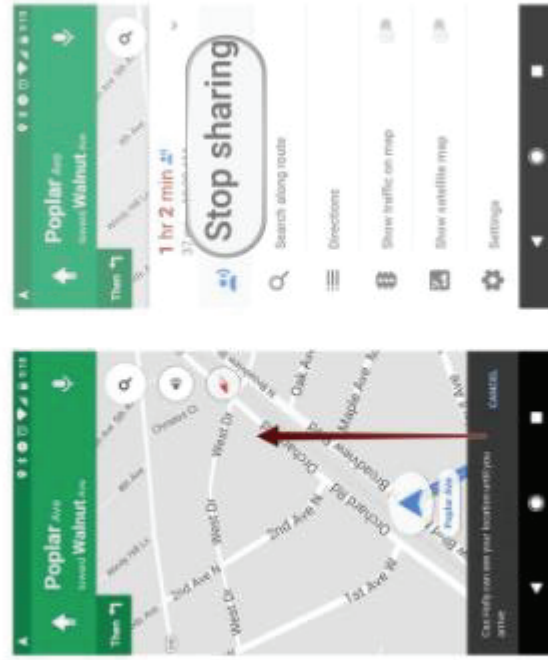
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!





Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>
As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

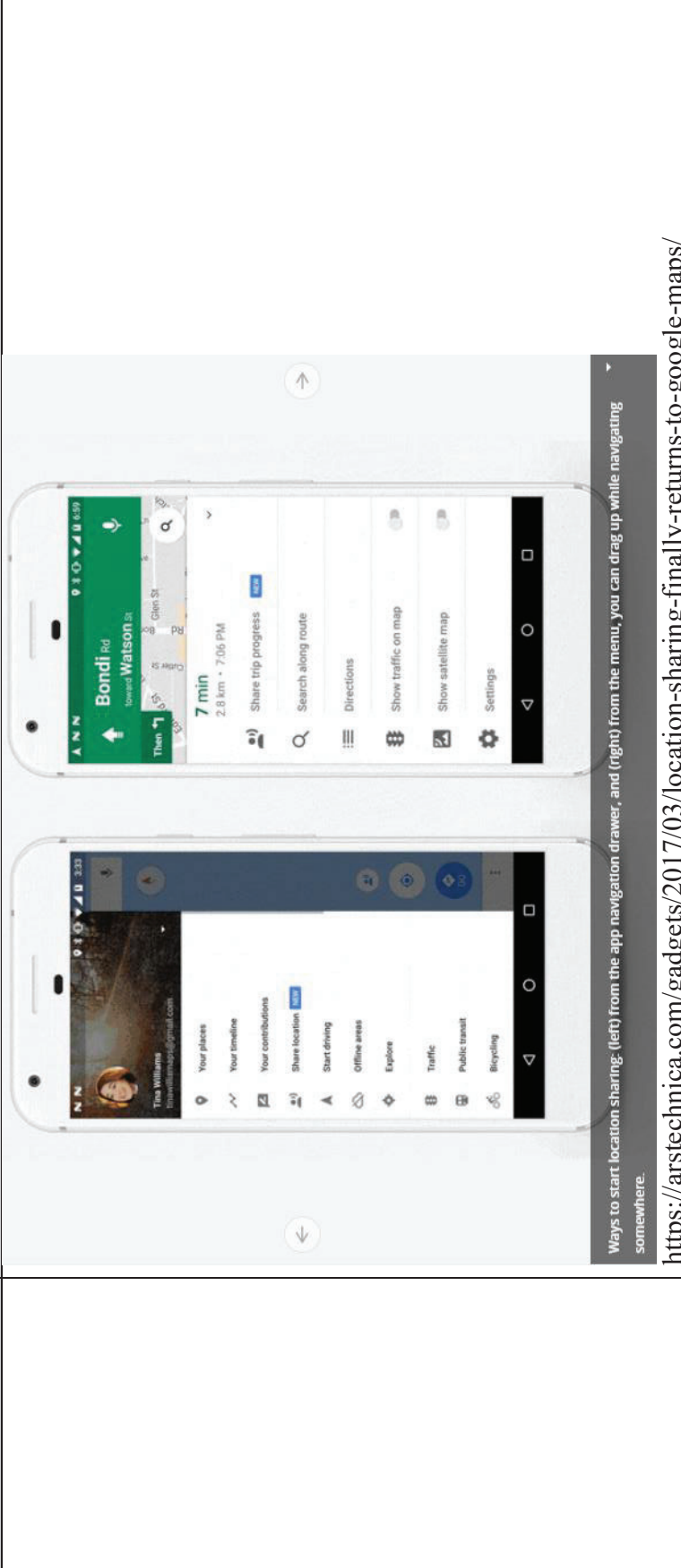


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

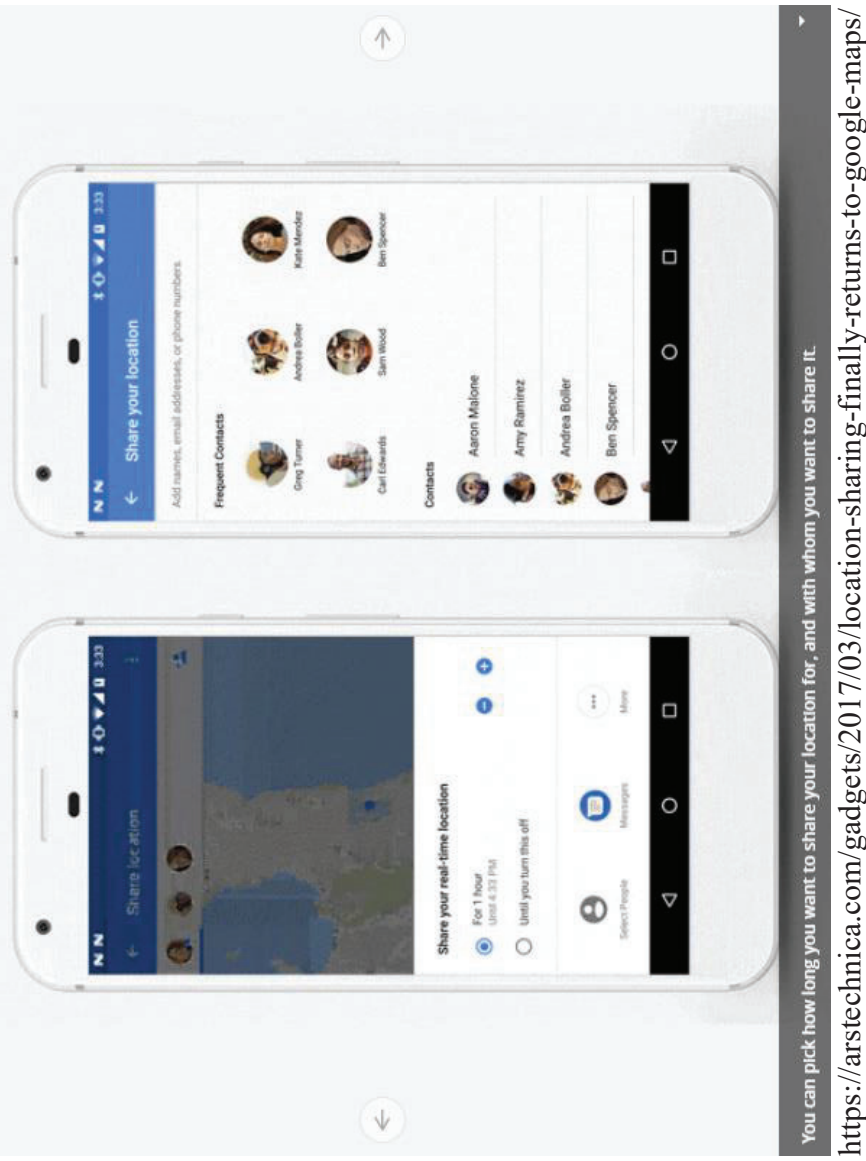


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

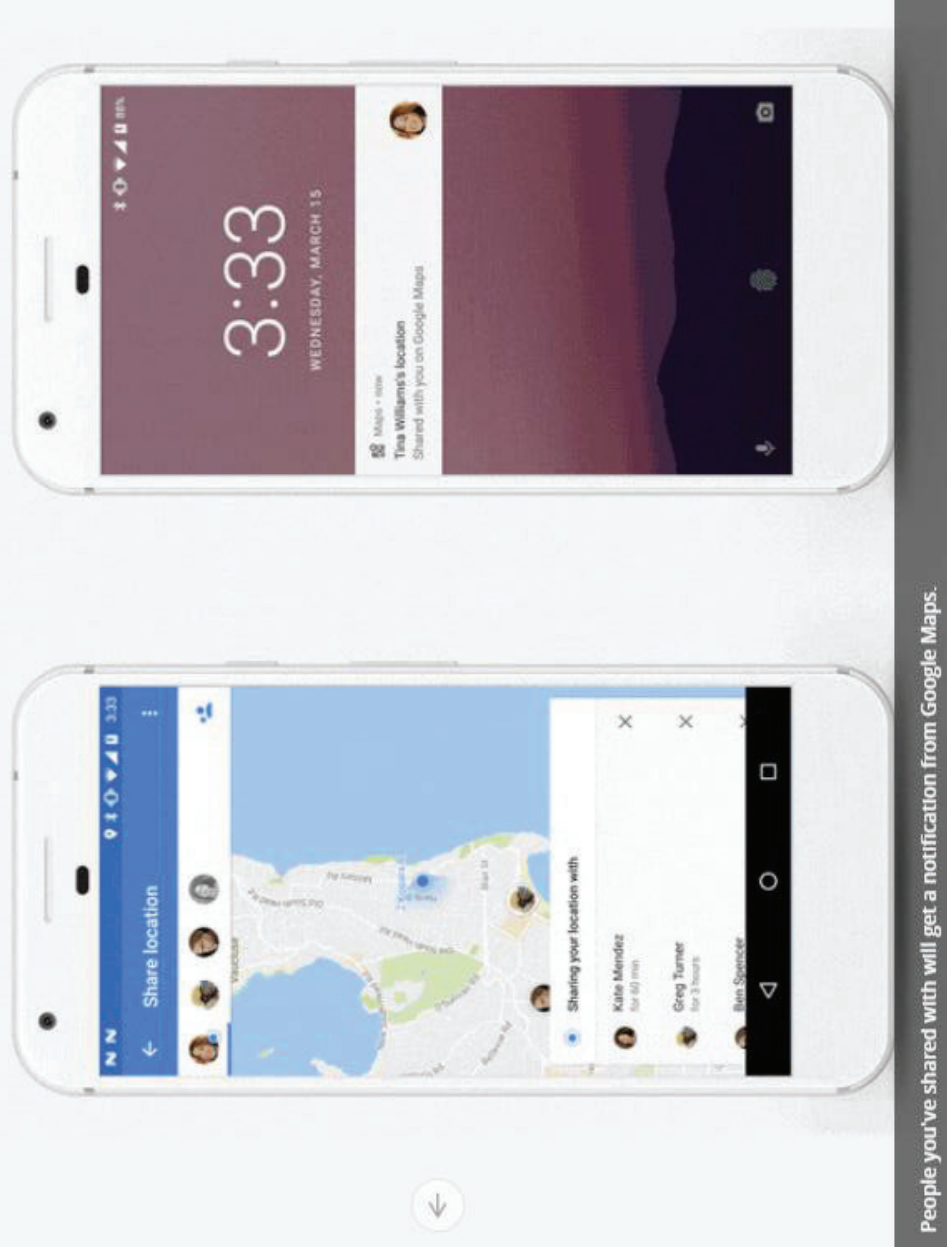
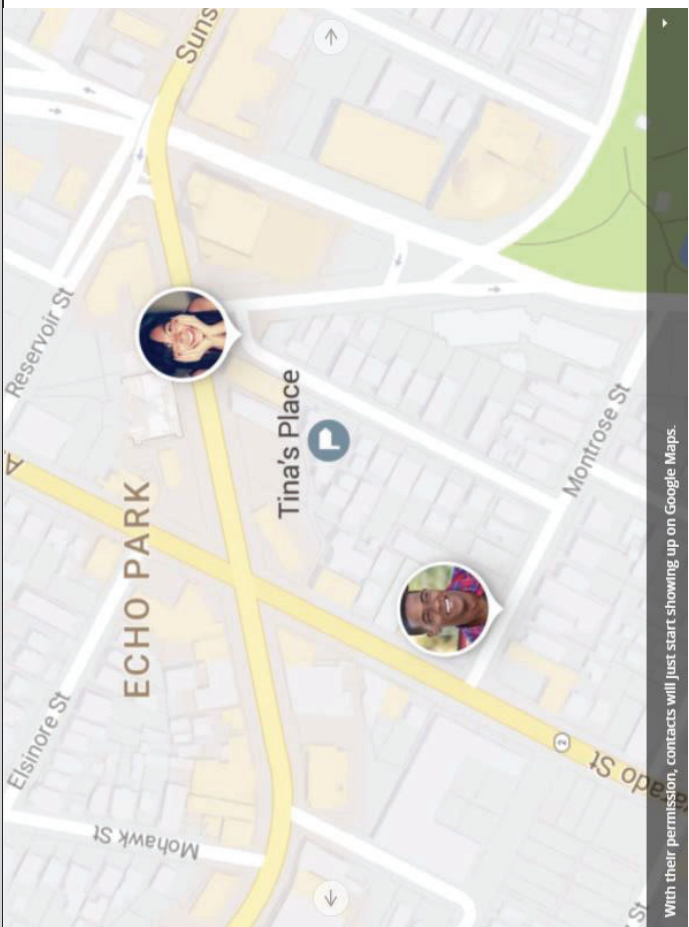


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

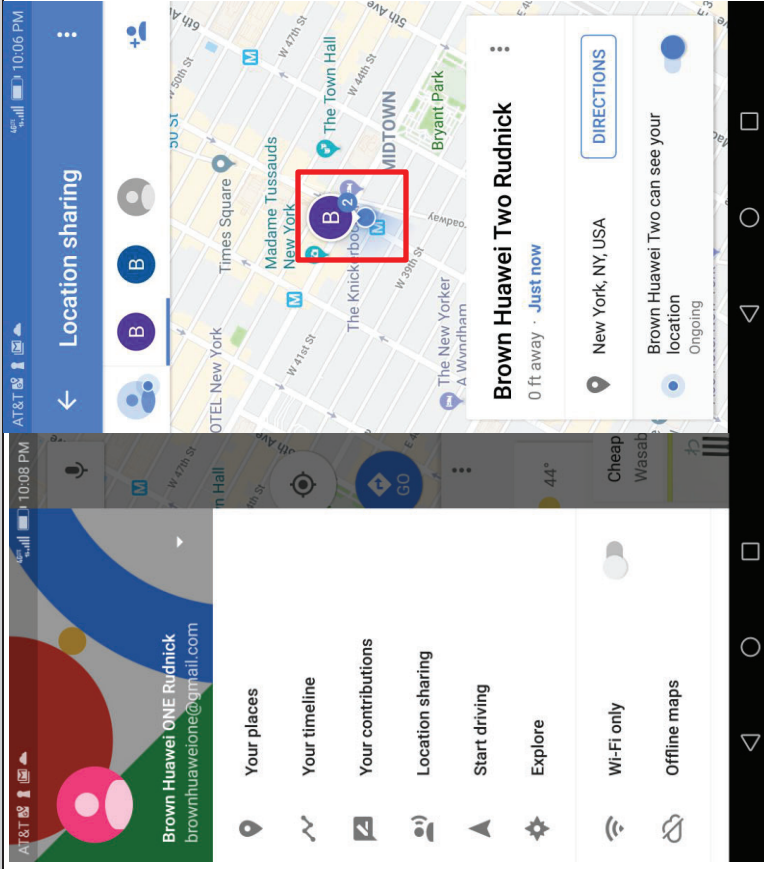


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>
Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Location information is shared via IP-based communication resulting in map that displays location information

Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | ZTE <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider-provider.html</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contact's _id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|--|-------------|---|-----------|---|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-436

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS       = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-441

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

B-442

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44  * Class that sends chat message via SMS.
45  *
46  * The interface emulates a blocking sending similar to making an HTTP request.
47  * It calls the SmsManager to send a (potentially multipart) message and waits
48  * on the sent status on each part. The waiting has a timeout so it won't wait
49  * forever. Once the sent status of all parts received, the call returns.
50  * A successful sending requires success status for all parts. Otherwise, we
51  * pick the highest level of failure as the error for the whole message, which
52  * is used to determine if we need to retry the sending.
53  */
54  public class SmsSender {
55      private static final String TAG = LogUtil.BUGLE_TAG;
56
57      public static final String EXTRA_PART_ID = "part_id";
58
59      /*
60       * A map for pending sms messages. The key is the random request UUID.
61       */
62      private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63          new ConcurrentHashMap<Uri, SendResult>();
64
65      private static final Random RANDOM = new Random();
66
67      // Whether we should send multipart SMS as separate messages
68      private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*status*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178 } else if (METHOD_GET.equals(method)) {
179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180         logHttpHeaders(connection.getRequestProperties());
181     }
182     connection.setRequestMethod(METHOD_GET);
183 }
184 // Get response
185 final int responseCode = connection.getResponseCode();
186 final String responseMessage = connection.getResponseMessage();
187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189     logHttpHeaders(connection.getHeaderFields());
190 }
191 if (responseCode / 100 != 2) {
192     throw new MmsHttpException(responseCode, responseMessage);
193 }
194 final InputStream in = new BufferedInputStream(connection.getInputStream());
195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196 final byte[] buf = new byte[4096];
197 int count = 0;
198 while ((count = in.read(buf)) > 0) {
199     byteOut.write(buf, 0, count);
200 }
201 in.close();
202 final byte[] responseBody = byteOut.toByteArray();
203 Log.d(MmsService.TAG, "HTTP: response size="
204     + (responseBody != null ? responseBody.length : 0));
205 return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-453

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1520 220 1587">ZTE</p> <p data-bbox="233 506 302 1587">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <pre data-bbox="362 1121 391 1562">public static LocationRequest create ()</pre> <p data-bbox="423 1073 448 1572">Create a location request with default parameters.</p> <p data-bbox="480 464 540 1572">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="565 1472 589 1551">Returns</p> <ul data-bbox="610 1289 634 1551" style="list-style-type: none">• a new location request <p data-bbox="651 306 678 1587">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-----|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <pre data-bbox="243 1081 284 1564">public Task<Location> getLastLocation ()</pre> <p data-bbox="308 997 341 1575">Returns the best most recent location currently available.</p> <p data-bbox="365 399 430 1575">If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p data-bbox="454 367 519 1575">This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre data-bbox="576 882 617 1564">public Task<LocationAvailability> getLocationAvailability ()</pre> <p data-bbox="641 409 706 1575">Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p data-bbox="730 630 763 1575">If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p data-bbox="787 430 852 1575">Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p data-bbox="860 199 925 1585">https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|--------------------|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|--------------------|------------|

| | |
|--|---|
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |
|--|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p>Public Constructors</p> <pre> public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) </pre> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> <pre> public void getMapAsync (OnMapReadyCallback callback) </pre> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback The callback object that will be triggered when the map is ready to be used.</p> <pre> public final void onCreate (Bundle savedInstanceState) </pre> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[28D] transmitting IP-based messages including a location of the first device to the respective second devices;</p> | <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claim 1[D], which is incorporated herein by reference in its entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C] and [28C].</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.


4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:

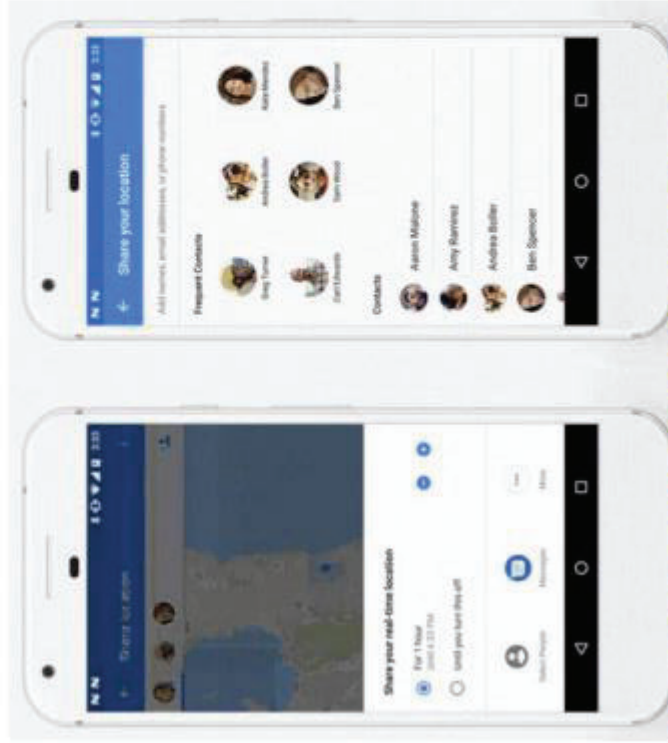
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. If you haven't already, add their Gmail address to your Google Contacts [\[2\]](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap the Menu  > **Share location** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

<https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&hl=en>



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Below are exemplary methods used by Google applications to obtain, send, and receive locations.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

The Google Play services Location API

The Google Play services Location API is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:

- Determine the device location.
- Listen for location changes.
- Determine the mode of transportation, if the device is moving.
- Create and monitor predefined geographical regions, known as geofences.

The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class Making Your App Location Aware or the Location API Reference. Code examples are included as part of the Google Play services SDK.

<https://developers.google.com/maps/documentation/android-api/location>

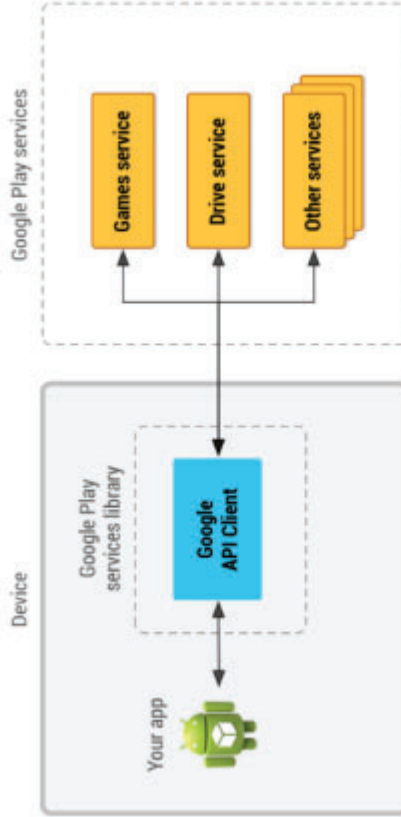


Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.

<https://developers.google.com/android/guides/api-client#Starting>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get the Last Known Location

Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's `getLastLocation()` method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the [Specify App Permissions](#) section of this document.

To request the last known location, call the `getLastLocation()` method, passing it your instance of the `GoogleApiClient` object. Do this in the `onConnected()` callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:

```

public class MainActivity extends AppCompatActivity implements
    ConnectionCallbacks, OnConnectionFailedListener {
    ...
    @Override
    public void onConnected(Bundle connectionHint) {
        mLastLocation = LocationServices.FusedLocationApi.getLastLocation(
            mGoogleApiClient);
        if (mLastLocation != null) {
            mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));
            mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));
        }
    }
}

```

The `getLastLocation()` method returns a `Location` object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.

<https://developer.android.com/training/location/retrieve-current.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Determining the user's current location</p> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() { var startPos; var geoSuccess = function(position) { startPos = position; document.getElementById('startLat').innerHTML = startPos.coords.latitude; document.getElementById('startLon').innerHTML = startPos.coords.longitude; }; navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p>https://developers.google.com/web/fundamentals/native-hardware/user-location/</p> |
| <p>[28E] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claim 1[E], which is incorporated herein by reference in its entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>corresponding to the respective locations of the second devices;</p> | <p>ZTE</p> <p>Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p>Exemplary Support for Google Maps:</p> <p>Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


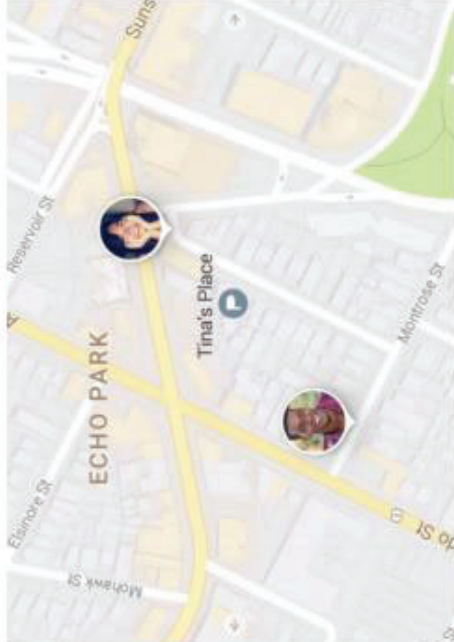
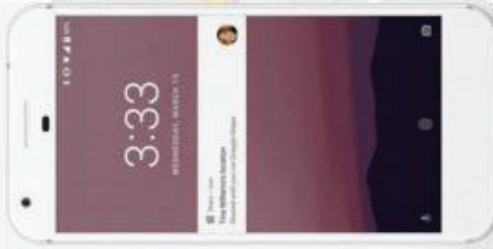
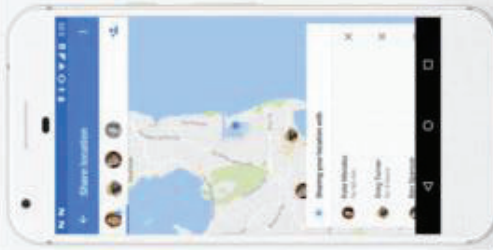
| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app.
2. Tap the menu icon > **Share location**.
3. Choose someone.

- To see an updated location, tap on a friend's icon > **More** > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app.
2. On the map, tap the 'Share' icon.
3. At the bottom, tap **More**.
4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

https://support.google.com/maps/answer/7326816?hl=en&ref_topic=3092425&co=GENIE.Platform%3DAndroid&oc=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap More  > **Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share.**
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap More .
4. To temporarily hide someone, tap **Hide from map.** You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

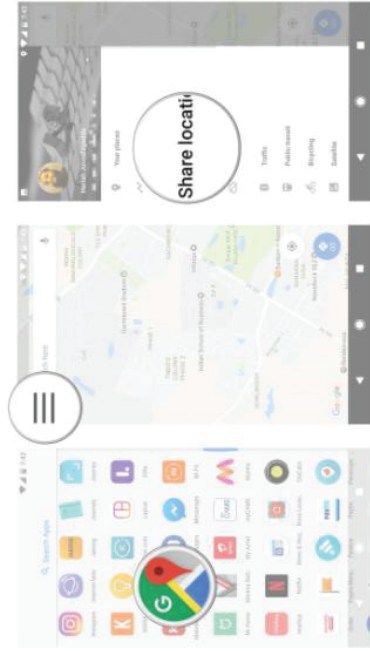
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

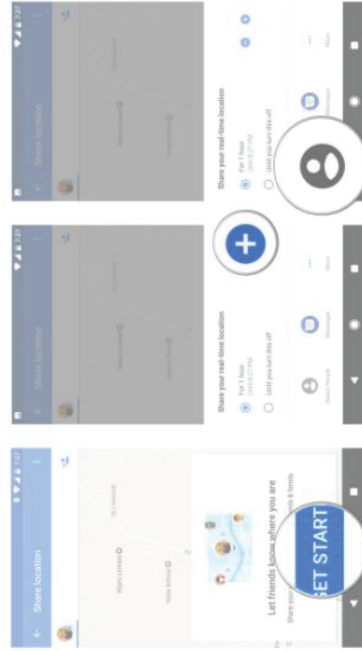
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



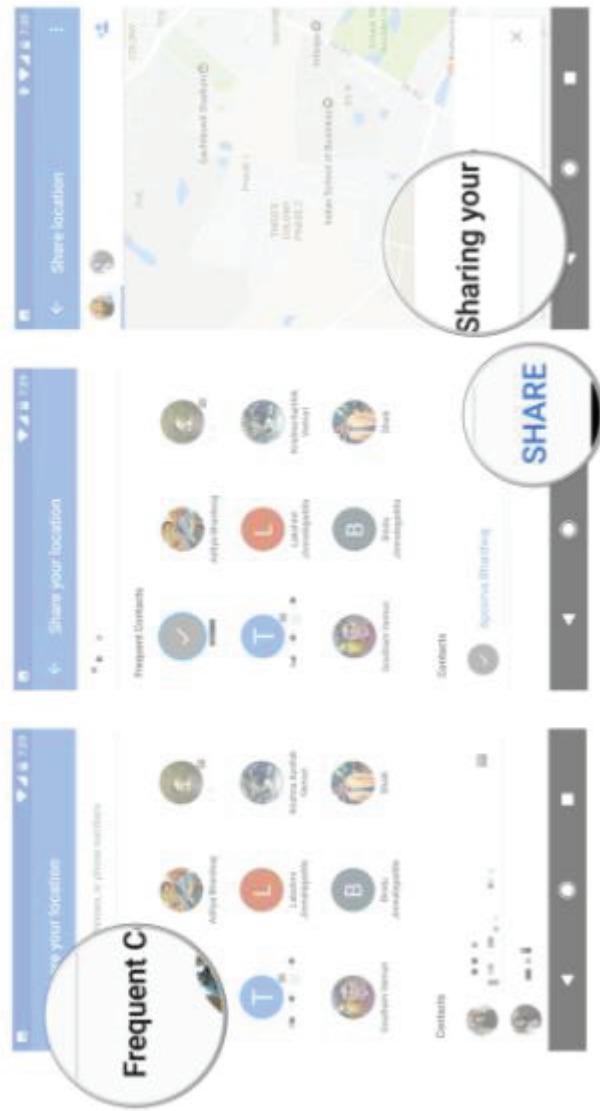
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

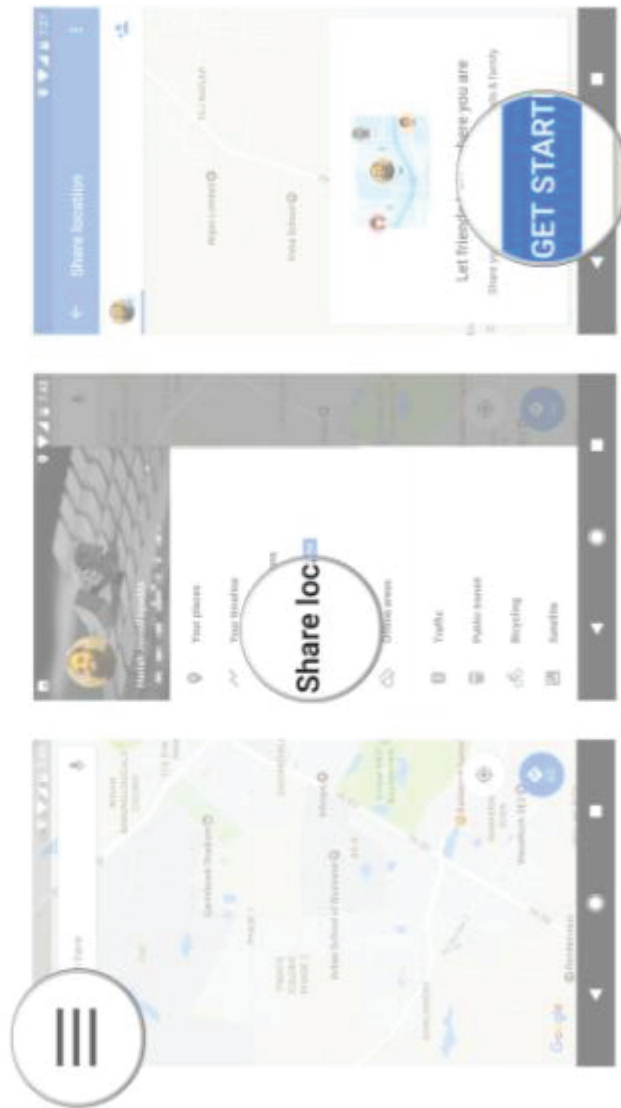
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



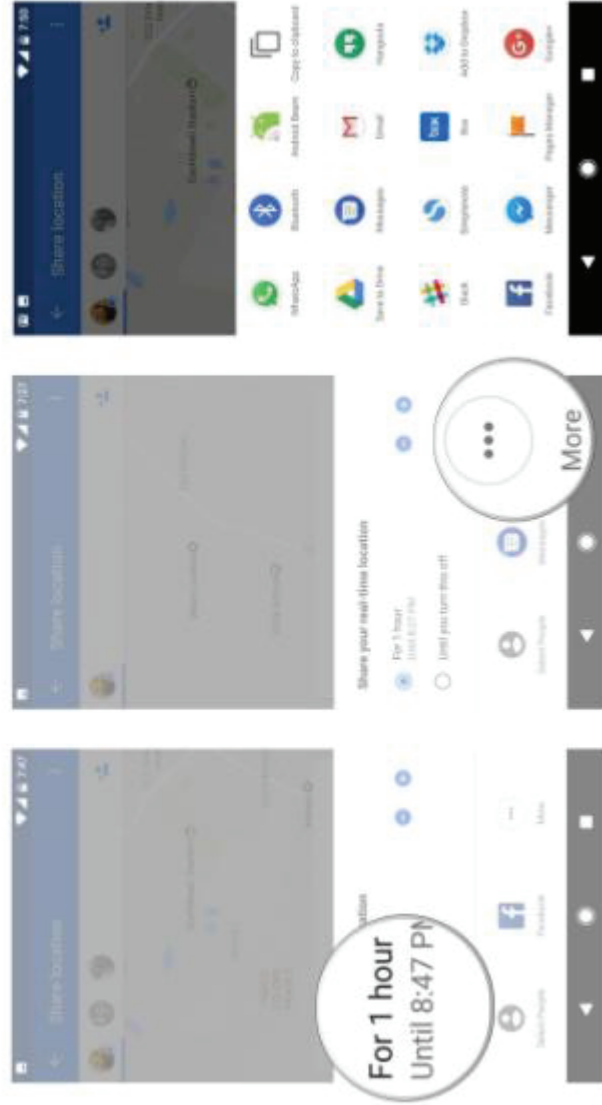
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

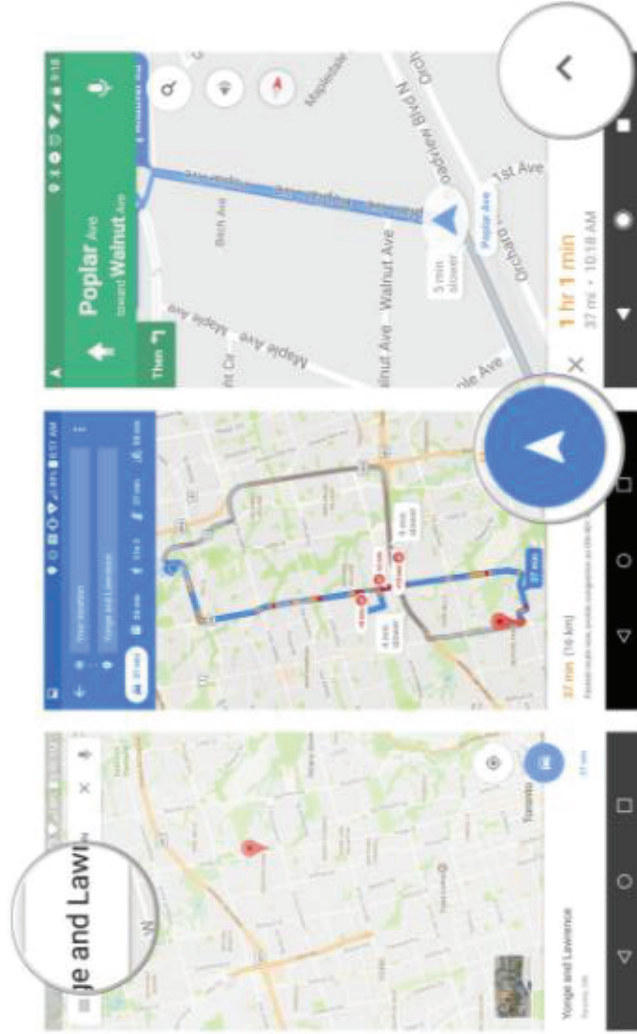
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



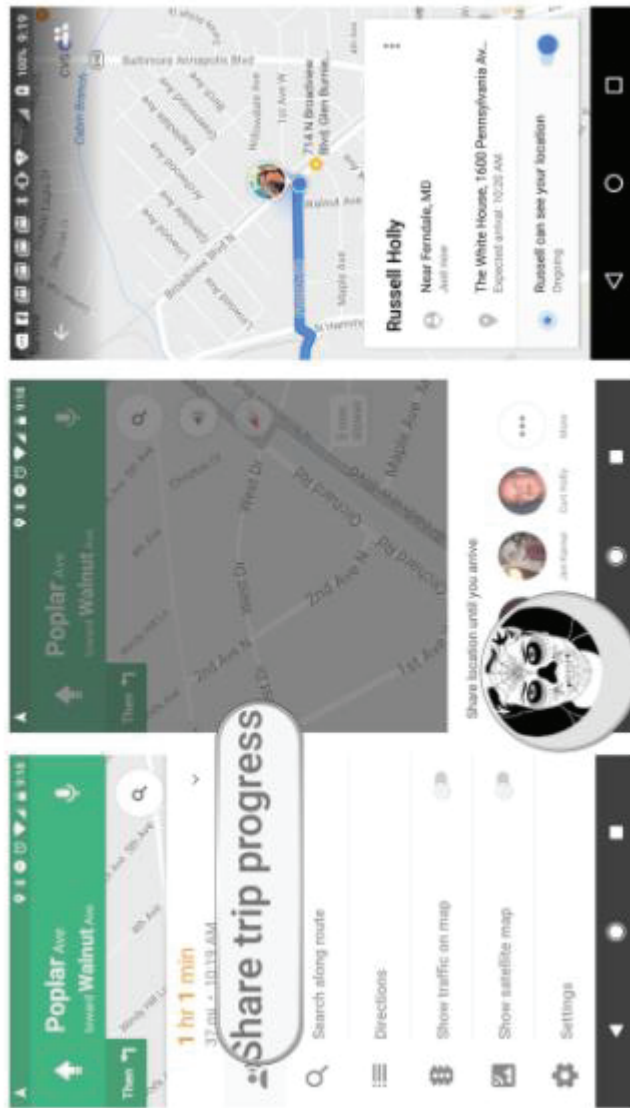
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



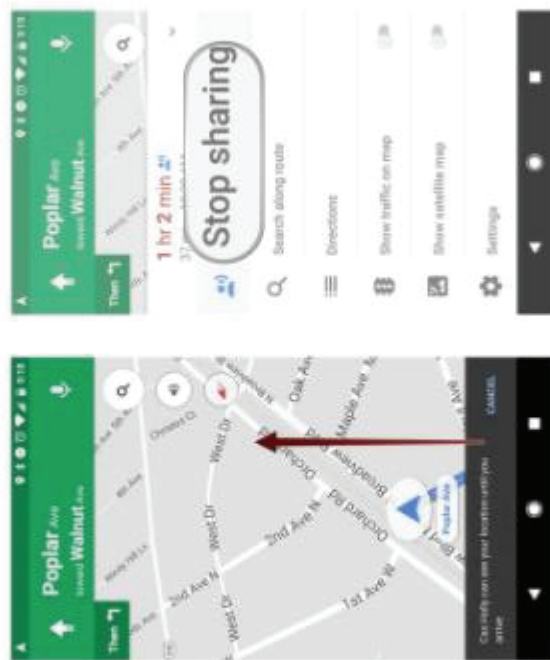
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

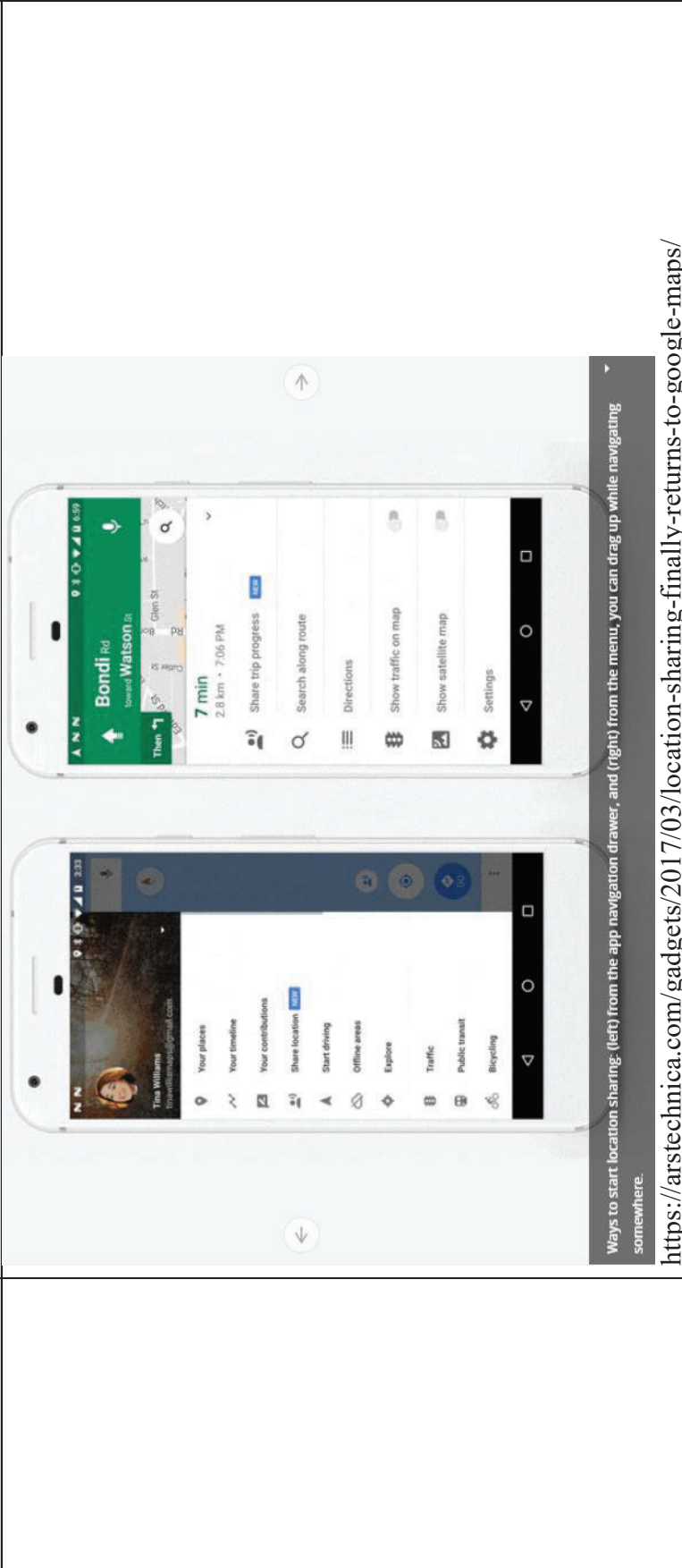


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

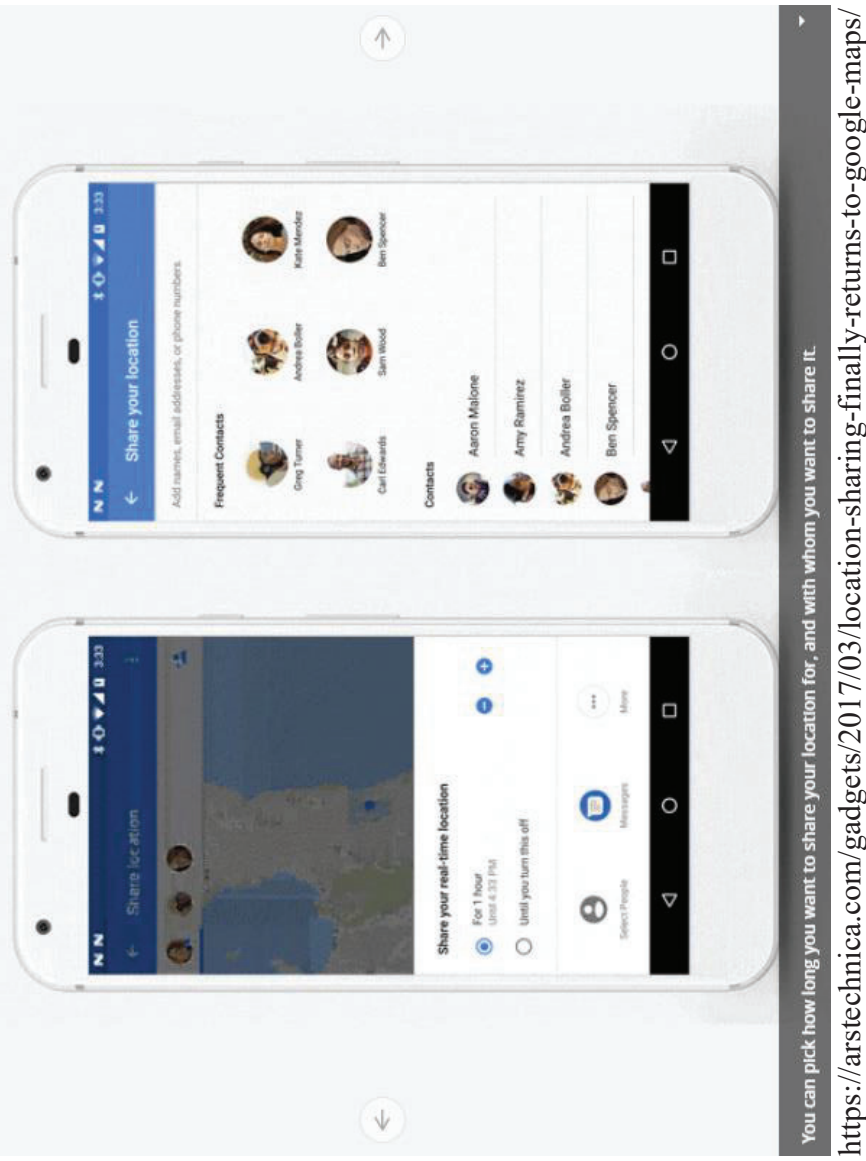
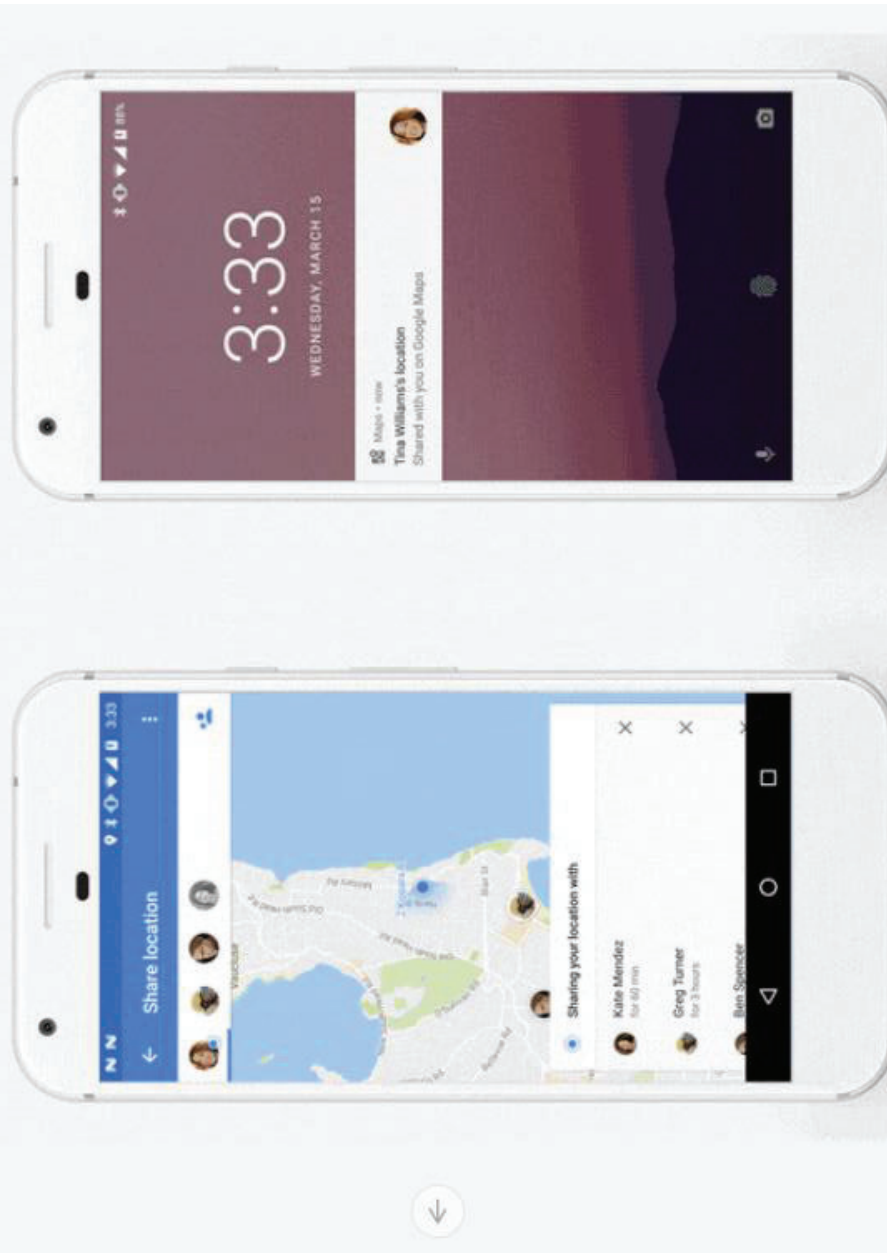


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



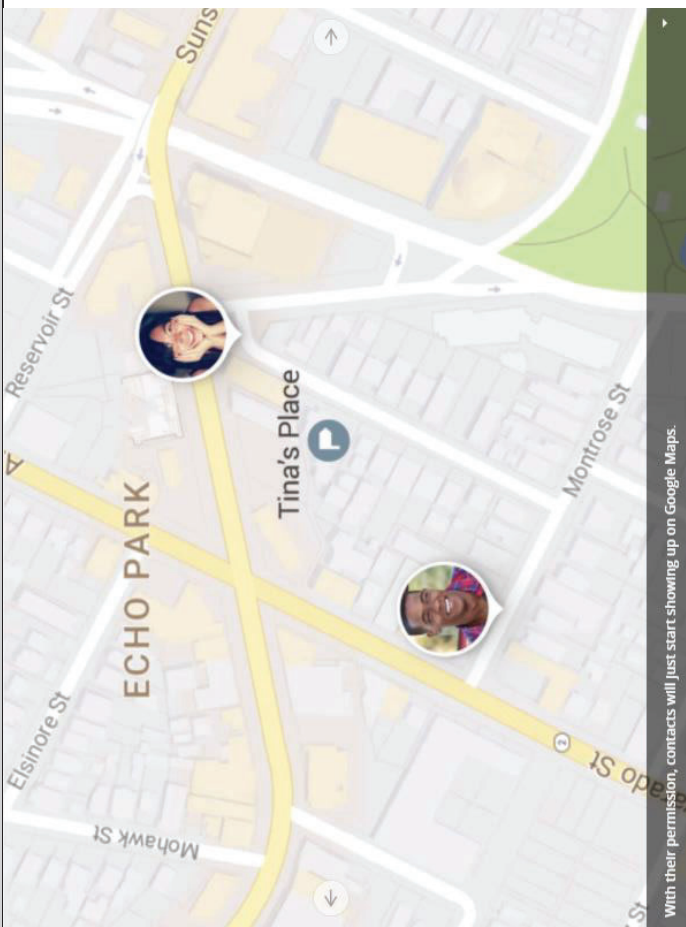
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



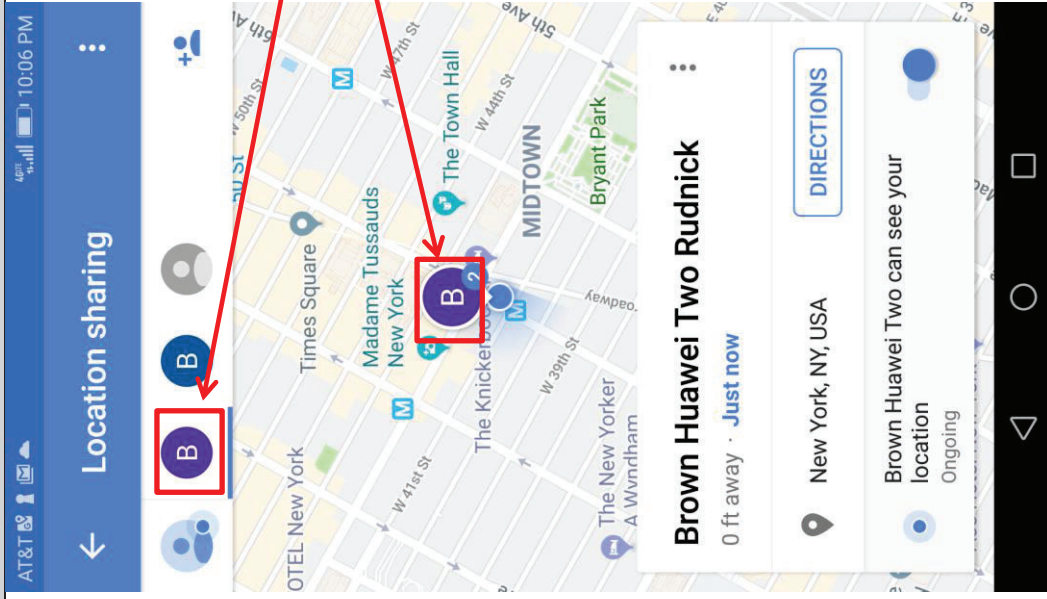
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary User
Selectable Symbols

Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p>ZTE</p> <p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre>public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p>Returns</p> <ul style="list-style-type: none">• a new location request <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>public Task<Location> getLastLocation ()</pre> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre>public Task<LocationAvailability> getLocationAvailability ()</pre> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|---------------------------|---|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|--------------------|------------|

| | |
|---|--|
| <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> | |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener Public Constructors public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) https://developers.google.com/android/reference/com/google/android/gms/maps/MapView |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>public void getMapAsync (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>Goog1eMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>Goog1eMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void onCreate (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[28F] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claim 1[F], which is incorporated herein by reference in its entirety.</p> | <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> <p>to the one or more second devices;</p> | <p>ZTE</p> <p><u>Exemplary Support for Google Maps:</u></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |
|--|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1596">ZTE</p> <p data-bbox="259 693 300 1228">ZTE Mobile Location Service System</p> <p data-bbox="332 903 365 1050">2004-01-31</p> <p data-bbox="430 1386 462 1564">I. Introduction</p> <p data-bbox="470 357 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 357 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1596">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 🗣️ > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap [Settings] > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
|---------------------------|---|




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER **ANDROID** IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts [\[?\]](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap **More**  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > **More**  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAn

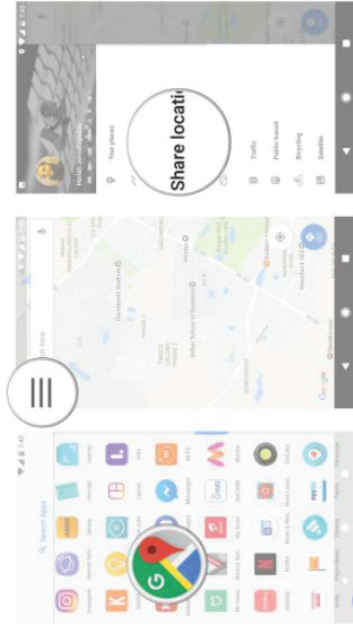
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

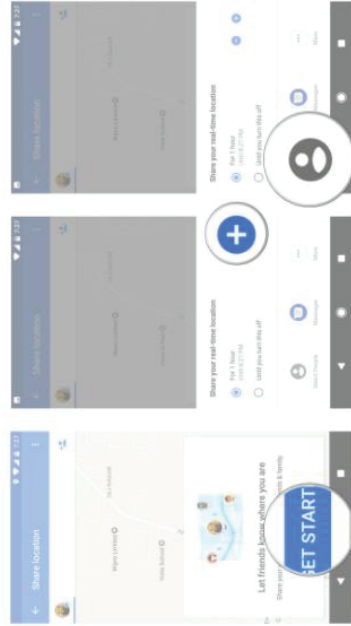
ZTE
droid&oco=1

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



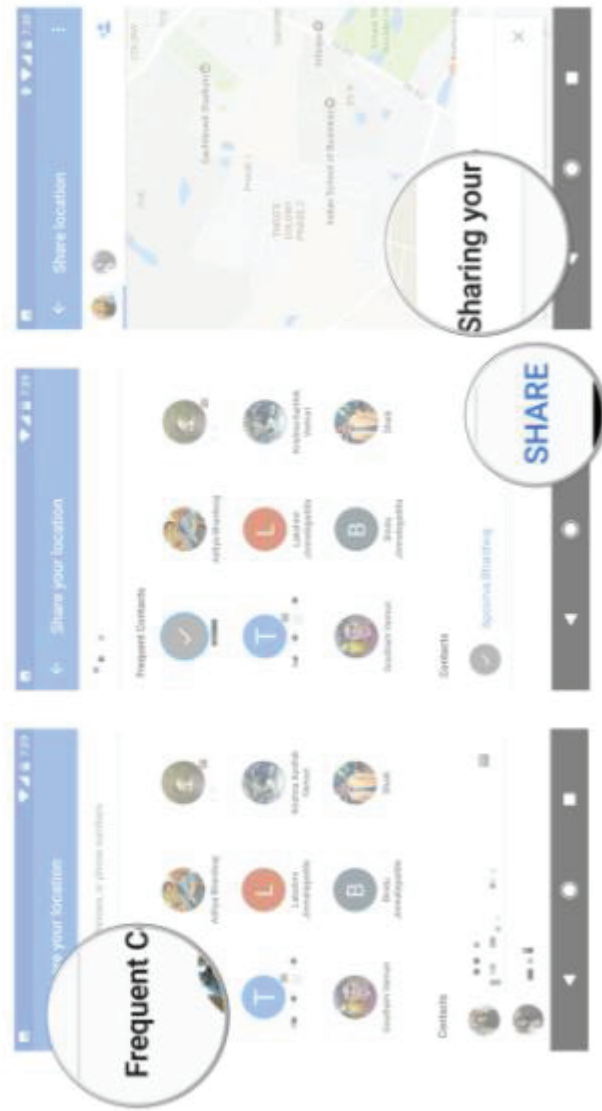
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

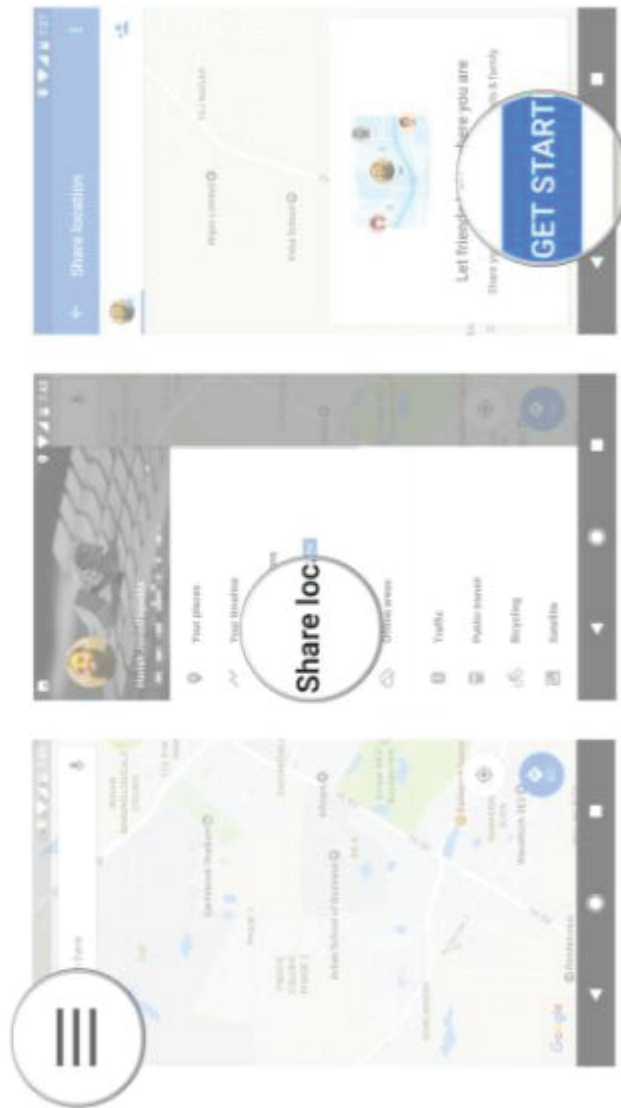
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



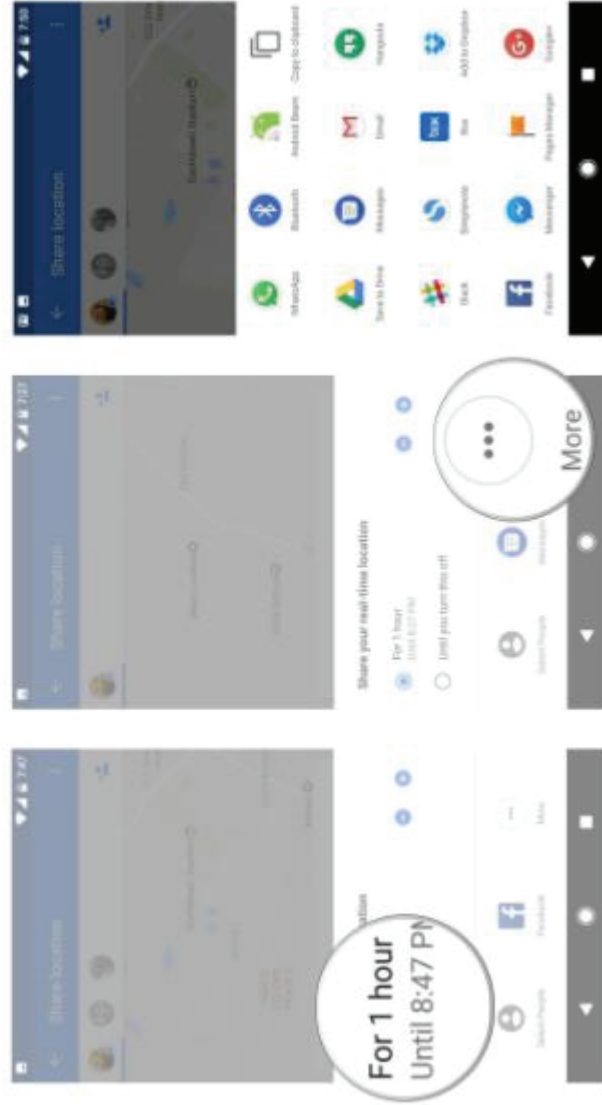
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

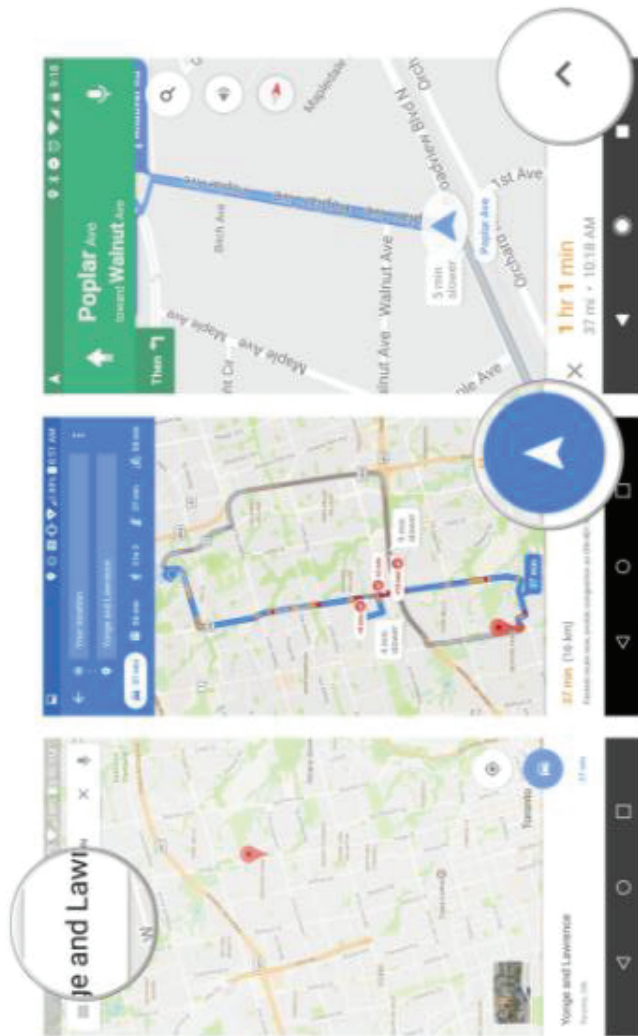
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



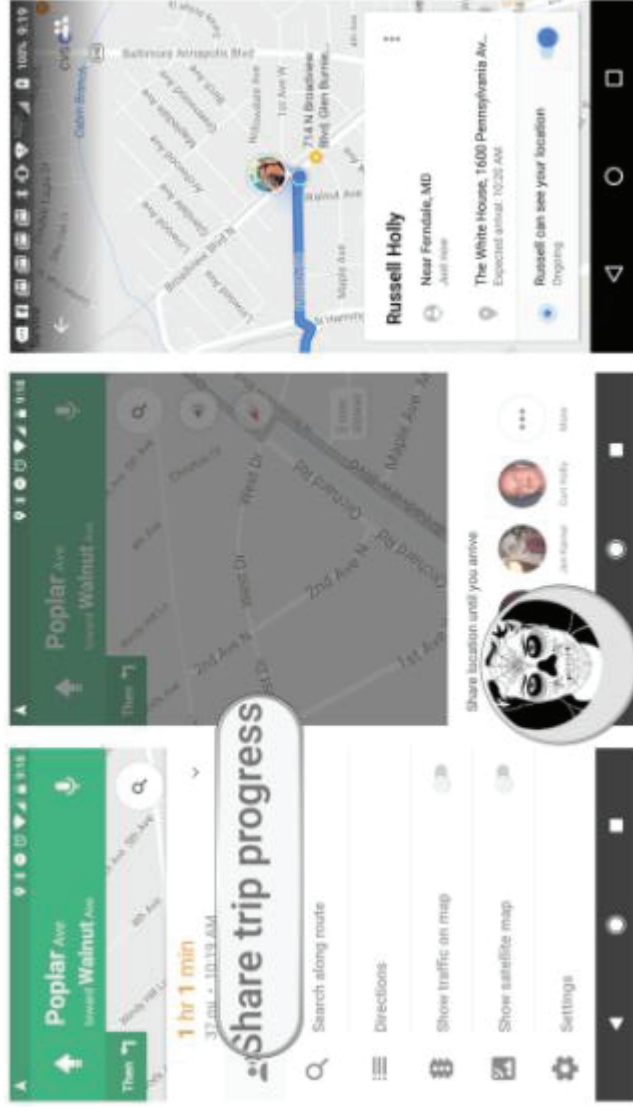
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



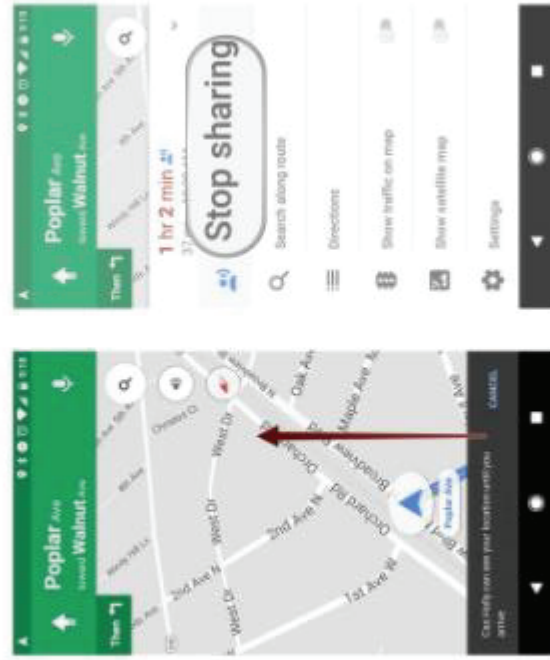
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>



As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

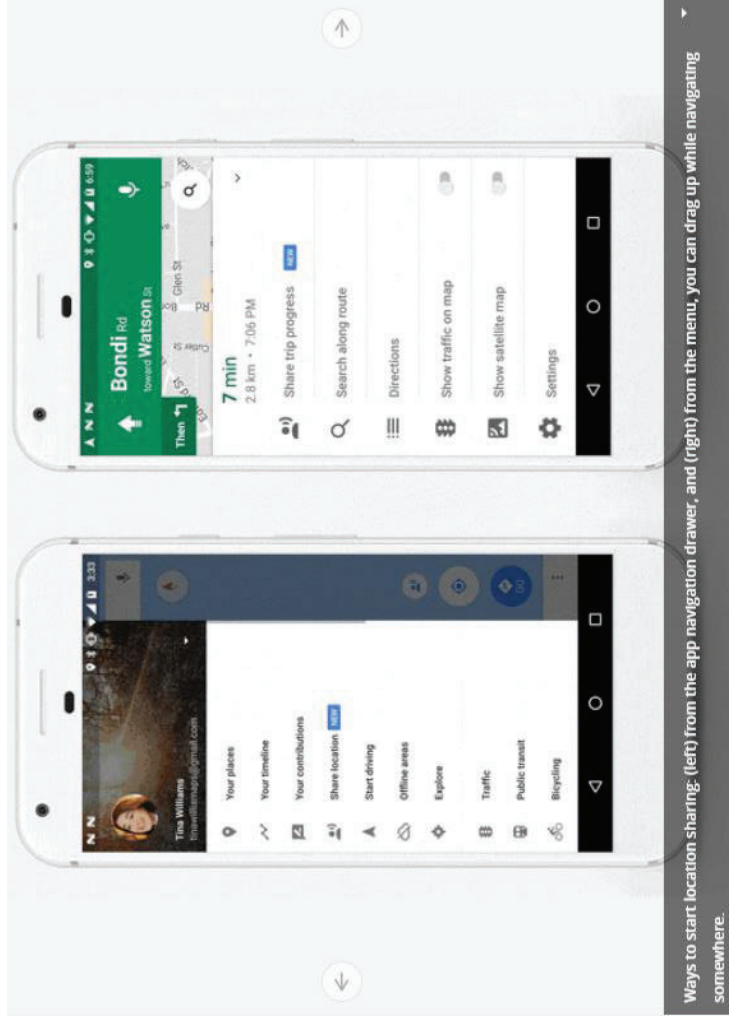
US9408055B2

ZTE

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

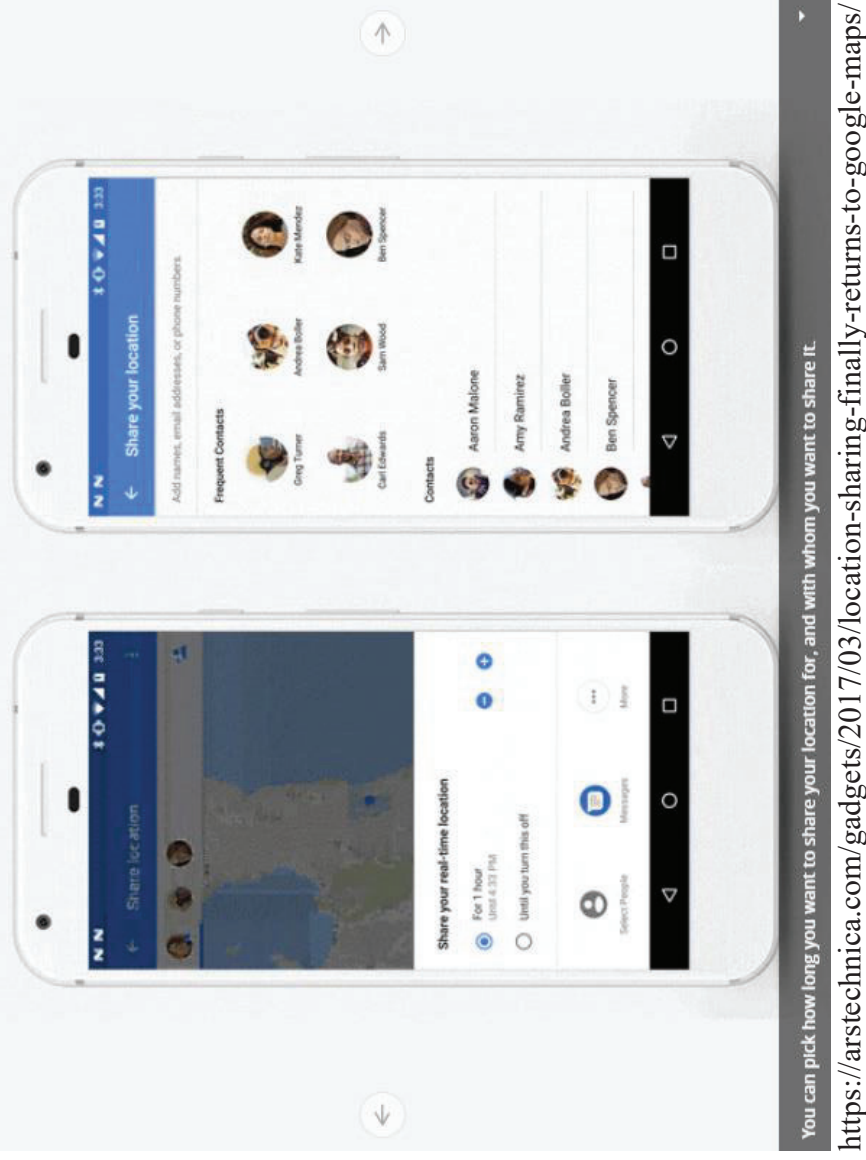
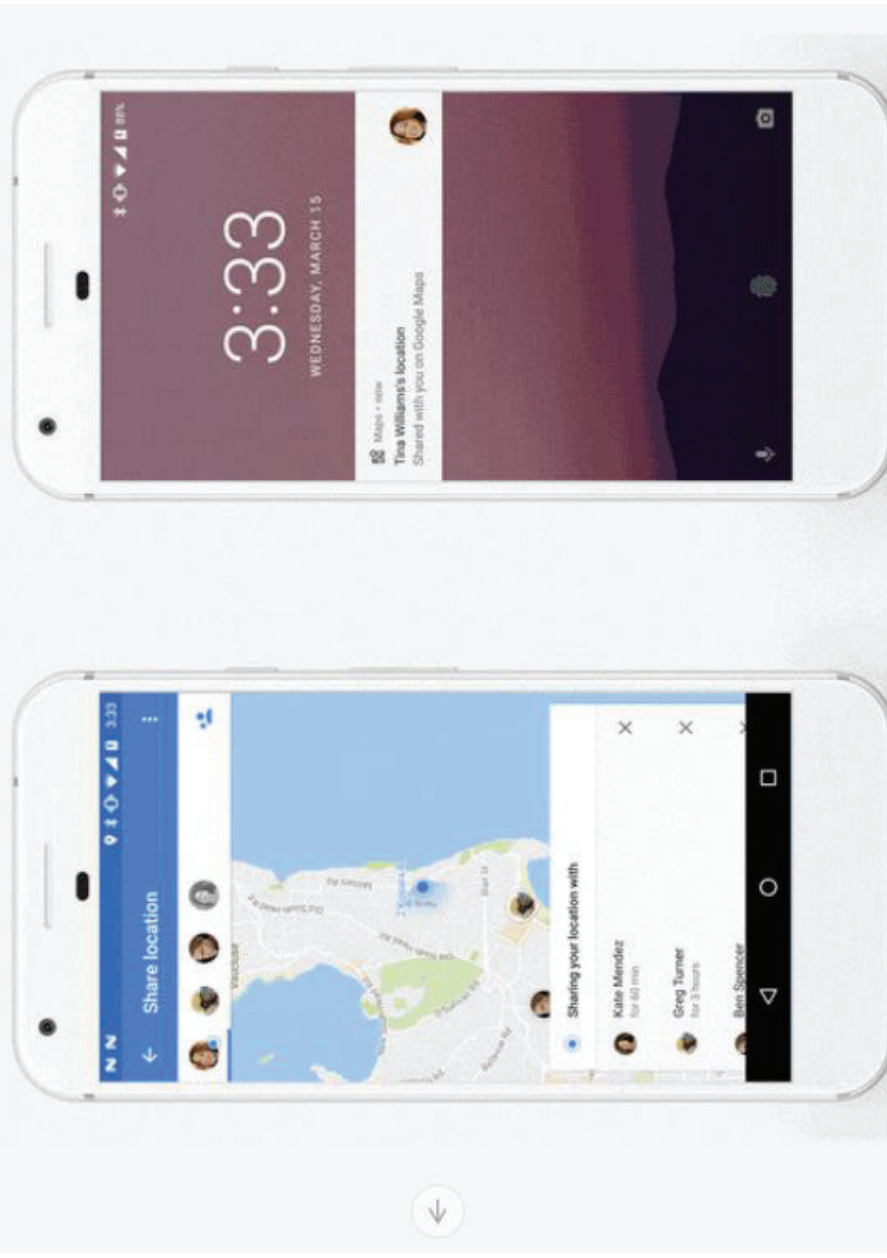


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



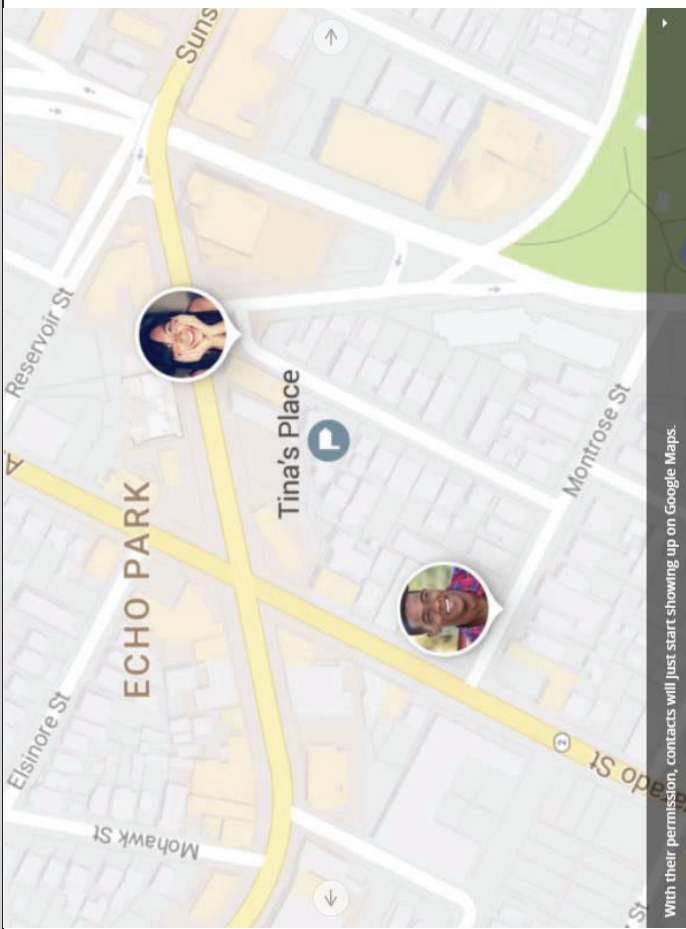
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Stop sharing

1. Open the Google Maps app .
2. Tap the Menu  > **Share location**.
3. Next to the person with whom you want to stop sharing, tap Remove .



Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

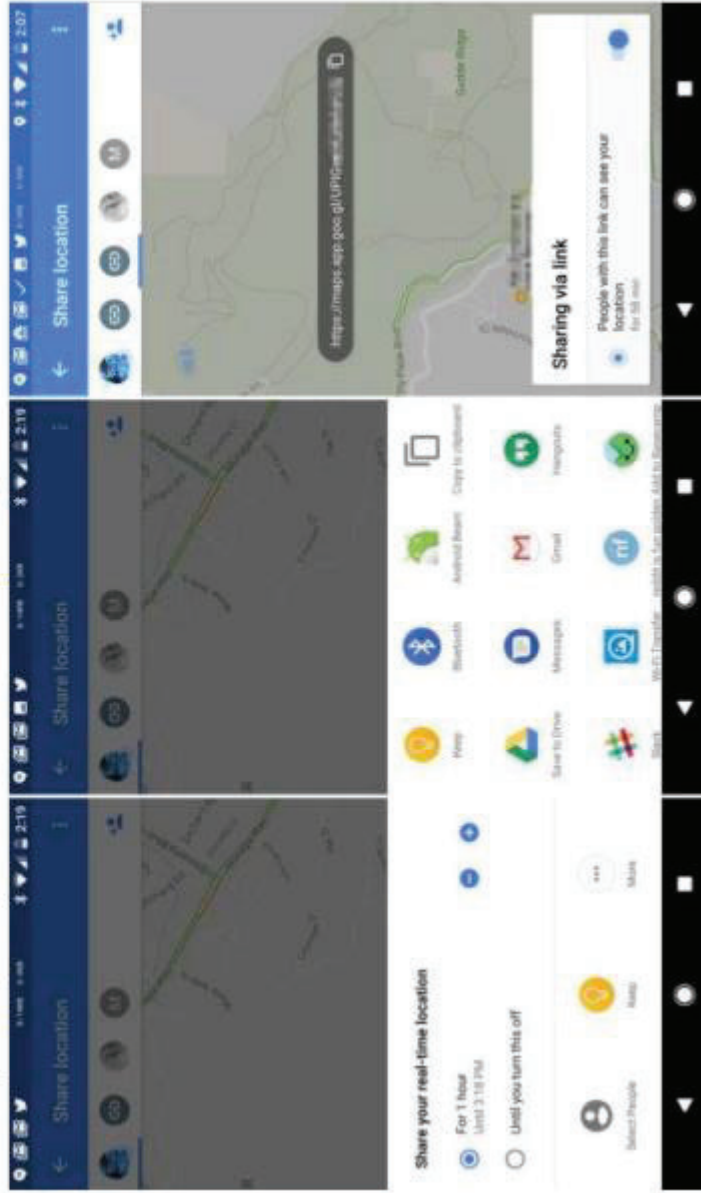
ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
2. Set a driving destination. [Learn how to navigate to a place.](#)
3. After you start navigation, tap **More**  > **Share trip progress**.
4. Choose a person from the list.
5. Tap **Share**.
6. Location Sharing will stop when you reach your destination or stop navigating.

- To stop sharing before you arrive, tap **More**  > **Stop sharing**.

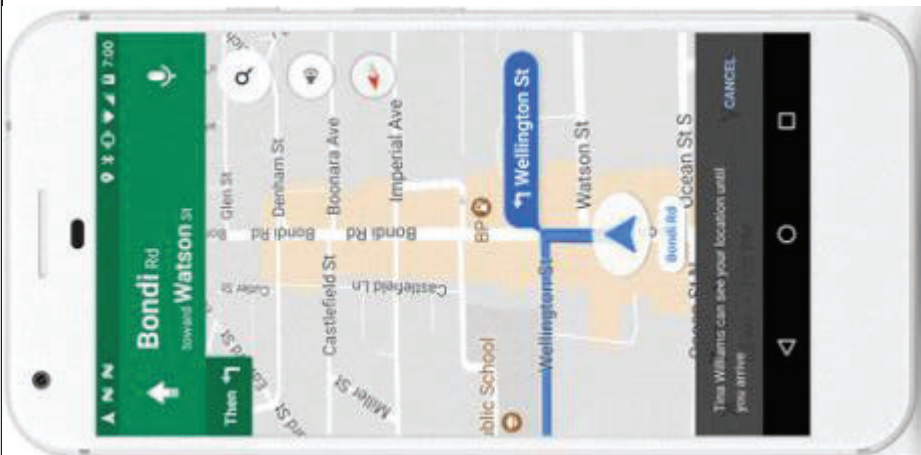


B-516

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

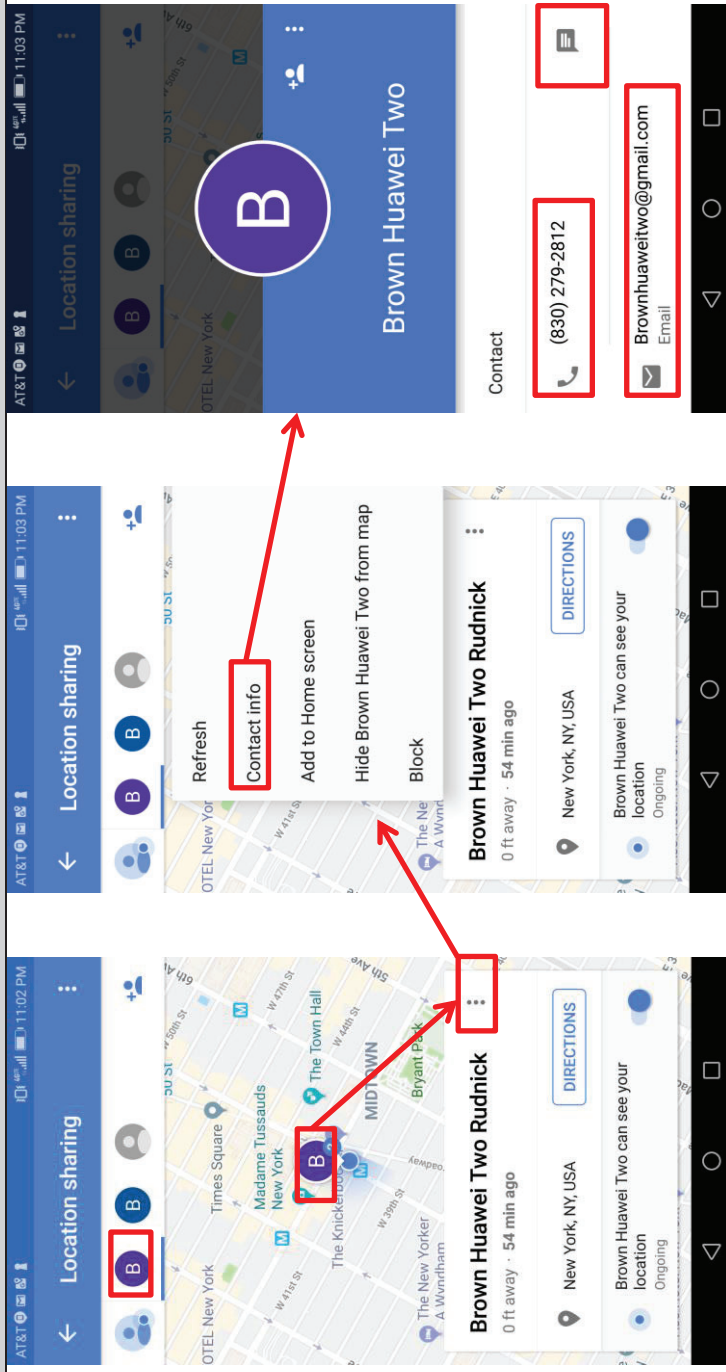


Exemplary Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "")
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-529

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1518 224 1587">ZTE</p> <p data-bbox="232 506 302 1587">https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <div data-bbox="354 359 397 1570"><pre>public static LocationRequest create ()</pre></div> <p data-bbox="423 1071 448 1575">Create a location request with default parameters.</p> <p data-bbox="480 464 540 1575">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="565 1470 589 1554">Returns</p> <ul data-bbox="610 1287 634 1547" style="list-style-type: none">• a new location request <p data-bbox="651 304 678 1587">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-------------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <hr/> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <hr/> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <pre> public Task<Location> getLastLocation () Returns the best most recent location currently available. If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned. This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates. public Task<LocationAvailability> getLocationAvailability () Returns the availability of location data. When isLocationAvailable() returns true, then the location returned by getLastLocation() will be reasonably up to date within the hints specified by the active LocationRequest s. If the client isn't connected to Google Play services and the request times out, null is returned. Note it's always possible for getLastLocation() to return null even when this method returns true (e.g. location settings were disabled between calls). https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient </pre> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|---|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener Public Constructors public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) https://developers.google.com/android/reference/com/google/android/gms/maps/MapView |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public void getMapAsync (OnMapReadyCallback callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p><code>callback</code></p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void onCreate (Bundle savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[28G] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claim 1[G], which is incorporated herein by reference in its entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.

Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.

Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.

Selection with Markers:

<https://developers.google.com/maps/documentation/android-api/marker>

Queries with Geo Tagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app.
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share. If you don't see this, tap More.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Share your E. T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app.
2. Set a driving destination. Learn how to navigate to a place.
3. After you start navigation, tap More.
4. Choose a person from the list.
5. Tap Share.
6. Location Sharing will stop when you reach your destination or stop navigating.

To stop sharing before you arrive, tap More.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en>

Markers (adding location information to the link associated with the database):

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions() .position(PERTH) .draggable(true));</pre> |
| <p>[28H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claim 1[H], which is incorporated herein by reference in its entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claim 1[H], which is incorporated herein by reference in its entirety.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1520 224 1587">ZTE</p> <p data-bbox="302 709 337 1220">ZTE Mobile Location Service System</p> <p data-bbox="375 915 402 1041">2004-01-31</p> <p data-bbox="469 1398 496 1562">I. Introduction</p> <p data-bbox="513 373 711 1562">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="727 373 841 1562">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="857 205 927 1587">http://www.cn.zte.com.cn/endata/magazine/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.

Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.

Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.

Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.

Placing a Marker:

<https://developers.google.com/maps/documentation/android-api/marker>

based on queries with GeoTagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share  if you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E.T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More  > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More  > Stop sharing. | <p>29[A]. The system of operations further</p> |
| <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number.</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| <p>comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;</p> | <p>See claims 2[A] and 28, which are incorporated herein by reference in their entirety.</p> |
| <p>[29B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 2[B] and 28, which are incorporated by reference in their entirety.</p> |
| <p>30. The system of claim 28 wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech. See claims 3 and 28, which are incorporated by reference in their entirety.</p> |
| <p>31[A]. The system of claim 28 wherein: the SMS messages include an Internet Protocol (IP) address of the first device;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein: the SMS messages include an Internet Protocol (IP) address of the first device. See claims 4[A] and 28, which are incorporated by reference in their entirety.</p> |
| <p>[31B] and the IP-based</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| <p>ZS9408055B2</p> | <p>ZTE</p> |
|--|--|
| <p>responses include respective IP addresses of the second devices. 32. The system of claim 28 wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, or a passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time.</p> | <p>performance of [] the IP-based responses include respective IP addresses of the second devices. See claims 4[B] and 28, which are incorporated by reference in their entirety. ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claims 5 and 28, which are incorporated by reference in their entirety.</p> |
| <p>33[A]. The system of claim 28 wherein the operations further</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claims 6[A] and 28, which are</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| <p>comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;</p> | <p>incorporated by reference in their entirety.</p> |
| <p>[33B] and receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 6[B] and 28, which are incorporated by reference in their entirety.</p> |
| <p>34[A]. The system of claim 33 wherein the operations further comprise: receiving user selection of the sub-net;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 7[A], 33 and 28, which are incorporated by reference in their entirety.</p> |
| <p>[34B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 7[B], 33, and 28, which are incorporated by reference in their entirety.</p> |
| <p>35. The system of claim 28, wherein the first device is a</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claims 8 and 28, which are incorporated by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| <p>cellular phone or a personal digital assistant (PDA).</p> <p>36. The system of claim 28, wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the at least one second device. See claims 9 and 28, which are incorporated by reference in their entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the at least one second device. See claims 9 and 28, which are incorporated by reference in their entirety.</p> |
| <p>37. The system of claim 30, wherein the video comprises a video clip or a video transmission.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claims 10, 30, and 28, which are incorporated by reference in their entirety.</p> |
| <p>38[A]. The system of claim 33, wherein the operations further comprise: receiving</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 11[A], 33, and 28, which are incorporated by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|--|
| <p>user selection of the sub-net;</p> <p>[38B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 11[B], 33, and 28, which are incorporated by reference in their entirety.</p> |
| <p>39. The system of claim 28, wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency. See claim 28, which is incorporated herein by reference in its entirety.</p> <p>A user can send an audio message or a video message that includes to announce an emergency to the one or more second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

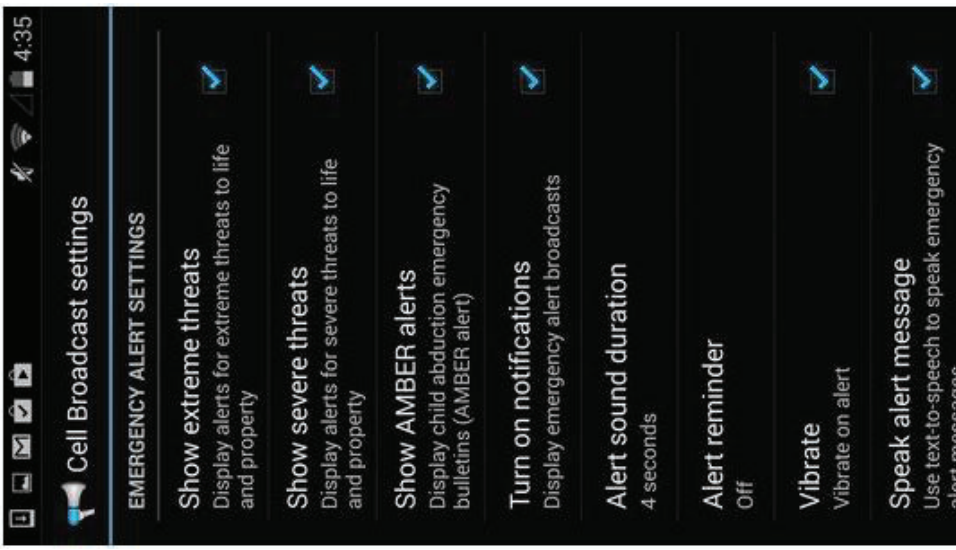
| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| |  <p>Cell Broadcast settings</p> <p>EMERGENCY ALERT SETTINGS</p> <p>Show extreme threats Display alerts for extreme threats to life and property <input checked="" type="checkbox"/></p> <p>Show severe threats Display alerts for severe threats to life and property <input checked="" type="checkbox"/></p> <p>Show AMBER alerts Display child abduction emergency bulletins (AMBER alert) <input checked="" type="checkbox"/></p> <p>Turn on notifications Display emergency alert broadcasts <input checked="" type="checkbox"/></p> <p>Alert sound duration 4 seconds</p> <p>Alert reminder Off</p> <p>Vibrate Vibrate on alert <input checked="" type="checkbox"/></p> <p>Speak alert message Use text-to-speech to speak emergency alert message <input checked="" type="checkbox"/></p> <p>https://www.greenbot.com/article/2689993/how-to-change-settings-for-emergency-alerts-on-android-phones.html</p> |
| <p>40. The system of claim 28, wherein the data sent to the one or more second devices</p> | <p>to the one or more second devices causes at least one of the second devices to place a phone call to the first device. See claims 23[A] and 28, which are incorporated by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2 causes at least one of the second devices to place a phone call to the first device.</p> | <p>ZTE</p> |
| <p>41[P] A non-transitory storage device having instructions stored thereon that, when executed by a first device, cause the first device to perform operations comprising:</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of this this non-transitory storage device having instructions stored thereon that, when executed by a first device, cause the first device to perform operations as set forth below. See claims 1[P] and 28[P], which are incorporated herein by reference in their entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><u>Google Maps Share Location</u></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps' Share Location</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> <p><i>See, e.g.,</i> https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/; https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html; https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html; http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html; https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html</p> <p>Control within reach, even when your device isn't</p> <p>One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching Find My Device as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit android.com/find or check out the app.</p> <p><i>See, e.g.,</i> https://www.blog.google/products/android/google-play-protect/</p> |
|---------------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Find your device using Android Device Manager

If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.

Before you can use Android Device Manager to locate your device: Your device's location access needs to be turned on and you need to be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.

Tip: If you've linked your phone to Google, you can locate or ring it by searching for **find my phone** on [google.com](https://support.google.com/pixelphone/answer/6160491).

<https://support.google.com/pixelphone/answer/6160491>

Link your phone to Google

You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.

Link your Android phone

Step 1: Update the Google app

1. On your phone, go to the Google app page on the Play Store.
2. Tap **Update**.

Step 2: Turn on Google Now

1. On your phone, open the Google app.
2. At the top left, tap Menu > Settings > **Now cards**.
3. Turn on **Show cards**.
4. Turn on **Show notifications**.

Step 3: Turn on Web & App Activity

1. Visit the [Account History](#) page.
2. Make sure the switch is on (green).

Step 4: Sign in to your browser

1. On your phone, open the Google app.
2. At the top left, tap the Menu.
3. At the top left, you'll see the email address you use for the Google app.
4. Visit www.google.com on your computer.
5. If you aren't signed in already, click **Sign in** in the top right corner of the page.
6. Sign in using the Google Account you use for the Google app.

Step 5: Send information to your phone

1. Do one of the searches below, like **note to self**, or send directions to my phone.
2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

What you can do once your phone is linked

Find my phone

You can get the current location of your phone if you can't find it.

1. On your computer's browser, search on www.google.com for **find my phone**.
2. If your phone is turned on and connected to the Internet, you'll see your phone's location.
3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking **Ring**. You can stop the ringing from your phone when you find it.

Tip: You can also find your missing phone using the **Android Device manager** which lets you find your device or remotely ring, lock, or erase it.

Send directions to my phone

Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.

1. On your computer's browser, search on www.google.com for **send directions to my phone**.
2. Enter in your destination.
3. Click **Send directions to your phone**.
4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.

Send a note to my phone

1. On your computer's browser, search on www.google.com for **send a note to my phone**.

2. Type your note in the box.
3. Click **Send note to your phone**.
4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.

Set an alarm

1. On your computer's browser, search on www.google.com for **set an alarm**.

2. Choose the time you want the alarm to go off.
3. Click **Set an alarm on your phone**.
4. An alarm will now be set on your phone's Clock app.

Set a reminder

1. On your computer's browser, search on www.google.com for **set an reminder**.

2. Type what you want to be reminded about, and either when or where you want the reminder to go off.
3. Click **Remind me on my devices**.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your location using Google Maps

You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.

<https://support.google.com/plus/answer/3302509?hl=en&co=GENIE.Platform%3DAndroid&oco=1>

Location

Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.

- **Mode** – Sets the how your current location information is determined.
- **Recent Location Request** – Displays applications and services that have recently requested your location information.
- **Camera** – Checkmark to tag photos or videos with their locations.
- **Google Location History** – Allows you to view and manage your Google location history.

Accounts & sync

Use the Accounts & sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.

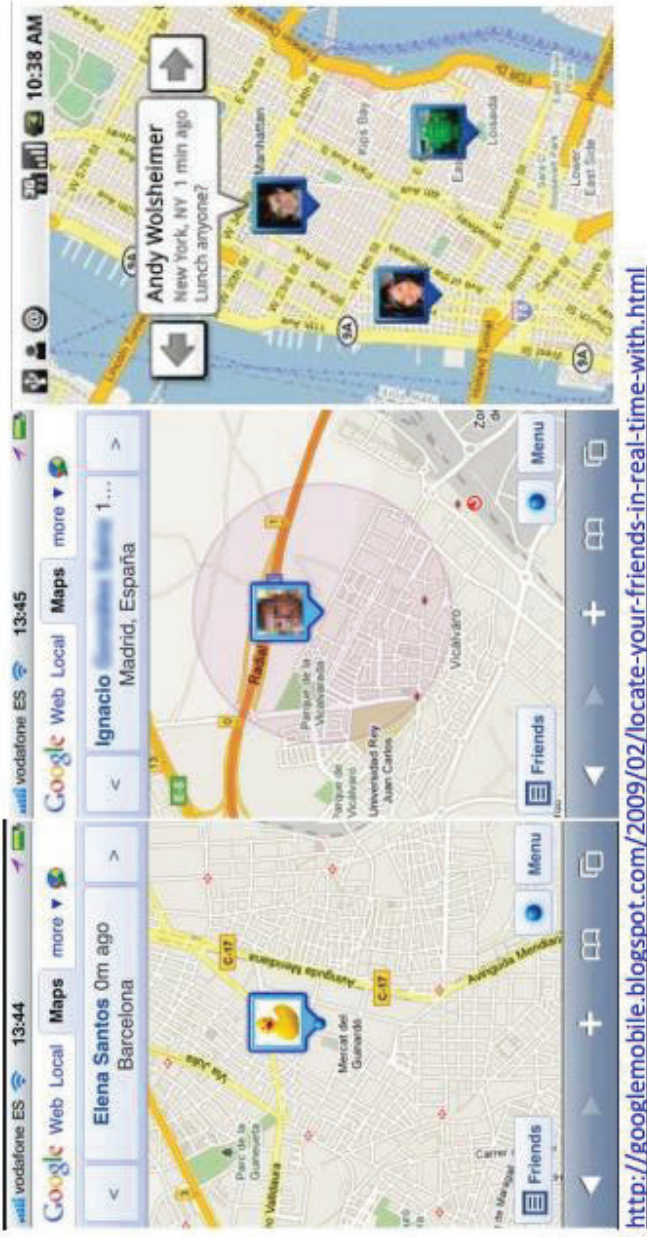
Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch **Add account** to add new account.

... ..

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.

ZTE makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as ZTE phones and ZTE tablets, as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example ZTE Android devices that perform each step of this method as set forth below.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE phones
 mob.org * Mobile phones and smartphones catalogue

Sort by: Popularity Date Price

HTC
 LG
 Samsung
 Motorola
 Fly
 Sony-Ericsson
 Apple
 Nokia
 Wobisado
 Vertu
 BenQ-Siemens
 Sagem
 Alcatel
 Philips

All brands

ZTE Rapido
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 4.5 inch.
 Android 4.1.2

ZTE Grand S3
 Mobile phone
 2015 year
 Touchscreen: 1080 x 1920
 5.5 inch.
 Android 4.4

ZTE Grand X Quad
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.1

ZTE ZMAX
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.4.2

ZTE Geek 2
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE V5s
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE Grand Memo Lite
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.3

https://mob.org/phone/zte/page_3/sort_date_down/

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p>  <p>https://www.zteusa.com/products/tablets</p> |
| <p>41[A] obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claims 1[A] and 28[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Phone calls

How to make calls



There are many ways to make a call with your phone, and they're all easy to do.

Calling from the dialer



1. From the home screen, tap .
2. Enter the phone number with the on-screen keypad. Tap  to delete wrong digits.
3. Tap  to place the call.

Tip: To make international calls, press and hold **0+** to enter the "+".

Calling from your contacts

1. From the home screen, tap .
2. Swipe your finger up or down to scroll through the contacts list and tap  next to the contact you want to call.

Tips:

- You can search for a contact by tapping  and entering the contact name.
- You can also access your contacts by tapping  > **Favorites**.



People

You can add contacts on your phone and synchronize them with the contacts in your Google account or other accounts that support contact syncing.

To see your contacts, tap  on the home screen. From there, you can tap the tabs on the top to quickly switch to **Groups**, or **Favorites**.

Importing and exporting contacts

You can import/export contacts from/to your SIM card, phone storage, or microSDHC card. This is especially useful when you need to transfer contacts between different devices. You can also quickly share your contacts using Bluetooth, Email, Messaging, etc.

Importing contacts from the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Manage SIM card contacts**.
2. If you have added contact accounts other than the phone, select an account in which to save the contacts.
3. Tap the contacts you want to import one by one, or tap  > **Import all**.

Importing contacts from a microSDHC card or phone storage


1. From the Contacts screen, tap  > **Import/export** > **Import from phone storage**.
2. If you have added contact accounts other than the phone, select an

Calling from your call history

1. From the home screen, tap  > **Call log**.
2. Tap  next to the number you want to call.

Calling from a text message

If a text message contains a phone number that you want to call, you can make the call while viewing the text message.

1. From the home screen, tap .
2. Tap the conversation and then find the message that contains the phone number you need.
3. Tap the number and then tap .

Using speed dial

Press and hold the **1-9** key from the dialer to call the corresponding speed dial number.

The **1** key is reserved for your voicemail.

Assigning a speed dial key

1. From the home screen, tap .
2. In the Phone tab, tap  > **Speed dial setting**.
3. Tap a speed dial key and tap **Set speed dial contact**.
4. Select a contact from the contact list.

Note: You can also set your speed dial keys from the dialer. Press and hold the **2-9** key and then tap **OK**.


3. Select the vCard file(s) in the microSDHC card or the phone storage and tap **OK**.

Note: If the microSDHC card is not installed in the phone, you can import vCard file(s) in the phone storage.

Exporting contacts to the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Export to SIM card**.
2. Select the contacts you want to export and then tap .

Exporting contacts to the microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Export to phone storage**.
2. The phone will prompt you with the name of the vCard file and the directory in which the file will be saved. Tap **OK** to create the file.

Note: If the microSDHC card is not installed in the phone, you can export vCard file(s) into the phone storage.

Sharing contact information

1. From the Contacts screen, tap  > **Import/export** > **Share visible contacts**.
2. Choose how to share the contacts. Options depend on the applications and services installed.

Creating a contact

1. From the Contacts screen, tap  to add a new contact.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

2. Tap the account field near the top of the screen to choose where to save the contact. If a sync account is selected, the contacts will be synced automatically with your account online.
3. Enter the contact name, phone numbers, email addresses, and other information.
4. Tap **DONE** to save the contact.

Adding a contact to Favorites

You can add the contacts you use frequently to Favorites so that you can find them quickly.

1. From the Contacts screen, tap the contact you want to add to **Favorites**.
2. Tap  next to the contact's name.

Searching for a contact

1. Tap the search field above the contacts list.
2. Enter the contact name you want to search for. Matching contacts will be listed.

Joining contacts


As your phone synchronizes with multiple online accounts, you may see duplicate entries for the same contact. You can merge all the separate information of a contact into one entry in the Contacts list.

1. From the home screen, tap .
2. Tap a contact to display the contact's details.
3. Tap  > **Edit** >  > **Join**.

Entering text

You can enter text using the onscreen keyboard. Some apps open it automatically. In others, you open it by tapping where you want to type. You can also enter text by speaking with the Google voice typing feature. Tap  to hide the onscreen keyboard.

Changing input methods

1. When you use the onscreen keyboard to enter text, the icon  appears on the notification bar.
2. Open the notification panel and tap **Choose input method**.
3. Select an input method you need.

Google keyboard

The Google keyboard provides a layout similar to a desktop computer keyboard. Turn the phone sideways and the keyboard will change from portrait to landscape.

To use the landscape keyboard, tap the **Auto-rotate screen** check box in  > **System settings** > **Accessibility**.

Note: The landscape keyboard is not supported in all applications.





4. Tap the contact whose information you want to join with the first entry.
 5. Tap **DONE**.
- The information from the second contact is added to the first, and the second contact is no longer displayed in the contacts list. You can repeat these steps to join another contact to the main contact.

Separating contact information

If contact information from different sources was joined in error, you can separate the information back into individual contacts on your phone.













1. From the home screen, tap .
2. Tap a contact you have merged and want to separate.
3. Tap  > **Edit** >  > **Separate**.
4. Tap **OK** to confirm.

Creating a new group

1. From the Contacts screen, tap .
2. Tap .
3. If you have added contact accounts other than the phone, choose an account for the new group.
4. Enter the group name and tap **DONE**.
5. Tap  and select the contacts you wish to be the group members.
6. Tap .

To send messages to the group members, you can tap a group and then tap  > **Send group message**.



- Tap the alphabetic keys to enter letters. Press and hold the keys to enter associated accented letters or numbers. For example, to enter É, press and hold  and the available accented letters and number 3 appear. Then slide your finger to choose **E**.
- Tap  to use uppercase or lowercase letters. This key also changes to indicate the current case you are using:  for lowercase,  for uppercase, and  when locked in uppercase. Press and hold or double-tap  to lock the keyboard in uppercase.
- Tap  to delete any text you have entered.
- Tap  to select numbers and symbols. You can then tap  to access more.
- Tap  to enter miniature icons.
- Tap  to use Google's networked voice input.
- Press and hold  to change the input language or the Google keyboard settings.

In other examples, the Accused products run Android Messages and Google Hangouts which both access contact information for second users using respective second devices.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | ZTE <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider-provider.html</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contacts_id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/provider-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|--|-------------|---|-----------|---|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

B-567

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-570

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 }</pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,            // 1
57             Data.LOOKUP_KEY,           // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,     // 4
60             Data.CONTACT_STATUS,       // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY  = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-575

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




US9408055B2

ZTE

Send & receive text messages in Android Messages





You can send and receive text messages with friends and contacts on Android Messages.

Start a conversation

1. Open the Android Messages app .
2. Tap Compose .
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next .

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329

See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](#) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
2. Tap Menu  > **Create label**.
3. Enter a label name and tap **OK**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
- **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER ANDROID IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

Contact someone

You can call, email, or send text messages to your contacts.

1. Open your device's Contacts app.
2. Tap a contact in the list.
3. Choose an option:

- Call
- Email
- New message

https://support.google.com/nexus/answer/6118731?hl=en&ref_topic=6118711

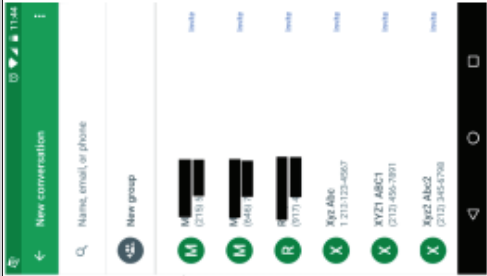


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a conversation

1. Open the Android Messages app
2. Tap Compose
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329
<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

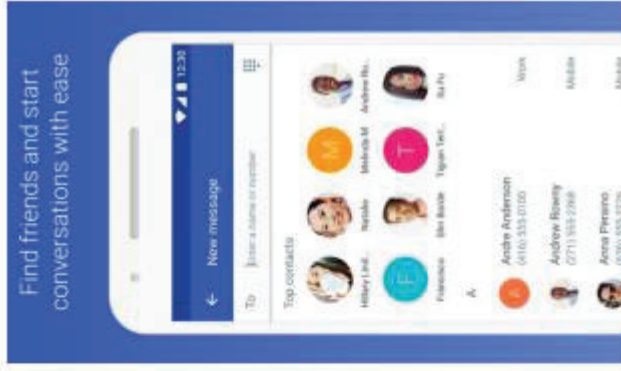


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add + > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

1. Open the Hangouts app.
2. At the bottom, tap Add + > New conversation > New group.
3. Enter and select the names, phone numbers, or email addresses of people in your group.
4. Tap Done.

https://support.google.com/hangouts/answer/3111943?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products





| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <h3 data-bbox="235 1186 276 1554">Contact someone</h3> <p data-bbox="300 903 332 1554">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="357 630 625 1554" style="list-style-type: none"><li data-bbox="357 1113 397 1554">1. Open your device's Contacts app .<li data-bbox="406 1260 446 1554">2. Tap a contact in the list.<li data-bbox="454 1323 625 1554">3. Choose an option:<ul data-bbox="495 1291 625 1522" style="list-style-type: none"><li data-bbox="495 1407 535 1522">• Call <li data-bbox="544 1386 584 1522">• Email <li data-bbox="592 1291 625 1522">• New message  <p data-bbox="641 630 673 1585">https://support.google.com/hexus/answer/611873?hl=en&ref_topic=6118711</p> <p data-bbox="706 1071 738 1585"><u>Exemplary Google Maps Screenshots:</u></p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

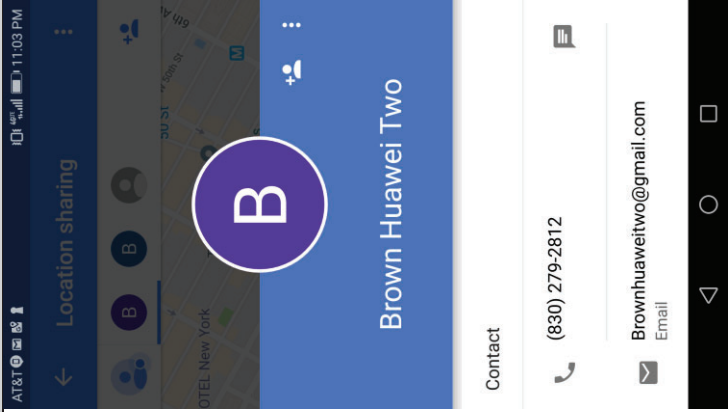
| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p>  <p>The screenshot shows an Android contact card for 'Brown Huawei Two'. At the top, there is a 'Location sharing' section with a back arrow, a plus sign, and a 'ZTEL New York' location. Below this is a large purple circle with a white letter 'B'. Underneath the circle, the name 'Brown Huawei Two' is displayed. At the bottom of the card, there is a 'Contact' section with a phone icon, the number '(830) 279-2812', an email icon, and the email address 'Brownhuaweitwo@gmail.com'. A navigation bar is visible at the very bottom of the screen.</p> |
| <p>[41B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claims 1[B] and 28[B], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, from the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claims 1[B] and 28[B], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products



| | |
|--|--|
| <p>US9408055B2</p> <p>the first device to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p> | <p>ZTE</p> <p>SMS message, with additional information, to a contact.</p> <p>Using Wi-Fi Direct</p> <p>Wi-Fi Direct allows Wi-Fi devices to connect to each other without the need for wireless access points (hotspots).</p> <p>Connecting to another device via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > System settings > Wi-Fi. 2. If Wi-Fi is off, slide the Wi-Fi switch to the On position. 3. Tap  > Wi-Fi Direct. Your phone will search for other devices enabled with Wi-Fi Direct connections. 4. Tap a device name under Peer Devices to connect with it. The other device will receive a Wi-Fi Direct connection prompt and need to accept the request for connection. Both devices may need to enter a common PIN. If prompted, tap Connect. 5. Once connected, the device is displayed as "Connected." <p>Sending data via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. Open the appropriate application and select the file or item you want to share. 2. Select the option for sharing via Wi-Fi Direct. The method may vary by application and data type. 3. Tap a device the phone has connected with or wait for it to search for new devices and tap one of them. <p>Receiving data via Wi-Fi Direct</p> <p>When an attempt to transfer data via Wi-Fi Direct is received, you can see a notification in the status bar. Tap Accept to start receiving the data. Received files are stored automatically in a dedicated folder (<i>WiFiShare</i>, for instance) in the phone storage or microSDHC directory. You can access them with the File Manager app.</p> |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Messaging

You can use Messaging to exchange text messages (SMS) and multimedia messages (MMS).

Message box



Instead of an inbox and outbox, your phone organizes all messages you sent and received into one box, where messages exchanged with the same number are grouped into one message thread on the Messaging screen. You can tap a thread to see the conversation you have had with someone.

Message threads are sorted in chronological order with the latest one on top.

Opening the messaging screen

From the home screen, tap .

The Messaging screen opens, where you can create a new message, search for messages, or open an ongoing message thread.

- Tap  to write a new text or multimedia message.
- Tap  to search for a message with keywords.
- Tap an existing message thread to open the conversation you've had with a certain number.

Sending a message





1. From the messaging screen, tap  at the bottom.
2. Add recipients by one of the following ways:
 - Tap the **To** field and manually enter the recipient's number or the contact name. If the phone presents a few suggestions, tap the one you want to add.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



- Select recipients from your contacts by tapping .
- 3. Tap the **Type message** field and enter the content of your text message.
- 4. If you want to send a multimedia message, tap the paper clip icon  to attach a file or a slideshow to the message.
- 5. Tap  to send your message.

Notes:

- You can also include email addresses as recipients for multimedia messages.
- Do not add any attachment if you want to send a text message. Otherwise you may be charged for a multimedia message.


Replying to a message

Messages you receive are appended to existing threads of the same number. If the new message comes from a new number, a new thread is created.


1. From the Messaging screen, tap the thread that has the message you want to reply to.
2. Type your reply in the text box at the bottom. You can tap the icon  if you want to reply with an MMS.
3. Tap  to send your message.

Forwarding a message

1. From the Messaging screen, tap the thread that has the message you want to forward.
2. Press and hold the message.
3. Tap **Forward** in the menu that opens.

4. Enter a recipient for the message and edit the content if you want.
5. Tap  to send your message.

Changing message settings

The phone's message settings are pre-configured for you to use immediately. To change them, tap  > **Settings** from the Messaging screen.

Storage settings:

- **Delete old messages:** Delete old messages as limits are reached.
- **Text message limit:** Set the maximum number of text messages allowed in a single thread.
- **Multimedia message limit:** Set the maximum number of multimedia messages allowed in a single thread.

Text (SMS) message settings:

- **Manage SIM card messages:** Manage the messages stored on your SIM card.

- **Service Center:** Enables you to view and edit the service center number.

Multimedia (MMS) message settings:

- **Auto-retrieve:** Automatically download multimedia messages.

Display settings:

- **Bubble and background:** Set the appearance of the messaging bubbles and background.

Notification settings:

- **Notifications:** Show message notifications in the status bar.
- **Choose ringtone:** Choose a ringtone for your incoming messages.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Email

From the home screen, tap . You can receive and send emails from your webmail or other accounts using POP3 or IMAP, or access your Exchange ActiveSync account for your corporate email needs.

Setting up the first email account

1. When you open **Email** for the first time, enter your email address and password.
2. Tap **Next** to let the phone retrieve the network parameters automatically.
Note: You can also enter these details manually by tapping **Manual setup** or when automatic setup fails.
3. Follow the on-screen instructions to finish the setup.

Your phone will show the inbox of the email account and start to download email messages.

Checking your emails

Your phone can automatically check for new emails at the interval you set when setting up the account.

You can also check new emails manually by tapping  in any of the email account's boxes. Tap **Load more messages** at the bottom of the email list to download earlier messages.

Responding to an email

You can reply to or forward a message that you receive. You can also delete messages and manage them in other ways.

Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.

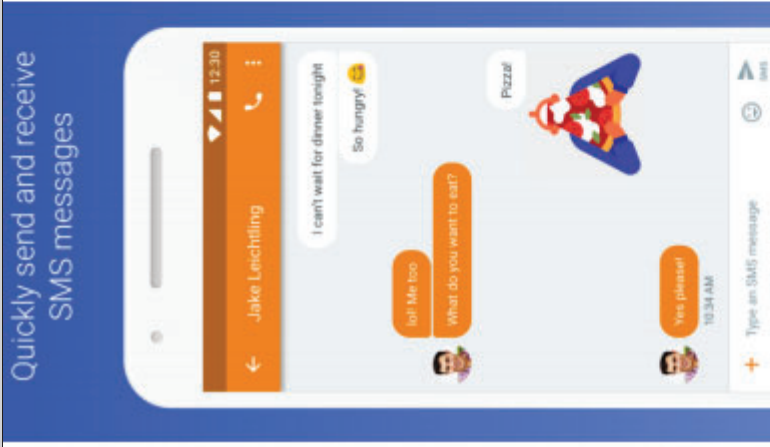
• **Enhanced features:** On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.

<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER **ANDROID** IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap **Add** > **New Conversation**.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap **Send**.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

1. Open the Hangouts app.
2. At the bottom right, tap **Add**.
3. Choose **New SMS**.
4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.
5. Tap the number or contact.
6. Tap **Continue**.
7. Type your message and tap **Send**.

<https://support.google.com/hangouts/answer/3441321?hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><u>Google Maps Share Location</u></p> <p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group</p> <p><u>Exemplary Support for Google Maps:</u></p> |
|---------------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. <p>Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap  > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > **Add People** .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > **Add People** .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap **More**  **> Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap **More**  **> Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap **Menu**  **> Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon **> More**  **> Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap **More** .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD

Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%63DAn droid&oco=1

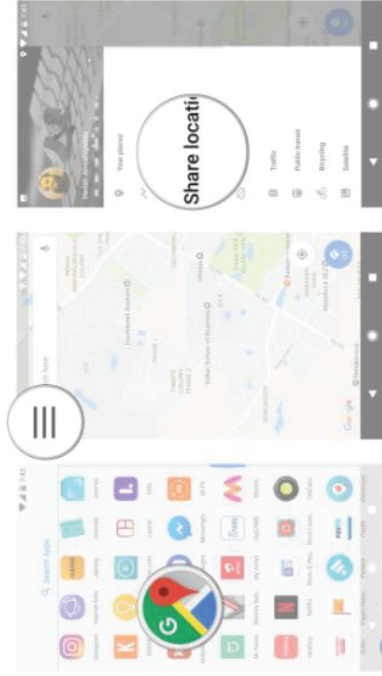
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

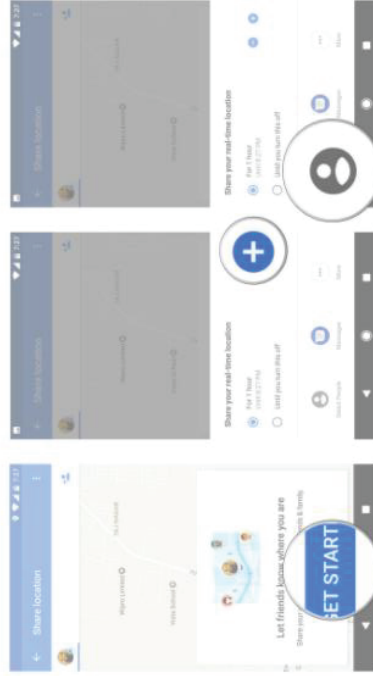
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



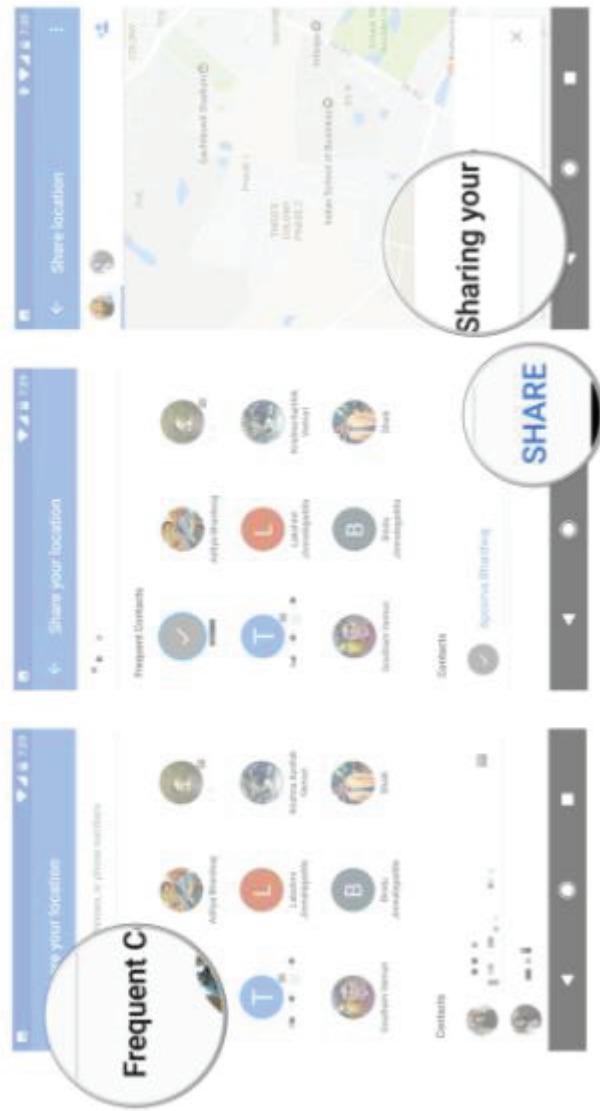
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

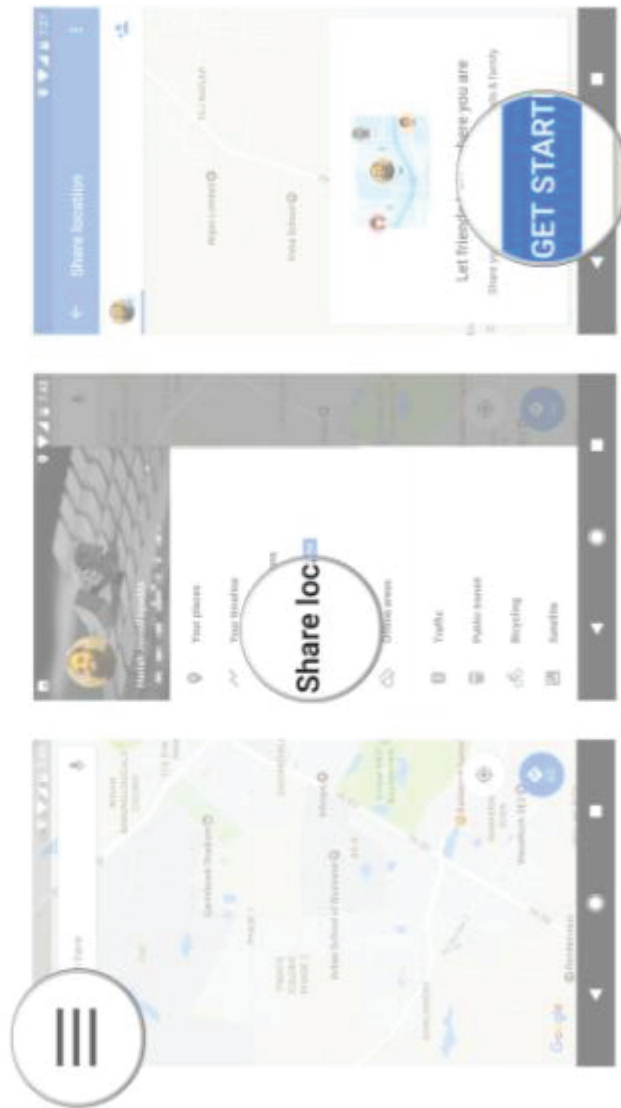
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



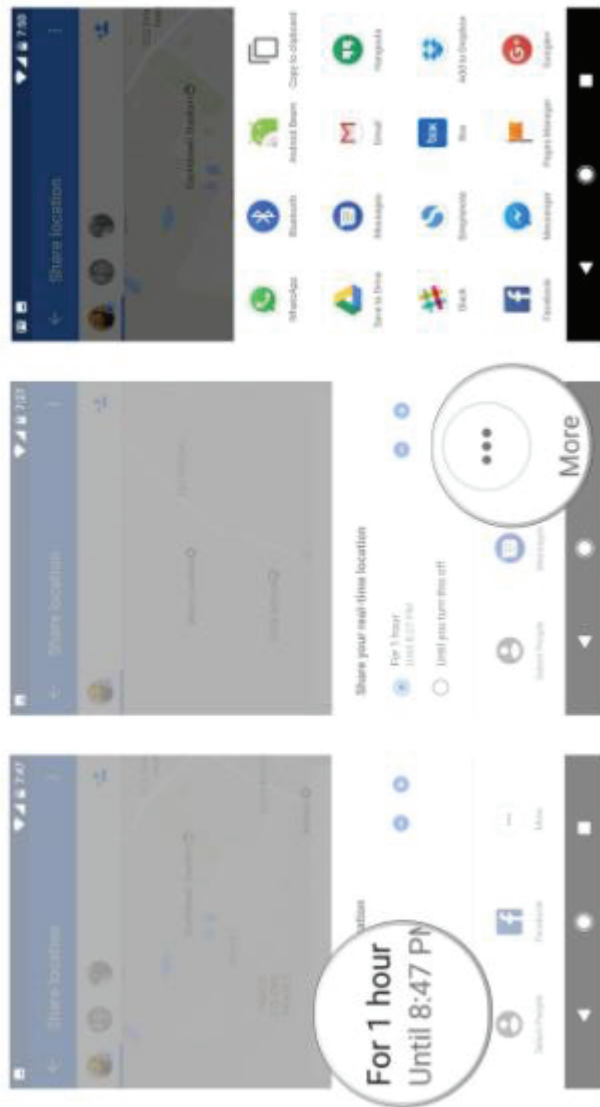
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

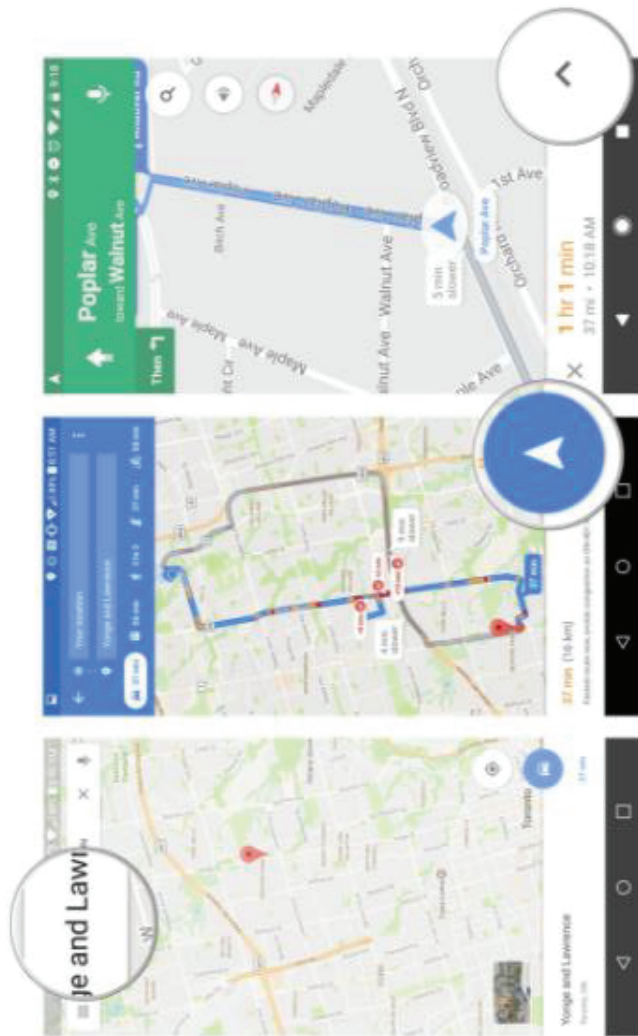
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



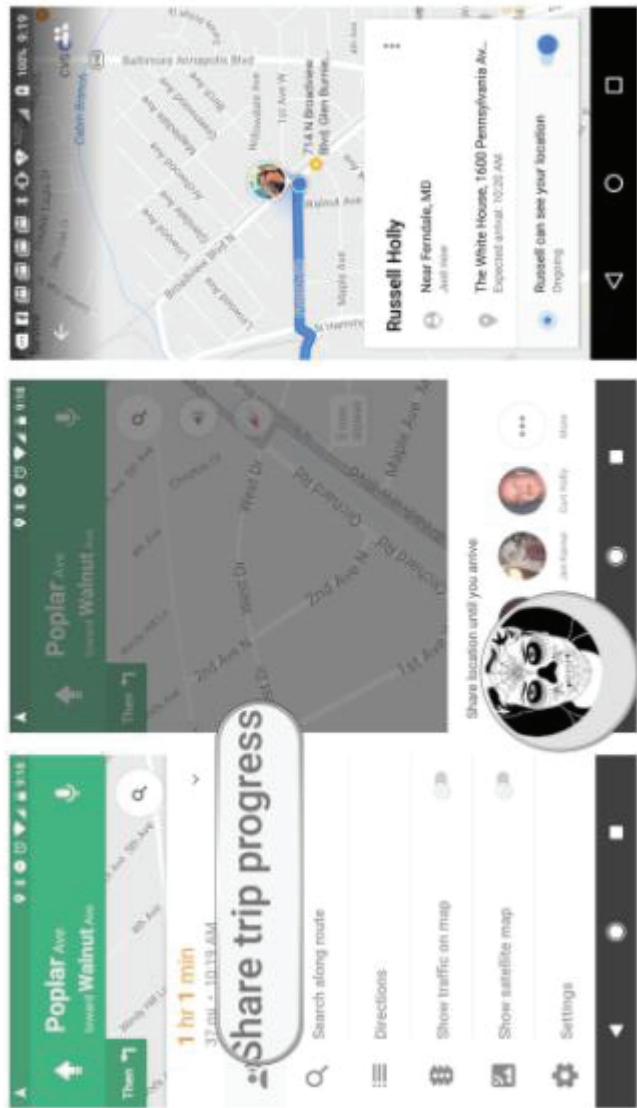
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



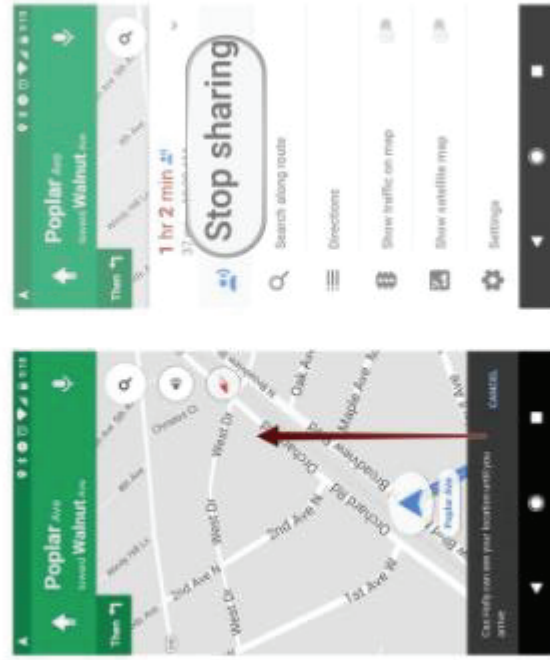
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>





As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

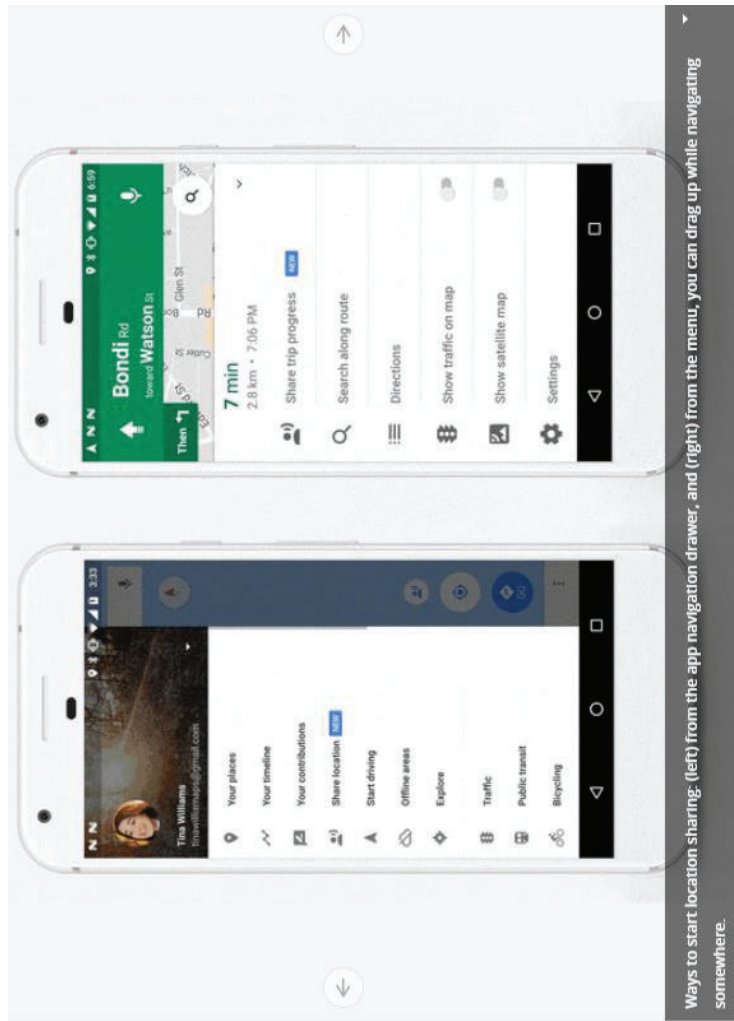
1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

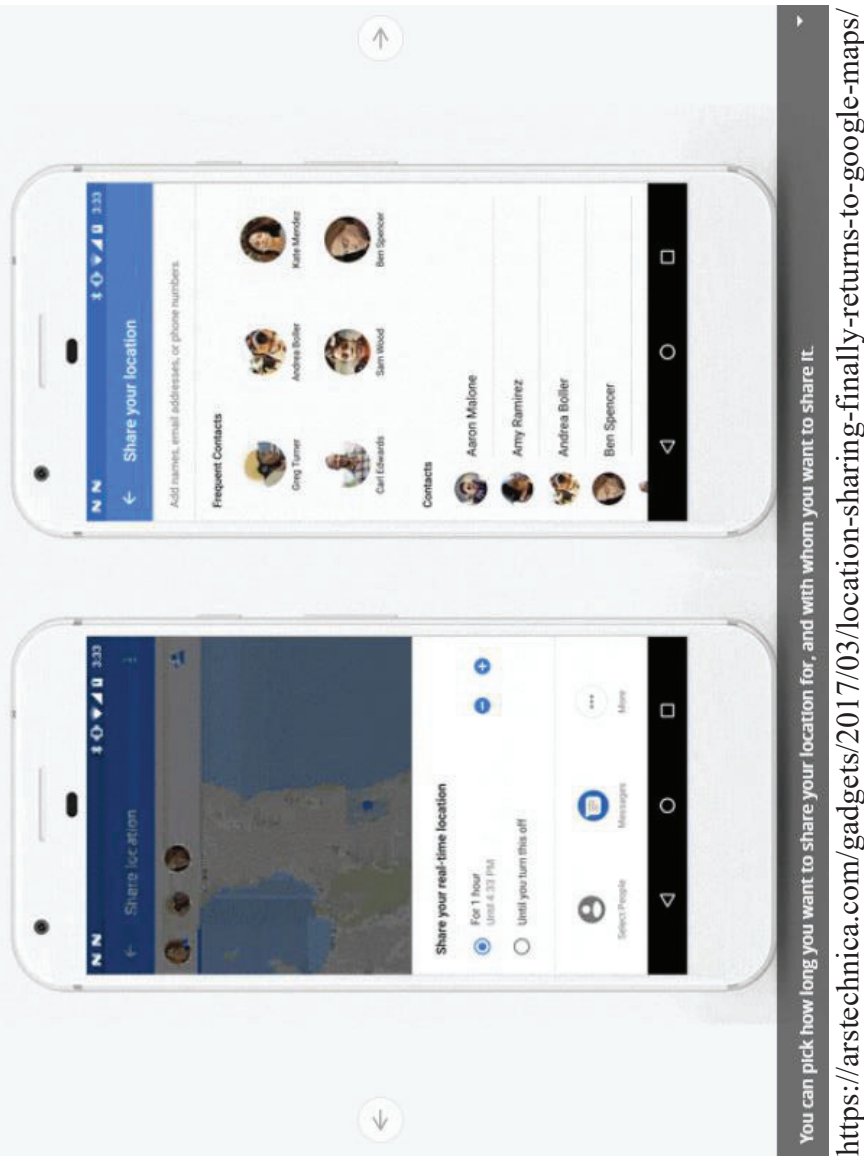
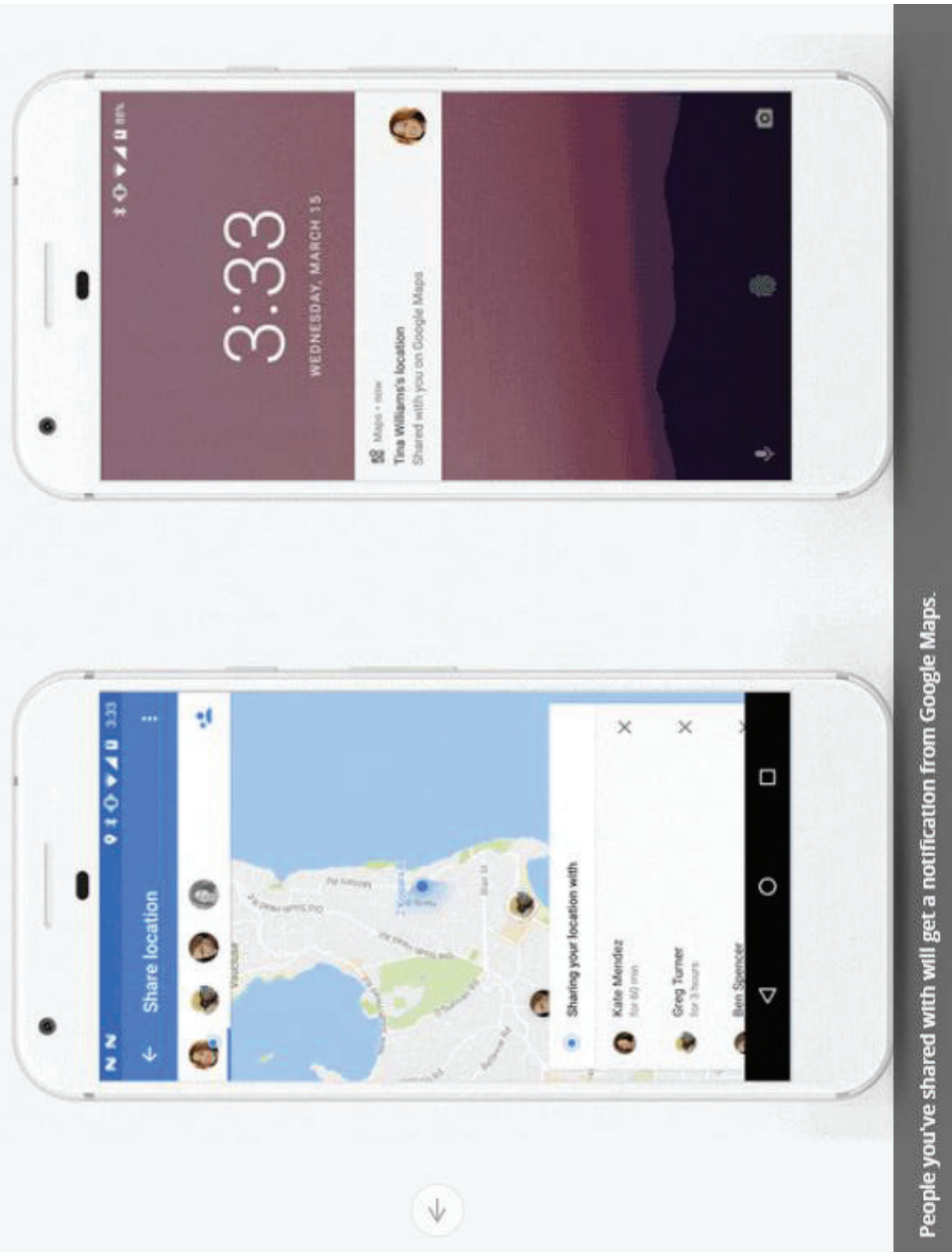


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

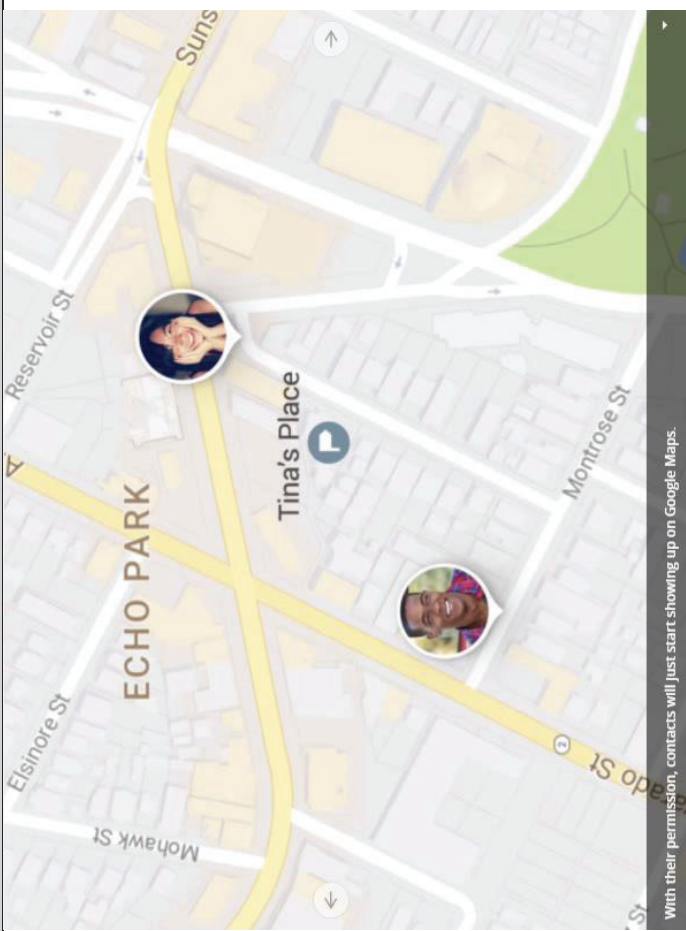


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



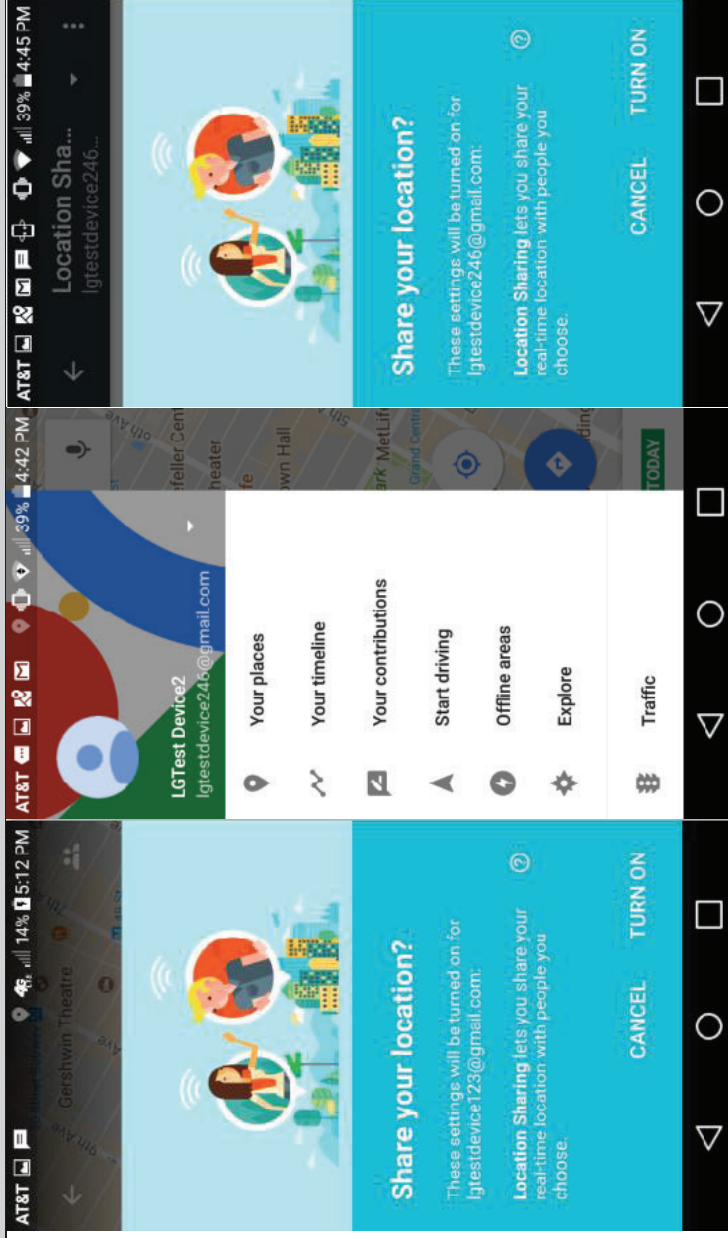
With their permission, contacts will just start showing up on Google Maps.
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <h2 data-bbox="235 1081 292 1575">Contacts Provider</h2> <p data-bbox="324 630 592 1575">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="625 1260 649 1575">This guide describes the following:</p> <ul data-bbox="673 724 852 1575" style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p data-bbox="860 609 885 1585">https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contact's _id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | | |
|---|-------------|--|-----------|---|--|
| Task | Action | Data | MIME type | Notes | |
| Pick a contact from a list | ACTION_PICK | One of: <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. | |
| https://developer.android.com/guide/topics/providers/contacts-provider.html | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 458 private void configureFragments(boolean fromRequest) { 459 if (fromRequest) { 460 ContactListFilter filter = null; 461 int actionCode = mRequest.getActionCode(); 462 boolean searchMode = mRequest.isSearchMode(); 463 final int tabToOpen; 464 switch (actionCode) { 465 case ContactsRequest.ACTION_ALL_CONTACTS: 466 filter = ContactListFilter.createFilterWithType(467 ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468 tabToOpen = TabState.ALL; 469 break; 470 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471 filter = ContactListFilter.createFilterWithType(472 ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473 tabToOpen = TabState.ALL; 474 break; 475 case ContactsRequest.ACTION_FREQUENT: 476 case ContactsRequest.ACTION_STREQUENT: 477 case ContactsRequest.ACTION_STARRED: 478 tabToOpen = TabState.FAVORITES; 479 break; 480 case ContactsRequest.ACTION_VIEW_CONTACT: 481 tabToOpen = TabState.ALL; 482 break; 483 default: 484 tabToOpen = -1; 485 break; 486 } 487 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID       = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI    = 3;
50         public static final int CONTACT_LOOKUP_KEY   = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,            // 1
57             Data.LOOKUP_KEY,           // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,     // 4
60             Data.CONTACT_STATUS,       // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI    = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS       = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

B-620

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/-nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-622

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

B-624

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

B-627

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : ""))
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpParams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> <p>[41C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices;</p> | <p>ZTE</p> <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claims 1[C] and 28[C], which are incorporated herein by reference in their entirety.</p> <p>For example, the ZTE accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/findustr_enterprise/m2m/201112/20111208_352166.html</p> |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE Mobile Location Service System

2004-01-31




I. Introduction

Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.

The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.

http://www.en.zte.com.cn/endata/magazine/zte technologies/2003 year/no 14/articles/200401/t20040131_161273.html

Send your location

1. Open the Android Messages app .
2. Open or start a conversation.
3. Tap Attach .
4. Tap Location on .
5. To send your location, tap Send .

https://support.google.com/pixelphone/answer/6159880?hl=en&ref_topic=6211804

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

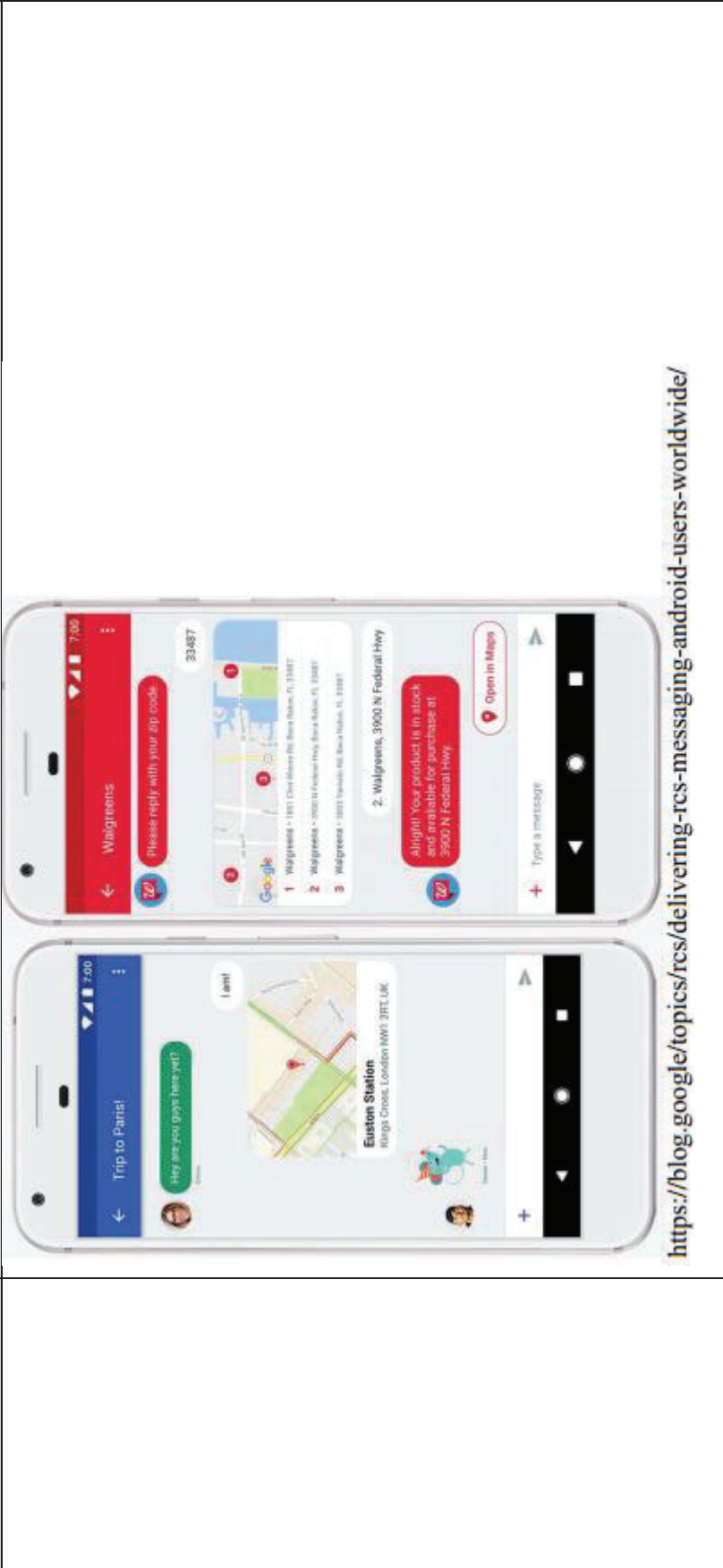


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share a location or place

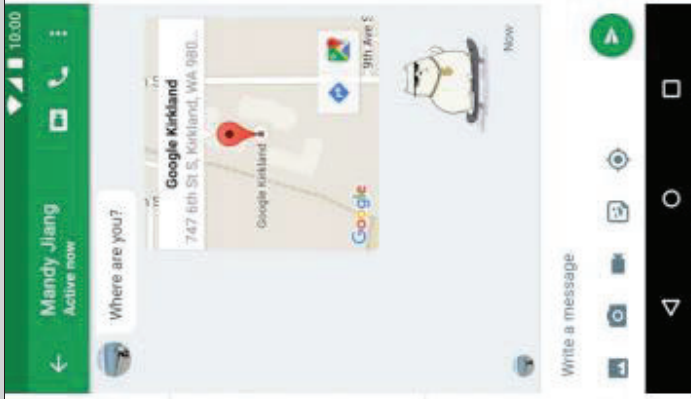
Share your location

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location.
- 4. Tap Select this location > Select.

Share a place

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location > Search Q.
- 4. Type in a location or address.
- 5. Tap Select.

https://support.google.com/hangouts/answer/3115410?visit_id=l-636271867303650973-2491837168&rd=1&co=GENIE.Platform%3DAndroid&oco=1
<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>



Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.

The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="196 1520 224 1587">ZTE</p> <p data-bbox="269 205 412 1587">By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p data-bbox="453 197 813 1587"><u>Further regarding Google Maps</u>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p data-bbox="818 1073 846 1587"><u>Exemplary Support for Google Maps:</u></p> |
|--------------------|---|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD


Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1

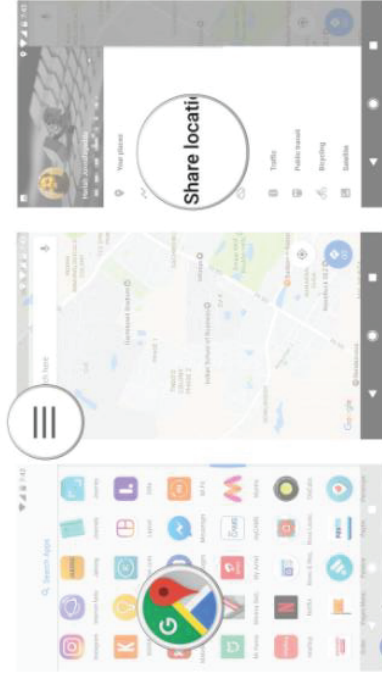
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

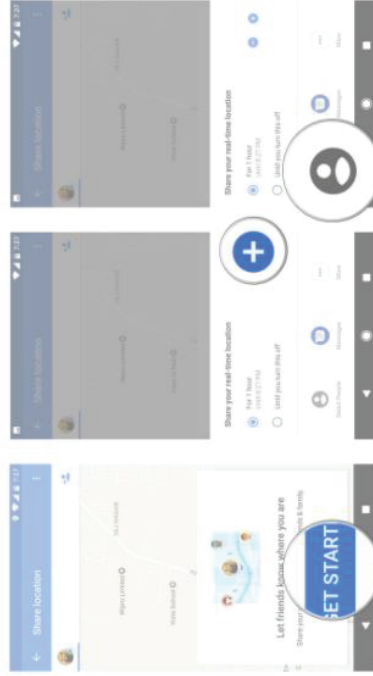
ZTE

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



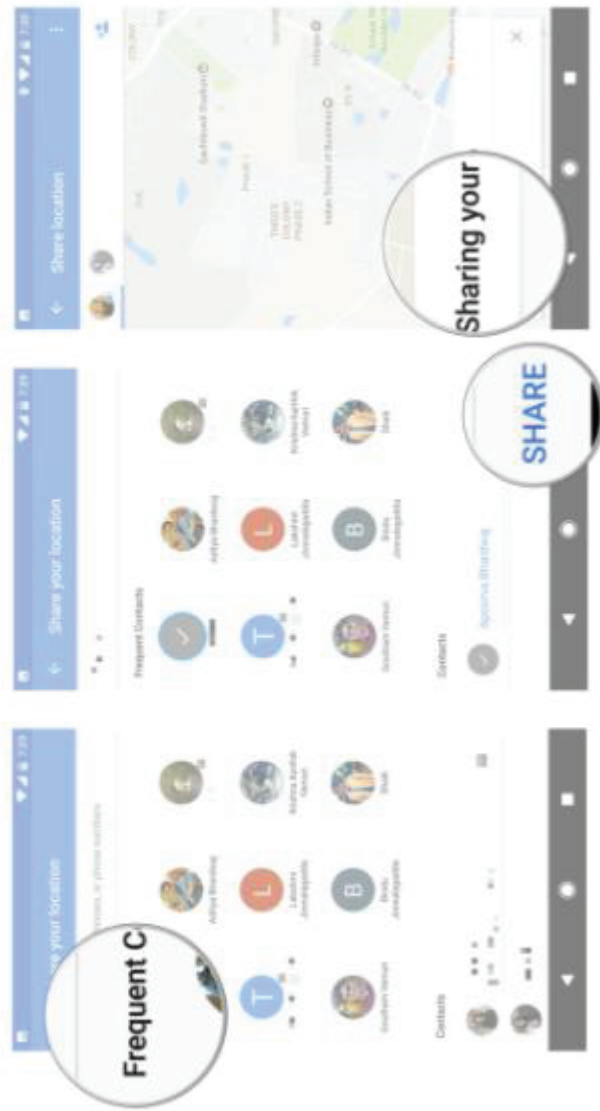
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7.** You'll see a list of your frequent contacts at the top, along with a full list of contacts. **Pick the contacts by tapping their name.**
- 8.** Once you've selected the contacts you want to share your location to, tap **Share**.
- 9.** You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

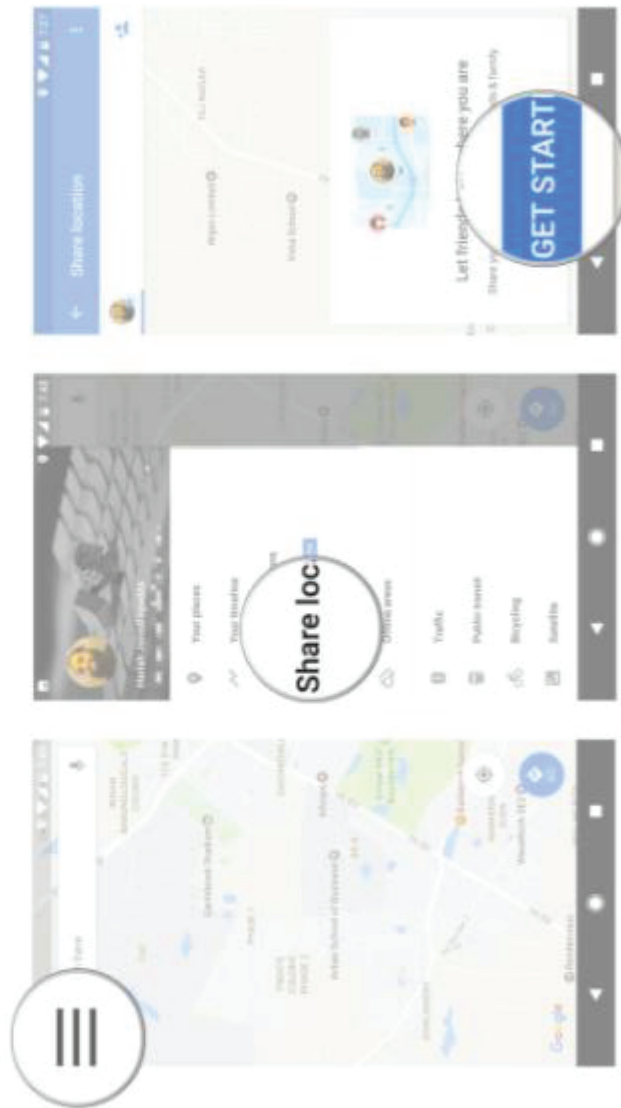
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



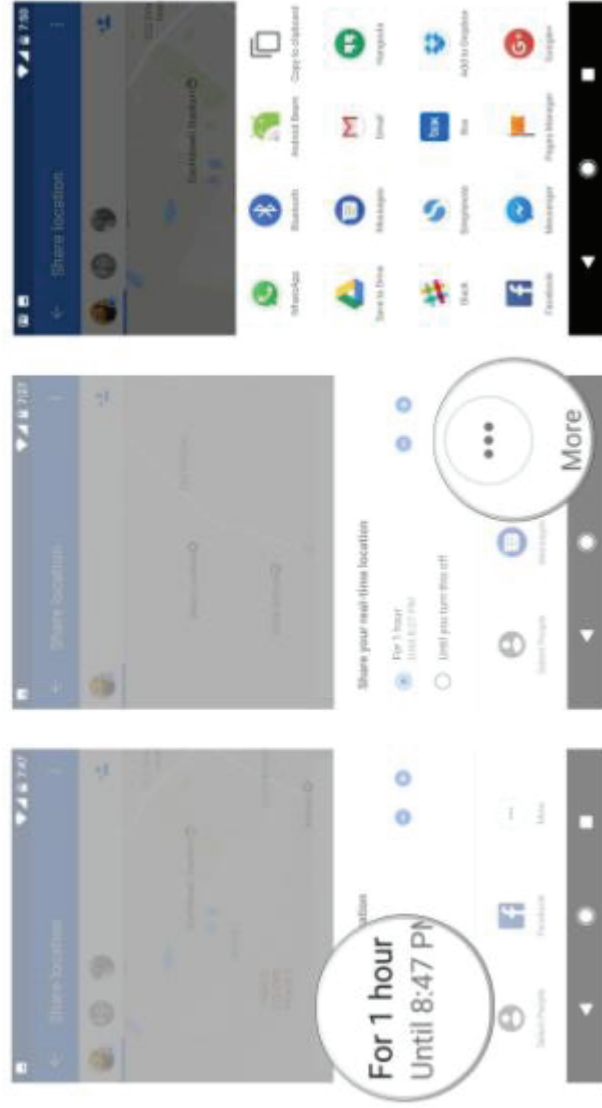
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

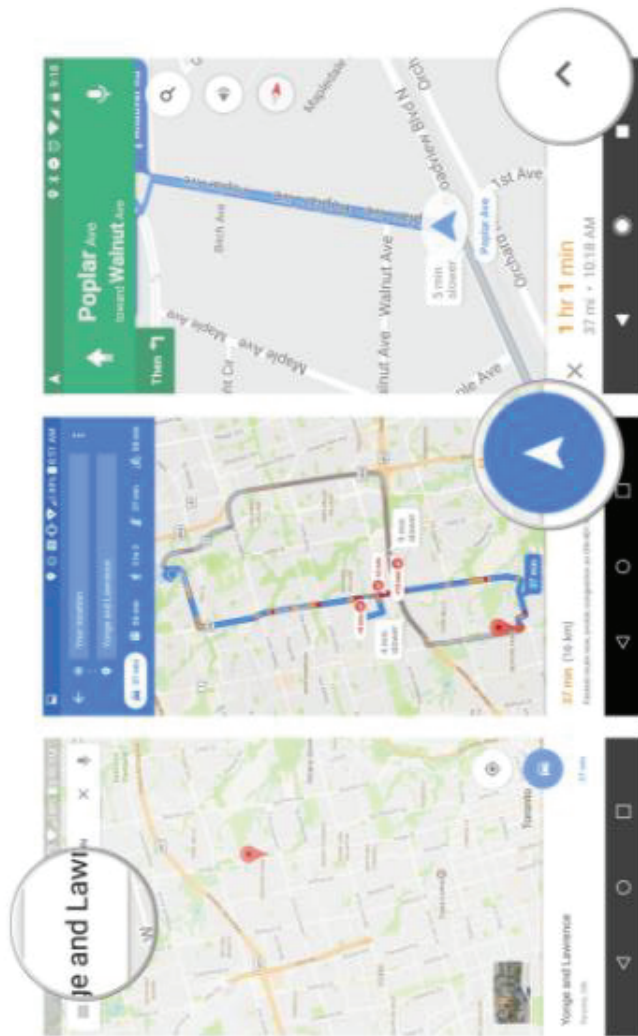
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



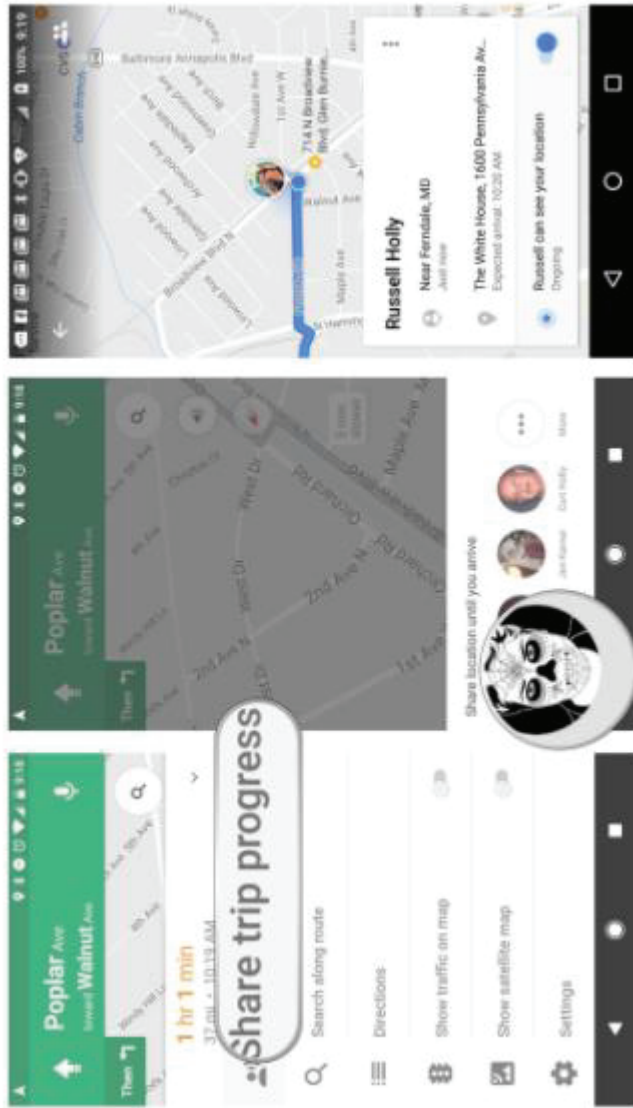
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



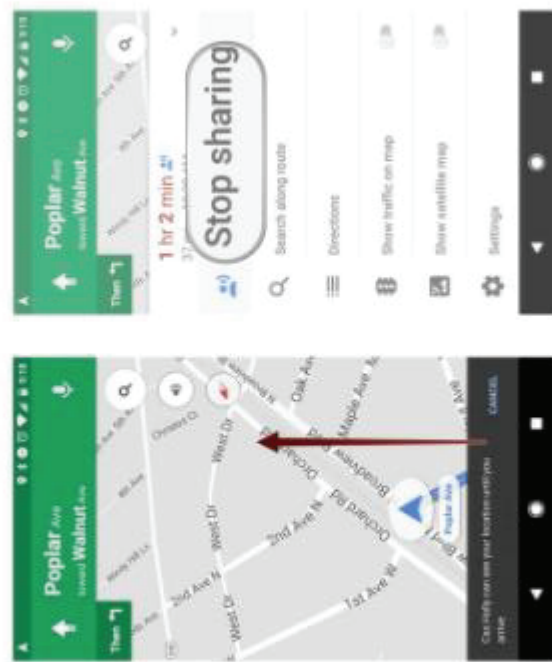
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?

<https://www.androidcentral.com/how-share-location-google-maps>





As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

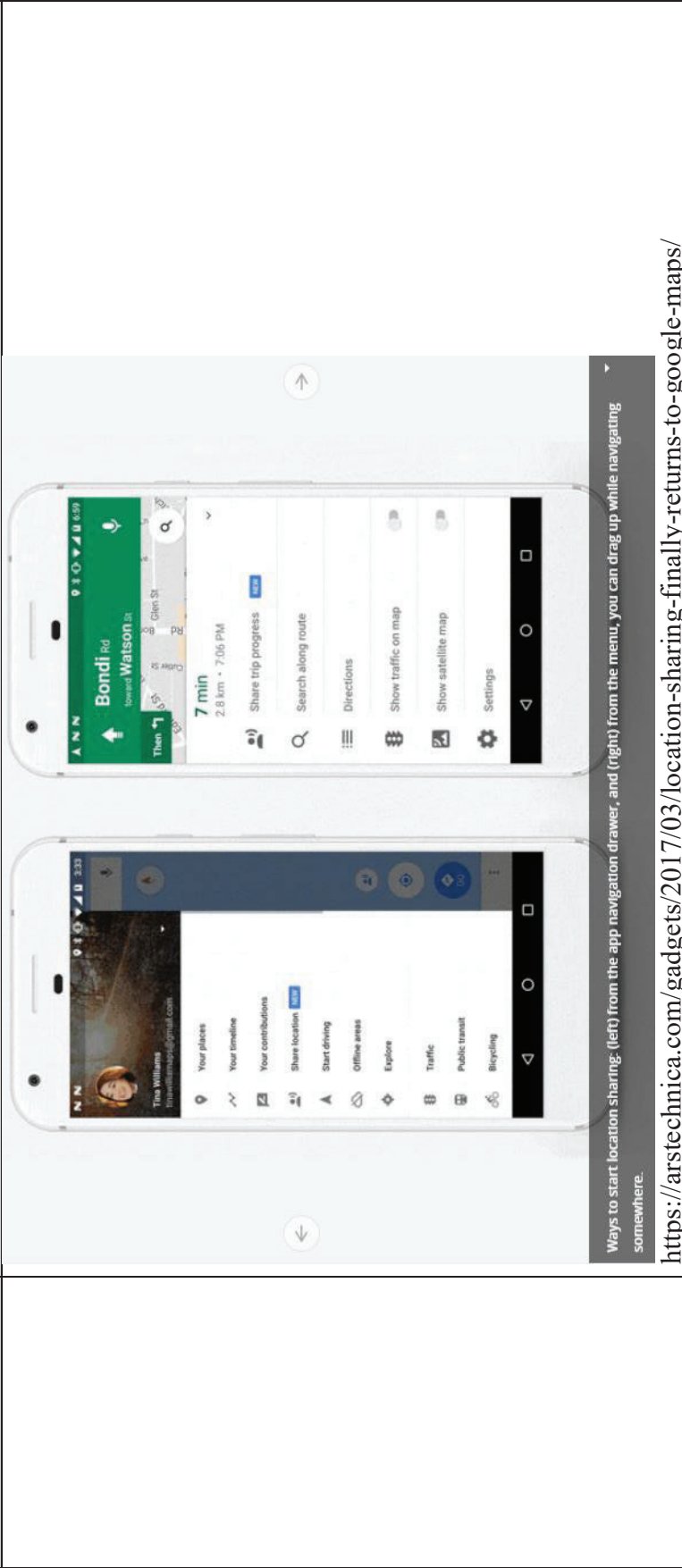


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

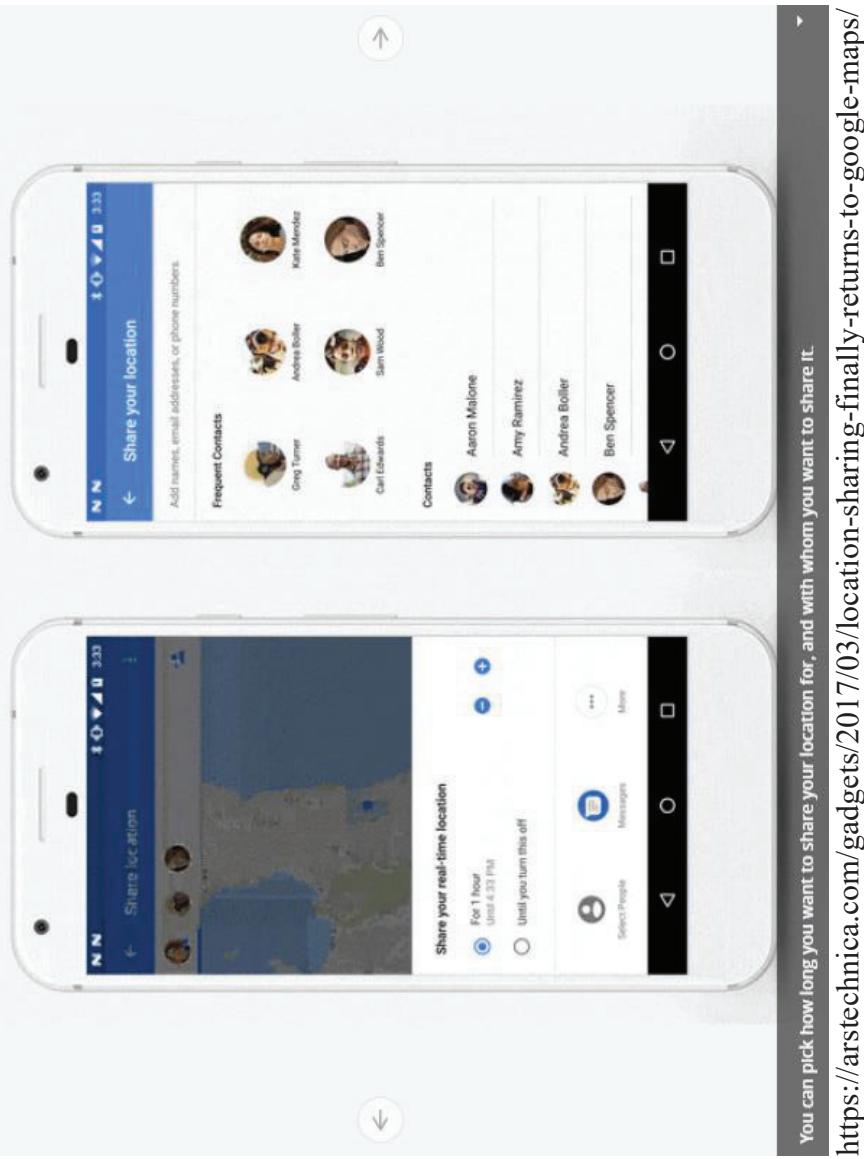
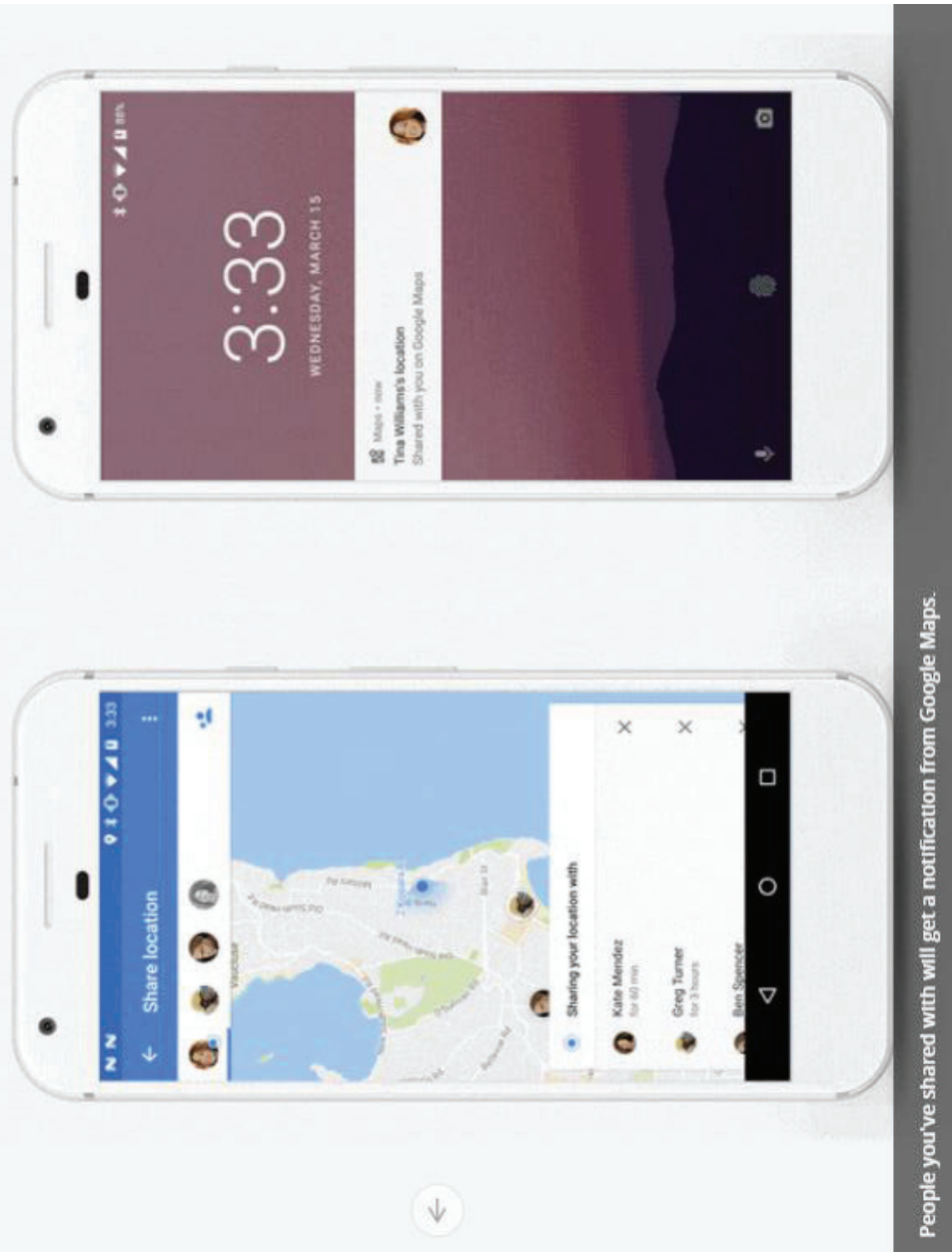


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

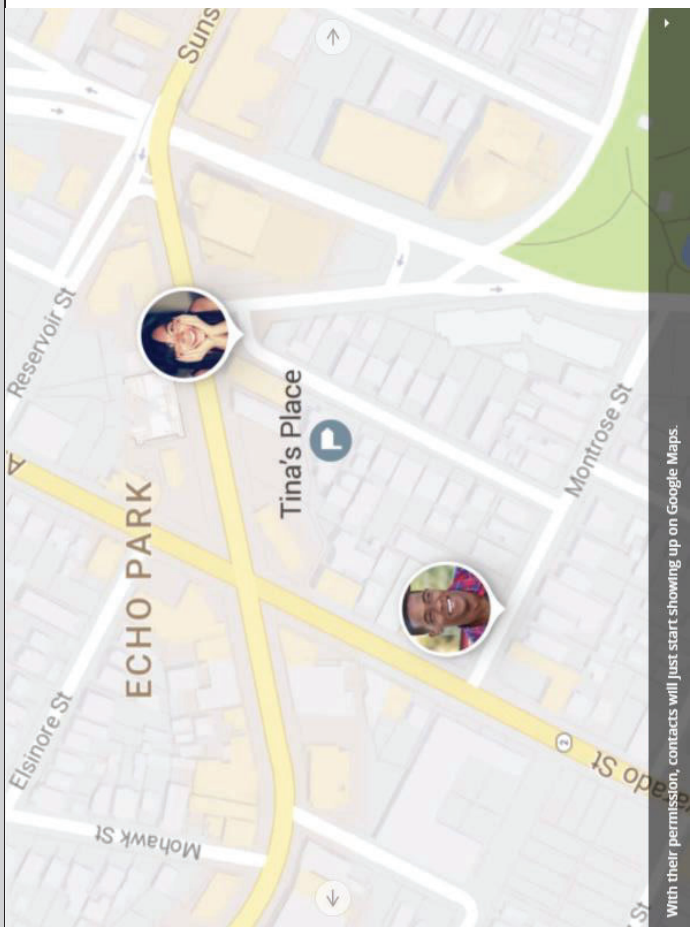


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

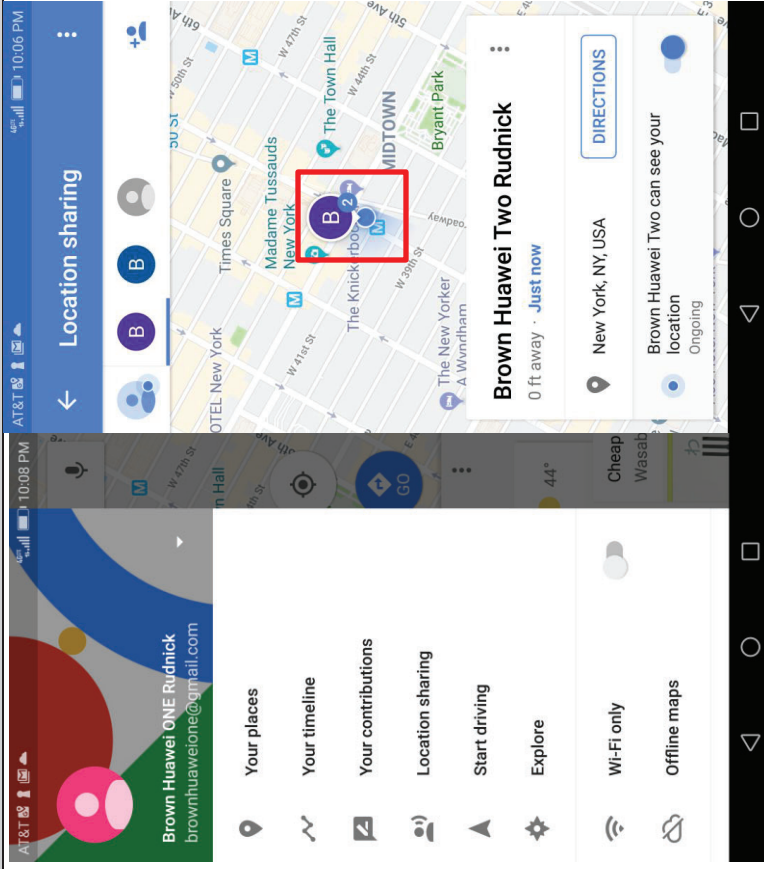


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>
Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Location information is shared via IP-based communication resulting in map that displays location information

Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | ZTE <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider-provider.html</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contact's _id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|---|-------------|--|-----------|--|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | One of: <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. |
| https://developer.android.com/guide/topics/providers/contacts-provider.html | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

B-659

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-662

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,           // 1
57             Data.LOOKUP_KEY,          // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,    // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-667

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

B-669

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "")
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*status*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178 } else if (METHOD_GET.equals(method)) {
179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180         logHttpHeaders(connection.getRequestProperties());
181     }
182     connection.setRequestMethod(METHOD_GET);
183 }
184 // Get response
185 final int responseCode = connection.getResponseCode();
186 final String responseMessage = connection.getResponseMessage();
187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189     logHttpHeaders(connection.getHeaderFields());
190 }
191 if (responseCode / 100 != 2) {
192     throw new MmsHttpException(responseCode, responseMessage);
193 }
194 final InputStream in = new BufferedInputStream(connection.getInputStream());
195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196 final byte[] buf = new byte[4096];
197 int count = 0;
198 while ((count = in.read(buf)) > 0) {
199     byteOut.write(buf, 0, count);
200 }
201 in.close();
202 final byte[] responseBody = byteOut.toByteArray();
203 Log.d(MmsService.TAG, "HTTP: response size="
204     + (responseBody != null ? responseBody.length : 0));
205 return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUri, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-679

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="196 1520 220 1587">ZTE</p> <p data-bbox="233 506 302 1587">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <div data-bbox="350 359 399 1570" style="background-color: #f0f0f0; padding: 5px;"><pre data-bbox="360 1121 386 1570">public static LocationRequest create ()</pre></div> <p data-bbox="423 1073 448 1575">Create a location request with default parameters.</p> <p data-bbox="480 464 540 1575">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="565 1472 589 1556">Returns</p> <ul data-bbox="610 1289 634 1549" style="list-style-type: none">• a new location request <p data-bbox="651 306 675 1587">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <pre> public Task<Location> getLastLocation () Returns the best most recent location currently available. If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned. This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates. public Task<LocationAvailability> getLocationAvailability () Returns the availability of location data. When isLocationAvailable() returns true, then the location returned by getLastLocation() will be reasonably up to date within the hints specified by the active LocationRequest s. If the client isn't connected to Google Play services and the request times out, null is returned. Note it's always possible for getLastLocation() to return null even when this method returns true (e.g. location settings were disabled between calls). https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient </pre> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|-----------------------|--|----------------------|---------------------------------------|-----------------------|--|---------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callback</code></td> <td>The callback for the location updates.</td> </tr> <tr> <td><code>looper</code></td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | <code>request</code> | The location request for the updates. | <code>callback</code> | The callback for the location updates. | <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| <code>request</code> | The location request for the updates. | | | | | | |
| <code>callback</code> | The callback for the location updates. | | | | | | |
| <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|---------|---------------------------------------|----------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="831 1264 899 1570">request</td> <td data-bbox="831 357 899 1264">The location request for the updates.</td> </tr> <tr> <td data-bbox="899 1264 967 1570">callbackIntent</td> <td data-bbox="899 357 967 1264">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p>Public Constructors</p> <pre> public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) </pre> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> <pre> public void getMapAsync (OnMapReadyCallback callback) </pre> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback The callback object that will be triggered when the map is ready to be used.</p> <pre> public final void onCreate (Bundle savedInstanceState) </pre> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[41D] transmitting IP-based messages including a location of the first device to the respective second devices;</p> | <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claim 1[D] and 28[D], which are incorporated herein by reference in their entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:

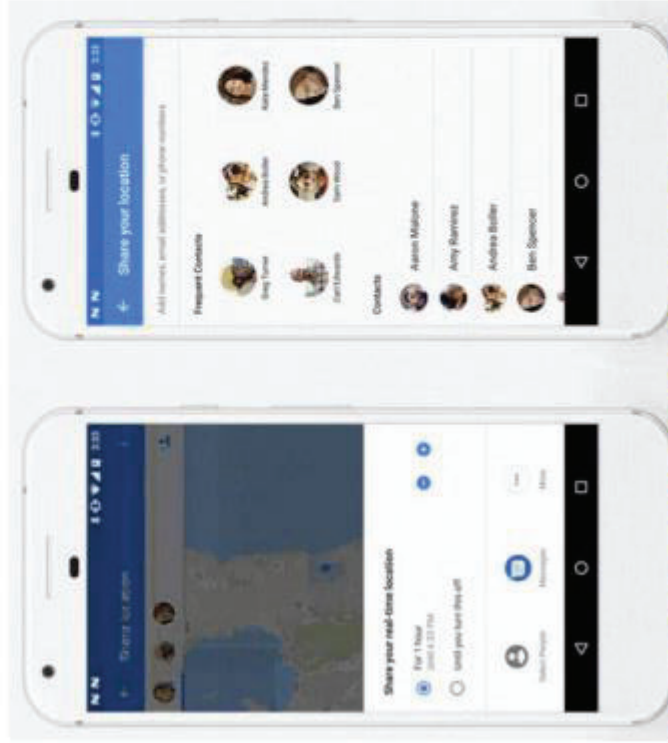
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. If you haven't already, add their Gmail address to your Google Contacts [\[2\]](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap the Menu  > **Share location** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

<https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&hl=en>



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Below are exemplary methods used by Google applications to obtain, send, and receive locations.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

The Google Play services Location API

The Google Play services Location API is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:

- Determine the device location.
- Listen for location changes.
- Determine the mode of transportation, if the device is moving.
- Create and monitor predefined geographical regions, known as geofences.

The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class Making Your App Location Aware or the Location API Reference. Code examples are included as part of the Google Play services SDK.

<https://developers.google.com/maps/documentation/android-api/location>

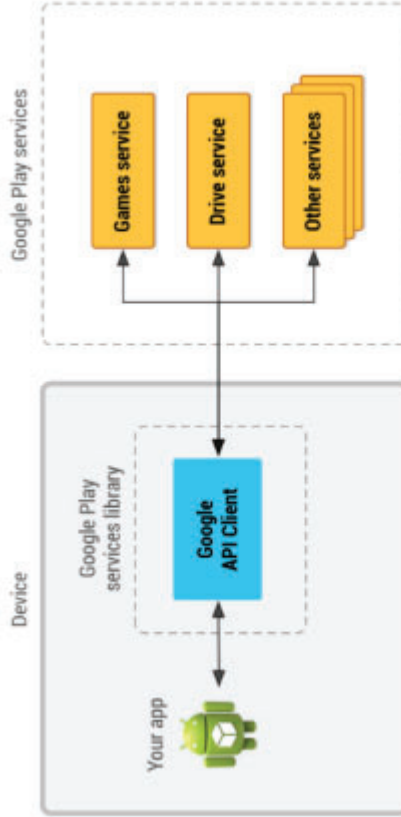


Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.

<https://developers.google.com/android/guides/api-client#Starting>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get the Last Known Location

Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's `getLastLocation()` method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the [Specify App Permissions](#) section of this document.

To request the last known location, call the `getLastLocation()` method, passing it your instance of the `GoogleApiClient` object. Do this in the `onConnected()` callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:

```

public class MainActivity extends AppCompatActivity implements
    ConnectionCallbacks, OnConnectionFailedListener {
    ...
    @Override
    public void onConnected(Bundle connectionHint) {
        mLastLocation = LocationServices.FusedLocationApi.getLastLocation(
            mGoogleApiClient);
        if (mLastLocation != null) {
            mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));
            mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));
        }
    }
}

```

The `getLastLocation()` method returns a `Location` object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.

<https://developer.android.com/training/location/retrieve-current.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>Determining the user's current location</p> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() { var startPos; var geoSuccess = function(position) { startPos = position; document.getElementById('startLat').innerHTML = startPos.coords.latitude; document.getElementById('startLon').innerHTML = startPos.coords.longitude; }; navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p>https://developers.google.com/web/fundamentals/native-hardware/user-location/</p> | <p>[41E] presenting, via an interactive display of the first device, an interactive map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions</p> |
| <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claims 1[E] and 28[E], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user-selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claims 1[E] and 28[E], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>corresponding to the respective locations of the second devices;</p> | <p>ZTE</p> <p>Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p>Exemplary Support for Google Maps:</p> <p>Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1596">ZTE</p> <p data-bbox="259 693 300 1228">ZTE Mobile Location Service System</p> <p data-bbox="332 903 365 1050">2004-01-31</p> <p data-bbox="430 1386 462 1564">I. Introduction</p> <p data-bbox="470 357 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="682 357 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="812 199 893 1596">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


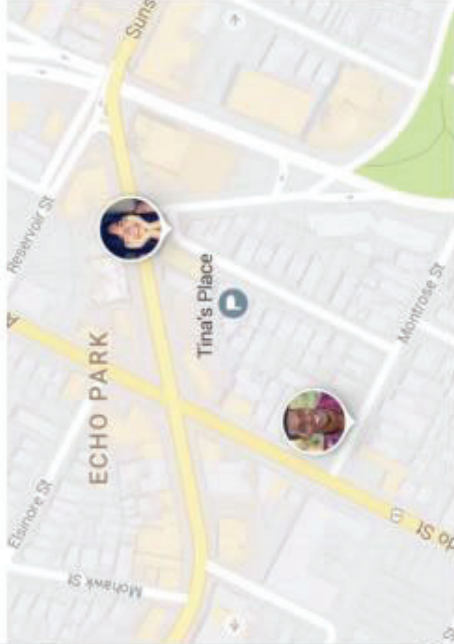
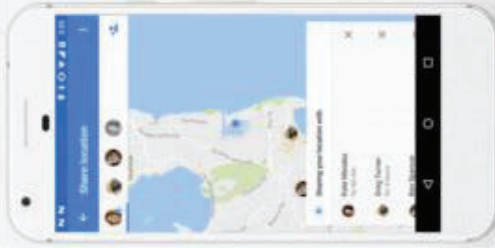
| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app.
2. Tap the menu icon > **Share location**.
3. Choose someone.

- To see an updated location, tap on a friend's icon > **More** > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app.
2. On the map, tap the location icon.
3. At the bottom, tap **More**.
4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

https://support.google.com/maps/answer/7326816?hl=en&ref_topic=3092425&co=GEMIE.Platform%3DAndroid&oc=1



Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap More  > **Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share.**
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap More .
4. To temporarily hide someone, tap **Hide from map.** You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

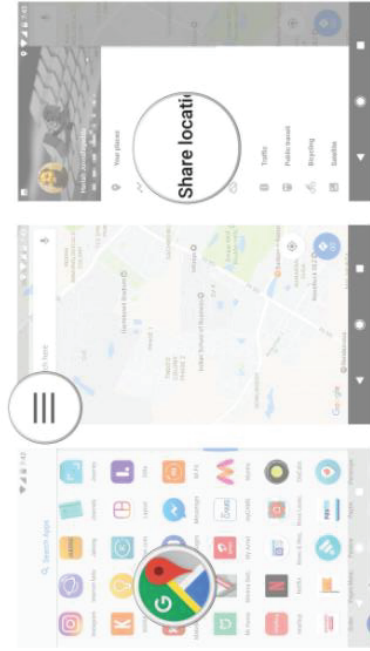
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

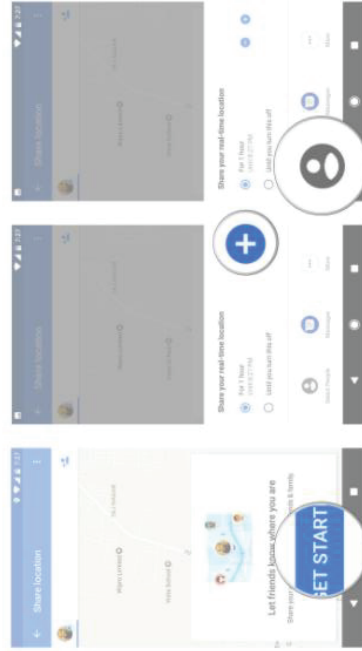
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



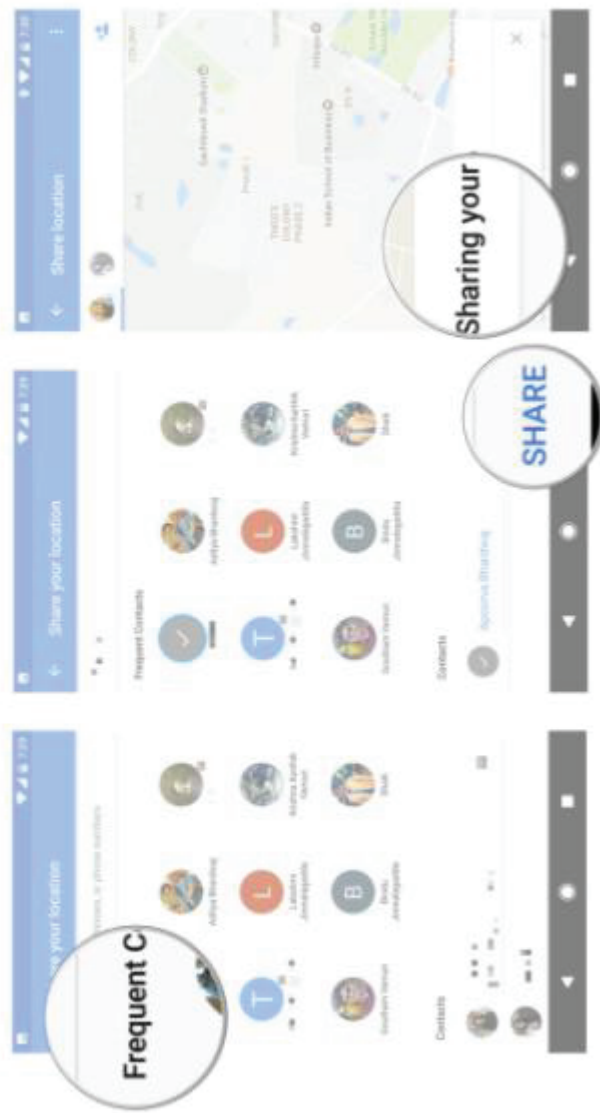
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

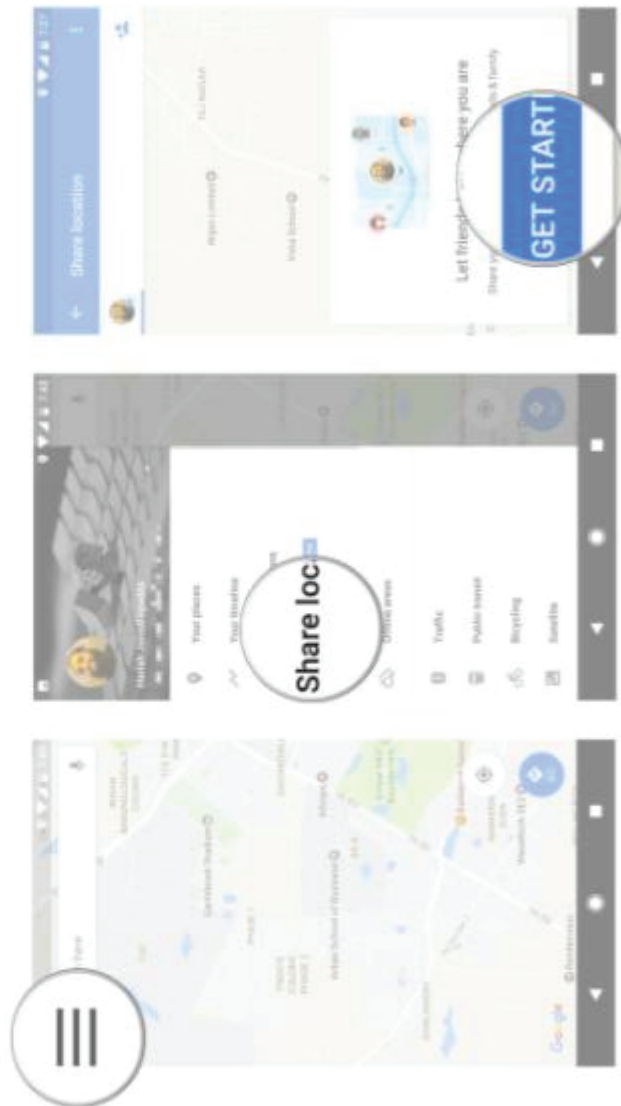
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



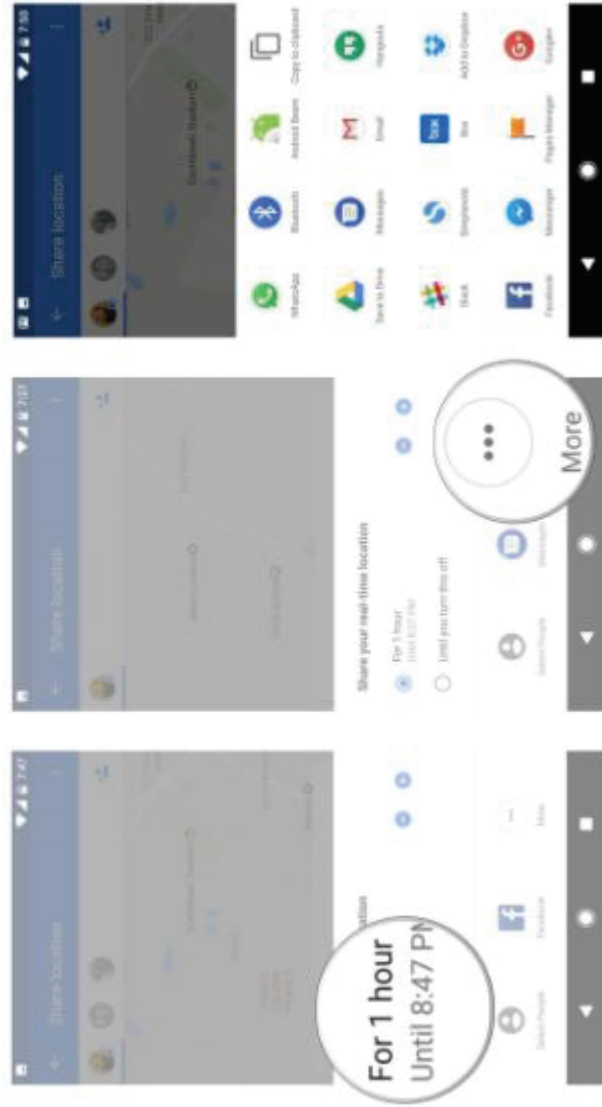
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

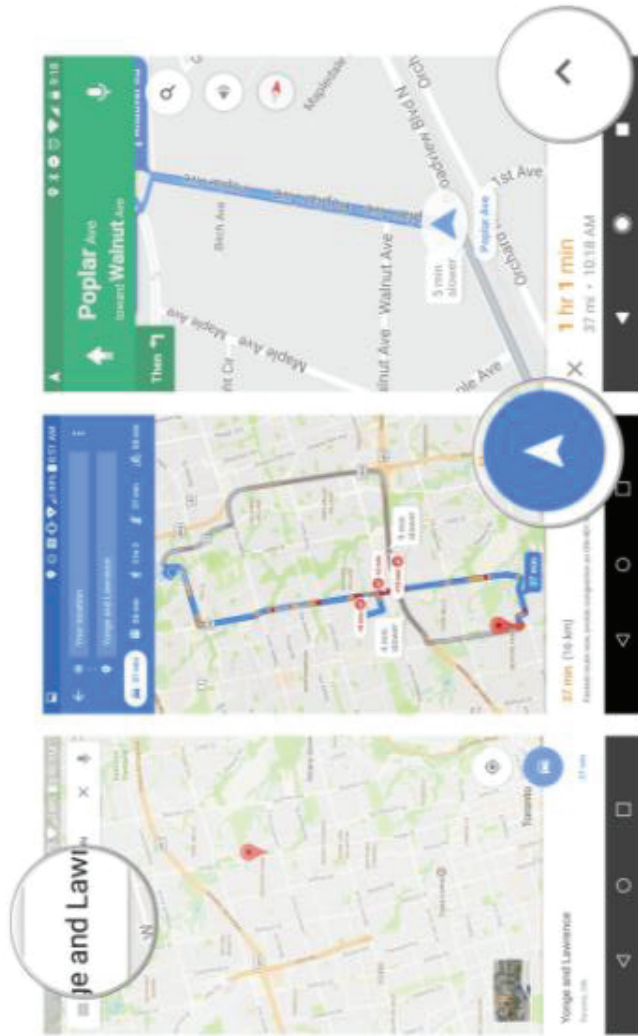
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



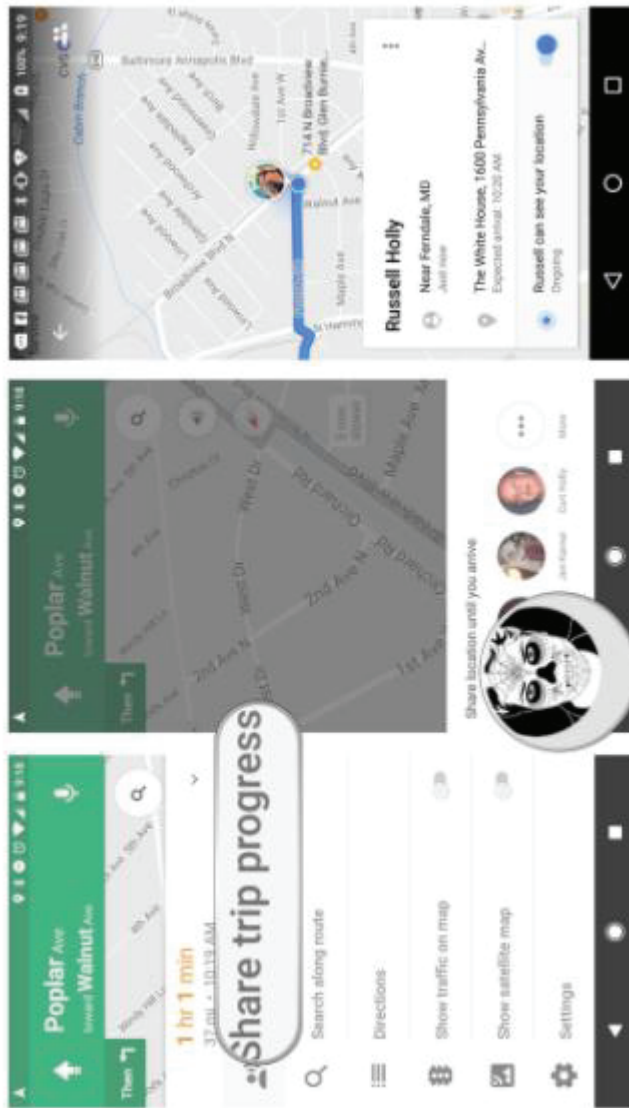
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



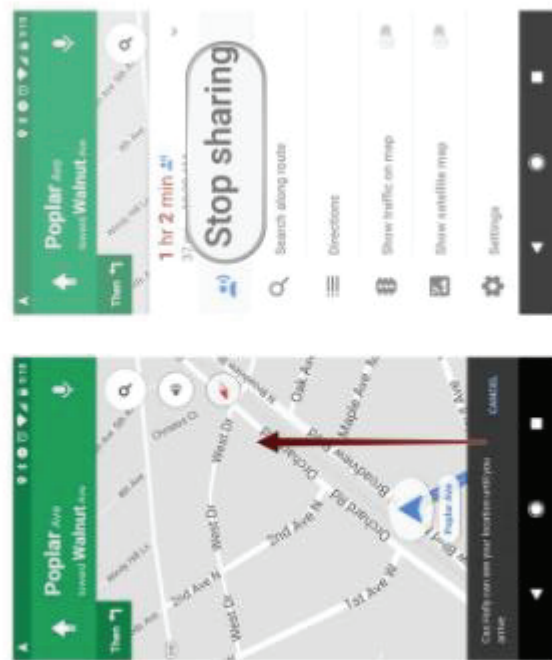
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

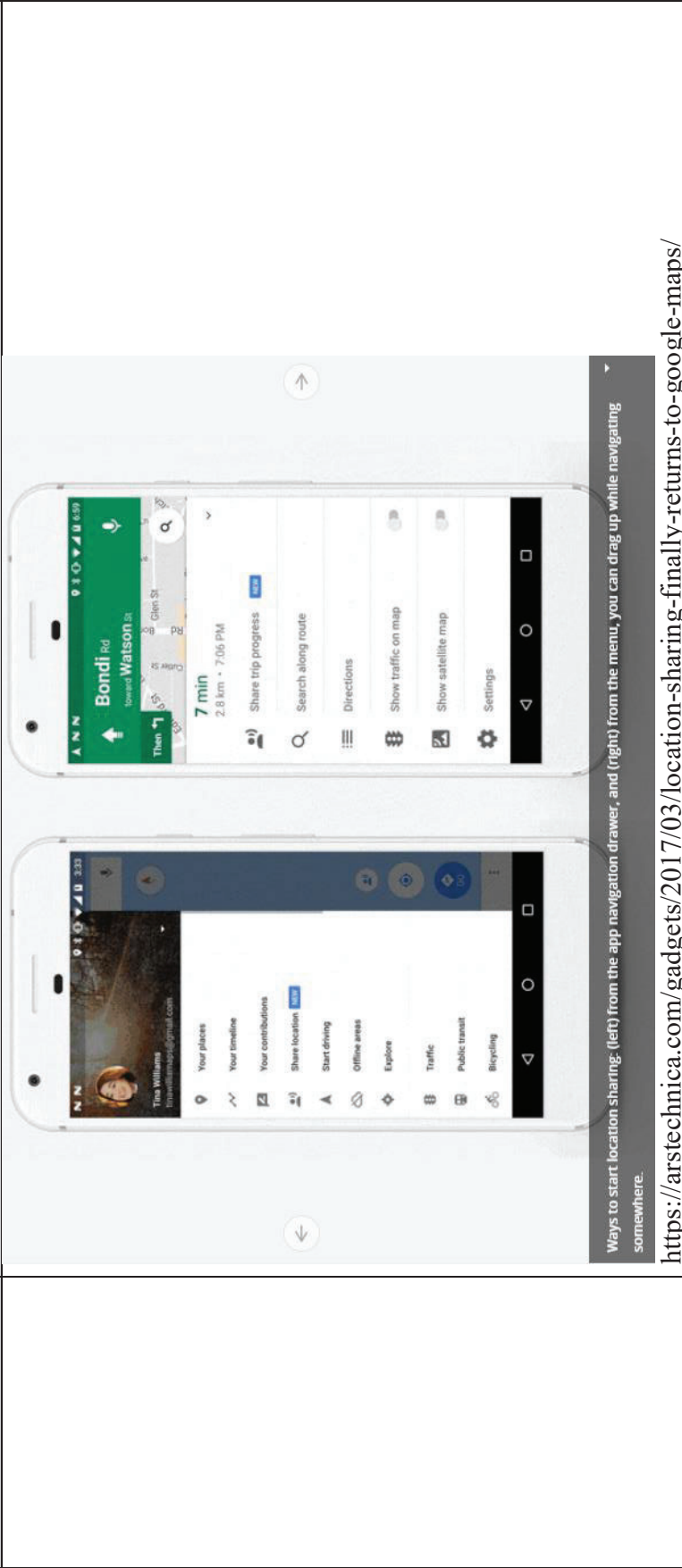


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

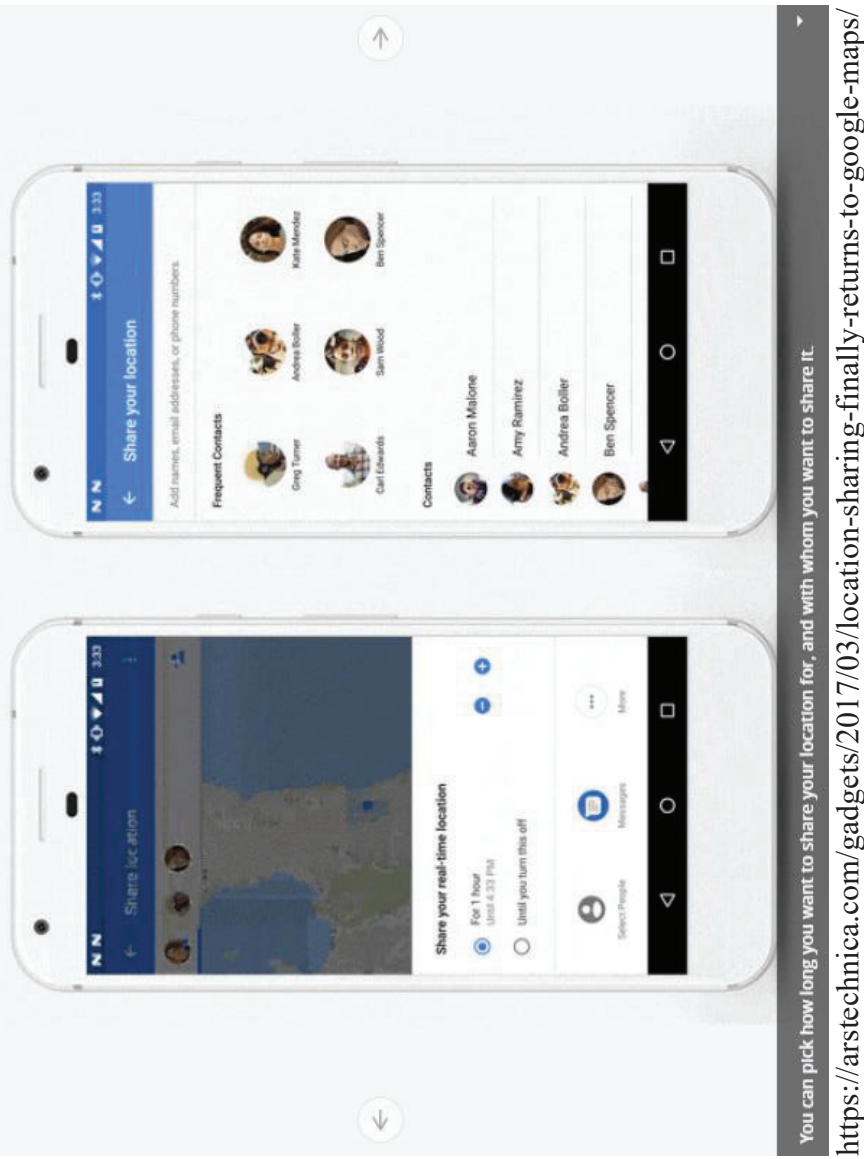
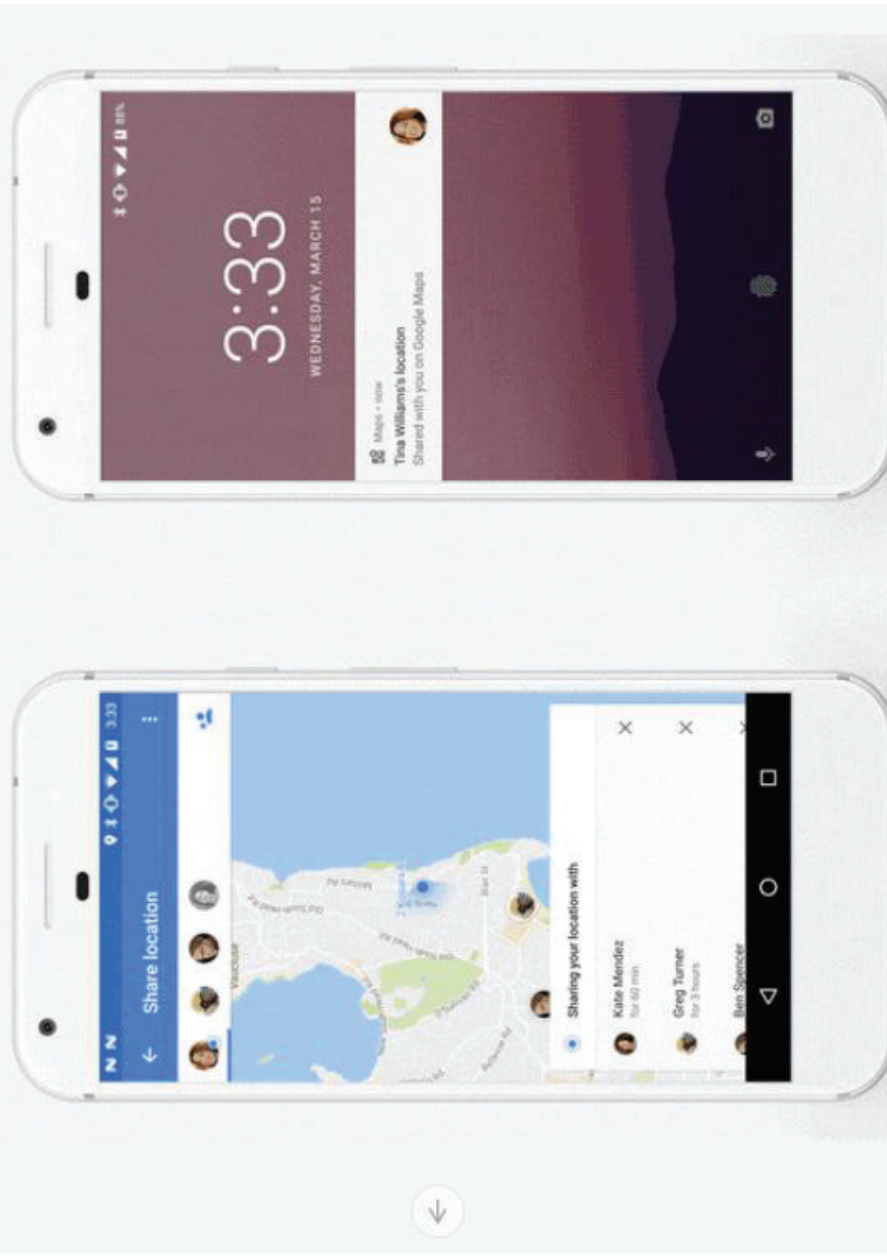


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



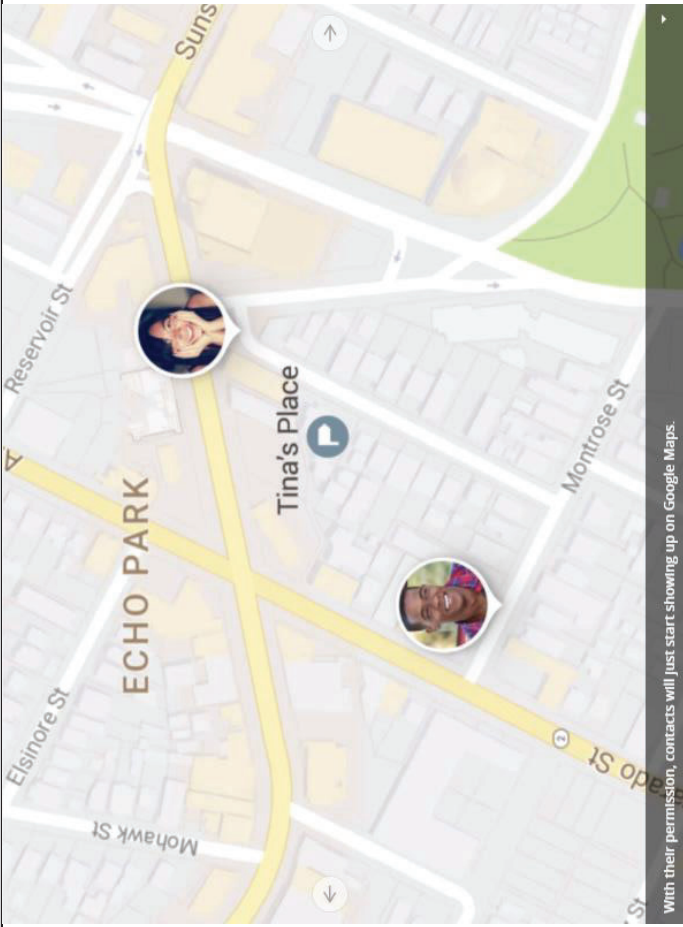
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



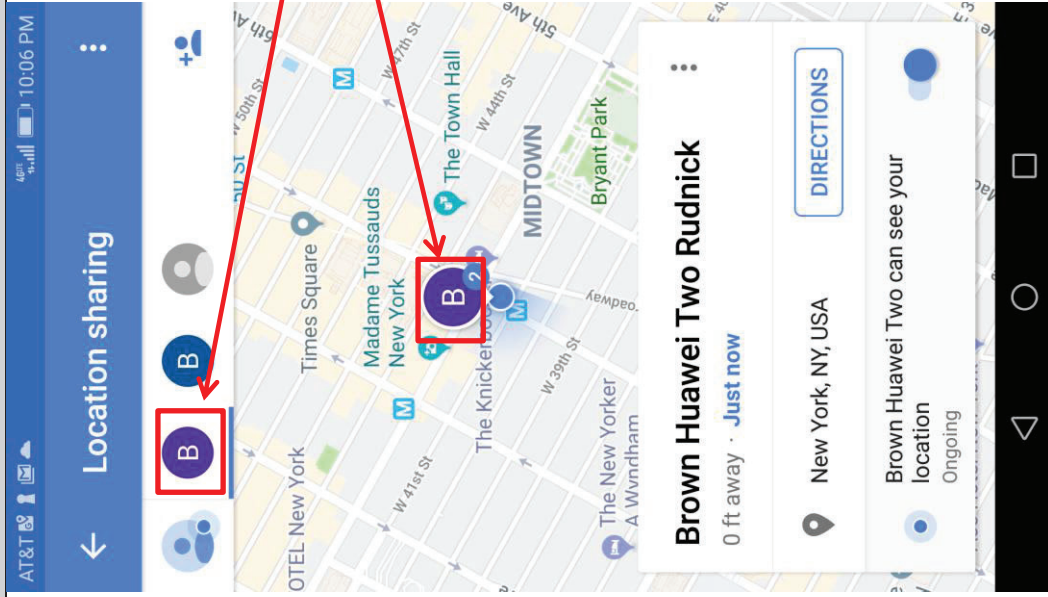
With their permission, contacts will just start showing up on Google Maps.
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary User
Selectable Symbols

Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p>ZTE</p> <p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre>public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p>Returns</p> <ul style="list-style-type: none">• a new location request <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <pre> public Task<Location> getLastLocation () Returns the best most recent location currently available. If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned. This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates. public Task<LocationAvailability> getLocationAvailability () Returns the availability of location data. When isLocationAvailable() returns true, then the location returned by getLastLocation() will be reasonably up to date within the hints specified by the active LocationRequest s. If the client isn't connected to Google Play services and the request times out, null is returned. Note it's always possible for getLastLocation() to return null even when this method returns true (e.g. location settings were disabled between calls). https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient </pre> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|-----------------------|--|----------------------|---------------------------------------|-----------------------|--|---------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callback</code></td> <td>The callback for the location updates.</td> </tr> <tr> <td><code>looper</code></td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | <code>request</code> | The location request for the updates. | <code>callback</code> | The callback for the location updates. | <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| <code>request</code> | The location request for the updates. | | | | | | |
| <code>callback</code> | The callback for the location updates. | | | | | | |
| <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener Public Constructors public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) https://developers.google.com/android/reference/com/google/android/gms/maps/MapView |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public void getMapAsync (OnMapReadyCallback callback)</code></p> <p>Returns a non-null instance of the <code>Goog1eMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>Goog1eMap</code> object provided by the callback is non-null. <p>Parameters</p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void onCreate (Bundle savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method. https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[41F] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claims 1[F] and 28[F], which are incorporated herein by reference in their entirety.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>action and, based thereon, sending data to the one or more second devices;</p> | <p>ZTE</p> <p>specified, data is sent from the first device to the second device via a server.</p> <p><u>Exemplary Support for Google Maps:</u></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 🗣️ > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap ⚙️ > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
|---------------------------|---|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap **More**  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > **More**  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAn

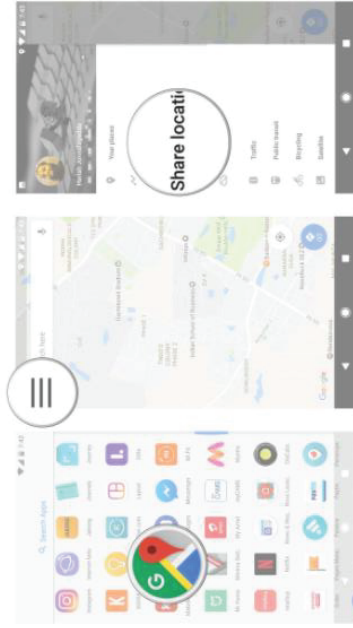
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

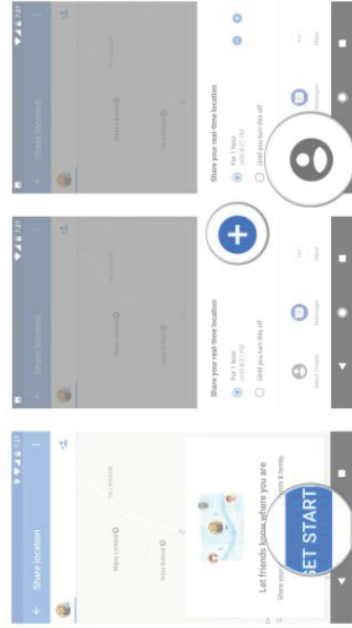
ZTE
droid&oco=1

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



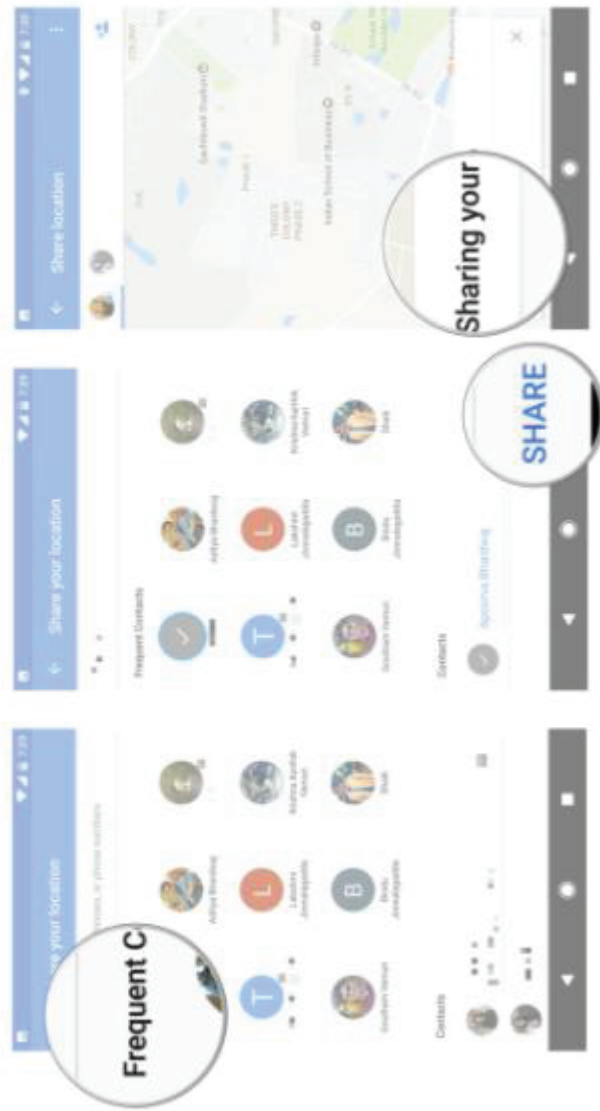
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

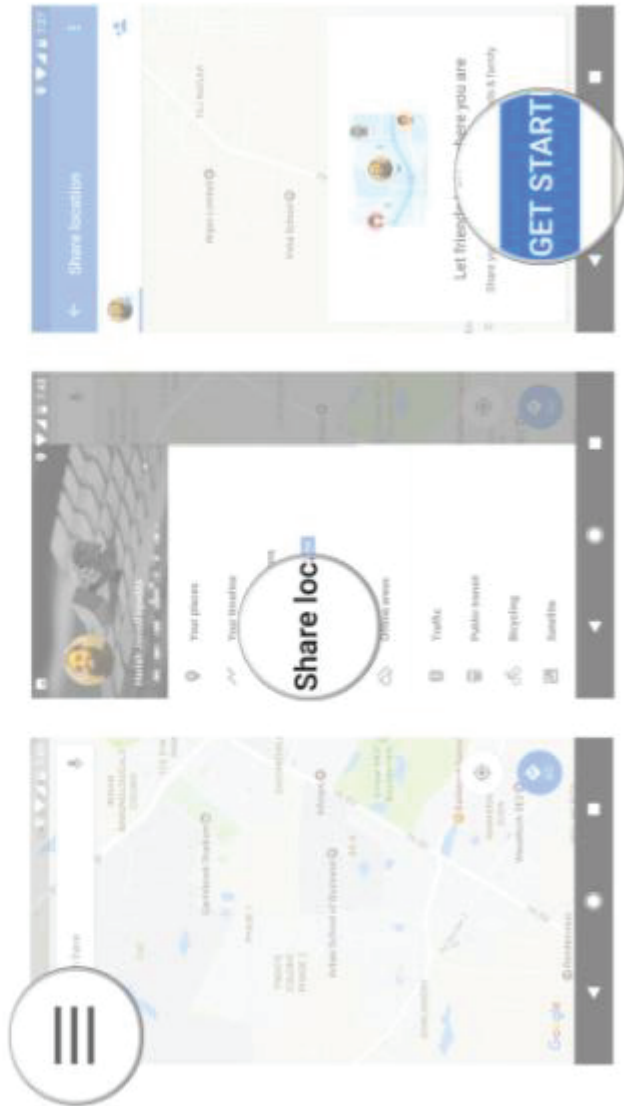
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



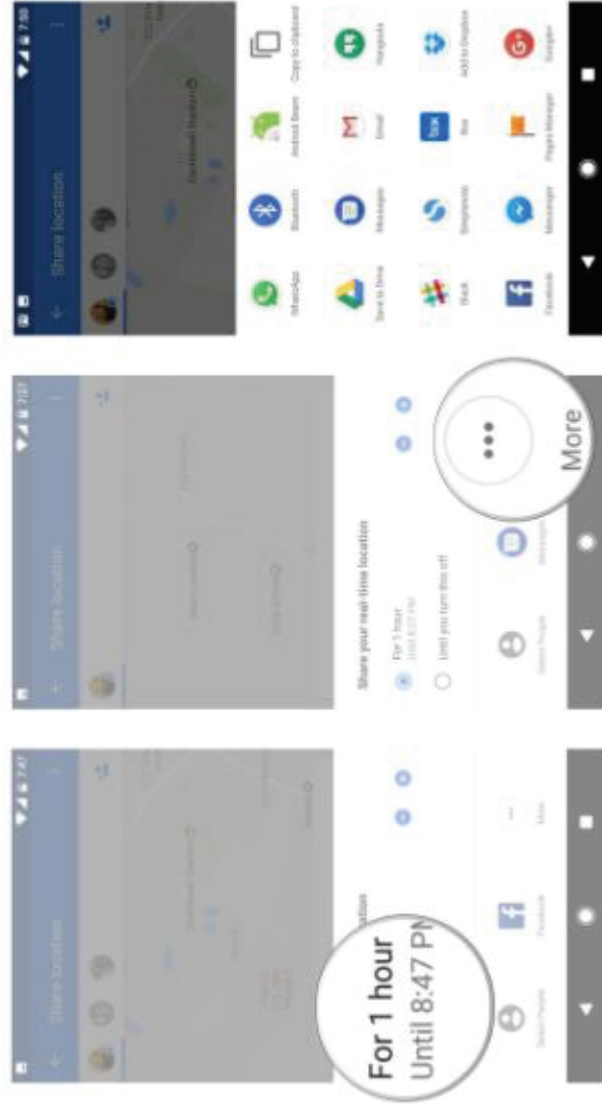
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

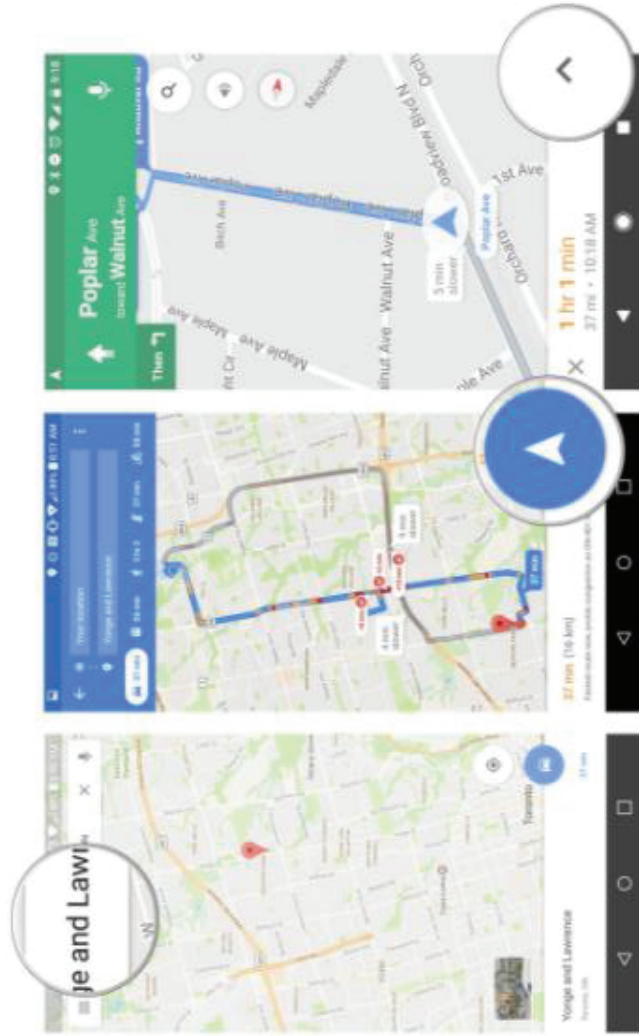
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



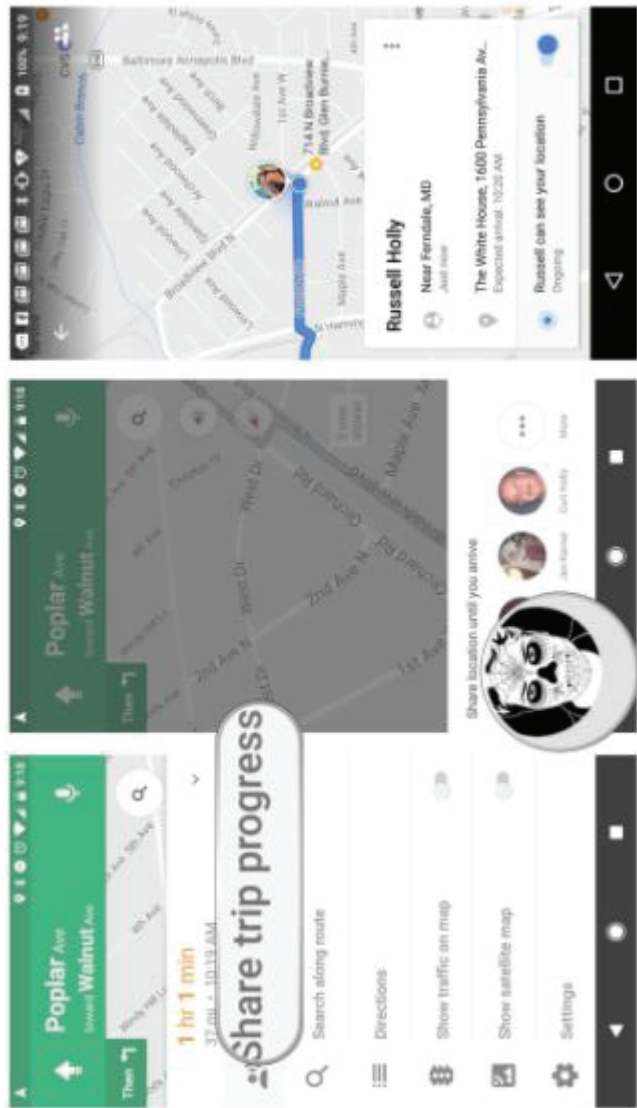
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



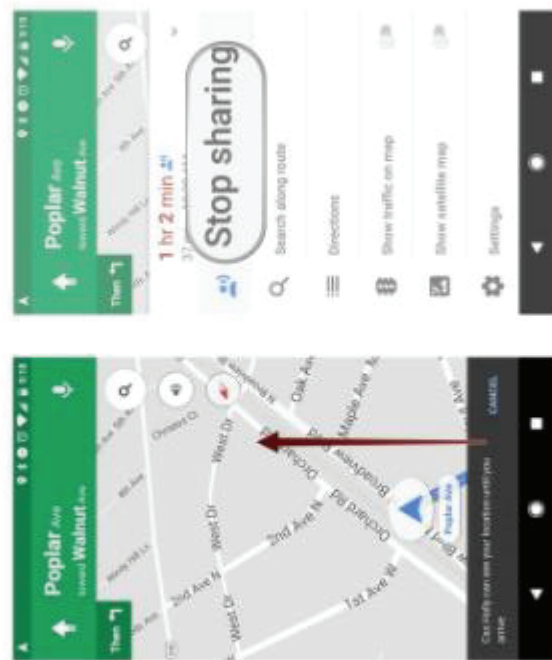
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>



As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

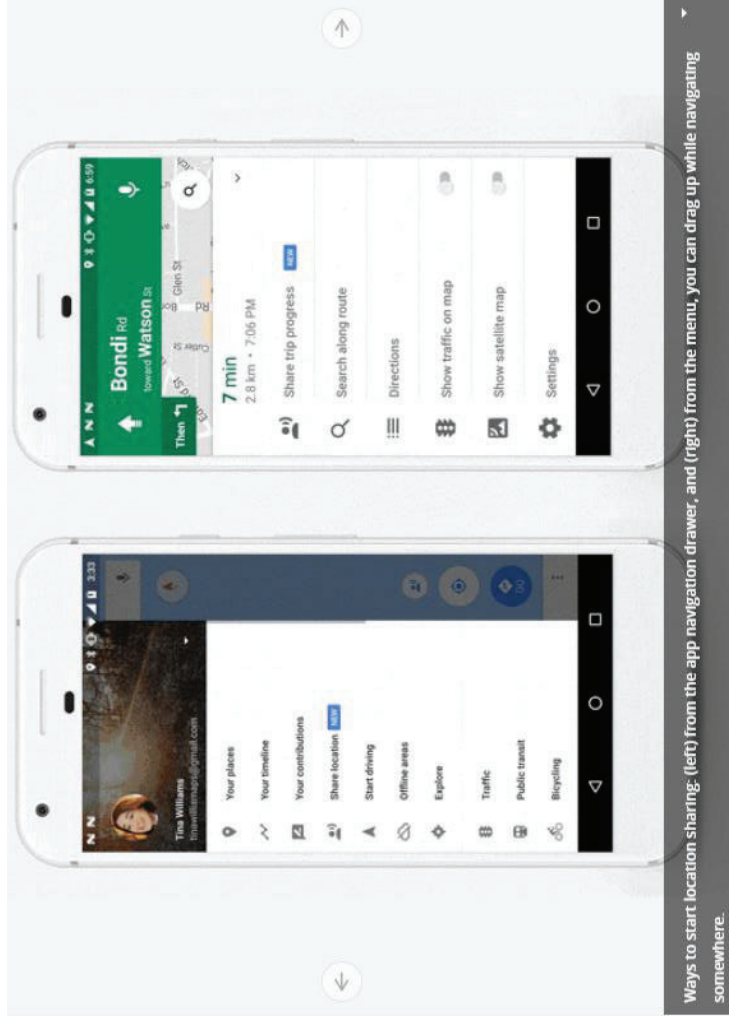
US9408055B2

ZTE

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

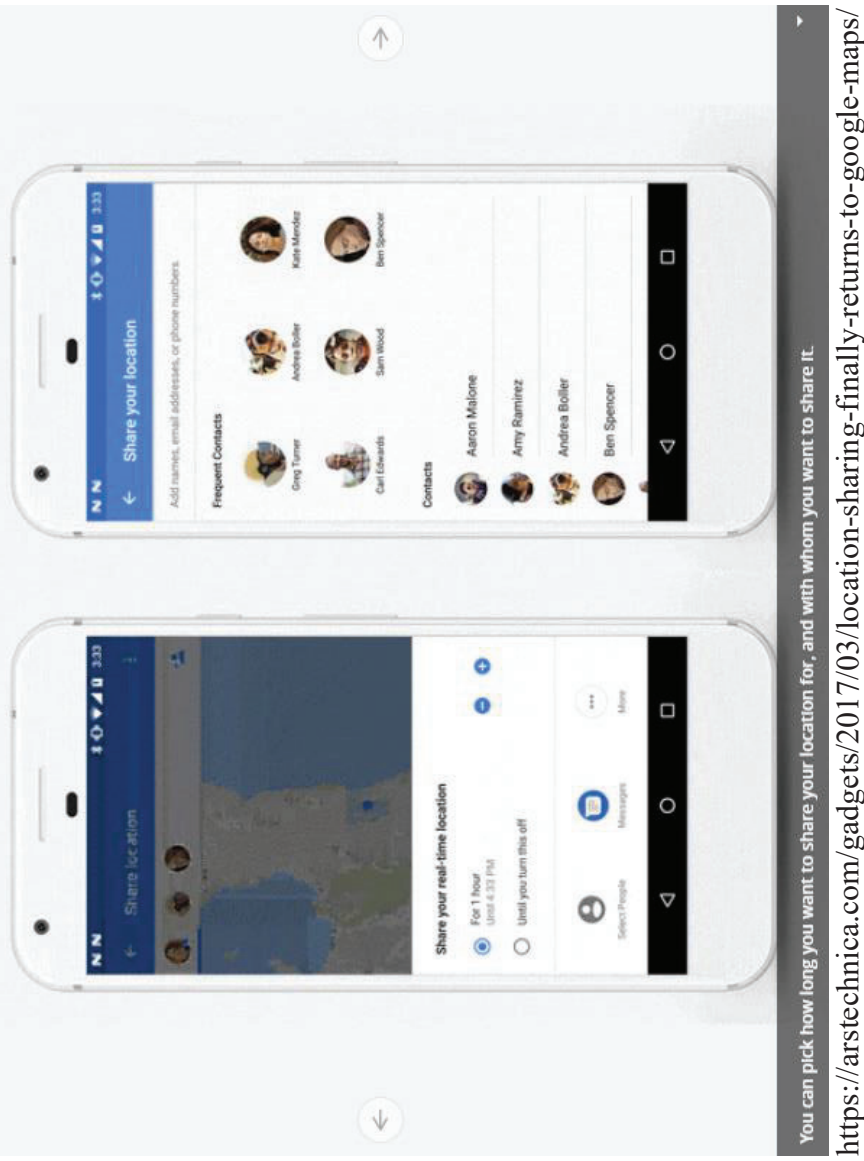
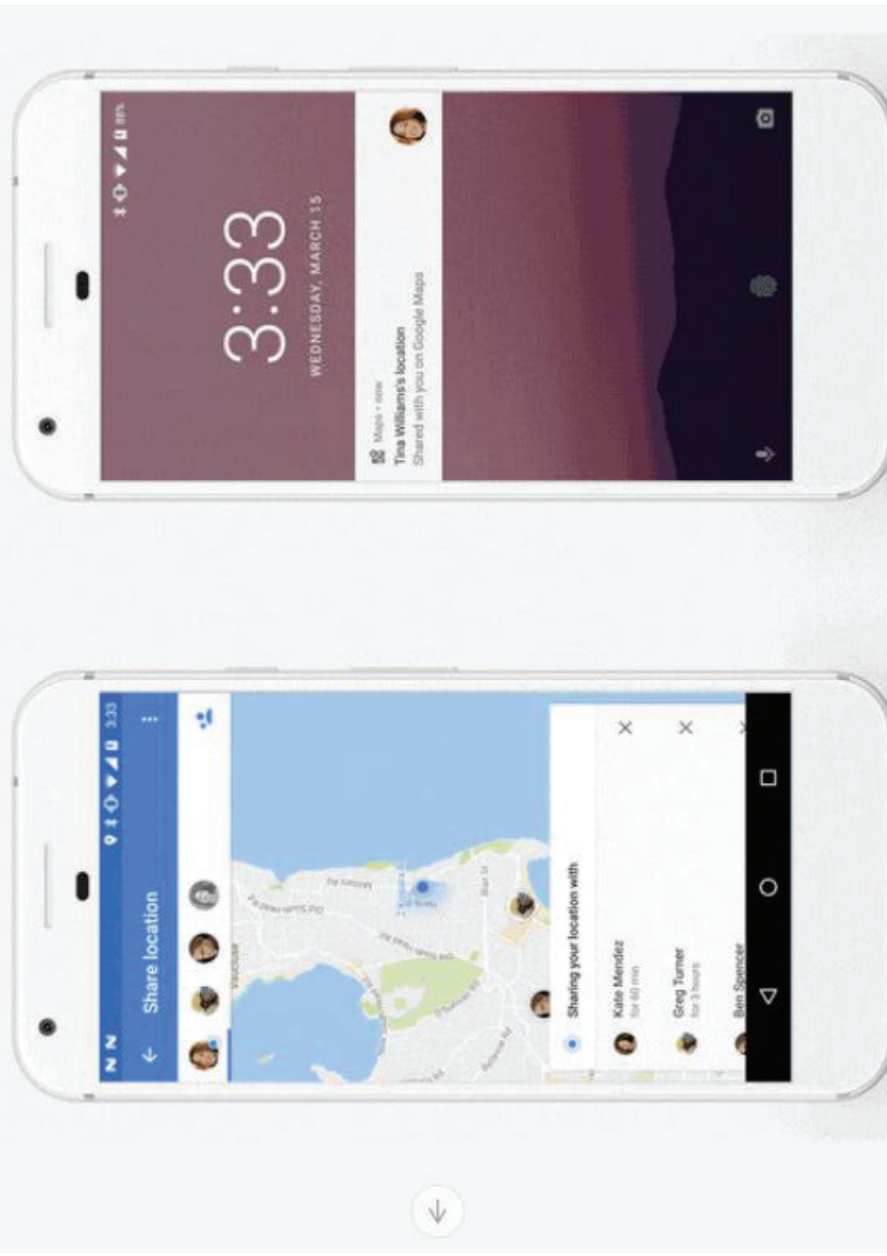


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



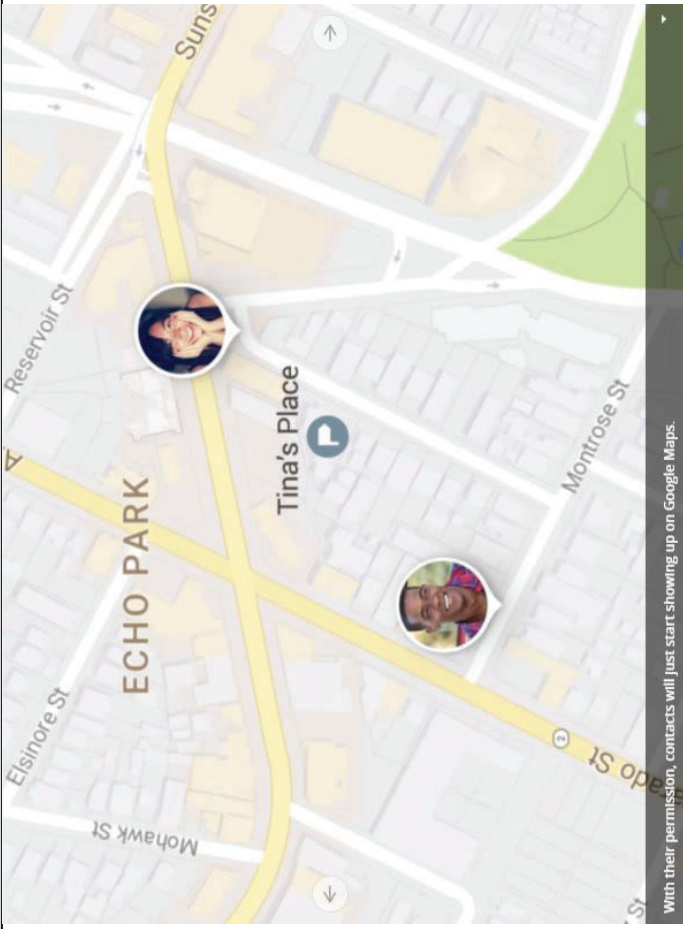
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Stop sharing

1. Open the Google Maps app .
2. Tap the Menu  > **Share location**.
3. Next to the person with whom you want to stop sharing, tap Remove .

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app.
2. Set a driving destination. Learn how to navigate to a place.
3. After you start navigation, tap **More** > **Share trip progress**.
4. Choose a person from the list.
5. Tap **Share**.
6. Location Sharing will stop when you reach your destination or stop navigating.

• To stop sharing before you arrive, tap **More** > **Stop sharing**.

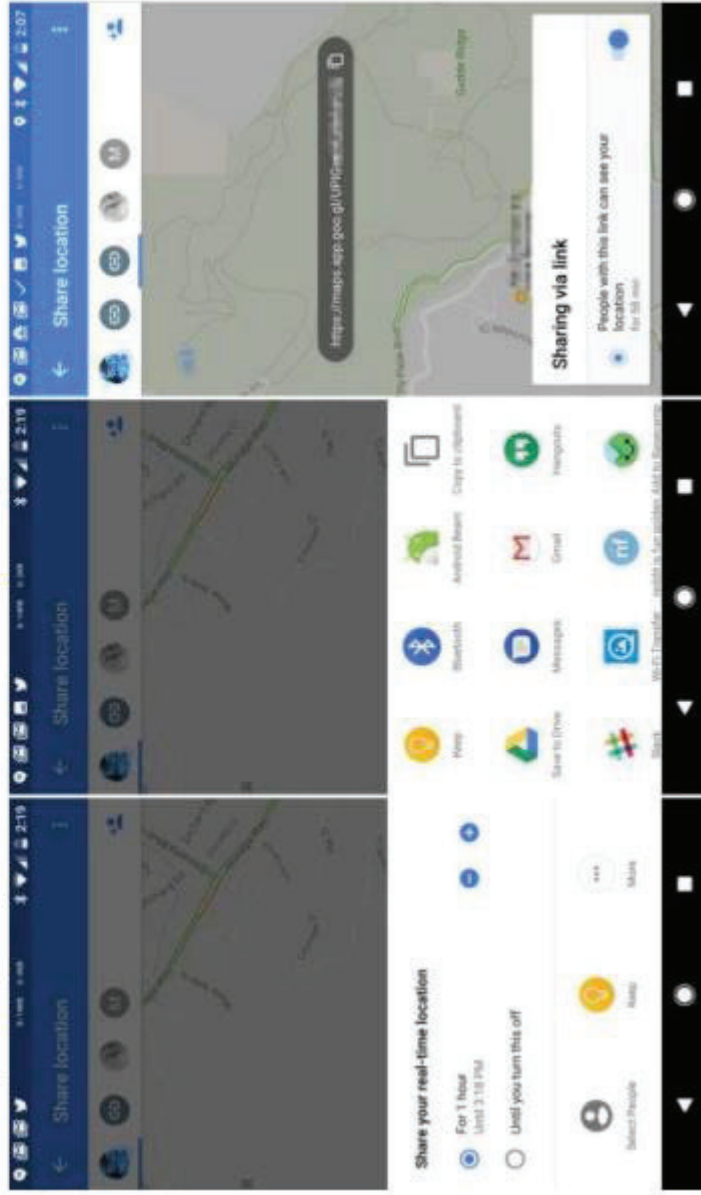
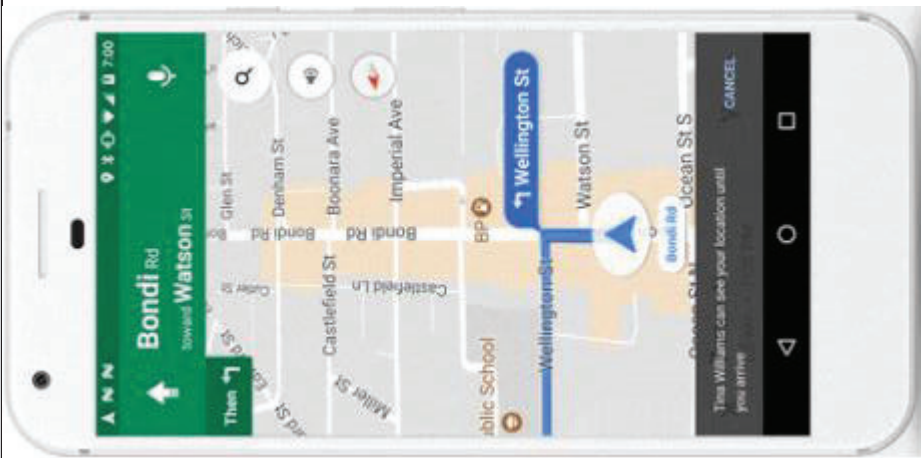


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

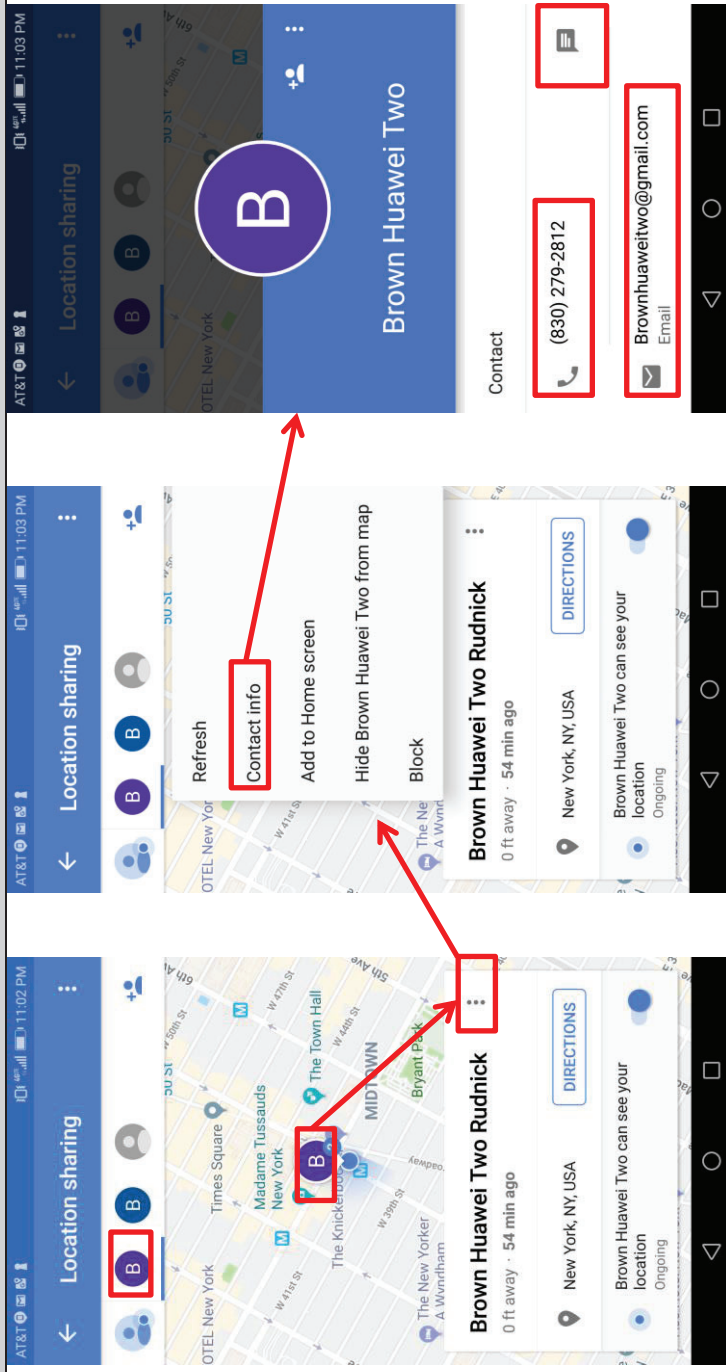


Exemplary Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 */ 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97 * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98 *
99 * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *   it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113 String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));
131     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUri, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUri, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-755

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <p data-bbox="235 504 300 1585">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <div data-bbox="349 357 397 1575"><pre>public static LocationRequest create ()</pre></div> <p data-bbox="422 1071 446 1575">Create a location request with default parameters.</p> <p data-bbox="479 462 544 1575">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="560 1470 584 1554">Returns</p> <ul data-bbox="609 1281 633 1554" style="list-style-type: none">• a new location request <p data-bbox="649 304 673 1585">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-----|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><code>public Task<Location> getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task<LocationAvailability> getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|--------------------|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|---------|---------------------------------------|----------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by <code>equals(Object)</code>) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using <code>hasResult(Intent)</code>, <code>extractResult(Intent)</code>, <code>hasLocationAvailability(Intent)</code>, and <code>extractLocationAvailability(Intent)</code>.</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="831 1264 899 1570">request</td> <td data-bbox="831 357 899 1264">The location request for the updates.</td> </tr> <tr> <td data-bbox="899 1264 967 1570">callbackIntent</td> <td data-bbox="899 357 967 1264">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check <code>isSuccessful()</code> to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | ZTE https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener Public Constructors public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) https://developers.google.com/android/reference/com/google/android/gms/maps/MapView |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>public void getMapAsync (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void onCreate (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[41G] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claims 1[G] and 28[G], which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.


Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.


Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.

Selection with Markers:

<https://developers.google.com/maps/documentation/android-api/marker>

Queries with Geo Tagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app.
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share. If you don't see this, tap More.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Share your E. T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app.
2. Set a driving destination. Learn how to navigate to a place.
3. After you start navigation, tap More.
4. Choose a person from the list.
5. Tap Share.
6. Location Sharing will stop when you reach your destination or stop navigating.

To stop sharing before you arrive, tap More.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en>

Markers (adding location information to the link associated with the database):

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions() .position(PERTH) .draggable(true));</pre> |
| <p>[41H] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H] and 28[H], which are incorporated herein by reference in their entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H] and 28[H], which are incorporated herein by reference in their entirety.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1520 224 1587">ZTE</p> <p data-bbox="302 709 337 1220">ZTE Mobile Location Service System</p> <p data-bbox="375 915 402 1041">2004-01-31</p> <p data-bbox="469 1398 496 1562">I. Introduction</p> <p data-bbox="513 373 711 1562">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="727 373 841 1562">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="857 205 927 1587">http://www.cn.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.

Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.


Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.

Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.

Placing a Marker:

<https://developers.google.com/maps/documentation/android-api/marker>

based on queries with GeoTagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Embed a map or share a location</p> <p>On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.</p> <p>ANDROID COMPUTER IPHONE & IPAD</p> <p>Share a map or location</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Search for a place. Or find a place on the map then touch and hold to drop a pin. 3. At the bottom, tap the place's name or address. 4. Tap Share  if you don't see this, tap More  > Share. 5. Select an app. It'll send a link that shows the place in Google Maps. <p>https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en</p> <p>Share your E.T.A</p> <p>After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.</p> <ol style="list-style-type: none"> 1. Open the Google Maps app . 2. Set a driving destination. Learn how to navigate to a place. 3. After you start navigation, tap More  > Share trip progress. 4. Choose a person from the list. 5. Tap Share. 6. Location Sharing will stop when you reach your destination or stop navigating. <ul style="list-style-type: none"> • To stop sharing before you arrive, tap More  > Stop sharing. |
| <p>42[A]. The storage device of claim 41 wherein the operations</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number. See claims 2[A] and 41, which</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| ZTE | ZTE |
|---|--|
| <p>US9408055B2</p> <p>further comprise: presenting another symbol on the interactive map corresponding to a fixed location and associated with a telephone number;</p> | <p>are incorporated by reference in their entirety.</p> |
| <p>[42B] and receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user selection of the other symbol and, based thereon, initiating a telephone call to the telephone number associated with the symbol. See claims 2[B] and 41, which are incorporated by reference in their entirety.</p> |
| <p>43. The storage device of claim 41 wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data comprises a text message, an image, a video, or a command to cause the second devices corresponding to the selected symbols to convert text to speech. See claims 3 and 41, which are incorporated by reference in their entirety.</p> |
| <p>44[A]. The storage device of claim 41 wherein: the SMS messages include an Internet Protocol (IP) address of the first</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein: the SMS messages include an Internet Protocol (IP) address of the first device. See claims 4[A] and 41, which are incorporated by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|--|
| <p>device; [44B] and the IP-based responses include respective IP addresses of the second devices.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] the IP-based responses include respective IP addresses of the second devices. See claims 4[B] and 41, which are incorporated by reference in their entirety.</p> |
| <p>45. The storage device of claim 41 wherein the operations further comprise: transmitting location information including an updated location of the first device to the second device by at least a predetermined distance relative to a previous location of the first device, or a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claims 5 and 41, which are incorporated by reference in their entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: transmitting location information including an updated location of the first device to the second devices based on displacement of the first device by at least a predetermined distance relative to a previous location of the first device, passage of at least a predetermined time interval since transmitting information including a location of the first device, or a combination of the displacement of the first device and the passage of time. See claims 5 and 41, which are incorporated by reference in their entirety.</p> |
| <p>46[A]. The storage</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| ZS9408055B2 | ZTE |
|--|---|
| <p>device of claim 41 wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices; receiving second user selection of one or more of the symbols corresponding to one or more of the second devices;</p> | <p>performance of wherein the operations further comprise: receiving second user selection of one or more of the symbols corresponding to one or more of the second devices. See claims 6[A] and 41, which are incorporated by reference in their entirety.</p> |
| <p>[46B] and receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] receiving user input assigning the one or more second devices corresponding to the second selected one or more symbols to a sub-net. See claims 6[B] and 41, which are incorporated by reference in their entirety.</p> |
| <p>47[A]. The storage device of claim 46 wherein the operations further comprise: receiving user selection of the sub-net;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 7[A], 41, and 46, which are incorporated herein by reference in their entirety.</p> |
| <p>[47B] and establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] establishing a conference among the one or more second devices of the sub-net for sharing voice, text, photographs, or video communications. See claims 7[B], 41, and 46, which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|--|--|
| <p>48. The storage device of claim 41, wherein the first device is a cellular phone or a personal digital assistant (PDA).</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the first device is a cellular phone or a personal digital assistant (PDA). See claims 8 and 41, which are incorporated by reference in their entirety.</p> |
| <p>49. The storage device of claim 41, wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the at least one second device. See claims 9 and 41, which are incorporated by reference in their entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the operations further comprise: identifying second user interaction with the interactive display selecting at least one of the user-selectable symbols corresponding to at least one of the second devices and user interaction with the at least one second device. See claims 9 and 41, which are incorporated by reference in their entirety.</p> |
| <p>50. The storage device of claim 43, wherein the video comprises a video clip or a video transmission.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the video comprises a video clip or a video transmission. See claims 10, 41, and 43, which are incorporated herein by reference in their entirety.</p> |
| <p>51[A]. The storage</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| ZS9408055B2 | ZTE |
|--|---|
| <p>device of claim 46, wherein the operations further comprise: receiving user selection of the sub-net;</p> | <p>performance of wherein the operations further comprise: receiving user selection of the sub-net. See claims 11[A], 41, and 46, which are incorporated herein by reference in their entirety.</p> |
| <p>[51B] and causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] causing the one or more second devices of the sub-net to place a call, make a verbal announcement, convert text to speech, vibrate, or increase sound levels. See claims 11[B], 41, and 46, which are incorporated herein by reference in their entirety.</p> |
| <p>52. The storage device of claim 41, wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to play an audio message announcing an emergency. See claims 39 and 41, which are incorporated by reference in their entirety.</p> |
| <p>53. The storage device of claim 41, wherein the data sent to the one or more second devices causes at least one of the second devices to place a phone call to the first</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of wherein the data sent to the one or more second devices causes at least one of the second devices to place a phone call to the first device. See claims 40 and 41, which are incorporated by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2 device.</p> | <p>ZTE</p> |
| <p>54[P]. A method comprising: performing by a first device:</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of each step of this method as set forth below. See claims 1[P], 28[P], and 41[P], which are incorporated herein by reference in their entirety.</p> <p>The Accused Products meet the claim limitations by providing device-location tracking features such as those features described below. For example, the Accused Products meet the claim limitations because they are pre-installed with Android mobile operating systems containing code for providing device-location tracking features as provided in the claims limitations herein. For example, the Accused Products run applications and/or software that run within the Android mobile operating system and that use components of the Android mobile operating system to provide device-location tracking features. Upon information and belief, in addition to the components and features of the Android mobile operating system itself, the following applications and/or software run within the Android mobile operating system and use components of the Android mobile operating system to provide device-location tracking features: Google Latitude, Google Plus, Google Hangouts (including Allo and Duo), Google Maps, Google Chrome, Google Messages, and Android Messenger.</p> <p><u>Google Maps Share Location</u></p> <p>Share Location is currently included as a standard feature on the Accused Devices operating as a feature of Google Maps. Google Maps is a pre-installed software application in Android OS. The Accused Devices have included the Share Location functionalities since 2009 as part of Google Latitude, which was an opt-in feature for Google Maps on Android OS-based mobile devices, such as the Accused Products. Share Location functionalities were briefly shifted from Latitude for Google Maps to Google Plus and Google Hangouts, until reappearing as a standard feature in Google Maps. Upon information and belief, the Share Location method also uses and/or works in conjunction with functionalities associated with Google Maps, Google Messages, Android Messenger, Location Access, and other features, which are pre-installed on the Accused Products. For the purposes of these contentions, AGIS sets forth Google Maps’ Share Location feature of the Accused Products as representative of this exemplary software. AGIS reserves the right to supplement these contentions to the extent that defendant requires additional information in accordance with P.R. 3-1 and for any other reason.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

See, e.g., <https://techcrunch.com/2017/03/22/google-maps-now-lets-you-share-your-location-with-friends-and-family-for-a-specific-period-of-time/>; <https://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html>; <https://googleblog.blogspot.jp/2013/07/a-new-google-maps-app-for-smartphones.html>; <http://googleplusproject.blogspot.com/2013/05/google-for-android-42.html>; <https://googleblog.blogspot.com/2013/10/google-hangouts-and-photos-save-some.html>

Control within reach, even when your device isn't

One of the biggest security risks you're likely to face is simply losing your phone. To help in these times of need, we're launching Find My Device as part of Google Play Protect. With Find My Device you can locate, ring, lock and erase your Android devices—phones, tablets, and even watches. This feature is built in and enabled on all devices; visit android.com/find or check out the app.

See, e.g., <https://www.blog.google/products/android/google-play-protect/>

Find your device using Android Device Manager

If you've lost a device, you can use Android Device Manager to find its approximate location on a map and when it was last used. When Android Device Manager locates your device, that device will get a notification.

Before you can use Android Device Manager to locate your device: Your device's location access need to be turned on and be signed in to your Google Account. Android Device Manager won't work for devices that are turned off or that don't have a mobile data or Wi-Fi connection.

Tip: If you've linked your phone to Google, you can locate or ring it by searching for **find my phone** on google.com.

<https://support.google.com/pixelphone/answer/6160491>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Link your phone to Google



You can connect your Android phone to Google, which lets you send information from your computer to your phone. For example, you can send directions you searched for on your computer to Google Maps on your phone.

Link your Android phone

Step 1: Update the Google app

1. On your phone, go to the Google app page on the Play Store.
2. Tap **Update**.

Step 2: Turn on Google Now

1. On your phone, open the Google app .
2. At the top left, tap Menu  > **Settings** > **Now cards**.
3. Turn on **Show cards**.
4. Turn on **Show notifications**.

Step 3: Turn on Web & App Activity

1. Visit the **Account History** page.
2. Make sure the switch is on (green).

Step 4: Sign in to your browser

1. On your phone, open the Google app .
2. At the top left, tap the Menu .
3. At the top left, you'll see the email address you use for the Google app.
4. Visit www.google.com  on your computer.
5. If you aren't signed in already, click **Sign in** in the top right corner of the page.
6. Sign in using the Google Account you use for the Google app.

Step 5: Send information to your phone

1. Do one of the searches below, like **note** to **self**, or **send directions** to **my phone**.
2. If a box doesn't pop up with the option to send information to your phone, try refreshing the page. If you just turned on Google Now, it may take a few minutes for the box to show up.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

What you can do once your phone is linked

Find my phone

You can get the current location of your phone if you can't find it.

1. On your computer's browser, search on www.google.com for **find my phone**.
2. If your phone is turned on and connected to the Internet, you'll see your phone's location.
3. If your phone's location is unavailable, you can still make it ring for 5 minutes on full volume by clicking **Ring**. You can stop the ringing from your phone when you find it.

Tip: You can also find your missing phone using the **Android Device manager** which lets you find your device or remotely ring, lock, or erase it.

Send directions to my phone

Once you've looked up directions on your computer, you can send them to your phone so you have them on your trip.

1. On your computer's browser, search on www.google.com for **send directions to my phone**.
2. Enter in your destination.
3. Click **Send directions to your phone**.
4. You'll get a notification on your phone. Tap to navigate to your destination using Google Maps.

Send a note to my phone

1. On your computer's browser, search on www.google.com for **send a note to my phone**.
2. Type your note in the box.
3. Click **Send note to your phone**.
4. You'll get a notification on your phone with your note that you can either save to one of your apps or copy.

Set an alarm

1. On your computer's browser, search on www.google.com for **set an alarm**.
2. Choose the time you want the alarm to go off.
3. Click **Set an alarm on your phone**.
4. An alarm will now be set on your phone's Clock app.

Set a reminder

1. On your computer's browser, search on www.google.com for **set an reminder**.
2. Type what you want to be reminded about, and either when or where you want the reminder to go off.
3. Click **Remind me on my devices**.

<https://support.google.com/websearch/answer/6128427>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your location using Google Maps

You can't share your location in Google+ anymore. If you used to share your location in Google+ and want to keep sharing it, you'll need to share it again in Google Maps.

<https://support.google.com/plus/answer/3302509?hl=en&co=GENIE.Platform%3DAndroid&oco=1>

Location

Turn on location service, your phone determines your approximate location using Wi-Fi and mobile networks. When you select this option, you're asked whether you consent to allowing Google to use your location when providing these services.

- **Mode** – Sets the how your current location information is determined.
- **Recent Location Request** – Displays applications and services that have recently requested your location information.
- **Camera** – Checkmark to tag photos or videos with their locations.
- **Google Location History** – Allows you to view and manage your Google location history.

Accounts & sync

Use the Accounts & sync settings menu to add, remove, and manage your Google and other supported accounts. You also use these settings to control how and whether all applications send, receive, and sync data on their own schedules and whether all applications can synchronize user data automatically.

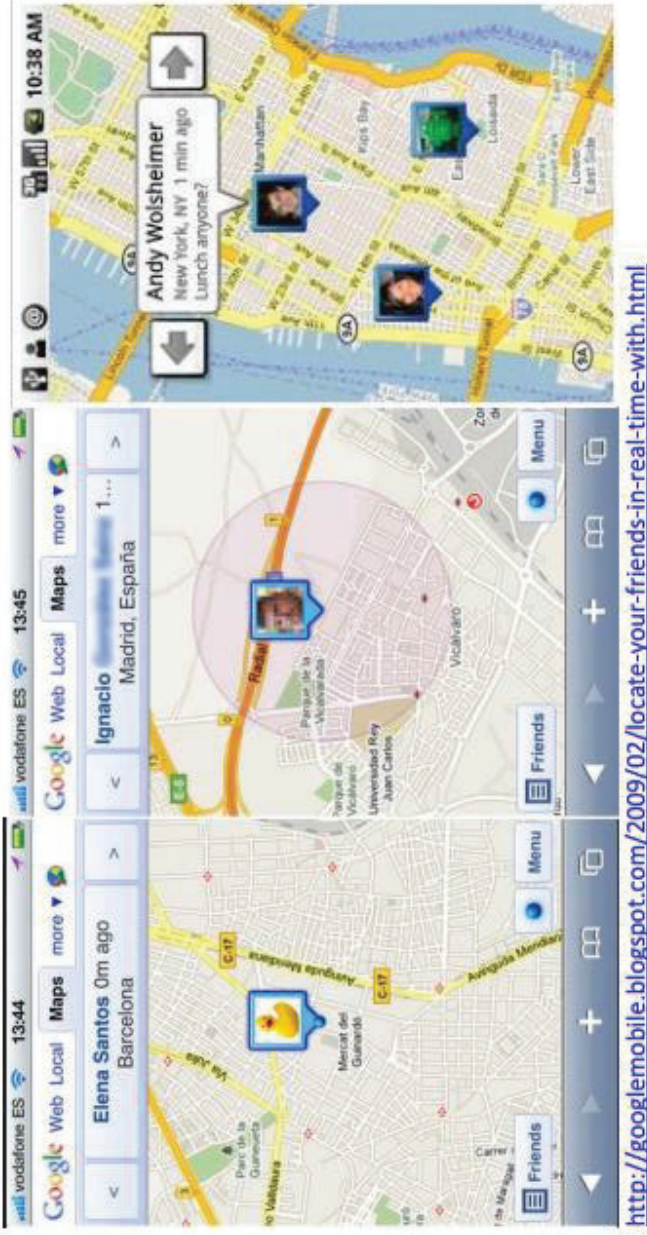
Gmail™, Calendar, and other applications may also have their own settings to control how they synchronize data; see the sections on those applications for details. Touch **Add account** to add new account.

... ..

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<http://googlemobile.blogspot.com/2009/02/locate-your-friends-in-real-time-with.html>

Google's location-sharing feature also appeared in Google+, Google Trust Contacts, and Google Hangouts services until its current integration in Google Maps.

ZTE makes, uses, sells, and otherwise provides this first device by making, using, selling, and importing Android devices such as ZTE phones and ZTE tablets, as well as by providing its servers or using third party servers (e.g., Google servers) for use with Android devices to enable features such as Maps. Below are example ZTE Android devices that perform each step of this method as set forth below.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE phones
 mob.org * Mobile phones and smartphones catalogue

Sort by: Popularity Date Price

HTC
 LG
 Samsung
 Motorola
 Fly
 Sony-Ericsson
 Apple
 Nokia
 Wobisado
 Vertu
 BenQ-Siemens
 Sagem
 Alcatel
 Philips

All brands

ZTE Rapido
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 4.5 inch.
 Android 4.1.2

ZTE Grand S3
 Mobile phone
 2015 year
 Touchscreen: 1080 x 1920
 5.5 inch.
 Android 4.4

ZTE Geek 2
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE V5s
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.4

ZTE Grand X Quad
 Mobile phone
 2015 year
 Touchscreen: 720 x 1280
 5 inch.
 Android 4.1

ZTE ZMAX
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.4.2

ZTE Grand Memo Lite
 Mobile phone
 2014 year
 Touchscreen: 720 x 1280
 5.7 inch.
 Android 4.3

https://mob.org/phone/zte/page_3/sort_date_down/

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|--|
| <p>US9408055B2</p> | <p>ZTE</p>  <p>https://www.zteusa.com/products/tablets</p> |
| <p>54[A] obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of: obtaining contact information of a plurality of second devices, wherein the contact information comprises respective telephone numbers of the second devices. See claims 1[A], 28[A], and 41[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products include a contacts app to access contact information for second users using respective second devices.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Phone calls

How to make calls



There are many ways to make a call with your phone, and they're all easy to do.

Calling from the dialer



1. From the home screen, tap .
2. Enter the phone number with the on-screen keypad. Tap  to delete wrong digits.
3. Tap  to place the call.

Tip: To make international calls, press and hold **0+** to enter the "+".

Calling from your contacts

1. From the home screen, tap .
2. Swipe your finger up or down to scroll through the contacts list and tap  next to the contact you want to call.

Tips:

- You can search for a contact by tapping  and entering the contact name.
- You can also access your contacts by tapping  > **Favorites**.



People

You can add contacts on your phone and synchronize them with the contacts in your Google account or other accounts that support contact syncing.

To see your contacts, tap  on the home screen. From there, you can tap the tabs on the top to quickly switch to **Groups**, or **Favorites**.

Importing and exporting contacts

You can import/export contacts from/to your SIM card, phone storage, or microSDHC card. This is especially useful when you need to transfer contacts between different devices. You can also quickly share your contacts using Bluetooth, Email, Messaging, etc.



Importing contacts from the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Manage SIM card contacts**.
2. If you have added contact accounts other than the phone, select an account in which to save the contacts.
3. Tap the contacts you want to import one by one, or tap  > **Import all**.

Importing contacts from a microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Import from phone storage**.
2. If you have added contact accounts other than the phone, select an

Calling from your call history

1. From the home screen, tap  > **Call log**.
2. Tap  next to the number you want to call.

Calling from a text message

If a text message contains a phone number that you want to call, you can make the call while viewing the text message.

1. From the home screen, tap .
2. Tap the conversation and then find the message that contains the phone number you need.
3. Tap the number and then tap .

Using speed dial

Press and hold the **1-9** key from the dialer to call the corresponding speed dial number.

The **1** key is reserved for your voicemail.

Assigning a speed dial key

1. From the home screen, tap .
2. In the Phone tab, tap  > **Speed dial setting**.
3. Tap a speed dial key and tap **Set speed dial contact**.
4. Select a contact from the contact list.

Note: You can also set your speed dial keys from the dialer. Press and hold the **2-9** key and then tap **OK**.


3. Select the vCard file(s) in the microSDHC card or the phone storage and tap **OK**.

Note: If the microSDHC card is not installed in the phone, you can import vCard file(s) in the phone storage.

Exporting contacts to the SIM card

1. From the Contacts screen, tap  > **Import/export** > **Export to SIM card**.
2. Select the contacts you want to export and then tap .

Exporting contacts to the microSDHC card or phone storage

1. From the Contacts screen, tap  > **Import/export** > **Export to phone storage**.
2. The phone will prompt you with the name of the vCard file and the directory in which the file will be saved. Tap **OK** to create the file.

Note: If the microSDHC card is not installed in the phone, you can export vCard file(s) into the phone storage.

Sharing contact information

1. From the Contacts screen, tap  > **Import/export** > **Share visible contacts**.
2. Choose how to share the contacts. Options depend on the applications and services installed.

Creating a contact

1. From the Contacts screen, tap  to add a new contact.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

2. Tap the account field near the top of the screen to choose where to save the contact. If a sync account is selected, the contacts will be synced automatically with your account online.
3. Enter the contact name, phone numbers, email addresses, and other information.
4. Tap **DONE** to save the contact.

Adding a contact to Favorites

You can add the contacts you use frequently to Favorites so that you can find them quickly.

1. From the Contacts screen, tap the contact you want to add to **Favorites**.
2. Tap  next to the contact's name.

Searching for a contact


1. Tap the search field above the contacts list.
2. Enter the contact name you want to search for. Matching contacts will be listed.

Joining contacts


As your phone synchronizes with multiple online accounts, you may see duplicate entries for the same contact. You can merge all the separate information of a contact into one entry in the Contacts list.

1. From the home screen, tap .
2. Tap a contact to display the contact's details.
3. Tap  > **Edit** >  > **Join**.

Entering text

You can enter text using the onscreen keyboard. Some apps open it automatically. In others, you open it by tapping where you want to type. You can also enter text by speaking with the Google voice typing feature. Tap  to hide the onscreen keyboard.

Changing input methods

1. When you use the onscreen keyboard to enter text, the icon  appears on the notification bar.
2. Open the notification panel and tap **Choose input method**.
3. Select an input method you need.

Google keyboard

The Google keyboard provides a layout similar to a desktop computer keyboard. Turn the phone sideways and the keyboard will change from portrait to landscape.

To use the landscape keyboard, tap the **Auto-rotate screen** check box in  > **System settings** > **Accessibility**.

Note: The landscape keyboard is not supported in all applications.





4. Tap the contact whose information you want to join with the first entry.
 5. Tap **DONE**.
- The information from the second contact is added to the first, and the second contact is no longer displayed in the contacts list. You can repeat these steps to join another contact to the main contact.

Separating contact information

If contact information from different sources was joined in error, you can separate the information back into individual contacts on your phone.

1. From the home screen, tap .
2. Tap a contact you have merged and want to separate.
3. Tap  > **Edit** >  > **Separate**.
4. Tap **OK** to confirm.

Creating a new group

1. From the Contacts screen, tap .
2. Tap .
3. If you have added contact accounts other than the phone, choose an account for the new group.
4. Enter the group name and tap **DONE**.
5. Tap  and select the contacts you wish to be the group members.
6. Tap .

To send messages to the group members, you can tap a group and then tap  > **Send group message**.



- Tap the alphabetic keys to enter letters. Press and hold the keys to enter associated accented letters or numbers. For example, to enter É, press and hold  and the available accented letters and number 3 appear. Then slide your finger to choose **E**.
- Tap  to use uppercase or lowercase letters. This key also changes to indicate the current case you are using:  for lowercase,  for uppercase, and  when locked in uppercase. Press and hold or double-tap  to lock the keyboard in uppercase.
- Tap  to delete any text you have entered.
- Tap  to select numbers and symbols. You can then tap  to access more.
- Tap  to enter miniature icons
- Tap  to use Google's networked voice input.
- Press and hold  to change the input language or the Google keyboard settings.

In another example, the Accused products run Android Messages and Google Hangouts which both access

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p>ZTE</p> <p>contact information for second users using respective second devices.</p> <h2>Contacts Provider</h2> <p>The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p>This guide describes the following:</p> <ul style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p>https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contacts_id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/provider-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|--|-------------|---|-----------|---|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-795

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 }</pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,            // 1
57             Data.LOOKUP_KEY,           // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,     // 4
60             Data.CONTACT_STATUS,      // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS      = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

B-798

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

B-800

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




US9408055B2

ZTE

Send & receive text messages in Android Messages





You can send and receive text messages with friends and contacts on Android Messages.

Start a conversation

1. Open the Android Messages app .
2. Tap Compose .
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next .

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329

See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](#) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
2. Tap Menu  > **Create label**.
3. Enter a label name and tap **OK**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
- **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER **ANDROID** IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

Contact someone

You can call, email, or send text messages to your contacts.

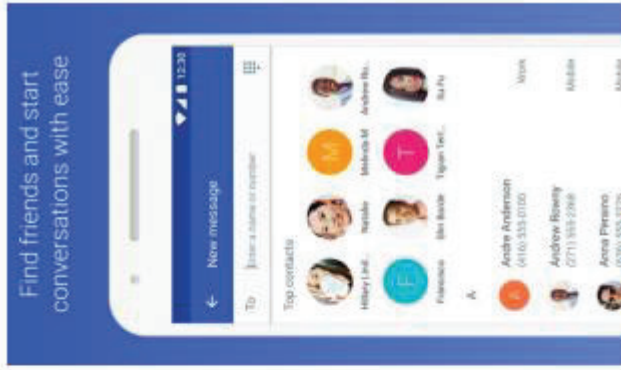
1. Open your device's Contacts app.
2. Tap a contact in the list.
3. Choose an option:
 - Call
 - Email
 - New message

https://support.google.com/nexus/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Start a conversation

1. Open the Android Messages app
2. Tap Compose
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329
<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap Add + > New Conversation.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap Send.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

1. Open the Hangouts app.
2. At the bottom, tap Add + > New conversation > New group.
3. Enter and select the names, phone numbers, or email addresses of people in your group.
4. Tap Done.

https://support.google.com/hangouts/answer/3111943?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products





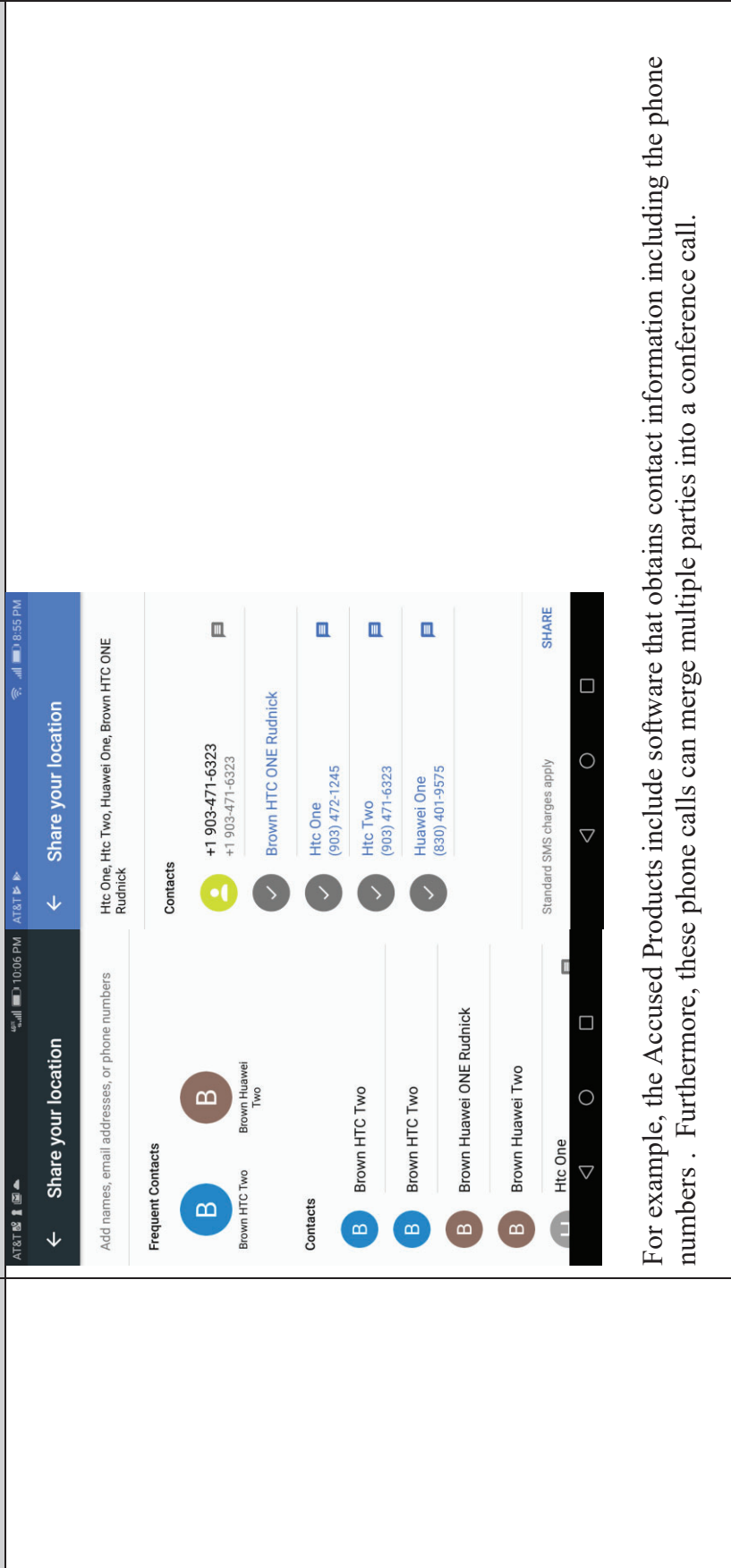
| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <h3 data-bbox="235 1186 276 1554">Contact someone</h3> <p data-bbox="300 903 332 1554">You can call, email, or send text messages to your contacts.</p> <ol data-bbox="357 630 625 1554" style="list-style-type: none"><li data-bbox="357 1113 397 1554">1. Open your device's Contacts app .<li data-bbox="406 1260 446 1554">2. Tap a contact in the list.<li data-bbox="454 1323 487 1554">3. Choose an option:<ul data-bbox="495 1291 625 1522" style="list-style-type: none"><li data-bbox="495 1407 535 1522">• Call <li data-bbox="544 1386 584 1522">• Email <li data-bbox="592 1291 625 1522">• New message  <p data-bbox="641 630 673 1585">https://support.google.com/hexus/answer/611873?hl=en&ref_topic=6118711</p> <p data-bbox="706 1071 738 1585"><u>Exemplary Google Maps Screenshots:</u></p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



For example, the Accused Products include software that obtains contact information including the phone numbers . Furthermore, these phone calls can merge multiple parties into a conference call.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

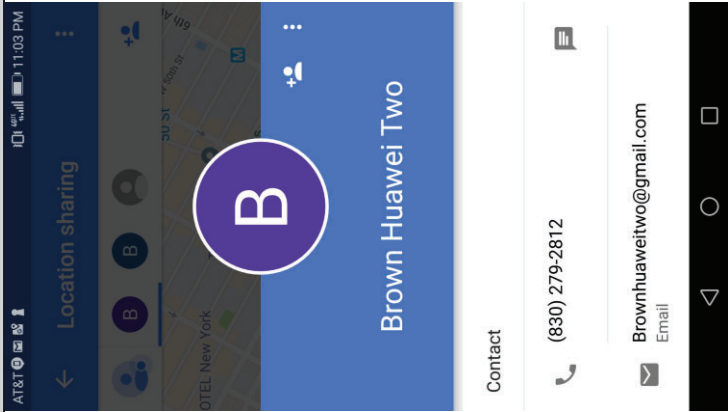
| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p>  |
| <p>[54B] facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, to the second devices, the second devices, respective Short</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of facilitating initiation of Internet Protocol (IP) based communication between the first device and the respective second devices by using the respective telephone numbers to send, to the second devices, respective Short Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device. See claims 1[B], 28[B], and 41[B], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products utilize SMS-based messages to initiate IP communication between participants of Maps location sharing. For example, both Android Messages and Hangouts, in conjunction with Maps, utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> <p>Message Service (SMS) messages including a telephone number of the first device and information usable by the respective second device to send IP-based communication to the first device;</p> | <p style="text-align: center;">ZTE</p> <p style="color: green;">Using Wi-Fi Direct</p> <p>Wi-Fi Direct allows Wi-Fi devices to connect to each other without the need for wireless access points (hotspots).</p> <p>Connecting to another device via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. From the home screen, tap [Settings] > System settings > Wi-Fi. 2. If Wi-Fi is off, slide the Wi-Fi switch to the On position. 3. Tap [Wi-Fi] > Wi-Fi Direct. Your phone will search for other devices enabled with Wi-Fi Direct connections. 4. Tap a device name under Peer Devices to connect with it. The other device will receive a Wi-Fi Direct connection prompt and need to accept the request for connection. Both devices may need to enter a common PIN. If prompted, tap Connect. 5. Once connected, the device is displayed as "Connected." <p>Sending data via Wi-Fi Direct</p> <ol style="list-style-type: none"> 1. Open the appropriate application and select the file or item you want to share. 2. Select the option for sharing via Wi-Fi Direct. The method may vary by application and data type. 3. Tap a device the phone has connected with or wait for it to search for new devices and tap one of them. <p>Receiving data via Wi-Fi Direct</p> <p>When an attempt to transfer data via Wi-Fi Direct is received, you can see a notification in the status bar. Tap Accept to start receiving the data. Received files are stored automatically in a dedicated folder (WiFiShare, for instance) in the phone storage or microSDHC directory. You can access them with the File Manager app.</p> |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Messaging

You can use Messaging to exchange text messages (SMS) and multimedia messages (MMS).

Message box



Instead of an inbox and outbox, your phone organizes all messages you sent and received into one box, where messages exchanged with the same number are grouped into one message thread on the Messaging screen. You can tap a thread to see the conversation you have had with someone.

Message threads are sorted in chronological order with the latest one on top.

Opening the messaging screen

From the home screen, tap .

The Messaging screen opens, where you can create a new message, search for messages, or open an ongoing message thread.

- Tap  to write a new text or multimedia message.
- Tap  to search for a message with keywords.
- Tap an existing message thread to open the conversation you've had with a certain number.

Sending a message


1. From the messaging screen, tap  at the bottom.
2. Add recipients by one of the following ways:
 - Tap the **To** field and manually enter the recipient's number or the contact name. If the phone presents a few suggestions, tap the one you want to add.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products









| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <ul style="list-style-type: none"> Select recipients from your contacts by tapping . Tap the Type message field and enter the content of your text message. If you want to send a multimedia message, tap the paper clip icon  to attach a file or a slideshow to the message. Tap  to send your message. <p>Notes:</p> <ul style="list-style-type: none"> You can also include email addresses as recipients for multimedia messages. Do not add any attachment if you want to send a text message. Otherwise you may be charged for a multimedia message. <p>Replying to a message</p> <p>Messages you receive are appended to existing threads of the same number. If the new message comes from a new number, a new thread is created.</p> <ol style="list-style-type: none"> From the Messaging screen, tap the thread that has the message you want to reply to. Type your reply in the text box at the bottom. You can tap the icon  if you want to reply with an MMS. Tap  to send your message. <p>Forwarding a message</p> <ol style="list-style-type: none"> From the Messaging screen, tap the thread that has the message you want to forward. Press and hold the message. Tap Forward in the menu that opens. |
| | <ol style="list-style-type: none"> Enter a recipient for the message and edit the content if you want. Tap  to send your message. <p>Changing message settings</p> <p>The phone's message settings are pre-configured for you to use immediately. To change them, tap  > Settings from the Messaging screen.</p> <p>Storage settings:</p> <ul style="list-style-type: none"> Delete old messages: Delete old messages as limits are reached. Text message limit: Set the maximum number of text messages allowed in a single thread. Multimedia message limit: Set the maximum number of multimedia messages allowed in a single thread. <p>Text (SMS) message settings:</p> <ul style="list-style-type: none"> Manage SIM card messages: Manage the messages stored on your SIM card. Service Center: Enables you to view and edit the service center number. <p>Multimedia (MMS) message settings:</p> <ul style="list-style-type: none"> Auto-retrieve: Automatically download multimedia messages. <p>Display settings:</p> <ul style="list-style-type: none"> Bubble and background: Set the appearance of the messaging bubbles and background. <p>Notification settings:</p> <ul style="list-style-type: none"> Notifications: Show message notifications in the status bar. Choose ringtone: Choose a ringtone for your incoming messages. |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Email

From the home screen, tap . You can receive and send emails from your webmail or other accounts using POP3 or IMAP, or access your Exchange ActiveSync account for your corporate email needs.

Setting up the first email account

1. When you open **Email** for the first time, enter your email address and password.
2. Tap **Next** to let the phone retrieve the network parameters automatically.
Note: You can also enter these details manually by tapping **Manual setup** or when automatic setup fails.
3. Follow the on-screen instructions to finish the setup.

Your phone will show the inbox of the email account and start to download email messages.

Checking your emails

Your phone can automatically check for new emails at the interval you set when setting up the account.

You can also check new emails manually by tapping  in any of the email account's boxes. Tap **Load more messages** at the bottom of the email list to download earlier messages.

Responding to an email

You can reply to or forward a message that you receive. You can also delete messages and manage them in other ways.

Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.

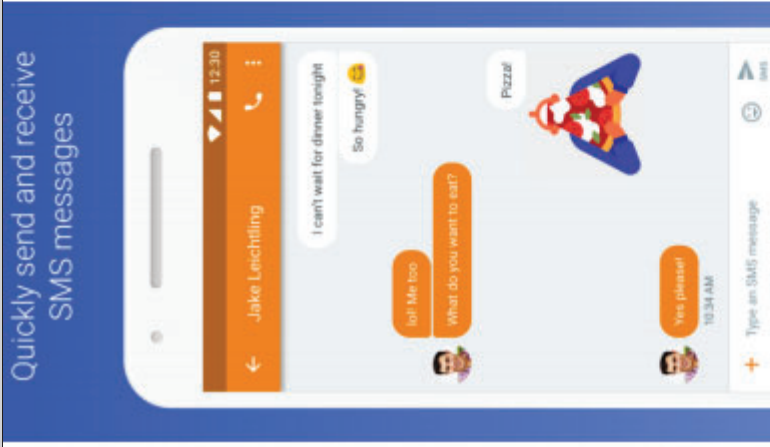
• **Enhanced features:** On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more.

<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Start a Hangout

You can send and receive messages with one person or multiple people.

COMPUTER **ANDROID** IPHONE & IPAD

Start a conversation

1. On your Android phone or tablet, open the Hangouts app.
2. At the bottom right, tap **Add** > **New Conversation**.
3. Type and select a person's name.
4. Enter your message. You can also add emojis, photos, your location, or a sticker.
5. Tap **Send**.

https://support.google.com/hangouts/answer/3115553?hl=en&ref_topic=6386410&co=GENIE.Platform%3DAndroid&oco=0

- Connect your Google Voice account for phone calling, SMS texting, and voicemail integration.
- Keep in touch with contacts across Android, iOS, and the web, and sync chats across all your devices.
- Message contacts anytime, even if they're offline.

<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>

1. Open the Hangouts app.
2. At the bottom right, tap **Add**.
3. Choose **New SMS**.
4. Type the name or phone number. If you're traveling, use the "+" sign and country code when texting.
5. Tap the number or contact.
6. Tap **Continue**.
7. Type your message and tap **Send**.

<https://support.google.com/hangouts/answer/3441321?hl=en>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-------------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><u>Google Maps Share Location</u></p> <p>Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). The sign-in process takes place within the Google Maps software on the Accused Product or by navigating to maps.google.com within the Google Chrome browser on the Accused Product. Alternatively, the sign-in process may partially or completely take place using credentials already provided when the user associates a Google Account with the Accused Product, e.g., during initial setup of the Accused Product. Subject to discovery, one or more additional or substitute identifiers may correspond to the group. The sign-in process involves a user entering its Google Account and additional authentication data on the interface of the Accused Product and sending a message containing the Google Account and additional authentication data over a network to members of a group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group. Further regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, Google Duo, Google Allo, and Google Chrome, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). Subject to discovery, additional identifiers may be assigned or used to correspond to the group. The request may be an invitation or message that associates a Google Account with one or more Google Accounts for the purposes of sharing locations within the group. The group comprises the multiple identifiers, individuals, profiles, and/or devices associated with the group.</p> <p><u>Exemplary Support for Google Maps:</u></p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > **Add People** .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > **Add People** .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap **More**  **> Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap **More**  **> Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap **Menu**  **> Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon **> More**  **> Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap **More** .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD


Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%63DAn droid&oco=1

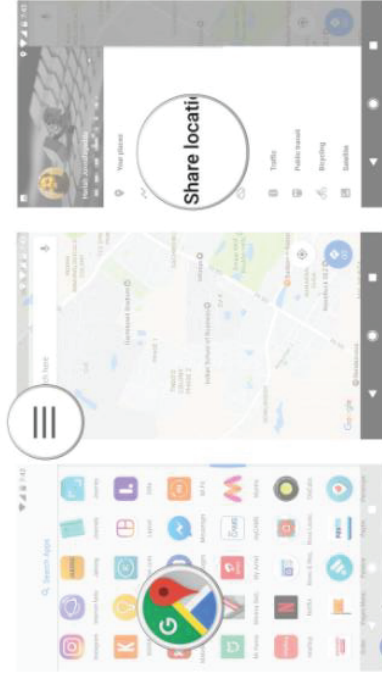
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

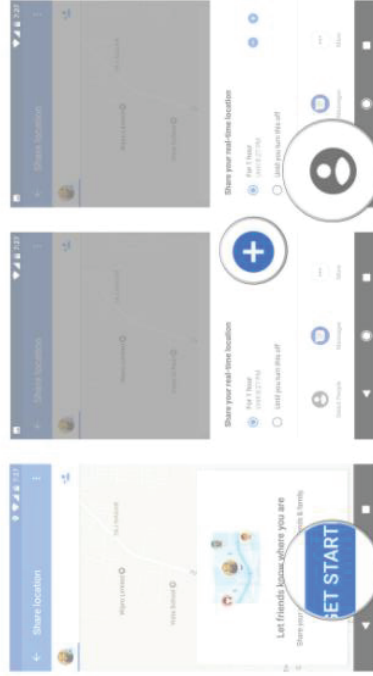
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



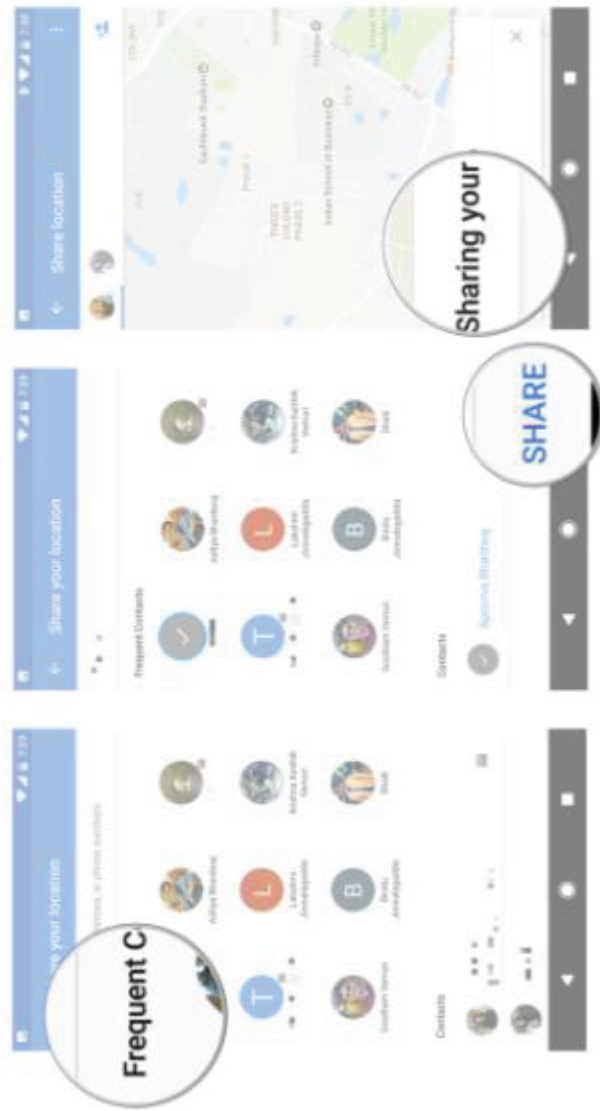
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

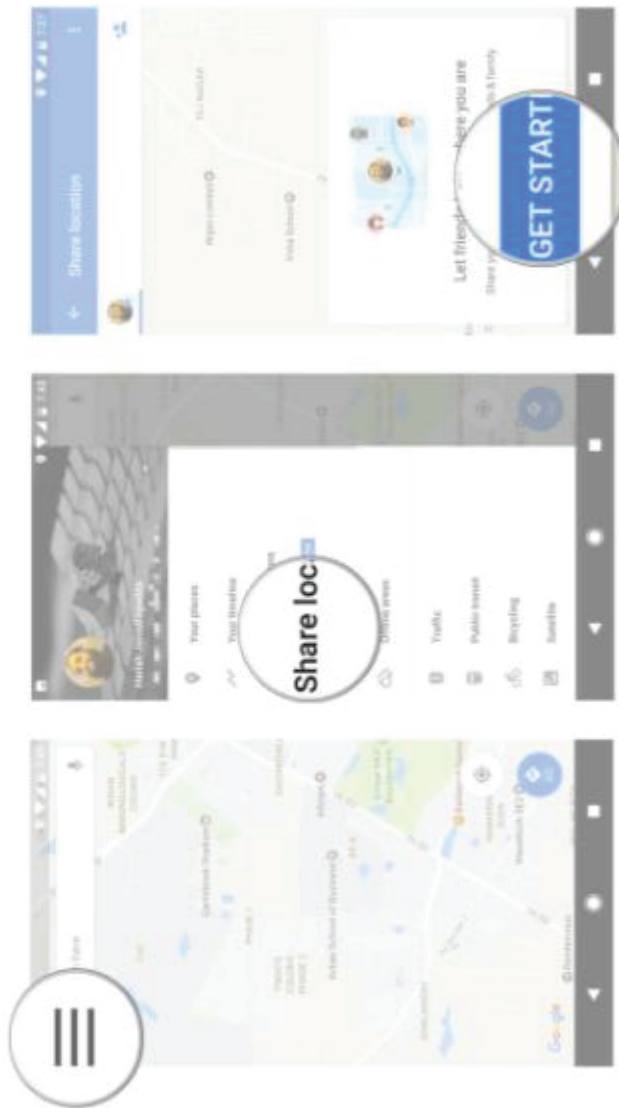
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



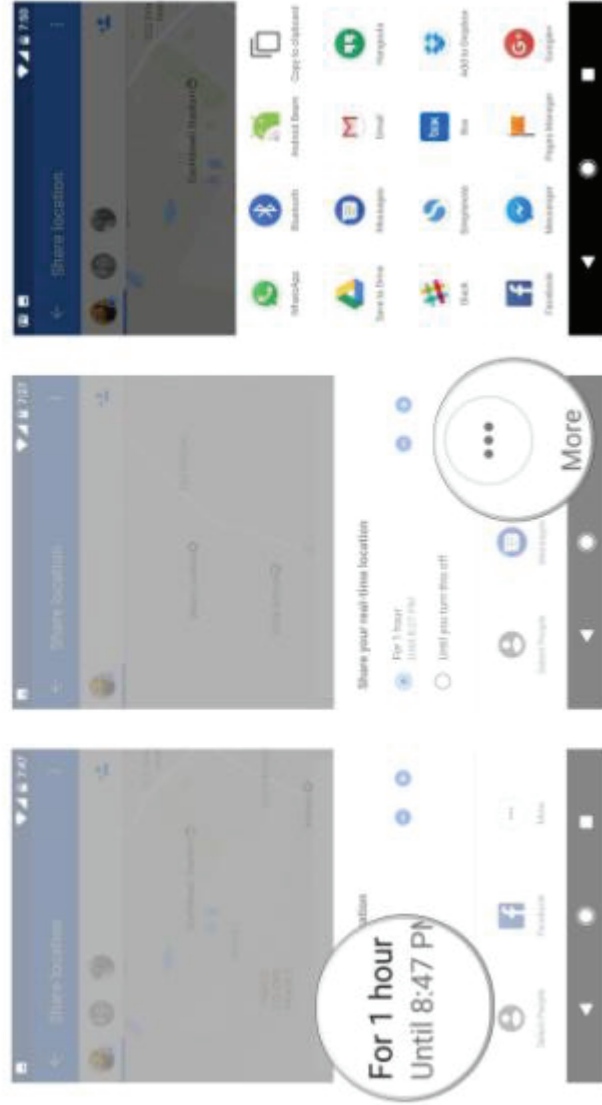
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

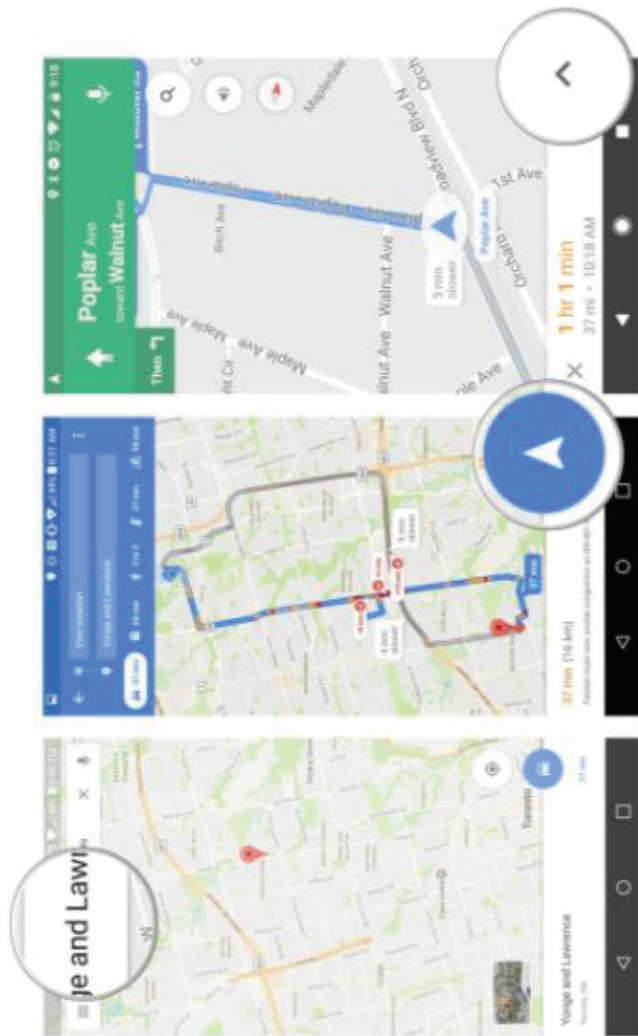
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



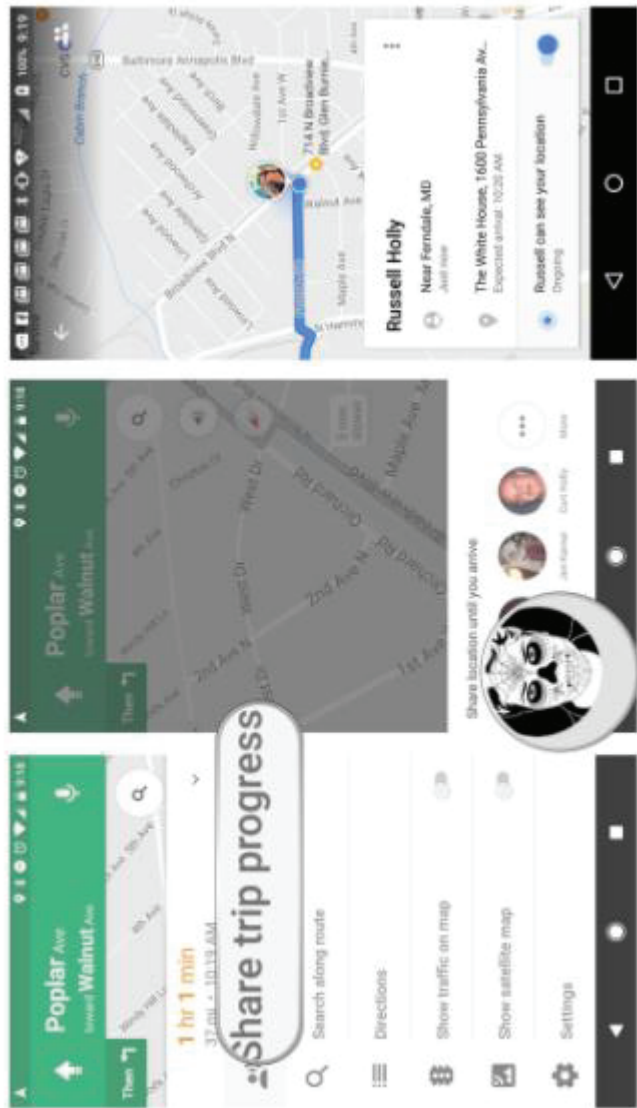
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



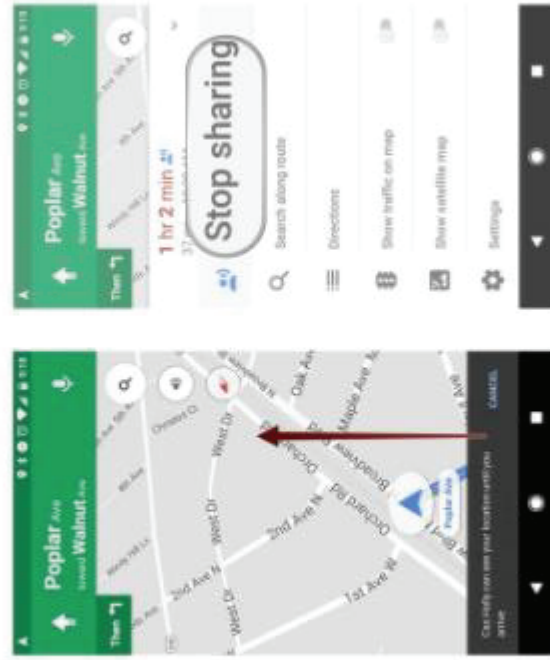
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>





As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE





See your contacts

1. Open your device's Contacts app .
 2. Tap Menu .
- **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.
- Tip:** If you have multiple contacts with the same information, the information will be grouped into one contact.
- **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



Label your contacts

You can group contacts together using labels.

1. Open your device's Contacts app .
 2. Tap Menu  > **Create label**.
 3. Enter a label name and tap **Ok**.
- **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

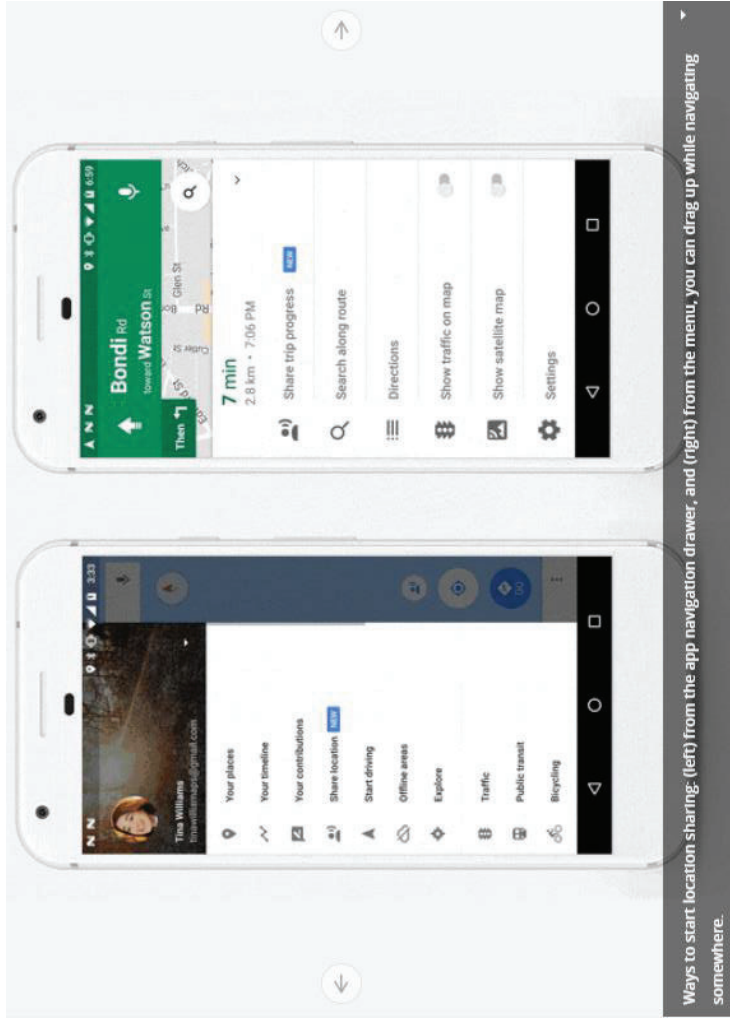
1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

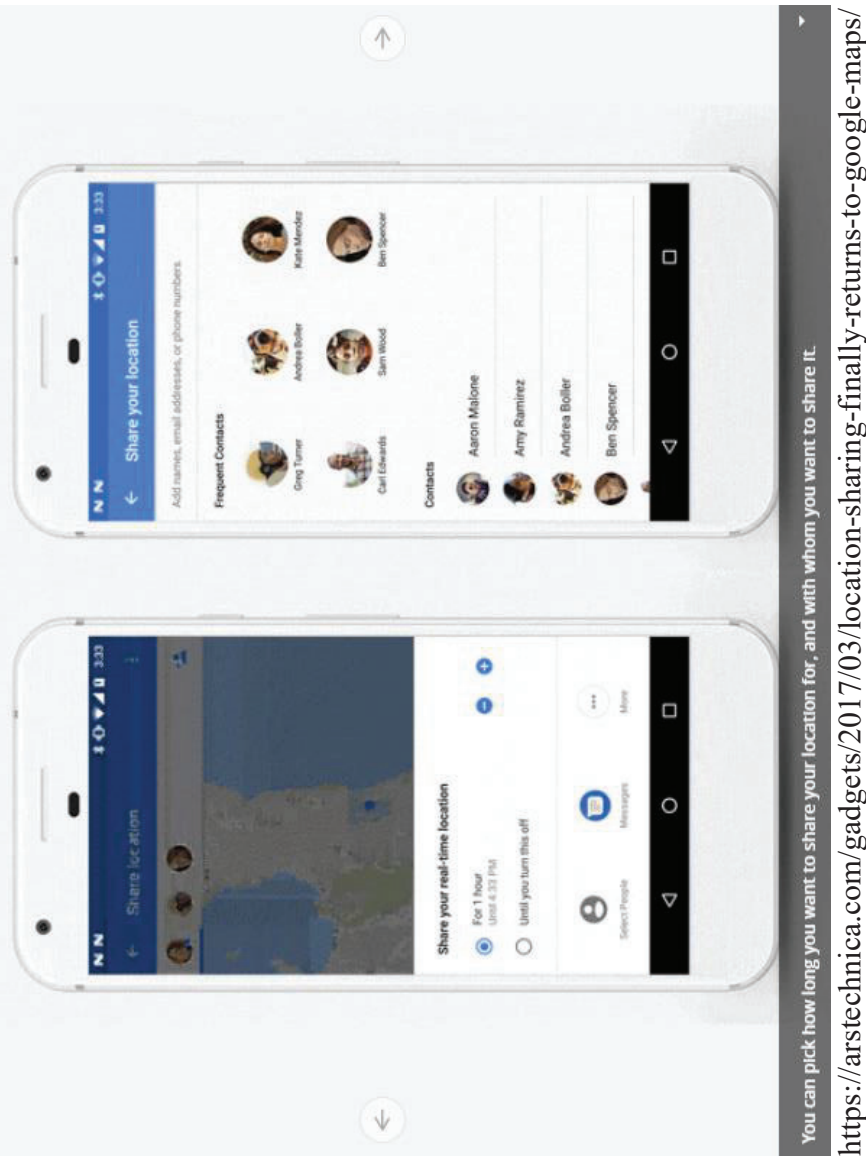
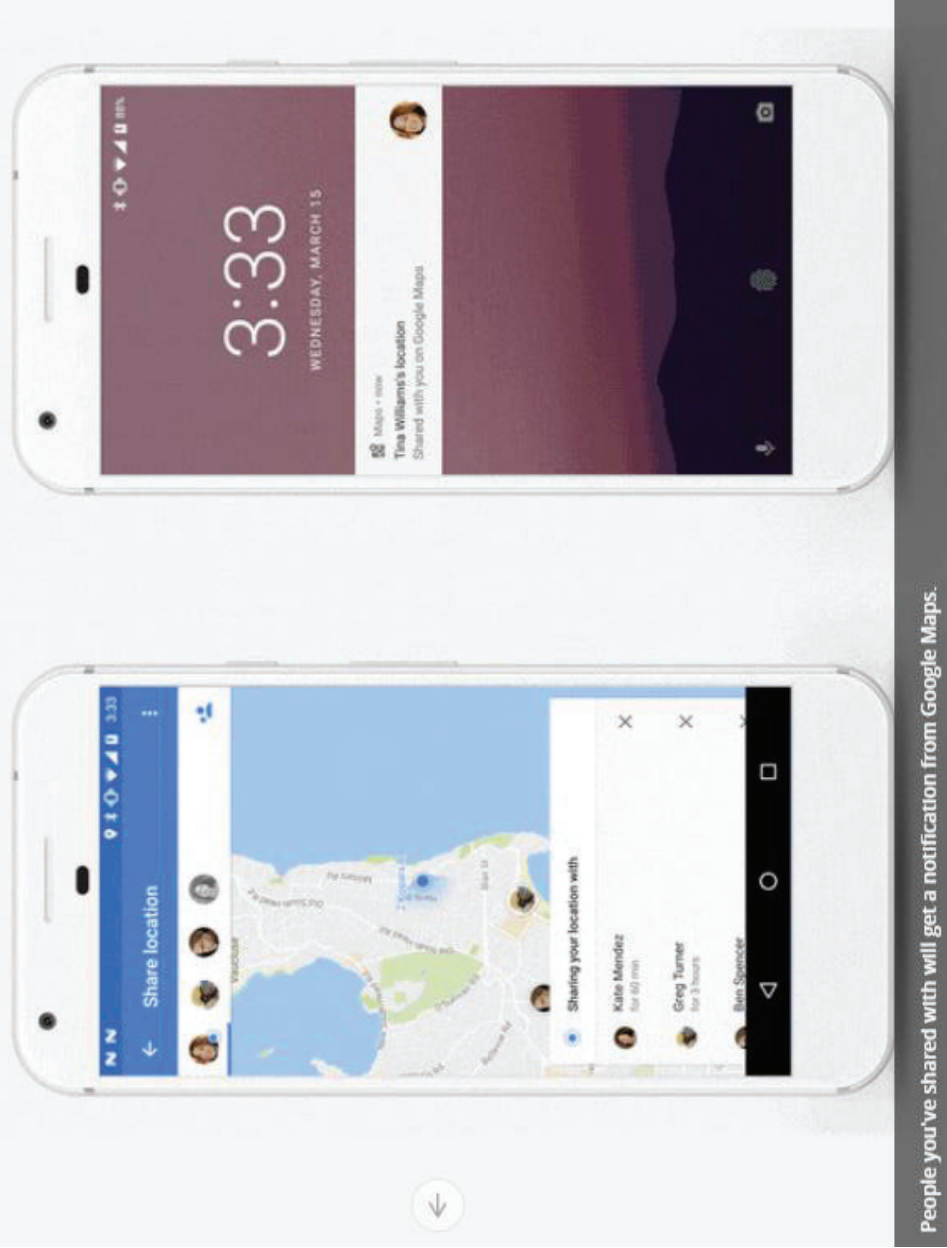


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



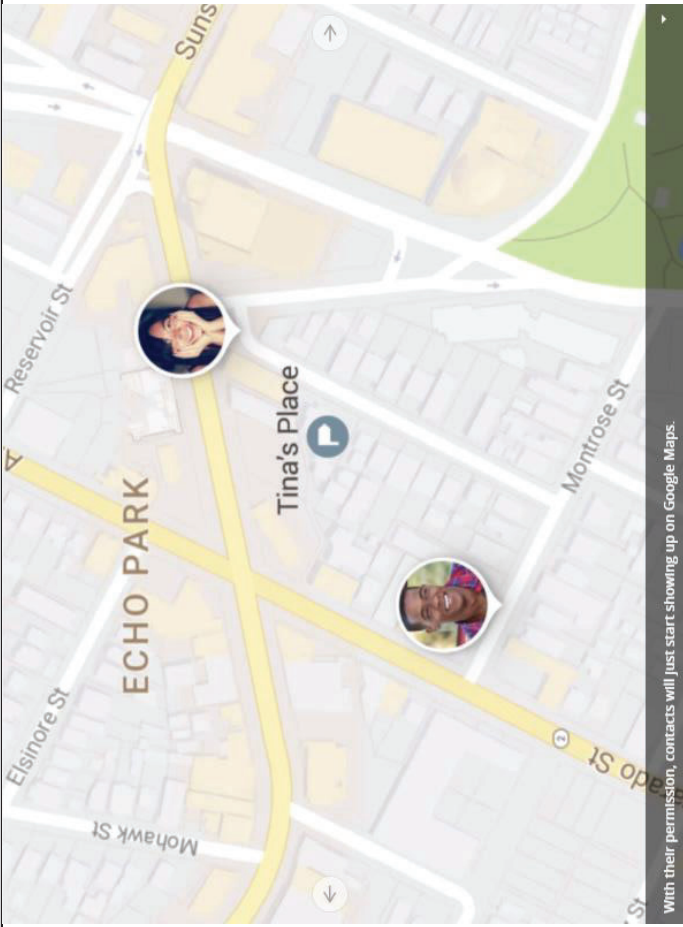
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



With their permission, contacts will just start showing up on Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>The first screenshot shows a mobile phone interface with a 'Share your location?' dialog box. The dialog contains the text: 'These settings will be turned on for lgtestdevice123@gmail.com.' and 'Location Sharing lets you share your real-time location with people you choose.' There are 'CANCEL' and 'TURN ON' buttons at the bottom of the dialog. The background shows a map with 'Gershwin Theatre' and '14%' battery.</p> <p>The second screenshot shows a mobile phone interface with a menu. The menu items are: 'Your places', 'Your timeline', 'Your contributions', 'Start driving', 'Offline areas', 'Explore', and 'Traffic'. The background shows a map with 'feller, Cen', 'heater', 'wn Hall', 'rk MetLif', and 'Original Com'. There is a 'TODAY' label at the bottom right.</p> <p>The third screenshot shows a mobile phone interface with a 'Share your location?' dialog box. The dialog contains the text: 'These settings will be turned on for lgtestdevice246@gmail.com.' and 'Location Sharing lets you share your real-time location with people you choose.' There are 'CANCEL' and 'TURN ON' buttons at the bottom of the dialog. The background shows a map with 'Location Sha...' and 'lgtestdevice246...'.</p> |
| | <p>Exemplary Source Code: The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE): AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <h2 data-bbox="235 1081 292 1575">Contacts Provider</h2> <p data-bbox="324 630 592 1575">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="625 1260 649 1575">This guide describes the following:</p> <ul data-bbox="673 724 852 1575" style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p data-bbox="860 609 885 1585">https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the `ContactsContract.Data` table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the `ContactsContract.RawContacts` table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the `ContactsContract.Contacts` table represents an aggregate of one or more `RawContacts` presumably describing the same person. When data in or associated with the `RawContacts` table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- `ContactsContract.Groups`, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- `ContactsContract.StatusUpdates`, which contains social status updates including IM availability.
- `ContactsContract.AggregationExceptions`, which is used for manual aggregation and disaggregation of raw contacts
- `ContactsContract.Settings`, which contains visibility and sync settings for accounts and groups.
- `ContactsContract.SyncState`, which contains free-form data maintained on behalf of sync adapters
- `ContactsContract.PhoneLookup`, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a `ContactsContract.Data` row that is linked to the raw contact's `_id` value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for `emilyd@gmail.com` (the raw contact row for Thomas Higginson associated with the Google account `emilyd@gmail.com`) has a home email address of `thigg@gmail.com` and a work email address of `thomas.higginson@gmail.com`, the `Contacts Provider` stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the `ContactsContract.Data` table. To help manage this, the `ContactsContract.Data` table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | | |
|---|-------------|--|-----------|---|--|
| Task | Action | Data | MIME type | Notes | |
| Pick a contact from a list | ACTION_PICK | One of: <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply. Call <code>startActivityForResult()</code> , which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details. | |
| https://developer.android.com/guide/topics/providers/contacts-provider.html | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p data-bbox="1040 533 1104 1589">https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 458 private void configureFragments(boolean fromRequest) { 459 if (fromRequest) { 460 ContactListFilter filter = null; 461 int actionCode = mRequest.getActionCode(); 462 boolean searchMode = mRequest.isSearchMode(); 463 final int tabToOpen; 464 switch (actionCode) { 465 case ContactsRequest.ACTION_ALL_CONTACTS: 466 filter = ContactListFilter.createFilterWithType(467 ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS); 468 tabToOpen = TabState.ALL; 469 break; 470 case ContactsRequest.ACTION_CONTACTS_WITH_PHONES: 471 filter = ContactListFilter.createFilterWithType(472 ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY); 473 tabToOpen = TabState.ALL; 474 break; 475 case ContactsRequest.ACTION_FREQUENT: 476 case ContactsRequest.ACTION_STREQUENT: 477 case ContactsRequest.ACTION_STARRED: 478 tabToOpen = TabState.FAVORITES; 479 break; 480 case ContactsRequest.ACTION_VIEW_CONTACT: 481 tabToOpen = TabState.ALL; 482 break; 483 default: 484 tabToOpen = -1; 485 break; 486 } 487 } </pre> <p> https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java </p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 } https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java</pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

35 public final class GroupMemberLoader extends CursorLoader {
36
37     public static class GroupEditorQuery {
38         private static final String[] PROJECTION = new String[] {
39             Data.CONTACT_ID,           // 0
40             Data.RAW_CONTACT_ID,      // 1
41             Data.DISPLAY_NAME_PRIMARY, // 2
42             Data.PHOTO_URI,           // 3
43             Data.LOOKUP_KEY,          // 4
44         };
45
46         public static final int CONTACT_ID           = 0;
47         public static final int RAW_CONTACT_ID      = 1;
48         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2;
49         public static final int CONTACT_PHOTO_URI   = 3;
50         public static final int CONTACT_LOOKUP_KEY  = 4;
51     }
52
53     public static class GroupDetailQuery {
54         private static final String[] PROJECTION = new String[] {
55             Data.CONTACT_ID,           // 0
56             Data.PHOTO_URI,            // 1
57             Data.LOOKUP_KEY,           // 2
58             Data.DISPLAY_NAME_PRIMARY, // 3
59             Data.CONTACT_PRESENCE,     // 4
60             Data.CONTACT_STATUS,       // 5
61         };
62
63         public static final int CONTACT_ID           = 0;
64         public static final int CONTACT_PHOTO_URI   = 1;
65         public static final int CONTACT_LOOKUP_KEY   = 2;
66         public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3;
67         public static final int CONTACT_PRESENCE_STATUS = 4;
68         public static final int CONTACT_STATUS       = 5;
69     }
70
71     private final long mGroupId;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/-/nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100  *   it is the message URL
101  * @param pdu For POST (sending) only, the PDU to send
102  * @param method HTTP method, POST for sending and GET for downloading
103  * @param isProxySet Is there a proxy for the MMSC
104  * @param proxyHost The proxy host
105  * @param proxyPort The proxy port
106  * @param mmsConfig The MMS config to use
107  * @param userAgent The user agent header value
108  * @param uaProfUrl The UA Prof URL header value
109  * @return The HTTP response body
110  * @throws MmsHttpException For any failures
111  */
112  public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113  String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114  throws MmsHttpException {
115  Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116  + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117  + ", PDU size=" + (pdu != null ? pdu.length : 0));
118  checkMethod(method);
119  HttpURLConnection connection = null;
120  try {
121  Proxy proxy = Proxy.NO_PROXY;
122  if (isProxySet) {
123  proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124  }
125  final URL url = new URL(urlString);
126  // Now get the connection
127  connection = (HttpURLConnection) url.openConnection(proxy);
128  connection.setDoInput(true);
129  connection.setConnectTimeout(
130  mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131  CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpParams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[54C] receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices; information of the respective second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving respective IP-based responses to the SMS messages, wherein the IP-based responses to the SMS messages include location information of the respective second devices. See claims 1[C], 28[C], and 41[C], which are incorporated herein by reference in their entirety.</p> <p>For example, the ZTE accused devices running Maps are configured to receive IP-based communications from the respective second devices that include location information of the second devices.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

ZTE Mobile Location Service System

2004-01-31




I. Introduction

Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.

The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.

http://www.en.zte.com.cn/endata/magazine/zte technologies/2003 year/no 14/articles/200401/t20040131_161273.html

Send your location

1. Open the Android Messages app .
2. Open or start a conversation.
3. Tap Attach .
4. Tap Location on .
5. To send your location, tap Send .

https://support.google.com/pixelphone/answer/6159880?hl=en&ref_topic=6211804

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

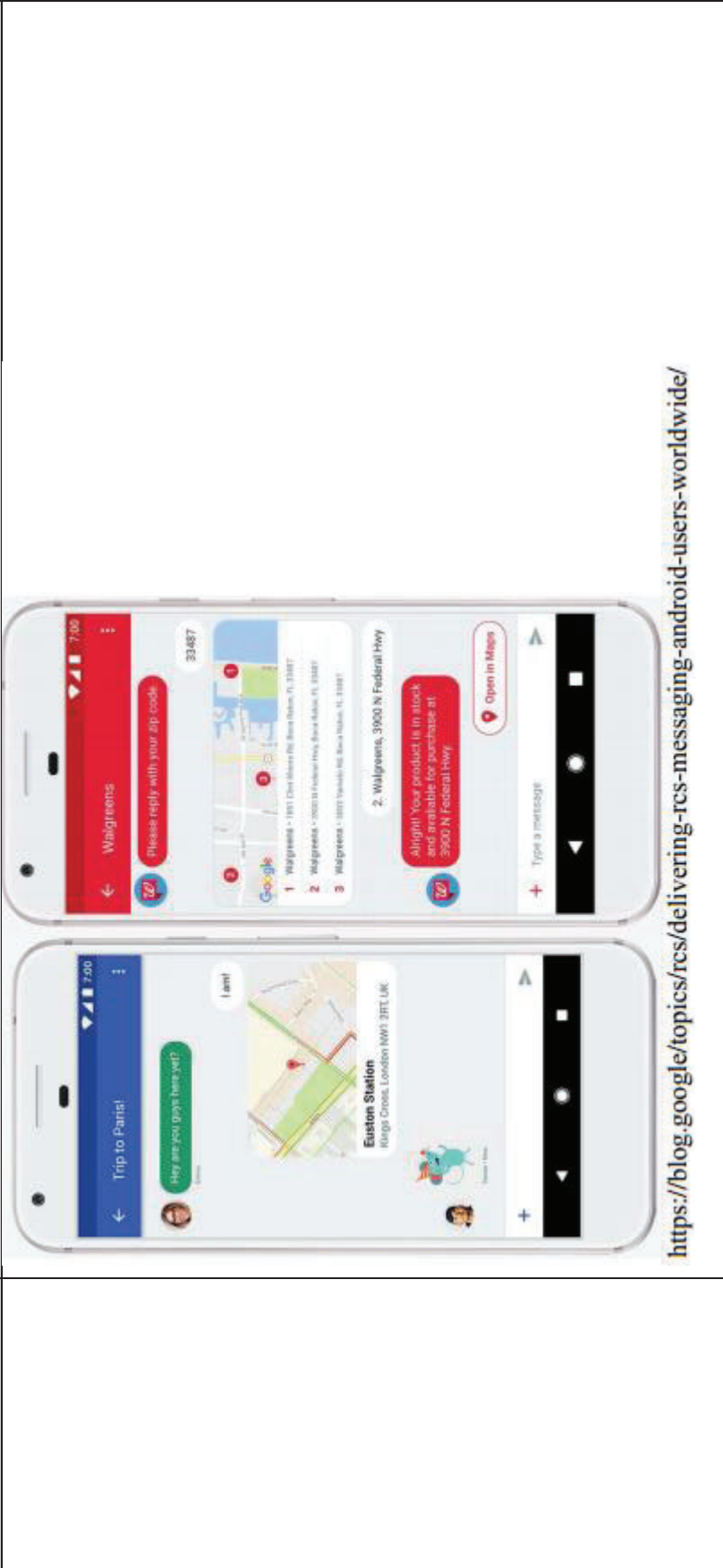


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share a location or place

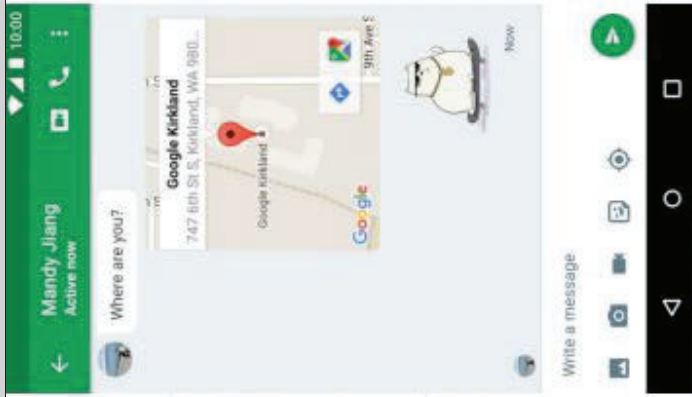
Share your location

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location.
- 4. Tap Select this location > Select.

Share a place

- 1. On your Android phone or tablet, open the Hangouts app.
- 2. Open a conversation.
- 3. Tap Location > Search Q.
- 4. Type in a location or address.
- 5. Tap Select.

https://support.google.com/hangouts/answer/3115410?visit_id=l-636271867303650973-2491837168&rd=1&co=GENIE.Platform%3DAndroid&oco=1
<https://play.google.com/store/apps/details?id=com.google.android.talk&hl=en>



Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products require a user to join the corresponding network by signing-in to the device with an identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user enables sharing to one or more contacts (of respective devices) and the one or more contacts enable sharing their location to the user of the first device, the user of the first device receives the locations of the one or more contacts.

The first device's participation in the group is based on receiving the message from the second device, i.e. a message indicating that the second device is sharing its location.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="196 1520 224 1587">ZTE</p> <p data-bbox="269 205 412 1587">By participating in the Maps location sharing functionality, the device sends location information to a server (e.g., a network server provided by an ISP such as AT&T and/or a server running Google's services). The device also receives location information from the server indicating the location of other devices that are sharing location information via Maps.</p> <p data-bbox="453 197 813 1587"><u>Further regarding Google Maps</u>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products alternatively require a user to send a request containing the identifier (e.g., Google Account). When the signed-in user enables Google Maps Share Location and Location Services on the first device, the user shares its location and the first device's location is sent to a server. When the user sends a message to another contact through Google Maps, Google Messages, and/or another means from within the Google Maps application, the message including location information are sent to a server before transmission to the intended contact. When one or more contacts enable sharing their location to the user of the first device, or alternatively send a message containing location information, or alternatively accept a request to share their location with the first user, the user of the first device receives the locations of the one or more contacts.</p> <p data-bbox="818 1073 846 1587"><u>Exemplary Support for Google Maps:</u></p> |
|--------------------|---|





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD

If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap More .
4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2




ZTE

Create a list of places



In Google Maps, you can create a list of places, like your favorite places or places you want to visit.

COMPUTER **ANDROID** IPHONE & IPAD


Make a new list

1. On your Android phone or tablet, open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. In the bottom right, tap Add .
4. Enter a name and description.
5. Tap **Save**.

Save a place to a list

1. Open the Google Maps app .
2. Search for a place or tap it on the map.
3. At the bottom, tap the place's name or address.
4. Tap **Save**.
5. Choose a list. To create a new list, tap **New list** .

See your lists

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap More  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list


If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > More  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAndroid&oco=1

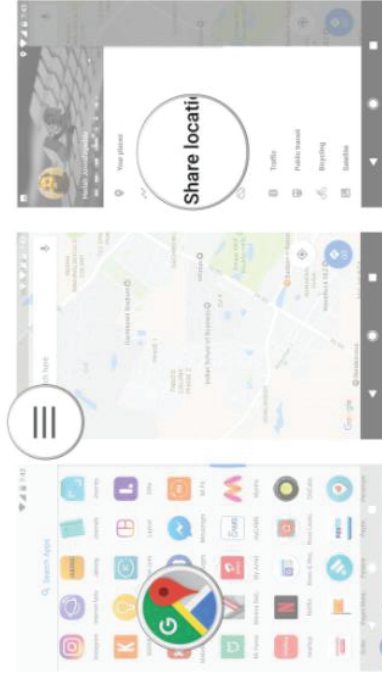
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

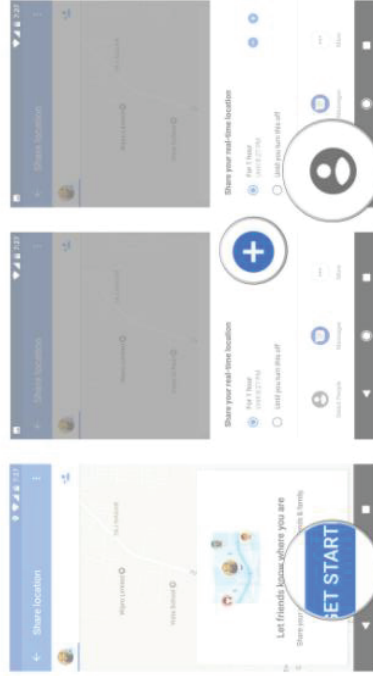
ZTE

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
6. Tap Select People.



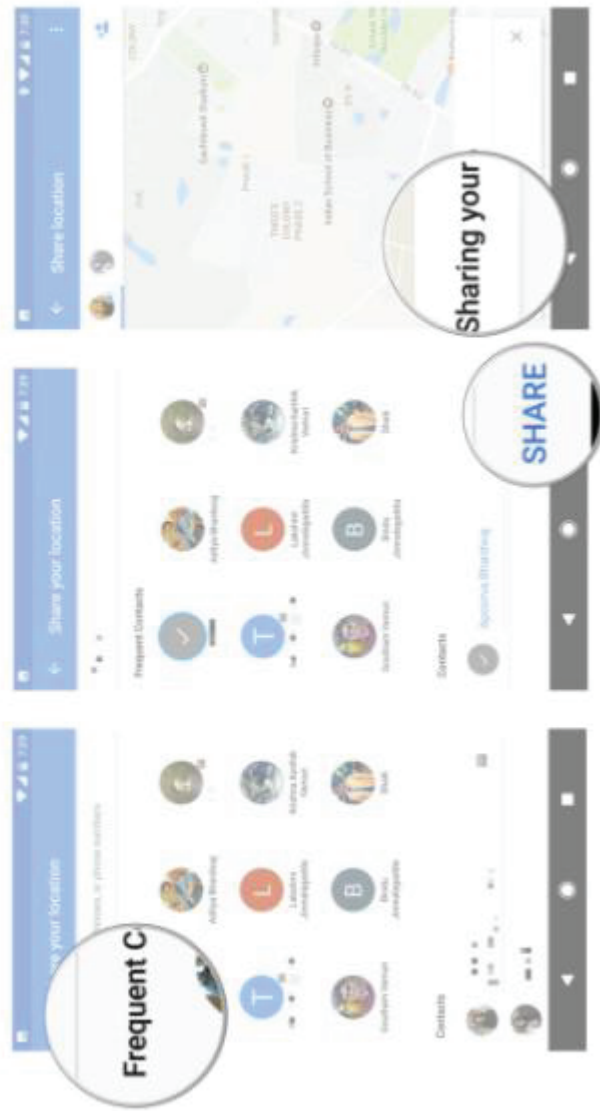
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

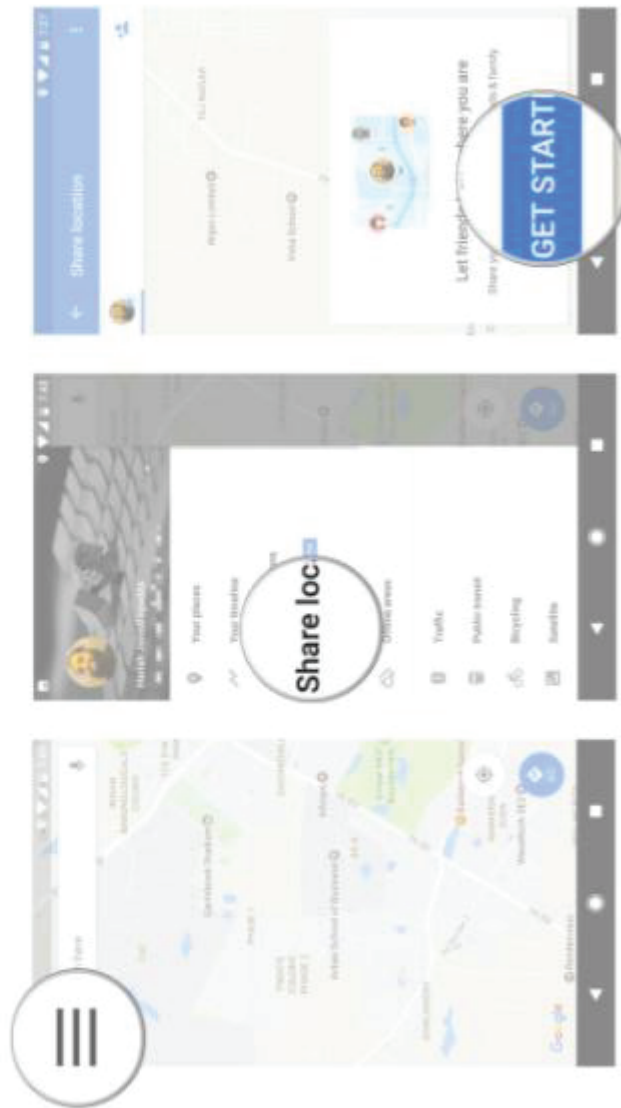
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

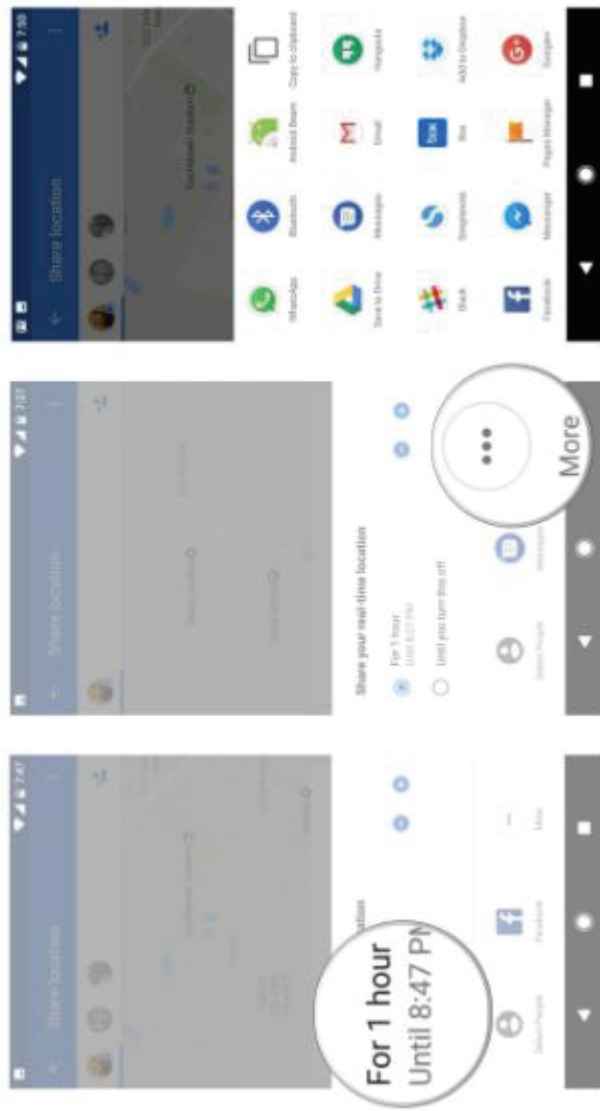
US9408055B2

ZTE

4. Select the amount of time you want to share your location.

5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

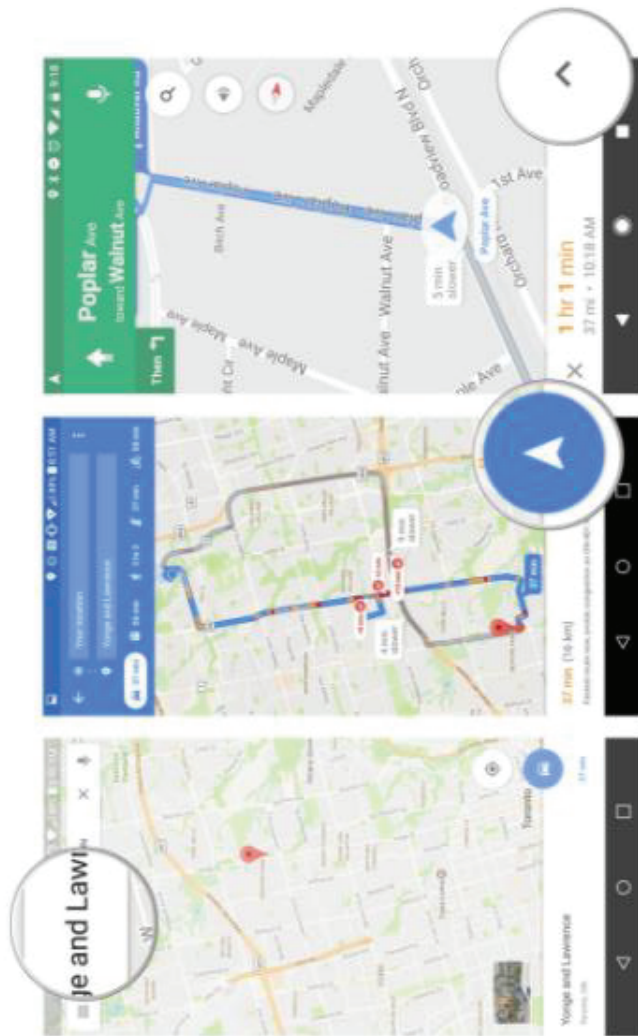
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



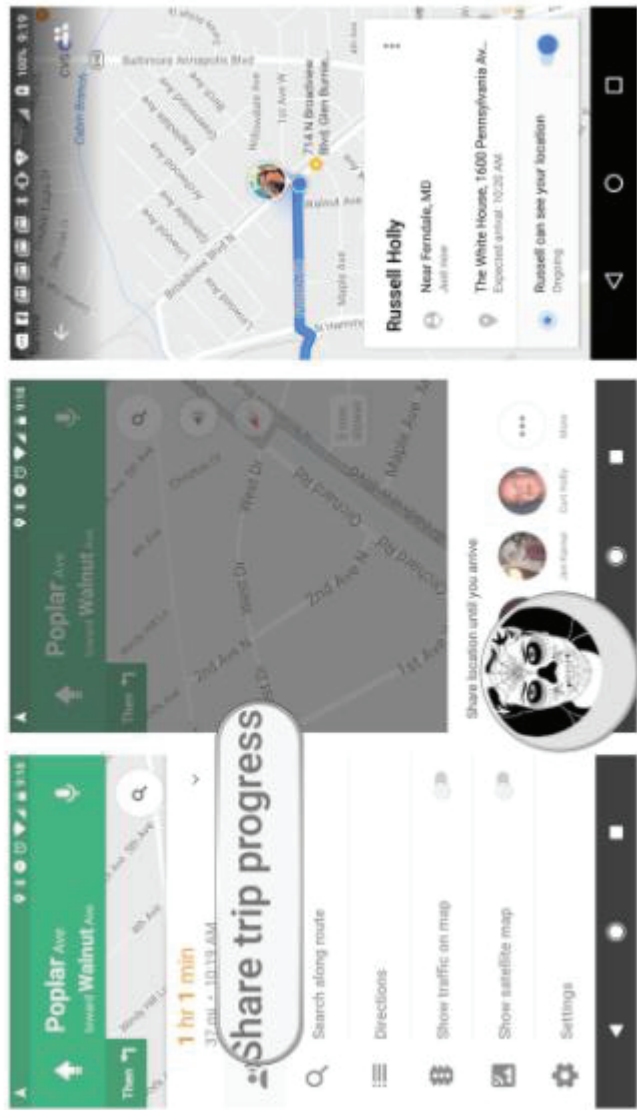
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



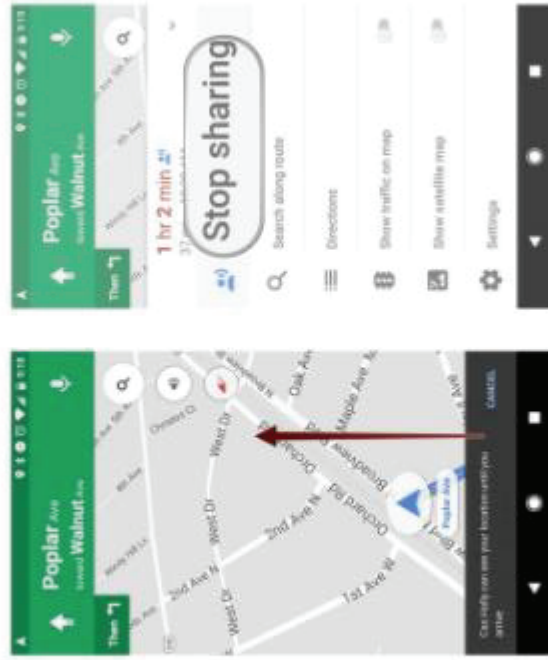
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!



Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>
As shown below, a group may also be defined within Google Contacts.


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2


ZTE

See your contacts

1. Open your device's Contacts app .
2. Tap Menu .

 - **See contacts by label:** Choose a label from the list.
 - **See contacts for another account:** Tap Down arrow  > pick an account.
 - **See the contacts for all your accounts:** Choose **All contacts**.


Tip: If you have multiple contacts with the same information, the information will be grouped into one contact.


 - **See your Google Account contacts on the web:** Go to [Google Contacts](https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711) .

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Label your contacts



You can group contacts together using labels.

1. Open your device's Contacts app .
2. Tap Menu  > **Create label**.
3. Enter a label name and tap **Ok**.

 - **Add one contact to a label:** Tap Add contact  > choose a contact.
 - **Add multiple contacts to a label:** Tap Add contact  > touch and hold a contact > tap the other contacts > tap **Add**.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

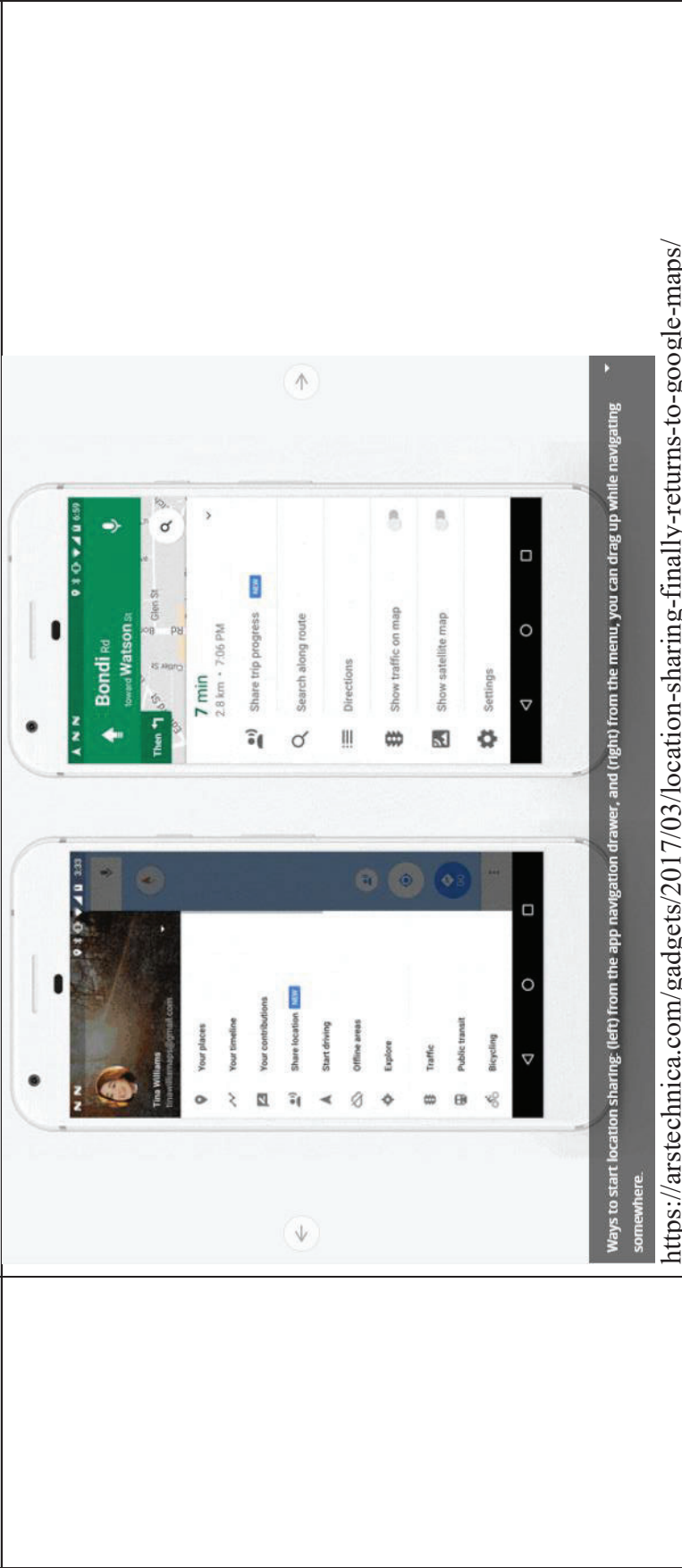


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

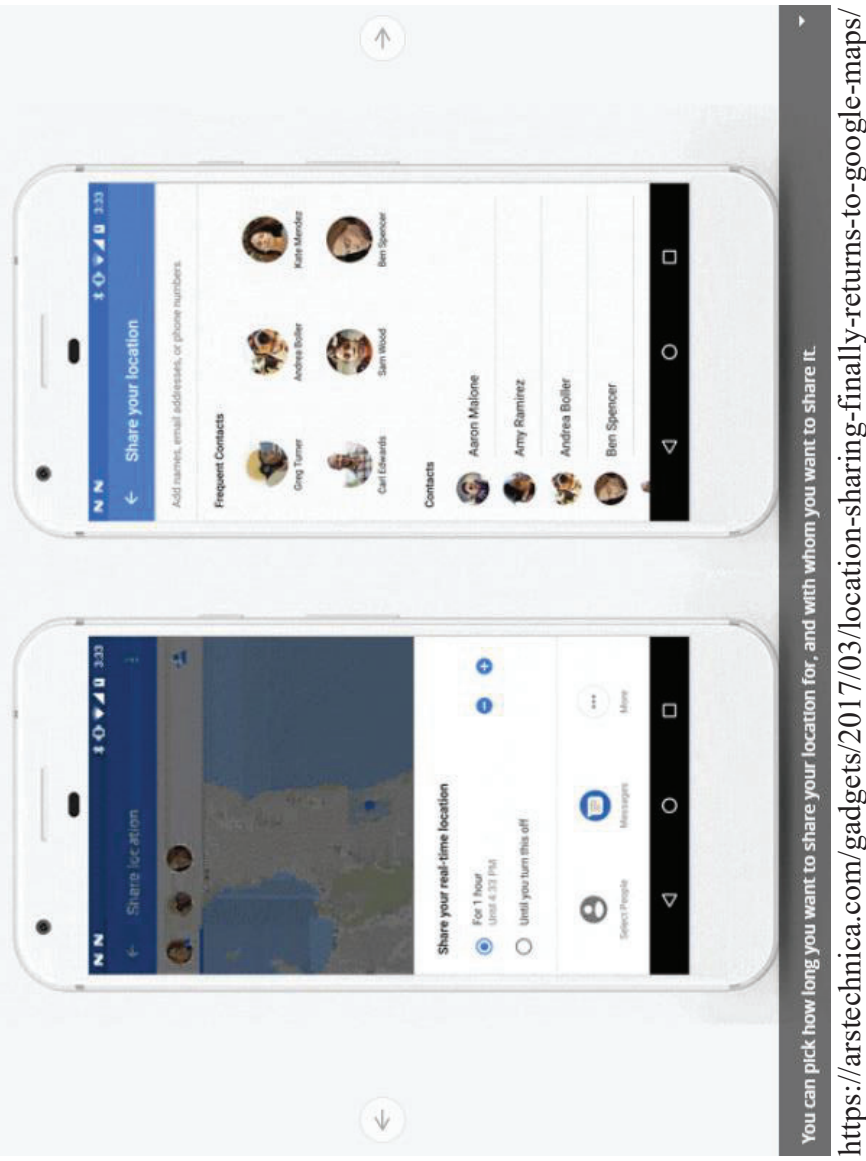
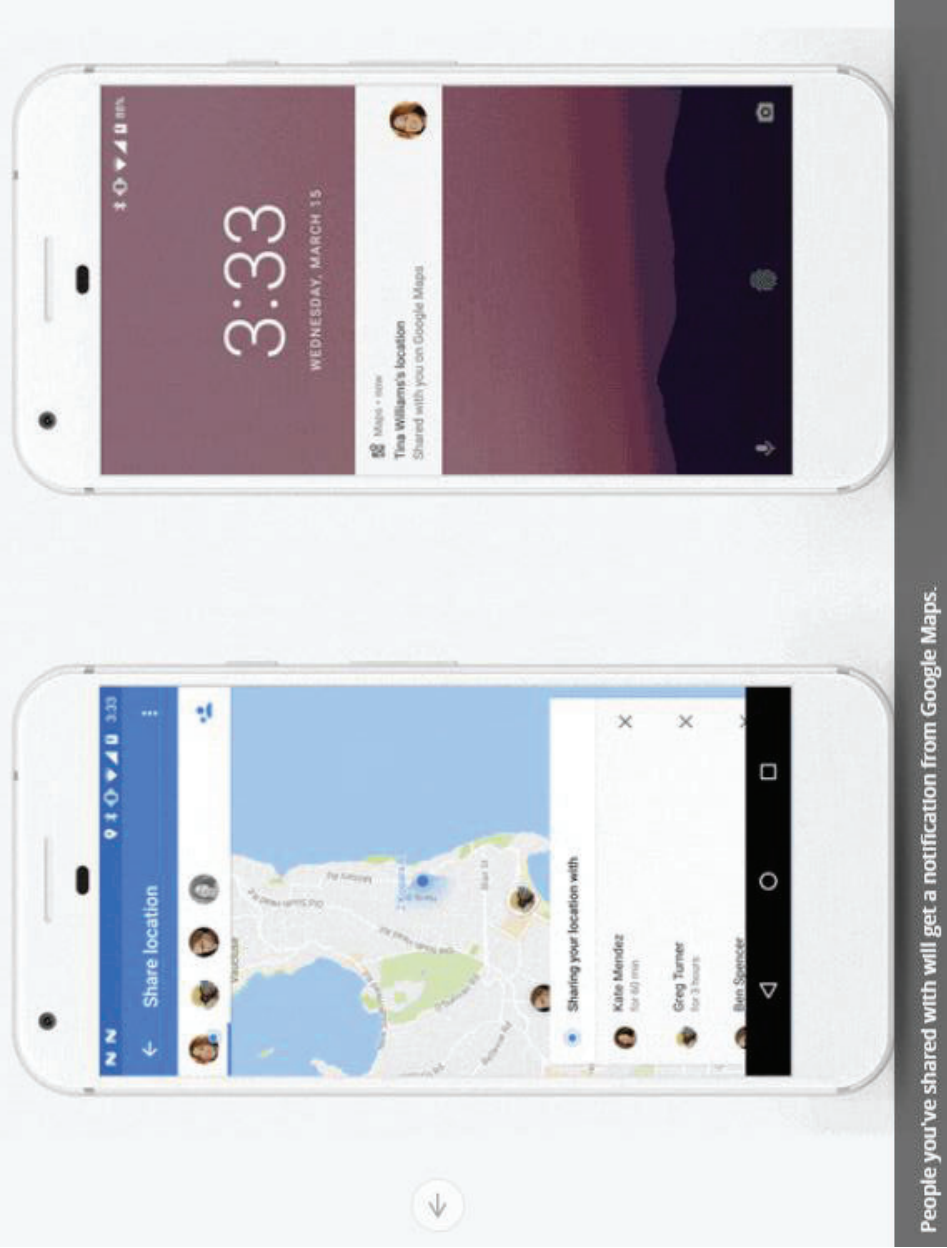


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



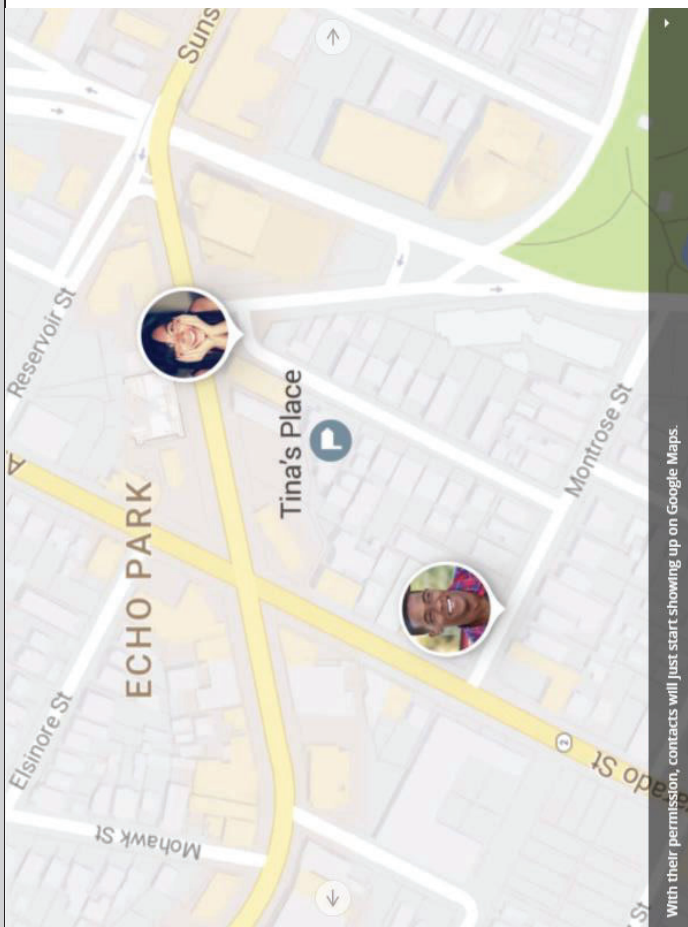
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

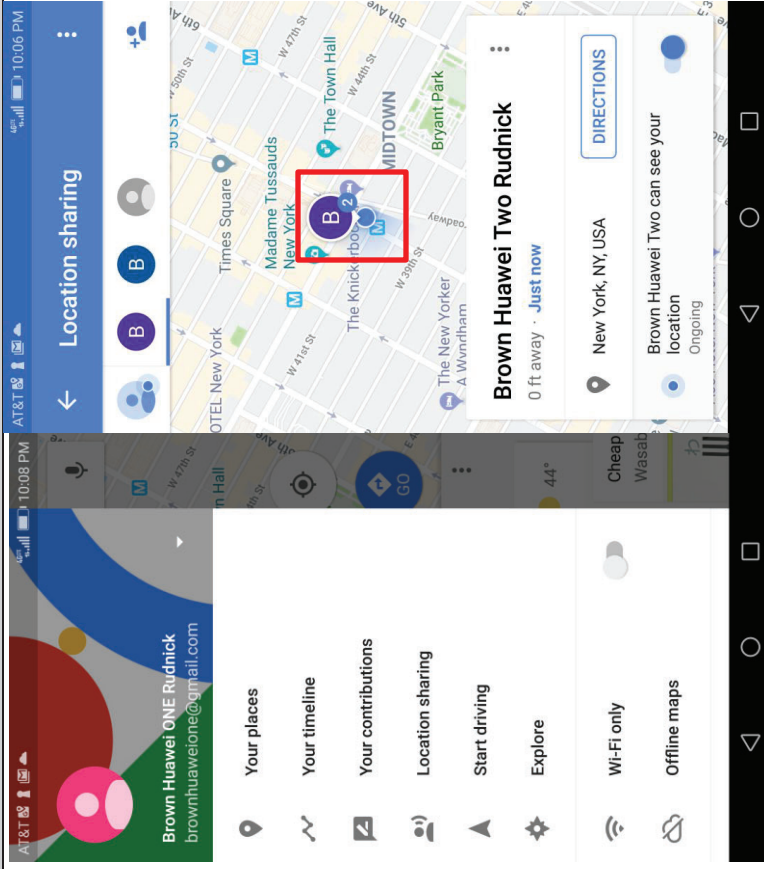


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>
Exemplary Google Maps Screenshots

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to amend these contentions to include additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <h2 data-bbox="235 1081 292 1575">Contacts Provider</h2> <p data-bbox="324 630 592 1575">The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services. The provider accommodates a wide range of data sources and tries to manage as much data as possible for each person, with the result that its organization is complex. Because of this, the provider's API includes an extensive set of contract classes and interfaces that facilitate both data retrieval and modification.</p> <p data-bbox="617 1260 641 1575">This guide describes the following:</p> <ul data-bbox="673 724 852 1575" style="list-style-type: none">• The basic provider structure.• How to retrieve data from the provider.• How to modify data in the provider.• How to write a sync adapter for synchronizing data from your server to the Contacts Provider. <p data-bbox="860 609 885 1585">https://developer.android.com/guide/topics/providers/provider.html</p> |
|--------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Overview

ContactsContract defines an extensible database of contact-related information. Contact information is stored in a three-tier data model:

- A row in the ContactsContract.Data table can store any kind of personal data, such as a phone number or email addresses. The set of data kinds that can be stored in this table is open-ended. There is a predefined set of common kinds, but any application can add its own data kinds.
- A row in the ContactsContract.RawContacts table represents a set of data describing a person and associated with a single account (for example, one of the user's Gmail accounts).
- A row in the ContactsContract.Contacts table represents an aggregate of one or more RawContacts presumably describing the same person. When data in or associated with the RawContacts table is changed, the affected aggregate contacts are updated as necessary.

Other tables include:

- ContactsContract.Groups, which contains information about raw contact groups such as Gmail contact groups. The current API does not support the notion of groups spanning multiple accounts.
- ContactsContract.StatusUpdates, which contains social status updates including IM availability.
- ContactsContract.AggregationExceptions, which is used for manual aggregation and disaggregation of raw contacts
- ContactsContract.Settings, which contains visibility and sync settings for accounts and groups.
- ContactsContract.SyncState, which contains free-form data maintained on behalf of sync adapters
- ContactsContract.PhoneLookup, which is used for quick caller-ID lookup

<https://developer.android.com/reference/android/provider/ContactsContract.html>

Data

As noted previously, the data for a raw contact is stored in a ContactsContract.Data row that is linked to the raw contact's _id value. This allows a single raw contact to have multiple instances of the same type of data such as email addresses or phone numbers. For example, if "Thomas Higginson" for emilyd@gmail.com (the raw contact row for Thomas Higginson associated with the Google account emilyd@gmail.com) has a home email address of thigg@gmail.com and a work email address of thomas.higginson@gmail.com, the Contacts Provider stores the two email address rows and links them both to the raw contact.

Notice that different types of data are stored in this single table. Display name, phone number, email, postal address, photo, and website detail rows are all found in the ContactsContract.Data table. To help manage this, the ContactsContract.Data table has some columns with descriptive names, and others with generic names. The contents of a descriptive-name column have the same meaning regardless of the type of data in the row, while the contents of a generic-name column have different meanings depending on the type of data.

<https://developer.android.com/guide/topics/providers/contact-provider.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | | ZTE | | |
|--|-------------|---|-----------|---|
| Task | Action | Data | MIME type | Notes |
| Pick a contact from a list | ACTION_PICK | <p>One of:</p> <ul style="list-style-type: none"> Contacts.CONTENT_URI, which displays a list of contacts. Phone.CONTENT_URI, which displays a list of phone numbers for a raw contact. StructuredPostal.CONTENT_URI, which displays a list of postal addresses for a raw contact. Email.CONTENT_URI, which displays a list of email addresses for a raw contact. | Not used | <p>Displays a list of raw contacts or a list of data from a raw contact, depending on the content URI type you supply.</p> <p>Call <code>startActivityForResult()</code>, which returns the content URI of the selected row. The form of the URI is the table's content URI with the row's <code>LOOKUP_ID</code> appended to it. The device's contacts app delegates read and write permissions to this content URI for the life of your activity. See the Content Provider Basics guide for more details.</p> |
| <p>https://developer.android.com/guide/topics/providers/contacts-provider.html</p> | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

25 * Parsed form of the intent sent to the Contacts application.
26 */
27 public class ContactsRequest {
28
29     /** Default mode: browse contacts */
30     public static final int ACTION_DEFAULT = 10;
31
32     /** Show all contacts */
33     public static final int ACTION_ALL_CONTACTS = 15;
34
35     /** Show all contacts with phone numbers */
36     public static final int ACTION_CONTACTS_WITH_PHONES = 17;
37
38     /** Show contents of a specific group */
39     public static final int ACTION_GROUP = 20;
40
41     /** Show all starred contacts */
42     public static final int ACTION_STARRED = 30;
43
44     /** Show frequently contacted contacts */
45     public static final int ACTION_FREQUENT = 40;
46
47     /** Show starred and the frequent */
48     public static final int ACTION_STREQUENT = 50;
49
50     /** Show all contacts and pick them when clicking */
51     public static final int ACTION_PICK_CONTACT = 60;
52
53     /** Show all contacts as well as the option to create a new one */
54     public static final int ACTION_PICK_OR_CREATE_CONTACT = 70;
55
56     /** Show all contacts and pick them for edit when clicking, and allow creating a new contact */
57     public static final int ACTION_INSERT_OR_EDIT_CONTACT = 80;

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 59 /** Show all phone numbers and pick them when clicking */ 60 public static final int ACTION_PICK_PHONE = 90; 61 62 /** Show all postal addresses and pick them when clicking */ 63 public static final int ACTION_PICK_POSTAL = 100; 64 65 /** Show all postal addresses and pick them when clicking */ 66 public static final int ACTION_PICK_EMAIL = 105; 67 68 /** Show all contacts and create a shortcut for the picked contact */ 69 public static final int ACTION_CREATE_SHORTCUT_CONTACT = 110; 70 71 /** Show all phone numbers and create a call shortcut for the picked number */ 72 public static final int ACTION_CREATE_SHORTCUT_CALL = 120; 73 74 /** Show all phone numbers and create an SMS shortcut for the picked number */ 75 public static final int ACTION_CREATE_SHORTCUT_SMS = 130; 76 77 /** Show all contacts and activate the specified one */ 78 public static final int ACTION_VIEW_CONTACT = 140; 79 80 /** Show contacts recommended for joining with a specified target contact */ 81 public static final int ACTION_PICK_JOIN = 150; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/list/ContactsRequest.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 104 * Displays a list to browse contacts. 105 */ 106 public class PeopleActivity extends ContactsActivity implements 107 View.OnCreateContextMenuListener, 108 View.OnClickListener, 109 ActionBarAdapter.Listener, 110 DialogManager.DialogShowingViewActivity, 111 ContactListFilterController.ContactListFilterListener, 112 ProviderStatusListener, 113 MultiContactDeleteListener, 114 JoinContactsListener { https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java 145 * Showing a list of Contacts. Also used for showing search results in search mode. 146 */ 147 private MultiSelectContactsListFragment mAllFragment; 148 private ContactTileListFragment mFavoritesFragment; https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1- release/src/com/android/contacts/activities/PeopleActivity.java </pre> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

458 private void configureFragments(boolean fromRequest) {
459     if (fromRequest) {
460         ContactListFilter filter = null;
461         int actionCode = mRequest.getActionCode();
462         boolean searchMode = mRequest.isSearchMode();
463         final int tabToOpen;
464         switch (actionCode) {
465             case ContactsRequest.ACTION_ALL_CONTACTS:
466                 filter = ContactListFilter.createFilterWithType(
467                     ContactListFilter.FILTER_TYPE_ALL_ACCOUNTS);
468                 tabToOpen = TabState.ALL;
469                 break;
470             case ContactsRequest.ACTION_CONTACTS_WITH_PHONES:
471                 filter = ContactListFilter.createFilterWithType(
472                     ContactListFilter.FILTER_TYPE_WITH_PHONE_NUMBERS_ONLY);
473                 tabToOpen = TabState.ALL;
474                 break;
475             case ContactsRequest.ACTION_FREQUENT:
476             case ContactsRequest.ACTION_STREQUENT:
477             case ContactsRequest.ACTION_STARRED:
478                 tabToOpen = TabState.FAVORITES;
479                 break;
480             case ContactsRequest.ACTION_VIEW_CONTACT:
481                 tabToOpen = TabState.ALL;
482                 break;
483             default:
484                 tabToOpen = -1;
485                 break;
486         }
487     }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java>

B-887

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre>488 if (tabToOpen != -1) { 489 mActionBarAdapter.setCurrentTab(tabToOpen); 490 } 491 492 if (filter != null) { 493 mContactListFilterController.setContactListFilter(filter, false); 494 searchMode = false; 495 } 496 497 if (mRequest.getContactUri() != null) { 498 searchMode = false; 499 } 500 501 mActionBarAdapter.setSearchMode(searchMode); 502 configureContactListFragmentForRequest(); 503 } 504 505 configureContactListFragment(); 506 507 invalidateOptionsMenuIfNeeded(); 508 }</pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/activities/PeopleActivity.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public class ProfileAndContactsLoader extends CursorLoader { 36 37 private boolean mLoadProfile; 38 39 private String[] mProjection; 40 41 private Uri mExtraUri; 42 private String[] mExtraProjection; 43 private String mExtraSelection; 44 private String[] mExtraSelectionArgs; 45 private boolean mMergeExtraContactsAfterPrimary; 46 47 public ProfileAndContactsLoader(Context context) { 48 super(context); 49 } </pre> <p>https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/list/ProfileAndContactsLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|---|
| | <pre> 35 public final class GroupMemberLoader extends CursorLoader { 36 37 public static class GroupEditorQuery { 38 private static final String[] PROJECTION = new String[] { 39 Data.CONTACT_ID, // 0 40 Data.RAW_CONTACT_ID, // 1 41 Data.DISPLAY_NAME_PRIMARY, // 2 42 Data.PHOTO_URI, // 3 43 Data.LOOKUP_KEY, // 4 44 }; 45 46 public static final int CONTACT_ID = 0; 47 public static final int RAW_CONTACT_ID = 1; 48 public static final int CONTACT_DISPLAY_NAME_PRIMARY = 2; 49 public static final int CONTACT_PHOTO_URI = 3; 50 public static final int CONTACT_LOOKUP_KEY = 4; 51 } 52 53 public static class GroupDetailQuery { 54 private static final String[] PROJECTION = new String[] { 55 Data.CONTACT_ID, // 0 56 Data.PHOTO_URI, // 1 57 Data.LOOKUP_KEY, // 2 58 Data.DISPLAY_NAME_PRIMARY, // 3 59 Data.CONTACT_PRESENCE, // 4 60 Data.CONTACT_STATUS, // 5 61 }; 62 63 public static final int CONTACT_ID = 0; 64 public static final int CONTACT_PHOTO_URI = 1; 65 public static final int CONTACT_LOOKUP_KEY = 2; 66 public static final int CONTACT_DISPLAY_NAME_PRIMARY = 3; 67 public static final int CONTACT_PRESENCE_STATUS = 4; 68 public static final int CONTACT_STATUS = 5; 69 } 70 71 private final long mGroupId; </pre> <p>https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMemberLoader.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group loader for the group list that includes details such as the number of contacts per group
25 * and number of groups per account. This list is sorted by account type, account name, where the
26 * group names are in alphabetical order. Note that the list excludes default, favorite, and deleted
27 * groups.
28 */
29 public final class GroupListLoader extends CursorLoader {
30
31     private final static String[] COLUMNS = new String[] {
32         Groups.ACCOUNT_NAME,
33         Groups.ACCOUNT_TYPE,
34         Groups.DATA_SET,
35         Groups._ID,
36         Groups.TITLE,
37         Groups.SUMMARY_COUNT,
38     };
39
40     public final static int ACCOUNT_NAME = 0;
41     public final static int ACCOUNT_TYPE = 1;
42     public final static int DATA_SET = 2;
43     public final static int GROUP_ID = 3;
44     public final static int TITLE = 4;
45     public final static int MEMBER_COUNT = 5;
46
47     private static final Uri GROUP_LIST_URI = Groups.CONTENT_SUMMARY_URI;
48
49     public GroupListLoader(Context context) {
50         super(context, GROUP_LIST_URI, COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
51             + Groups.ACCOUNT_NAME + " NOT NULL AND " + Groups.AUTO_ADD + "=0 AND " +
52             Groups.FAVORITES + "=0 AND " + Groups.DELETED + "=0", null,
53             Groups.ACCOUNT_TYPE + ", " + Groups.ACCOUNT_NAME + ", " + Groups.DATA_SET + ", " +
54             Groups.TITLE + " COLLATE LOCALIZED ASC");
55     }
56 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+/nougat-mr1-release/src/com/android/contacts/GroupListLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

24 * Group meta-data loader. Loads all groups or just a single group from the
25 * database (if given a {@link Uri}).
26 */
27 public final class GroupMetadataLoader extends CursorLoader {
28
29     private final static String[] COLUMNS = new String[] {
30         Groups.ACCOUNT_NAME,
31         Groups.ACCOUNT_TYPE,
32         Groups.DATA_SET,
33         Groups.ID,
34         Groups.TITLE,
35         Groups.AUTO_ADD,
36         Groups.FAVORITES,
37         Groups.GROUP_IS_READ_ONLY,
38         Groups.DELETED,
39     };
40
41     public final static int ACCOUNT_NAME = 0;
42     public final static int ACCOUNT_TYPE = 1;
43     public final static int DATA_SET = 2;
44     public final static int GROUP_ID = 3;
45     public final static int TITLE = 4;
46     public final static int AUTO_ADD = 5;
47     public final static int FAVORITES = 6;
48     public final static int IS_READ_ONLY = 7;
49     public final static int DELETED = 8;
50
51     public GroupMetadataLoader(Context context, Uri groupUri) {
52         super(context, ensureIsGroupUri(groupUri), COLUMNS, Groups.ACCOUNT_TYPE + " NOT NULL AND "
53             + Groups.ACCOUNT_NAME + " NOT NULL", null, null);
54     }
55
56     /**
57      * Ensures that this is a valid group URI. If invalid, then an exception is
58      * thrown. Otherwise, the original URI is returned.
59      */
60     private static Uri ensureIsGroupUri(final Uri groupUri) {
61         // TODO: Fix ContactsProvider2 getType method to resolve the group Uri's
62         if (groupUri == null) {
63             throw new IllegalArgumentException("Uri must not be null");
64         }
65         if (!groupUri.toString().startsWith(Groups.CONTENT_URI.toString())) {
66             throw new IllegalArgumentException("Invalid group Uri: " + groupUri);
67         }
68         return groupUri;
69     }
70 }

```

<https://android.googlesource.com/platform/packages/apps/Contacts/+nougat-mr1-release/src/com/android/contacts/GroupMetadataLoader.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

19 * Meta-data for a contact group. We load all groups associated with the contact's
20 * constituent accounts.
21 */
22 public final class GroupMetaData {
23     private String mAccountName;
24     private String mAccountType;
25     private String mDataSet;
26     private long mGroupId;
27     private String mTitle;
28     private boolean mDefaultGroup;
29     private boolean mFavorites;
30
31     public GroupMetaData(String accountName, String accountType, String dataSet, long groupId,
32         String title, boolean defaultGroup, boolean favorites) {
33         this.mAccountName = accountName;
34         this.mAccountType = accountType;
35         this.mDataSet = dataSet;
36         this.mGroupId = groupId;
37         this.mTitle = title;
38         this.mDefaultGroup = defaultGroup;
39         this.mFavorites = favorites;
40     }
41
42     public String getAccountName() {
43         return mAccountName;
44     }
45
46     public String getAccountType() {
47         return mAccountType;
48     }
49
50     public String getDataSet() {
51         return mDataSet;
52     }
53
54     public long getGroupId() {
55         return mGroupId;
56     }
57
58     public String getTitle() {
59         return mTitle;
60     }
61
62     public boolean isDefaultGroup() {
63         return mDefaultGroup;
64     }
65
66     public boolean isFavorites() {
67         return mFavorites;

```

<https://android.googlesource.com/platform/packages/apps/ContactsCommon/+/nougat-mr1-release/src/com/android/contacts/common/GroupMetaData.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 */ 62 * @param context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

240 * Download an MMS message.
241 *
242 * @param context Context
243 * @param contentLocation The url of the MMS message
244 * @throws MmsFailureException
245 * @throws InvalidHeaderValueException
246 */
247 public static void downloadMms(final Context context, final int subId,
248     final String contentLocation, Bundle extras) throws MmsFailureException,
249     InvalidHeaderValueException {
250     final Uri requestUri = Uri.parse(contentLocation);
251     final Uri contentUri = MmsFileProvider.buildRawMmsUri();
252
253     final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION,
254         requestUri,
255         context,
256         SendStatusReceiver.class);
257     downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri);
258     if (extras != null) {
259         downloadedIntent.putExtras(extras);
260     }
261     final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(
262         context,
263         0 /*request code*/,
264         downloadedIntent,
265         PendingIntent.FLAG_UPDATE_CURRENT);
266
267     MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri,
268         downloadedPendingIntent);
269 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java>

B-900

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97  * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98  *
99  * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *     it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113     String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? ("", proxy=" + proxyHost + ":" + proxyPort) : "");
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));
131     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: uaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*statusCode*/, "Sending empty PDU");
156     }
157     connection.setOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178 } else if (METHOD_GET.equals(method)) {
179     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180         logHttpHeaders(connection.getRequestProperties());
181     }
182     connection.setRequestMethod(METHOD_GET);
183 }
184 // Get response
185 final int responseCode = connection.getResponseCode();
186 final String responseMessage = connection.getResponseMessage();
187 Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188 if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189     logHttpHeaders(connection.getHeaderFields());
190 }
191 if (responseCode / 100 != 2) {
192     throw new MmsHttpException(responseCode, responseMessage);
193 }
194 final InputStream in = new BufferedInputStream(connection.getInputStream());
195 final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196 final byte[] buf = new byte[4096];
197 int count = 0;
198 while ((count = in.read(buf)) > 0) {
199     byteOut.write(buf, 0, count);
200 }
201 in.close();
202 final byte[] responseBody = byteOut.toByteArray();
203 Log.d(MmsService.TAG, "HTTP: response size="
204     + (responseBody != null ? responseBody.length : 0));
205 return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-904

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | <p>ZTE</p> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <pre>public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p>Returns</p> <ul style="list-style-type: none">• a new location request <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p><code>public Task<Location> getLastLocation ()</code></p> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <p><code>public Task<LocationAvailability> getLocationAvailability ()</code></p> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|---|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| | <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-------------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p>Public Constructors</p> <pre> public MapView (Context context) public MapView (Context context, AttributeSet attrs) public MapView (Context context, AttributeSet attrs, int defStyle) public MapView (Context context, GoogleMapOptions options) </pre> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> <pre> public void getMapAsync (OnMapReadyCallback callback) </pre> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback The callback object that will be triggered when the map is ready to be used.</p> <pre> public final void onCreate (Bundle savedInstanceState) </pre> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[54D] transmitting IP-based messages including a location of the first device to the respective second devices;</p> | <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting IP-based messages including a location of the first device to the respective second devices. See claims 1[D], 28[D], and 41[D], which are incorporated herein by reference in their entirety.</p> <p>For example, users send their location to a server and receive the location of other devices with whom the location is being shared. To send a location to the network, a user enables location service which enables the device to determine and send its location. If location service is already enabled, the device sends its location to the server as needed by the application (e.g. Google Maps). If location service is not enabled, the application will ask the user to enable location service in order to continue with full functionality, which includes using the device's location. Google Maps applications receive the location of other devices when those devices have location service enabled while using the same respective application. Android Device Manager and Google Maps use the received locations to display those locations on the map, indicating the locations of other devices.</p> <p>See, e.g., location sharing including corresponding code described above with regard to limitation [1C].</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution. http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

To use Google Maps and find your location, you must enable location services on your phone.

1. From the home screen, tap > **System settings** > **Location access**.
2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

Maps can provide directions for travel by foot, public transportation, or car.

1. From the home screen, tap > **Maps**.
2. Tap beside the search box.
3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products




| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |
| | <p>Using Google Maps, a user enables location services to send its location the network, but the user can also choose to share its location, as shown below. Again, each device that participates is able to see the location of the other device using Google Maps' share your location feature. For example:</p> |

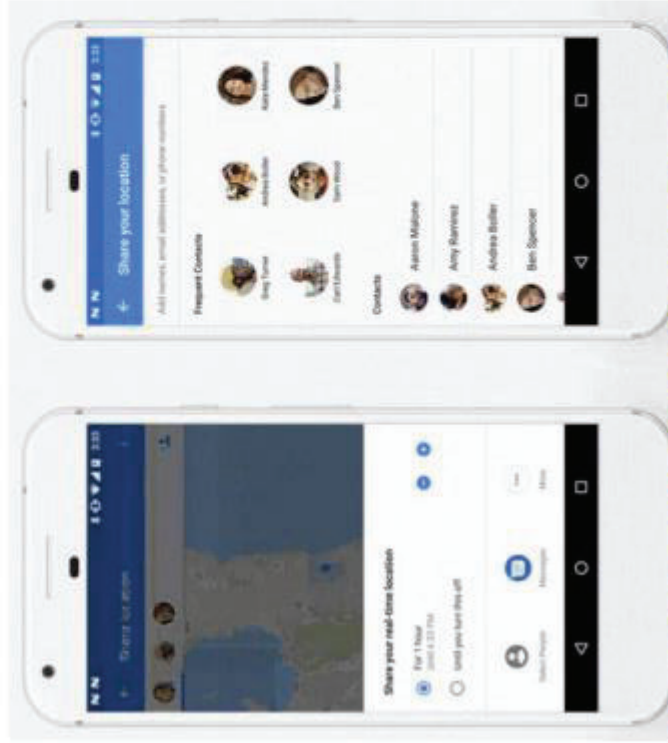
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. If you haven't already, add their Gmail address to your Google Contacts [\[2\]](#).
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap the Menu  > **Share location** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

<https://support.google.com/plus/answer/3302509?co=GENIE.Platform%3DAndroid&hl=en>

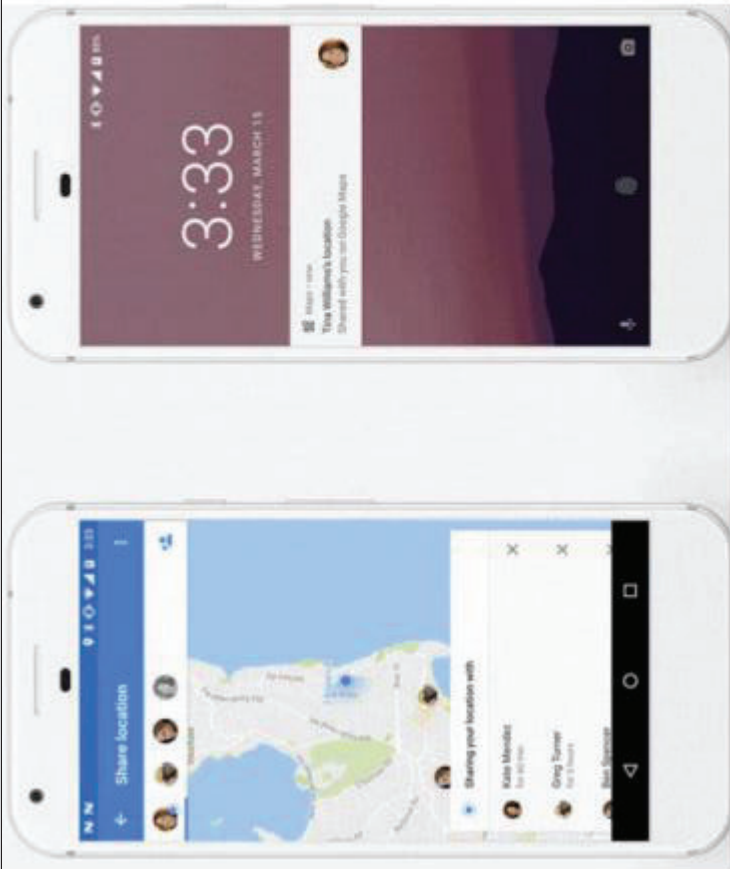


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Below are exemplary methods used by Google applications to obtain, send, and receive locations.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

The Google Play services Location API

The Google Play services Location API is the preferred method for adding location awareness to your Android application. It includes functionality that lets you:

- Determine the device location.
- Listen for location changes.
- Determine the mode of transportation, if the device is moving.
- Create and monitor predefined geographical regions, known as geofences.

The location APIs make it easy for you to build power efficient, location-aware applications. Like the Google Maps Android API, the Location API is distributed as part of the Google Play services SDK. For more information on the Location API, please refer to the Android training class Making Your App Location Aware or the Location API Reference. Code examples are included as part of the Google Play services SDK.

<https://developers.google.com/maps/documentation/android-api/location>

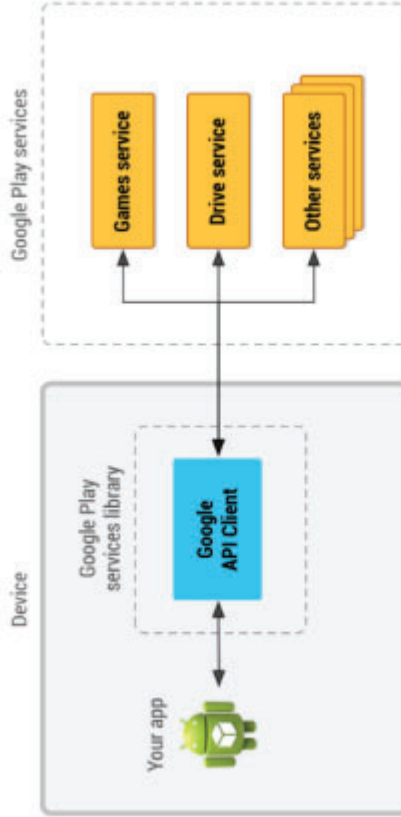


Figure 1: An illustration showing how the Google API Client provides an interface for connecting and making calls to any of the available Google Play services such as Google Play Games and Google Drive.

<https://developers.google.com/android/guides/api-client#Starting>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Get the Last Known Location

Once you have connected to Google Play services and the location services API, you can get the last known location of a user's device. When your app is connected to these you can use the fused location provider's `getLastLocation()` method to retrieve the device location. The precision of the location returned by this call is determined by the permission setting you put in your app manifest, as described in the [Specify App Permissions](#) section of this document.

To request the last known location, call the `getLastLocation()` method, passing it your instance of the `GoogleApiClient` object. Do this in the `onConnected()` callback provided by Google API Client, which is called when the client is ready. The following code snippet illustrates the request and a simple handling of the response:

```

public class MainActivity extends AppCompatActivity implements
    ConnectionCallbacks, OnConnectionFailedListener {
    ...
    @Override
    public void onConnected(Bundle connectionHint) {
        mLastLocation = LocationServices.FusedLocationApi.getLastLocation(
            mGoogleApiClient);
        if (mLastLocation != null) {
            mLatitudeText.setText(String.valueOf(mLastLocation.getLatitude()));
            mLongitudeText.setText(String.valueOf(mLastLocation.getLongitude()));
        }
    }
}

```

The `getLastLocation()` method returns a `Location` object from which you can retrieve the latitude and longitude coordinates of a geographic location. The location object returned may be null in rare cases when the location is not available.

<https://developer.android.com/training/location/retrieve-current.html>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>Determining the user's current location</p> <p>The Geolocation API offers a simple, "one-shot" method to obtain the user's location: <code>getCurrentPosition()</code>. A call to this method asynchronously reports on the user's current location.</p> <pre> window.onload = function() { var startPos; var geoSuccess = function(position) { startPos = position; document.getElementById('startLat').innerHTML = startPos.coords.latitude; document.getElementById('startLon').innerHTML = startPos.coords.longitude; }; navigator.geolocation.getCurrentPosition(geoSuccess); }; </pre> <p>If this is the first time that an application on this domain has requested permissions, the browser typically checks for user consent. Depending on the browser, there may also be preferences to always allow—or disallow—permission lookups, in which case the confirmation process is bypassed.</p> <p>Depending on the location device your browser is using, the position object might actually contain a lot more than just latitude and longitude; for example, it might include an altitude or a direction. You can't tell what extra information that location system uses until it actually returns the data.</p> <p>https://developers.google.com/web/fundamentals/native-hardware/user-location/</p> |
| <p>[54E] transmitting an IP-based text message to at least one of the second devices via a cellular communications network;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of transmitting an IP-based text message to at least one of the second devices via a cellular communications network. See claims 1[A], 8, 12, 28[A], 41[A], and 54[A], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused products run Android Messages which access contact information for second users using respective second devices.</p> <p>Upon information and belief, the Accused Products are forms of cellular devices or PDAs in that the functionality of a PDA has been subsumed into smartphones, tablets, and portable media players having the functionalities of a PDA that include cellular transceivers to enable cellular communications. To the extent that it is necessary, AGIS submits that the Accused Products meet the claim limitation "the first device is a</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2



ZTE

personal digital assistant (PDA) or a cellular phone” under the doctrine of equivalents. For example, U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes ZTE Android phones as smartphones for which U.S. Cellular provides cellular services.
<https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod11060051>

Send & receive text messages in Android Messages

You can send and receive text messages with friends and contacts on Android Messages.

Start a conversation

1. Open the Android Messages app .
2. Tap Compose .
3. In "To," enter the names, phone numbers, or email addresses that you'd like to message. You can also pick from your top contacts or your whole contact list.
4. Tap Next .

https://support.google.com/nexus/answer/6080324?hl=en&ref_topic=6080329

For example, the Accused Products are known to use IP-based communication over wireless or data connections. Both Android Messages and Google Hangouts utilize SMS messages, including group messages from one device to several devices, to send an SMS message, with additional information, to a contact. U.S. Cellular which is one of the largest mobile cellular service providers in the United States and categorizes ZTE Android phones as smartphones for which U.S. Cellular provides cellular communications services to enable the IP-based communications over its cellular network.

<https://www.uscellular.com/uscellular/cell-phones/showPhoneDetails.jsp?productId=prod11060051>

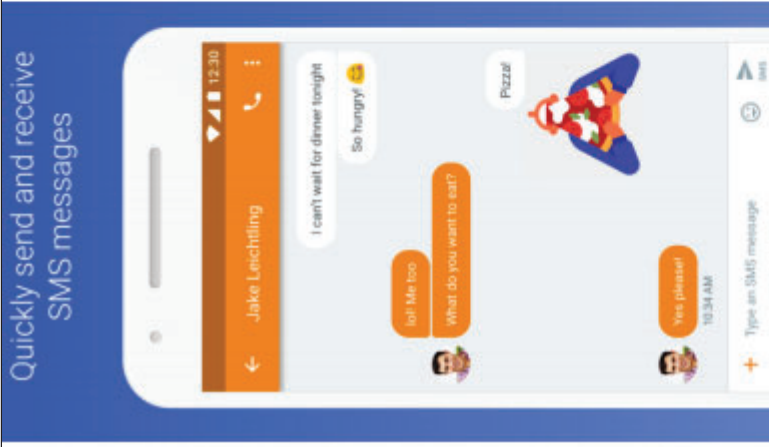
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="196 1520 220 1583">ZTE</p> <p data-bbox="261 583 358 1556">Android Messages makes it easy to communicate with anyone by using SMS, MMS, and more. Stay in touch with friends and family, send group texts, and share your favorite pictures, videos, audio messages.</p> <ul data-bbox="391 485 456 1556" style="list-style-type: none">• Enhanced features: On supported carriers, you can send messages over Wi-Fi or your data network, see when friends have read your message, and more. <p data-bbox="477 478 505 1570">https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://play.google.com/store/apps/details?id=com.google.android.apps.messaging&hl=en>

Get started with Hangouts

You can use Hangouts to:

- Start a chat conversation or video call.
- Make phone calls using Wi-Fi or data.
- Send text messages with your Google Voice or Project Fi phone number.

Hangouts sync automatically across devices. If you start a Hangout on your computer, you can continue your chat on another device, like your phone.

https://support.google.com/hangouts/answer/2944865?hl=en&ref_topic=6386410

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|---|---|
| <p>[54F] presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices; the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of presenting, via an interactive display of the first device, an interactive map and a plurality of user selectable symbols corresponding to the plurality of second devices, wherein the symbols are positioned on the map at respective positions corresponding to the respective locations of the second devices. See claims 1[E], 28[E], and 41[E], which are incorporated herein by reference in their entirety.</p> <p>For example, the Accused Products use Android Device Manager, and Google Maps to display an interface with a map and symbols representing devices.</p> <p>Using Android Device Manager, the user is presented with a map that appears to be based on or imported from Google Maps. The map is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on the number of devices linked to the Google Account, Android Device Manager places symbols on the map and in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p><u>Regarding Google Maps</u>, Google Latitude, Google Plus, Google Hangouts, Google Messages, Google Allo, Google Duo, Google Chrome, and Android Messenger, the Accused Products display, to the user on the display of the first device, a map with one or more symbols corresponding to one or more second users (or second devices corresponding to the second users). The map is interactive because the user may control the display of the map, e.g., pan, zoom, and/or effect change to the map in an otherwise interactive manner. The map is georeferenced for at least the reason that one or more symbols are associated with spatial locations, i.e., coordinates. The symbols are user-selectable because a user may touch the display to select the user or device associated with the symbol. For example, the user may input a touch selection directly on or near the portion of the display corresponding to the symbol's coordinates on the map to effect a selection of the user or device.</p> <p><u>Exemplary Support for Google Maps</u>: Using Google Maps and its location sharing feature, the user is presented with a map that is interactive because the user can pan, zoom, and make selections, among other interactivities. Depending on how many other devices or Google Accounts are sharing their locations, Google Maps places symbols on the map and</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>in a list, where those symbols represent the linked devices and their positions. The symbols are selectable by touching the display. The map is georeferenced, at least because its contents, including the symbols, are placed and spaced according to geographical coordinates.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> |
|---------------------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

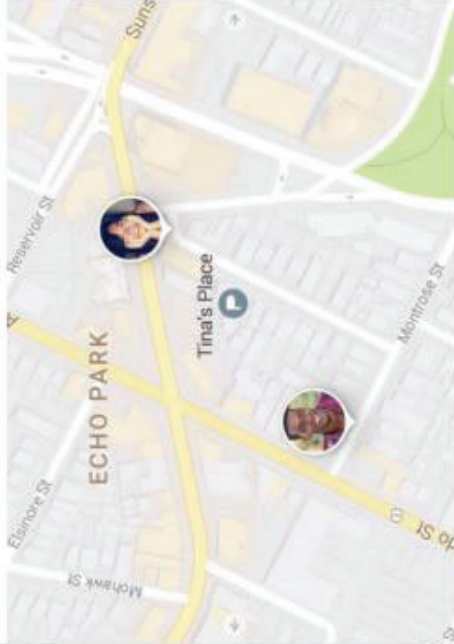
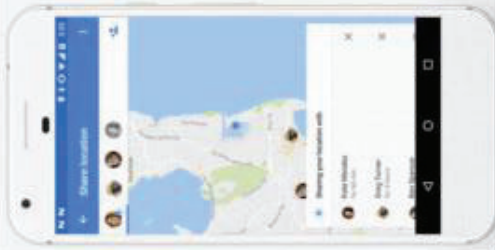
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.</p> <p>Note: The Explore nearby feature is not available for all areas.</p> | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 🗣️ > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap [Settings] > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app 
 2. Tap the Menu  > **Share location**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app 
2. On the map, tap **Share** icon.
3. At the bottom, tap **More** .
4. To temporarily hide someone, tap **hide from map**. You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

https://support.google.com/maps/answer/7326816?hl=en&ref_topic=3092425&co=GENIE.Platform%3DAndroid&oc=1

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap More  > **Share trip progress.**
 4. Choose a person from the list.
 5. Tap **Share.**
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing.**

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing.**
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh.**

Stop seeing someone's location

1. Open the Google Maps app .
2. On the map, tap their icon.
3. At the bottom, tap More .
4. To temporarily hide someone, tap **Hide from map.** You can stop hiding them at any time.

Note: You can stop someone's location from ever appearing on your map. [Learn how to block another person's account.](#)

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

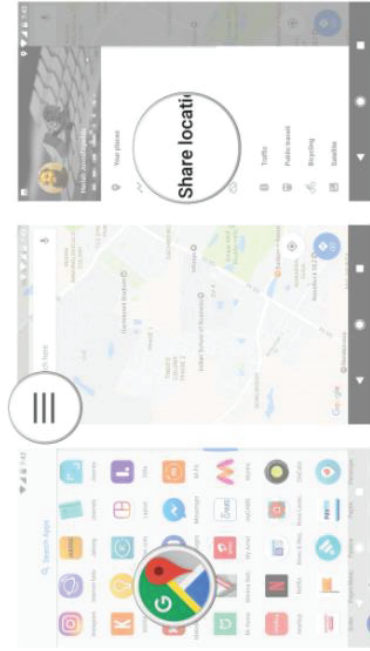
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

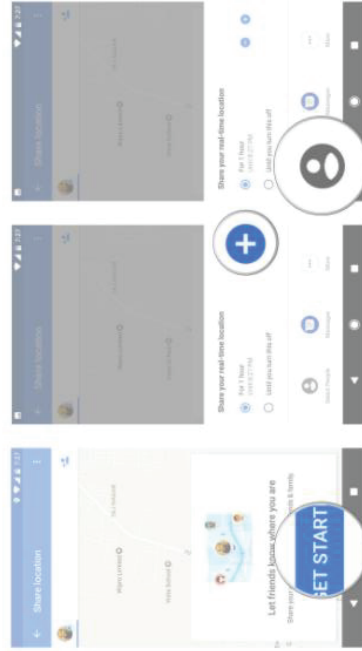
ZTE

How to share your location in Google Maps

- 1. Open Google Maps from the app drawer or the home screen.
- 2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
- 3. Select Share location.



- 4. Tap Get Started.
- 5. Use the + icon to select a time period or select the Until you turn this off setting to share your location indefinitely.
- 6. Tap Select People.



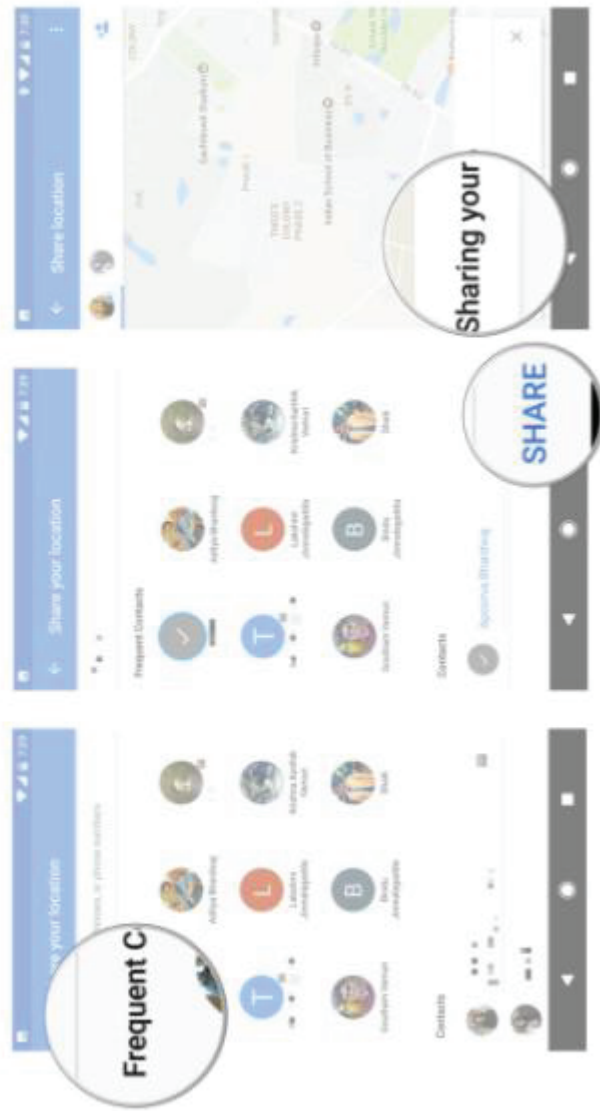
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

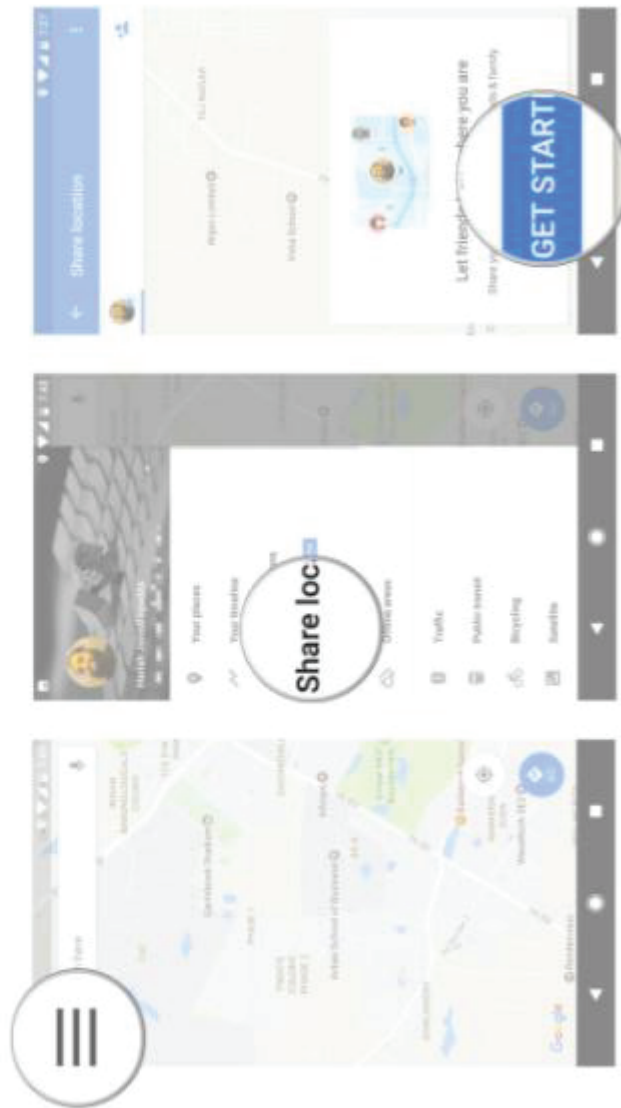
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



<https://www.androidcentral.com/how-share-location-google-maps>

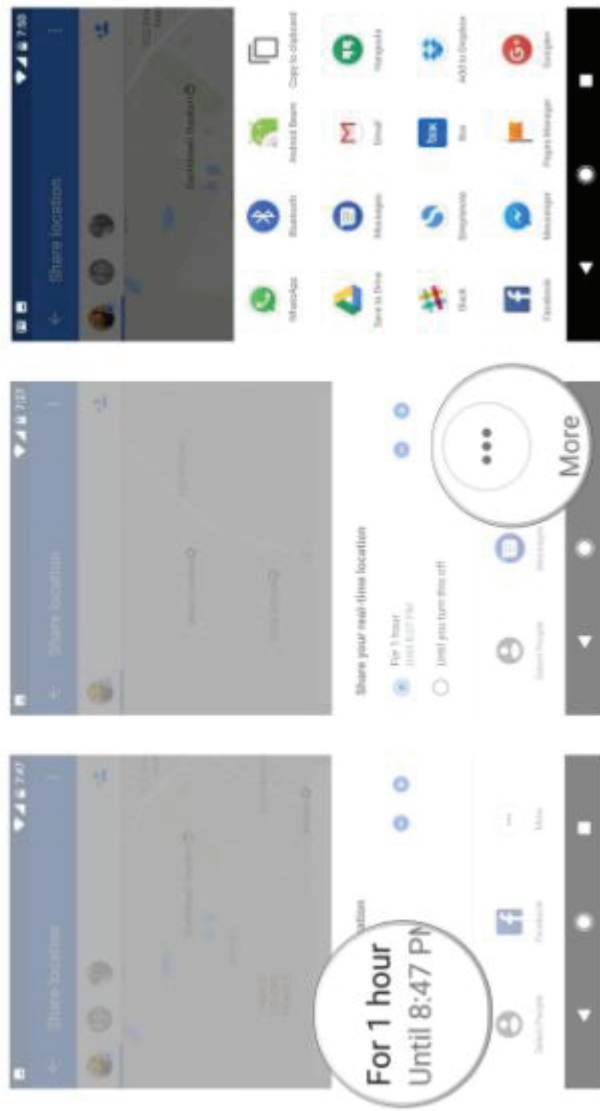
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.

6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

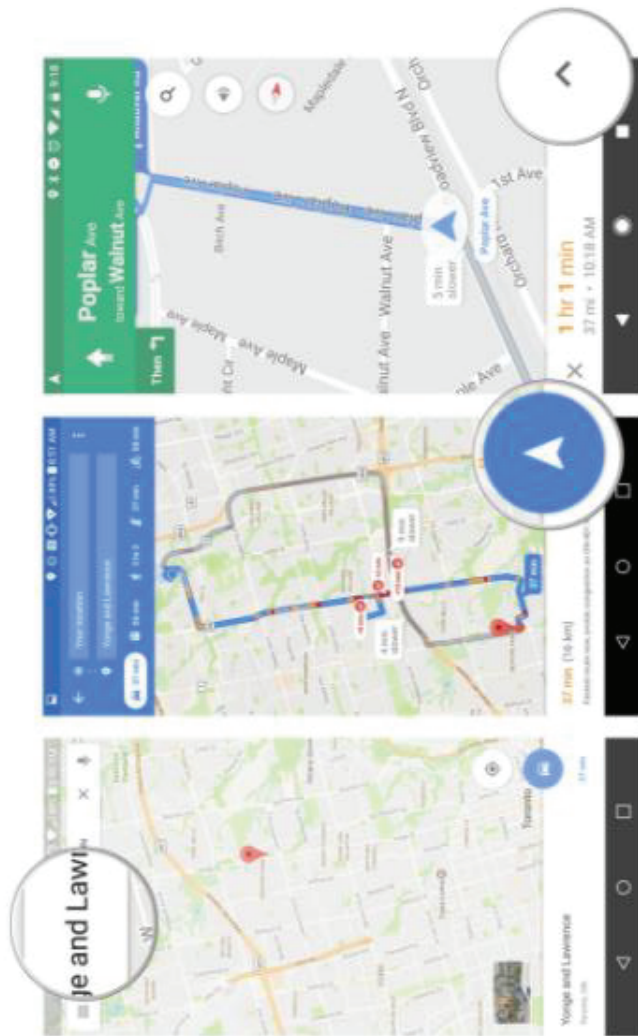
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



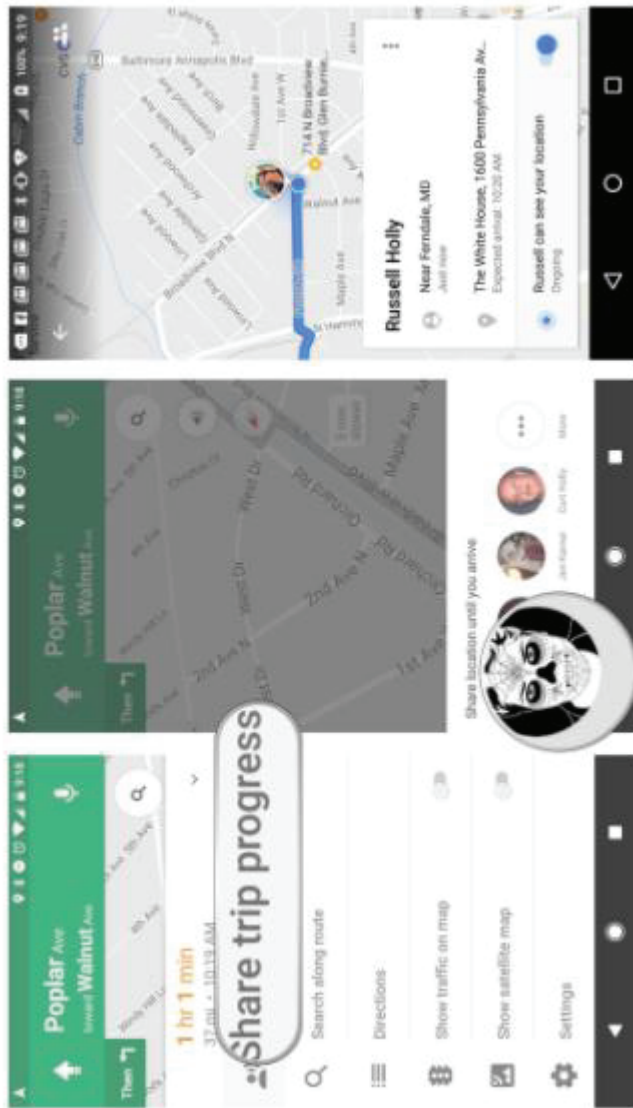
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



You can also stop sharing your location with someone before a trip ends.

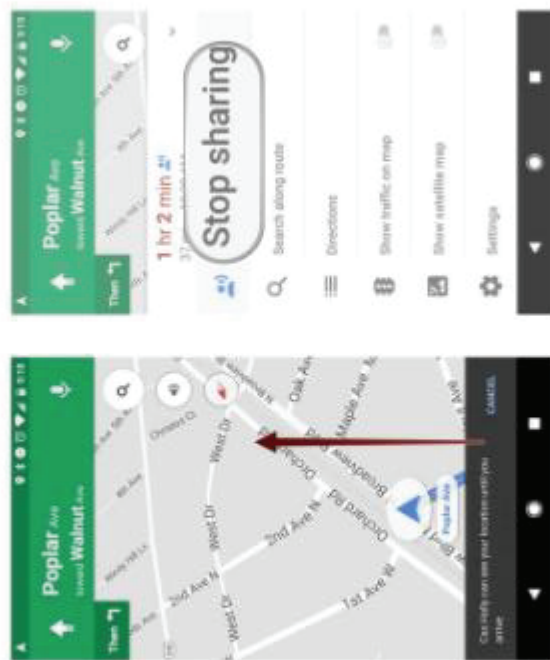
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?

<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

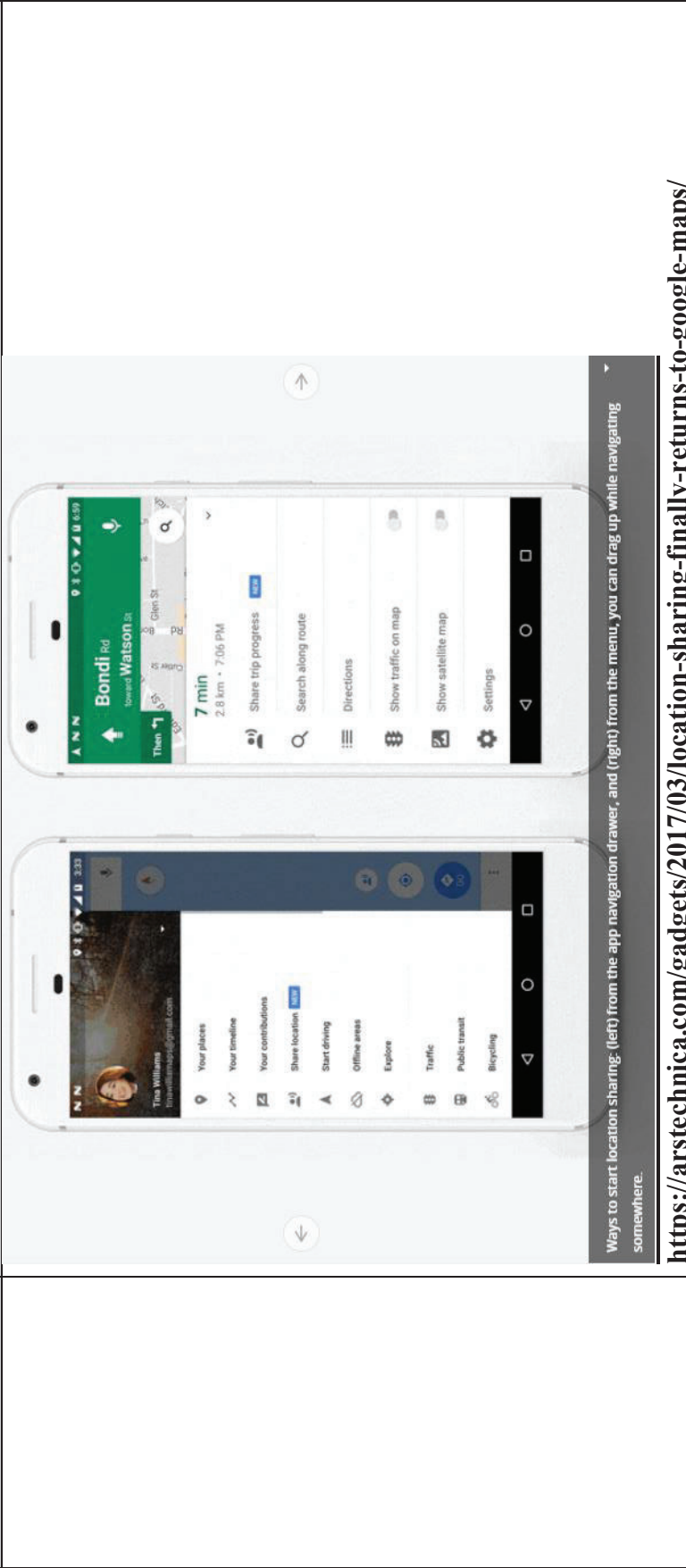
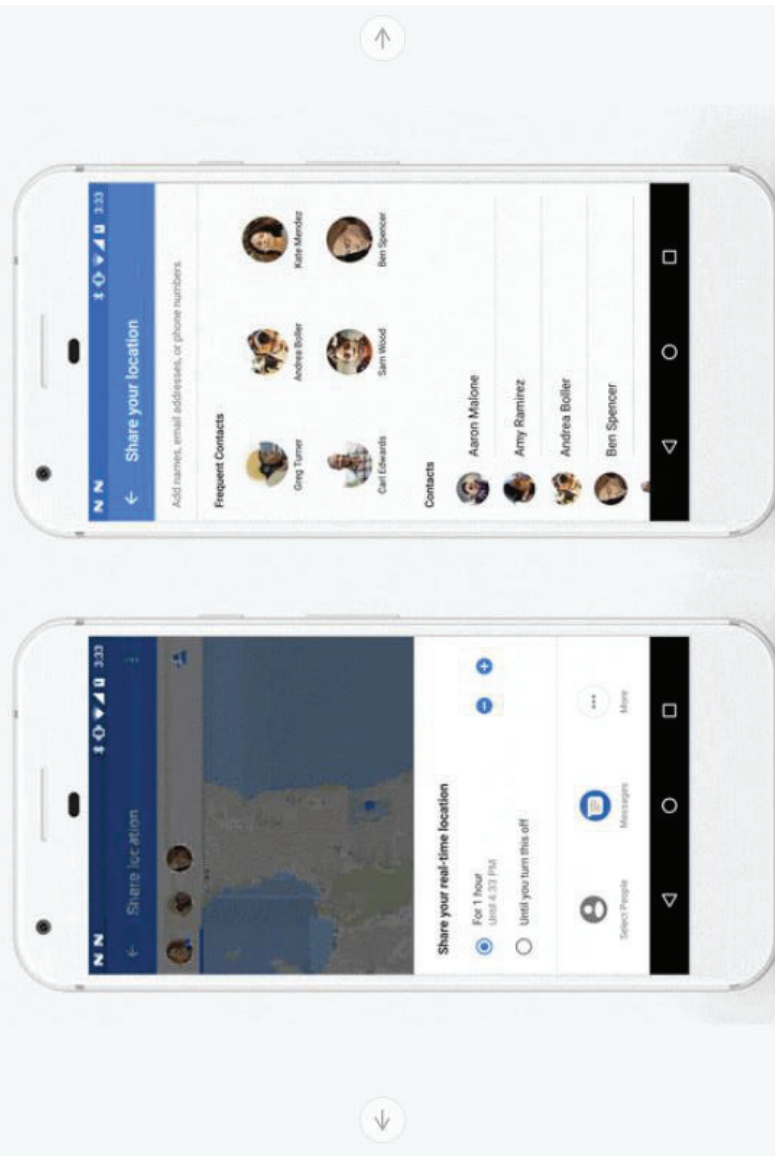


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



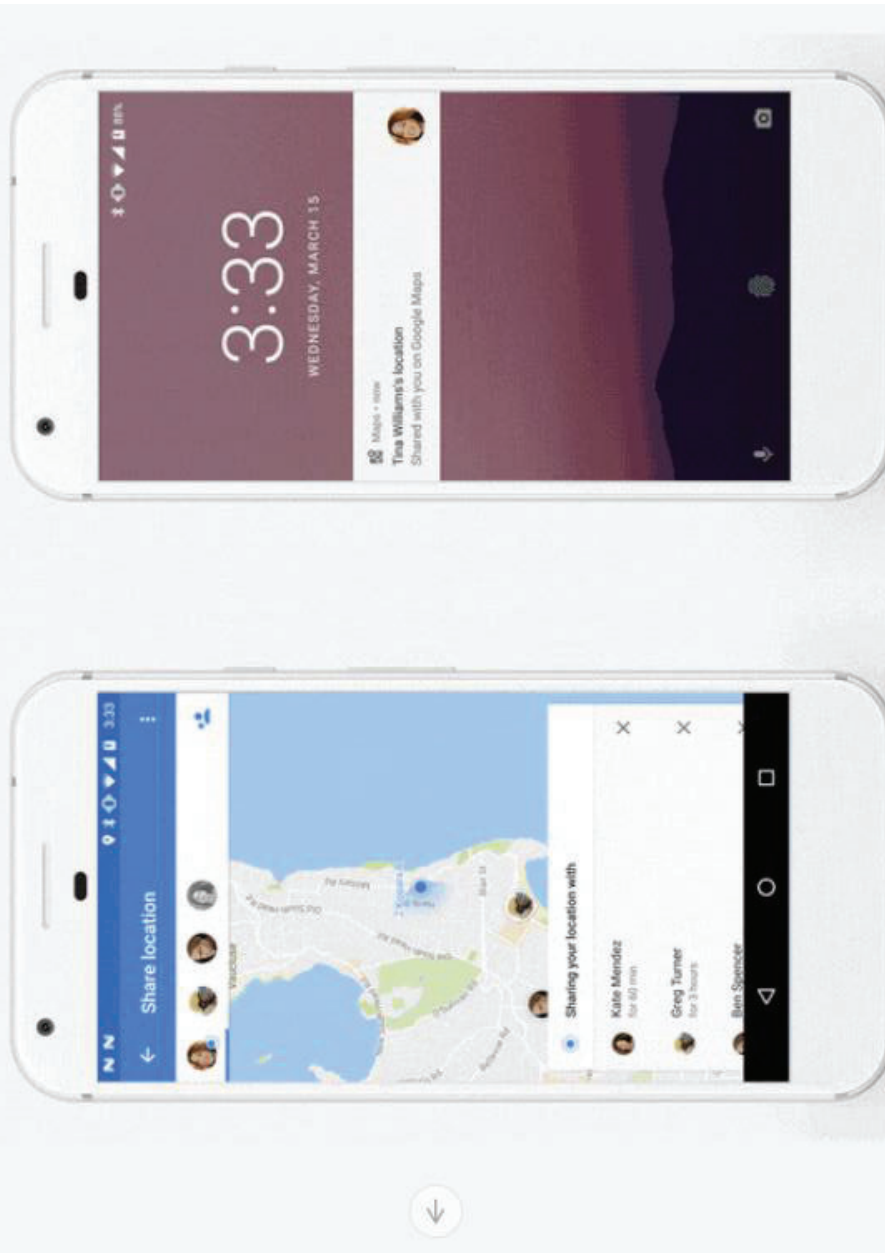
You can pick how long you want to share your location for, and with whom you want to share it.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



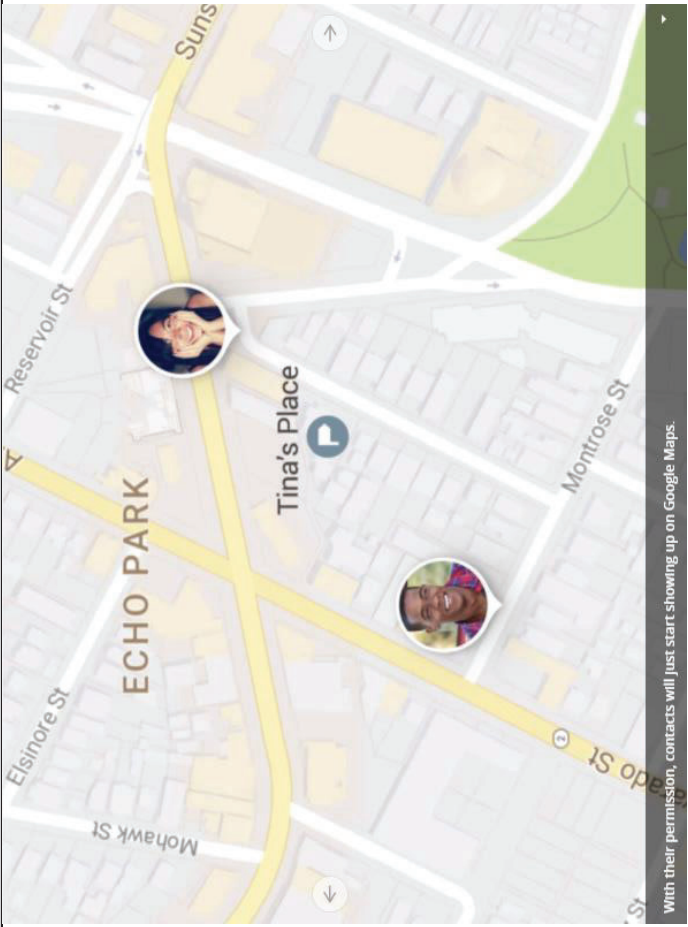
People you've shared with will get a notification from Google Maps.

<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



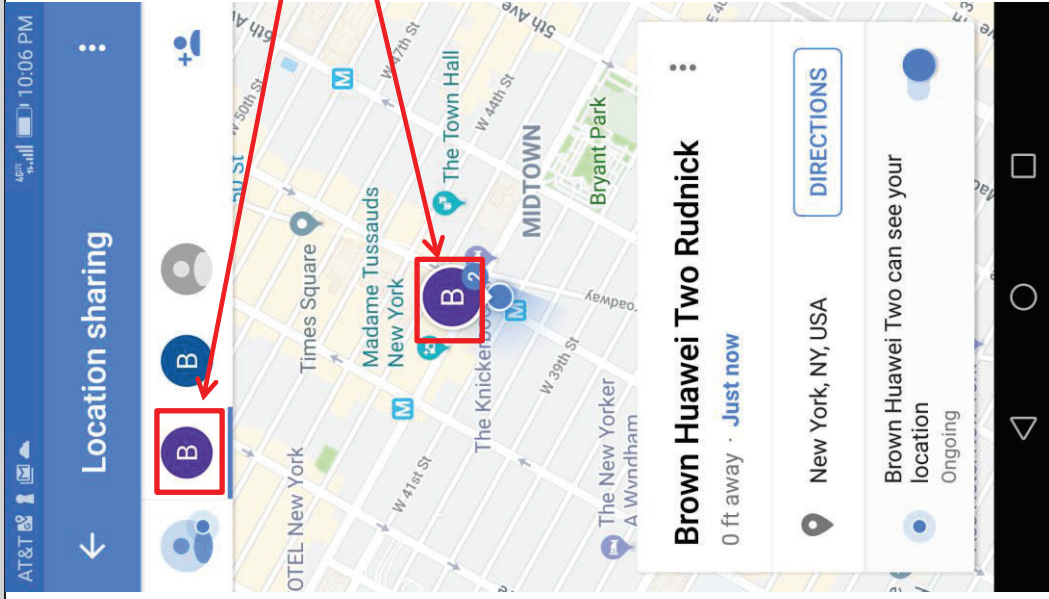
<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exemplary Google Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary User
Selectable Symbols

Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or

B-943

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| US9408055B2 | <p>ZTE</p> <p>source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.</p> <pre>public static LocationRequest create ()</pre> <p>Create a location request with default parameters.</p> <p>Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the <code>FusedLocationProviderApi</code>.</p> <p>Returns</p> <ul style="list-style-type: none">• a new location request <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---------------------------|-------------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|---------------------------|-------------------|

| | |
|---|---|
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> | <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> |
| <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>public Task<Location> getLastLocation ()</pre> <p>Returns the best most recent location currently available.</p> <p>If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned.</p> <p>This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates.</p> <pre>public Task<LocationAvailability> getLocationAvailability ()</pre> <p>Returns the availability of location data. When <code>isLocationAvailable()</code> returns true, then the location returned by <code>getLastLocation()</code> will be reasonably up to date within the hints specified by the active <code>LocationRequest</code> s.</p> <p>If the client isn't connected to Google Play services and the request times out, null is returned.</p> <p>Note it's always possible for <code>getLastLocation()</code> to return null even when this method returns true (e.g. location settings were disabled between calls).</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> |
|--------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|--------------------|--|----------------|---------------------------------------|-----------------|--|---------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous LocationRequests registered on this LocationListener will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td>request</td> <td>The location request for the updates.</td> </tr> <tr> <td>callback</td> <td>The callback for the location updates.</td> </tr> <tr> <td>looper</td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | request | The location request for the updates. | callback | The callback for the location updates. | looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| request | The location request for the updates. | | | | | | |
| callback | The callback for the location updates. | | | | | | |
| looper | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1575">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1575">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|--------------------|------------|

| | |
|--|--|
| <p>public void onLocationAvailable (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> | |
|--|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p>ZTE</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/gms/location/LocationListener</p> <p>Public Constructors</p> <p>public MapView (Context context)</p> <p>public MapView (Context context, AttributeSet attrs)</p> <p>public MapView (Context context, AttributeSet attrs, int defStyle)</p> <p>public MapView (Context context, GoogleMapOptions options)</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>public void getMapAsync (OnMapReadyCallback callback)</p> <p>Returns a non-null instance of the <code>Goog1eMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>Goog1eMap</code> object provided by the callback is non-null. <p>Parameters</p> <p>callback</p> <p>The callback object that will be triggered when the map is ready to be used.</p> <p>public final void onCreate (Bundle savedInstanceState)</p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[54G] identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of identifying user interaction with the interactive display selecting one or more of the user-selectable symbols corresponding to one or more of the second devices and user interaction with the display specifying an action and, based thereon, sending data to the one or more second devices. See claims 1[F], 28[F], and 41[F], which are incorporated herein by reference in their entirety.</p> <p>Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> <p>action and, based thereon, sending data to the one or more second devices;</p> | <p>ZTE</p> <p>specified, data is sent from the first device to the second device via a server.</p> <p><u>Exemplary Support for Google Maps:</u></p> <p>Using Google Maps, a user may choose a symbol and send data to that device. For example, a user who is already sharing her location with another user can stop sharing by making a selection resulting in the second device no longer displaying the first device's location. Additionally, a user can share an ETA message with another user or send another user a link in a message to share her location. Additionally, a user who is sharing a location until she arrives can make a selection to stop her location from showing on the second device.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> |
|---|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


| | |
|---------------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap ▲ to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. <p>Note: The Explore nearby feature is not available for all areas.</p> |
| | <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap 📍 > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap 📍 > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> |





Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2





ZTE

COMPUTER ANDROID IPHONE & IPAD


If they have a Google Account

1. If you haven't already, add their Gmail address to your Google Contacts .
2. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
3. Tap Menu  > **Location sharing** > Add People .
4. Choose how long you want to share your location.
5. Tap **Select People**.
 - If you're asked about your contacts, give Google Maps access.
6. Choose who you want to share with.
7. Tap **Share**.

If they don't have a Google Account

1. On your Android phone or tablet, open the Google Maps app  and sign in. [Learn how to sign in.](#)
2. Tap Menu  > **Location sharing** > Add People .
3. Tap More  > **Copy to clipboard**. People with this link can see your location for as long as you choose, up to 72 hours.

Share using another app

You can also share through messaging apps. Tap More  > select an app.

Stop sharing

1. Open the Google Maps app .
2. Tap Menu  > **Location sharing**.
3. Next to the person with whom you want to stop sharing, tap Remove .

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.



1. Open the Google Maps app .
 2. Set a driving destination. Learn how to navigate to a place.
 3. After you start navigation, tap More  > **Share trip progress**.
 4. Choose a person from the list.
 5. Tap **Share**.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > **Stop sharing**.

See where someone is

If someone shares their location with you, you can see them on the map.

1. Open the Google Maps app .
 2. Tap Menu  > **Location sharing**.
 3. Choose someone.
- To see an updated location, tap on a friend's icon > More  > **Refresh**.

Stop seeing someone's location

1. Open the Google Maps app .
 2. On the map, tap their icon.
 3. At the bottom, tap More .
 4. To temporarily hide someone, tap **Hide from map**. You can stop hiding them at any time.
- Note:** You can stop someone's location from ever appearing on your map. Learn how to block another person's account.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&oco=1>




Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Hide or share lists

Note: You can't share starred places.

1. Open the Google Maps app .
2. Tap Menu  > **Your places** > **Saved**.
3. Next to the list you want to share, tap **More**  > choose an option:
 - **Hide/Show on your map:** Display or hide your saved places when looking at the map.
 - **Share list:** Allow others to see your saved list.
 - **Sharing options:** You can make your list public, private, or shared. To let anyone with the link see your list, tap **Shared**. To let anyone find and follow your list, tap **Public**.

Follow a list

If you follow a list made by someone else, their saved places will show up in Your Places. The places will also appear as suggested locations in Google Maps.

Follow a list using a link

1. Tap on the link you received to open it.
2. Tap **Follow**. This list will now be added to the group of lists you follow.
3. **Optional:** To unfollow a list someone shared with you, tap the list > **Following**.

See lists made by others

If a user has any Google Maps lists that were made public, you can follow them.

1. Tap on the name of a user whose list you want to follow.
2. Tap **Lists**.
3. Tap on the list you want to follow > **More**  > **Follow**.

https://support.google.com/maps/answer/7280933?hl=en&ref_topic=7301134&co=GENIE.Platform%3DAn

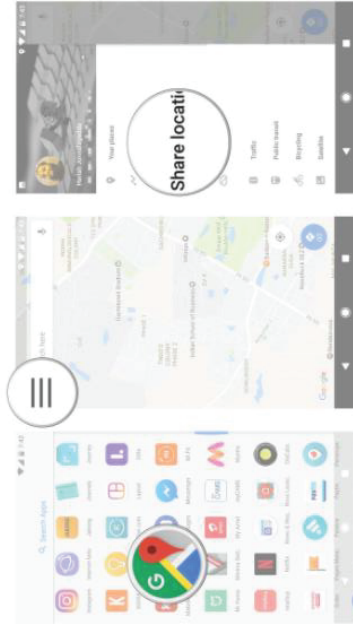
Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

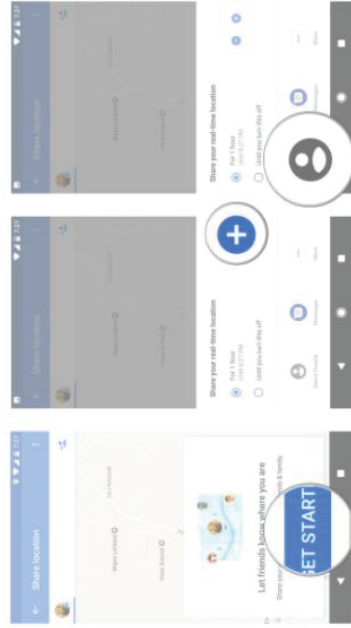
ZTE
droid&oco=1

How to share your location in Google Maps

1. Open Google Maps from the app drawer or the home screen.
2. Tap the hamburger menu (the three horizontal lines) on the top left corner of the screen.
3. Select Share location.



4. Tap Get Started.
5. Use the + icon to select a time period or select the **Until** you turn this off setting to share your location indefinitely.
6. Tap Select People.



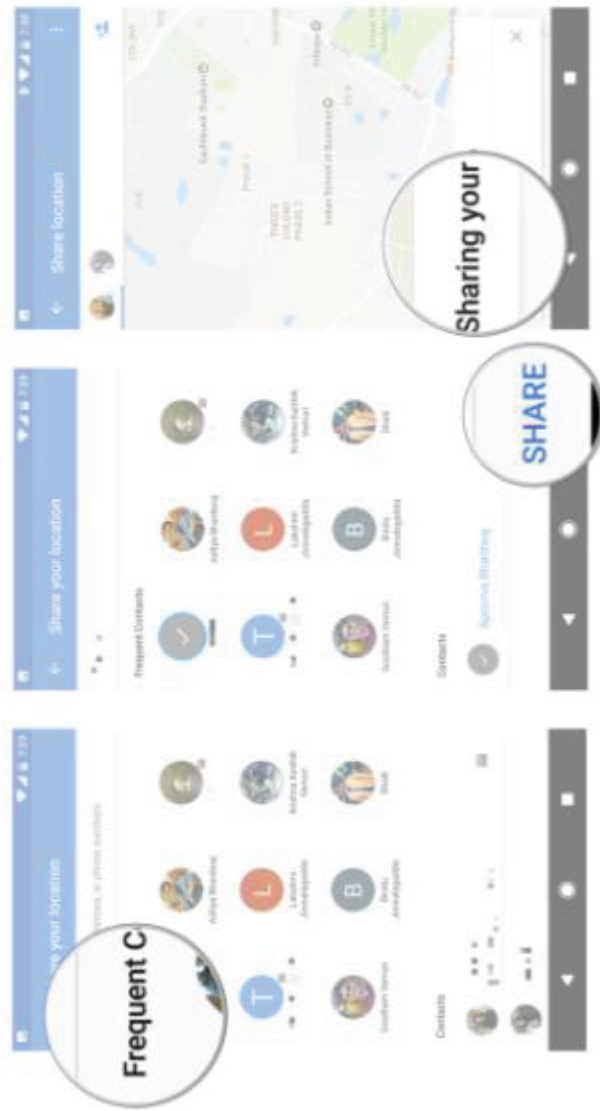
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 7. You'll see a list of your frequent contacts at the top, along with a full list of contacts. Pick the contacts by tapping their name.
- 8. Once you've selected the contacts you want to share your location to, tap Share.
- 9. You'll see a message saying that the selected contact can view your location.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

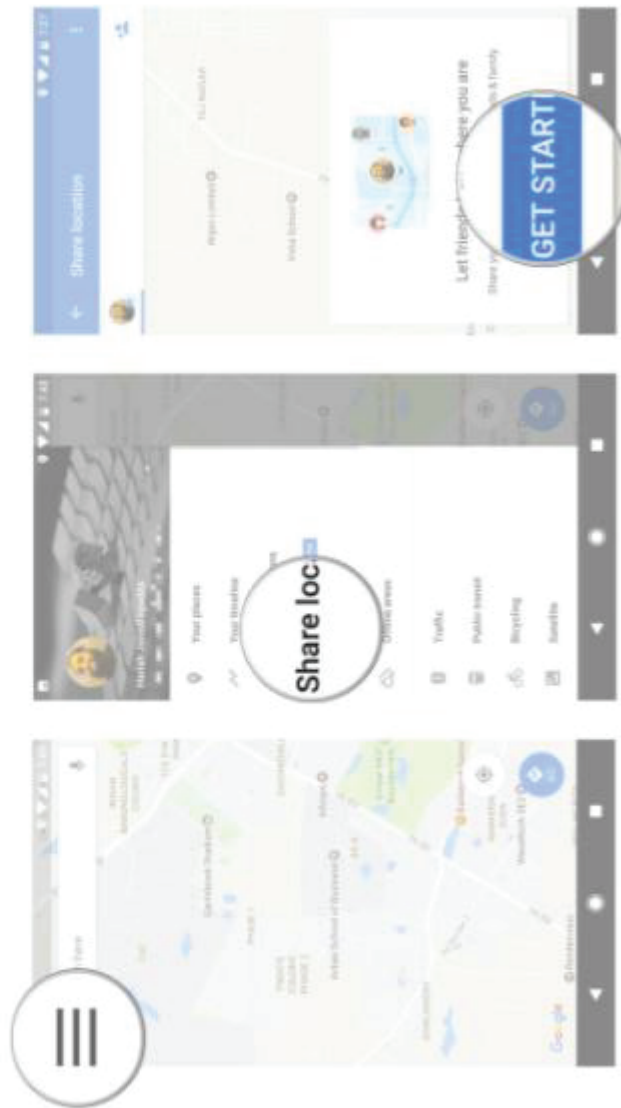
US9408055B2

ZTE

How to create a shareable link

You can also create a link and use it to share your location easily. Here's how to do it:

1. Tap the hamburger menu on the top left corner of the screen.
2. Select Share location.
3. Tap Get Started.



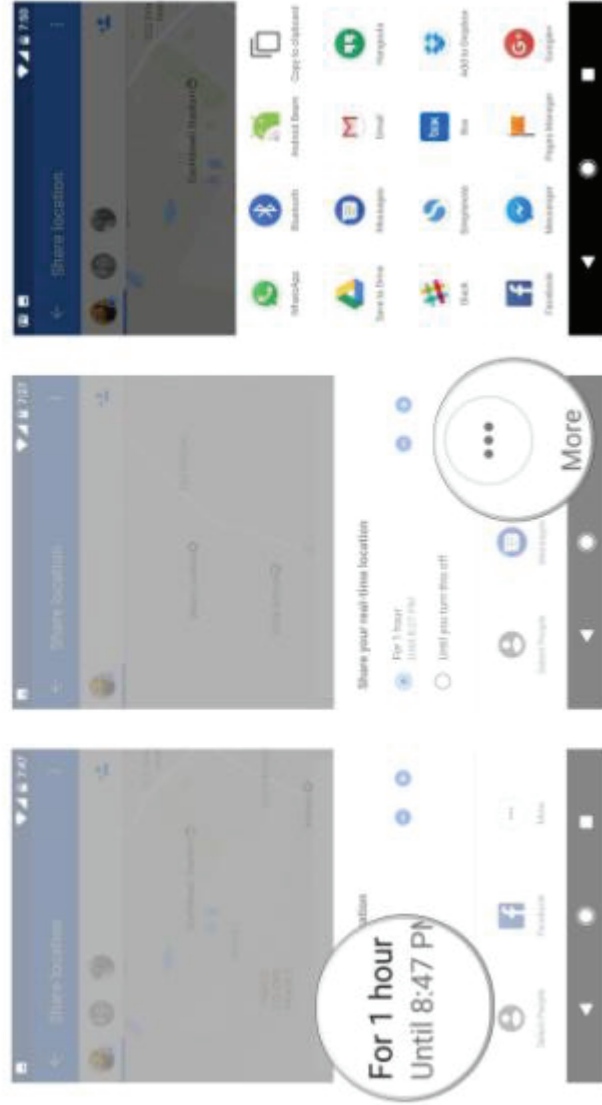
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Select the amount of time you want to share your location.
- 5. Tap More.
- 6. Select your app of choice to create and send a unique URL that broadcasts your current location. You can email it, send the link via Messenger, or even tweet it to the intended recipient.



<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

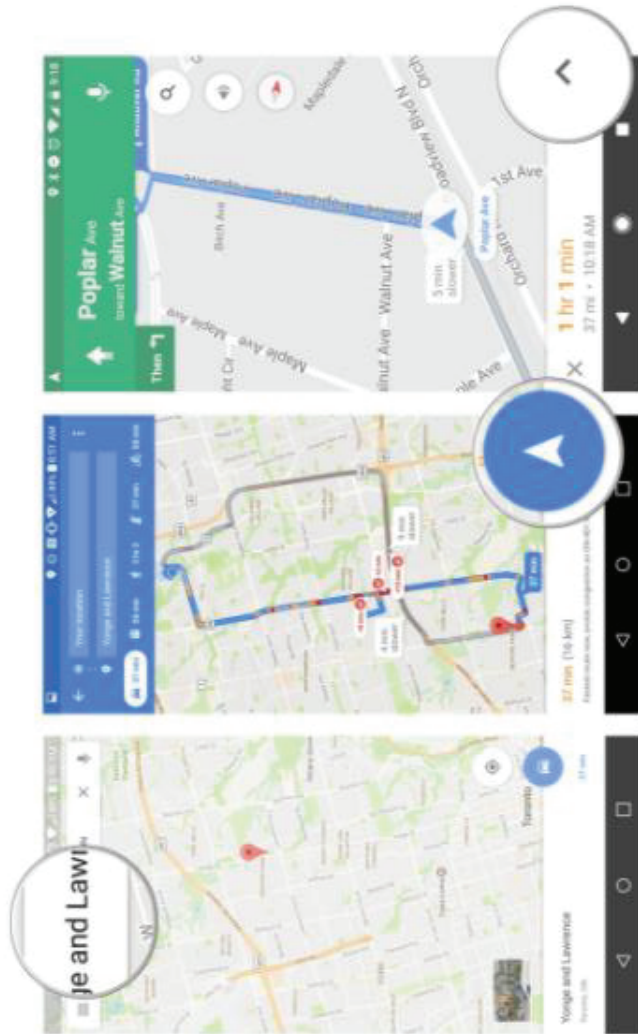
US9408055B2

ZTE

How to share your navigation directions while you walk, drive or transit

One of the best ways to share your location with someone is during a drive, walk or transit. If you're meeting a friend or family member somewhere, or navigating towards their home, Google Maps lets you share your location with them for the duration of the trip. It's magic!

1. In the search bar enter your destination.
2. Pick your navigation type (drive, transit, walk) and press the blue navigate button.
3. Tap the arrow next to the time-to-destination number at the bottom of the screen.



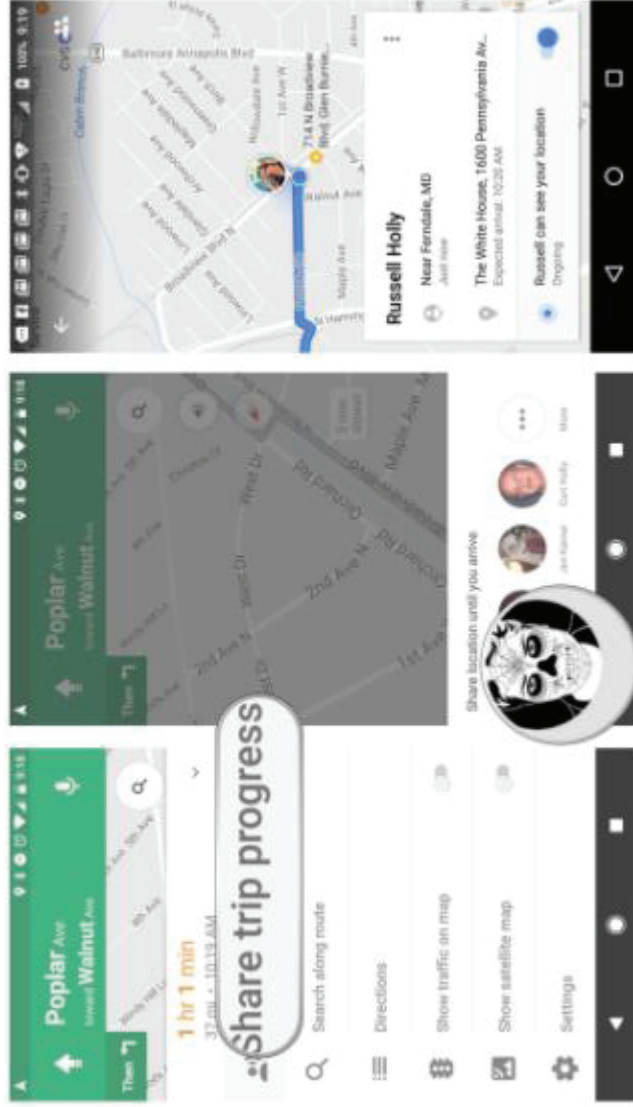
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

- 4. Tap Share trip progress.
- 5. Choose one or more contacts to share trip progress.



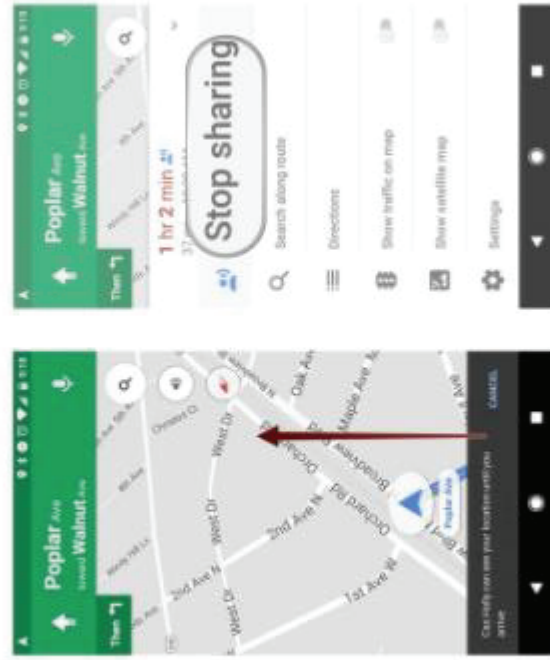
You can also stop sharing your location with someone before a trip ends.
<https://www.androidcentral.com/how-share-location-google-maps>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

1. Tap the arrow next to the time-to-destination number at the bottom of the screen.
2. Tap Stop sharing.



That's it!

Are you excited that location sharing is back in Google Maps? How often do you use the feature?
<https://www.androidcentral.com/how-share-location-google-maps>



As shown below, a group may also be defined within Google Contacts.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

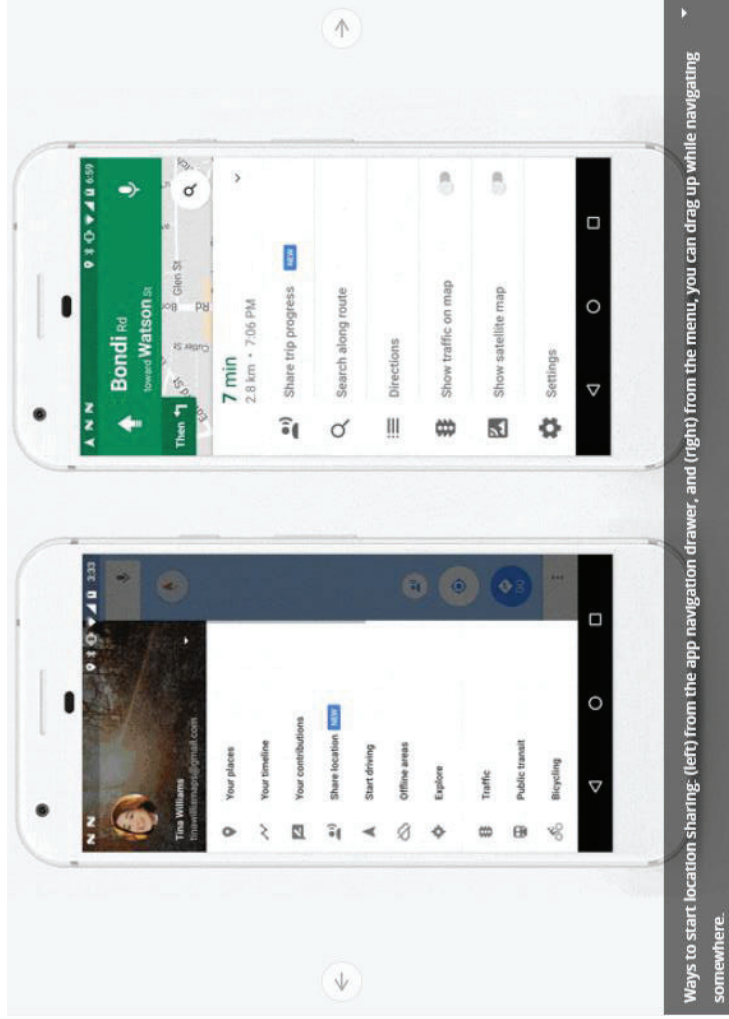
US9408055B2

ZTE

Share your contacts

1. Open your device's Contacts app .
2. Tap a contact in the list.
3. Tap More  > **Share**.
4. Choose how you want to share the contact.

https://support.google.com/android/answer/6118731?hl=en&ref_topic=6118711



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

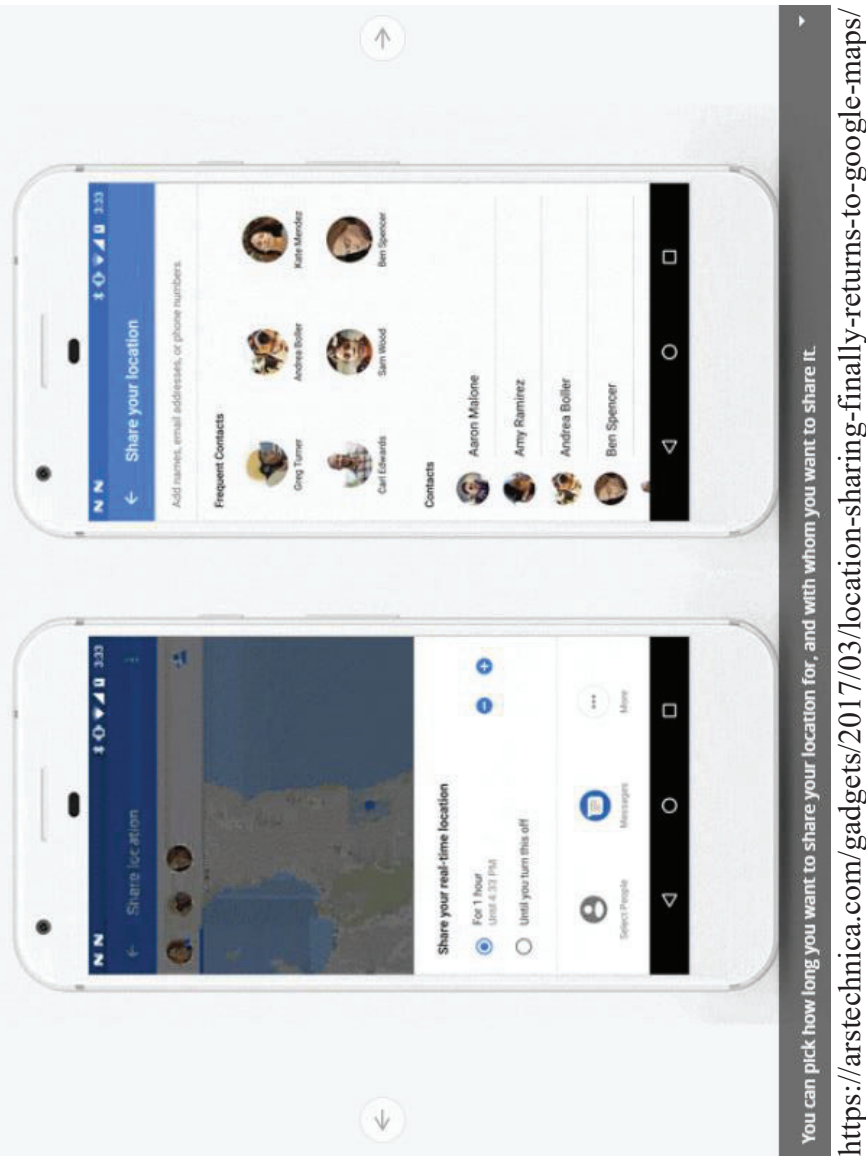
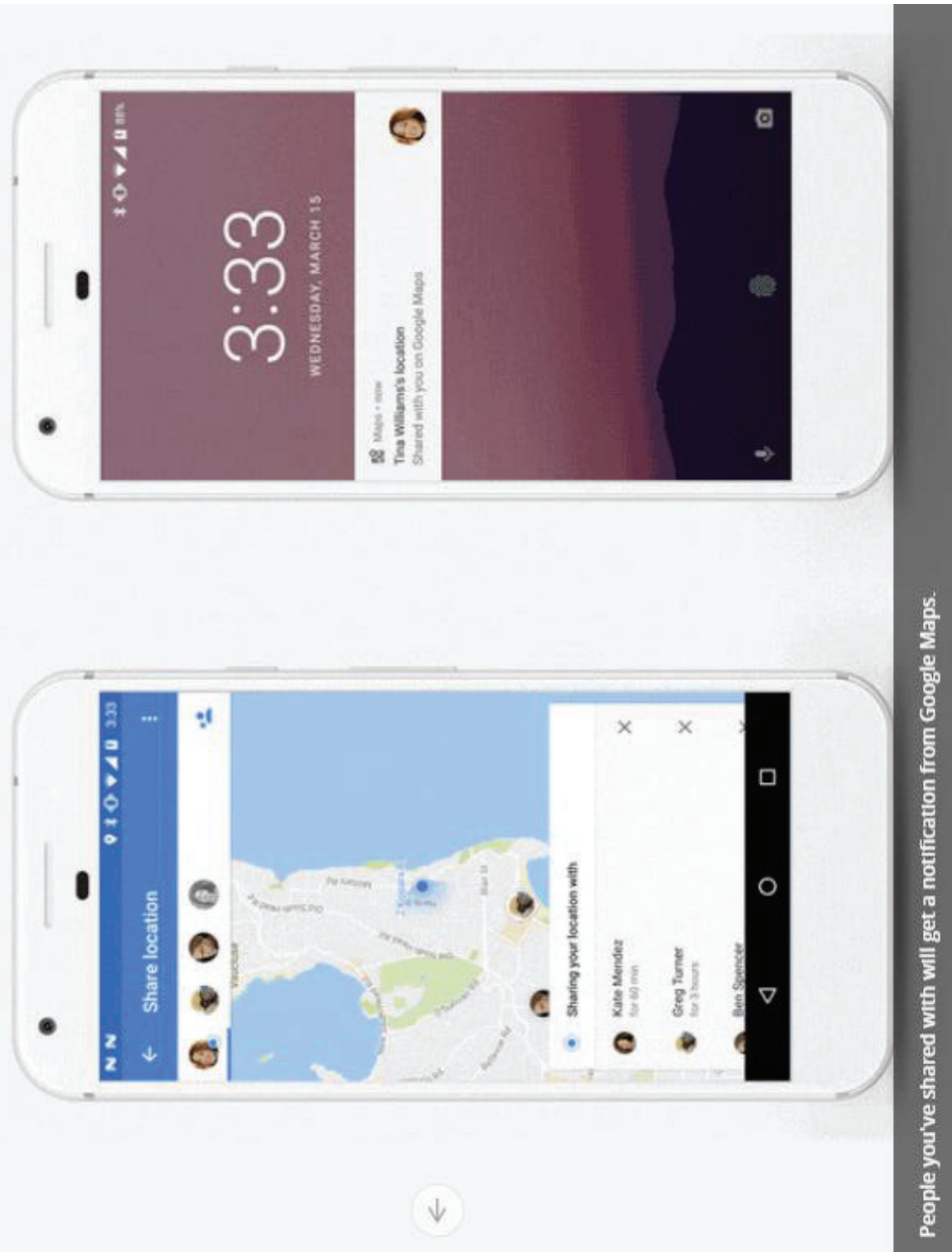


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

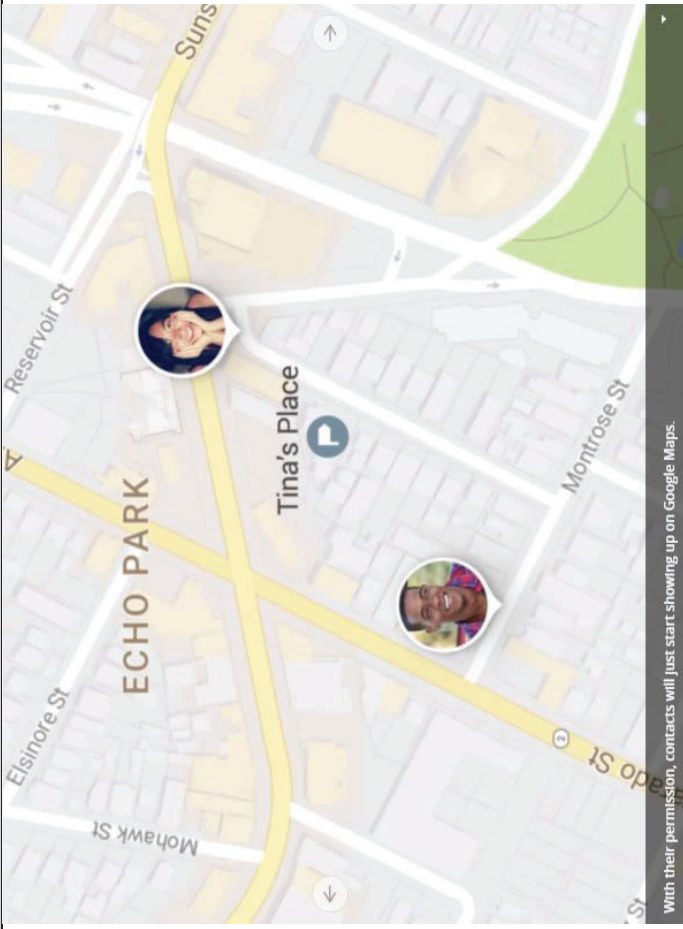


<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



<https://arstechnica.com/gadgets/2017/03/location-sharing-finally-returns-to-google-maps/>

Stop sharing

1. Open the Google Maps app .
2. Tap the Menu  > **Share location**.
3. Next to the person with whom you want to stop sharing, tap Remove .



Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Share your E.T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
2. Set a driving destination. [Learn how to navigate to a place.](#)
3. After you start navigation, tap **More**  > **Share trip progress**.
4. Choose a person from the list.
5. Tap **Share**.
6. Location Sharing will stop when you reach your destination or stop navigating.

• To stop sharing before you arrive, tap **More**  > **Stop sharing**.

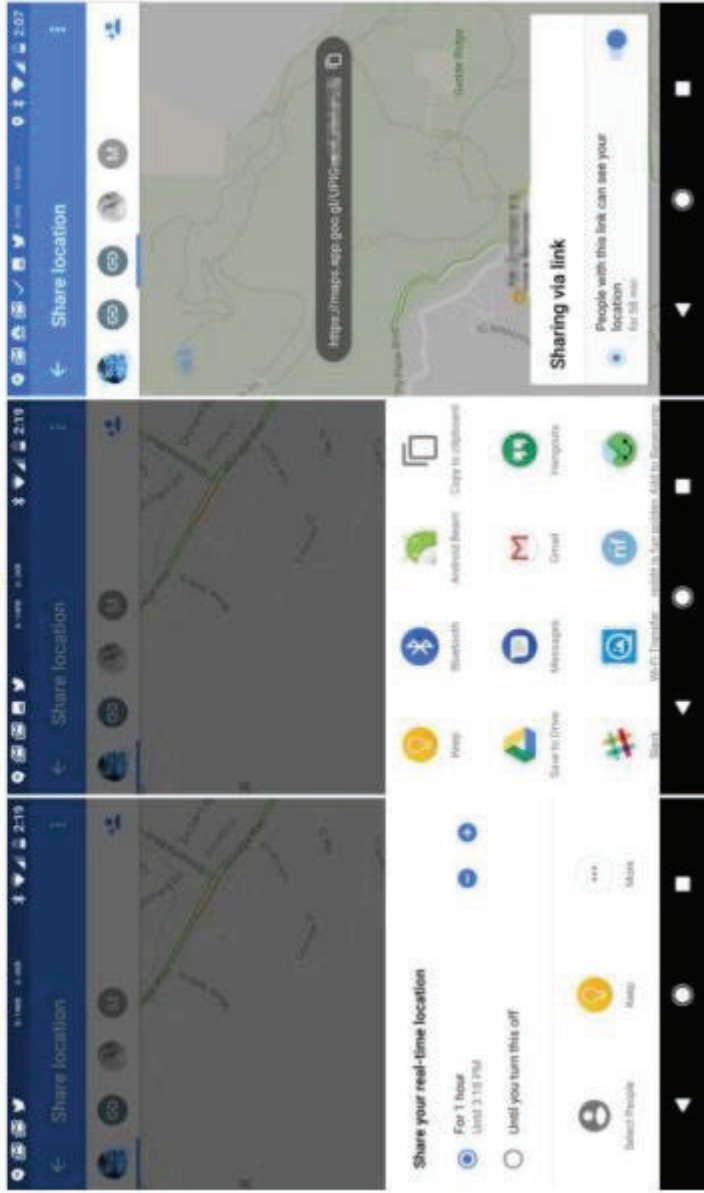
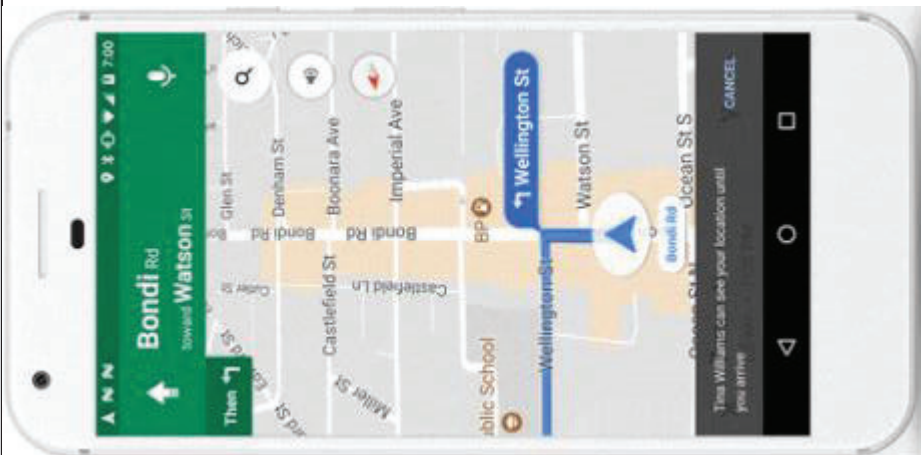


Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

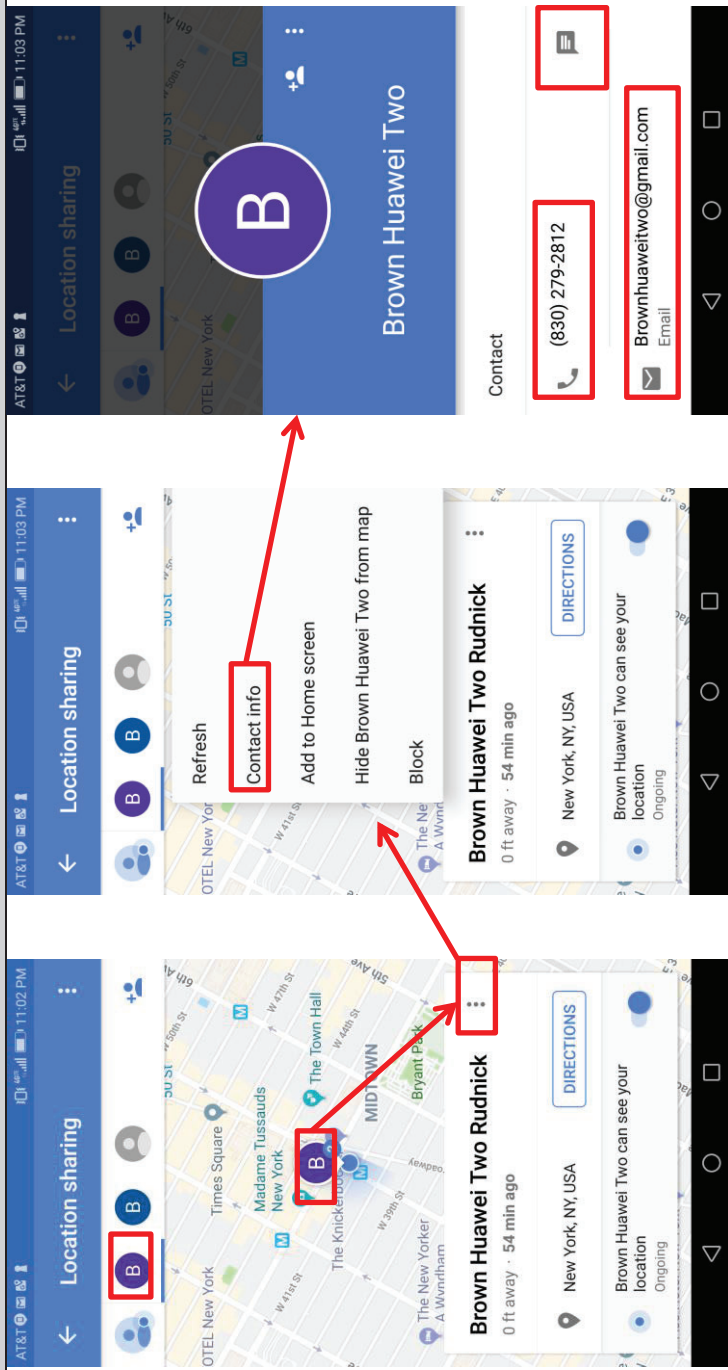


Exemplary Maps Screenshots:

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE



Exemplary Source Code:

The above functionality is performed at least in part by the following publicly available source code and/or source code that invokes or is invoked by the following source code (or a substantially similar copy compiled and loaded onto the Accused Products by ZTE). AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available. AGIS reserves the right to supplement these contentions with additional source code as discovery progresses and as additional source code is made available.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

44 * Class that sends chat message via SMS.
45 *
46 * The interface emulates a blocking sending similar to making an HTTP request.
47 * It calls the SmsManager to send a (potentially multipart) message and waits
48 * on the sent status on each part. The waiting has a timeout so it won't wait
49 * forever. Once the sent status of all parts received, the call returns.
50 * A successful sending requires success status for all parts. Otherwise, we
51 * pick the highest level of failure as the error for the whole message, which
52 * is used to determine if we need to retry the sending.
53 */
54 public class SmsSender {
55     private static final String TAG = LogUtil.BUGLE_TAG;
56
57     public static final String EXTRA_PART_ID = "part_id";
58
59     /*
60     * A map for pending sms messages. The key is the random request UUID.
61     */
62     private static ConcurrentHashMap<Uri, SendResult> sPendingMessageMap =
63         new ConcurrentHashMap<Uri, SendResult>();
64
65     private static final Random RANDOM = new Random();
66
67     // Whether we should send multipart SMS as separate messages
68     private static Boolean sSendMultipartSmsAsSeparateMessages = null;
69

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

253 // Actually sending the message using SmsManager
254 private static void sendInternal(final Context context, final int subId, String dest,
255     final ArrayList<String> messages, final String serviceCenter,
256     final boolean requireDeliveryReport, final Uri messageUri) throws SmsException {
257     Assert.notNull(context);
258     final SmsManager smsManager = PhoneUtils.get(subId).getSmsManager();
259     final int messageCount = messages.size();
260     final ArrayList<PendingIntent> deliveryIntents = new ArrayList<PendingIntent>(messageCount);
261     final ArrayList<PendingIntent> sentIntents = new ArrayList<PendingIntent>(messageCount);
262     for (int i = 0; i < messageCount; i++) {
263         // Make pending intents different for each message part
264         final int partId = (messageCount <= 1 ? 0 : i + 1);
265         if (requireDeliveryReport && (i == (messageCount - 1))) {
266             // TODO we only care about the delivery status of the last part
267             // Shall we have better tracking of delivery status of all parts?
268             deliveryIntents.add(PendingIntent.getBroadcast(
269                 context,
270                 partId,
271                 getSendStatusIntent(context, SendStatusReceiver.MESSAGE_DELIVERED_ACTION,
272                     messageUri, partId, subId),
273                 0/*flag*/));
274         } else {
275             deliveryIntents.add(null);
276         }
277         sentIntents.add(PendingIntent.getBroadcast(
278             context,
279             partId,
280             getSendStatusIntent(context, SendStatusReceiver.MESSAGE_SENT_ACTION,
281                 messageUri, partId, subId),
282             0/*flag*/));
283     }
284     if (sSendMultiPartSmsAsSeparateMessages == null) {
285         sSendMultiPartSmsAsSeparateMessages = MmsConfig.get(subId)
286             .setSendMultiPartSmsAsSeparateMessages();
287     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

288     try {
289         if (sSendMultipartSmsAsSeparateMessages) {
290             // If multipart sms is not supported, send them as separate messages
291             for (int i = 0; i < messageCount; i++) {
292                 smsManager.sendMessage(dest,
293                     serviceCenter,
294                     messages.get(i),
295                     sentIntents.get(i),
296                     deliveryIntents.get(i));
297             }
298         } else {
299             smsManager.sendMultipartTextMessage(
300                 dest, serviceCenter, messages, sentIntents, deliveryIntents);
301         }
302     } catch (final Exception e) {
303         throw new SmsException("SmsSender: caught exception in sending " + e);
304     }
305 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/SmsSender.java>

```

56 * Class that receives incoming SMS messages through android.provider.Telephony.SMS_RECEIVED
57 *
58 * This class serves two purposes:
59 * - Process phone verification SMS messages
60 * - Handle SMS messages when the user has enabled us to be the default SMS app (Pre-KLP)
61 */
62 public final class SmsReceiver extends BroadcastReceiver {
63     private static final String TAG = LogUtil.BUGLE_TAG;
64
65     private static ArrayList<Pattern> sIgnoreSmsPatterns;
66

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

200 public static void deliverSmsMessages(final Context context, final int subId,
201     final int errorCode, final android.telephony.SmsMessage[] messages) {
202     final ContentValues messageValues =
203         MmsUtils.parseReceivedSmsMessage(context, messages, errorCode);
204
205     LogUtil.v(TAG, "SmsReceiver.deliverSmsMessages");
206
207     final long nowInMillis = System.currentTimeMillis();
208     final long receivedTimestampMs = MmsUtils.getMessageDate(messages[0], nowInMillis);
209
210     messageValues.put(Sms.Inbox.DATE, receivedTimestampMs);
211     // Default to unread and unseen for us but ReceiveSmsMessageAction will override
212     // seen for the telephony db.
213     messageValues.put(Sms.Inbox.READ, 0);
214     messageValues.put(Sms.Inbox.SEEN, 0);
215     if (OsUtil.isAtLeastL_MR1()) {
216         messageValues.put(Sms.SUBSCRIPTION_ID, subId);
217     }
218
219     if (messages[0].getMessageClass() == android.telephony.SmsMessage.MessageClass.CLASS_0 ||
220         DebugUtils.isDebugEnabled()) {
221         Factory.get().getUIIntents().launchClassZeroActivity(context, messageValues);
222     } else {
223         final ReceiveSmsMessageAction action = new ReceiveSmsMessageAction(messageValues);
224         action.start();
225     }
226 }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 228 @Override 229 public void onReceive(final Context context, final Intent intent) { 230 LogUtil.v(TAG, "SmsReceiver.onReceive " + intent); 231 // On KLP+ we only take delivery of SMS messages in SmsDeliverReceiver. 232 if (PhoneUtils.getDefault().isSmsEnabled()) { 233 final String action = intent.getAction(); 234 if (OsUtil.isSecondaryUser() && 235 (Telephony.Sms.Intents.SMS_RECEIVED_ACTION.equals(action) 236 // TODO: update this with the actual constant from Telephony 237 "android.provider.Telephony.MMS_DOWNLOADED".equals(action))) { 238 postNewMessageSecondaryUserNotification(); 239 } else if (!OsUtil.isAtLeastKLP()) { 240 deliverSmsIntent(context, intent); 241 } 242 } 243 } </pre> |
| | <p>https://android.googlesource.com/platform/packages/apps/Messaging/+/nougat-mr1-release/src/com/android/messaging/receiver/SmsReceiver.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 52 * Class that sends chat message via MMS. 53 * 54 * The interface emulates a blocking send similar to making an HTTP request. 55 */ 56 public class MmsSender { 57 private static final String TAG = LogUtil.BUGLE_TAG; 58 59 /** 60 * Send an MMS message. 61 * 62 * @param context Context 63 * @param messageUri The unique URI of the message for identifying it during sending 64 * @param sendReq The SendReq PDU of the message 65 * @throws MmsFailureException 66 */ 67 public static void sendMms(final Context context, final int subId, final Uri messageUri, 68 final SendReq sendReq, final Bundle sentIntentExtras) throws MmsFailureException { 69 sendMms(context, 70 subId, 71 messageUri, 72 null /* locationUrl */, 73 sendReq, 74 true /* responseImportant */, 75 sentIntentExtras); 76 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| US9408055B2 | ZTE |
|-------------|--|
| | <pre> 240 * Download an MMS message. 241 * 242 * @param context Context 243 * @param contentLocation The url of the MMS message 244 * @throws MmsFailureException 245 * @throws InvalidHeaderValueException 246 */ 247 public static void downloadMms(final Context context, final int subId, 248 final String contentLocation, Bundle extras) throws MmsFailureException, 249 InvalidHeaderValueException { 250 final Uri requestUri = Uri.parse(contentLocation); 251 final Uri contentUri = MmsFileProvider.buildRawMmsUri(); 252 253 final Intent downloadedIntent = new Intent(SendStatusReceiver.MMS_DOWNLOADED_ACTION, 254 requestUri, 255 context, 256 SendStatusReceiver.class); 257 downloadedIntent.putExtra(SendMessageAction.EXTRA_CONTENT_URI, contentUri); 258 if (extras != null) { 259 downloadedIntent.putExtras(extras); 260 } 261 final PendingIntent downloadedPendingIntent = PendingIntent.getBroadcast(262 context, 263 0 /*request code*/, 264 downloadedIntent, 265 PendingIntent.FLAG_UPDATE_CURRENT); 266 267 MmsManager.downloadMultimediaMessage(subId, context, contentLocation, contentUri, 268 downloadedPendingIntent); 269 } </pre> <p>https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/com/android/messaging/sms/MmsSender.java</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

97 * Execute an MMS HTTP request, either a POST (sending) or a GET (downloading)
98 *
99 * @param urlString The request URL, for sending it is usually the MMSC, and for downloading
100 *   it is the message URL
101 * @param pdu For POST (sending) only, the PDU to send
102 * @param method HTTP method, POST for sending and GET for downloading
103 * @param isProxySet Is there a proxy for the MMSC
104 * @param proxyHost The proxy host
105 * @param proxyPort The proxy port
106 * @param mmsConfig The MMS config to use
107 * @param userAgent The user agent header value
108 * @param uaProfUrl The UA Prof URL header value
109 * @return The HTTP response body
110 * @throws MmsHttpException For any failures
111 */
112 public byte[] execute(String urlString, byte[] pdu, String method, boolean isProxySet,
113 String proxyHost, int proxyPort, Bundle mmsConfig, String userAgent, String uaProfUrl)
114     throws MmsHttpException {
115     Log.d("MmsService.TAG", "HTTP: " + method + " + Utils.redirectUrlForNonVerbose(urlString)
116         + (isProxySet ? (" proxy=" + proxyHost + ":" + proxyPort) : ""))
117         + ", PDU size=" + (pdu != null ? pdu.length : 0));
118     checkMethod(method);
119     HttpURLConnection connection = null;
120     try {
121         Proxy proxy = Proxy.NO_PROXY;
122         if (isProxySet) {
123             proxy = new Proxy(Proxy.Type.HTTP, new InetSocketAddress(proxyHost, proxyPort));
124         }
125         final URL url = new URL(urlString);
126         // Now get the connection
127         connection = (HttpURLConnection) url.openConnection(proxy);
128         connection.setDoInput(true);
129         connection.setConnectTimeout(
130             mmsConfig.getInt(CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT,
131                 CarrierConfigValuesLoader.CONFIG_HTTP_SOCKET_TIMEOUT_DEFAULT));
131     }

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

132 // ----- COMMON HEADERS -----
133 // Header: Accept
134 connection.setRequestProperty(HEADER_ACCEPT, HEADER_VALUE_ACCEPT);
135 // Header: Accept-Language
136 connection.setRequestProperty(
137     HEADER_ACCEPT_LANGUAGE, getAcceptLanguage(Locale.getDefault()));
138 // Header: User-Agent
139 Log.i(MmsService.TAG, "HTTP: User-Agent=" + userAgent);
140 connection.setRequestProperty(HEADER_USER_AGENT, userAgent);
141 // Header: x-wap-profile
142 final String uaProfUrlTagName = mmsConfig.getString(
143     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME,
144     CarrierConfigValuesLoader.CONFIG_UA_PROF_TAG_NAME_DEFAULT);
145 if (uaProfUrl != null) {
146     Log.i(MmsService.TAG, "HTTP: UaProfUrl=" + uaProfUrl);
147     connection.setRequestProperty(uaProfUrlTagName, uaProfUrl);
148 }
149 // Add extra headers specified by mms_config.xml's httpparams
150 addExtraHeaders(connection, mmsConfig);
151 // Different stuff for GET and POST
152 if (METHOD_POST.equals(method)) {
153     if (pdu == null || pdu.length < 1) {
154         Log.e(MmsService.TAG, "HTTP: empty pdu");
155         throw new MmsHttpException(0/*status*/, "Sending empty PDU");
156     }
157     connection.setDoOutput(true);
158     connection.setRequestMethod(METHOD_POST);
159     if (mmsConfig.getBoolean(
160         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER,
161         CarrierConfigValuesLoader.CONFIG_SUPPORT_HTTP_CHARSET_HEADER_DEFAULT)) {
162         connection.setRequestProperty(HEADER_CONTENT_TYPE,
163             HEADER_VALUE_CONTENT_TYPE_WITH_CHARSET);
164     } else {
165         connection.setRequestProperty(HEADER_CONTENT_TYPE,
166             HEADER_VALUE_CONTENT_TYPE_WITHOUT_CHARSET);

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

167     }
168     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
169         logHttpHeaders(connection.getRequestProperties());
170     }
171     connection.setFixedLengthStreamingMode(pdu.length);
172     // Sending request body
173     final OutputStream out =
174         new BufferedOutputStream(connection.getOutputStream());
175     out.write(pdu);
176     out.flush();
177     out.close();
178     } else if (METHOD_GET.equals(method)) {
179         if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
180             logHttpHeaders(connection.getRequestProperties());
181         }
182         connection.setRequestMethod(METHOD_GET);
183     }
184     // Get response
185     final int responseCode = connection.getResponseCode();
186     final String responseMessage = connection.getResponseMessage();
187     Log.d(MmsService.TAG, "HTTP: " + responseCode + " " + responseMessage);
188     if (Log.isLoggable(MmsService.TAG, Log.VERBOSE)) {
189         logHttpHeaders(connection.getHeaderFields());
190     }
191     if (responseCode / 100 != 2) {
192         throw new MmsHttpException(responseCode, responseMessage);
193     }
194     final InputStream in = new BufferedInputStream(connection.getInputStream());
195     final ByteArrayOutputStream byteOut = new ByteArrayOutputStream();
196     final byte[] buf = new byte[4096];
197     int count = 0;
198     while ((count = in.read(buf)) > 0) {
199         byteOut.write(buf, 0, count);
200     }
201     in.close();
202     final byte[] responseBody = byteOut.toByteArray();
203     Log.d(MmsService.TAG, "HTTP: response size="
204         + (responseBody != null ? responseBody.length : 0));
205     return responseBody;

```

<https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/MmsHttpClient.java>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

```

38 * Request to send an MMS
39 */
40 class SendRequest extends MmsRequest {
41     // Max send response PDU size in bytes (exceeding this may cause problem with
42     // system intent delivery).
43     private static final int MAX_SEND_RESPONSE_SIZE = 1000 * 1024;
44
45     private byte[] mPduData;
46
47     SendRequest(final String locationUrl, final Uri pduUri, final PendingIntent sentIntent) {
48         super(locationUrl, pduUri, sentIntent);
49     }
50
51     @Override
52     protected boolean loadRequest(final Context context, final Bundle mmsConfig) {
53         mPduData = readPduFromContentUri(
54             context,
55             mPduUri,
56             mmsConfig.setInt(
57                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE,
58                 CarrierConfigValuesLoader.CONFIG_MAX_MESSAGE_SIZE_DEFAULT));
59         return (mPduData != null);
60     }
61
62     @Override
63     protected boolean transferResponse(final Context context, final Intent fillIn,
64         final byte[] response) {
65         // SendConf pdus are always small and can be included in the intent
66         if (response != null && fillIn != null) {
67             if (response.length > MAX_SEND_RESPONSE_SIZE) {
68                 // If the response PDU is too large, it won't be able to fit in
69                 // the PendingIntent to be transferred via system IPC.
70                 return false;
71             }
72             fillIn.putExtra(SmsManager.EXTRA_MMS_DATA, response);
73         }
74         return true;
75     }

```

B-983

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 219 1585">ZTE</p> <p data-bbox="235 504 300 1585">https://android.googlesource.com/platform/packages/apps/Messaging/+nougat-mr1-release/src/android/support/v7/mms/SendRequest.java</p> <div data-bbox="349 357 397 1564"><pre>public static LocationRequest create ()</pre></div> <p data-bbox="422 1071 446 1564">Create a location request with default parameters.</p> <p data-bbox="479 462 544 1564">Default parameters are for a block accuracy, slowly updated location. It can then be adjusted as required by the applications before passing to the FusedLocationProviderApi.</p> <p data-bbox="560 1470 584 1564">Returns</p> <ul data-bbox="609 1281 633 1543" style="list-style-type: none">• a new location request <p data-bbox="649 304 673 1585">https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|-----|
| US9408055B2 | ZTE |
| <p>public static final int PRIORITY_BALANCED_POWER_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request "block" level accuracy.</p> <p>Block level accuracy is considered to be about 100 meter accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 102</p> <p>public static final int PRIORITY_HIGH_ACCURACY</p> <p>Used with <code>setPriority(int)</code> to request the most accurate locations available.</p> <p>This will return the finest location available.</p> <p>Constant Value: 100</p> <p>public static final int PRIORITY_LOW_POWER</p> <p>Used with <code>setPriority(int)</code> to request "city" level accuracy.</p> <p>City level accuracy is considered to be about 10km accuracy. Using a coarse accuracy such as this often consumes less power.</p> <p>Constant Value: 104</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationRequest</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <pre> public Task<Location> getLastLocation () Returns the best most recent location currently available. If a location is not available, which should happen very rarely, null will be returned. The best accuracy available while respecting the location permissions will be returned. This method provides a simplified way to get location. It is particularly well suited for applications that do not require an accurate location and that do not want to maintain extra logic for location updates. public Task<LocationAvailability> getLocationAvailability () Returns the availability of location data. When isLocationAvailable() returns true, then the location returned by getLastLocation() will be reasonably up to date within the hints specified by the active LocationRequest s. If the client isn't connected to Google Play services and the request times out, null is returned. Note it's always possible for getLastLocation() to return null even when this method returns true (e.g. location settings were disabled between calls). https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient </pre> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | | | |
|---|--|----------------------|---------------------------------------|-----------------------|--|---------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | | | |
| <p><code>public Task<Void> requestLocationUpdates (LocationRequest request, LocationCallback callback, Looper looper)</code></p> <p>Requests location updates with a callback on the specified Looper thread.</p> <p>This method is suited for the foreground use cases. For background use cases, the <code>PendingIntent</code> version of the method is recommended, see <code>requestLocationUpdates(LocationRequest, PendingIntent)</code>.</p> <p>Any previous <code>LocationRequests</code> registered on this <code>LocationListener</code> will be replaced.</p> <p>This call will keep the Google Play services connection active, so make sure to call <code>removeLocationUpdates(LocationCallback)</code> when you no longer need it, otherwise you lose the benefits of the automatic connection management.</p> <p>Callbacks for <code>LocationCallback</code> will be made on the specified thread, which must already be a prepared looper thread.</p> <p>Parameters</p> <table border="1"> <tr> <td><code>request</code></td> <td>The location request for the updates.</td> </tr> <tr> <td><code>callback</code></td> <td>The callback for the location updates.</td> </tr> <tr> <td><code>looper</code></td> <td>The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread.</td> </tr> </table> <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | <code>request</code> | The location request for the updates. | <code>callback</code> | The callback for the location updates. | <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. |
| <code>request</code> | The location request for the updates. | | | | | | |
| <code>callback</code> | The callback for the location updates. | | | | | | |
| <code>looper</code> | The Looper object whose message queue will be used to implement the callback mechanism, or null to make callbacks on the calling thread. | | | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | | | | | |
|--|---|----------------|---------------------------------------|-----------------------|---|
| <p>US9408055B2</p> | <p>ZTE</p> | | | | |
| <p>public Task<Void> requestLocationUpdates (LocationRequest request, PendingIntent callbackIntent)</p> <p>Requests location updates with a callback on the specified PendingIntent.</p> <p>This method is suited for the background use cases, more specifically for receiving location updates, even when the app has been killed by the system. In order to do so, use a PendingIntent for a started service. For foreground use cases, the LocationCallback version of the method is recommended, see requestLocationUpdates(LocationRequest, LocationCallback, Looper).</p> <p>Any previously registered requests that have the same PendingIntent (as defined by equals(Object)) will be replaced by this request.</p> <p>Both LocationResult and LocationAvailability are sent to the given PendingIntent. You can extract data from an Intent using hasResult(Intent), extractResult(Intent), hasLocationAvailability(Intent), and extractLocationAvailability(Intent).</p> <p>Parameters</p> <table border="1"> <tr> <td data-bbox="836 1260 901 1564">request</td> <td data-bbox="836 357 901 1260">The location request for the updates.</td> </tr> <tr> <td data-bbox="901 1260 966 1564">callbackIntent</td> <td data-bbox="901 357 966 1260">A pending intent to be sent for each location update.</td> </tr> </table> <p>Returns</p> <ul style="list-style-type: none"> • a Task for the call, check isSuccessful() to determine if it was successful. <p>https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient</p> | | request | The location request for the updates. | callbackIntent | A pending intent to be sent for each location update. |
| request | The location request for the updates. | | | | |
| callbackIntent | A pending intent to be sent for each location update. | | | | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--------------------|------------|
| <p>US9408055B2</p> | <p>ZTE</p> |
|--------------------|------------|

| | |
|---|--|
| <p>public void onLocationAvailability (LocationAvailability locationAvailability)</p> <p>Called when there is a change in the availability of location data.</p> <p>When <code>isLocationAvailable()</code> returns <code>false</code> you can assume that location will not be returned in <code>onLocationResult(LocationResult)</code> until something changes in the device's settings or environment. Even when <code>isLocationAvailable()</code> returns <code>true</code> the <code>onLocationResult(LocationResult)</code> may not always be called regularly, however the device location is known and both the most recently delivered location and <code>getLastLocation(GoogleApiClient)</code> will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>locationAvailability The current status of location availability.</p> <p>public void onLocationResult (LocationResult result)</p> <p>Called when device location information is available.</p> <p>The most recent location returned by <code>getLastLocation()</code> is not guaranteed to be immediately fresh, but will be reasonably up to date given the hints specified by the active <code>LocationRequest</code>s.</p> <p>Parameters</p> <p>result The latest location result available.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/location/LocationCallback</p> <p>public abstract void onLocationChanged (Location location)</p> <p>Called when the location has changed.</p> <p>Parameters</p> <p>location The updated location.</p> | |
|---|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="196 1520 220 1583">ZTE</p> <p data-bbox="233 302 261 1583">https://developers.google.com/android/reference/com/google/android/gms/location/LocationListener</p> <p data-bbox="315 1283 342 1566">Public Constructors</p> <p data-bbox="415 1173 443 1556">public MapView (Context context)</p> <p data-bbox="505 968 532 1556">public MapView (Context context, AttributeSet attrs)</p> <p data-bbox="594 831 621 1556">public MapView (Context context, AttributeSet attrs, int defStyle)</p> <p data-bbox="683 852 711 1556">public MapView (Context context, GoogleMapOptions options)</p> <p data-bbox="748 428 776 1583">https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|--|
| <p>US9408055B2</p> | <p>ZTE</p> |
| <p>[54H] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p><code>public void getMapAsync (OnMapReadyCallback callback)</code></p> <p>Returns a non-null instance of the <code>GoogleMap</code>, ready to be used.</p> <p>Note that:</p> <ul style="list-style-type: none"> • This method must be called from the main thread. • The callback will be executed in the main thread. • In the case where Google Play services is not installed on the user's device, the callback will not be triggered until the user installs it. • The <code>GoogleMap</code> object provided by the callback is non-null. <p>Parameters</p> <p><code>callback</code> The callback object that will be triggered when the map is ready to be used.</p> <p><code>public final void onCreate (Bundle savedInstanceState)</code></p> <p>You must call this method from the parent Activity/Fragment's corresponding method.</p> <p>https://developers.google.com/android/reference/com/google/android/gms/maps/MapView</p> |
| <p>[54H] receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices;</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of receiving user input via user interaction with the interactive display of the first device, the user input specifying a location and a symbol corresponding to an entity other than the first device and the second devices. See claims 1[G], 28[G], and 41[G], which are incorporated herein by reference in their entirety.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|--|------------|
| US9408055B2 | ZTE |
| <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> <p>http://www.zte.com.cn/en/products/anyservice/industr_enterprise/m2m/201112/20111208_352166.html</p> | |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|--|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="259 703 300 1218">ZTE Mobile Location Service System</p> <p data-bbox="332 913 365 1039">2004-01-31</p> <p data-bbox="430 1396 462 1564">I. Introduction</p> <p data-bbox="470 367 673 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="690 367 803 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="820 199 893 1585">http://www.en.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the top.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products


US9408055B2

ZTE

You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list.
Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.

Searching for local places

Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.

1. From the home screen, tap  > **Maps**.
2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location.
3. Tap the search box at the top.
4. Tap the **Explore nearby** card and choose one option in the new screen. Results will appear on cards.
5. Tap a location to see it on the map or get directions.

You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others.

Note: The Explore nearby feature is not available for all areas.

Google Search

You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.


Searching with text

You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.

1. From the home screen, tap  > **Google**.
2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box.

Searching by speaking

You can also search the Web or perform certain tasks by speaking.

1. From the home screen, tap  > **Google**.
2. Tap the microphone icon to the right of the search box.

Note: You can also tap  > **Voice Search**.

3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated.

Changing search settings

Open the **Google** app and tap  > **Settings** to set phone search options, voice recognition and output settings, and to change privacy settings for your account.

Regarding Google Maps, Google Latitude, Google Plus, Google Hangouts, Google Messages, and Android Messenger, the Accused Products are configured to allow a user of the first device to interact with the display, to select a user, contact, or device, and to select an action to be performed, such as: sending a message, initiating a call, initiating a data conference, sharing a location, stop sharing a location, block a user from location sharing, sending a location, requesting a location, or sending other data. When an action is specified, data is sent from the first device to the second device via a server. In an example, using Google Maps, a user can interact with the display to specify a location that does not correspond to the first or second devices. Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices.

Selection with Markers:

<https://developers.google.com/maps/documentation/android-api/marker>

Queries with Geo Tagging database:

<https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient>

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD

Share a map or location

1. Open the Google Maps app.
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share. If you don't see this, tap More.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Share your E. T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app.
2. Set a driving destination. Learn how to navigate to a place.
3. After you start navigation, tap More. Share trip progress.
4. Choose a person from the list.
5. Tap Share.
6. Location Sharing will stop when you reach your destination or stop navigating.

To stop sharing before you arrive, tap More. Stop sharing.

<https://support.google.com/maps/answer/7326816?co=GENIE.Platform%3DAndroid&hl=en>

Markers (adding location information to the link associated with the database):

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|---|---|
| <p>US9408055B2</p> | <p>ZTE</p> <pre>static final LatLng PERTH = new LatLng(-31.90, 115.86); Marker perth = mMap.addMarker(new MarkerOptions() .position(PERTH) .draggable(true));</pre> |
| <p>[54] and based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H], 28[H], and 41[H], which are incorporated herein by reference in their entirety.</p> | <p>ZTE infringes directly and/or indirectly by performing, inducing others to perform, and/or contributing to the performance of [] based on the user input, adding the user-specified symbol to the interactive display at a position on the interactive map corresponding to the user-specified location, and transmitting the user-specified symbol and location to the second devices for addition of the user-specified symbol to respective interactive displays of the second devices at respective positions on respective interactive maps corresponding to the user-specified location. See claims 1[H], 28[H], and 41[H], which are incorporated herein by reference in their entirety.</p> <p style="text-align: center;">Location Based Service (LBS)</p> <p>The Location Based Service (LBS) is a value-added service that obtains information about positions (longitude and latitude) of mobile terminal users, and provides users with corresponding service under the help of the digital map platform. As mobile phones become an indispensable part of people's life, the LBS is increasingly important. The LBS enables terminal users to easily know where they are, and inquire about their vicinities by using their terminals: where am I? Where is the nearest hospital? Which banks are around? How can I get there from here? Where is my best friend? Other services include first aid, tracking an elderly and vehicle management. The most fascinating feature of the LBS is that it enables you to send the right information to a right person in a right place and at a right time. The ZTE LBS product supports a number of location techniques such as CELL ID, CELL ID+TA, and GPSONE, and provides an end-to-end full LBS solution.</p> |

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

| | |
|-------------|---|
| US9408055B2 | <p data-bbox="194 1512 227 1585">ZTE</p> <p data-bbox="235 357 267 1585">http://www.zte.com.cn/en/products/anyervice/industr_enterprise/m2m/201112/20111208_352166.html</p> <p data-bbox="341 703 381 1218">ZTE Mobile Location Service System</p> <p data-bbox="414 913 446 1039">2004-01-31</p> <p data-bbox="511 1396 544 1564">I. Introduction</p> <p data-bbox="552 367 755 1564">Mobile positioning service, also called location service, is a new type of value-added service provided by mobile communication network. The service acquires the location information (i.e. longitude and latitude, mobile speed) of the mobile station by virtue of wireless positioning technology, and then provides the information to the subscribers, mobile communication network or other external entities, realizing various service applications relating to the location.</p> <p data-bbox="763 367 885 1564">The special features of CDMA system, such as soft handoff, GPS information offered by base station, and code snippet delay for distance measurement, provide the CDMA network with incomparable advantages to realize the wireless positioning service.</p> <p data-bbox="893 199 966 1585">http://www.zte.com.cn/endata/magazine/zte technologies/2003year/no14/articles/200401/t20040131_161273.html</p> |
|-------------|---|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Google+

Google+ is a social networking service offered by Google. The Google+ application on your phone enables you to manage your data, chat with online contacts, organize them in different circles, and share your selected information.

From the home screen, tap > **Google+**.

Note: For a detailed guide of Google+, open Google+ and tap > **Help**.

Getting started with Google+

Open the Google+ app and tap at the top left to use the following features:

- **Home:** See posts people have shared with you, a circle you are in, or everyone.
- **People:** Find people to add to your Google+ circles, create new circles, or view and organize people into circles based on your relationships. You can also follow content posted by people you find interesting.
- **Photos:** View photos you uploaded, photos from your posts, and more.
- **Communities:** Search and join all kinds of online communities, where people gather for the sharing and discussion of a common topic.

- **Locations:** See your friends' locations on the map and configure your location sharing settings.
 - **Hangouts:** Chat with your Google+ contacts through text messages or video chats.
 - **Events:** Manage your social calendar. You can add events, invite people, and then share photos in real time from the event.
 - **Search:** Search for content posted on Google+.
- Tap the headshot image next to your account name to view and edit your basic information, and view your posts and photos.

Google Maps

Activating location services

- To use Google Maps and find your location, you must enable location services on your phone.
1. From the home screen, tap > **System settings** > **Location access**.
 2. Select the location options you want to use.
 - Slide the **Access to my location** switch to **ON** to let apps that have your permission use your location information.
 - Check **GPS satellites** to use GPS satellites to determine your location.
 - Check **Wi-Fi & mobile network location** to use Wi-Fi and mobile networks to determine your approximate location.

Note: To let Google apps such as Maps access your location, you need to enable location access for Google apps. From the home screen, tap > **Google Settings** > **Location** and check **Access location**.

Getting your location

1. From the home screen, tap > **Maps**.
 2. Tap at the bottom.
- The map centers on a circle that indicates your location.

Searching for a location

1. From the home screen, tap > **Maps**.
2. Tap the search box on the tap.
3. Enter the address or the type of business you're looking for in the search box. If suitable suggestions appear below the search box, tap one to search for it.

Tip: You can also tap beside the search box to use voice search.

4. Tap the search key on the keyboard.
5. Swipe the search result at the bottom left or right to see other results. You can slide the result up to find more information about the location as well as options for getting directions and more.

Getting directions to your destination

- Maps can provide directions for travel by foot, public transportation, or car.
1. From the home screen, tap > **Maps**.
 2. Tap beside the search box.
 3. Select the mode of transportation and then enter a start and end point. If your location is found, it will appear in the start point field by default.
 4. Tap a suggested route to view it on the map.

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products







| | |
|---------------------------|--|
| <p>US9408055B2</p> | <p>ZTE</p> <p>You can swipe the bottom card left or right to choose other routes. Slide the bottom card up to see the route directions in a list. Note: Tap  to use Google Maps Navigation and get turn-by-turn voice directions.</p> <p>Searching for local places Google Maps helps you find all kinds of businesses and establishments around you. You can rate these places and get recommendations.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Maps. 2. Navigate the map view to the area you'd like to explore. You can explore nearby locations or pan the map to another location. 3. Tap the search box at the top. 4. Tap the Explore nearby card and choose one option in the new screen. Results will appear on cards. 5. Tap a location to see it on the map or get directions. <p>You can also slide up the bottom card for more information such as the street address, website, or street view, and options for calling, saving to your Google Account, or sharing with others. Note: The Explore nearby feature is not available for all areas.</p> <p>Google Search You can search for information on the Web or your phone using Google Search, or perform certain tasks by speaking.</p> <p>Searching with text You can search for information on the Internet or on the phone (such as apps and contacts) by entering text.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Enter the terms you want to search for in the search box and tap the search key on the keyboard, or tap a search suggestion that appears below the search box. <p>Searching by speaking You can also search the Web or perform certain tasks by speaking.</p> <ol style="list-style-type: none"> 1. From the home screen, tap  > Google. 2. Tap the microphone icon to the right of the search box. <p>Note: You can also tap  > Voice Search.</p> <ol style="list-style-type: none"> 3. Speak the terms you want to search for or the question you want to ask. When you're finished, your speech is analyzed and the search is initiated. <p>Changing search settings Open the Google app and tap  > Settings to set phone search options, voice recognition and output settings, and to change privacy settings for your account.</p> <p>A user can interact with the display to specify a location that does not correspond to the first or second devices. A user can drop a symbol pin on the specified location. A user can then share that location and transmit the location to one or more second devices using Android Messages, Google Hangouts, or another application.</p> <p>Alternatively, a user can share an ETA which includes a route, where the route is not the location of any of the first or second devices. Again, this route can be shared with users over Android Messages, Google Hangouts, or another application.</p> <p>Placing a Marker: https://developers.google.com/maps/documentation/android-api/marker based on queries with GeoTagging database: https://developers.google.com/android/reference/com/google/android/gms/location/places/GeoDataClient</p> |
|---------------------------|--|

Exhibit B for U.S. Patent No. 9,408,055 Against ZTE Accused Products

US9408055B2

ZTE

Embed a map or share a location

On your computer, you can embed a map, Street View image, driving directions, or search into your website or blog. On your computer, phone, or tablet, you can share a location or map with others over email, Google+, Facebook, Twitter, or text.

ANDROID COMPUTER IPHONE & IPAD




Share a map or location

1. Open the Google Maps app .
2. Search for a place. Or find a place on the map then touch and hold to drop a pin.
3. At the bottom, tap the place's name or address.
4. Tap Share  if you don't see this, tap More  > Share.
5. Select an app. It'll send a link that shows the place in Google Maps.

<https://support.google.com/maps/answer/144361?co=GENIE.Platform%3DAndroid&hl=en>

Share your E. T.A

After you start your drive, you can share your destination, estimated arrival time, and where you are on the route.

1. Open the Google Maps app .
 2. Set a driving destination. [Learn how to navigate to a place.](#)
 3. After you start navigation, tap More  > Share trip progress.
 4. Choose a person from the list.
 5. Tap Share.
 6. Location Sharing will stop when you reach your destination or stop navigating.
- To stop sharing before you arrive, tap More  > Stop sharing.