EXHIBIT 6

Paper No. ________

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE INC. and MOTOROLA MOBILITY LLC
Petitioners
v.

ARENDI S.A.R.L.
alleged Patent Owner

Patent 6,323,853

PETITION FOR *INTER PARTES* REVIEW UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.100 *ET. SEQ.*

TABLE OF CONTENTS

TABLE OF EXHIBITS

Exhibit No.	Exhibit Description
1001	U.S. Pat. No. 6,323,853 ("the '853 patent")
1002	Declaration of Dennis Allison
1003	U.S. Pat. No. 5,923,848 ("Goodhand")
1004	Padwick, et al., "Using Microsoft Outlook 97" (Microsoft Press) 1996 (Chapters 1, 8-9, 12-13, 19-20, 22, 24 only)
1005	U.S. Pat. No. 6,026,410 ("Allen")
1006	U.S. Pat. No. 5,644,735 ("Luciw")
1007	Bonura and Miller, "Drop Zones An Extension to LiveDocs", SIGCHI Bulletin Volume 30, Number 2 April 1998.
1008	Curriculum Vitae of Dennis Allison
1009	U.S. Pat. No. 5,754,306 ("Taylor")
1010	U.S. Patent No. 5,790,532 ("Sharma")
1011	U.S. Pat. No. 5,859,636 ("Pandit")
1012	Magnanelli, et al., "ACADEMIA: An Agent-Maintained Database based on Information Extraction from Web Documents", 14th European Meeting on Cybernetics and Systems Research on April 15, 1998.
1013	Structure of the LC Control Number, http://www.loc.gov/marc/lccn_structure.html
1014	Preliminary Infringement Analysis for Defendant Google, U.S. Pat. No. 7,496,854 (highlighting added)

NOTICE OF LEAD AND BACKUP COUNSEL

Counsel for Petitioners Motorola Mobility LLC and Google Inc.:

Lead Counsel: Matthew A. Smith (Reg. No. 49,003); **Tel:** 650.265.6109

Backup Counsel: Zhuanjia Gu (Reg. No. 51,758); **Tel:** 650.529.4752

Address: Turner Boyd LLP, 702 Marshall St., Ste. 640
Redwood City, CA 94063. **FAX:** 650.521.5931.

NOTICE OF EACH REAL-PARTY-IN-INTEREST

The real-parties-in-interest for this Petition are Google Inc. for Petitioner Google Inc. and Motorola Mobility LLC for Petitioner Motorola Mobility LLC.

NOTICE OF RELATED MATTERS

U.S. Patent No. 6,323,853 ("the '853 patent") at issue has been asserted in the U.S. District Court for the District of Delaware in the following cases: 1-12-cv-01601, 1-12-cv-01602, 1-12-cv-01599, 1-12-cv-01598, and 1-12-cv-01595, all filed on Nov. 29, 2012, and 1-13-cv-00919 and 1-13-cv-00920, filed May 22, 2013. The patent was previously asserted in the U.S. District Court for the District of Rhode Island in case no. CA No. 02-343-T, filed on July 31, 2002.

NOTICE OF SERVICE INFORMATION

Please address all correspondence to the lead counsel at the addresses shown above. Petitioners also consent to electronic service by email at the following addresses: smith@turnerboyd.com, docketing@turnerboyd.com, gu@turnerboyd.com, kent@turnerboyd.com, turner@turnerboyd.com.

GROUNDS FOR STANDING

Petitioner hereby **certifies** that the patent for which review is sought is available for *inter partes* review and that the Petitioner is not barred or estopped from requesting an *inter partes* review challenging the patent claims on the grounds identified in the petition.

STATEMENT OF PRECISE RELIEF REQUESTED

The Petitioner respectfully requests that claims 1-79 of U.S. Patent No. 6,323,853 ("the '853 patent") (Ex. 1001) be canceled based on the following grounds of unpatentability, explained in detail in the next section:

Ground 1. Claims 1-9, 11, 13-29, 38-45, 57-64, 66, 68-75, 77 and 79 are invalid under 35 USC § 103 as obvious over Goodhand.

Ground 2. Claims 6, 10, 12, 21, 27, 30-37, 42, 46-56, 61, 65, 67, 72, 76 and 78 are invalid under 35 U.S.C. § 103 as obvious over Goodhand and Padwick.

Ground 3. Claims 1, 2, 7-11, 13-17, 22-23, 28-30, 35-38, 43-46, 57, 62-66, 68, 73-77, and 79 are invalid under 35 U.S.C. § 102(e) over Allen.

THRESHOLD REQUIREMENT FOR *INTER PARTES* REVIEW

This petition presents "a reasonable likelihood that the Petitioner would prevail with respect to at least one of the claims challenged in the petition". 35 USC § 314(a), as shown in the Grounds explained below.

I. INTRODUCTION

A. Declaration of Dennis Allison

The declaration of Dennis Allison is attached as Exhibit 1002.

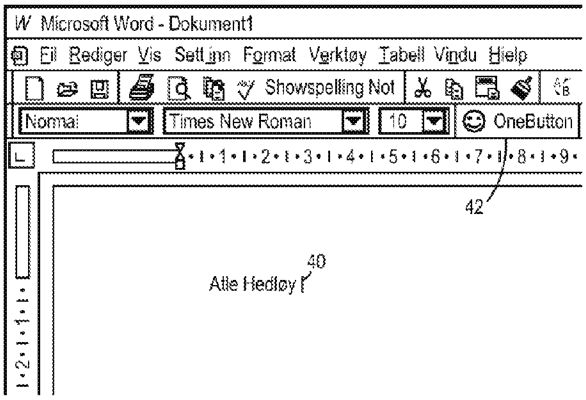
B. Technical Background

1. Overview of the '853 Patent

The disclosure of the '853 patent relates to the computerized handling of contact information. Contact information is information that is related to a person—such as the person's name, telephone number, postal address, email address, etc. Ex. 1002 at ¶ 46.

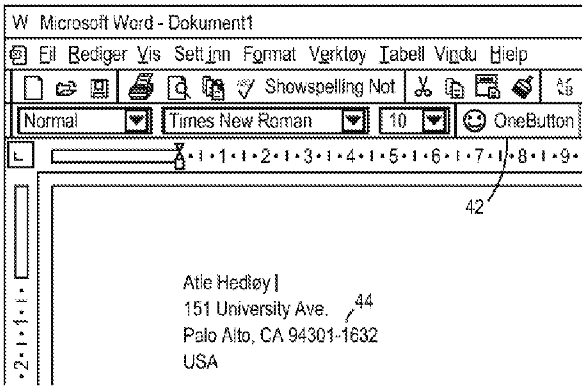
The '853 patent "handles" such contact information with a system that facilitates interaction between programs that use text documents (like word processors) and databases of contact information. Ex. 1002 at ¶ 46. Such databases can be called "contact databases" or "address books". Ex. 1002 at ¶ 46. These databases can contain information relating to people, such as their names, telephone numbers, email addresses, postal addresses, and notes relating to the person. *Id.*

The interaction between programs like word processors and contact databases can be illustrated with reference to Figures 3 and 4 of the '853 patent. These figures depict screens that a person might see when using a word processing program. *Id.* The relevant portions of the figures are shown side-by-side here:


W Microsoft Word - Dokument1
Fil Rediger Vis Sett inn Format Verktøy Tabell Vindu Hjelp
Showspelling Not
Normal
Times New Roman
10
OneButton
42
40

Atle Hedløy

Fig. 3


W Microsoft Word - Dokument1
Fil Rediger Vis Sett inn Format Verktøy Tabell Vindu Hjelp
Showspelling Not
Normal
Times New Roman
10
OneButton
42
44

Atle Hedløy
151 University Ave.
Palo Alto, CA 94301-1632
USA

Fig. 4

Figure 3 on the left shows a word processor window, in which a user has entered a name. The name is processed by the '853 patent system after the user clicks the "OneButton" 42 in the upper right part of the window. Clicking the "OneButton" causes the system to "retrieve the name… from the document" and "search[] a database for the name….". Ex. 1001 at 5:64-65. Assuming that the search finds an address associated with the name, the system then inserts the address into the word processing document, as depicted in Fig. 4 on the right. Ex. 1002 at ¶47.

The bulk of the '853 patent relates to a high-level description of operations like these. The specification describes the user taking certain actions in a GUI, which result in operations being performed on contact information. These actions can include adding a contact to a contact database, or sending an email based on the contact information. Ex. 1002 at ¶48.

The specification of the '853 patent, however, relates mainly to the end-result of contact information handling, that is, *what the user of the computer system experiences as he or she uses the system*. Exactly how these end-results are achieved is described only at the highest level. For example, the '853 patent provides no source code or pseudo code. High-level flowcharts for some embodiments are included, Ex. 1001 at Figs. 1a, 1b, 2a and 2b, but each of these is limited to a general description of the desired functionality, with no implementation detail. Ex. 1002 at ¶¶48-49.

In fact, the '853 patent relies on existing word processors and existing databases to implement its contact management method, assuming that the person of ordinary skill can fill in the detail. The methods of the '853 patent are implemented on standard well-known operating systems and ordinary commodity computer hardware, all of which were readily available well before the filing of the application leading to the '853 patent. Ex. 1002 at ¶¶48-60.

II. STATE OF THE ART AT THE CLAIMED PRIORITY DATE

In the years leading up to earliest possible priority date (Sep. 1998), numerous systems existed that used personal computers to manage personal contact information. These systems integrated sophisticated contact database technology available at the time with applications like word processors and applications that performed communications (such as email applications). Ex. 1002 at ¶26-44.

For example, systems like the one in U.S. Pat. No. 5,923,848 ("Goodhand")(Ex. 1003) had been developed for analyzing text in a document, and assisting the user in taking appropriate actions based on the information discovered. Goodhand taught identifying (upon command) nicknames or shorthands for email addresses, and then searching a contact database for corrected contact information to insert. Figures 6a-6c of Goodhand are shown at right, with highlighting added by the Petitioners to shown how correct contact information is found and inserted into the document. Ex. 1002 at ¶¶42-44.

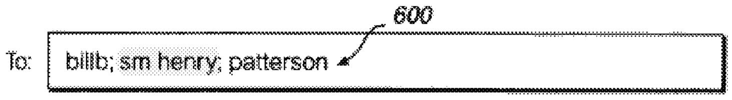


FIG.6a

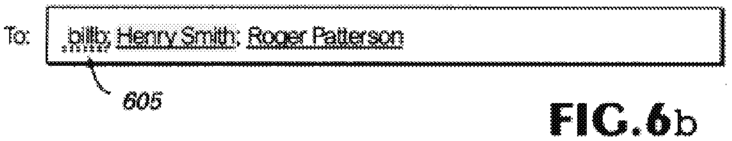


FIG.6b

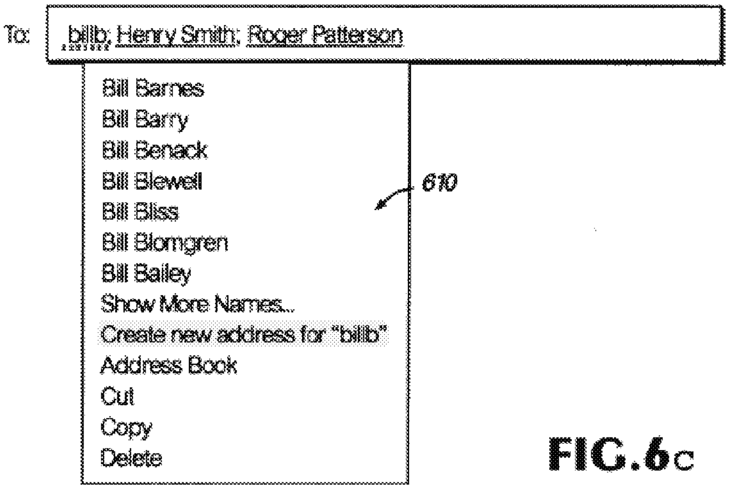
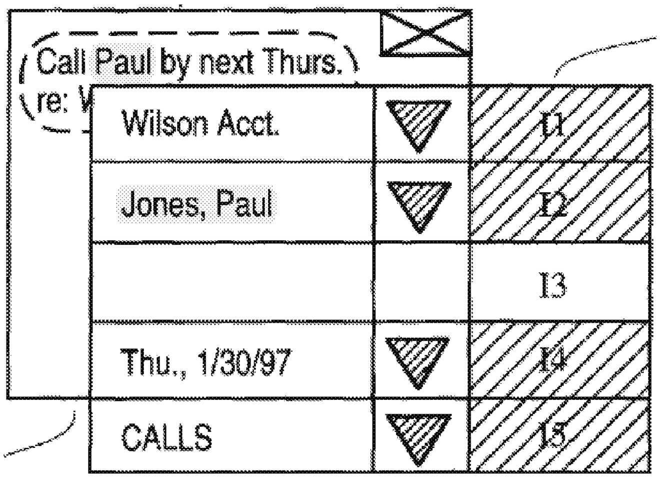


FIG.6c

Another such system is taught in U.S. Pat. No. 6,026,410 ("Allen")(Ex. 1005) which dealt with the Lotus Notes^{TM} application. Allen taught a system that analyzed text entered in an intelligent note editor, and identified certain keywords. These keywords were then mapped to contact information and other useful system



datasets. In Figure 7 (at right, highlighting added), shows the Allen system recognizing the keyword "Paul", matching the keyword with the contact database entry for "Paul Jones", and displaying the results to the user. Ex. 1002 at ¶45.

In another example, U.S. Patent No. 5,644,735 to Luciw (Ex. 1006) describes a system for detecting structures in text and using a template-based system to offer the user options for handling the data so identified. Figures 6a and 6b, which illustrate a user entering a name and having the system provide a full name, are shown below. Ex. 1002 at ¶28.

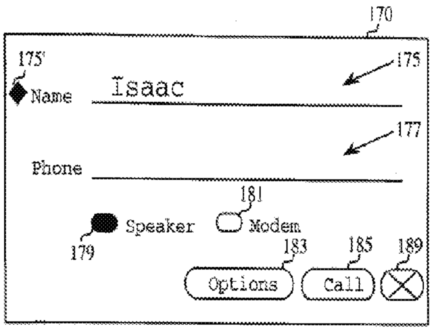


Figure 6a

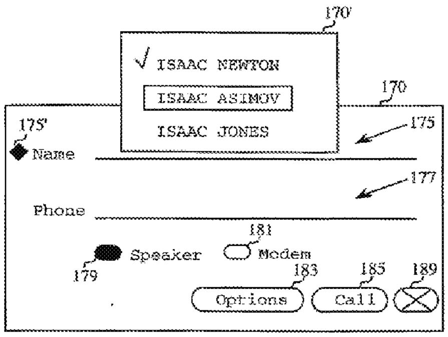
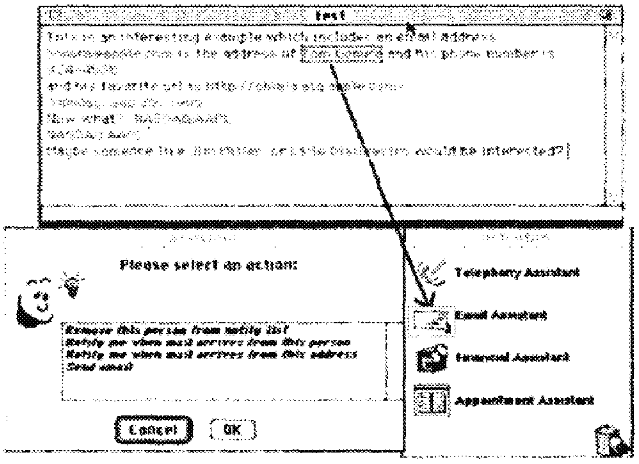


Figure 6b

Another example was the "Drop Zones" system described in an article by Bonura and Miller (Ex. 1007). Drop Zones integrated a text recognition approach akin to Luciw into common applications like word processors. The text recognition system of Drop Zones identified things like names, telephone numbers and email addresses, and allowed the designer of the system to create arbitrary tasks. The Drop Zones system also used an electronic address book to convert between different kinds of contact information, and allowed the applications to update the address book with identified contact information. Fig. 2 of the Bonura article is shown below, and depicts how a name identified in a document can be used to cause a lookup on a name to retrieve an email address, thereby allowing an email to be sent. Ex. 1002 at ¶¶29-30.



This is an interesting example which includes an email address [illegible]

Maybe someone [illegible] would be interested?

Please select an action:

Remove this person from notify list
Notify me when mail arrives from this person
Notify me when mail arrives from this address
Send email

Figure 2: A user interaction with Drop Zones

III. OVERVIEW OF THE CLAIMS AND THEIR DEPENDENCIES

The '853 patent has 79 claims, but only one independent claim. Independent claim 1 provides:

> A computerized **method for information handling** within a document created using an application program, the document including first information provided therein, the method comprising:
> **providing** a **record retrieval program**;

> **providing** an input device configured to enter an execute command which **initiates a record retrieval from an information source** using the record retrieval program;
>
> **upon a single entry of the execute command** by means of the input device:
>
> **analyzing** the document to determine if the first information is contained therein, and
>
> if the first information is contained in the document, **searching**, using the record retrieval program, the information source **for second information** associated with the first information;
>
> and when the information source includes second information associated with the first information, **performing at least one of**,
>
> (a) **displaying** the second information,
>
> (b) **inserting** the second information in the document, and
>
> (c) **completing** the first information in the document based on the second information.

As explained in the Allison Declaration, the <u>dependent claims</u> can be conceptually divided into three groups: claims 2-14, claims 15-16 and claims 17-79. Ex. 1002 at ¶¶61-75.

Claims 2-14 are dependent from claim 1, and each specify a different limitation.

Claims 15 and 16 attempt, using only two claims, to convert method claims 1-14 into "system" and "storage medium" claims. Claim 15 recites "a computer system configured to perform the steps recited in one of claims 1-14." Likewise, claim 16 recites "a storage medium storing a program for performing the steps recited in one of claims 1-14." Ex. 1002 at ¶¶61-75.

Claims 17-79 repeat the limitations of claims 8-14 in different dependency relationships. The dependency relationships are done in blocks. For example, in the first block of claims 17-22, each claim has the same limitation as claim 8, but is dependent from a different claim. Claim 17 is dependent from claim 2, claim 18 is dependent from claim 3, and so forth, up to claim 22 being dependent from claim 7. Because the content of claim 8 cannot be dependent from claim 8, the first block ends there. The next block of dependent claims is 23-29. Each of claims 23-29 recites the same limitation as claim 9, but is dependent on a different claim. Claim 23 is dependent from claim 2, claim 24 is dependent on claim 3 and so forth, up to claim 29 being dependent on claim 8. The second block ends there. Similarly, there are third, fourth, fifth, sixth and seventh blocks, corresponding to the content of claims 10-14. Ex. 1002 at ¶¶61-75.

The result of this claiming is shown in the following table:

Content of Claim:	**Dependent From Claim:**												
	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
8	17	18	19	20	21	22							
9	23	24	25	26	27	28	29						
10	30	31	32	33	34	35	36	37					
11	38	39	40	41	42	**43**	44	45	46				
12	47	48	49	50	51	52	53	54	55	56			
13	57	58	59	60	61	62	63	64	65	66	67		
14	68	69	70	71	72	73	74	75	76	77	78	79	

For example, in the chart above, highlighted claim 43 is dependent from claim 7, and has the same content as claim 11. Ex. 1002 at ¶¶61-75.

IV. CONSTRUCTION OF THE CLAIMS

A claim in *inter partes* review is given the "broadest reasonable construction in light of the specification." *See* 37 C.F.R. § 42.100(b). As stated by the Federal Circuit in the case *In re ICON Health and Fitness, Inc.*:

> "[T]he PTO must give claims their broadest reasonable construction consistent with the specification. Therefore, we look to the specification to see if it provides a definition for claim terms, but otherwise apply a broad interpretation."

496 F.3d 1374, 1379 (Fed. Cir. 2007). In particular, claims in *inter partes* review should not be limited by party *argument* (whether in this or a prior proceeding). To the extent that the Patent Owner desires a claim term to be interpreted more narrowly than its broadest reasonable interpretation in light of the specification, the Patent Owner must show that the *specification* provides an express definition for the relevant portions of the claims, or amend the claims. *See SAP v. Versata*, CBM2012-00001, Pat. App. LEXIS 3788, *8 (PTAB June 11, 2013). As found by the *en banc* Federal Circuit:

> "If, in reexamination, an examiner determines that particular claims are invalid and need amendment to be allowable, one would expect an examiner to require amendment rather than accept argument alone."

Marine Polymer Tech., Inc. v. HemCon, Inc., 672 F.3d 1350, 1364 (Fed. Cir. 2012)(*en banc*).

For the purposes of this proceeding, claim terms are presumed to take on their broadest reasonable ordinary meaning. This meaning is explained in certain instances in the following subsections. The Petitioners note that the standard of claim construction used in district courts differs from the standard applied before the USPTO. Any claim constructions in this Petition are directed to the USPTO standard, and are not necessarily the constructions that the Petitioners believe would be adopted in court. The Petitioners do not acquiesce or admit to the constructions reflected herein for any purpose outside of this proceeding.

A. Claims 1, 9, 11, 23-29, and 38-46 — "input device"

In the '853 patent, the term "input device" includes a GUI element on screen, and is thus not limited to only hardware devices. Ex. 1002 at ¶78.

B. Claims 15 and 16 — "perform[ing] the steps recited in one of claims 1-14"

Claims 15 recites "15. A computer system configured to perform the steps recited in one of claims 1-14." Claim 16 recites "16. A storage medium storing a program for performing the steps recited in one of claims 1-14."

These claims are not multiple dependent claims, because do not further limit any of a group of superior claims. Rather, claims 15 and 16 are independent claims that incorporate as body elements only "one of" claims 1-14. The broadest reading for these claims results by choosing claim 1 as the "one of claims 1-14". Therefore, under the broadest reasonable interpretation of the claims, claims 15 and 16 recite system and storage medium claims respectively, each having the body elements of claim 1. Ex. 1002 at ¶79.

C. Claims 6, 12 and 47-56 — "first information includes an identification of a list of addressees"

Claims 6, 12 and 47-56 recite that the "first information includes an identification of a list of addressees". This phrase is ambiguous. It could mean that, in the document, there must be a name of a list. It could also mean that the first information identifies addressees in a list. Ex. 1002 at ¶80. The specification does not use the phrase "list of addressees" nor the phrase "identification of a list". Ex. 1002 at ¶81. The phrase "mailing list" is used (4:14-15 and 4:38-42), but it is unclear whether this is the name of a list or the list itself. Ex. 1002 at ¶81. Under the broadest reasonable interpretation, then, the phrase "[the] first information includes an identification of a list of addressees" should mean "[the] first information is sufficient to identify multiple addressees". Ex. 1002 at ¶83.

V. CLAIM-BY-CLAIM EXPLANATION OF GROUNDS FOR UNPATENTABILITY.

Ground 1. Claims 1-9, 11, 13-29, 38-45, 57-64, 66, 68-75, 77 and 79 are invalid as obvious under 35 U.S.C. § 103 over Goodhand.

Claims 1-9, 11, 13-29, 38-45, 57-64, 66, 68-75, 77 and 79 are invalid under 35 U.S.C. § 103 as obvious over U.S. Pat. No. 5,923,848 ("Goodhand") (Ex. 1003). Goodhand was filed on May 31, 1996 and issued on July 13, 1999, making it prior art under at least 35 U.S.C. § 102(e). Ex. 1002 at ¶85.

Goodhand teaches a "system and method for resolving email recipients' names." Ex. 1003 at Title. The background technology of Goodhand's system is an email application, for example Microsoft Outlook. Ex. 1003 at 8:37-43. Petitioner notes that in a co-pending litigation, the Patent Owner has asserted a related patent (U.S. Pat. No. 7,496,854), which is a continuation of the patent at issue here. Claim 1 of the '854 patent similarly requires information handling in a "document", which the Patent Owner reads on an "email document" in Gmail. This is shown in the attached Exhibit 1014, which is the relevant portion of the Patent Owner's infringement contentions, with highlighting added on page 1 by the Petitioners. Ex. 1014 at 1.

In the Goodhand email document, a user is allowed to enter some first text. The first text should be text related to a person to whom the user would like to send the email. Ex. 1002 at ¶90. When the user enters an execute command, the system analyzes the document, takes some of the text input, searches a database, and comes up with second text. Ex. 1002 at 91-104. The second text is a name or email address of the person to whom the email is being sent, and is used to correct or supplement the first text. The insertion of a proper recipient allows the email to be sent. This process is called "**address resolution**". Ex. 1003 at 16:48-52; 2:17-25; Ex. 1002 at ¶¶88-104.

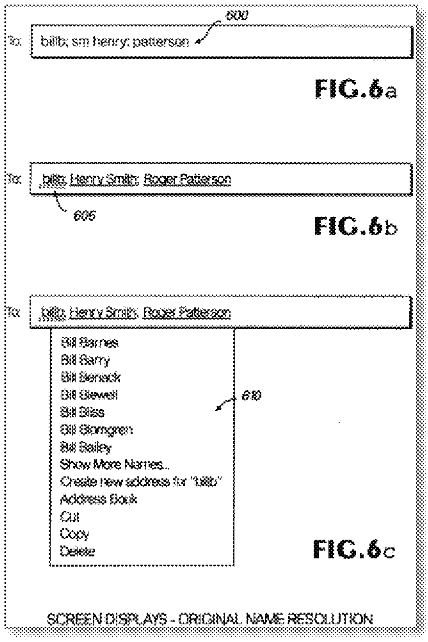


FIG.6a

FIG.6b

FIG.6c

SCREEN DISPLAYS - ORIGINAL NAME RESOLUTION

Figures 6a-6c of Goodhand (at right) show the process of **address resolution**. A user enters one or more names (here "billb", "sm henry" and "patterson") in the "To:" field. When the user exits the field (or alternatively, clicks a specific button), the system analyzes the user-entered text, breaking it up into smaller pieces. Ex. 1002 at ¶¶101, 122. It then recognizes "billb", "sm henry" and "patterson" as unresolved addresses, and searches for them in an external address book. Ex. 1003 at 17:22-30. The system tries to match each of "billb", "sm henry" and "patterson" with specific address book entries. Ex. 1003 at 17:34-37; Ex. 1002 at ¶¶88-114.

If the search for any text string results in one unambiguous hit in the address book, the full name of the person located by the search will be inserted in the "To:" line (here: "sm henry" is replaced with "Henry Smith" and "patterson" is replaced with "Roger Patterson" in Fig. 6b). Ex. 1003 at 17:37-40. If the search result is ambiguous (as was the case for "billb"), the user is given several options to resolve the ambiguity. Ex. 1003 at 17:53-62; Ex. 1002 at ¶¶88-114.

Thus, like the '853 patent, the Goodhand system responds to a user input command (e.g., the "check names" command), analyzes text in an email document (the user-entered text string) to find "first information" (the text string subdivided into tokens that can be used to search a database), uses the first information to search a separate address book, returns the results of the search in the form of an improved name or email address (second information), and inserts the improved name or email address in the document, thereby correcting, completing and displaying the improved name or email address.

Goodhand anticipates the claims challenged in this ground. The ground presents Goodhand under 35 U.S.C. § 103, however, for two reasons. First, claim

1 of the '853 patent requires a "record retrieval program", and requires "initiat[ing] a record retrieval from an information source using the record retrieval program". In Goohand, the information source is contained in one or more address books. These address books can be searched. In the case of Goodhand, the search is done by first analyzing text in an address line of an email to identify one or more search terms (called "display names"). These search terms are used to search the address book(s), in order to obtain better addressing information. Goodhand states:

> "As mentioned above, 'resolving' the names means **attempting to match the display names in the address field to specific user aliases that are included in a centralized address book or directory**, which is typically stored on a remote server, such as remote memory storage device 33 (FIG. 1)." Ex. 1003 at 17:29-41 (emphasis added); Ex. 1002 at ¶118.

There is no doubt from the above that a record retrieval is carried out, and that that the retrival is done by program code. Ex. 1002 at ¶¶96, 118. However, to the extent that the Patent Owner argues that Goodhand does not teach a separate "record retrieval program", it would have been obvious to provide one. Goodhand notes that its email system is conceptually divided into several components:

> "Like many personal information managers, the preferred application program is **divided into several modules**, including a calendar manager, a task list manager, **a contact manager**, **a message**

> **manager (e-mail)**, and a notes manager." Ex. 1003 at 8:45-49; Ex. 1002 at ¶118.

A person of ordinary skill would have understood the "contact manager" to have an address book function, and to be separate from the "message manager (email)". Ex. 1002 at ¶¶35, 118. Such a "contact manager" (like most databases) would have a data structure to contain information, and program code to access and modify the information. Ex. 1002 at ¶¶96, 118.

Goodhand further emphasizes that the different modules (including the contact manager) can be separated and distributed:

> "In a distributed computing environment, **program modules may be physically located in different local and remote memory storage** devices. **Execution of the program modules may occur** locally in a stand-alone manner or **remotely in a client/server manner**." Ex. 1003 at 8:58-62; Ex. 1002 at ¶118.

In fact, as shown in the quote above on page 17, the information that the system searches for is "included in a centralized address book or directory, **which is typically stored on a remote server**...." Ex. 1003 at 17:29-41 (emphasis added); Ex. 1002 at ¶118. The fact that the address book is on a remote server suggests that it is its own program.

Moreover, Goodhand states that access to address books is provided through a **M**essaging **A**pplication **P**rogramming **I**nterface (MAPI). The MAPI is part of the operating system, not part of the email program. Goodhand states:

> "The preferred operating system incorporates the Messaging Application Programming Interface (MAPI). The MAPI architecture is designed to make it easy for programmers to write messaging-enabled applications that are independent of the underlying messaging system. **MAPI provides high-level function that can be used to implement sophisticated messaging features** with a relatively small amount of code. The code deals only with functions for sending, receiving, and addressing messages. The underlying messaging system is completely transparent. **MAPI also provides other message-related functionality, such as access to address books**." Ex. 1003 at 12:37-49; Ex. 1002 at ¶118.

Goodhand further expressly states that searching of address books is done through MAPI functions:

> "If, at step 915 the computer determines that there are additional display names to resolve, **the computer** goes to step 930 and **attempts to resolve the remaining display names**. In the preferred e-mail program, **this is accomplished by calling the appropriate MAPI functions**, such as MAPIResolveName. Those skilled in the art will appreciate that t**his MAPI function handles the addressing chore of resolving informal names with actual e-mail aliases**." Ex. 1003 at 19:41-48 (emphasis added); Ex. 1002 at ¶118.

In this sense, the preferred embodiment of Goodhand facilitates communication between the email system and the address book at least through MAPI functions in the operating system, as opposed to the email program directly accessing the data structure that contains the contact information. Ex. 1002 at ¶118.

Thus, the inclusion of a separate record retrieval program would probably have been considered disclosed in the Goodhand system by a person of ordinary skill in the art, but is at a minimum trivially obvious over Goodhand's teachings. Ex. 1002 at ¶118. This is especially true given the level of skill in the art. Ex. 1002 at ¶¶19-60.

The second reason for presenting this ground under § 103 is that claim 1 of the '853 patent requires "analyzing the document to determine if the first information is contained therein". In Goodhand, the "first information" consists of partial names or email addresses (called "display names") that a user enters into an address line in an email document. Ex. 1003 at 17:15-20; Ex. 1002 at ¶122. This is shown in Fig. 6a, which is reproduced with highlighting added to show each item of "first information" (display name):

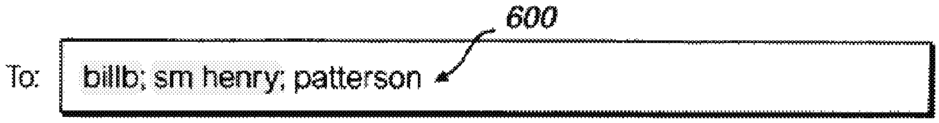


FIG.6a

Here the user has entered three "display names", which are partial (or simply incorrect) names or email addresses. Ex. 1003 at 17:15-20; Ex. 1002 at ¶¶89-90.

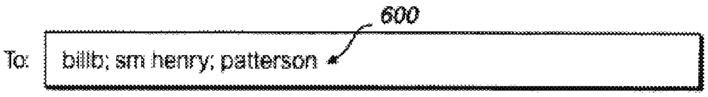
Goodhand's system takes each display name individually and checks it against a nickname list (Ex. 1003 at 19:24-40). If the name is not resolved, the system uses a MAPI function to perform an address book search using the current entered display name. Ex. 1003 at 19:42-54; Ex. 1002 at ¶118.

Goodhand's system must also analyze the text in the To: line to determine if there is anything there to process at all. Ex. 1002 at ¶122. Furthermore, Goodhand expressly discloses that the system is using each display name separately to perform searches. Therefore, the system must be correctly identifying each display name, or it would not be able to use the name as a search term. Ex. 1002 at ¶122.

Therefore, it is inherent in Goodhand's disclosure that the system "analyz[es] the document to determine if the first information is contained therein". Ex. 1002 at ¶122. At a minimum, however, this would have been trivially obvious to a person of ordinary skill in the art, because performing that analysis would allow the system to use the identified display names in the searches expressly taught by Goodhand. Ex. 1002 at ¶122.

Goodhand renders obvious claims 1-9, 11, 13-29, 38-45, 57-64, 66, 68-75, 77 and 79. An element-by-element mapping of these claims is provided in the following:

'853 Claims	Goodhand Disclosure
1. A computerized	Goodhand discloses a **computerized method for information handling within a document** created using an

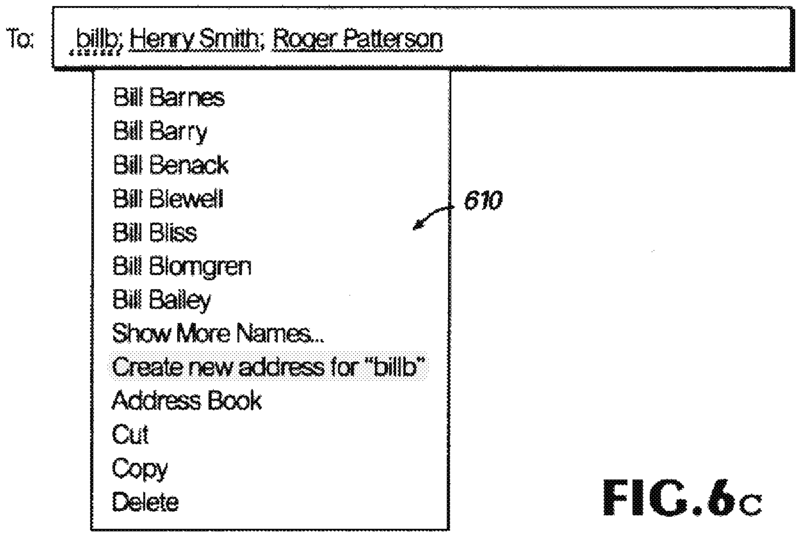
'853 Claims	Goodhand Disclosure
method for information handling within a document created using an application program, the document including first information provided therein, the method comprising:	application program. Ex. 1002 at ¶116. Specifically, Goodhand teaches "a method for resolving a display name associated with an intended recipient of a message item, which is sent in the context of an electronic mail system." Ex. 1003 at 5:1-4. The mail system is an **application program**. Ex. 1002 at ¶116. Goodhand further discloses receiving a text string representing an intended recipient (**first information**) in an address field of an email **document**. Ex. 1003 at 5:1-5; Ex. 1014 at 1. Figure 6a, reproduced below, illustrates a text strings in an address field (Ex. 1002 at ¶117):  FIG.6a
[1a] providing a record retrieval program;	Goodhand discloses an "address book" (aka "contact manager") that includes directory information such as recipients' names and addresses. Ex. 1003 at 13:40-46; Ex. 1002 at ¶118. The address book (contact manager) is a distinct software module, which can be a separate program, and can even be located across a network. Ex. 1003 at 15:40-49; 11:20-22; 8:50-65; Ex. 1002 at ¶118. Goodhand further discloses MAPI functions that provide access to the address book (Ex. 1003 at 12:36-49) and perform search functions (Ex. 1003 at 19:41-54). The MAPI functions and address book together make up the **record retrieval program**. Ex. 1002 at ¶118. Goodhand further discloses **retrieving records** using the address book based on the first information. Goodhand discloses that: "As mentioned above, 'resolving' the names means **attempting to match the display names in the address field to specific user aliases that are included in a centralized address book or directory**, **which is typically stored on a remote server**, such as remote memory storage device 33

'853 Claims	Goodhand Disclosure
	(FIG. 1). In the preferred application program, the e-mail system searches several address book fields in an effort to match the display names with the first name, last name, and/or alias of a registered user. Thus, in this example, the e-mail program will attempt to match 'billb,' "sm henry,' and 'patterson' with specific address book entries belonging to registered users. FIG. 6b illustrates the results of the effort to resolve the names. **If a display name is unambiguous and matches only one registered user, the name of that user is inserted in the address field**." Ex. 1003 at 17:29-41 (emphasis added); Ex. 1002 at ¶118.
[1b] providing an input device configured to enter an execute command which initiates a record retrieval from an information source using the record retrieval program;	Goodhand discloses providing an **input device to initiate record retrieval from an information source**. As explained above, the address resolution process initiates the record retrieval. There are three alternative execute commands that could be used to initiate address resolution. Ex. 1002 at ¶119. First, Goodhand states that using a mouse or keyboard (**input devices**)(Ex. 1003 at 10:45-49) to move a cursor to another field (**execute command**) triggers email address resolution: "[a]s soon as the user moves the cursor to another field on the e-mail form, the email program module begins to resolve the recipient names in the background,…. As mentioned above, 'resolving' the names means attempting to match the display names in the address field to specific user aliases that are included in a centralized address book or directory." Ex. 1003 at 17:21-29; Ex. 1002 at ¶119. Next, Goodhand also discloses that the **execute command** could be a "send mail" or "check names" command. As Goodhand states: "Those skilled in the art will appreciate that in the preferred application program, addresses are also resolved when the user sends the message or if the

'853 Claims	Goodhand Disclosure
	user selects the 'check names' command." Ex. 1003 at 20:18-21; 16:54-56; Ex. 1002 at ¶119-120.
[1c] upon a single entry of the execute command by means of the input device:	*See* immediately above. Goodhand discloses a **single entry of the execute command**. Depending on the particular embodiment, the single execute command is any of (1) moving the cursor to another field, (2) sending the email, or (3) clicking a 'check names' button. Ex. 1003 at 17:21-29, 20:18-21; 16:54-56; Ex. 1002 at ¶121.
[1d] analyzing the document to determine if the first information is contained therein, and	Goodhand discloses that after the user enters the execute command, the computer **analyzes the document** to find display names or addresses **(first information)** (Ex. 1002 at ¶122), and determine whether they need to be resolved. This happens in two ways. First, the system **determines** (as in Fig. 6a) **whether there is one or more than one name** in the To: line. Ex. 1002 at ¶122. Second, Goodhand discloses that display names (first information) are **analyzed** and identified, because the system later uses the display names as search terms. In order to identify the names, the system must **determine that they are there**. Ex. 1002 at ¶122. In other words, the system has analyzed the user-entered text string to find smaller strings that can be used as a search term in a database search. Ex. 1002 at ¶¶101, 122. Goodhand explains that the system uses the names in later steps, for example, to test against the nickname list: "At step 910 the computer first checks to see if the display names in the address field corresponds to a nickname that is stored in the nicknames memory cache, which is stored in the computer's memory storage devices as part of the user's profile. At step 915 the computer determines whether any display names remain to be resolved. If all of the display names were resolved by matching nicknames, the computer goes to step 920 and displays the address data with the proper indicia. As discussed above, in the case of nicknames, the full name of the recipient is inserted in the address field and is preferably marked with a blue dashed line beneath it. From step

'853 Claims	Goodhand Disclosure
	920, the computer proceeds to step 925 and the method 900 terminates." Ex. 1003 at 19:25-44; Ex. 1002 at ¶122. The nickname search of Goodhand is also similar to one of the '853 patent's methods of analysis, namely the use of a name first or last name database. Ex. 1001 at 4:36-37. Goodhand further notes that text entries not identified as nicknames are used for address book searching, as discussed below. This shows that the system has determined that the text entries are in the document. Ex. 1002 at ¶¶101-102.
[1e] if the first information is contained in the document, searching, using the record retrieval program, the information source for second information associated with the first information; and	Goodhand discloses **searching the information source** (address book database) **for second information associated with the first information**. For example, Goodhand discloses that: "[i]n the preferred application program, the e-mail system **<u>searches several address book fields in an effort to match the display names with the first name, last name, and/or alias of a registered user</u>**. Thus, in this example, the e-mail program will attempt to match 'billb,' 'sm henry,' and 'patterson' with specific address book entries belonging to registered users" Ex. 1003 at 17:30-36 (emphasis added); see also 29:41-55; Ex. 1002 at ¶¶123-125. This searching is in addition to nickname searching, which may not resolve entered text. Ex. 1002 at ¶¶101-102. As explained before the claim chart, the searching is performed using the MAPI functions and the address books (**using the record retrieval program**). Ex. 1003 at 19:41-54; Ex. 1002 at ¶102; see also Ex. 1003 at 17:30-36. The "**second information**" is the information found in the address book, such as a correct name or email address. Ex. 1002 at ¶123-124. Goodhand also discloses, in the flow chart of Figure 9, that second information is in fact located and used: "[i]f all of the display names were resolved by matching nicknames, the computer…displays the

'853 Claims	Goodhand Disclosure
	address data with the proper indicia." Ex. 1003 at 19:33-35. For example, Goodhand states: "FIG. 6b indicates that the display names 'sm henry' and 'patterson' were unambiguously matched to 'Henry Smith'" and 'Roger Patterson,' respectively." Ex. 1003 at 19:43-45; Ex. 1002 at ¶126.
[1f] when the information source includes second information associated with the first information, performing at least one of, (a) displaying the second information, (b) inserting the second information in the document, and (c) completing the first information in the document based on the second information.	As discussed in element [1e] above, Goodhand discloses using display names (**first information**) to locate information such as first name, last name, aliases, and address data (**second information**). Goodhand also discloses **displaying the second information, inserting the second information, and completing the first information**. Regarding the "display" of information, Goodhand discloses that: "[i]f all of the display names were resolved by matching nicknames, the computer…**displays** the address data with the proper indicia." Ex. 1003 at 19:33-35 (emphasis added); Ex. 1002 at ¶125. This also has the effect of **inserting** the second information in the document **completing** the first information (unresolved names). Ex. 1002 at ¶¶127-131. Furthermore, in the event that a display name is ambiguous Goodhand states that: "[i]n the preferred system, the user places the cursor over the unresolved display name and clicks the right mouse button. In response, the e-mail program displays a context menu 610 that includes a list of possible matches." Ex. 1003 at 17:53-58; Ex. 1002 at ¶127. Goodhand further teaches **completing the first information based on the second information**. For example, Goodhand discloses that "[i]f a display name is unambiguous and matches only one registered user, the name of that user is inserted in the address field." Ex. 1003 at 17:39-40; Ex. 1002 at ¶128. This both completes and corrects the field. As shown in Figs. 6a-6c, for example,

'853 Claims	Goodhand Disclosure
	the name "patterson" is both "completed" and "corrected" by "inserting" the full name "Roger Patterson" into the field. Ex. 1002 at ¶¶128-130.
2. The method of claim 1, further comprising one of the following steps: storing the first information in the information source if no second information associated with the first information is found in the information source during said searching step, changing the second information in the information source based on one of differences and similarities between the first information and the second information, completing the first information in the document using the second information, adding one of all and part of the	Goodhand discloses each of the "correcting", "completing", "storing", and "adding one of all and part of the first information" steps. Regarding the "**correcting**" and "**completing**" steps, see claim 1, limitation [1f], above. Goodhand teaches correcting a name that was initially typed into an address field with a name found in the address book. This constitutes both "correcting" and "completing" the name. Ex. 1002 at ¶¶132-133, 127-131. For example, "sm henry" was corrected and completed to "Henry Smith" and "patterson" was completed / completed to "Roger Patterson". Ex. 1002 at ¶129. Regarding the "**storing**" step, Goodhand discloses that if an entry is not recognized as belonging to a specific entry in the address book (**no second information is found**) Ex. 1003 at 17:53-62), the user has the option to add it to the address book (**storing the first information in the information source**). This is shown in Fig. 6c for example (where the relevant option has been highlighted by the Petitioner):  To: billb; Henry Smith; Roger Patterson Bill Barnes Bill Barry Bill Benack Bill Blewell Bill Bliss Bill Blomgren Bill Bailey Show More Names... Create new address for "billb" Address Book Cut Copy Delete 610 FIG.6c

first information to an existing record in the information source associated with one of all and part of the first information, correcting the first information in the document using the second information, adding information about said document to said information source, and adding information about said document to said information source, said added information associated with said second information.	Goodhand explains the "create new address" option as follows: "The 'create new address for billb' option in the context menu 610 allows the user to create an entry in his or her personal address book. Those skilled in the art will appreciate that this is typically used to store addresses of e-mail recipients who are not registered users on the local e-mail system. For example, if 'billb' is a friend that the user communicates with via Internet e-mail, the **user can record Bill's Internet e-mail address in his or her personal address book**." Ex. 1003 at 18:2-9 (emphasis added); Ex. 1002 at ¶134. Regarding the step of "**adding one of all and part of the first information to an existing record**", Goodhand discloses that the user can use a nickname for a person, such as "billb" for "Bill Bailey". When a nickname is used for the first time, it is added to a nickname list. Goodhand explains: "In addition to the features described in conjunction with 15 FIGS. 6a-c, the preferred e-mail program module automatically creates a list of nicknames that are based on how the user resolves ambiguous display names. This allows a user to use convenient, but ambiguous, display names to identify intended recipients." Ex. 1003 at 18:14-19; Ex. 1002 at ¶137-139. As noted above, the nickname (**first information**) is **associated with** a particular person in the address book, and is thus associated with **existing record in the information source**". The nickname is also stored in the address book. Goodhand summarizes the nickname features as follows: "As described above, the nickname list is stored in a memory cache that is part of the user's profile, and is used to automatically create nicknames. **The nickname is stored with the actual e-mail alias or address book entry of the intended recipient**. If this is the first time a nickname is resolved, it is

	added to the nickname cache. If the nickname was earlier matched to a different alias, the nickname list is updated to reflect the current recipient. In the preferred system, the nickname file is part of the user's profile, which is stored on one or more of the computer's memory storage devices." Ex. 1003 at 20:1-11 (emphasis added); Ex. 1002 at ¶138.

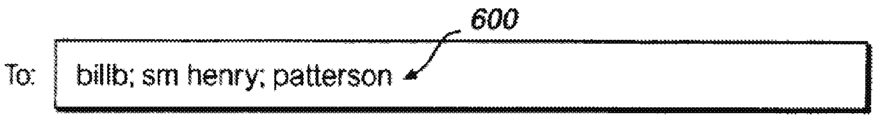
3. The method of claim 1, wherein said second information includes at least one of a zip code, a city, a state, a county, a country, a street name, a house number, an apartment number, a telephone number, an email address and abbreviations or misspellings thereof, further comprising: performing at least one of completing and correcting at least one of a zip code, a city, a state, a county, a country, a street name, a house number, an apartment number, a telephone	Goodhand teaches that the second information can be an **email address or an abbreviation or misspelling of an email address**. Ex. 1002 at ¶¶140-146. Specifically, Goodhand teaches that a user can enter "display names" into an address field. "Display names" can be personal names or email addresses. Goodhand states: "The entered **display name** may include all **or part of** the intended recipient's **first name**, **last name**, and/**or e-mail alias**." Ex. 1003 at 17:17-21)(emphasis added); Ex. 1002 at ¶141. Goodhand notes that "[t]he **e-mail addresses** of all registered users are **referred to as aliases**…." Ex. 1003 at 16:45-48 (emphasis added); *see also* 16:40-53; Ex. 1002 at ¶141. The Goodhand system will then correct or complete the entered display name with the correct display name. The correct display name can be a name (which is an abbreviation for or misspelling of an email address) or an email address (alias). Goodhand states: "As mentioned above, 'resolving' the names means attempting to match the display names in the address field to specific user aliases that are included in a centralized address book or directory, which is typically stored on a remote server, such as remote memory storage device 33 (FIG. 1). In the preferred application program, the e-mail system searches several address book fields in an effort to match the display names with the first name, last name, and/or alias of a registered user." Ex. 1003 at 17:24-34; Ex. 1002 at ¶141.

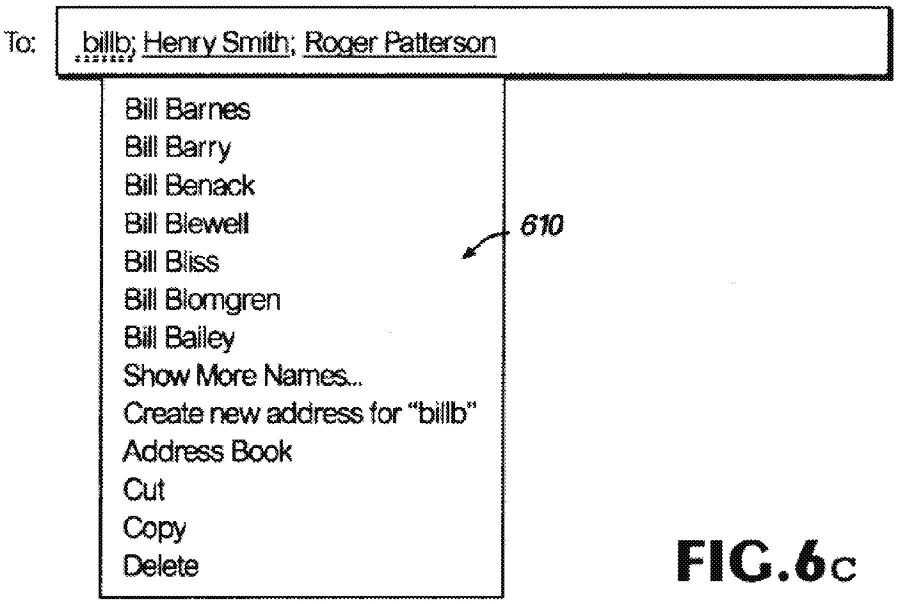
number, an email address and abbreviations or misspellings thereof in the first information based on the second information.	As explained above under claim 1 (limitation [1f]) and claim 2, the second information that is located is used to **complete / correct the first information** entered by the user. Ex. 1003 at 17:37-40; Figs. 6a-6c; Ex. 1002 at ¶132-133, 127-131. Note again that a "display name" includes an **email address**, as discussed above, and includes names that serve as abbreviations of email addresses. Ex. 1002 at ¶141, 143.

4. The method of claim 1, where in said second information includes at least one of a zip code, a city, a state, a county, a country, a street name, a house number, an apartment number, a telephone number, an email address and abbreviations or misspellings thereof, further comprising: performing at least one of completing and correcting at least one of a zip code, a city, a state, a county, a country, a street name, a house number, an apartment number, a telephone number, an email address and abbreviations or misspellings thereof in the first information based on the second information **automatically**.	*See* claim 3, above. The process disclosed by Goodhand and described under claim 3, above, is performed **automatically** when the single execute command (see claim 1, limitations [1b] and [1c]) is activated by the user, as long as the search for "second information" (**email address or abbreviation thereof**) results in an unambiguous match. Ex. 1003 at 17:37-40; Ex. 1002 at ¶148-152.

5. The method of claim 1, where in said second information includes at least one of a zip code, a city, a state, a county, a country, a street name, a house number,	*See* claims 3 and 4, above. If the process disclosed by Goodhand and described under claim 3, above, results in an ambiguous match for second information (**email address or abbreviation thereof**), then Goodhand describes

an apartment number, a telephone number, an email address and abbreviations or misspellings thereof, further comprising: performing at least one of completing and correcting at least one of a zip code, a city, a state, a county, a country, a street name, a house number, an apartment number, a telephone number, an email address and abbreviations or misspellings thereof in the first information based on the second information **with assistance** from a user.	a process where the ambiguity is resolved **with assistance from the user**. Goodhand states: "If the display name is ambiguous, the e-mail program indicates that the display name needs to be **manually resolved** by displaying the display name and a predetermined indicia, such as a squiggly line 605 beneath the display name….FIG. 6c illustrates the process by which **a user manually resolves ambiguous names**. In the preferred system, the user places the cursor over the unresolved display name and clicks the right mouse button. In response, the e-mail program displays a context menu 610 that includes a list of possible matches. In this case, the possible matches include users whose first name is Bill and whose last name begins with the letter "B.' If the intended recipient's name is displayed in the context menu 610, **the user may select the correct name from the list**." Ex. 1003 at 17:41-62 (emphasis added); Ex. 1002 at ¶155-156.

6. The method of claim 1, wherein said first information includes an identification of a list of addressees, further comprising: addressing said document to all of said addressees based on the	Goodhand discloses emails that **identify a list of addressees**. Ex. 1002 at ¶157-159. For example Fig. 6a shows such an email:  FIG.6a It is clear from Goodhand that **the entered display names can be email addresses** instead of the partial/incorrect names shown in Fig. 6a. Ex. 1003 at 16:45-48 and 7:17-21; *see also* 16:40-53; Ex. 1002 at ¶141.

second information associated with said identification of said list of addressees.	Goodhand teaches properly **addressing the email to all of the addressees** by correcting their email addresses through its resolution process using correct display names/addresses **(second information)**, as discussed above under claim 1, and shown in Fig. 6c (Ex. 1002 at ¶160-161):  To: billb; Henry Smith; Roger Patterson Bill Barnes Bill Barry Bill Benack Bill Biewell Bill Bliss Bill Blomgren Bill Bailey Show More Names... Create new address for "billb" Address Book Cut Copy Delete 610 FIG.6c

7. The method of claim 1, further comprising: providing a user the option of making changes to the second information directly in the information source.	Goodhand discloses that an "address book" allows direct modification. Goodhand notes that in its preferred application, Microsoft Outlook, "allows users to manage their own calendar, messages, tasks, notes, and contacts…." Ex. 1003 at 8:43-44; Ex. 1002 at ¶163. Furthermore, Goodhand teaches a MAPI personal address book that allows user editing. Goodhand states: "The users of client applications can view the contents of address book containers and in some cases modify it. MAPI's Personal Address Book is an example of a modifiable address book container that allows new entries to be added and exiting entries to be modified or deleted." Ex. 1003 at 13:48-54; Ex. 1002 at ¶163.

	Goodhand further discloses that in cases where display names cannot be resolved without ambiguity, the user is given the option to add to the address book. Ex. 1003 at 18:2-9; Ex. 1002 at ¶164-165. *See* claim 2, above.

8. The method of claim 1, wherein: the step of using said application program comprises using said application program to enter first information comprising one of a person's name, a person's title, a person's name and address, a business name, a business name and address, a telephone number, and an email address, or a part thereof, into said document; and	Goodhand discloses entering display names (first information) which "may include all or part of the intended recipient's first name, last name, and/or e-mail alias." Ex. 1003 at 17:15-20; Ex. 1002 at ¶167-168. The email alias is an email address. Ex. 1003 at 16:40-53; Ex. 1002 at ¶141.
[8a] the step of searching comprises searching, using the record retrieval program, the information source for second information comprising one of a person's name, a person's title, a person's name and address, a business name, a business name and address, a telephone number, and an email address, associated with the first information.	As discussed above under claims 1 and 3, Goodhand discloses that its system searches an address book for a corrected display name, which can be a person's name or email address. Ex. 1003 at 17:24-34; Ex. 1002 at ¶169; Ex. 1003 at 16:40-53; Ex. 1002 at ¶141.

9. The method of claim 1, wherein:	As discussed above under claim 1, element [1b], Goodhand discloses a "check names" or

the step of providing an input device comprises providing an input device comprising one of a touch screen, a keyboard button, an icon, a menu and a voice command device, and configured to enter an execute command which initiates a record retrieval from an information source using the record retrieval program; and	"send mail" command. Ex. 1003 at 20:18-21; 16:54-56; Ex. 1002 at ¶173. In an email message, commands are verbs (e.g. "check names") that appear as **menu items**. Ex. 1003 at 15:8-11; Ex. 1002 at ¶175; see also 172-177. Goodhand further discloses that the user enters commands using a **keyboard** or mouse. Ex. 1003 at 10:45-48; Ex. 1002 at ¶174.
[9a] the step of displaying the second information comprises displaying the second information comprising one of displaying a message screen with the second information and providing a voiced response of the second information.	Goodhand discloses **displaying the second information by displaying a message screen**. For example, in the event that a display name is ambiguous, Goodhand teaches that: "[i]n the preferred system, the user places the cursor over the unresolved display name and clicks the right mouse button. In response, the e-mail program displays a context menu 610 that includes a list of possible matches." Ex. 1003 at 17:53-58; Ex. 1002 at ¶178-180. Goodhand also discloses that: "if the user attempts to send the message without resolving the ambiguous display names displayed by the email program, the process defaults back to the normal process for resolving names, which displays a dialog box from which the user must choose the correct name." Ex. 1003 at 20:27-31; Ex. 1002 at ¶¶178-180.

11. The method of claim 1, wherein the step of providing an input device comprises:	As discussed above under claim 1, limitation [1b], Goodhand discloses an input device configured to enter an execute command which

Providing an input device configured to enter an execute command which initiates a record retrieval from an information source comprising at least one of a file, a database, a database program, a computer network, and a contact management program, using the record retrieval program.	initiates a record retrieval from an information source. Goodhand also discloses, with reference to Figure 1, that the information source may be "a centralized address book or directory, which is typically stored on a remote **server**, such as remote **memory** storage device 33." Ex. 1003 at 17:28-30 (emphasis added); 2:28-32; Ex. 1002 at ¶181-186. This information source represents at least a "**file**" and a "**network**". The address book encompasses a "**contact manager**", which stores its information in a **database**. Ex. 1003 at 15:40-48; Ex. 1002 at ¶181-186. Moreover, Goodhand notes that its processes can use **networked file systems**. Ex. 1003 at 9:4-10; Ex. 1002 at ¶185.

13. The method of claim 1, further comprising the step of indicating which part of information in said document is said first information.	Goodhand discloses indicating the part of information in a document is a display name (first information). For example, Goodhand discloses that: "[i]f all of the display names were resolved by matching nicknames, the computer…**displays** the address data **with the proper indicia**." Ex. 1003 at 19:33-35 (emphasis added); Ex. 1002 at ¶125. Goodhand also discloses that: "[i]f the display name is ambiguous, the email program indicates that the display name needs to be manually resolved by **displaying the display name and a predetermined indicia**, such as a squiggly line 605 beneath the display name." Ex. 1003 at 17:41-45 (emphasis added); Ex. 1002 at ¶187-189.

14. The method of claim 1, further comprising the step of automatically interpreting which part of information in said document is said first information.	As discussed in claim element [1e] Goodhand discloses automatically detecting and resolving display names, which are preferably separated by delimiters such as a semicolon, when the user moves the cursor from the address field or activates a "check names" or "send mail" command. Ex. 1003 at 17:15-23; Ex. 1002 at ¶190-192.

Regarding claims 15 and 16, Goodhand discloses a "computer system" Ex. 1003 at 10:10-12) and a "storage medium storing a program". Ex. 1003 at 11:20-31. In order to meet the language of claims 15 and 16, the computer system and stored program need only carry out one of claims 1-14. Goodhand teaches that its computer system and stored program carry out claim 1, as shown above. Ex. 1002 at ¶193.

Claims 17-29, 38-45, 57-64, 66, 68-75, 77 and 79 have the same limitations as claims 8, 9, 11 or 14. The additional dependencies do not affect disclosure of the methods having these limitations, as arranged in the respective claims. Ex. 1002 at ¶¶194-197.

Claims 17-22 have the same limitation as claim 8. Goodhand teaches this limitation as shown above for claim 8. Ex. 1002 at ¶¶61-75, 194-197. **Claim 21** makes the content of claim 8 dependent from claim 6. In claim 6, the first information "includes an identification of a list of addressees". In claim 8, there must be first information of a particular type "comprising one of a person's name, a person's title, a person's name and address, a business name, a business name and address, a telephone number, and an email address, or a part thereof." It is noted that neither claim 6's requirement nor claim 8's requirement limits the overall system to a specific type of first information. It is within the scope of these claims, for example, to have information of the recited type alongside information that is not recited. Moreover, as explained in the claim construction section, claim 6's limitation is properly construed to include information sufficient to identify more than one address, and thus encompasses the specific information recited in claim 8. Ex. 1002 at ¶68.

Claims 23-29 have the same limitation as claim 9. Goodhand teaches this limitation as shown above for claim 9. Ex. 1002 at ¶61-75, 194-197.

Claims 38-45 have the same limitation as claim 11. Goodhand teaches this limitation as shown above for claim 11. Ex. 1002 at ¶61-75, 194-197.

Claims 68-75, 77 and 79 have the same limitation as claim 14. Goodhand teaches this limitation as shown above for claim 14. Ex. 1002 at ¶61-75, 194-197.

Ground 2. Claims 6, 10, 12, 21, 27, 30-37, 42, 46-56, 61, 65, 67, 72, 76 and 78 are invalid under 35 U.S.C. § 103 as obvious in view of Goodhand and Padwick.

Claims 6, 10, 12, 21, 27, 30-37, 42, 46-56, 61, 65, 67, 72, 76 and 78 are invalid under 35 U.S.C. § 103 as obvious in view of Goodhand (as applied above in Ground 1) and Padwick, et al., "Using Microsoft Outlook 97" (Microsoft Press)(Ex. 1004).

Padwick was deposited in the Library of Congress in 1996 Ex. 1002 at ¶199, thus making it prior art under 35 U.S.C. § 102(b). Padwick also bears a copyright date of 1997, thus making it prior art under 35 U.S.C. § 102(a) or (b).

Goodhand is applied as in Ground 1, above. Padwick is a general reference book for the Microsoft Outlook program. Padwick teaches various aspects of the Microsoft Outlook system recited in the above-referenced claims.

REASONS TO COMBINE GOODHAND AND PADWICK

There was ample motivation to combine Goodhand's address resolution system for email with Padwick's general disclosure of a well-known email application. First, Goodhand is a patent directed to an electronic mail system. Ex. 1003 at Abstract; Ex. 1002 at ¶¶213-215. Goodhand teaches that there was a design need, specifically:

> "a need for an e-mail system that simplifies the process of resolving recipient addresses and minimizes the input required from the user." Ex. 1003 at 4:29-31; Ex. 1002 at ¶213-215.

Goodhand further teaches that it meets the design need:

> "The present invention satisfies the above described needs by providing an **improved system and method for composing, processing, and organizing electronic mail message items**. The present invention automatically resolves recipient display names while the user is composing the message. The invention provides multiple options for resolving ambiguous names and automatically

> creates nicknames based on how ambiguous names are resolved." Ex. 1003 at 4:45-53 (emphasis added); Ex. 1002 at ¶215.

Goodhand expressly recommends the use of the Microsoft Outlook program. Ex. 1003 at 8:37-45. Padwick, entitled "Using Microsoft Outlook^{TM} 97", is directed to exactly the recommended platform of Goodhand. Padwick expressly teaches a form of automatic address resolution (Ex. 1004 at pp. 793-794, bridging paragraph; Ex. 1002 at ¶¶214-216), for which Goodhand is an improvement. Goodhand is also assigned to Microsoft Corporation, the maker of Microsoft Outlook. *See Ex parte Mettke*, Appeal 2008-0610, 2008 Pat. App. LEXIS 6761, *43-*44 (BPAI Sept. 30, 2008) ("Exhibits C, D, E, and F are all from the same corporation and all relate to versions of the same pay-for-use terminal. One skilled in the art would have been motivated to combine the teachings in one reference with teaching in another reference because they are all related to the same terminal apparatus, i.e., there is not the usual obviousness problem of explaining why one skilled in the art would have sought to combine two references from unrelated sources."). Ex. 1002 at ¶¶213-220.

Furthermore, Goodhand and Padwick represent known elements that could have been combined for their known functions, with no unpredictable results, for example to satisfy the design need (explained above) noted in Goodhand. Ex. 1002 at ¶220. The combination is therefore obvious under *KSR Int'l Co. v.*

Teleflex, Inc., 127 S.Ct. 1727, 1739-40 (2007).

The combination renders claims 6, 10, 12, 21, 27, 30-37, 42, 46-56, 61, 65, 67, 72, 76 and 78 obvious, as discussed in the following. Ex. 1002 at ¶¶243-244.

Claim 10 requires "using one of a word processing program and a spreadsheet program to enter first information into a respective one of a word processing document and a spreadsheet document." Goodhand discloses using an email editor (preferably Microsoft Outlook) to enter email messages. Ex. 1002 at ¶233. Goodhand does not expressly state that the email editor is a word processor.

Padwick, however, teaches that "Microsoft Outlook can be configured to use Microsoft Word as its e-mail editor." Ex. 1004 at p. 518 above Fig. 22.7; Ex. 1002 at ¶234. As taught by the '853 patent itself, Microsoft Word was a word processor. Ex. 1001, 1:29, Figs. 3-5 at top; Ex. 1002 at ¶233. Using Microsoft Word to edit email in Outlook is called the "WordMail" function. Padwick states that "[o]ne reason for choosing WordMail over Outlook's regular e-mail message tool is that WordMail offers you more tools and options from which to choose when creating messages". Padwick provides a list of such additional features on page 519. Ex. 1002 at ¶235.

Claims 30-33 and 35-37 have the same content as claim 10, and depend from claims rendered obvious by Goodhand as shown in Ground 1. **Claims 46, 65 and 76** have the content of claims 11, 13 and 14, respectively, which are rendered obvious by Goodhand as shown in Ground 1. Claims 46, 65 and 76 add the same limitation as claim 10, and are therefore obvious for the same reasons. The additional dependencies do not affect obviousness of the methods having these limitations. Ex. 1002 at ¶¶231-237, 61-75.

Claims 6 and 12 are likewise obvious over Goodhand in view of Padwick. Note that this ground applies to claim 6, even if it is construed narrowly. Claim 6 recites "wherein said first information includes an identification of a list of addressees, further comprising: addressing said document to all of said addressees based on the second information associated with said identification of said list of addressees." Claim 12 recites "said first information includes an identification of a list of addressees, further comprising: creating copies of said document, each addressed to one of addressees in said list identified by said first information, based on said second information associated with said identification of said list of addressees."

Claim 6 would have been obvious over Goodhand in view of Padwick. Padwick teaches that users can establish distribution lists to send mail to a number of people at once. Padwick states "[y]ou may want to create, for example, a distribution list with the names of everyone in your department or one for preferred vendors." Ex. 1004 at p. 339; Ex. 1002 at ¶224. On page 339, Padwick shows in Fig. 13.14 that distribution lists have nicknames like "Admin". Ex. 1004 at p. 339, Fig. 13.14 and caption; Ex. 1002 at ¶225.

It would have been obvious to practice the method disclosed in Goodhand to replace a distribution list nickname with the individual addresses associated with the distribution list. This represents no more than recognizing that, associated with the first information (distribution list nickname), there is more than one piece of second information (addresses associated with the distribution list), and providing each of those pieces of second information. *See* MPEP § 2144.04.VI.B ("mere duplication of parts has no patentable significance unless a new and unexpected result is produced."). Furthermore, this modification would have provided the common-sense advantage of the user being able to confirm the individual addressees in the distribution list using fewer mouse clicks, while having no unpredictable result. Ex. 1002 at ¶228. This desire to decrease mouse clicks is expressly taught by Goodhand. Ex. 1003 at 4:29-31; Ex. 1002 at ¶228. Furthermore, claim 6 represents a mere automation of what the user could do manually, by viewing the distribution list in Microsoft Outlook and copying email addresses manually. Ex. 1004 at p. 339, Fig. 13.14; Ex. 1002 at ¶229. *See* MPEP § 2144.04.III. ("[P]roviding an automatic or mechanical means to replace a manual activity which accomplished the same result is not sufficient to distinguish over the prior art.").

Claim 12 is obvious for the same reasons, because emails sent to distribution lists would obviously have a separate copy arrive at each recipient's inbox, and would need to be addressed to that recipient in order to arrive correctly. Ex. 1002 at ¶242; see also Ex. 1002 at ¶241.

Claims 21, 27, 42, 61 and 72 are dependent from claim 6, but recite only the limitations of claims 8, 9, 11, 13 and 14, which are disclosed by Goodhand as in Ground 1. The discussion regarding claim 21, found above on page 36, is relevant here. Ex. 1002 at ¶68. **Claim 34** is dependent from claim 6, but recites only the limitation of claim 10 rendered obvious by Goodhand in view of Padwick. Ex. 1002 at ¶¶221-222, 61-75.

Claims 47-56 recite only the limitation of claim 12, which is obvious. Among these claims, **claims 47-50** are dependent on claims 2-5, respectively, which are rendered obvious by Goodhand as in Ground 1. **Claim 51** is dependent on claim 6, which is obvious over Goodhand and Padwick, even if claim 6 is construed narrowly. **Claims 52-54 and 56** are dependent on claims 7-9 and 11, respectively, which are rendered obvious by Goodhand as in Ground 1. **Claim 55** is dependent on claim 10, which is obvious over Goodhand and Padwick. Thus, claims 47-56 are obvious over Goodhand and Padwick. The additional dependencies do not affect obviousness of the methods having these limitations. Ex. 1002 at ¶¶238-239, 61-75.

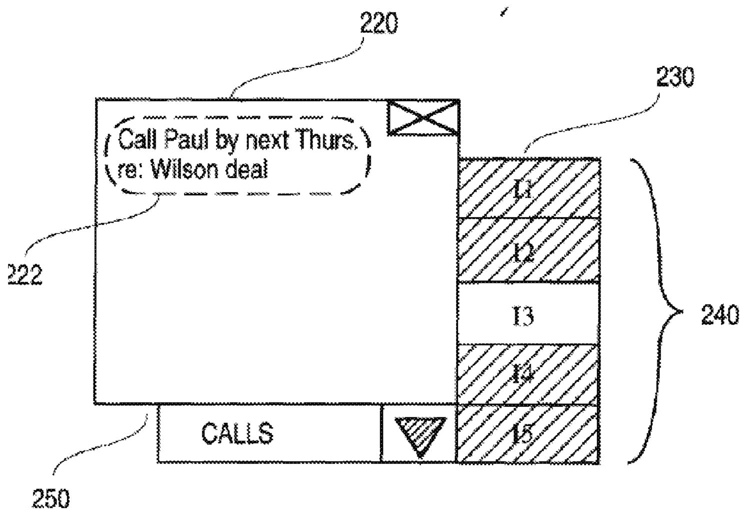
Claims 67 and 78 are dependent from claim 12, but recite only the limitations of claims 13 or 14, which are rendered obvious by Goodhand as in Ground 1.

Petitioners note that the claim 53 makes the limitation of claim 12 dependent on claim 8. This is very similar to the situation of claim 21, where the limitation of claim 6 (having similar language) is dependent on claim 8. Thus, the discussion above on page 36 regarding claim 21 is applicable here. Ex. 1002 at ¶68.

Ground 3. Claims 1, 2, 7-11, 13-17, 22-23, 28-30, 35-38, 43-46, 57, 62-66, 68, 73-77, and 79 are invalid under 35 U.S.C. § 102(e) over Allen.

Claims 1, 2, 7-11, 13-17, 22-23, 28-30, 35-38, 43-46, 57, 62-66, 68, 73-77, and 79 are invalid under 35 U.S.C. § 102(e) over U.S. Pat. No. 6,026,410 ("Allen") (Ex. 1005). Allen was filed on Feb. 10, 1997 and is thus prior art under 35 U.S.C. § 102(e). Ex. 1002 at ¶¶245-246.

Allen teaches a computer system with multiple applications, where the applications can be used with notes made by the user. The user creates notes using a text editor, as shown in Fig. 6, at right. After the entry of a keystroke in the text editor, the Allen system analyzes the text for recognizable keywords. If keywords are found, they are used to search a database for information related to the keywords, which is then displayed to the



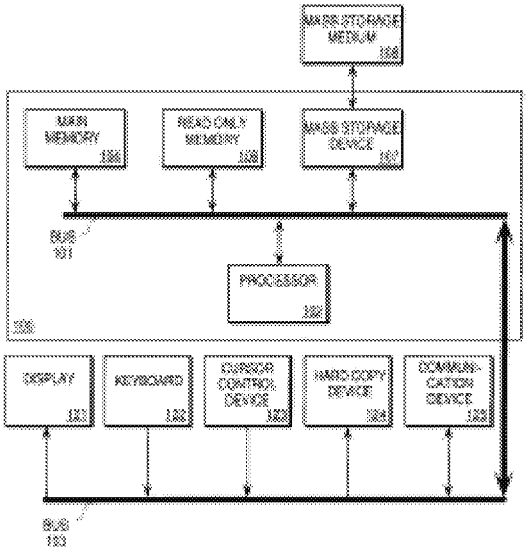
user. Ex. 1005 at 2:59-3:14; claims 1 and 2; Ex. 1002 at ¶246-249; *see also* Ex. 1002 at ¶¶246-272.

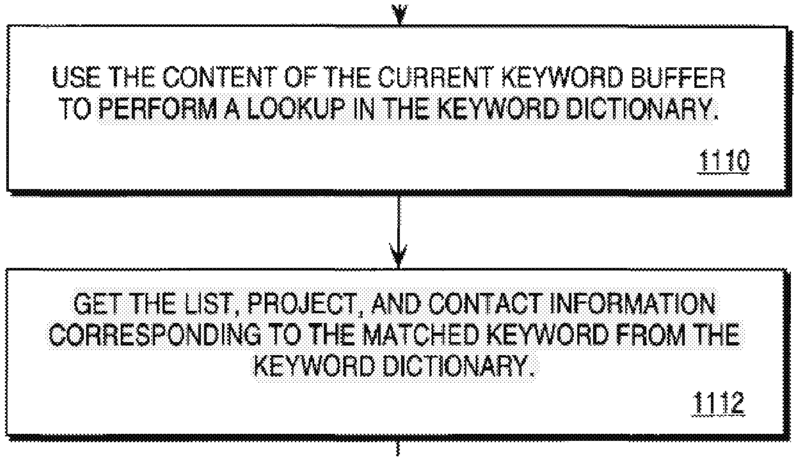
Claims 10 and 11 of Allen provide a useful summary of some of its teachings:

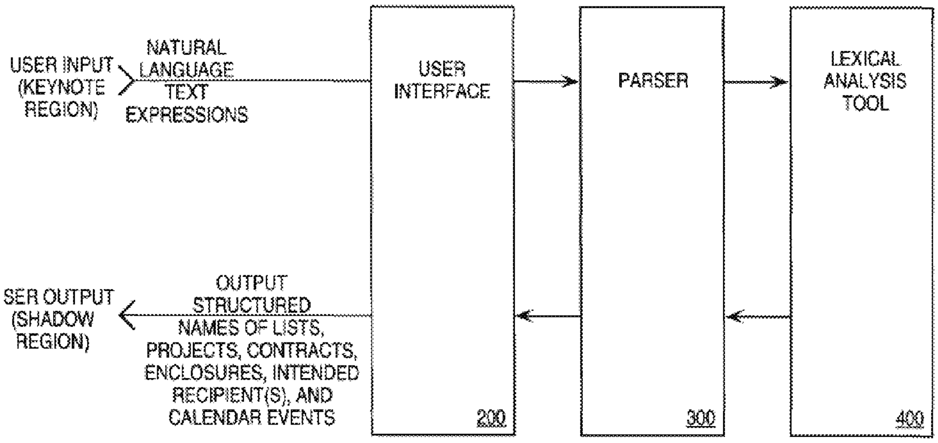
> "10. A system for processing text expressions to facilitate organization, the system comprising:
>
> a user interface for receiving user input in a natural language format;
>
> a parser for extracting key words from the user input and linking the user input to information objects corresponding to the key words; and
>
> the user interface further for displaying the user input and the linked information objects to the user.
>
> 11. The system of claim 10, wherein the information objects comprise one or more of the following: lists, projects, contacts, e-mail addresses, enclosed document identifiers, and events having date/time for use in a calendar."

This ground is different from grounds 1 and 2 for at least several reasons. First, this ground is based on anticipation, not obviousness. Second, the type of document (a "keynote" in Allen versus an email in Goodhand) is different, as is the single execute command (a keyboard press in Allen versus a field change, "check names" or "send message" command in Goodhand).

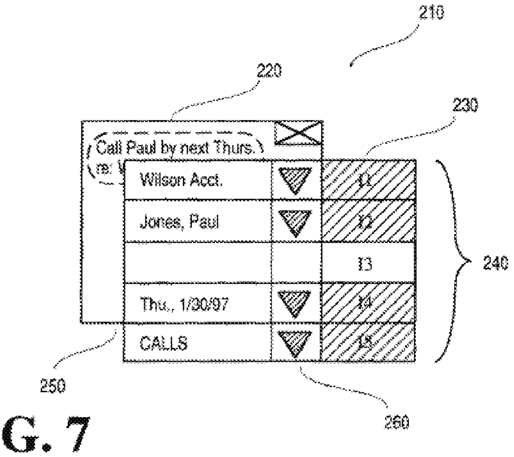
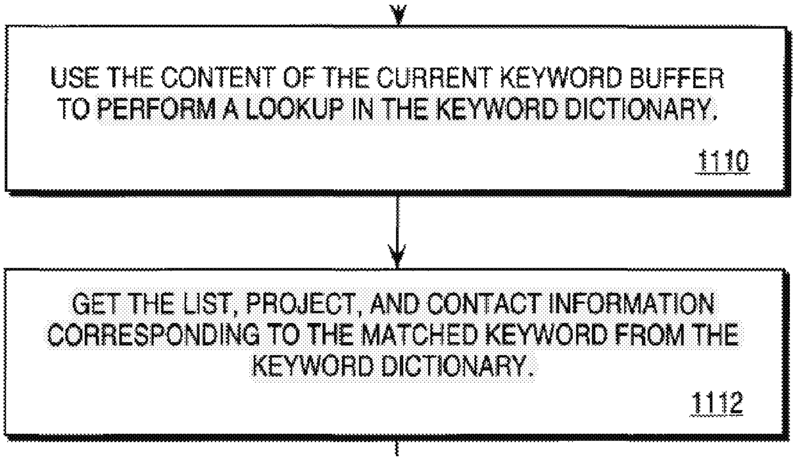
Allen anticipates claims 1, 2, 7-11, 13-17, 22-23, 28-30, 35-38, 43-46, 57, 62-66, 68, 73-77, and 79. An element-by-element mapping of these claims is provided in the following:

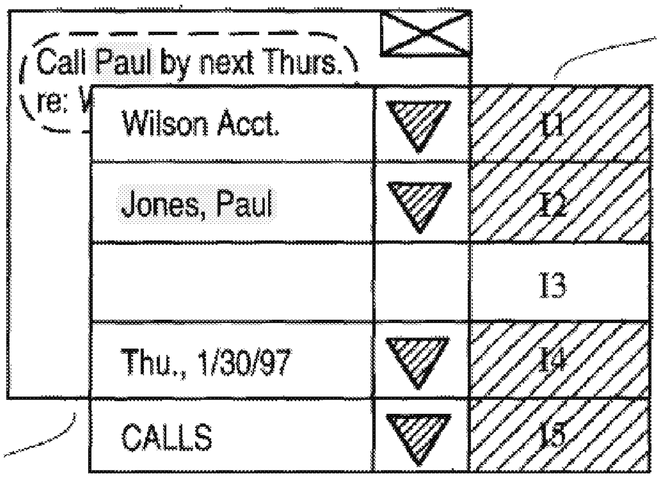
'853 Claims	Allen Disclosure
1. A computerized method for information handling within a document created using an application program, the document including first information provided therein, the method comprising:	Allen discloses a computerized method for **information handling within a document** created using an application program. Ex. 1002 at ¶274. For example, Fig. 1 of Allen "illustrates a typical data processing system" (**computer** system) including a processor 102, a random access memory (RAM) or other volatile storage device 104, a data storage device 107 for storing information and instructions, a display device 121, and "[a]n alphanumeric input device 122, including alphanumeric and other keys." Ex. 1005 at 4:4-31; Ex. 1002 at ¶274.  MASS STORAGE MEDIUM MAIN MEMORY 104 READ ONLY MEMORY MASS STORAGE DEVICE 107 BUS 101 PROCESSOR 102 DISPLAY 121 KEYBOARD 122 CURSOR CONTROL DEVICE 123 HARD COPY DEVICE 124 COMMUNI-CATION DEVICE BUS 103 Allen also discloses application programs such as an "intelligent note editor" (**application program**)(Ex. 1005 at 5:25) with which a user may enter or create text input expressions referred to as "keynotes" (**documents**). Keynotes can include "an action, a memo, a personal keynote, a shared keynote, an action request, an FYI (for your information) message, or one of several other different types of keynotes." Ex. 1005 at 5:38-42; Ex. 1002 at ¶274. Keynotes can include one or more keywords (**first information**). Ex. 1005 at 5:24-31; 15:1-7, Figure 14; Ex. 1002 at ¶274.

'853 Claims	Allen Disclosure
[1a] providing a record retrieval program;	Allen provides a keyword parser (**record retrieval program**) that retrieves records from a "keyword dictionary", and then from a "keyword record list". This is shown in Fig. 11, the relevant portion of which is shown here, with highlighting:  USE THE CONTENT OF THE CURRENT KEYWORD BUFFER TO PERFORM A LOOKUP IN THE KEYWORD DICTIONARY. 1110 GET THE LIST, PROJECT, AND CONTACT INFORMATION CORRESPONDING TO THE MATCHED KEYWORD FROM THE KEYWORD DICTIONARY. 1112 Allen explains this portion of Fig. 11, as follows: "Referring now to FIG. 11, **processing continues for the keyword parser** of keyword and date/time parser 810 at the bubble labeled B. In this situation, the current keyword buffer contains the greatest number of continuous tokens found in the user input keynote that form a predefined keyword in keyword dictionary 852. In this case, **the current keyword in the current keyword buffer is used to perform a look up** for the associated keyword **in keyword dictionary 852** (processing block 1110). **Once the keyword is found** in keyword dictionary 852, **the corresponding keyword definition from keyword definition table 854 is retrieved.** The corresponding keyword definition includes the list, the project, and the contact object information corresponding to the matched keyword from the keyword dictionary (processing block 1112)." Ex. 1005 at 13:20-34 (emphasis added); Ex. 1002 at ¶275-278.

'853 Claims	Allen Disclosure
	The actions in Fig. 11 are carried out by parser 300 (**record retrieval program**) of Fig. 8. Ex. 1005 at 12:46-48; Ex. 1002 at ¶277. The parser is a program separate from the user interface that provides document editing, as shown in Fig. 1, for example:  USER INPUT (KEYNOTE REGION) NATURAL LANGUAGE TEXT EXPRESSIONS USER INTERFACE PARSER LEXICAL ANALYSIS TOOL SER OUTPUT (SHADOW REGION) OUTPUT STRUCTURED NAMES OF LISTS, PROJECTS, CONTRACTS, ENCLOSURES, INTENDED RECIPIENT(S), AND CALENDAR EVENTS 200 300 400 As Allen explains: "A user provides natural language text expressions (i.e., keynotes) representing notes, thoughts, or action requests which are provided to user interface 200. User interface 200 passes these text expressions to parser 300." Ex. 1005 at 5:59-62; Ex. 1002 at ¶¶250-252.
[1b] providing an input device configured to enter an execute command which initiates a record retrieval from an information source using the record retrieval program;	Allen provides an **input device** in the form of a keyboard 122. Ex. 1005 at Fig. 1; Ex. 1002 at ¶279. Allen discloses that each keystroke in a note is an **execute command** that triggers analysis of the document by parser 300. Allen states: "The parser 300 of the present invention is used to analyze this keynote in real-time as the user enters the keynote character by character. Note that the entire keynote is parsed after the entry of each new character." Ex. 1005 at 9:51-54; see also 6:64-67; Ex. 1002 at ¶279. As discussed above under claim element [1a], the parser 300 (**record retrieval program**) also initiates record retrieval from the object dictionary. Because "the entire keynote is

'853 Claims	Allen Disclosure
	parsed after the entry of each new character" Ex. 1005 at 9:51-54), the entry of a keystroke on the keyboard is an **execute command that initiates a record retrieval**. Ex. 1005 at 11:4-14:11; Ex. 1002 at ¶279. It is noted that the '853 patent describes an input device as including a "**keyboard button**". Ex. 1001 at 3:47-53.
[1c] upon a single entry of the execute command by means of the input device:	Allen discloses a **single entry of the execute command**. As discussed with reference to [1b] above, Allen discloses using an input device such as a keyboard to enter a text expression. The entry of the text, or single entry of the execute command, then triggers record retrieval using keywords. Ex. 1002 at ¶280. Each keystroke also causes a separate analysis under step [1d]. Ex. 1002 at ¶280.
[1d] analyzing the document to determine if the first information is contained therein, and	Allen discloses **analyzing the document to determine whether it contains the first information**. Allen discloses using "natural language parsing to identify keywords". Ex. 1005 at 5:27-28. For example, as shown in Figure 7 below, the text input "Call Paul by next Thurs. re: Wilson deal" results in the identification of the keywords "call", "Wilson deal," "Paul" and "next Thursday" that correspond to the objects "CALLS," "Wilson Acct.," "Jones, Paul" and "Thu., 1/30/97". Ex. 1005 at 7:59-8:45; Ex. 1002 at ¶281. The identified text strings in the document are looked up in a keyword dictionary to determine if they are keywords. Ex. 1005 at 12:46-13:6; Ex. 1002 at ¶¶251-257.

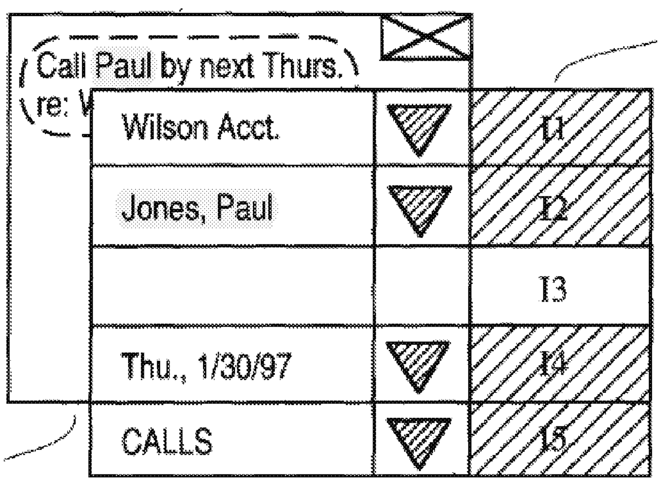
'853 Claims	Allen Disclosure
	 210 220 230 Call Paul by next Thurs. re: W Wilson Acct. Jones, Paul Thu., 1/30/97 CALLS 11 12 13 14 15 240 250 260 FIG. 7
[1e] if the first information is contained in the document, searching, using the record retrieval program, the information source for second information associated with the first information; and	As discussed above for element [1d], Allen discloses locating keywords (**first information**) in a note **document**. As discussed above for element [1a], keywords are used to search the keyword definition table, as shown in the excerpt from Fig. 11:  USE THE CONTENT OF THE CURRENT KEYWORD BUFFER TO PERFORM A LOOKUP IN THE KEYWORD DICTIONARY. 1110 GET THE LIST, PROJECT, AND CONTACT INFORMATION CORRESPONDING TO THE MATCHED KEYWORD FROM THE KEYWORD DICTIONARY. 1112 This results in the retrieval of information (**second information**) **associated with** the keyword (**first information**). The second information includes the "keyword definition". As explained by Allen: "[t]he corresponding keyword definition includes the list, the project, and the contact object information corresponding to the matched keyword from the keyword dictionary (processing block 1112)." Ex. 1005 at 13:29-34; Ex. 1002 at ¶282.

'853 Claims	Allen Disclosure
	The types of second information are explained in more detail below under claim element [1f]. The second information also includes "supplemental information". Ex. 1005 at Abstract; Ex. 1002 at ¶282.
[1f] when the information source includes second information associated with the first information, performing at least one of, (a) displaying the second information, (b) inserting the second information in the document, and (c) completing the first information in the document based on the second information.	Allen teaches that when an information object (**second information**) associated with a keyword (**first information**) is found, it is displayed to the user. Allen states: "a parsing device for identifying the keyword in the input text expression, the parsing device including functions for linking the input text expression to the information object based on the keyword identified in the input text expression; and 4) a **user output device for displaying to the user the identity of the information object** to which the input text expression was linked." Ex. 1005 at Abstract) (emphasis added); Ex. 1002 at ¶¶283-284. *See also* Allen, claims 2, 9-10 and 12, 2:59-3:11. For example, as shown in Fig. 7, the relevant portion of which is reproduced below with added highlighting, when the user types a note having the keyword "Paul", the system recognizes and displays the full contact name "Jones, Paul". Ex. 1002 at ¶283.  Call Paul by next Thurs. re: W Wilson Acct. I1 Jones, Paul I2 I3 Thu., 1/30/97 I4 CALLS I5 Allen explains this as follows: "As a result of parsing input keynote 222, parser 300 has linked the reference to 'wilson deal' in input

'853 Claims	Allen Disclosure
	keynote 222 to the previously specified 'Wilson Account' project object. The linked project object 'Wilson Account' is displayed in region 250 adjacent to corresponding icon I1. Similarly, parser 300 has linked the reference to 'Paul' in input keynote 222 to the previously specified contact object 'Paul Jones'. The linked contact object 'Paul Jones' is displayed in region 250 adjacent to its corresponding icon I2. The parser 300 has linked a date/time calendar event object as a result of parsing the 'next Thursday' text in keynote 222. This processed time/date calendar event object is displayed in region 250 of shadow 230 adjacent to the corresponding icon I4. Finally, parser 300 has linked the keyword 'call' in input keynote 222 to the previously specified 'Calls' list previously defined as a list object. The identification of the linked Calls list is displayed in region 250 of shadow 230 adjacent to the corresponding icon I5." Ex. 1005 at 8:28-44; Ex. 1002 at ¶283-285. As noted by Allen, the displayed information can constitute "one or more of the following: lists, projects, contacts, e-mail addresses, enclosed document identifiers, and events having date/time for use in a calendar." Ex. 1005 at claim 11; Ex. 1002 at ¶283.
2. The method of claim 1, further comprising one of the following steps: storing the first information in the information source if no second information associated with the first information is found in the information source during said searching step, changing the second information in the information source based on one of differences and similarities	Allen discloses the step of **"storing the first information** in the information source" if no second information is found. Specifically, if a word in a keynote is not recognized as a keyword, the system suggests the word as a keyword, which the user can then select. The process is explained well in claims 26-27 of Allen: "26. The method of claim 21, wherein the step of matching tokens to keywords

between the first information and the second information, completing the first information in the document using the second information, adding one of all and part of the first information to an existing record in the information source associated with one of all and part of the first information, correcting the first information in the document using the second information, adding information about said document to said information source, and adding information about said document to said information source, said added information associated with said second information.	comprises: determining if the token exists in a list of keywords; if a keyword is found, declaring a match; and **if no keyword is found**, suggesting the token as a possible keyword. 27. The method of claim 26, wherein the step of suggesting the token comprises permitting the user to select the token as a keyword." Ex. 1005 at claims 26-27)(emph. add.; Ex. 1002 at ¶286. The process is also explained in Allen at 15:1-15. Using the keyword necessitates that it will be added to the keyword dictionary. Ex. 1002 at ¶287.

7. The method of claim 1, further comprising: providing a user the option of making changes to the second information directly in the information source.	Allen has an object dictionary (**information source**) in the parser (**record retrieval program**) that is constructed from an object database. Allen discloses that the user can make changes to the object database that are synched to the parser. Allen states: "Of course, the parser 300 must be kept in synchronization with the data in the object database 850; changes in the object database 850 should be reflected in the parser 300. Updates are accomplished using Add, Delete, 25 and Rename function calls. As an example, consider the following situation: a **user deletes an existing project named "Paint Fence".** The application removes the project from the object database 850 and removes (or updates) its associated keywords. **This change must be reflected in the parser 300 and can be done with a single function call**…." Ex. 1005 at 19:22-32; Ex. 1002 at ¶288.

8. The method of claim 1, wherein: the step of using said application program comprises using said application program to enter first information comprising one of a person's name, a person's title, a person's name and address, a business name, a business name and address, a telephone number, and an email address, or a part thereof, into said document; and	As described with reference to Claim 1 and Figure 7 above, Allen discloses that a user uses the note editor (**application program**) to enter text containing keywords (first information). The first information can comprise at least a **person's name** and a **business name**, as shown in Fig. 7, where Paul is a person's name and "Wilson" a business name Ex. 1002 at ¶289:  Call Paul by next Thurs. re: V Wilson Acct. I1 Jones, Paul I2 I3 Thu., 1/30/97 I4 CALLS I5
[8a] the step of searching comprises searching, using the record retrieval program, the information source for second information comprising one of a person's name, a person's title, a person's name and address, a business name, a business name and address, a telephone number, and an email address, associated with the first information.	As described with reference to Claim 1 (limitations [1e] and [1f]), the second information can comprise a **person's name** ("Jones, Paul" in Fig. 7), a **business name** ("Wilson") in Fig. 7, an **email address** (Ex. 1005 at claim 11). Ex. 1002 at ¶289. Allen also discloses that the second information can comprise "contacts", which includes **address information**. Ex. 1002 at ¶289.

9. The method of claim 1, wherein: the step of providing an input device comprises	As discussed with reference to [1b] and [1c] above, Allen discloses that the user may enter keynotes and a parser then analyzes

providing an input device comprising one of a touch screen, a keyboard button, an icon, a menu and a voice command device, and configured to enter an execute command which initiates a record retrieval from an information source using the record retrieval program; and	"this keynote in real-time as the user enters the keynote character by character. Note that the entire keynote is parsed after the entry of each new character." Ex. 1005 at 9:51-54; Ex. 1002 at ¶290. Allen discloses using an input device such as a keyboard to enter a text expression. The entry of the text, or single entry of the execute command, then triggers record retrieval using keywords. Ex. 1002 at ¶290.
[9a] the step of displaying the second information comprises displaying the second information comprising one of displaying a message screen with the second information and providing a voiced response of the second information.	As described with reference to [1f], Allen discloses **displaying the second information**. The display of second information occurs in a shadow region (**message screen**). As stated by Allen: "Once parser 300 has classified the keynote type and has linked the keynote to the associated objects, the linked list, project, contact, associated e-mail addresses enclosed document identifiers, and any calendar event, is passed back to user interface 200 and displayed by user interface 200 in a keynote and shadow region on display device 121." Ex. 1005 at 6:5-11; Ex. 1002 at ¶291. Allen explains that: "Shadow region 230 is a window that appears beneath, or alternatively adjacent to, the keynote region 220 and contains linked object information in data fields that are automatically set as a result of parsing the keynote entered into keynote region 220." Ex. 1005 at 7:3-8; Ex. 1002 at ¶291-293.
10. The method of claim 1, wherein the step of using	As discussed above under claim 1, preamble, Allen discloses using **word processing**

said application program comprises: using one of a word processing program and a spreadsheet program to enter first information into a respective one of a word processing document and a spreadsheet document.	**programs** such as an "intelligent note editor" (Ex. 1005 at 5:25) to create "an action, a memo, a personal keynote, a shared keynote, an action request, an FYI (for your information) message, or one of several other different types of keynotes." Ex. 1005 at 5:38-42; Ex. 1002 at ¶294. The note editor allows the user to enter text, which contains keywords (first information). See claim 1, element [1d].

11. The method of claim 1, wherein the step of providing an input device comprises: Providing an input device configured to enter an execute command which initiates a record retrieval from an information source comprising at least one of a file, a database, a database program, a computer network, and a contact management program, using the record retrieval program.	As discussed under claim 1, element [1a], the information source (object dictionary) constitutes a database, and comprises a keyword dictionary (also a database) and a keyword definition table (also a database). Ex. 1002 at ¶295. Allen also discloses that its data can be stored as files or across computer networks. Ex. 1005 at claim 1; Ex. 1002 at ¶295-296. For example, claims 33 and 34 of Allen state: "33. The system of claim 32, further comprising a database on another system, the database including the information objects corresponding to the keywords. 34. The system of claim 33, wherein the database is accessed through the Internet." Ex. 1002 at ¶295.

13. The method of claim 1, further comprising the step of indicating which part of information in said document is said first information.	Allen discloses that the keywords or first information may be distinguished in the document. In one instance, Allen discloses that "the keywords of a keynote... may be distinctively displayed in the keynote itself. For example, the keyword or keywords... may be displayed in a first color or font type or style." Ex. 1005 at 7:22-27; Ex. 1002 at ¶297.

14. The method of claim 1, further comprising the step of automatically interpreting which part of information in said document is said first information.	Allen discloses automatically identifying keywords (first information) using a parser to tokenize the input text (keynote) and then search for each token in the keyword dictionary. *See* Ex. 1005 at 2:61-3:3:1, 6:64-67; 8:21-45; 15:1-15; Figure 14; Ex. 1002 at ¶298.

Regarding claims 15 and 16, Allen discloses a "computer system" and a "storage medium storing a program". Ex. 1005 at 4:4-5:7; Ex. 1002 at ¶299. In order to meet the language of claims 15 and 16, the computer system and stored program need only carry out one of claims 1-14. Allen teaches that its computer system and stored program carry out claim 1, as shown above. Ex. 1002 at ¶299.

Claims 17, 22-23, 28-30, 35-38, 43-46, 57, 62-66, 68, 73-77, and 79 have the same limitations as claims 8, 9, 10, 11, 13 or 14. The additional dependencies do not affect disclosure of the methods having these limitations, as arranged in the respective claims, sufficient for anticipation. Ex. 1002 at ¶¶61-75, 300.

Claims 17 and 22 have the same limitation as claim 8. Allen teaches this limitation as shown above for claim 8. Ex. 1002 at ¶¶61-75, 300.

Claims 23 and 28-29 have the same limitation as claim 9. Allen teaches this limitation as shown above for claim 9. Ex. 1002 at ¶¶61-75, 300.

Claims 38 and 43-46 have the same limitation as claim 10. Allen teaches this limitation as shown above for claim 10. Ex. 1002 at ¶¶61-75, 300.

Claims 57 and 62-66 have the same limitation as claim 13. Allen teaches this limitation as shown above for claim 13. Ex. 1002 at ¶¶61-75, 300.

Claims 68, 73-77 and 79 have the same limitation as claim 14. Allen teaches this limitation as shown above for claim 14. Ex. 1002 at ¶¶61-75, 300.

CONCLUSION

For the foregoing reasons, the Petitioners respectfully request that Trial be instituted and that claims 1-79 be canceled.

Respectfully submitted,

Dated: Feb. 20, 2014	By:	/Matthew A. Smith/ Matthew A. Smith Registration No. 49,003 Counsel for Petitioners Motorola Mobility LLC and Google Inc.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of the foregoing Petition for Inter Partes Review, together with all exhibits and other documents, was served on February 20, 2014 by electronic mail (**by prior agreement with the Patent Owner**) to the attorneys of record at

SUNSTEIN KANN MURPHY & TIMBERS LLP
125 SUMMER STREET
BOSTON MA 02110-1618

by transmitting the documents to the attorneys' email addresses at: RAsher@sunsteinlaw.com, BSunstein@sunsteinlaw.com, Jstickevers@sunsteinlaw.com, and Dwu@sunsteinlaw.com.

By: /Matthew A. Smith/
Matthew A. Smith
Registration No. 49,003
Counsel for Petitioner