lowered, allowing it to be killed if Android deems it necessary.

As you would expect, when the device is undocked whatever was running in the foreground in the Mobile View (that is, the top tab) becomes the foreground app on the device.

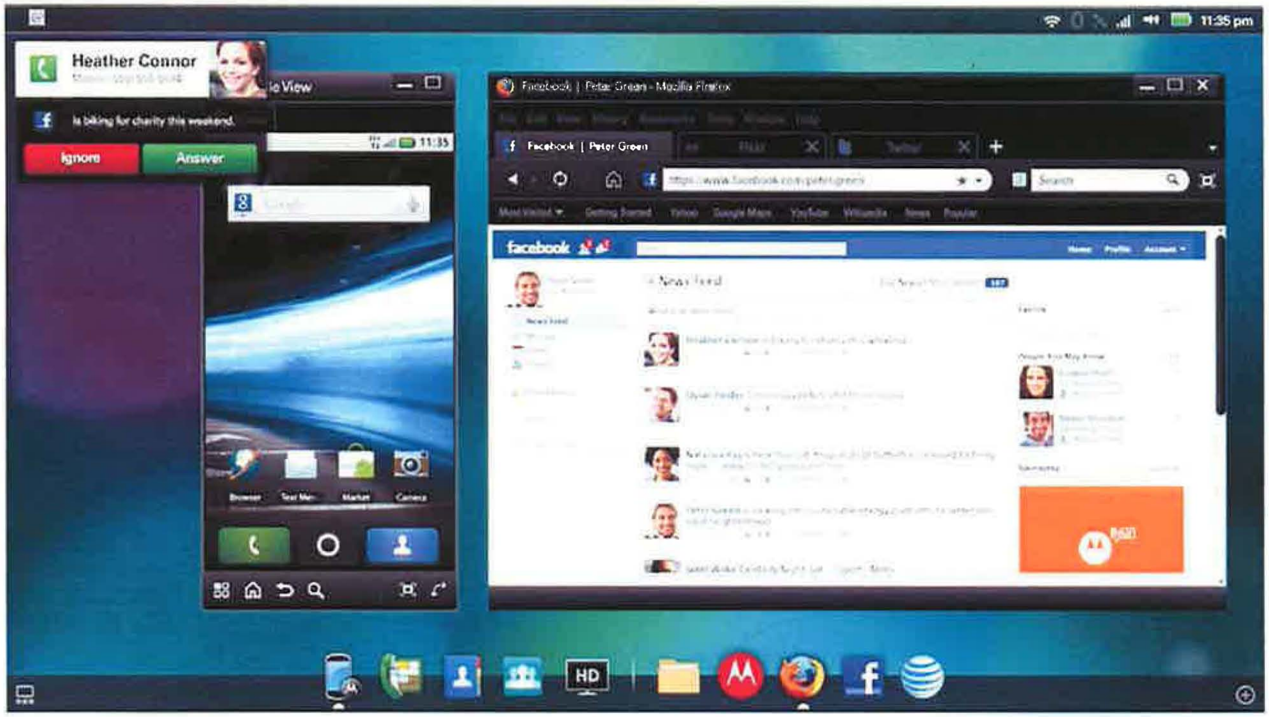
Once undocked, you can use the Webtop Connector app on the device to get a list of all browser pages that were open in Firefox, as well as a list of all of the Firefox bookmarks (bookmarks only; other information such as your browsing history is not included). Selecting one of these web pages or bookmarks in the Webtop Connector app broadcasts an Android Intent, allowing the URL to be opened either in the WebKit browser or in any other app that can handle that URL. For instance, a Google Maps URL could be opened in the WebKit browser, in the Google Maps app, or in a third-party maps application.

For easy access the Webtop Connector app appears in the device's status bar.

When you "re-dock" your device after undocking, the device restores the state of the webtop application as it was prior to undocking and moves the webtop application to the foreground. The Mobile View shows whatever is running on the device at the time it is re-docked (rather than what was showing when the device was undocked). If, while the device was undocked, a low memory situation caused Android to kill Firefox, upon re-docking Firefox is relaunched and its state (all windows and tabs) is restored.

The webtop application desktop

The desktop presented by the webtop application on the connected monitor looks like a typical computer desktop, with a status bar across the top, an "applications tray" across the bottom, and user-sized windows wherever the user chooses to position them. One window (the Mobile View) presents the device's display, while one or more Firefox windows are managed by the desktop browser.



The left side of the status bar shows the notification icons from the left side of the device's own status bar; you can double-click these to see details of the notifications with the results showing in the Mobile View. The right side of the webtop application's status bar shows a clock along with icons for various menus and status indicators, including Wi-Fi, Bluetooth, GPS, HSDPA, and cellular network status and settings; battery level, and audio volume level. Finally, the right side of the status bar has an icon for a status menu that allows you to specify desktop-specific settings (including printer and proxy settings).

As on the Android device, notifications are initially presented in a transient window. Phone calls that come in while the device is showing the webtop desktop are presented in one of these transient windows, along with buttons that allow the user to specify whether the call should be answered or ignored.

The applications tray is an array of icons across the bottom of the screen. A divider separates a fixed set of icons—these cannot be changed—on the left, representing commonly used Android apps (such as the Mobile View itself, the dialer, the Contacts and Messaging apps, and the Entertainment Center, which presents a "10 foot interface" from which you can enjoy music, videos, and pictures) from those on the right that provide one-click access to the desktop Firefox browser, a File Manager, and any user-specified bookmarks or web app shortcuts for use with the Firefox browser. Note that when the user adds a URL to the app tray, they can indicate whether the URL should be opened in a full browser window or in a window with minimal chrome: with little more than a title bar and a window frame.

The user has the option to open multiple Firefox windows, and can create multiple tabs within each window. These windows are fully resizable and can be positioned as desired by the user. While there can only be one window showing the Mobile View, that window, too, can be fully resized and positioned. Windows can be scaled up to the full screen, or minimized.

The user can also specify the wallpaper to be displayed on the webtop application desktop. The user can either select an image specifically for the webtop application, or they can specify that the Android device wallpaper be used. Note that live wallpapers are not supported on the webtop application desktop (they are supported within the Mobile View, however).

Mobile View

The Mobile View is a dedicated window that shows what would normally be on the Android device's display. This window can be resized and positioned as desired. As well, it can be rotated, as if the device itself were switched between portrait and landscape orientation. Outside of the device display itself, the window also presents a control to switch the device orientation and a set of tabs representing the currently running Android apps. This allows the user to quickly and easily switch between the running Android apps with a single mouse click.

For convenience, mouse (or the touchpad presented on the device screen, if a mouse is not connected) actions are mapped to common Android touchscreen gestures:

- Left click is equivalent to tapping the screen
- Right click is equivalent to a touch and hold
- The scroll wheel controls vertical scrolling
- Left click and drag simulates a swipe action

For the most part Android apps do not need to be aware that the device is running the webtop application. Resizing the Mobile View simply scales the device image as it is displayed on the monitor: the device resolution, from the perspective of the running Android apps, does not change. Only if the user elects to change the device orientation (from portrait to landscape, or vice-versa) will the app be notified; this is the standard notification that an app receives when the user physically reorients the device.

Firefox browser

On the device (and thus in the Mobile View), the web browser is WebKit-based. The webtop application, however, presents a full desktop version of the Firefox browser (version 4.0.1 on Gingerbread, version 3.6.13 on Froyo). It is important to note that this is not a mobile browser: it shows full sites rather than their mobile versions. The user agent string for the Firefox browser, unlike the WebKit-based browser on the device itself, does not contain the word "Mobile".

The Firefox browser supports add-ons, themes, and extensions, most notable of which is a hardware-accelerated plug-in for Flash 10.1. It provides excellent support for HTML5, as described later in HTML5 App Support.

The browser supports multiple windows, and each window can have multiple tabs. As well, through the use of the Prism plug-in that is included with the Firefox browser, the user can designate that certain web sites and HTML5 applications be presented in a window devoid of most browser chrome, resulting in a more app-like experience.

Click to call

The Firefox browser also has a "click to call" feature that recognizes US phone numbers: left-click the phone number to initiate a phone call, or right-click to bring up a menu that allows the user to call the number, send an SMS to it, save the number to contacts, or simply copy the phone number to the clipboard.

Bookmark synchronization

When the Firefox browser is launched, it automatically synchronizes with the device browser's top-level bookmarks (bookmarks within folders are not synced). Subsequent changes made to the top-level bookmarks in the Firefox browser are automatically reflected in the on-device WebKit browser.

File manager

The File Manager application gives the user access to internal phone memory, the phone's removable SD card, and any other connected storage (for instance, a USB flash drive plugged into a dock port). From within the File Manager you can move, copy, and delete files. "Smart folders" allow you to see files grouped by type rather than location. For instance, "Phone Images" presents all of the image files found on all recognized storage.

By employing Android's Intent framework, the File Manager enables you to specify which app to use when opening a file. Android apps are included in this; opening a file in the File Manager can cause a specified Android app to be launched in the Mobile View to operate on the file's contents.

Webtop application types

From within the webtop application, the user can interact with any of the following types of application:

- Android applications, running in the Mobile View.
- Android widgets, running in the Mobile View.
- Native webtop applications. Examples of these are the Firefox browser, and the File Manager. Developers cannot create new native webtop applications.
- Online and offline HTML5 web apps, running in a Firefox browser window. These are as defined in the HTML5 specifications, with a manifest.
- Adobe Flash 10.1 applications, running in a Firefox browser window. Note that Adobe AIR is not supported.

The only application types supported by the webtop application are Flash 10.1 and HTML5 applications. Note that the webtop application does not present any form of "shell" and does not support any form of scripting at this time.

Android app support

Because the resolution of the display doesn't change, from Android's perspective when the user resizes or moves the Mobile View window, no notifications are broadcast. Device rotation, whether the user physically rotates the device or clicks on the "rotate" control in the Mobile View window, is handled as dictated by Android.

Although Android's Configuration object is intended to indicate whether the device has a keyboard and some sort of navigation method (other than the touchscreen), this object does not update to reflect a connected keyboard and mouse when the webtop app is running. Instead, it always reflects the configuration of the device itself (which, for devices like ATRIX, is "no keyboard" and "no navigation method"). This will not affect most applications, but be aware that the user won't always be using the onscreen keyboard to enter text (as they would if using the stand-alone device), and won't always be physically touching the screen when your app is registering taps and swipes.

HTML5 app support

The Firefox browser that is the primary component of the webtop application has extensive support for HTML5. HTML5 includes a number of key technologies—some exposed through specialized HTML elements, others accessed from JavaScript—that enable the creation of very powerful apps. The webtop application's Firefox browser includes support for (but is not limited to) the following:

- Offline Application mode (including the ability to detect whether or not the browser is offline)
- Application Cache, which allows you, the developer, to specify which files should be cached by the browser so that they are available when offline
- Web Storage (localStorage) for storage of local data across browser sessions, as key/value pairs
- Web SQL Database for offline application support of SQL databases
- <canvas> element, enabling dynamic rendering of 2D shapes and bitmaps from JavaScript
- <video> element, for streaming of video
- <audio> element, for WAV and OGG playback

- Audio/Video JavaScript controls
- Geolocation, which enables your JavaScript code to obtain the device's current whereabouts
- Native Drag & Drop support
- Cross-origin Resource Sharing: apps can get resources from non-originating web sites
- Web Workers: accelerated JavaScript threads

There are numerous resources on the Internet for creating HTML5-enabled websites and applications. One useful site is www.html5rocks.com, which contains resources (samples, tutorials, and tools) for open web HTML5 developers.

Note that the user agent presented by the Android WebKit browser differs from that presented by the Firefox browser. Because the WebKit browser's user agent contains the word "Mobile", many web sites react quite differently when accessed from the two browsers. Because of this, and because bookmarks from one of the browsers can be accessed from the other, website and HTML5 web application developers should ensure that their creations be usable from both the Android WebKit and Firefox desktop browsers. Be aware that the capabilities of the two browsers differ: the Firefox browser is somewhat more capable than the Android WebKit browser, and thus HTML5 applications, for instance, that run in the Firefox browser may run poorly or not at all on the device browser. Testing, as always, is key to ensuring a good user experience.

App tray icons

When the user elects to add a link to an HTML5 application in the App Tray, there are a few possible icons that can be shown in the tray. The following logic is used to determine which to use:

1. If the URL has an associated "Apple Touch icon" (either it has <link rel="apple-touch-icon" href="..."> or there is a file named apple-touch-icon.png at the root of the host), that icon is used.
2. Otherwise, if the URL has a favicon, it is layered onto a blank icon image (to give the favicon scale; the favicon is not scaled up) and the result is used. When attempting to detect a favicon, the webtop application looks for a link tag with either rel="favicon" or "shortcut icon"; if neither is present it then looks for http://host/favicon.ico.
3. Otherwise, a default icon is used. The default icon differs for bookmarks and web apps.

For the best possible appearance, create Apple Touch icons for your HTML5 apps and websites.

Comment on this page

Created: February 16, 2011
Updated: July 29, 2011

ECCN 5E991.NR: In accordance with United States Export Administration Regulations (EAR), and specifically the Commerce Control List (CCL), this item has been classified 5E991.NR. Export or re-export of this commodity and compliance with the U.S. Export Administration Regulations is ultimately the responsibility of the exporter. For more detailed information related to export or re-export of this item, please consult the EAR at http://www.access.gpo.gov/bis/ear /ear_data.html.

Back to Top

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**