BlackBerry Browser for BlackBerry 7

Version: 7.0

New In This Release

Published: 2011-05-31
SWD-1677266-0531030407-001

Contents

Overview: The BlackBerry Browser

1

The BlackBerry® Browser is a full-featured browser that is designed to render and support most existing web content on a mobile device. Since BlackBerry 6, the BlackBerry Browser has been built upon WebKit, an open source browser engine developed by contributors from several different organizations working together to implement key W3C® standards.

On both BlackBerry smartphones and BlackBerry tablets, the BlackBerry Browser is powered by the full WebKit browser engine, rather than a lightweight mobile version of the engine. The full version of WebKit allows the BlackBerry Browser to more closely approximate the desktop browsing experience. With the full version of WebKit, the BlackBerry Browser supports most commonly used web standards as well as many of the "bleeding-edge" features of HTML 5, its related web APIs, and CSS 3.

Because BlackBerry smartphones and BlackBerry tablets are mobile devices, the BlackBerry Browser is designed to manage the unique constraints of the mobile browsing environment, while striving for desktop-quality rendering and performance. For example, the browser provides a number of usability features designed to improve the browsing experience on small screens, such as tapping or clicking to zoom. In addition, the BlackBerry Browser maintains a constant focus on network efficiency, to both minimize potential costs to the user and the burden on constrained wireless networks.

The browser also offers access to a variety of unique features not available to most desktop browsers, such as unique touch events, and access to mobile-centric features such as the accelerometer and location-based services. The BlackBerry Browser supports several BlackBerry JavaScript® extensions which allow you to provide a more integrated experience from the browser.

New features in this release

2

Web Inspector

Web Inspector is a built-in feature of the BlackBerry® Browser that allows developers to inspect and debug their web source code. Web Inspector allows developers to inspect web content that is displayed through the browser, or in standalone applications that use the browser engine, such as BlackBerry® WebWorks™ applications. Web developers can use Web Inspector to manipulate the DOM, edit and debug JavaScript® code, analyze resource requests, and audit the performance of your web content and web applications in near real-time.

Developers can access Web Inspector functionality using a BlackBerry smartphone or BlackBerry tablet, or using a BlackBerry Smartphone Simulator or BlackBerry Tablet Simulator.

The BlackBerry Browser uses a client-server architecture to make Web Inspector functionality available to developers. The BlackBerry Browser acts as a web server. It serves the page over HTTP over a Wi-Fi® connection; developers inspect the content remotely on a desktop browser. Developers can use any WebKit-based desktop browser on the same Wi-Fi network to navigate to the IP address and port number used by the BlackBerry Browser and begin inspecting the code.

By default, Web Inspector is disabled in the browser. To use the Web Inspector, it must be in the BlackBerry Browser options. Once Web Inspector is enabled, the browser provides the IP address and port number it will use to serve the content.

Access to native events

The BlackBerry® Browser now supports two new meta tags:

- `cursor-event-mode` allows you to define how the browser handles events such as trackpad clicks
- `touch-event-mode` allows you to define how the browser handles events such as touch screen gestures

In the past, access to certain user events was not possible from within web content because of a number of usability-related features implemented by the browser. The introduction or the `cursor-event-mode` and `touch-event-mode` meta tags allows developers to disable these UI features for a web page, so that browser passes the entire array of events to the web page.

By default, the BlackBerry Browser captures and processes most trackpad clicks or touch events at the UI level to allow for improved usability on small screens. For example, on BlackBerry devices with a trackpad, when the user clicks the track pad on an area of the content that is not a link, the browser zooms into that content block. On devices with a touch screen, users can swipe to scroll, double tap to zoom in to a content block, or touch and hold to display the context menu. In addition, on touch screen devices, most touch events are converted by the browser directly into mouse events; this behavior allows the browser to replicate the way desktop browsers manage user interaction with forms and links, but prevents web content from accessing touch events and gestures as input.

By defining event-handling behavior with the `cursor-event-mode` and `touch-event-mode` meta tags, web developers have access to the same interaction model available to Java® developers in the BlackBerry® Java® SDK. For example, web developers can track both the direction and distance of a swipe event, and respond differently based on the swipe direction or distance. Access to these events gives web developers the opportunity to create more dynamic and robust games and applications for the BlackBerry Browser.

Web Sockets API

The BlackBerry® Browser now includes support for the Web Sockets API, which is designed to permit near real-time communication with web server. A web socket is a bi-directional communication channel between a web server and a browser that allows messages to be sent back and forth without requiring an HTTP request and response each time.

The Web Socket API is designed to hide much of the complexity of socket communication. A socket is opened when a `WebSocket` object is constructed. Once the socket connection is established, the browser and server can send data back and forth. Once the socket is closed, it cannot be reopened; a new `WebSocket` objected must be constructed.

Web sockets can be an efficient solution for a web applications that require regular communication with a server. Because a web socket is established once and used repeatedly without the need for communication over HTTP, it can greatly reduce the overhead of sending multiple HTTP headers back and forth. In addition, web sockets can eliminate the need for inefficient communication practices such as polling, in which the browser checks repeatedly for new data on the server. Because a web socket is an open communication channel, the browser only needs to listen for incoming messages. When the server has new data, it sends a new message to the browser. The browser receives the message as it arrives.

Web sockets are not subject to the same-origin policy. The browser can open a socket with a server that is different from the server that originally served the web page.

By default, support for web sockets is disabled in the BlackBerry Browser. To establish a web socket between the browser and a server, the user must first select the **Enable Web Sockets** setting in the BlackBerry Browser options. Because this technology is dependent upon a user setting, web developers should provide alternate content in the event that web sockets are not available.

Embedded media

The BlackBerry® Browser now includes full support for HTML 5 embedded media, excluding the `<track>` element.

The embedded media feature of HTML 5 includes:

- the `<audio>` and `<video>` elements, which let you embed audio or video into your content without the need for a plug-in
- the `<source>` element , which lets you specify a media resource; multiple `<source>` elements can be included to specify a prioritized list of media resources
- a scripting API that lets you control playback
- a set of media-related events

Using `<audio>` or `<video>` elements to embed media into your content allows the media to be played within the web content without requiring a third-party plug-in, and without opening the file in the Media application and hiding the web page. Developers can also access and manipulate embedded media through the DOM using JavaScript® and CSS.

Related resources

3

Reference Guides

- *BlackBerry Browser Content and Standards Support Quick Reference*
- *BlackBerry Browser HTML Reference*
- *BlackBerry Browser CSS Reference*
- *BlackBerry Browser JavaScript Reference*

Glossary

AJAX
Asynchronous JavaScript® and XML

API
application programming interface

CSS
cascading style sheet

DES
Data Encryption Standard

DOM
Document Object Model

HTML
Hypertext Markup Language

HTTP
Hypertext Transfer Protocol

HTTPS
Hypertext Transfer Protocol over Secure Sockets Layer

IP
Internet Protocol

IPPP
Internet Protocol Proxy Protocol

JSON
JavaScript® Object Notation

KML
Keyhole Markup Language

LAN
local area network

MIME
Multipurpose Internet Mail Extensions

MPEG
Moving Picture Experts Group

NTLM
NT LAN Manager

PAP
Push Access Protocol

SSL
Secure Sockets Layer

SVG
Scalable Vector Graphics

Triple DES
Triple Data Encryption Standard

TCP
Transmission Control Protocol

TLS
Transport Layer Security

URI
Uniform Resource Identifier

WAP
Wireless Application Protocol

WLAN
wireless local area network

WML
Wireless Markup Language

WTLS
Wireless Transport Layer Security

WTP
WAP Transaction Protocol

XHTML
Extensible Hypertext Markup Language

XML

Extensible Markup Language

Provide feedback

5

To provide feedback on this deliverable, visit www.blackberry.com/docsfeedback.

Legal notice

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, IN NO EVENT SHALL RIM BE LIABLE FOR ANY TYPE OF DAMAGES RELATED TO THIS DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN INCLUDING WITHOUT LIMITATION ANY OF THE FOLLOWING DAMAGES: DIRECT, CONSEQUENTIAL, EXEMPLARY, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR AGGRAVATED DAMAGES, DAMAGES FOR LOSS OF PROFITS OR REVENUES, FAILURE TO REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES, AND INDEPENDENT CONTRACTORS.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing, or using any Third Party Products and Services, it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers might not offer Internet browsing functionality with a subscription to the BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM's products and services may require one or more patent, trademark, copyright, or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM's products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations, or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server, BlackBerry® Desktop Software, and/or BlackBerry® Device Software.

The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.

Research In Motion Limited
295 Phillip Street
Waterloo, ON N2L 3W8
Canada

Research In Motion UK Limited
Centrum House
36 Station Road
Egham, Surrey TW20 9LF
United Kingdom

Published in Canada