QUALCOMM®

Binary Runtime Environment for Wireless™

BREW™ Browser 2.0
Technical Specification

brew™

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA. 92121-1714
U.S.A.

This manual was written for use with the BREW™ Browser software version 2.0.0. This manual and the BREW Browser software described in it are copyrighted, with all rights reserved. This manual and the BREW Browser software may not be copied, except as otherwise provided in your software license or as expressly permitted in writing by QUALCOMM Incorporated.

BREW™ Browser 2.0 Technical Specification

80-D4388-1 Rev. B

April 25, 2003

Contents

BREW™ Browser 2.0 Technical Specification   4

BREW™ Browser 2.0 Technical Specification

1 Technical specification overview

This document is a high-level functional overview of the Binary Runtime Environment for Wireless™ (BREW™) Browser. It describes how BREW Browser interacts with BREW, the handset, and the Internet.

BREW Browser is a traditional web browser. It displays HTML pages with inline images or forms. It allows the user to select links and submit forms that retrieve information from other pages. For retrieval of information, BREW Browser uses established Internet protocols, such as HTTP, SSL, and DNS.

BREW Browser extensively leverages the capabilities of the BREW platform. Component software support in BREW allows components to be enhanced and extended independently of BREW Browser. As a result, the proper references for certain characteristics of the browser, such as HTML support, are documents specific to a release of BREW, not BREW Browser. This document describes behaviors specific to the BREW Browser implementation, and identifies behaviors that are dependent upon the underlying platform.

For minor updates, clarifications, and corrections to this document, see the Release Notes for the appropriate version of BREW Browser.

2 BREW issues

This section describes how BREW Browser can interact with other BREW applications, and how system-wide settings in BREW affect the behavior of BREW Browser.

2.1. Files

In addition to program files (BAR, MOD, MIF, and so forth), BREW Browser maintains a number of databases for persistent storage. These files are within the BREW Browser application directory—invisible to users and other applications—so the file contents and structure are not documented.

The following table lists the databases used by BREW Browser and the disk usage for each database.

Database	Disk usage
Prefs	Typically under 1 kilobyte.
Cookies	Stores all unexpired cookies.
Bookmarks	No limit. Each bookmark added by a user will consume an amount of space slightly larger than the URL and title associated with the bookmark.
GoTo History	Should not exceed 2 kilobytes.
Cache	Limited to 128 kilobytes.

2.2 Resources and localization

BREW Browser is fully internationalized and has been localized to English, Portuguese, Russian, and Romanian. BREW Browser's image resources need no localization.

2.3 ISHELL_BrowseURL()

BREW allows applications to launch other applications by specifying a URL. Similarly, BREW Browser can launch and be launched by other applications.

This is accomplished using the ISHELL_BrowseURL() method in BREW. Specifically, BREW Browser responds to the BREW message EVT_APP_BROWSE_URL, and registers as an application handler for http:. If an application-level handler exists for a URL encountered in browsing, ISHELL_BrowseURL() is called.

3 Network usage

BREW Browser relies on BREW to automatically initiate a network connection in response to use of networking APIs by BREW Browser or other applications. It also relies on BREW's default *linger* timeout to automatically close the network connection after a period of inactivity. Browsing local content, such as the list of bookmarks or built-in help pages, does not count as network activity.

BREW Browser uses the BREW IWeb interface to retrieve web pages and images. IWeb supports one or more protocol engines, including an HTTP protocol engine that handles http://... URLs.

BREW Browser performs one transaction, or document retrieval, at a time. First, the HTML document is loaded, followed by images referenced from the HTML. When the transactions are being handled by BREW's HTTP engine, connections are efficiently maintained by the keep-alive mechanism, so only one TCP connection is required to load an entire page when all images are served from the same server.

BREW's HTTP implementation uses BREW's DNS to resolve host names prior to establishing the TCP connection to the HTTP server. In BREW 2.0, the DNS implementation caches up to five responses, expiring the least recently used. So subsequent retrievals from the same server are somewhat quicker because the DNS query is avoided.

NOTE: Future versions of BREW may enhance DNS or HTTP support, and BREW extensions may make other types of URLs available to any user of IWeb, including BREW Browser.

4 HTTP support

The HTTP protocol engine in BREW handles protocol features that deal with transport-level issues, such as connection state and message parsing. This behavior is BREW-specific. HTTP support, as of BREW 2.0, is outlined below.

- HTTP/1.1 [RFC2616].

- Supports keep-alive connections.

- Does not support chunking.

For more information, see the documentation that pertains to the version of BREW you use.

BREW Browser handles the features left to the application level by BREW. Because BREW Browser uses IWeb, these features could actually apply to protocol engines other than HTTP, as long as they follow the HTTP convention for communicating them with response codes and headers.

For example, BREW's "file:" IWeb engine uses HTTP redirect semantics (Location header and 301 HTTP response code) to direct applications to FTP URLs that begin with file:.

NOTE: The rest of Section 4 highlights HTTP features of interest to content developers.

4.1 Authentication

BREW Browser supports HTTP Basic Authentication according to RFC2617.

BREW Browser holds an authentication cache. Credentials are associated with the URL up to and including the last slash, which differs slightly from RFC2617, but is common practice among modern browsers.

4.2 Redirection

Response codes (300, 301, 302, 303, and 307), and the presence of a Location: header in the response, result in BREW Browser redirecting the navigation to the URI offered in the Location: header. BREW Browser allows a relative URI in Location, which is outside the RFC2616 specification but used in common practice.

4.3 Refresh header

The Refresh header is not standardized, but it is widely used in content, and is almost universally supported by web browsers. BREW Browser accepts Refresh and supports the HTML <META ... > tag in HTTP headers. BREW Browser's implementation of this header was based on an examination of current implementations, as well as a review of publications.

For more information, see the following URL:
http://home.netscape.com/assist/net_sites/pushpull.html.

4.4 Cookies

BREW Browser supports cookies as specified in RFC2109. This means that Set-Cookie is supported, but not Set-Cookie2.

Cookie support	Description
Max-Age versus Expires	The BREW Browser accepts both Max-Age and Expires Set-Cookie attributes, but gives preference to Max-Age. If Max-Age or Expires cannot be parsed, the BREW Browser discards the cookie at application exit.
Caching policies	Max size, Least-Recently-Used (LRU), and so forth, are supported caching policies. BREW Browser discards cookies when the following occurs: • The cookie expires. • The application runs low on RAM. BREW Browser keeps a minimum of 2048 bytes of cookie data, culling according to an LRU scheme.
Cookie rejection	BREW Browser adheres to RFC2109 when rejecting cookies based on domain. The following text is from RFC2109, section 4.3.2 (rejection criteria): *The request-host is a FQDN (not IP address) and has the form HD, where D is the value of the Domain attribute, and H is a string that contains one or more dots.* Common practice is to ignore this restriction.

4.5 Request headers

The following table shows Request header values that are attached to all requests.

Header value	Description
User-Agent	BREWser/{version} (BREW {brewversion}; DeviceId {id}) For example: The 2.0.0.13 build of the BREW Browser running on BREW version 2.1.0.4, on the Synertek s200 sends the following User-Agent header: `"BREWser/2.0.0.13.0.0.x (BREW 2.1.0.4; DeviceId 28000)"`
Accept-Language	The value of the device's dwLang value from ISHELL_GetDeviceInfo().
Referer [*sic*]	The referring URL, if applicable.

Header value	Description
Host	The host name, sent by BREW's HTTP implementation.
Connection	Keep-Alive, sent by BREW's HTTP implementation.

The following table shows header values that are sometimes sent.

Header value	Description
Authorization	Sent if the URL requires authentication.
Proxy-Authorization	Sent if the URL requires proxying, and the proxy URL requires authentication.
Cookie	Sent if the URL has associated cookies.

4.6 Response headers

The following table shows Response headers that are accepted and used by BREW Browser.

Header value	Description
Connection	Keep-Alive or Close is used by BREW's HTTP implementation to manage HTTP keep-alives.
Content-Type/Content-Length	These are parsed and returned by BREW's HTTP implementation.
Set-Cookie	Used for cookies.
Refresh	See 4.3 Refresh header on page 7.
Location	See 4.2 Redirection on page 7.
Proxy-Authenticate/WWW-Authenticate	Contents are used to prompt the user for credentials.

5 SSL

BREW Browser uses the SSL support provided by BREW. Support for cipher suites and the protocol versions area as documented for the version of BREW being used. This section covers application-specific SSL options.

5.1 Root certificates

BREW Browser leverages BREW's root certificate list. See the BREW documentation for a BREW version-relevant list.

5.2 Overrides

BREW Browser allows simple certificate trust overrides. The user is prompted and the certificate for an otherwise *untrusted* server will be (conditionally) accepted until application exit.

6 HTML

BREW Browser uses the BREW IHTMLVIEWER interface with AEECLSID_HTML. For a description of the capabilities of the HTML viewer, see the relevant BREW documentation.