User Agent Caching Model

Candidate Version 1.1 – 27 Feb 2007

Open Mobile Alliance

OMA-TS-UACACHE-V1_1-20070227-C

Contents

1. Scope

This specification defines the caching model for the OMA application environment. The OMA caching model is an implementation of the HTTP/1.1 caching model used on the World Wide Web. The design is intended to allow the adoption of the HTTP/1.1 caching model with no loss of semantics or function while providing support for OMA enabled devices. See [RFC2616] for more information on HTTP/1.1 caching.

Even though WAP is considered as a deprecated technology this specification also contains specific responsibilities for WAP User Agents and WAP Gateways for historical backwards compatibility reasons, Wireless Application Protocol (WAP) is a result of continuous work by the WAP Forum, now continued by the Open Mobile Alliance (OMA), to define an industry-wide specification for developing applications that operate over wireless communication networks. For additional information on the WAP architecture, refer to "*Wireless Application Protocol Architecture Specification*" [WAPArch].

This specification addresses the following issues:

- User agent caching of resources fetched from the network
- Rules for calculation of freshness lifetime of a cached resource.
- Security considerations
- WAP user agent and WAP Gateway responsibilities (included for historical reasons)
    - WML history cache interaction
    - Caching rules for intra-resource interaction within WML decks and WMLScripts
    - WSP time of day header handling
    - WAP Gateway behavior – the role and responsibility of the WAP Gateway in the implementation of reliable caching.

2. References

2.1 Normative References

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1_1, URL:http://www.openmobilealliance.org/

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC2234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997, URL:http://www.ietf.org/rfc/rfc2234.txt

[RFC2616] "Hypertext Transfer Protocol - HTTP/1.1", R. Fielding, et al., June 1999. URL: http://www.ietf.org/rfc/rfc2616.txt

[WAE] "Wireless Application Environment". Open Mobile Alliance™, OMA-WAP-TS-WAESpec-V2_3, URL:http://www.openmobilealliance.org/

[WSP] "Wireless Session Protocol", Open Mobile Alliance™, OMA-WAP-TS-WSP-V1_0, URL:http://www.openmobilealliance.org/

2.2 Informative References

[NTP] "Network Time Protocol, Version 3, Specification Implementation and Analysis", D. Mills, University of Delaware, March 1992. URL: ftp://ftp.isi.edu/in-notes/rfc1305.txt

[NITZ] "Network Identity and Timezone (NITZ); Service description, Stage 1 (Release 6), 3rd Generation Partnership Project, 3GPP TS 22.042, URL: http://www.3gpp.org/

[WML] "Wireless Markup Language Version 1.3", WAP Forum ™, WAP-191-WML. URL:http://www.openmobilealliance.org/

[WAPArch] "WAP Architecture Specification", WAP Forum™, WAP-210-WAPArch, URL:http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

Cache	A local data store used for optimization of access and data exchange and the management thereof.
Content	Digitized work that is processed, stored, or transmitted. It includes such things as text, presentation, audio, images, video, executable files, etc. Content may have properties such as media type, mime type, etc.
Origin Server	The server on which a given resource resides or is to be created. Often referred to as a web server or an HTTP server.
Request	A WSP/HTTP request (see [WSP] / [RFC2616]). Also known as a "WSP/HTTP method request".
Response	A WSP/HTTP response (see [WSP] / [RFC2616]).
Resource	Any component, function, enabler, or application that can send, receive, or process requests
User	An entity which uses services. Example: a person using a device as a portable telephone.
User Agent	Any software or device that acts on behalf of a user, interacting with other entities and processing resources.

3.3 Abbreviations

HTTP	Hypertext Transfer Protocol [RFC2616]
OMA	Open Mobile Alliance
RFC	Request For Comments
TOD	Time Of Day
URI	Universal Resource Identifier
URL	Universal Resource Locator
W3C	World Wide Web Consortium
WAE	Wireless Application Environment
WAP	Wireless Application Protocol
WSP	Wireless Session Protocol
WML	Wireless markup Language

4. Introduction

The OMA caching model is based on HTTP/1.1 caching as defined in [RFC2616]. In addition, a number of extensions and clarifications have been specified to facilitate the operation and interoperability of HTTP/1.1 caching on limited function devices.

This document applies to all OMA user agents, including those with no cache (i.e., a zero byte cache).

5. User Agent Responsibilities

The user agent MUST implement resource caching as described in [RFC2616]. Specially, to promote minimum conformance for User Agent cache interoperability and User agent caching behaviour the following conformance statements as described in [RFC2616] apply:

- A User Agent MUST support the HTTP Cache-Control header including its cache-request and cache-response directives as defined by [RFC2616] section 14.9 Cache-control.
- A User Agent MUST support the "Cache-Control: no-cache" cache-control directive to force an end-to-end (i.e. force any intermediate caches to obtain a new copy of the resource from the origin server) revalidation of its cached resource.
- A User Agent MUST support the “max-age=0” cache-control directive for Specific and Unspecified end-to-end revalidation.
- A User Agent MUST support the Last-Modified entity-header field, If-Modified-Since request-header fields and ETag response-header fields.
- A User Agent MUST support the Date general-header field.
- When the cache of a User Agent receives a cache-control “max-age” directive and an Expires general header the cache-control “max-age” directive MUST take precedence over the Expires general header.
- A cache of a User Agent MUST NOT cache a response that has a Date header value older than an existing cached response for the same resource.A User Agent MUST store Last-Modified entity header field and ETag response-header field values received from an origin server as meta information (See [RFC2616] section 7.1) with the cached resource for use in subsequent conditional requests.

5.1 Freshness lifetime calculation

A User Agent needs to be able to decide whether a cached resource is fresh or stale. Expiration calculations are defined by [RFC2616]. For OMA User Agents the following apply:

- A User Agent MUST support calculation of cache freshness lifetime based on the max-age directive of the HTTP Cache-Control header. See [RFC2616].

- A User Agent MUST support calculation of cache freshness lifetime based on the HTTP Expires header and the HTTP Date header. See [RFC2616].

Note: Conflicts are resolved according to [RFC2616], section 13.2.4.

5.2 Time of Day Clock

The HTTP/1.1 caching model has built-in assumptions about the existence of a time-of-day (TOD) clock in the User Agent. Nearly all wireless communication devices have a real-time clock, and are capable of maintaining a reasonable sense of the TOD when given a means to synchronise their clocks to an outside source.

To achive reliable time-based caching the following applies:

- When a network based time source, e.g. 3GPP Network Identity and Timezone (NITZ), is available the User Agent's internal time-of-day clock SHOULD be synchronized with this network based time source.

- All time-based caching SHOULD be based on a reliable internal clock that is independent of the local timezone either configured manually or through a network based time source.

5.3 User Agent security considerations

The storage of sensitive information in non-volatile storage raises a variety of security considerations. Implementers MUST ensure that private information in the User Agent cache is protected from unintended or malicious access. For example, the cache must not be implemented in a removable storage media.

6. Proxy Responsibilities (Informative)

Proxies must faithfully implement the role of an HTTP/1.1 proxy with respect to caching and cache header transmission. See [RFC2616] for more information.

6.1 Proxy security considerations

The storage of sensitive information in non-volatile storage raises a variety of security considerations. Proxies implementing a caching function must obey all security-related considerations defined in [RFC2616], section 15.7.

7. Historic WAP Responsibilities

For historical and backwards compatibility reasons this section covers WAP user agent and WAP Gateway responsibilities.

7.1 WML-based User Agent Responsibilities

For WML-based user agents there are specific interactions between the cache and the user agent.

7.1.1 Interaction with the User Agent History Mechanism

The interaction between the user agent cache and the WML history is controlled by the `Cache-control: must-revalidate` header. When going "back" to a URI (e.g., a WML `PREV` task), the user agent should attempt to use the cached resource. The browser must support the following interaction between cached resources and the WML history:

- If the cached resource has the `must-revalidate` cache control attribute set, and the cached resource is stale, then the user agent MUST revalidate the resource when going "back" in the history. The original request MUST be made in precisely the same way (e.g., the same method, the same request entity if a POST, etc.). This re-fetch MUST occur without any user interaction.

- If the cached resource does not have the `must-revalidate` cache control attribute, and the cached resource is stale, the user agent MUST NOT revalidate the entity when doing "back" in the history.

In other words, the history normally shows the way things were at the time the user retrieved the resource. If the `must-revalidate` attribute is set on a resource, the history will show the "up-to-date" version of the resource.

7.1.2 Intra-Resource Navigation

Navigation and processing within a single cached resource does not require revalidation unless the content type specifies some other cache validation model. For example:

- Function calls within a single WMLScript compilation unit may occur without revalidation. Only the initial fetch of the compilation unit requires a check of the resource validity.

- Intra-deck navigation within a single WML deck may occur without revalidation. Only the initial fetch of the deck requires a check of the resource validity.

7.2 WAP Gateway Responsibilities

For WAP Gateways there specific responsibilities related to caching.

7.2.1 Time-of-Day

The HTTP/1.1 caching model is somewhat sensitive to a loss of time synchronisation. The WAP Gateway SHOULD maintain a reliable time -of-day clock, and it is RECOMMENDED that NTP (or an equivalent) be used by the WAP Gateway to maintain a reliable real-time base. See [NTP] for more information.

7.2.2 Caching Proxy

If the WAP Gateway caches its WSP responses, it MUST fulfil the role of an HTTP/1.1 caching proxy.

7.3 Time of Day Clock

Section 5.2 defines User Agent responsibilities for time-of-day (TOD) clock in the user agent. The two subsections below define specific WAP Gateway and WAP User Agent responsibilities for TOD

7.3.1 TOD Requests

The WAP Gateway MUST support requests for the current time of day. When the gateway receives a WSP method request that includes a header named `X-WAP-TOD`, it MUST include that header in the response, with the header value set to the Gateway's current time of day. See [WSP] for the format of the X-WAP-TOD header.

7.3.2 User Agent TOD Clock

If the user agent lacks an accurate TOD clock, it SHOULD use the response from a TOD request as the basis for synchronising its real-time clock with the WAP Gateway. For further accuracy, the user agent SHOULD estimate the round-trip time of the request, and add one-half of this value from the TOD response value.

It is RECOMMENDED that user agents containing a TOD clock use this mechanism as a means of determining the skew between the device clock and the network time base.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
WAP-120-UACACH-20010413-a	13 April 2001	Latest WAP Forum released version.

A.2 Draft/Candidate Version 1.1 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS-UACACHE-V1_0	30 May 2006	All	Initial baseline draft, incorporating text from WAP-120-UACACH-20010413-a into the OMA template. Scope section updated. References updated to latest OMA specifications. Definitions aligned with OMA Dictionary. Updated for caching related requirements in Browsing Interoperability work item. Where appropriate, changed to normative upper-case keywords.
	14 Jun 2006	All	Restructured the document to place all historical WAP responsibilities in a separate section 7. Updates according to MAE meeting in Osaka 20060613.
	16 Aug 2006	All	Updates according to mail discussions between Osaka meeting in June 2006 and Beijing meeting in August 2006.
	22 Aug 2006	All	Updates according to MAE meeting in Beijing 20060822.
Draft Versions OMA-TS-UACACHE-V1_1	24 Nov 2006	Title, Section 5, A.1	Version number changed from 1.0 to 1.1 Added conformance statements to section 5 according to OMA-MAE-2006-0375R01-CR_Conformance_statements_for_caching Updated SCR item UACache-C-001.
Candidate Version OMA-TS-UACACHE-V1_1	27 Feb 2007	n/a	Status changed to Candidate by TP OMA-TP-2007-0074R01-INP_Browsing_V2_3_ERP_for_Candidate_Re_approval

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [IOPPROC].

B.1 SCR for Client

Item	Function	Reference	Status	Requirement
UACache-C-001	Implement HTTP caching model, as described in [RFC2616] and as clarified by detailed conformance described in section 5 of this document.	Section 5	M	
UACache-C-002	Support for HTTP cache freshness lifetime calculation based on the max-age directive of the HTTP Cache-Control header.	Section 5.1	M	
UACache-C-003	Support for HTTP cache freshness lifetime calculation based on the HTTP Expires header and the HTTP Date header.	Section 5.1	M	
UACache-C-004	Synchronization of User Agent's internal time-of-day clock with network based time source.	Section 5.2	O	
UACache-C-005	Time-based caching based on a reliable internal clock that is independent of the local timezone.	Section 5.2	O	
UACache-C-006	Cache contents protected from malicious or unintended access.	Section 5.3	M	
UACache-C-007	WML-based User Agent responsibilities	Section 7.1	O	UACache-C-008 AND UACache-C-009 AND UACache-C-010
UACache-C-008	Revalidate stale cache resources when Cache Control header has a value of must-revalidate.	Section 7.1.1	O	
UACache-C-009	Does not revalidate stale cache resources when Cache-Control header does not have a value of must-revalidate.	Section 7.1.1	O	
UACache-C-010	No revalidation on intra resource navigation, including WMLScript intra-compilation unit function call, WML intra-deck navigation.	Section 7.1.2	O	

Item	Function	Reference	Status	Requirement
UACache-C-011	Time base is synchronized with Gateway.	Section 7.3.2	O	

B.2 SCR for Server

Item	Function	Reference	Status	Requirement
UACache-S-001	WAP Gateway responsibilities	Section 7.2	O	UACache-S-003 AND UACache-S-004
UACache-S-002	Time base is synchronized with NTP or an equivalent.	Section 7.2.1	O	
UACache-S-003	Caching of WSP responses in WAP gateway follow HTTP proxy semantics.	Section 7.2.2	O	
UACache-S-004	Gateway support for x-wap-tod header in WSP requests.	Section 7.3.1	O	