



Home
AOL Client
AOL Network
Netscape Gecko
Cookies
HTTP Headers
P3P
Security
FAQ
Glossary
<a href="mailto:Postmaster@AOL">Postmaster@AOL</a>
<a href="mailto:DNS@AOL">DNS@AOL</a>

## AOL Caching Info

[Caching](#) | [Connectivity](#) | [Proxy Info](#) | [Vary Header Info](#)

When using a browser integrated with the AOL client software, AOL members make web requests through a set of caching proxy servers. If the servers have a current copy of a requested web object, it will be served to the member directly from the cache server instead of the request going over the Internet to the origin web server. AOL will cache most types of web objects including JavaScript files, HTML, and images.

AOL's caching servers cache web objects according to header information provided by the originating web server. These headers are cached and served to members along with their respective object. This caching system follows most requirements of the HTTP/1.1 standard. Specifically, AOL follows HTTP/1.1 rules in the areas of determining cacheability and freshness. For more information, visit the [Internet FAQ consortium](#) and refer to Hypertext Transfer Protocol — HTTP/1.1 documentation or RFC 2616.

**IMPORTANT NOTE:** Cookies set in the HTTP headers of a cacheable object are also cached and served to members. Cookie values intended to be unique (customer id, session, id, etc.) should only be set with non-cacheable objects.

### Setting Caching Parameters with HTTP headers:

The following HTTP headers are used by AOL's cache to determine an object's cacheability. Web Servers can be configured to return the appropriate HTTP headers for the caching behavior you determine to be appropriate. This functionality can be used by a webmaster who may prefer to cache only pieces of their web site. The specific information for configuring a web server is server-dependent.

**IMPORTANT NOTE:** AOL's proxy system does not read HTML. Use of the META HTTP-EQUIV tags in the content of web pages (in either HEAD or BODY) does not control the behavior of AOL's proxy caches.

#### Cache-Control: no-store

This object may not be stored in any cache, even the requestor's browser cache.

#### Cache-Control: no-cache

This object may be held in any cache but it must be revalidated every time it is requested.

#### Cache-Control: private

This object can be stored in the requesting browser's cache but not in a shared cache ...

#### Cache-Control: max-age=x

Max-age an object can be stored where x is the number of seconds.

#### Content-Length:

States the size of the data in bytes sent to the client.

#### Vary:

The Vary header directs the cache to store multiple copies of an object based on the header or list of headers stated. For example, if the Vary header is set to vary on User-Agents it would look like this: "Vary: User-Agents".

When the Vary header is set, each version of that object will be cached and served to members making future requests as appropriate. Using the above example, Vary: User-Agents will cause the cache to store several versions of the same object and serve them to the corresponding requests. If Vary: User-Agent is not set, the cache will return the same object for all requests regardless of the User-Agent because the AOL Cache does not vary by default.

### Expires:

Indicates the date and time an object should become stale in a cache. The Expires header takes precedence over the values established when the maximum time in cache is computed (see below) if the header requires the object to expire in less than 24 hours. The AOL cache will not serve a cached object for more than 24 hours without revalidating it with the origin server even if this value is set to expire the object more than 24 hours into the future.

### Last-Modified:

States the time an object was modified. Objects with a "Last Modified" header without an "Expires:" header will be cached for 20% of the age of the object but not more than 24 hours. The age of the object is determined by subtracting the "Last-Modified:" time from the current time:

$$\text{Maximum Time in Cache before Marked Stale} = (\text{Date Served} - \text{Last Modified}) * 20\%$$

If the requested object is determined to be stale, the cache will send an If Modified Since request to the origin server. The cache will update the object if it has been modified. If no changes have occurred, the object will receive updated Date Served headers and remain fresh until it the age equation determines it to be stale again.

## Caching Examples using the Cache Formula

### Example 1:

An object not previously in cache was served on Sun 01 Oct 2000 14:10:56 GMT. It was last modified on Mon 18 Sep 2000 05:24:09 GMT.

This example would be:

$$\begin{aligned} \text{Maximum Time in Cache before Marked Stale} &= ((\text{Sun 01 Oct 2000 14:10:56}) - (\text{Mon 18 Sep 2000 05:24:09})) * 0.20 \\ &= (320:46:47) * 0.20 \\ &= 64:09:21 \end{aligned}$$

According to the calculation, this object should be cached for about 64 hours but this is over the 24 hour maximum allowed by AOL's cache. Therefore, the object will be cached for 24 hours.

The next time this object is requested after the 24 hour time period is elapsed, the cache will send an "If Modified Since" request to the origin server. The server will respond with either the modified object, which will update the cache, or "Not Modified" which will cause the cache to update the headers and cache the object for another 24 hours.

### Example 2:

An object was served Mon 02 Oct 2000 17:02:55 GMT and it was Last Modified Sun 01 Oct 2000 14:10:56 GMT.

This example would be:

$$\begin{aligned} \text{Maximum Time in Cache} &= ((\text{Mon 02 Oct 2000 17:02:55}) - \\ \text{before Marked Stale} &= (\text{Sun 01 Oct 2000 14:10:56}) * \\ &= 0.20 \\ &= (26:51:59) * 0.20 \\ &= 5:22:24 \end{aligned}$$

This object will be cached for a little over 5 hours. If the object is requested from the cache after that time, that cache will check for an update in the same manner as above.(link to above)

**IMPORTANT NOTE:** AOL's cache will NOT request an object from the origin web server until an AOL user asks the cache for the object and the cache determines that the object it has stored is stale or not in cache.

**IMPORTANT NOTE:** If replacing the front page of your website with a ""Temporarily Closed"" page that shares the same URL, make sure the new front page of your website has updated Last Modified tags when you replace the Temporarily Closed page. If the Last Modified date on the temporary page is fresher than the date on the new front page, the cache will continue to store the temporary page.

## Objects that will not be cached by the AOL Cache

There are several conditions when objects will not be cached:

- Objects returned with an error status code.
- Objects not requested with the HTTP ""GET"" or ""HEAD"" methods (ie: ""POST"").
- Objects containing the ""WWW-Authenticate"" or ""Authorization"" HTTP header will not be cached unless they contain the ""Cache-Control:must-revalidate"", ""proxy-revalidate"", or ""public"" headers. For more information on HTTP/1.1 refer to [Internet FAQ consortium](#) and lookup RFC 2616.
- Objects that have neither the ""Expires:"" nor ""Last-Modified:"" headers are not currently cached however, this behavior is subject to change. Use the appropriate HTTP headers to insure objects will not be cached.
- Requests that begin with https:// will not be cached.

---

[Help](#) | [Terms of Service](#) | [Privacy Policy](#) | [Download AOL 9.0](#)

Last Updated: Monday March 15, 2004  
© 2004 America Online, Inc.