

BetterExplained Books For Kindle And Print

Concrete math lessons that slice through the jargon.

[Math, Better Explained on Amazon](#)

[Calculus, Better Explained on Amazon](#)

How To Optimize Your Site With HTTP Caching

[Home](#) › [Guides, Web](#) ›

I've been on a web tweaking kick lately: how to [speed up your javascript](#), [gzip files with your server](#), and know how to set up caching. But the reason is simple: **site performance is a feature**.

For web sites, speed may be feature #1. **Users hate waiting**, we get frustrated by buffering videos and pages that pop together as images slowly load. It's a jarring (aka bad) user experience. Time invested in site optimization is well worth it, so let's dive in.

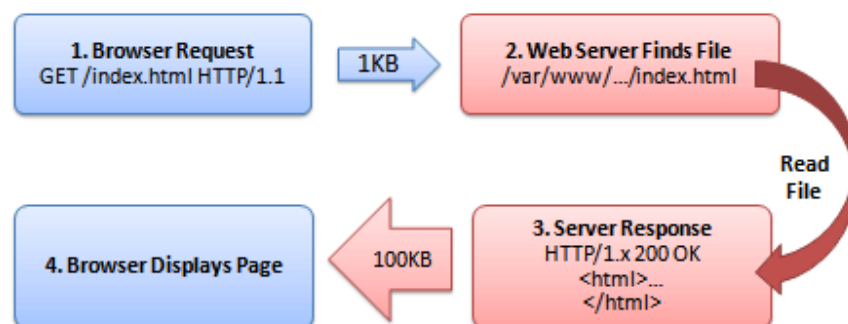
What Is Caching?

Caching is a great example of the ubiquitous time-space tradeoff in programming. You can **save time** by **using space** to store results.

In the case of websites, the browser can save a copy of images, stylesheets, javascript or the entire page. The next time the user needs that resource (such as a script or logo that appears on every page), the browser doesn't have to download it again. **Fewer downloads means a faster, happier site**.

Here's a quick refresher on how a web browser gets a page from the server:

HTTP Request and Response



1. Browser: Yo! You got index.html?
2. Server: (Looking it up...)
3. Server: Totally, dude! It's right here!
4. Browser: That's rad, I'm downloading it!

(The actual HTTP protocol may have minor differences)

Caching's Ugly Secret: It Gets Stale

Caching seems fun and easy. The browser saves a copy of the page and uses this cached (saved) copy on each page that it visits. It even downloads the image ever again and is perfect,

Wrong! What happens when the company logo changes? Amazon.com becomes Nile.com? Google becomes Quadrillion?

We've got a problem. The shiny new logo needs to go with the shiny new site, caches be damned.

So even though the browser has the logo, it doesn't know whether the image can be used. After all, the file may have changed on the server and there could be an updated version.

So why bother caching if we can't be sure if the file is good? Luckily, there's a few ways to fix this problem.

Caching Method 1: Last-Modified

One fix is for the server to tell the browser **what version** of the file it is sending. A server can return a `Last-Modified` date along with the file (let's call it `logo.png`), like this:

```
Last-Modified: Fri, 16 Mar 2007 04:00:25 GMT
```

```
File Contents (could be an image, HTML, CSS, Javascript...)
```

Now the browser knows that the file it got (`logo.png`) was created on Mar 16 2007. The next time the browser needs `logo.png`, it can do a special check with the server:

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

HTTP Cache: Last-Modified



1. Browser: Hey, give me logo.png, but only
2. Server: (Checking the modification date)
3. Server: Hey, you're in luck! It was **not modified**.
4. Browser: Great! I'll show the user the cached version.

Sending the short “Not Modified” message is a **lot faster** than needing to download the file again, especially for giant javascript or image files. Caching saves the day (err... the bandwidth).

Caching Method 2: ETag

Comparing versions with the modification time generally works, but could lead to problems. What if the server's clock was originally wrong and then got fixed? What if daylight savings time comes early and the server isn't updated? The caches could be inaccurate.

ETags to the rescue. An ETag is a unique identifier given to every file. It's like a hash or fingerprint: every file gets a unique fingerprint, and if you change the file (even by one byte), the fingerprint changes as well.

Instead of sending back the modification time, the server can send back the ETag (fingerprint):

ETag: ead145f File Contents (could be an image, HTML, CSS, Javascript...)

The ETag can be any string which uniquely identifies the file. The next time the browser needs logo.png, it can have a conversation like this:

HTTP Cache: If-None-Match



Just like last-modified, **ETags solve the problem of comparing file versions**, except that “if-none-match” is a bit harder to work into a sentence than “if-modified-since”. But that’s my problem, not yours. ETags work great.

Caching Method 3: Expires

Caching a file and checking with the server is nice, except for one thing: **we are still checking with the server**. It’s like analyzing your milk every time you make cereal to see whether it’s safe to drink. Sure, it’s better than buying a new gallon each time, but it’s not exactly wonderful.

And how do we handle this milk situation? With an **expiration date!**

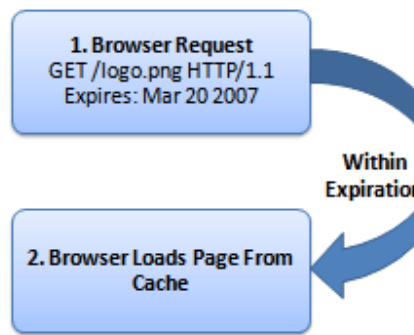
If we know when the milk (logo.png) expires, we keep using it until that date (and maybe a few days longer, if you’re a college student). As soon as it goes expires, we contact the server for a fresh copy, with a new expiration date. The header looks like this:

```
Expires: Tue, 20 Mar 2007 04:00:25 GMT
```

```
File Contents (could be an image, HTML, CSS, Javascript...)
```

In the meantime, we avoid even talking to the server if we’re in the expiration period:

HTTP Cache: Expires



Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Email Address

Join Newsletter!

There isn't a conversation here; the browser has the file.

1. Browser: Self, is it before the expiration date?
2. Browser: Verily, I will show the user the cached file.

And that's that. The web server didn't have to do anything. The user sees the file instantly.

Caching Method 4: Max-Age

Oh, we're not done yet. Expires is great, but it has to be computed for every date. The `max-age` header lets us say "This file expires 1 week from today", which is simpler than setting an explicit date.

Max-Age is measured in seconds. Here's a few quick second conversions:

- 1 day in seconds = 86400
- 1 week in seconds = 604800
- 1 month in seconds = 2629000
- 1 year in seconds = 31536000 (effectively infinite on internet time)

Bonus Header: Public And Private

The cache headers never cease. Sometimes a server needs to control when certain resources are cached.

- `Cache-control: public` means the cached version can be saved by proxies and other intermediate servers, where everyone can see it.
- `Cache-control: private` means the file is different for different users (such as their personal homepage). The user's private browser can cache it, but not public proxies.

- `Cache-control: no-cache` means the file should not be cached. This is useful for things like search results where the URL change.

However, be wary that some cache directives of you are doing special caching of authenticated

Ok, I'm Sold: Enable Caching

First, make sure Apache has `mod_headers` and

```
... list your current modules...  
apachectl -t -D DUMP_MODULES
```

```
... enable headers and expires if not in the list above...  
a2enmod headers  
a2enmod expires
```

The general format for setting headers is

- File types to match
- Header / Expiration to set

A general tip: the less a resource changes (images, pdfs, etc.) the longer you should cache it. If it never changes (every version has a different URL) then cache it for as long as you can (i.e. a year)!

One technique: Have a loader file (`index.html`) which is not cached, but that knows the locations of the items which are cached permanently. The user will always get the loader file, but may have already cached the resources it points to.

The following config settings are based on the ones at [AskApache](#).

Seconds Calculator

All the times are given in seconds ($A0 = \text{Access} + 0 \text{ seconds}$).

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

Using Expires Headers

```
ExpiresActive On
ExpiresDefault A0
```

```
# 1 YEAR - doesn't change often
<FilesMatch "\.(flv|ico|pdf|avi|mov|ppt|doc|mp3|wmv|wav)$">
ExpiresDefault A31536000
</FilesMatch>
```

```
# 1 WEEK - possible to be changed, unlikely
<FilesMatch "\.(jpg|jpeg|png|gif|swf)$">
ExpiresDefault A604800
</FilesMatch>
```

```
# 3 HOUR - core content, changes quickly
<FilesMatch "\.(txt|xml|js|css)$">
ExpiresDefault A10800
</FilesMatch>
```

Again, if you know certain content (like javascript) won't be changing often, have "js" files expire after a week.

Using max-age headers:

```
# 1 YEAR
<FilesMatch "\.(flv|ico|pdf|avi|mov|ppt|doc|mp3|wmv|wav)$">
Header set Cache-Control "max-age=31536000, public"
```

```
</FilesMatch>
```

```
# 1 WEEK
```

```
<FilesMatch "\.(jpg|jpeg|png|gif|swf)$">  
Header set Cache-Control "max-age=604800"  
</FilesMatch>
```

```
# 3 HOUR
```

```
<FilesMatch "\.(txt|xml|js|css)$">  
Header set Cache-Control "max-age=10800"  
</FilesMatch>
```

```
# NEVER CACHE - notice the extra directive
```

```
<FilesMatch "\.(html|htm|php|cgi|pl)$">  
Header set Cache-Control "max-age=0, private"  
</FilesMatch>
```

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

Final Step: Check Your Caching

To see whether your files are cached, do the following:

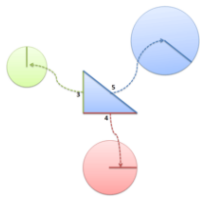
- Online: Examine your site in [Redbot](#) (You'll see the headers returned, and a cache summary on the side)
- In Browser: Use [FireBug](#) or [Live HTTP Headers](#) to see the HTTP response (304 Not Modified, Cache-Control, etc.). In particular, I'll load a page and use Live HTTP Headers to make sure no packets are being sent to load images, logos, and other cached files. If you press ctrl+refresh the browser will force a reload of all files.
- [MaxCDN](#) has a nice roundup of web and browser tools to check HTTP headers

Read more about [caching](#), or the [HTTP header fields](#). Caching doesn't help with the initial download (that's what gzip is for), but it makes the overall site experience much better.

Remember: Creating unique URLs is the simplest way to caching heaven. Have fun streamlining your site!

Join Over 450k Monthly Readers

Enjoy the article? There's plenty more to help you build a lasting, intuitive understanding of math. Join the newsletter for bonus content and the latest



updates.

Email Address

Join Newsletter

Facebook

Twitter

LinkedIn

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Email Address

Join Newsletter!

Other Posts In This Series

 Recommend 2  Tweet  Share

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS

Name

**kat** • 2 years ago

hi there, maybe it should be obvious, but where exactly and how? I mean, I'd like to set 1 week and I'm using wordpress. So what do I do exactly for htaccess?

2 ^ | ▾ • Reply • Share ›

**Sumit Wadhwa** → kat • a year ago

you can use `wp_headers` hook like this
`add_filter('wp_headers', 'function_that_has_headers')` and
add your headers in `function_that_has_headers` function.

^ | ▾ • Reply • Share ›

**El hassan** • 5 months ago

Thank you

^ | ▾ • Reply • Share ›

**Andre White** • 5 months ago • edited

Looks like my browsers are not caching my webpages

<https://forexstore.com/>

Could someone help me with it?

^ | ▾ • Reply • Share ›

**Seep Secreat (nick)** • 10 months ago • edited

Hi, there, If file "jquery-3.3.1.min.js" both on server A and B(different domain), then visitor first access server A, then access server B, what happen to server B? will it use cache from server A or re-download from server B? thanks!

^ | ▾ • Reply • Share ›

**Luiz Wynne** • a year ago

How to implement that in code? What to do on the .htaccess and headers? I saw a lot of theory and no practical...

^ | ▾ • Reply • Share ›

**Sumit Wadhwa** → Luiz Wynne • a year ago • edited

if you're using some CMS, wordpress - drupal - etc,
system, then they must have hooks for developers to add

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Email Address

Join Newsletter!

custom headers before they generate any response (refer to their developer documentation). if using frameworks

Laravel, CakePHP, etc, they mostly implement headers known as a Response Object to which you can add headers to (refer to their documentation). server side languages and scripts provide you a way to add your custom headers. For eg. PHP provides a function to add headers while compiling

Example usage:

```
header('Content-Type: application/pdf');
```

type in response header using PHP

^ | v • Reply • Share ›

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!



Adarshpal Brar • 2 years ago

Any idea how to compress HTTPS response? I know how to compress HTTP responses but there is a security issue with HTTPS response.

^ | v • Reply • Share ›



Sumit Wadhwa → Adarshpal Brar • a year ago

compressing HTTPS response? you can. but like you said, yes, you shouldn't. you can always fallback to mechanisms such as minifying your response resources (images, css, js files, etc). And, most importantly caching. But using any compression algorithm with ssl layer enabled can open up your application to - <https://en.wikipedia.org/wiki/BREACH>.

^ | v • Reply • Share ›



gintare • 2 years ago

thank yo so much, very useful!

^ | v • Reply • Share ›



Vin Boris • 3 years ago

I found it very interesting thanks for sharing the content. i was having a little issue of website caching.

^ | v • Reply • Share ›



Anan • 3 years ago

Great way to sum up the caching attributes!
Thank you very much :)

^ | v • Reply • Share ›



Khan • 4 years ago

It is the clearest article that i have read about caching so far. Thank you.

^ | v • Reply • Share ›



Stijn • 4 years ago



Thanks for writing this article. I use this doc as a reference document for anyone in my team new to web performance testing. Keep up the great work.

^ | v • Reply • Share ›



Jayeah Duggal • 4 years ago

My browsers are not caching my webpages. Please answer my this question
<http://stackoverflow.com/qu...>

^ | v • Reply • Share ›



Anime Decoder • 4 years ago

Could you tell me how to implement in in blog? I tried adding the tag to after the but when i did cache control: private, max-age=0. Any thoughts on how to do this.

^ | v • Reply • Share ›



Vidhya • 4 years ago

Thanks Kalid, Very informative.

^ | v • Reply • Share ›



UniFreak • 4 years ago

Wonderful tutorial, neat, structured, detailed and humored! it's been a very delightful experience reading your article. Thanks very much, sir!

^ | v • Reply • Share ›



penerjemah • 4 years ago

wow this is awesome themes
i will use this

^ | v • Reply • Share ›



boostwebid • 4 years ago

this is awesome,
i will digg it soon

^ | v • Reply • Share ›



Rick • 4 years ago

I read your article and understand most of it.
I do *not* have access to the .htaccess file. I am using BigCommerce.com shopping cart system.

I am wondering if i can hard code something in the image code itself?

if so can you edit the code listed below?

this code would bring up my site logo for example.....

^ | v • Reply • Share ›

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!



peter taulli • 4 years ago

peter taulli great guide ,thanks

^ | v • Reply • Share ›



Max Help Desk Software • 4 years ago

Great stuff! Especially for companies offering
really can't afford a nick of slowdown in their s

^ | v • Reply • Share ›



Max Sequeira • 4 years ago

#Create filter to match files you want to cache
(open crutch)FilesMatch "\.(jpg|jpeg|png|gif|sw
crutch)

Header add "Cache-Control" "max-age=6048
(open crutch)/FilesMatch(close crutch)
:)

^ | v • Reply • Share ›



Max Sequeira • 4 years ago

Hi, just wanted to say thank you very much for such helpful article.

Also to newbies like myself looking to optimize your site for Google
PageSpeed controls, adding the following code to my .htaccess
permitted me to comply at once with their test:

#Create filter to match files you want to cache

Header add "Cache-Control" "max-age=604800"

Thanks again!

^ | v • Reply • Share ›



kalid • 4 years ago

@nab: thanks just fixed.

^ | v • Reply • Share ›



nab • 4 years ago

KNOW how -----> not now how on line 2
thanks for the article !

^ | v • Reply • Share ›



Mansoor Omrani • 5 years ago

And finally someone appeared who could explain caching in
simple terms without confusing the reader. Thanks for your article.
That was great. :)

^ | v • Reply • Share ›



2tor ➔ Mansoor Omrani • 7 months ago

I wish I could agree. I got lost right off at

Ok, I'm Sold: Enable Caching

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain
an intuitive understanding of math.

Join Newsletter!

First, make sure Apache has mod_headers and mod_expires enabled:

... list your current modules...
apachectl -t -D DUMP_MODULES

I admit I have no idea what I'm doing, particular part of my job, and know no talk to you like you know nothing, they acronyms an assumptions.

^ | v • Reply • Share ›



devdeep • 5 years ago

Thanks for this.

^ | v • Reply • Share ›



lagu barat terbaru • 5 years ago

thank you for sharing..

^ | v • Reply • Share ›



Anonymous • 5 years ago

great

^ | v • Reply • Share ›



William • 5 years ago

This is a most clearly article about cache of apache which I saw.
Thanks.

^ | v • Reply • Share ›



Nileshe • 5 years ago

What if files has been changed before cache expiration date? How can we handle such a situation to load updated file instead of cached file even if cache expiry time was greater than file changed time?

^ | v • Reply • Share ›



Zachary S • 5 years ago

This is the best clear and no-nonsense guide I've seen to browser caching. I'm a complete newbie trying to learn this stuff for my site and you helped out a lot. Thanks Kalid!

^ | v • Reply • Share ›



Bruno • 5 years ago

Great!!

Excellent explaining.

^ | v • Reply • Share ›



KP • 5 years ago

Nice Article.

^ | v • Reply • Share ›

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

^ | v • Reply • Share ›



Ram • 5 years ago

awesome article. any layman can understand nice.

^ | v • Reply • Share ›



bryan • 5 years ago

Your article made my day, thank you for sharing

^ | v • Reply • Share ›



software sms gateway • 5 years ago

my mistake. It should be fixed now.

^ | v • Reply • Share ›



Dmitry • 5 years ago

Well written and easy to understand, thanks!

^ | v • Reply • Share ›



Swaragh • 5 years ago

Awesome Article. Best article about HTTP headers so far.

^ | v • Reply • Share ›



Hidayat Mundana • 5 years ago

This time I've been using w3 total cache, but the results of my tests showed F, what should I do

I still have to use the above code?

^ | v • Reply • Share ›



Nandan Kulkarni • 5 years ago

hey, thanks a lot for this very simple and elegant introduction on caching!

^ | v • Reply • Share ›



sam • 6 years ago

If we set the caching using the "Last modified date".

Does server automatically checks whether the file is changed? or should we write separate code to compare the header "If-Modified-Since" and file modified date?

We are using iis, do we have to do something in the iis for comparing the file changes and then return the 304?

^ | v • Reply • Share ›



rozliczenie pit 2014 program • 6 years ago

Hello! I just wanted to ask if you ever have any problems with hackers?

My last blog (wordpress) was hacked and I ended up losing many months of hard work due to no data backup.

Join 450k Monthly Readers

Enjoy the article? Keep reading to gain an intuitive understanding of math.

Join Newsletter!

Do you have any methods to prevent hackers?|

^ | v • Reply • Share ›



Popat Kharat • 6 years ago

Good One !!!

^ | v • Reply • Share ›



Anonymous • 6 years ago

Very Good Article. Thanks for posting

^ | v • Reply • Share ›



henk • 6 years ago

*Join 450k Monthly
Readers*

Enjoy the article? Keep reading to gain
an intuitive understanding of math.

Email Address

Join Newsletter!