

**Take Control of
Cache-Control**



Holger Bartel

Designer/Developer

 Hong Kong

h@foobartel.com

Twitter: [@foobartel](https://twitter.com/foobartel)



Harry Roberts ✓

@csswizardry



How is your knowledge of caching and Cache-Control headers?

♡ 26 6:22 AM - Mar 4, 2019



4% I know it all!

42% I could know more.

54% I'm kinda clueless, tbh.

1,696 votes • Final results

💬 16 people are talking about this



Fact

A majority of the data on the internet is static and unlikely to change over time.

Many of today's web performance issues are related to images, fonts and video.

What is Cache?

Cache Not Cash!



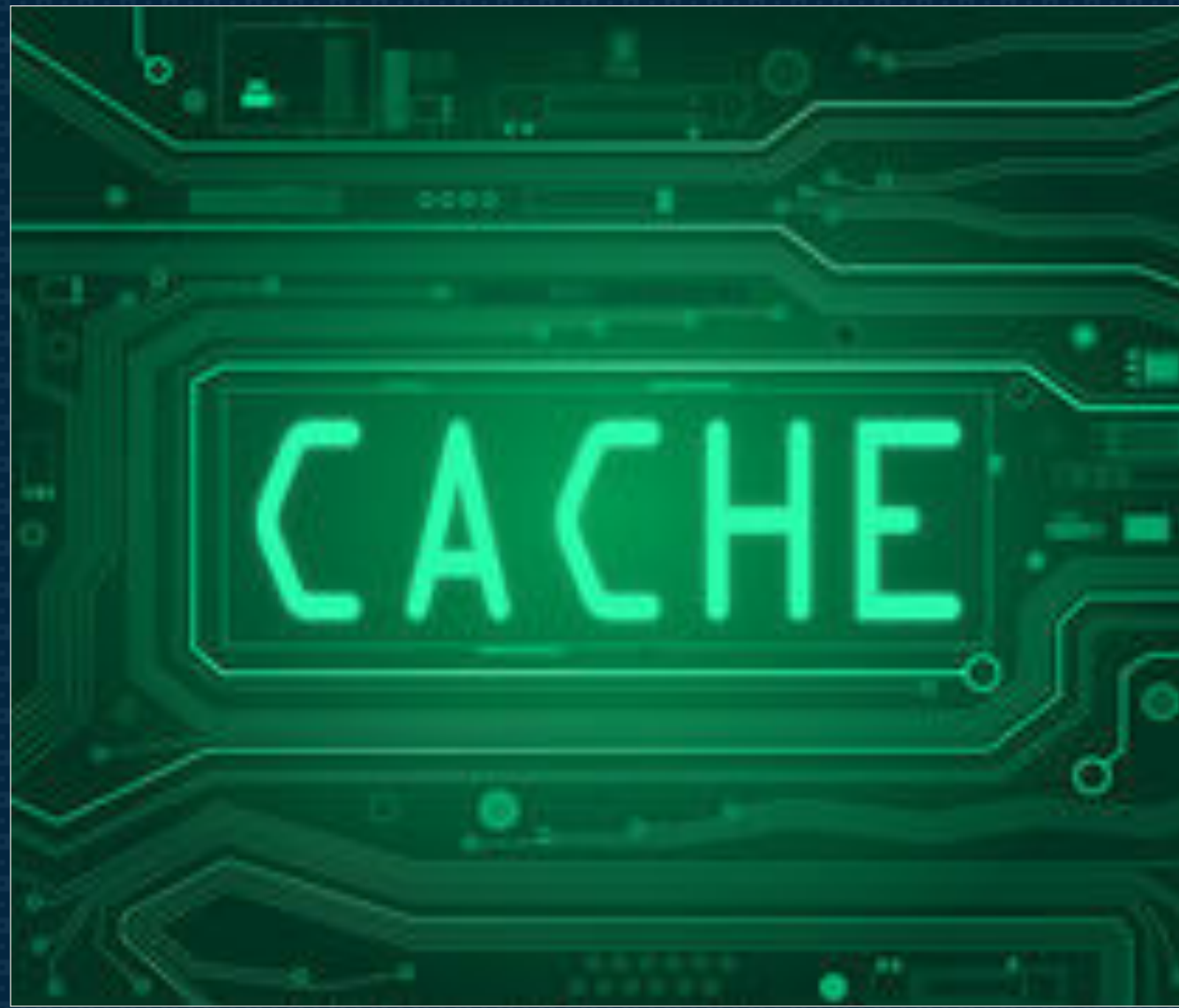
This joke was brought to you by my friend JJ!

古蹟
Heritage

The Conservancy Association Centre
for Heritage (CACHe)

Not!







Cache Is A Storage

- ❖ Store away in hiding or for future use.
- ❖ Helper memory enabling high-speed retrieval of data.

Origin

Late 18th century:

from French, from *catcher* 'to hide'.

```
for ( i = 0; i < count( AllTheThings ); i++ )  
    { ... }
```

Store a value in a variable for better performance.

```
let items = count( AllTheThings );  
for ( i = 0; i < items; i++ ) { ... }
```

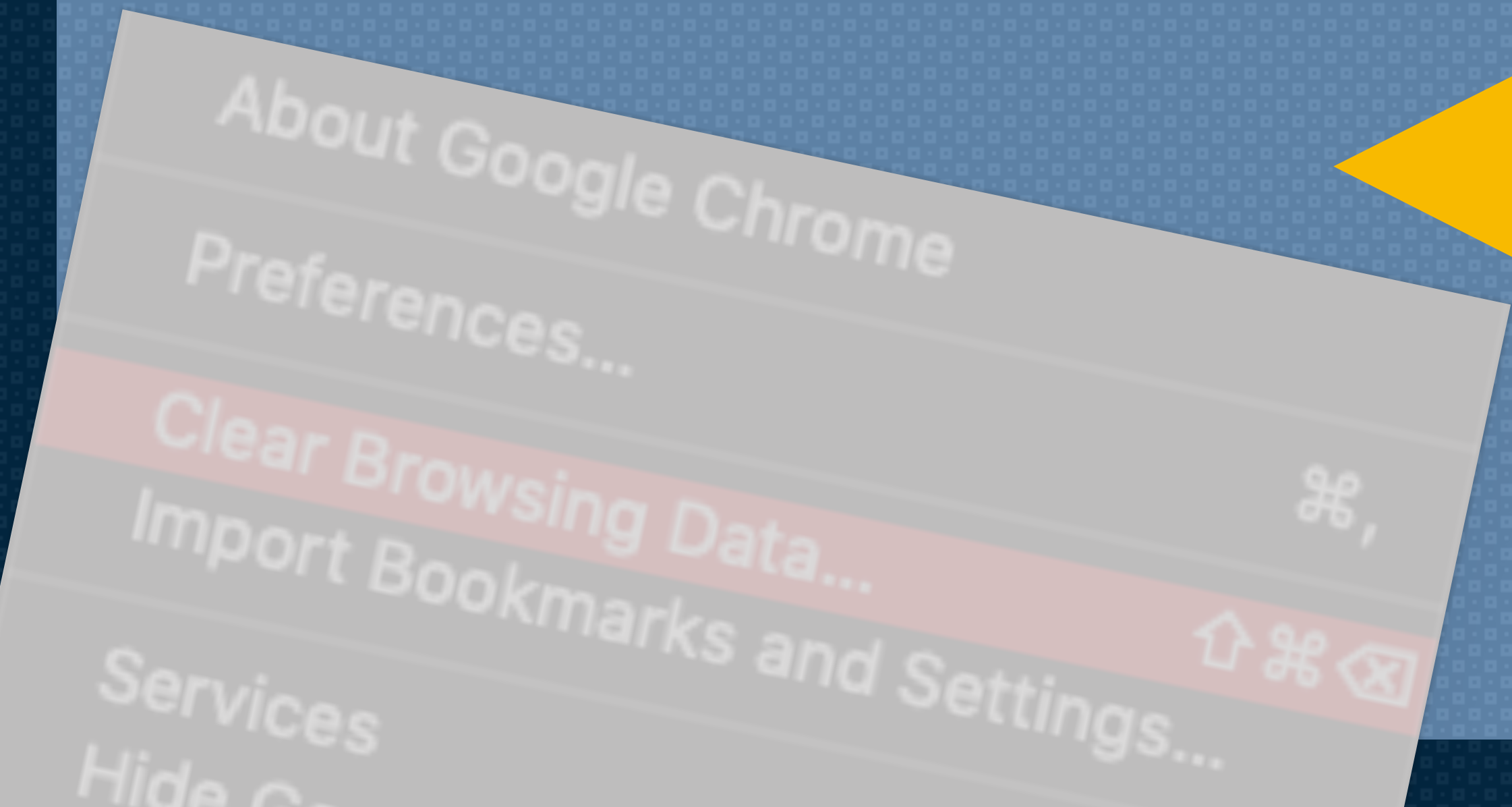
Store a value in a variable for better performance.

Developers know caching best from

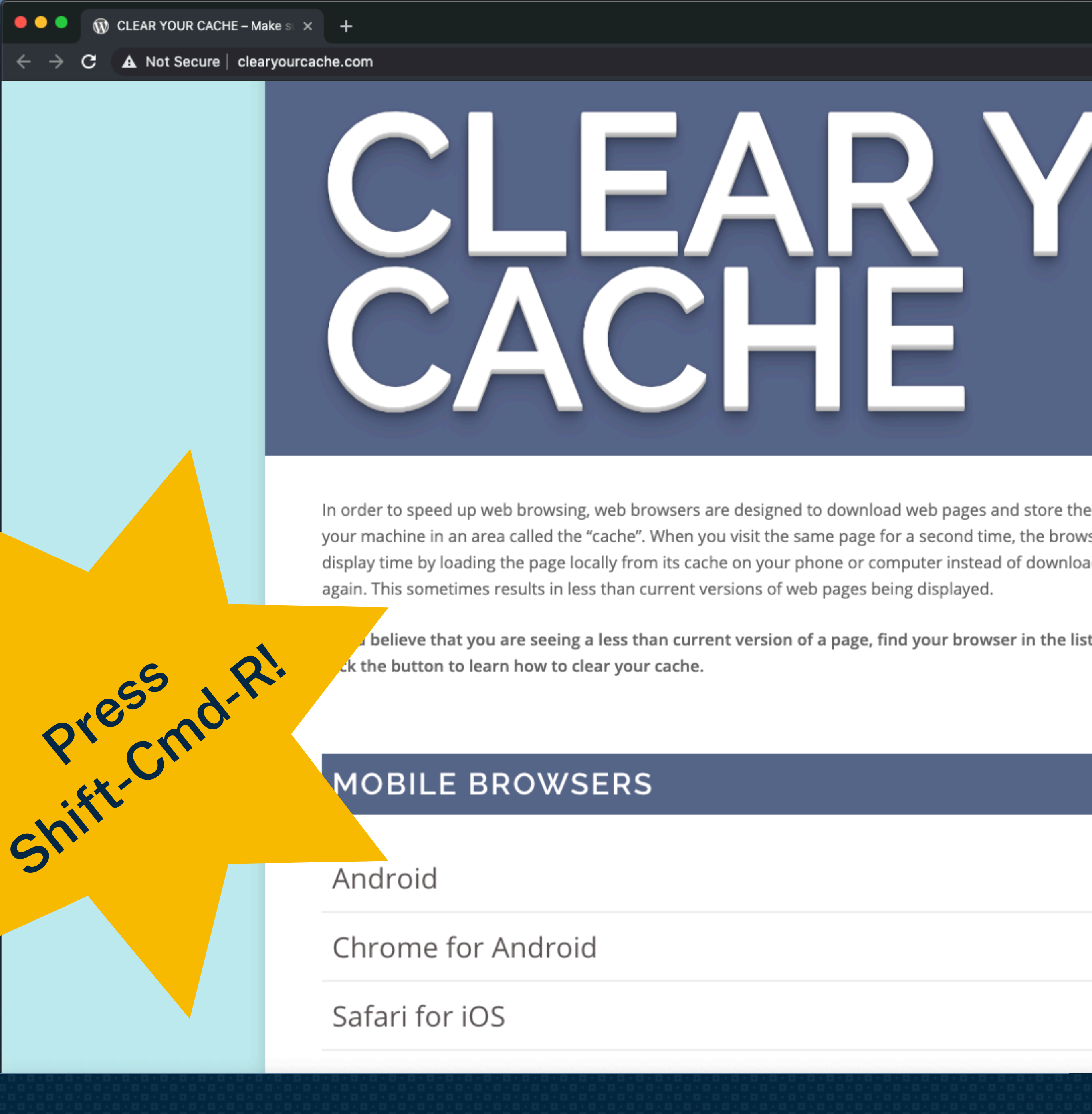
\$clients

F5!

<http://clearyourcache.com>



**Press
Shift-Command-R!**



CLEAR YOUR CACHE

In order to speed up web browsing, web browsers are designed to download web pages and store them on your machine in an area called the "cache". When you visit the same page for a second time, the browser displays the page by loading the page locally from its cache on your phone or computer instead of downloading it again. This sometimes results in less than current versions of web pages being displayed.

If you believe that you are seeing a less than current version of a page, find your browser in the list below and click the button to learn how to clear your cache.

MOBILE BROWSERS

- Android
- Chrome for Android
- Safari for iOS

Blame the Cache!



A good excuse ;)



The cache exists because of a basic assumption made by browser designers:

The internet is slow.

Caching is a technique that stores a copy of a given resource and serves it back when requested.

Why Care?



good old

The First Impression

We always want to make a great first impression.

A fast loading site does just that.



A wooden cutting board with fresh vegetables including green leafy herbs, red radishes, and a cucumber.

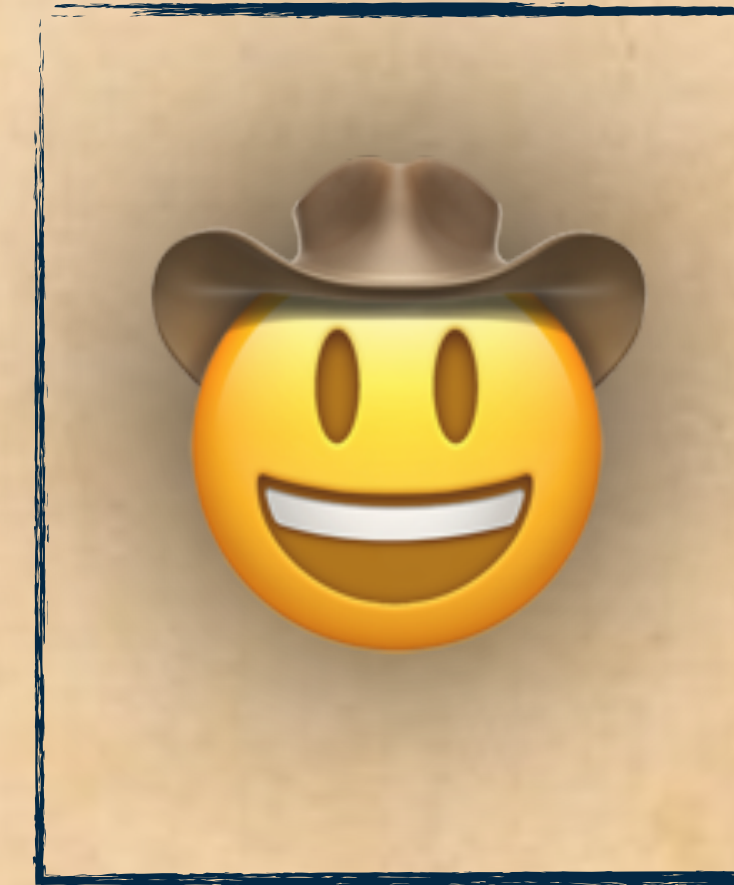
The cache serves
what you already have,
as long as it's
fresh.

A large, bold, white question mark is centered in the upper half of the image. The background is a solid, dark red color.

A AND Great Impression?

How To Delight Our Users

WANTED



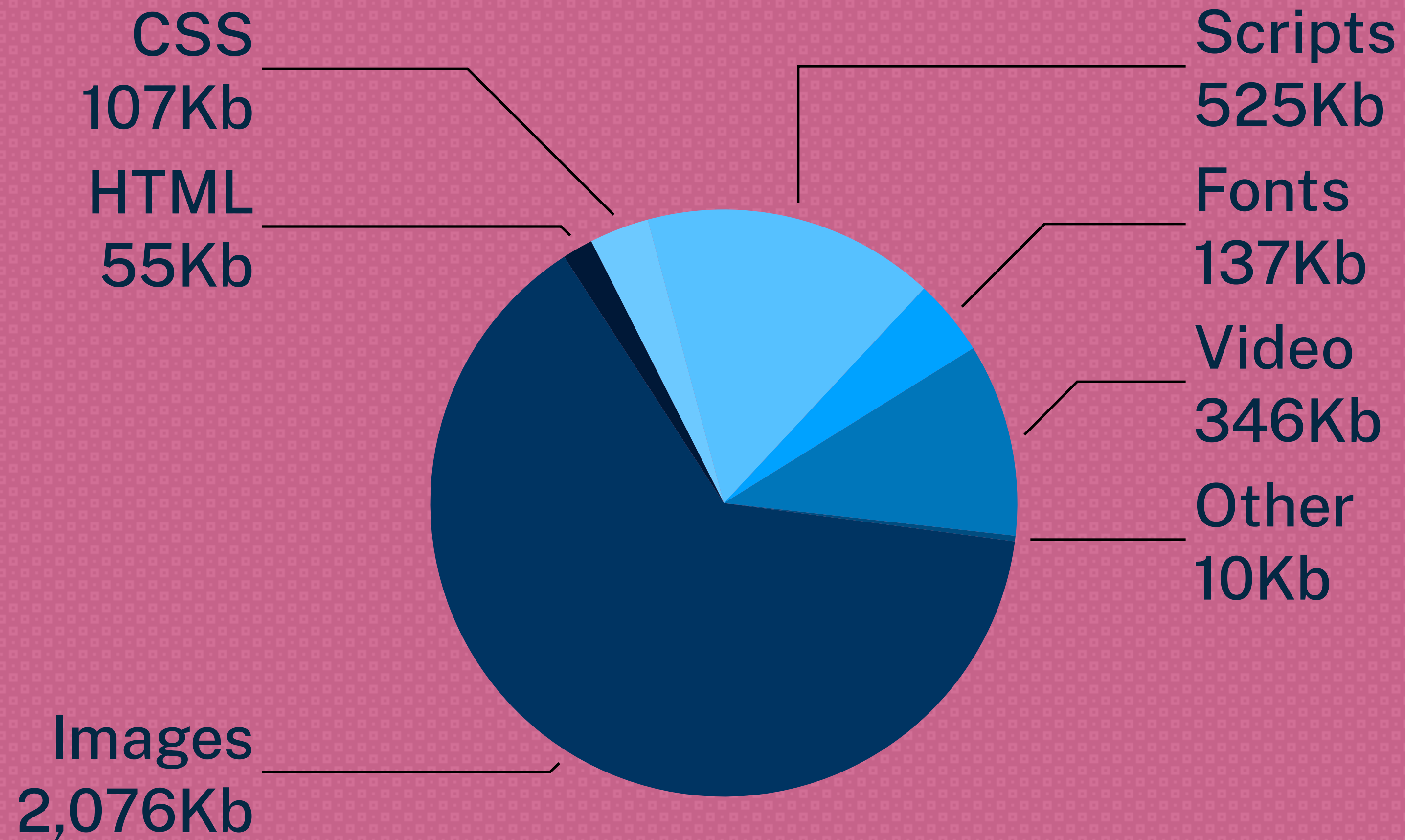
CUSTOMER LOYALTY

\$\$\$ REWARD \$\$\$

LOYAL USER == REPEAT VISITOR

FOR ALL EXTRA PERFORMANCE BENEFITS

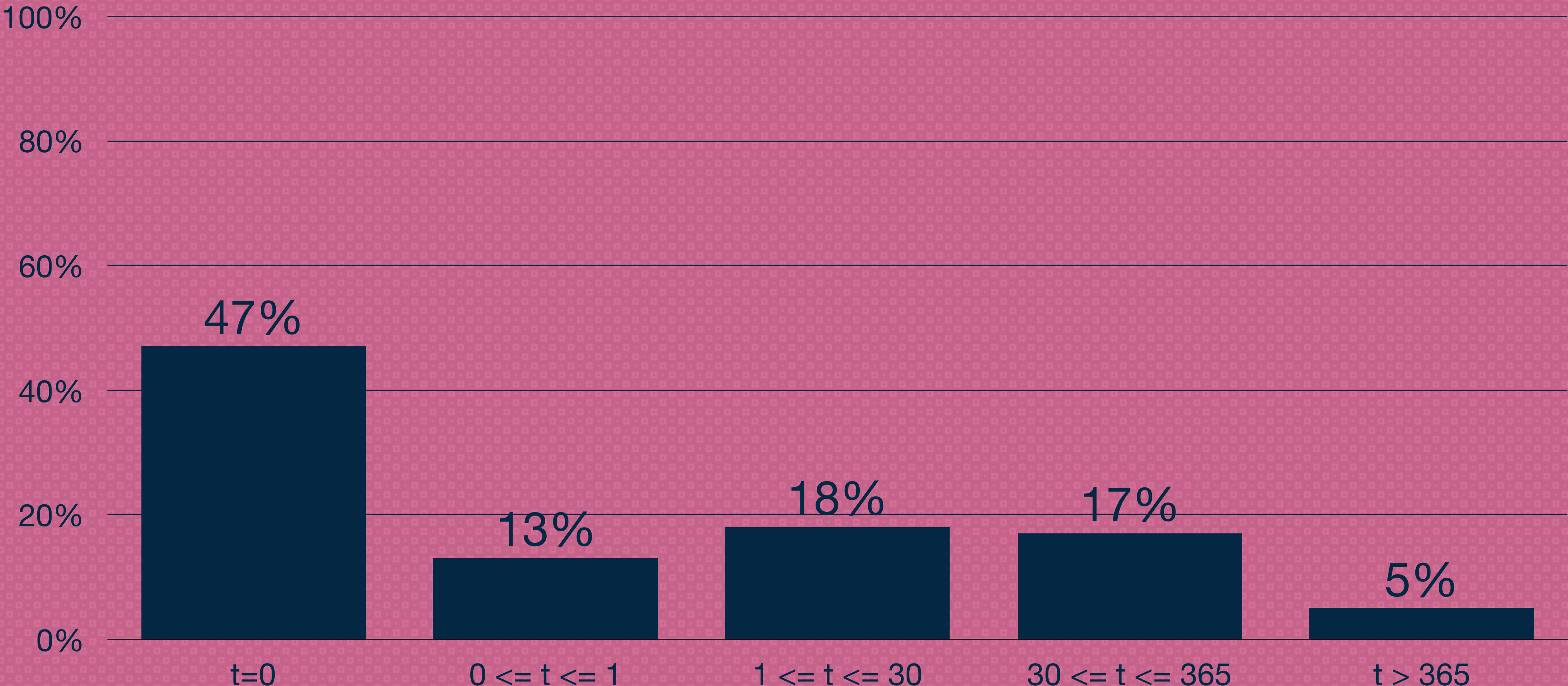
Average Bytes per Page by Content Type



Total
3272Kb

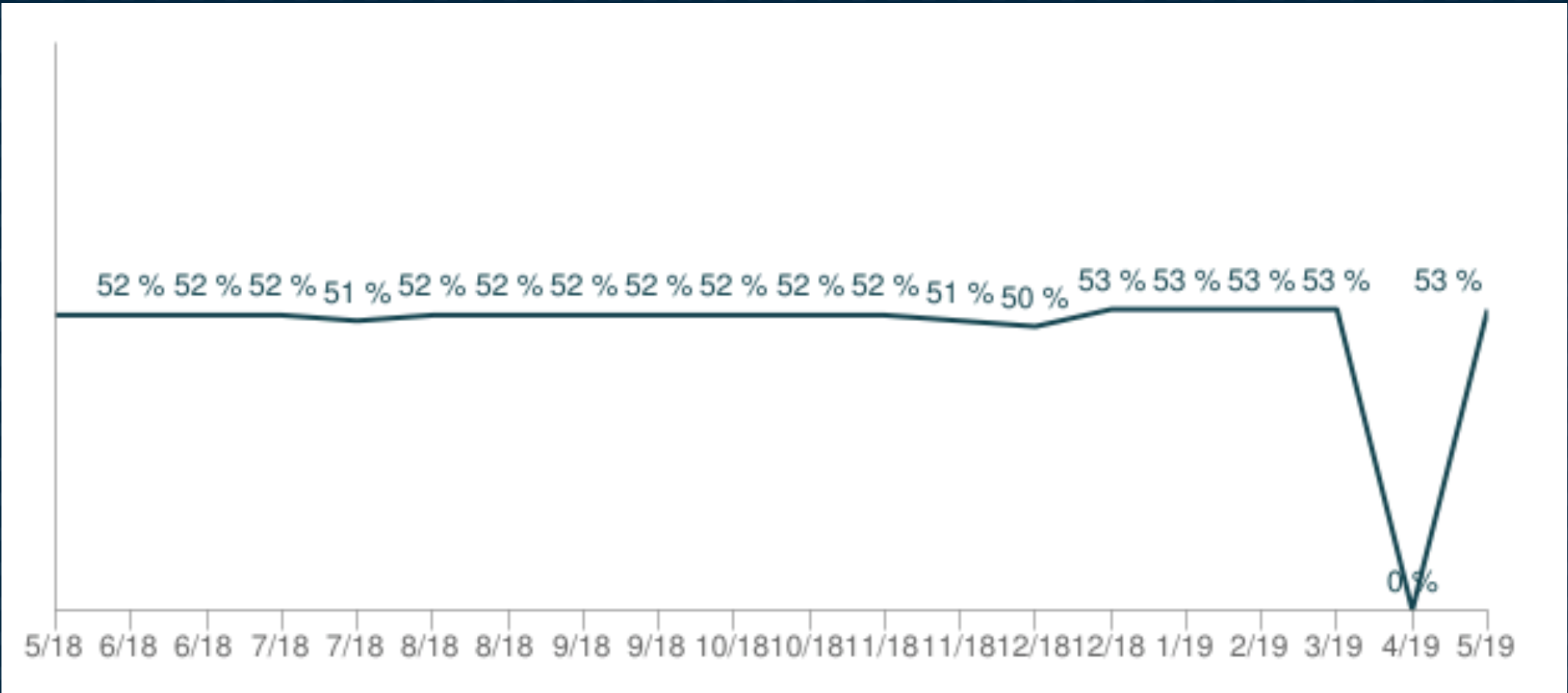


Cache Lifetime (Days)

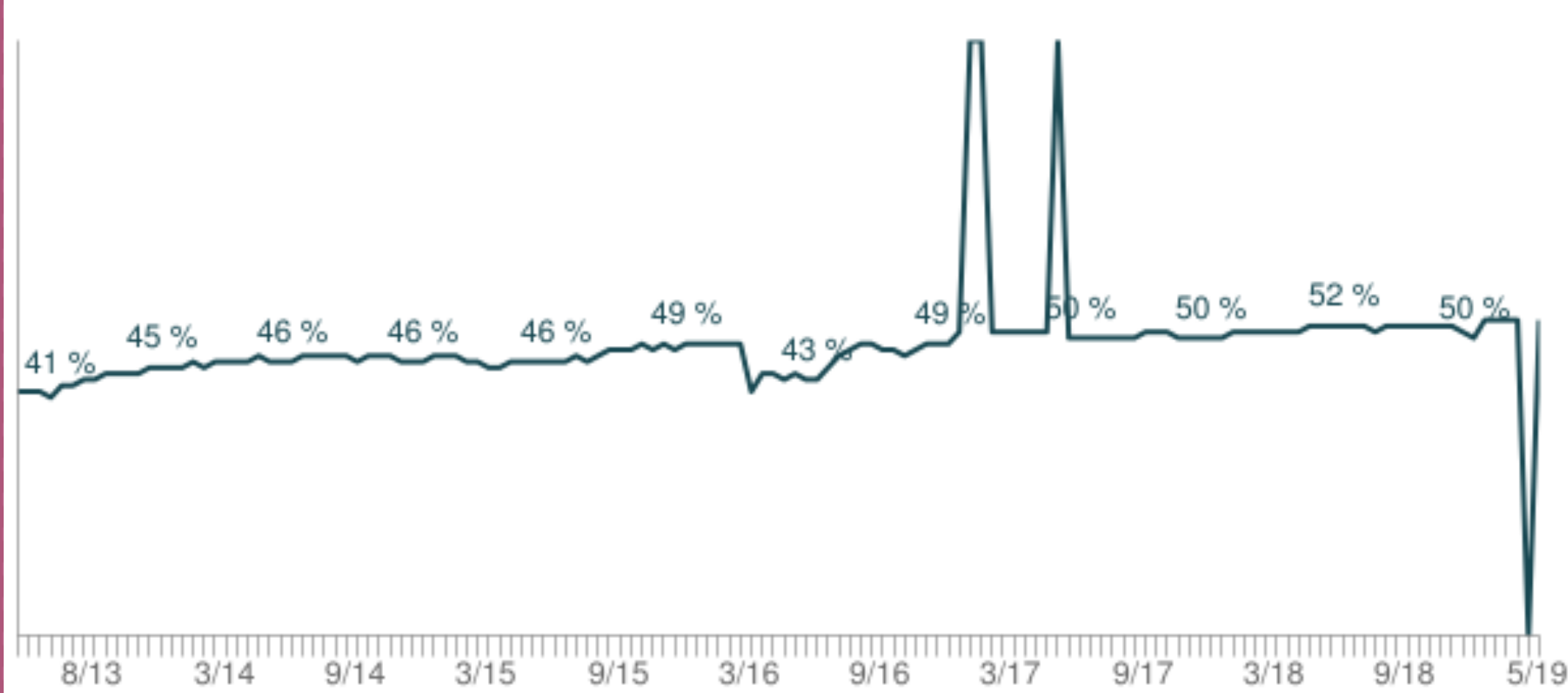


Cacheable Resources

Over Last Year



Over Last 6 Years



~50%

of all traffic could be cached!

Global Internet Traffic

- ❖ **3.138.420 GB per Minute in 2018**
- ❖ **Annual global IP traffic will reach 4.8 ZB per year by 2022**

ZB (Zettabyte)

1 ZB Zettabyte = 1.099.511.627.776 GB

Responsible Design


A LIST APART

NEW! MAY 09, 2019

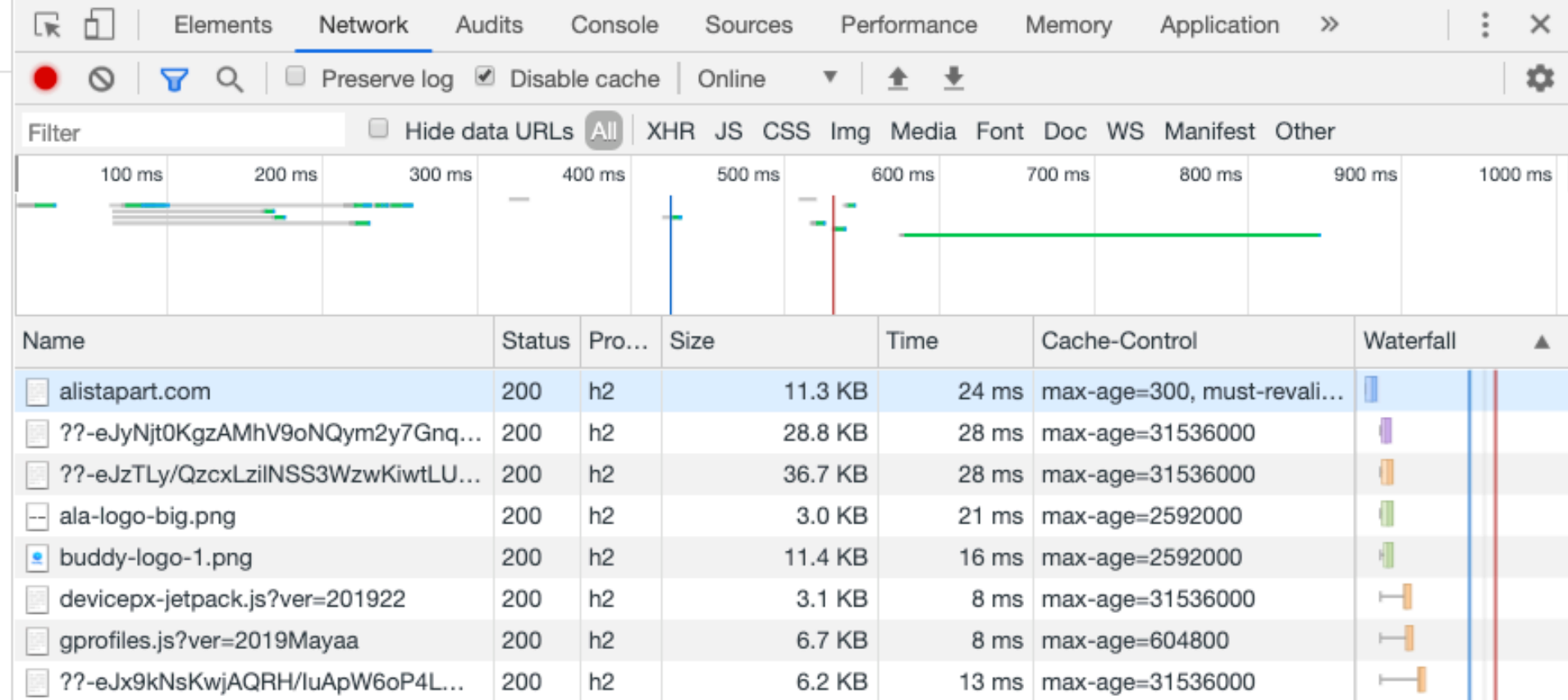
Trans-inclusive Design

by [Erin White](#)

Design decisions across our projects can mean the difference between affirmation and invalidation—and sometimes safety and danger. Erin White explores the repercussions for



Buddy
Get the fastest deployments in UI/UX, with speed that blows developers away. From static websites to WordPress to Kubernetes—whatever you do, Buddy's got your back.



Name	Status	Pro...	Size	Time	Cache-Control	Waterfall
alistapart.com	200	h2	11.3 KB	24 ms	max-age=300, must-revali...	
??-eJyNjt0KgzAMhV9oNQym2y7Gnq...	200	h2	28.8 KB	28 ms	max-age=31536000	
??-eJzTLy/QzcxLzliNSS3WzwKiwtLU...	200	h2	36.7 KB	28 ms	max-age=31536000	
ala-logo-big.png	200	h2	3.0 KB	21 ms	max-age=2592000	
buddy-logo-1.png	200	h2	11.4 KB	16 ms	max-age=2592000	
devicepx-jetpack.js?ver=201922	200	h2	3.1 KB	8 ms	max-age=31536000	
gprofiles.js?ver=2019Mayaa	200	h2	6.7 KB	8 ms	max-age=604800	
??-eJx9kNsKwjAQRH/luApW6oP4L...	200	h2	6.2 KB	13 ms	max-age=31536000	

32 requests | 276 KB transferred | 611 KB resources | Finish: 843 ms | **DOMContentLoaded: 422 ms** | **Load: 528 ms**

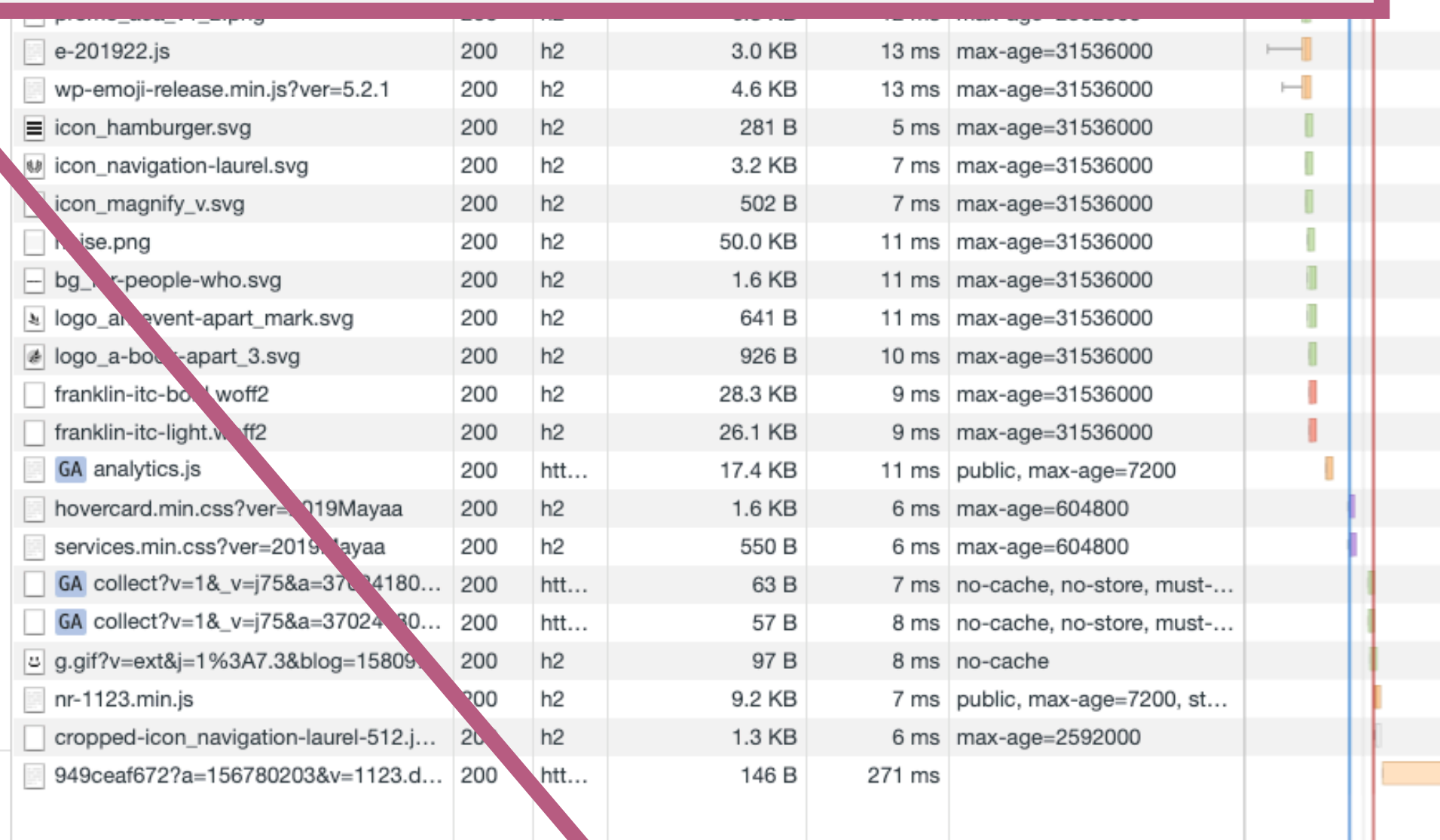
OCTOBER 25, 2018

From URL to Interactive

by [Aaron Gustafson](#)

When we think about it, our whole industry depends on our faith in a handful of “black boxes” few of us fully understand: browsers. We hand over our HTML, CSS, JavaScript, cross our fingers, and hope they render the experience we have in our heads. But knowing how they work can really get you out of a jam when things go wrong. That’s why we’ve assembled a handful of incredibly knowledgeable authors to take us under the hood in this four-part series. Join us on this trip across the web, into the often foggy valley between code and experience.

More from A List Apart



e-201922.js	200	h2	3.0 KB	13 ms	max-age=31536000	
wp-emoji-release.min.js?ver=5.2.1	200	h2	4.6 KB	13 ms	max-age=31536000	
icon_hamburger.svg	200	h2	281 B	5 ms	max-age=31536000	
icon_navigation-laurel.svg	200	h2	3.2 KB	7 ms	max-age=31536000	
icon_magnify_v.svg	200	h2	502 B	7 ms	max-age=31536000	
ise.png	200	h2	50.0 KB	11 ms	max-age=31536000	
bg_...-people-who.svg	200	h2	1.6 KB	11 ms	max-age=31536000	
logo_a-event-apart_mark.svg	200	h2	641 B	11 ms	max-age=31536000	
logo_a-board-apart_3.svg	200	h2	926 B	10 ms	max-age=31536000	
franklin-itc-bo...woff2	200	h2	28.3 KB	9 ms	max-age=31536000	
franklin-itc-light.woff2	200	h2	26.1 KB	9 ms	max-age=31536000	
GA analytics.js	200	htt...	17.4 KB	11 ms	public, max-age=7200	
hovercard.min.css?ver=2019Mayaa	200	h2	1.6 KB	6 ms	max-age=604800	
services.min.css?ver=2019Mayaa	200	h2	550 B	6 ms	max-age=604800	
GA collect?v=1&_v=j75&a=37024180...	200	htt...	63 B	7 ms	no-cache, no-store, must-...	
GA collect?v=1&_v=j75&a=37024180...	200	htt...	57 B	8 ms	no-cache, no-store, must-...	
g.gif?v=ext&j=1%3A7.3&blog=15809...	200	h2	97 B	8 ms	no-cache	
nr-1123.min.js	200	h2	9.2 KB	7 ms	public, max-age=7200, st...	
cropped-icon_navigation-laurel-512.j...	200	h2	1.3 KB	6 ms	max-age=2592000	
949ceaf672?a=156780203&v=1123.d...	200	htt...	146 B	271 ms		

32 requests | 276 KB transferred | 611 KB resources | Finish: 843 ms | **DOMContentLoaded: 422 ms** | **Load: 528 ms**

Trans-inclusive Design
by [Erin White](#)

Design decisions across our projects can mean the difference between affirmation and invalidation—and sometimes safety and danger. Erin White explores the repercussions for

From URL to Interactive
by [Aaron Gustafson](#)

When we think about it, our whole industry depends on our faith in a handful of “black boxes” few of us fully understand: browsers. We hand over our HTML, CSS, JavaScript, cross our fingers, and hope they render the experience we have in our heads. But knowing how they work can really get you out of a jam when things go wrong. That’s why we’ve assembled a handful of incredibly knowledgeable authors to take us under the hood in this four-part series. Join us on this trip across the web, into the often foggy valley between code and experience.

More from A List Apart

Performance Summary:
32 requests | 12.9 KB transferred | 611 KB resources | Finish: 604 ms | **DOMContentLoaded: 288 ms** | **Load: 382 ms**

Name	Status	Pro...	Size	Time	Cache-Control	Waterfall
alistapart.com	200	h2	11.3 KB	13 ms	max-age=300, must-revali...	
??-eJzTLy/QzcxLziINSS3WzwKiwTLU...	200	h2	(memory cache)	0 ms	max-age=31536000	
ala-logo-big.png	200	h2	(memory cache)	0 ms	max-age=2592000	
buddy-logo-1.png	200	h2	(memory cache)	0 ms	max-age=2592000	
logo_a-book-apart-color.png	200	h2	(memory cache)	0 ms	max-age=2592000	
AEA_2017_80x80_2.png	200	h2	(memory cache)	0 ms	max-age=2592000	
promo_aba_2.jpg	200	h2	(memory cache)	0 ms	max-age=2592000	
promo_aea_v1_2.png	200	h2	(memory cache)	0 ms	max-age=2592000	
e-201922.js	200	h2	(disk cache)	18 ms	max-age=31536000	
wp-emoji-release.min.js?ver=5.2.1	200	h2	(disk cache)	7 ms	max-age=31536000	
franklin-itc-bold.woff2	200	h2	(memory cache)	7 ms	max-age=31536000	
franklin-itc-light.woff2	200	h2	(memory cache)	0 ms	max-age=31536000	
icon_hamburger.svg	200	h2	(memory cache)	0 ms	max-age=31536000	
icon_navigation-laurel.svg	200	h2	(memory cache)	0 ms	max-age=31536000	
icon_magnify_v.svg	200	h2	(memory cache)	7 ms	max-age=31536000	
noise.png	200	h2	(memory cache)	1 ms	max-age=31536000	
bg_for-people-who.svg	200	h2	(memory cache)	1 ms	max-age=31536000	
logo_an-event-apart_mark.svg	200	h2	(memory cache)	0 ms	max-age=31536000	
logo_a-book-apart-3.svg	200	h2	(memory cache)	0 ms	max-age=31536000	
GA_analytics.js	200	htt...	(disk cache)	1 ms	public, max-age=7200	
hovercard.min.css?ver=2019Mayaa	200	h2	(disk cache)	1 ms	max-age=604800	
services.min.css?ver=2019Mayaa	200	h2	(disk cache)	1 ms	max-age=604800	
g.gif?v=ext&j=1%3A7.3&blog=5809...	200	h2	97 B	4 ms	no-cache	
GA_collect?v=1&_v=j75&a=1230628...	200	htt...	64 B	16 ms	no-cache, no-store, must-...	
GA_collect?v=1&_v=j75&a=1230682...	200	htt...	57 B	7 ms	no-cache, no-store, must-...	
nr-1123.min.js	200	h2	(disk cache)	2 ms	public, max-age=7200, st...	
cropped-icon_navigation-laurel-512.j...	200	h2	1.2 KB	7 ms	max-age=2592000	
949ceaf672?a=156780203&v=1123.d...	200	htt...	146 B	201 ms		

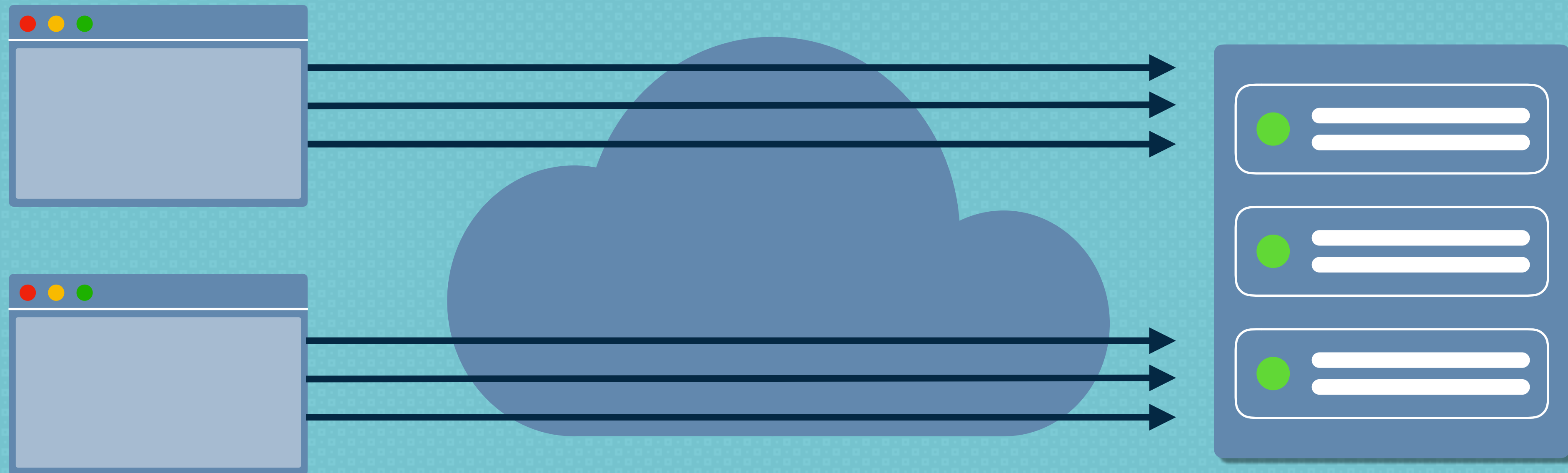
Advantages

- ❖ Eases the load on the server;
- ❖ Improves performance, b/c it takes less time to transmit the resource.



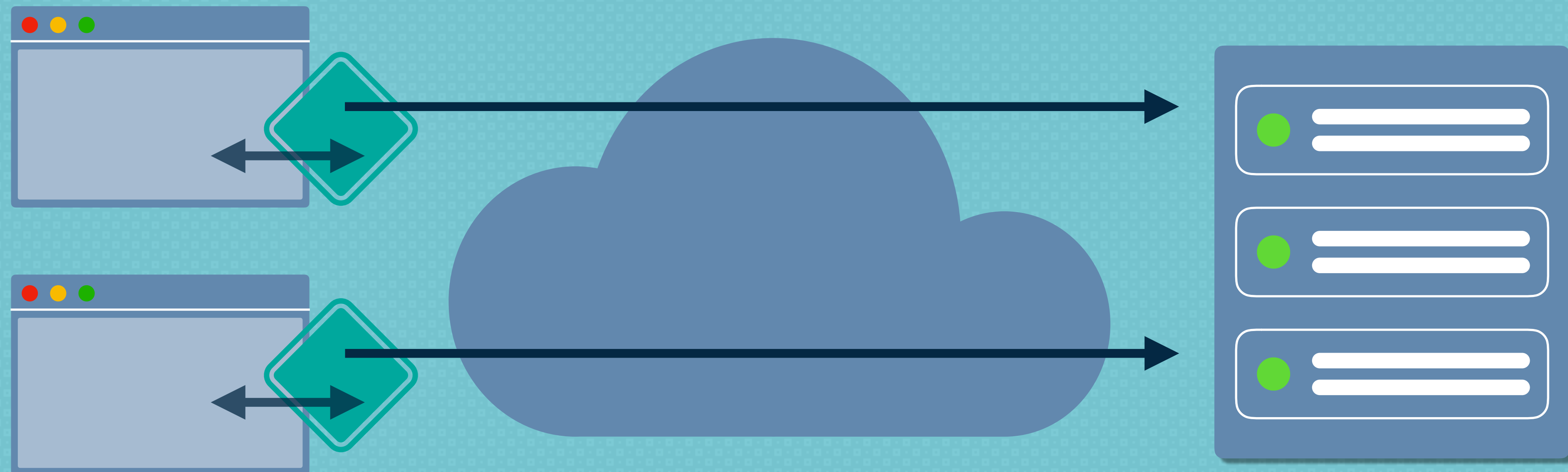
How Does Caching Work?

Without Any Caching in Place



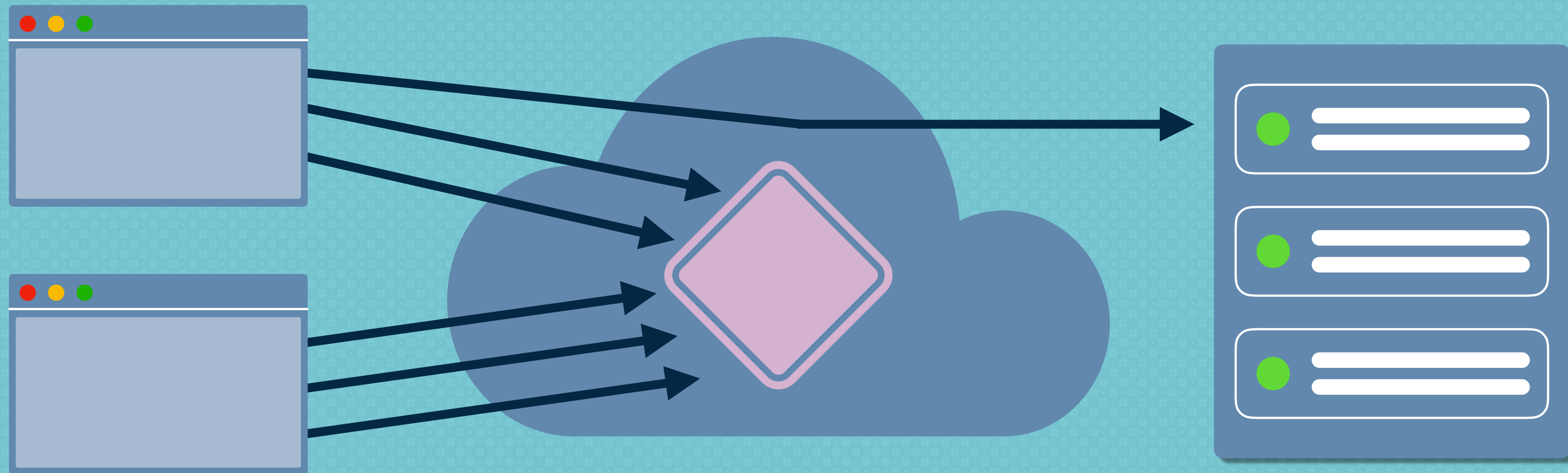
All identical requests are going through to the server

Local (Private) Cache



The first request of each client is going through to the server.
Identical, subsequent requests are not even sent, but served by the local cache.

Shared Cache



The first request of each client is going through to the server. Identical, subsequent requests are served by the shared cache.

Private Cache

A private cache is dedicated to a single user.

Shared Cache

A shared cache can be reused by more than one user.

Empty Cache

The browser makes the first request to the page.

Primed Cache

The browser has a cached version of the page.

Cache Hit

A requested resource is found in cache.

Cache Miss

A resource is not found in cache and has to be fully requested from its origin.

Fresh

When a resource is fresh, it can be served from cache.

Stale

A stale resource has expired and will not be served anymore.

Disk Cache

Lifetime: Restart

Disk cache is RAM
memory

Fast!

Memory Cache

Lifetime: Render
Process, Browser Tab

Located on the CPU chip

Faster!

200	h2	(memory cache)	0 ms	public, max-age=3153600...
200	h2	(memory cache)	0 ms	public, max-age=3153600...
200	h2	(memory cache)	0 ms	public, max-age=3153600...
200	h2	(memory cache)	0 ms	public, max-age=3153600...
200	h2	(disk cache)	5 ms	public, max-age=3153600...
200	h2	7.6 KB	249 ms	public, max-age=3153600...

It's **TIME**

To Take Control

History of HTTP Protocol

- ❖ HTTP/0.9 – The one-line protocol
- ❖ HTTP/1.0 – Building extensibility
- ❖ HTTP/1.1 – The standardized protocol

HTTP headers

 Languages Edit

Jump to: [Authentication](#) [Caching](#) [Client hints](#) [Conditionals](#) [Connection management](#) [Content negotiation](#) [Controls](#) [Cookies](#) [CORS](#) [Do Not Track](#)

[Downloads](#) [Message body information](#) [Proxies](#) [Redirects](#) [Request context](#) [Response context](#) [Range requests](#) [Security](#) [Server-sent events](#) [Transfer coding](#)

[WebSockets](#) [Other](#) [Contributing](#) [See also](#)

[Web technology for developers](#) > [HTTP](#) >

HTTP headers

Related Topics

HTTP

Guides:

- ▶ [Resources and URIs](#)
- ▶ [HTTP guide](#)
- ▶ [HTTP security](#)

[HTTP access control \(CORS\)](#)

[HTTP authentication](#)

[HTTP caching](#)

[HTTP compression](#)

[HTTP conditional requests](#)

[HTTP content negotiation](#)

[HTTP cookies](#)

[HTTP range requests](#)

[HTTP redirects](#)

[HTTP specifications](#)

[Feature policy](#)

References:

- ▼ [HTTP headers](#)

HTTP headers allow the client and the server to pass additional information with the request or the response. An HTTP header consists of its case-insensitive name followed by a colon ':', then by its value (without line breaks). Leading white space before the value is ignored.

Custom proprietary headers can be added using the 'X-' prefix, but this convention was deprecated in June 2012, because of the inconveniences it caused when non-standard fields became standard in [RFC 6648](#); others are listed in an [IANA registry](#), whose original content was defined in [RFC 4229](#). IANA also maintains a [registry of proposed new HTTP message headers](#).

Headers can be grouped according to their contexts:

- [General header](#): Headers applying to both requests and responses but with no relation to the data eventually transmitted in the body.
- [Request header](#): Headers containing more information about the resource to be fetched or about the client itself.
- [Response header](#): Headers with additional information about the response, like its location or about the server itself (name and version etc.).
- [Entity header](#): Headers containing more information about the body of the entity, like its content length or its MIME-type.

Headers can also be grouped according to how proxies handle them:

End-to-end headers

These headers must be transmitted to the final recipient of the message; that is, the server for a request or the client for a response. Intermediate proxies must retransmit end-to-end headers unmodified and caches must store them.

Hop-by-hop headers

These headers are meaningful only for a single transport-level connection and must not be retransmitted by proxies or cached. Such headers are: [Connection](#), [Keep-Alive](#), [Proxy-Authenticate](#), [Proxy-Authorization](#), [TE](#), [Trailer](#), [Transfer-Encoding](#), and [Upgrade](#). Note that only hop-by-hop headers may be set using the

Caching [↗](#)

Age

The time in seconds the object has been in a proxy cache.

Cache-Control

Specifies directives for caching mechanisms in both requests and responses.

Clear-Site-Data

Clears browsing data (e.g. cookies, storage, cache) associated with the requesting website.

Expires

The date/time after which the response is considered stale.

Pragma

Implementation-specific header that may have various effects anywhere along the request-response chain. Used for backwards compatibility with HTTP/1.0 caches where the `Cache-Control` header is not yet present.

Warning

A general warning field containing information about possible problems.

y

y-Report-Only

Policy

Chrome DevTools

The screenshot displays the Chrome DevTools Network tab. The top navigation bar includes tabs for Elements, Network, Audits, Console, Sources, Performance, and Memory. The Network tab is active, showing a list of requests. The selected request is for 'main.min.css' from 'foobartel.com'. The right-hand pane shows the details for this request, including the Request URL, Request Method (GET), Status Code (200), Remote Address, and Referrer Policy. Below this, the Response Headers and Request Headers are visible.

Name

- foobartel.com
- main.min.css**
- bg35.jpg

General

- Request URL:** https://foobartel.com/assets/css/main.min.css
- Request Method:** GET
- Status Code:** 200
- Remote Address:** 85.13.140.158:443
- Referrer Policy:** no-referrer-when-downgrade

Response Headers

- accept-ranges:** bytes
- cache-control:** max-age=2628000, public
- content-encoding:** gzip
- content-length:** 2193
- content-type:** text/css
- date:** Thu, 23 May 2019 06:17:43 GMT
- etag:** "1ea1-587a7be817220-gzip"
- last-modified:** Mon, 29 Apr 2019 09:16:48 GMT
- server:** Apache
- status:** 200
- vary:** Accept-Encoding

Request Headers

- Provisional headers are shown**
- DNT:** 1
- Referer:** https://foobartel.com/
- Sec-Fetch-Mode:** no-cors
- User-Agent:** Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3799.0 Safari/537.36

3 requests | 58.8 KB transferred | 66.6

Headers

Pragma: no-cache

Pragma is a HTTP/1.0 header and not a reliable replacement for the general HTTP/1.1 Cache-Control header.

Expires: Fri, 30 Oct 2019 14:19:41 GMT

The Expires header contains the date/time after which the response is considered stale.

If there is a **Cache-Control** header with the **max-age** or **s-maxage** directive in the response, the **Expires** header is ignored.

The **Cache-Control** HTTP/1.1 general-header field is used to specify directives for caching mechanisms in both requests and responses.

Cache-Control Directives

```
cache-control: public
```

The response may be cached by any cache, even if the response would normally be non-cacheable.

cache-control: private

The response is intended for a single user and must not be stored by a shared cache.

A private cache may store the response.

```
cache-control: max-age=86400
```

Response can be cached by browser and any intermediary caches ("public") for up to 1 day.

```
cache-control: private, max-age=600
```

Response can be cached by the client's browser only for up to 10 minutes (60 seconds x 10 minutes).

cache-control: no-cache

Forces the cache to validate the request before releasing a cached copy.

cache-control: no-store

- ❖ Response is not allowed to be cached
- ❖ Must be fetched in full on every request.

cache-control: max-age=0

A max-age of 0 equals no-store.

cache-control: s-maxage=<seconds>

- ❖ Takes precedence over **max-age** or the **Expires** header, but only applies to shared caches (e.g., proxies);
- ❖ Ignored by a private cache.

cache-control: no-transform

No transformations or conversions should be made to the resource. The Content-Encoding, Content-Range, Content-Type headers must not be modified by a proxy.

cache-control: must-revalidate

If a resource has become stale, a cache must not use the response without successful validation.

cache-control: immutable

Indicates that the response body will not change over time.

Build Your Caching Strategy And An Invalidation Strategy

Cache Invalidation Terms

- ❖ Hashing
- ❖ Cachebusting
- ❖ Revving

Cachebusting Options

- ❖ `/assets/css/styles.css?v1.2`
- ❖ `/assets/16415ef98913c2bcb2c/default.css`
- ❖ `/assets/js/main.16415ef9891.js`

Apache - .htaccess

```
Header set Cache-Control "max-age=31536000, public"
```


Apache - .htaccess

```
<filesMatch ".(jpg|jpeg|png|gif|ico)$">  
  Header set Cache-Control "max-age=31536000, public,  
  immutable"  
</filesMatch>
```

```
<filesMatch ".(css|js)$">  
  Header set Cache-Control "max-age=31536000, public"  
</filesMatch>
```

PHP header()

```
<?php
```

```
header("Cache-Control: no-cache, must-revalidate");
```

```
header("Cache-Control: Sat, 26 Jul 1997 05:00:00 GMT");
```

```
?>
```

Nginx - configuration file

```
location ~* \.(js|css|png|jpg|jpeg|gif|ico)$ {  
  
    add_header Cache-Control "public, no-transform";  
  
}
```

Cache

As Much As You Can

```
cache-control: public, max-age=31536000
```

- ✦ For files that won't change, you can add aggressive caching;
- ✦ Likely for images, CSS and JavaScript.

`cache-control: public, immutable`

- ❖ More aggressive;
- ❖ CSS and JavaScript;
- ❖ Don't forget your invalidation strategy!

```
<FilesMatch “^(never-changing-icon.svg)$”>
```

```
Header set Cache-Control "max-age=31536000, public,  
immutable"
```

```
</FilesMatch>
```

Summary

- ❖ We can limit traffic and spare sending lots of bytes
- ❖ Reward your loyal repeat visitors
- ❖ Improve overall web performance

- ❖ Create your invalidation/cachebusting strategy
- ❖ Ensure that the server provides an ETag
- ❖ Which resources can be cached publicly or private?
- ❖ Determine the best cache lifetime for your resources
- ❖ How aggressively can you cache your assets?

Thank You!

Visit foobartel.com or Twitter: [@foobartel](https://twitter.com/foobartel)