

# Dynamic Web Optimization Works!

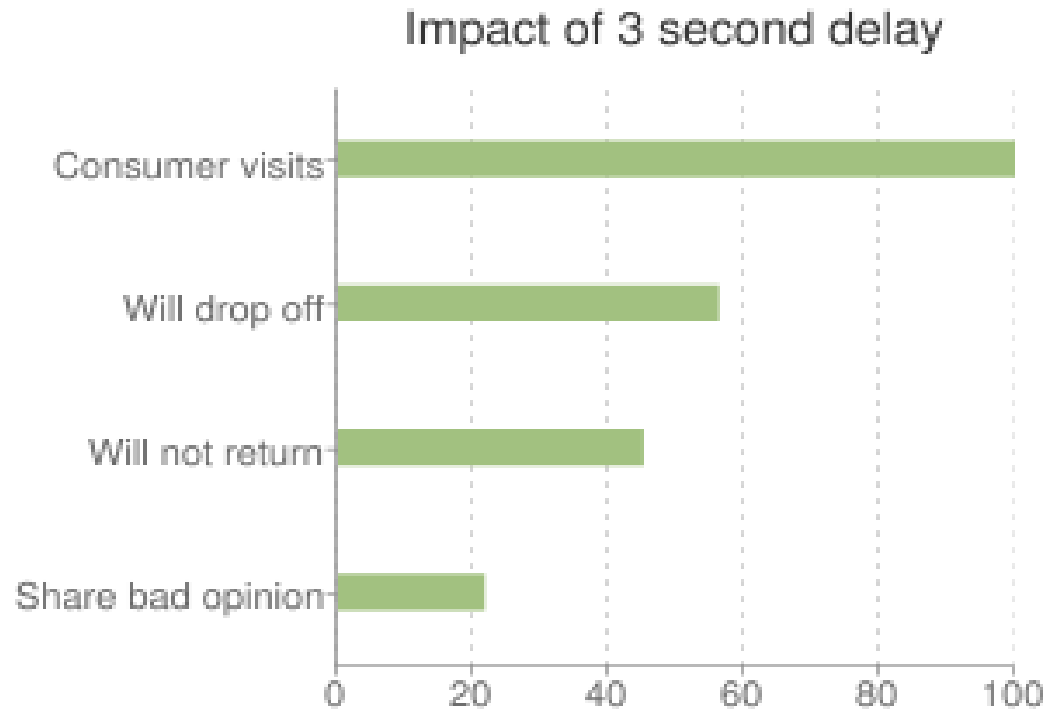
Experiences with mod\_pagespeed and Pagespeed Automatic

**Jan-Willem Maessen**  
Software Engineer,  
Pagespeed Automatic  
jmaessen@google.com

# Impact of Load Time on Effectiveness



- Amazon.com: -100 ms latency -> +1% revenue<sup>+</sup>
- Edmunds.com: -50% load time -> +17% page views per session<sup>++</sup>
- Stats<sup>+++</sup>



<sup>+</sup> Source: Mashable

<sup>++</sup> Source: Edmunds.com Technology blog

<sup>+++</sup> Source: Strangeloop Networks

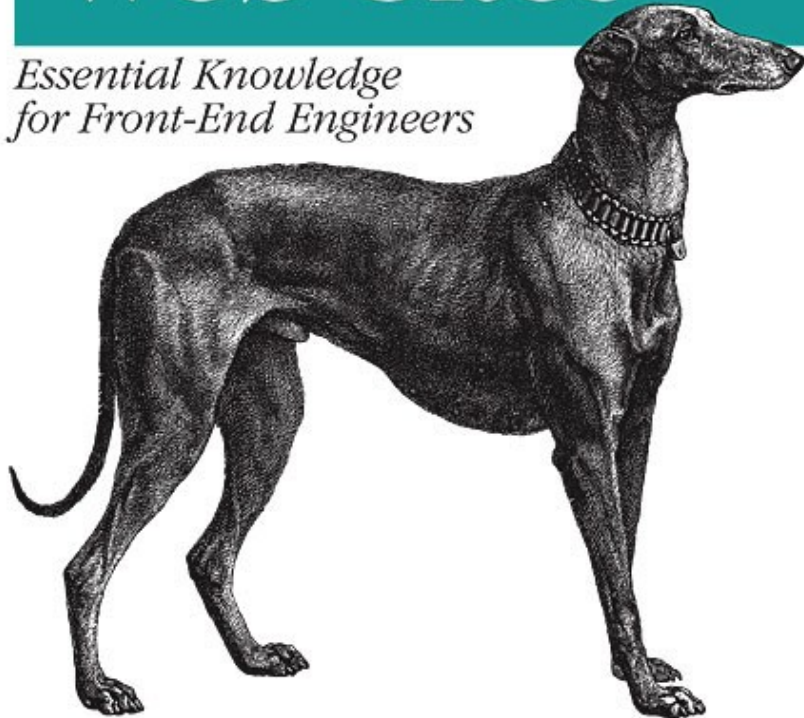
# We Know How to Speed the Web



*14 Steps to Faster-Loading Web Sites*

## High Performance Web Sites

*Essential Knowledge  
for Front-End Engineers*



O'REILLY®

*Steve Souders  
Foreword by Nate Koechley*

*Essential Knowledge for Frontend Engineers*



## Even Faster Web Sites

O'REILLY®

*Steve Souders*

# But It's a Balancing act

## SPEED

Inlined / sprited / minified resources

Long cache lifetimes

Exploit features in modern browsers

Track latest optimization techniques

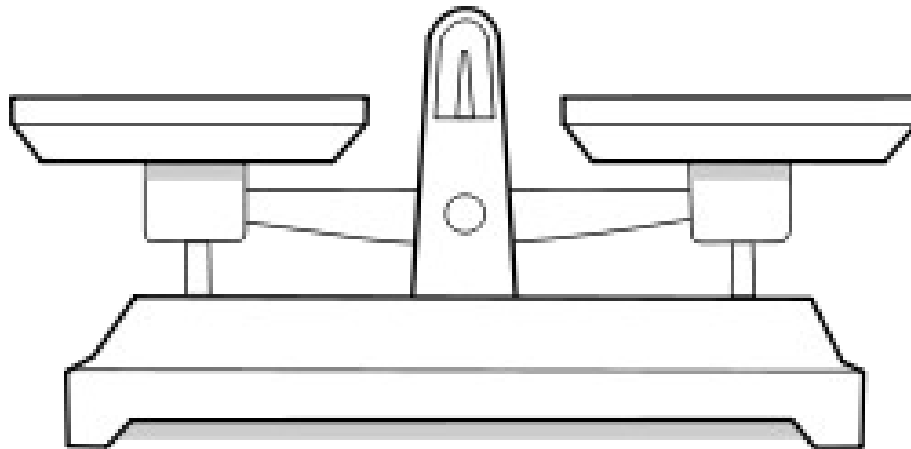
## EASE OF MAINTENANCE

Simple development & deployment flow

Ability to rapidly deploy changes to users

Support all browsers

Focus on content



Solution: Use Automated Tools

# Drop in usage in Apache

---



```
Wget \  
https://dl-ssl.google.com/dl/linux/direct/  
mod-pagespeed-beta_current_amd64.deb
```

```
sudo dpkg -i mod-pagespeed-*.deb
```

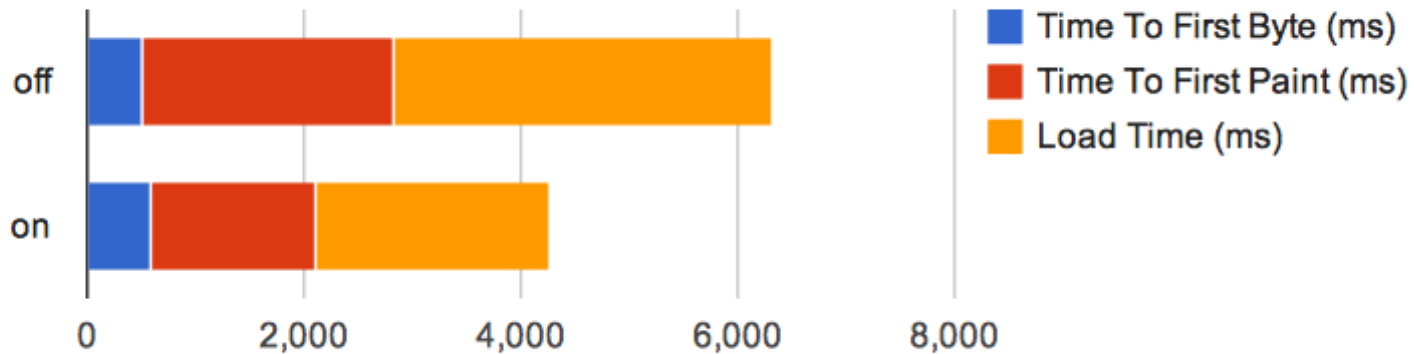
```
sudo apt-get -f install
```

```
sudo /etc/init.d/apache2 restart
```

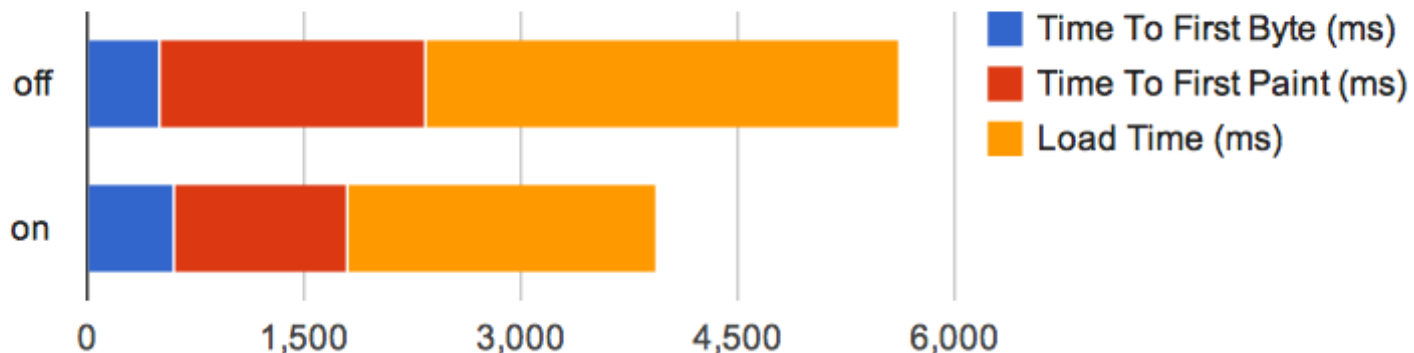
**Must not break the web**

[http://www.youtube.com/watch?feature=player\\_prof lepage&v=1FPUzUPnLR8](http://www.youtube.com/watch?feature=player_prof lepage&v=1FPUzUPnLR8)

### First View Timeline (ms)



### Repeat View Timeline (ms)



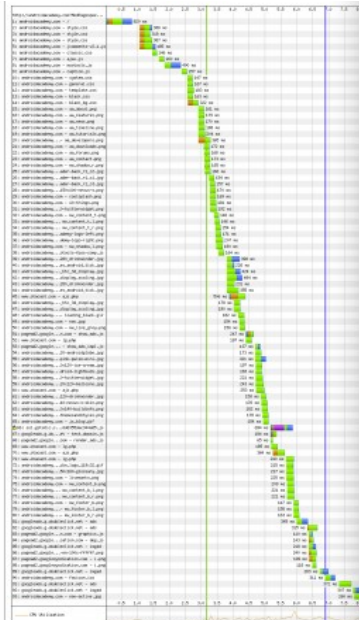
# Webpagetest.org waterfall for Androidacademy.com



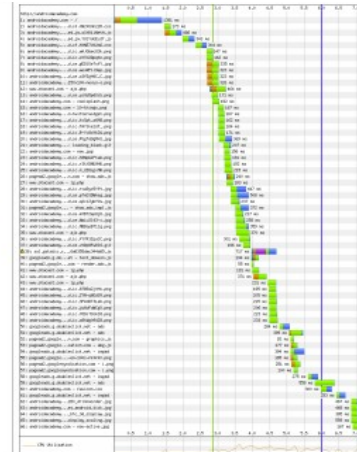
mod\_pagespeed off

mod\_pagespeed on

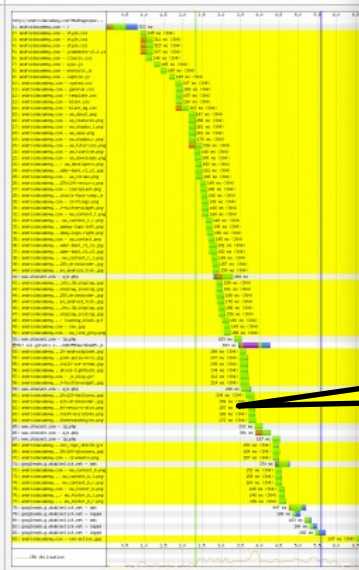
First View  
(6.903s)



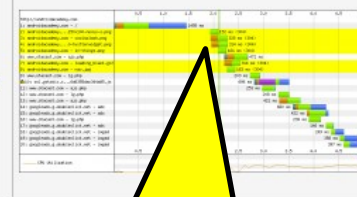
First View  
(5.987s)



Repeat View  
(5.592s)

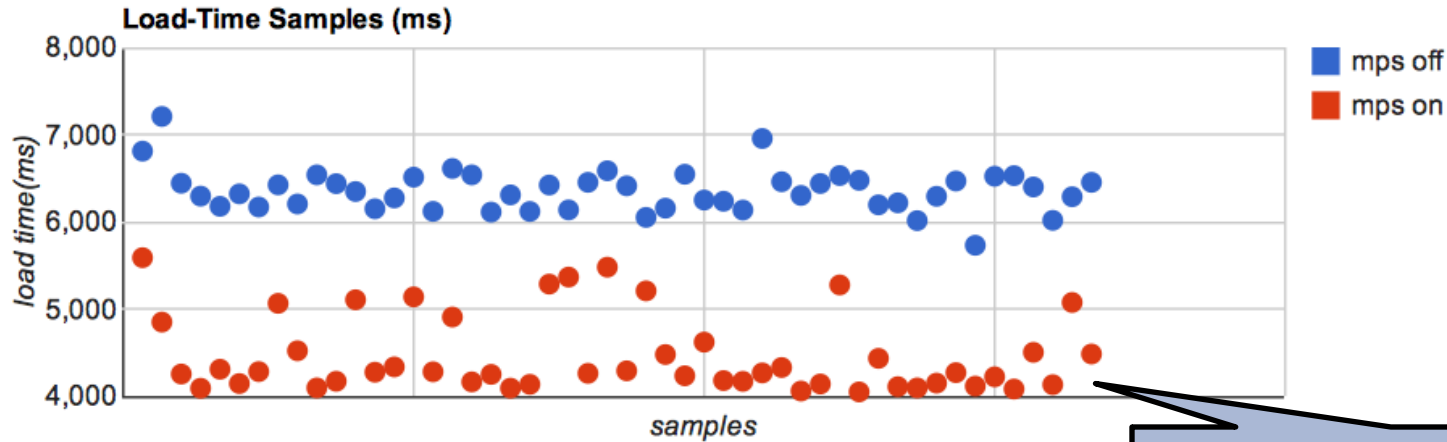


Repeat View  
(5.138s)



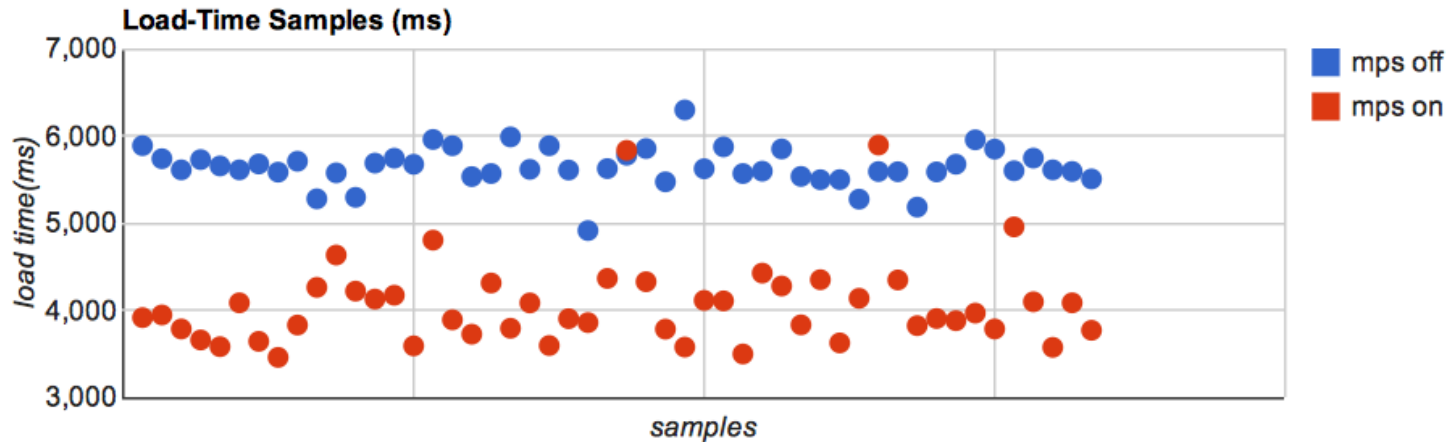
304 (not modified) responses

First View Load Time (ms) over 50 Samples



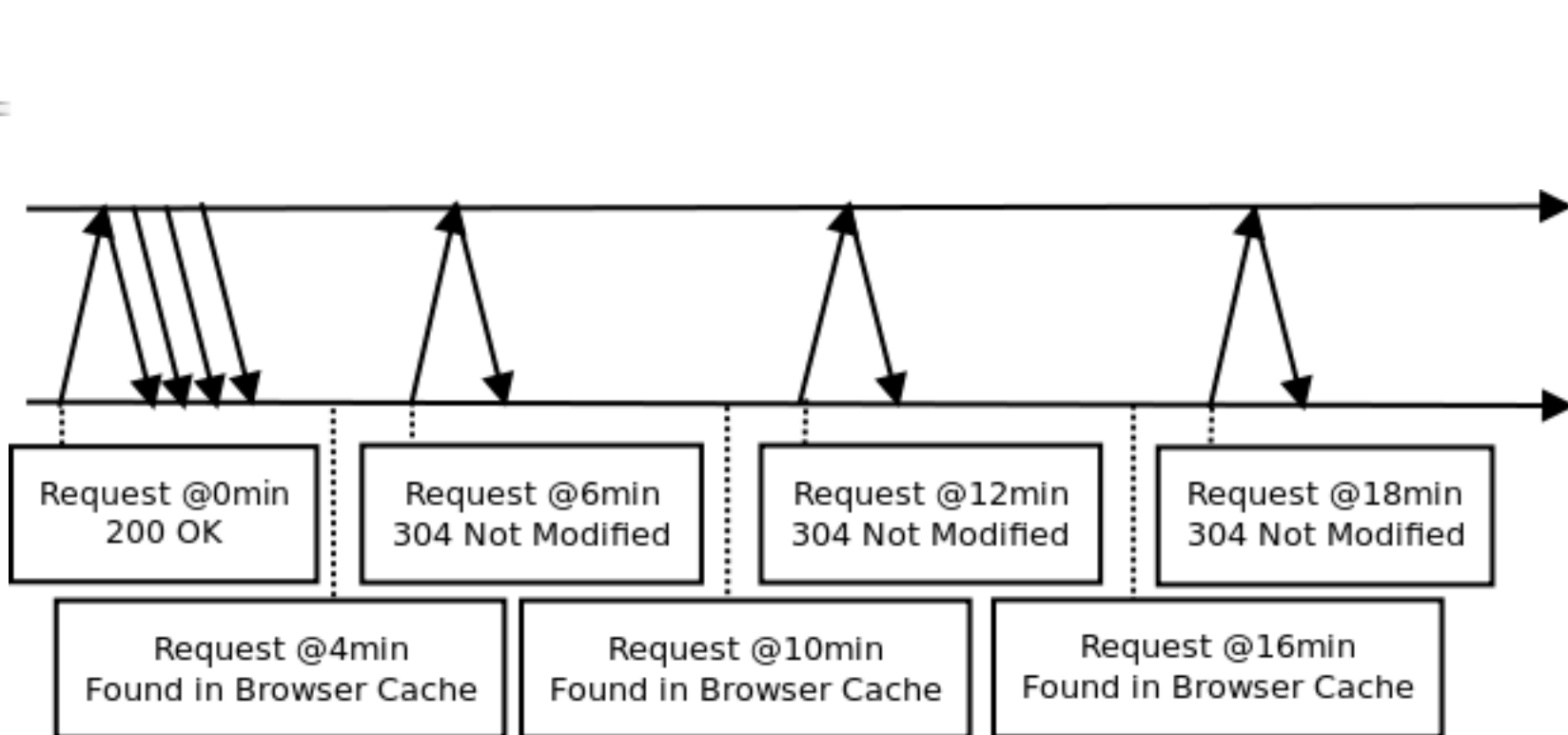
Measurement variance, but conclusive results

Repeat View Load Time (ms) over 50 Samples

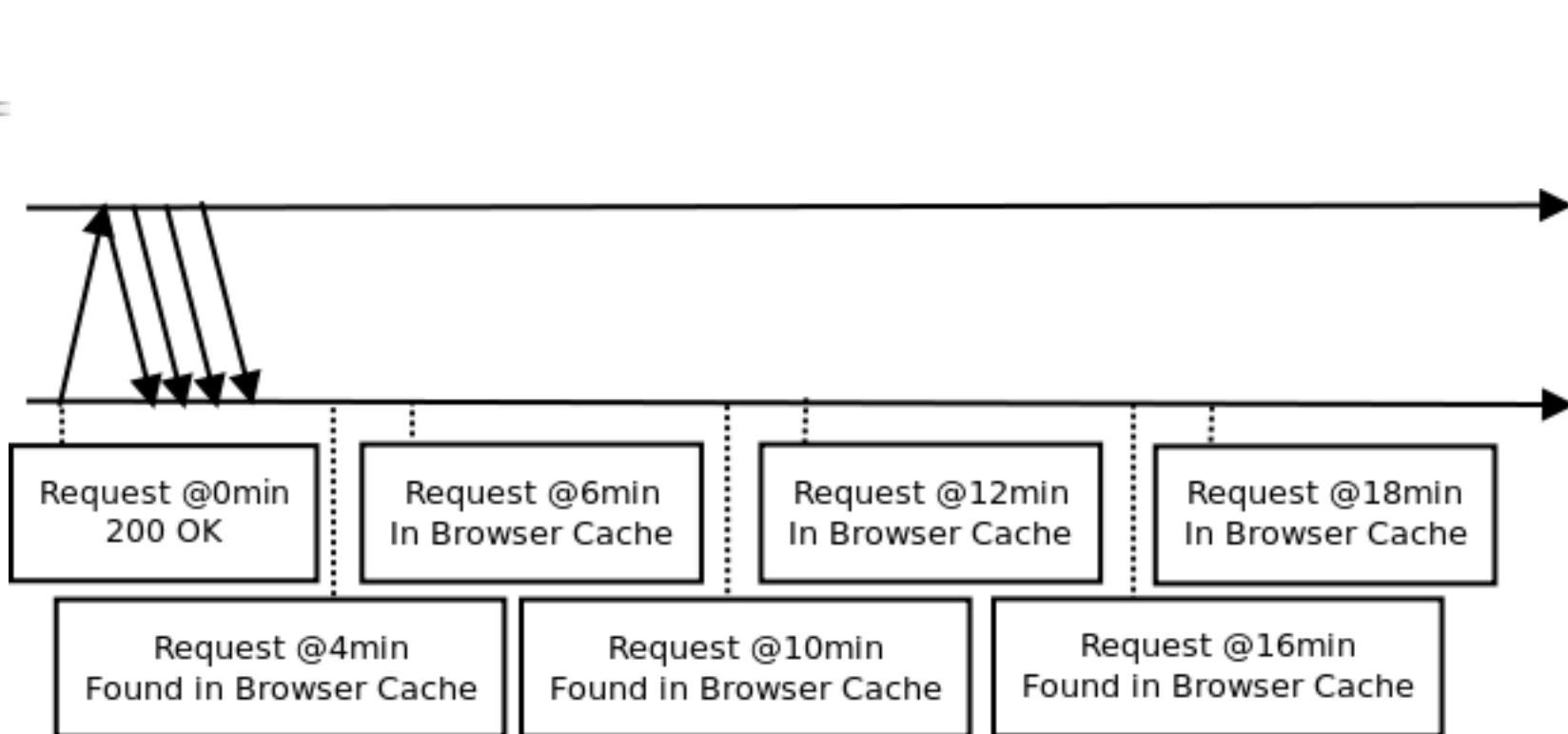




# extend\_cache in action

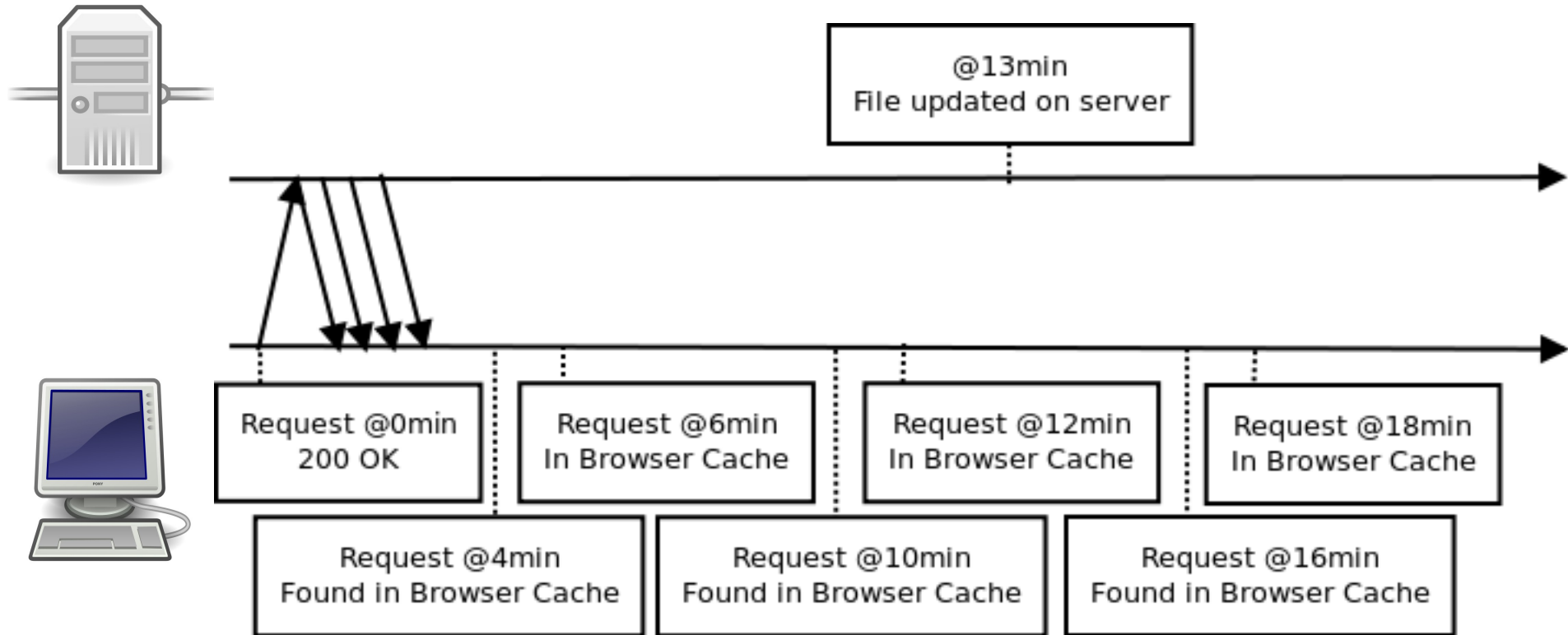


Normal behavior with Cache-Control: 5 min



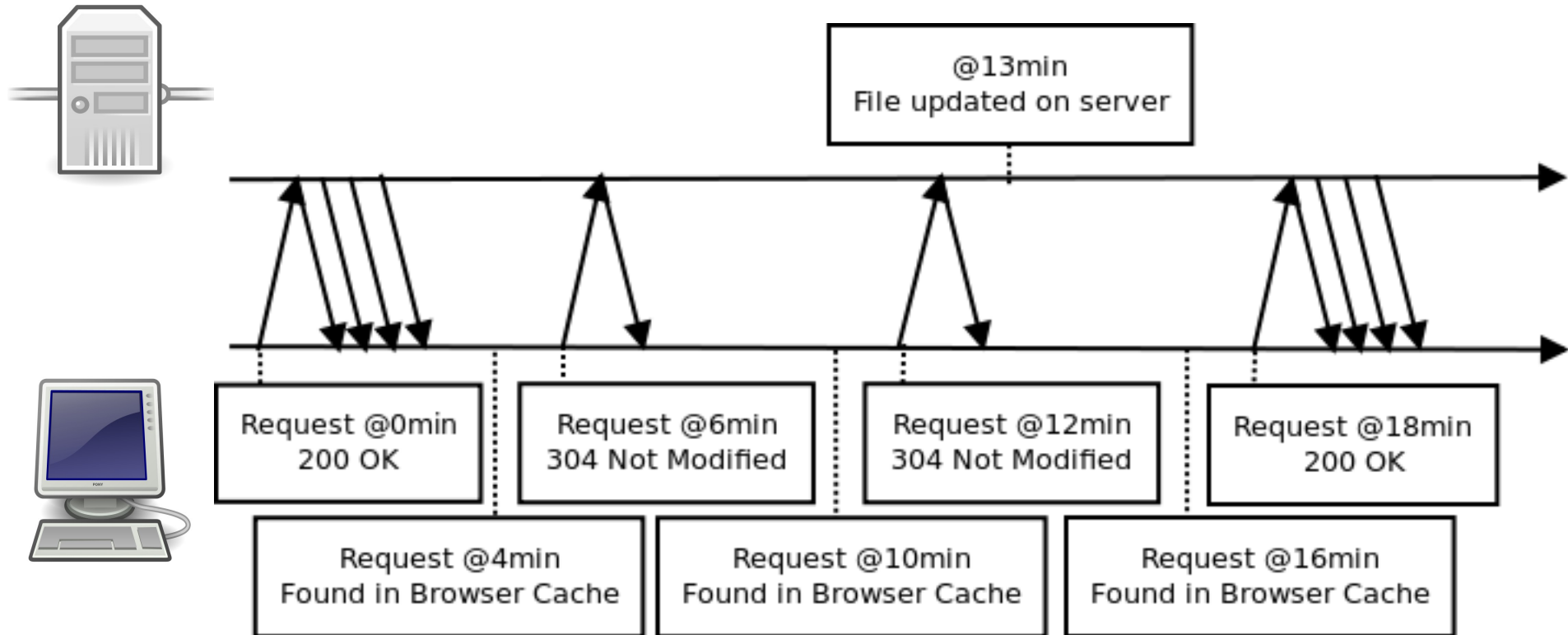
Normal behavior with Cache-Control: 1 year

# extend\_cache in action



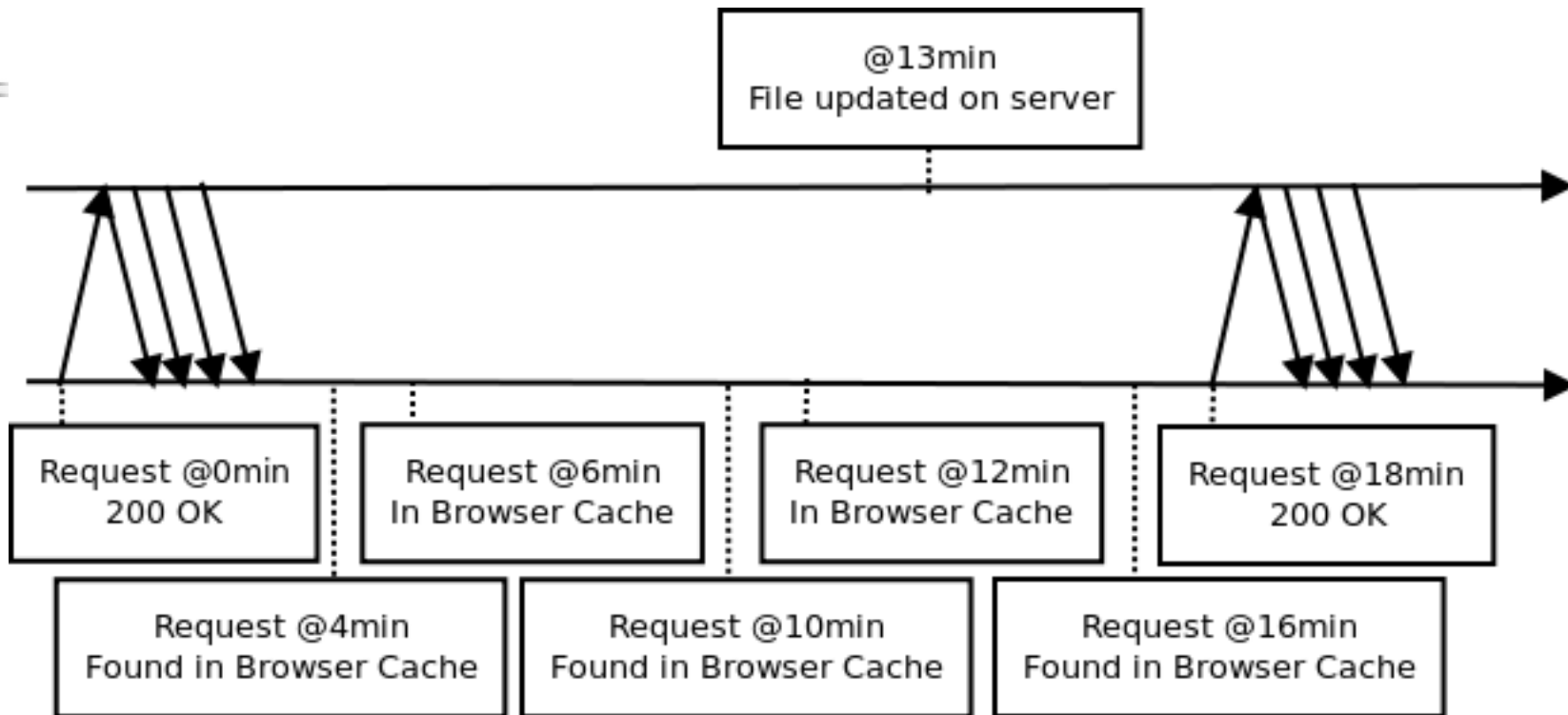
If we edit a file, the user may take 1 year to notice!

# extend\_cache in action



Short cache lifetimes are used to ensure that users see new versions of content.

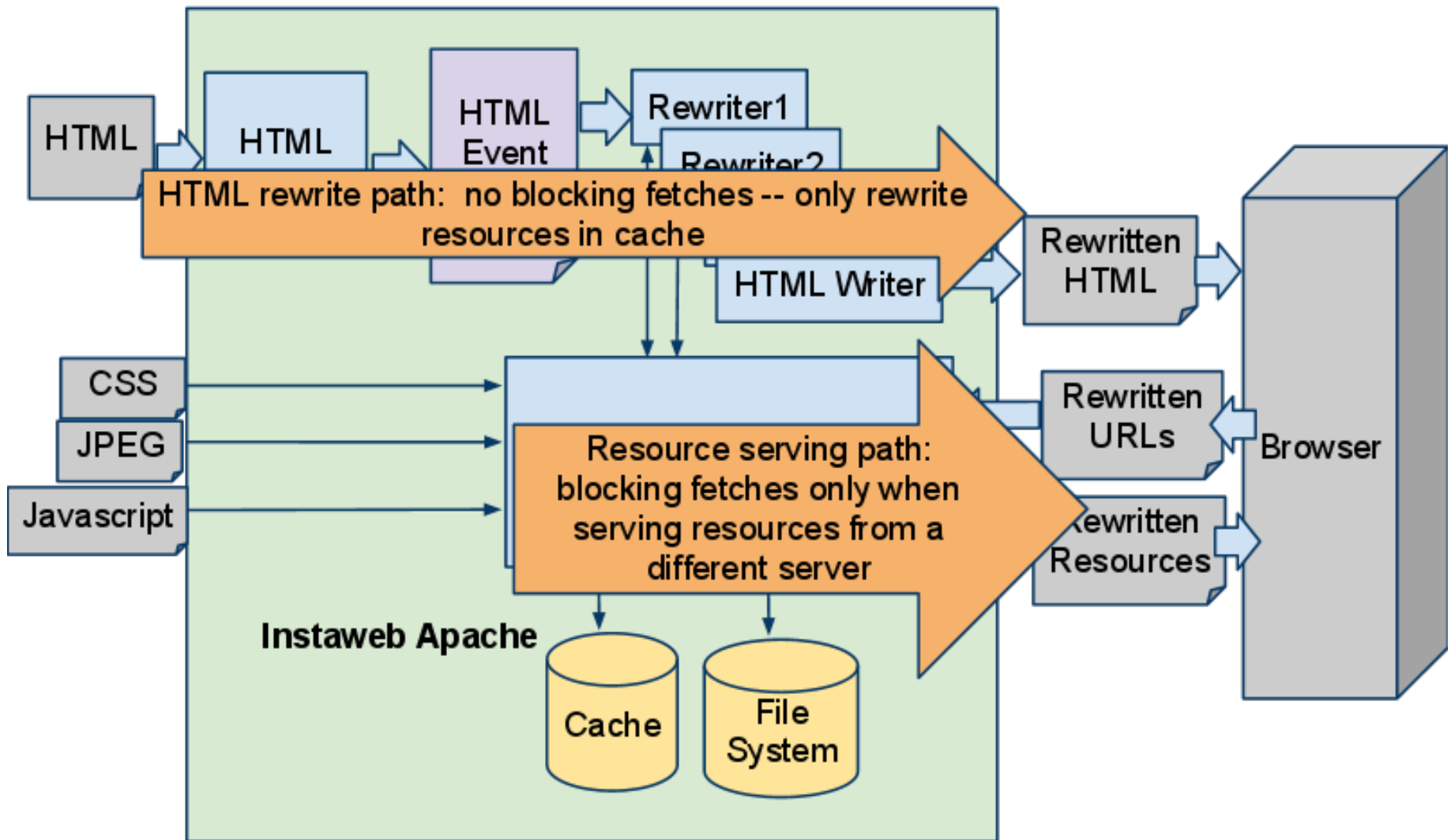
# extend\_cache in action



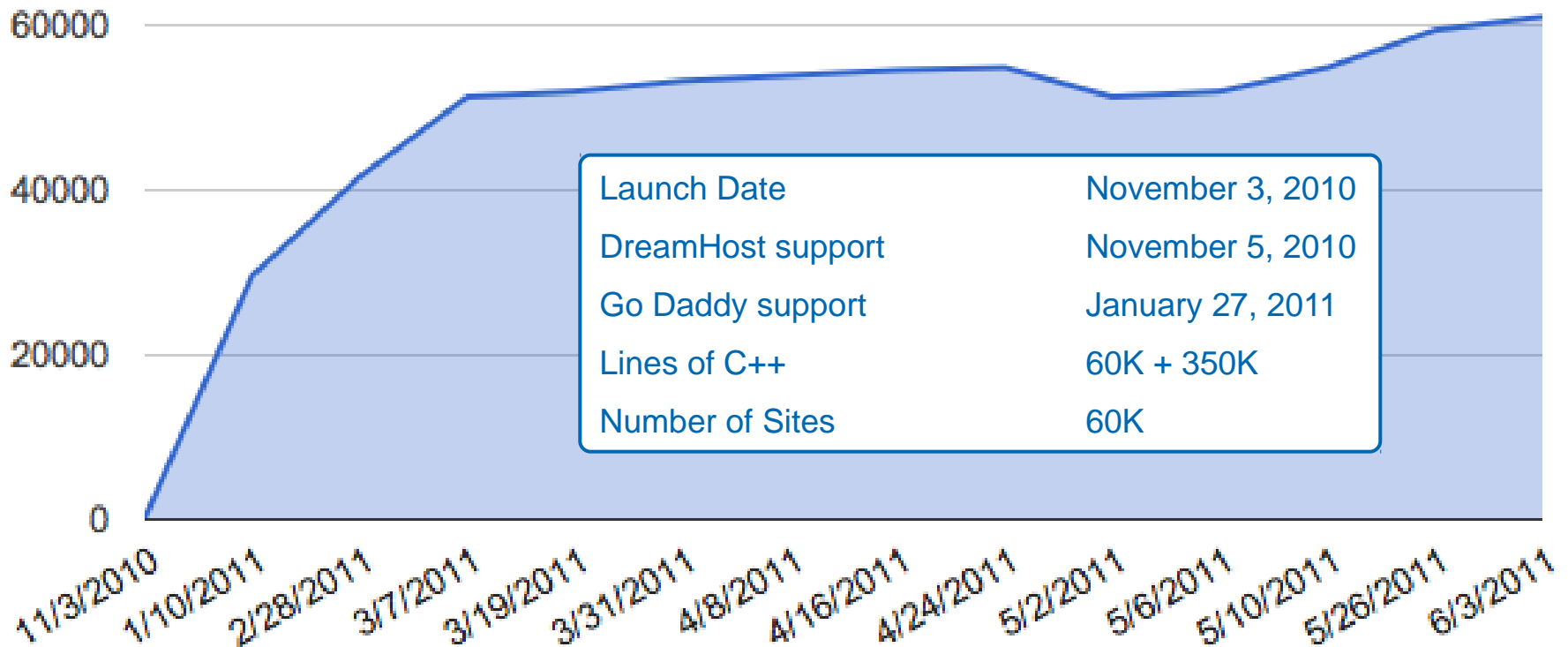
After `extend_cache`, resource urls contain a content hash. Updates are seen immediately.

- **Cache extension** (Sign URL with content hash, increase lifetime to 1 year)
- CSS optimizations
  - Outlining
  - **Minification**
  - **Combining**
  - **Inlining**
  - **Move CSS to head**
  - Sprite multiple background images
- Javascript optimizations
  - Outlining
  - **Minification**
  - Combining (string constants + eval)
  - **Inlining**
- Image optimizations
  - **Re-encode**
  - **Resize** based on context
  - **Add dimensions** to HTML
  - **Inline** using data: URLs
  - **Convert** gif→png, jpg→webp
- HTML optimizations
  - **Combine multiple HEADs**
  - **Remove attribute quotes**
  - Remove unused attributes
  - Remove comments
  - **Left trim urls**

# Streaming Architecture

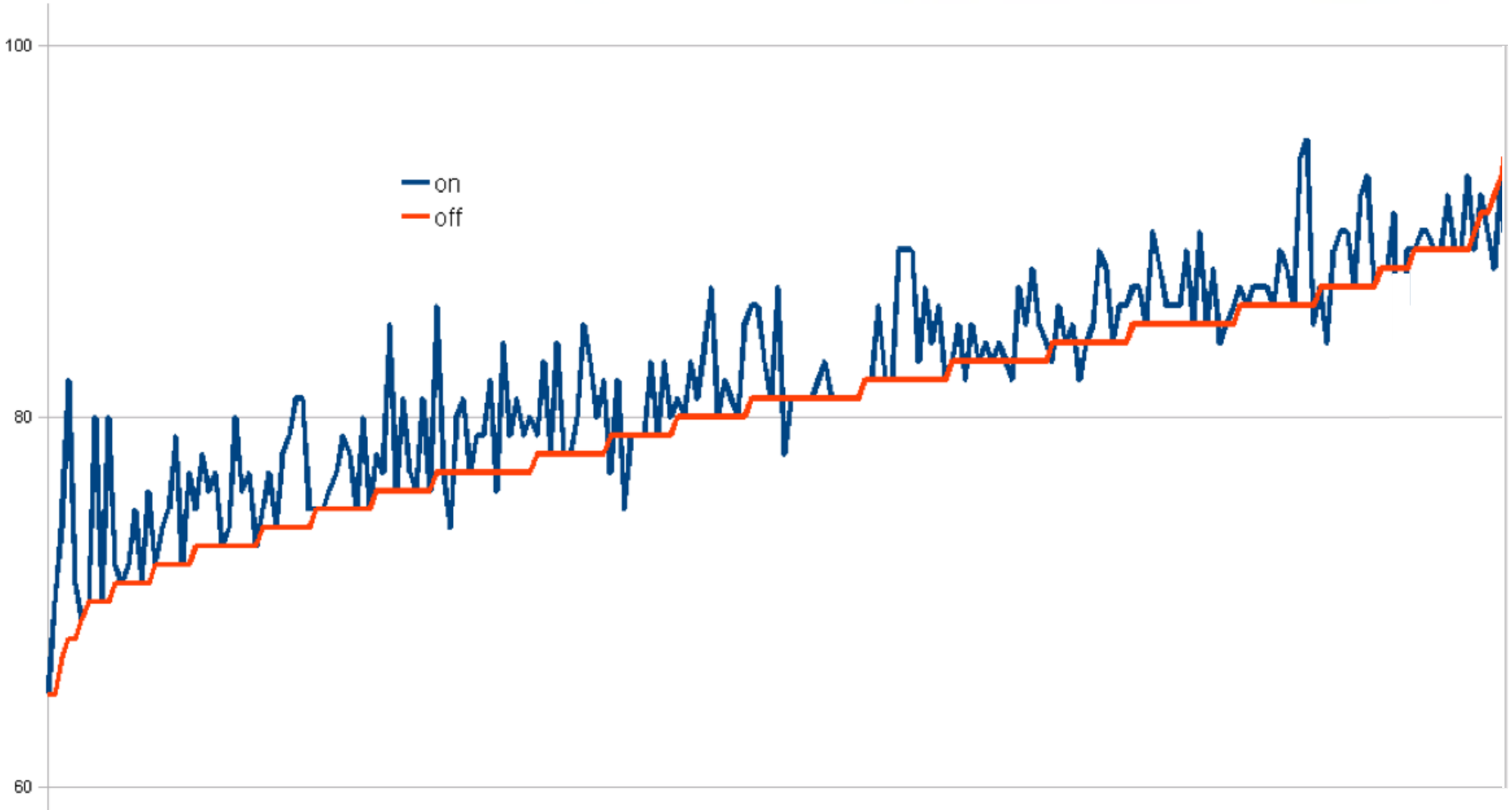


# Evaluating mod\_pagespeed on the web

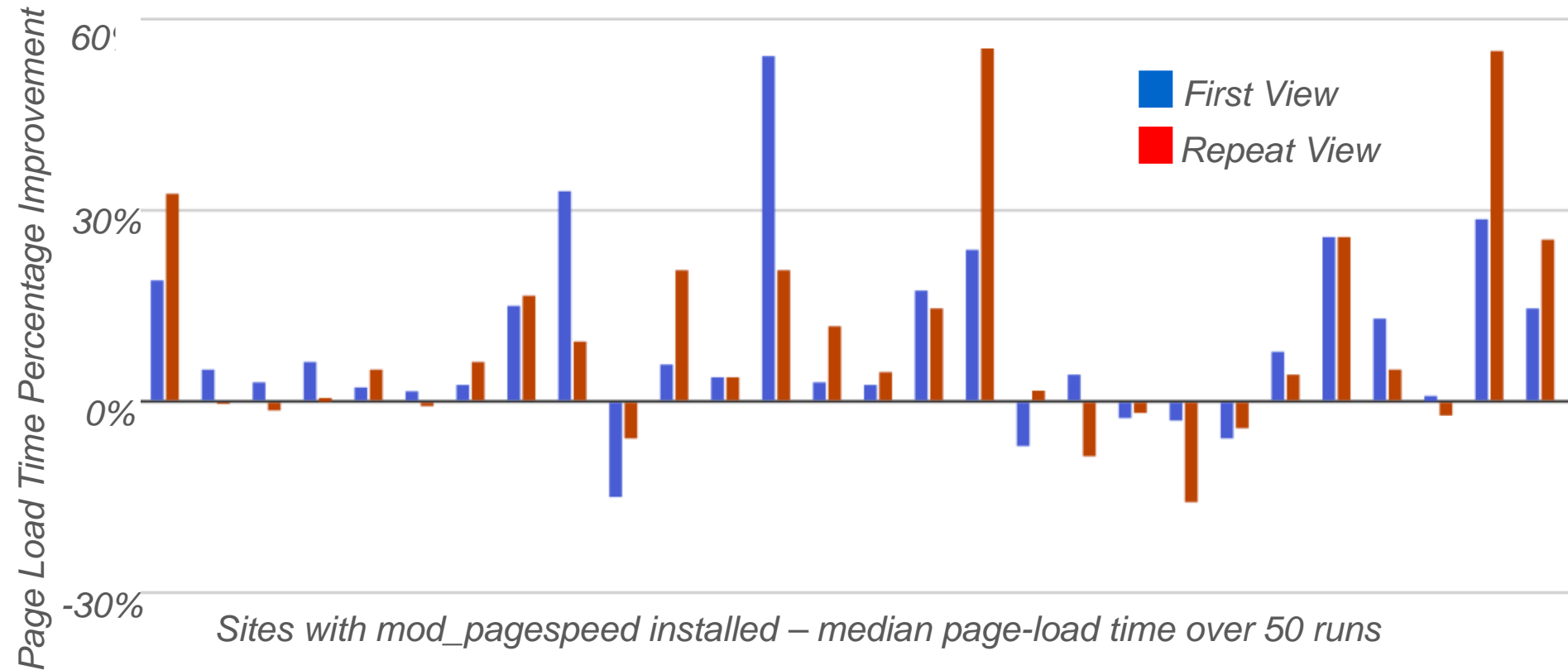




# Page Speed Score improvements



# Latency improvements on 28 sites

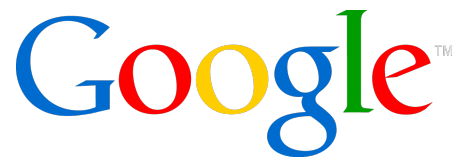


- Validate mod\_pagespeed on thousands of web pages
- Assume that bugs will visibly alter at least one page
- Turn off optimizations that don't validate
- JavaScript can *still* break any optimization we perform
  
- Webmasters can explicitly enable unsafe optimizations

- Current optimizations are simple, safe, and effective
- Plenty more proxy optimizations in the literature
- Browser-dependent optimizations (Ex: webp, HTML5)
- Do all our browsing through a mod\_pagespeed proxy
- Work even better with multi-server setups



# Thank you!



Dynamic Web Optimization Works!

---