

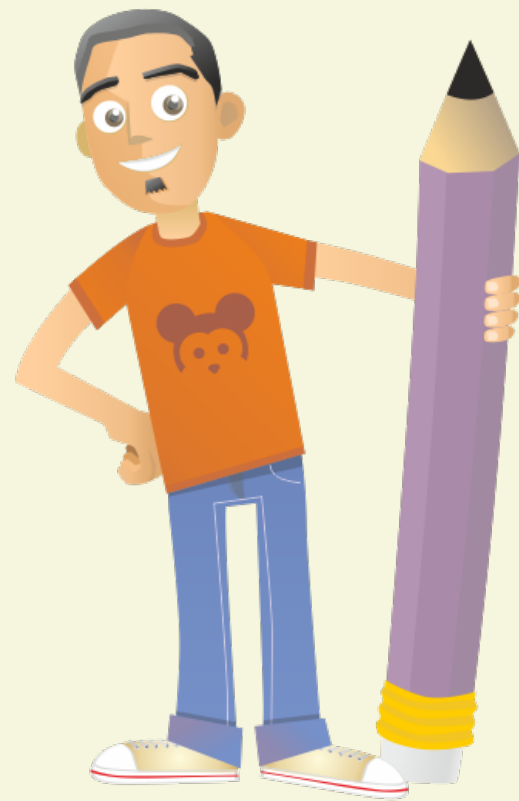
Every second counts

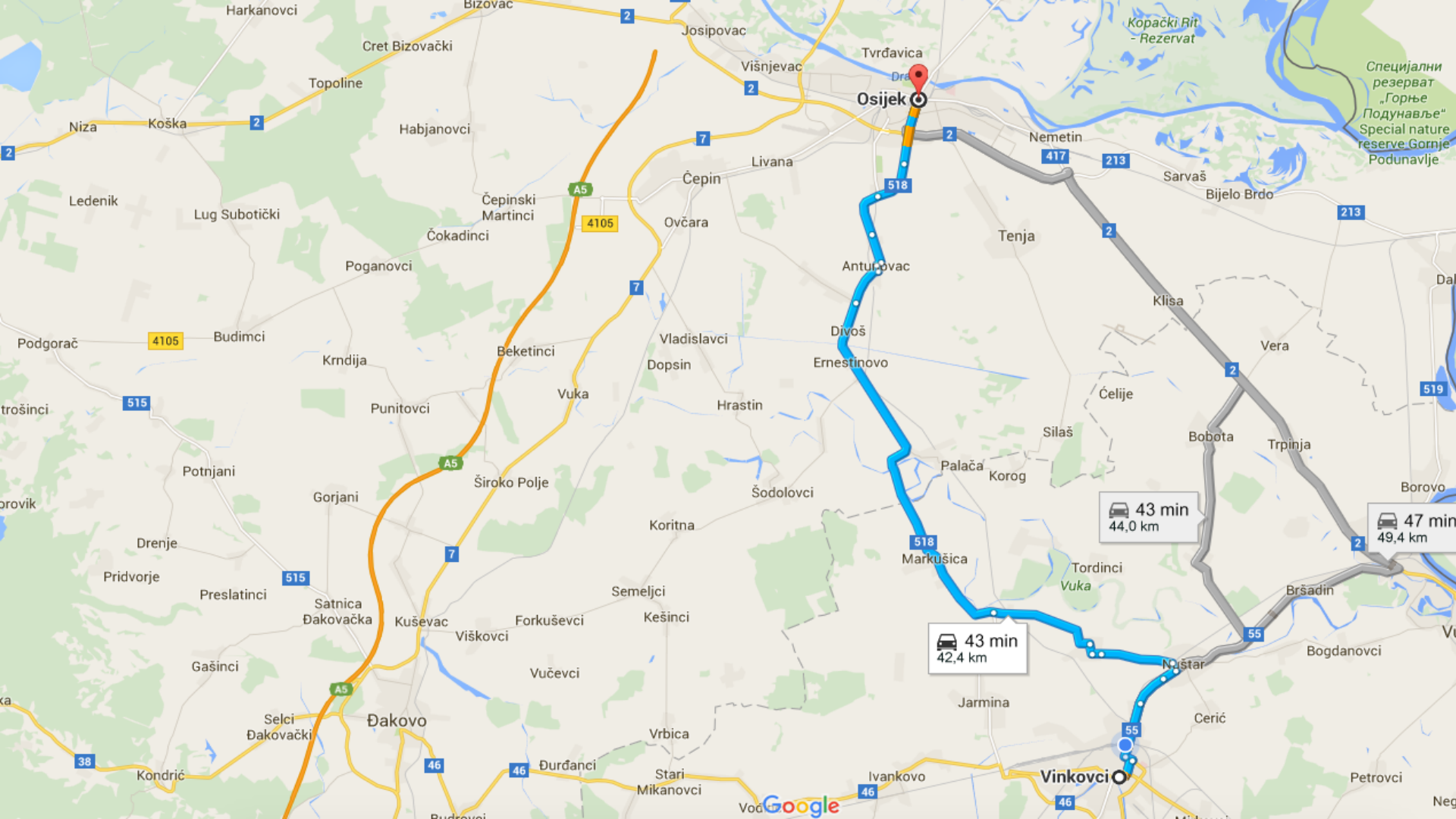
Build faster Magento stores

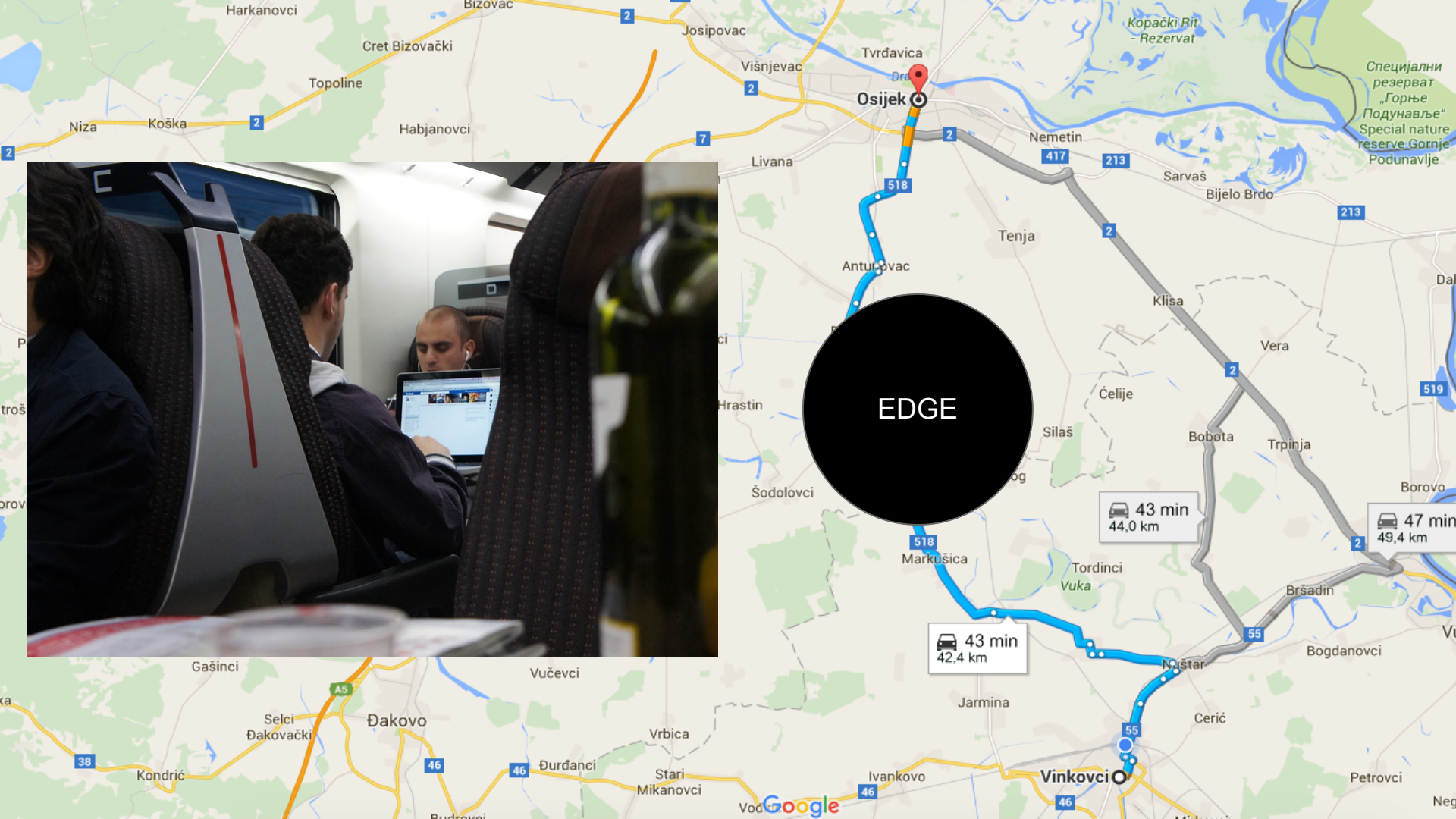
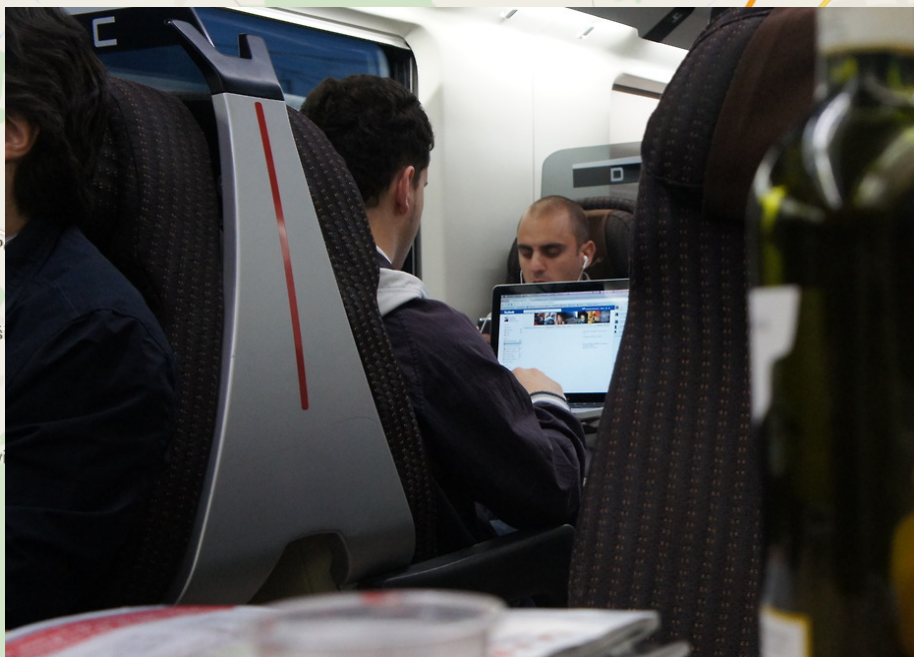
Hrvoje Jurišić, **Inchoo**

Hrvoje Jurisic

Magento frontend developer [@Inchoo](#)
Author of the famous [Inchooers](#) comic strip









Nobody likes waiting



Numbers

**Users expect
2 seconds**

After 3 seconds
40% will leave

**88% of customers
will never come back**

44% of shoppers
fear that slow checkout
means something went
wrong with the transaction

Fast page load builds TRUST

1 second delay
7% less conversions

Google: 500ms latency
20% less traffic

Amazon: 100 ms latency
1% less revenue

Bad performance

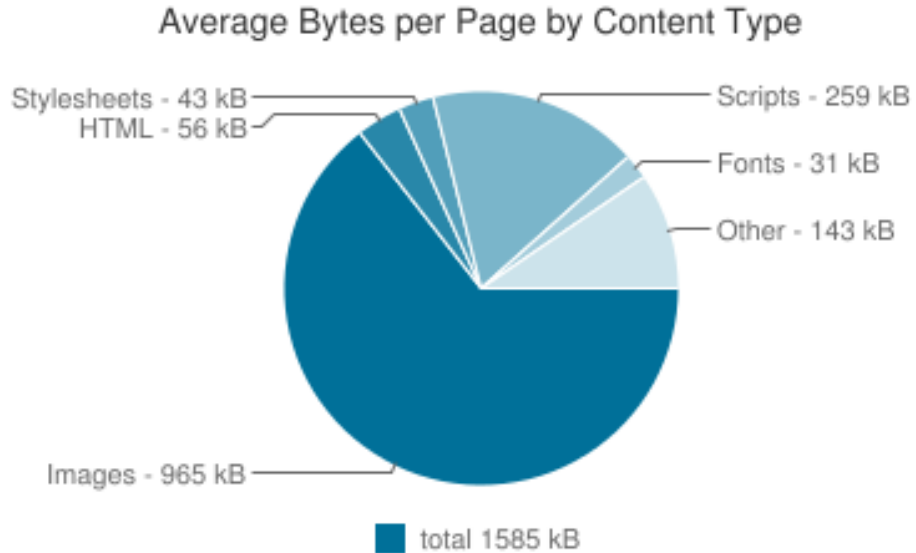
Bad business

2014: eCommerce websites 21% slower than 2013

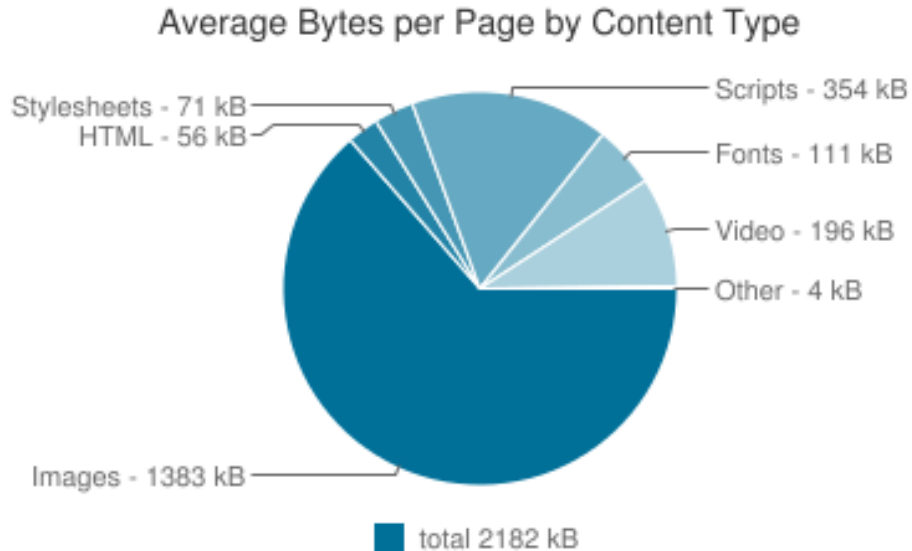
2012 load time: 6.8s

2014 load time: 10.0s

September 2013.



September 2015.





**55% of eCommerce traffic
happens on mobile device**

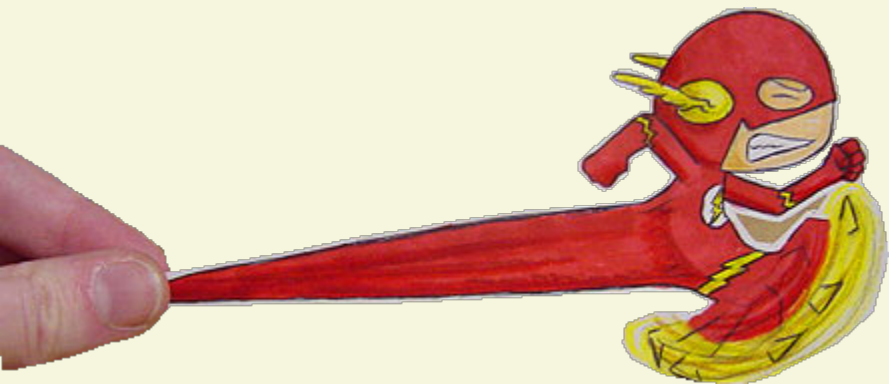
**Mobile users expect sites
to load as fast or even
faster than on their
desktops**

**Up to 97% of mobile carts
are abandoned**

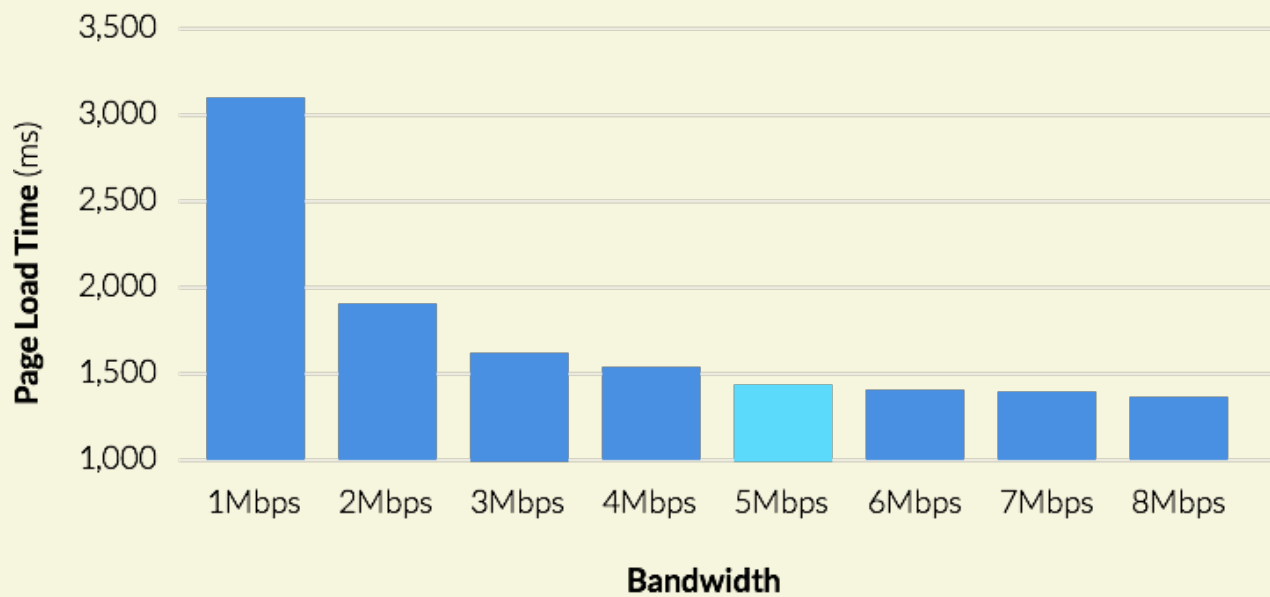


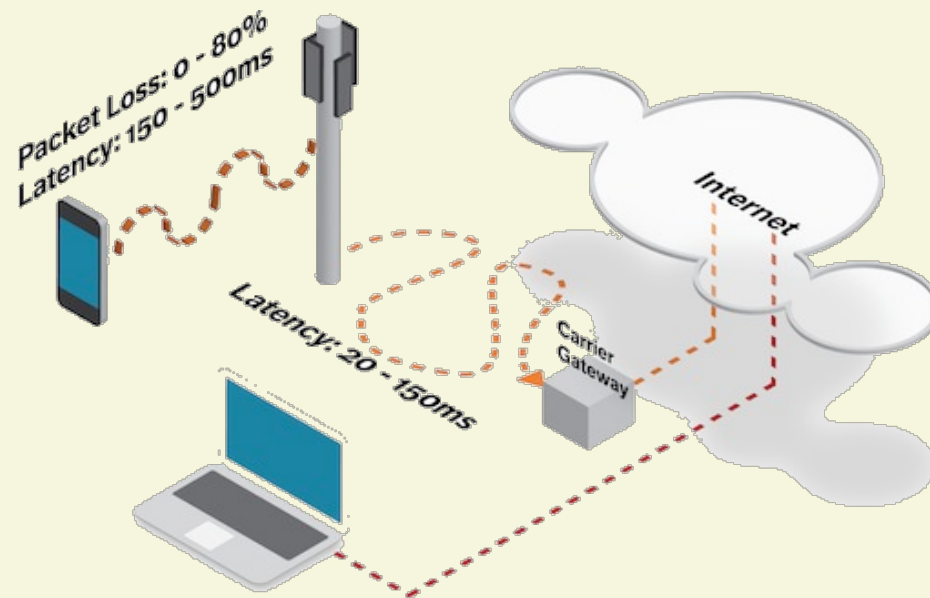


4G to the rescue??



**not so
fast....**





HOW?

Speed is a feature
treat it as a part of
UX design

**Every decision
comes with a cost**

Set a performance budget

Set a performance budget

- **< 3 sec on wi-fi**

Set a performance budget

- < 3 sec on wi-fi
- < 5 sec on 3G

Set a performance budget

- < 3 sec on wi-fi
- < 5 sec on 3G
- < 500 Kb

Set a performance budget

- < 3 sec on wi-fi
- < 5 sec on 3G
- < 500 Kb
- < 30 http requests

**It's a battle,
make it personal**

Optimization

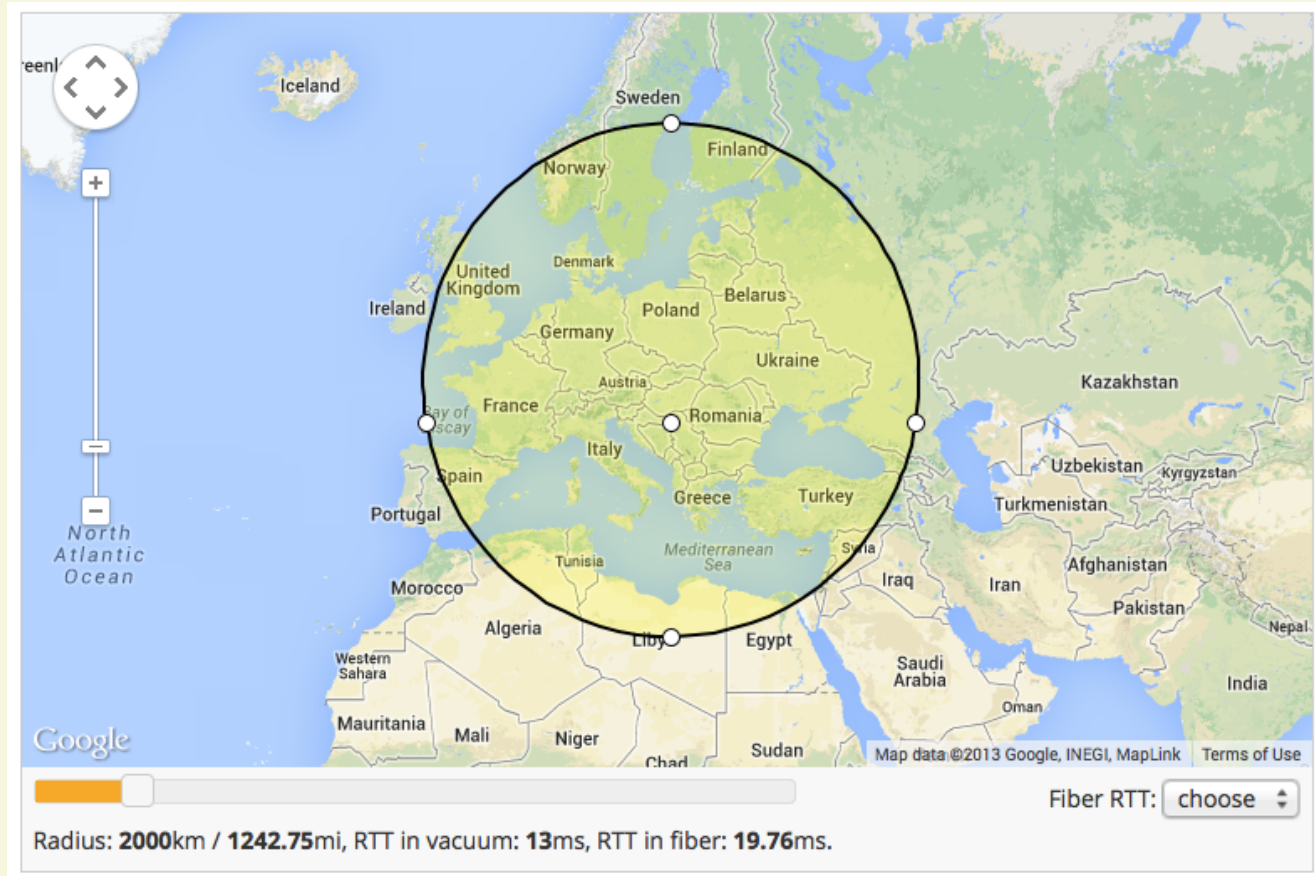
Images

**50% of top 100 retail stores
fail to compress images**

1. Optimize Images

- a. Compress**
- b. Use right format**
- c. Use Responsive Images**

Latency



1. **Optimize Images**
2. **Use CDN**



Requests

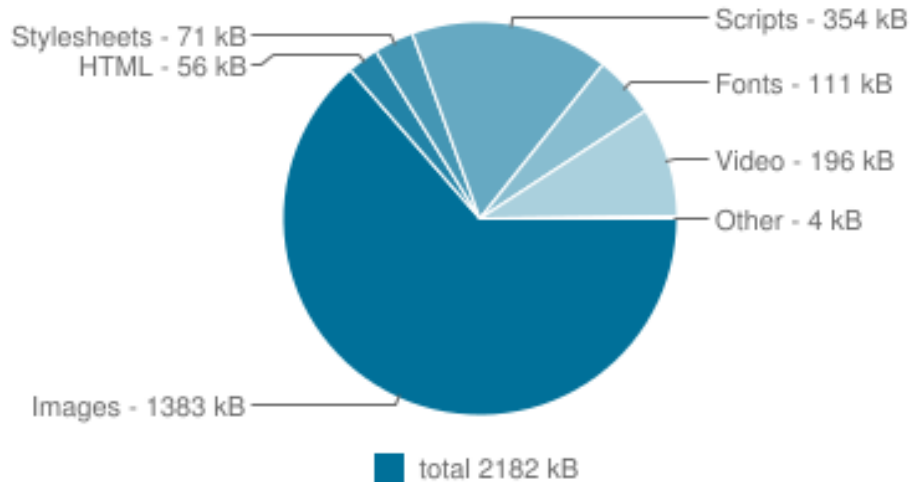
100 http requests in 2014
170 http requests in 2015

- 1. Optimize Images**
- 2. Use CDN**
- 3. Reduce http requests**

- a. Use Sprites**
- b. Use Icon Fonts**
- c. Merge Javascript files**
- d. Merge CSS files**
- e. Lazy Loading**

Page weight

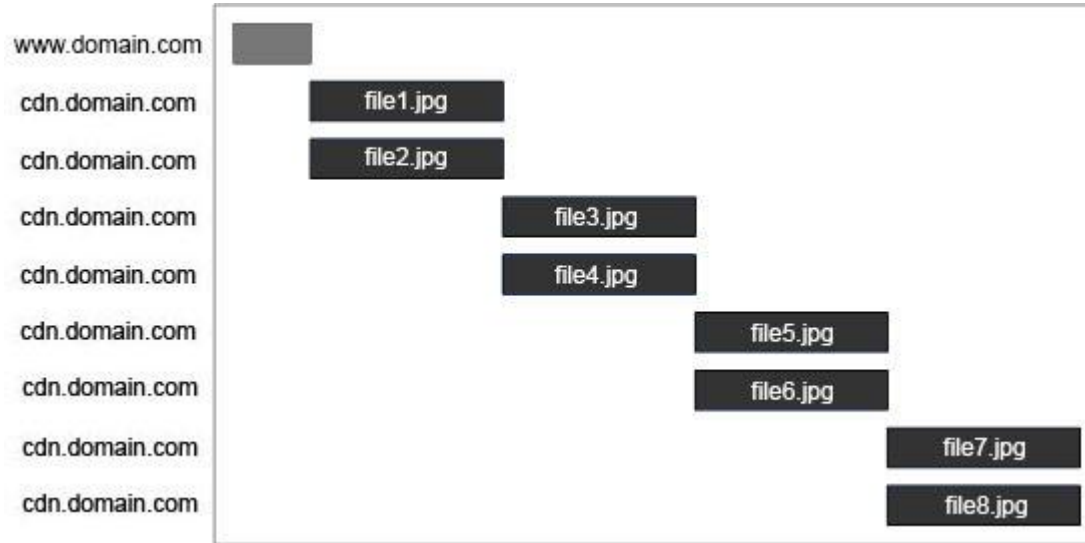
Average Bytes per Page by Content Type



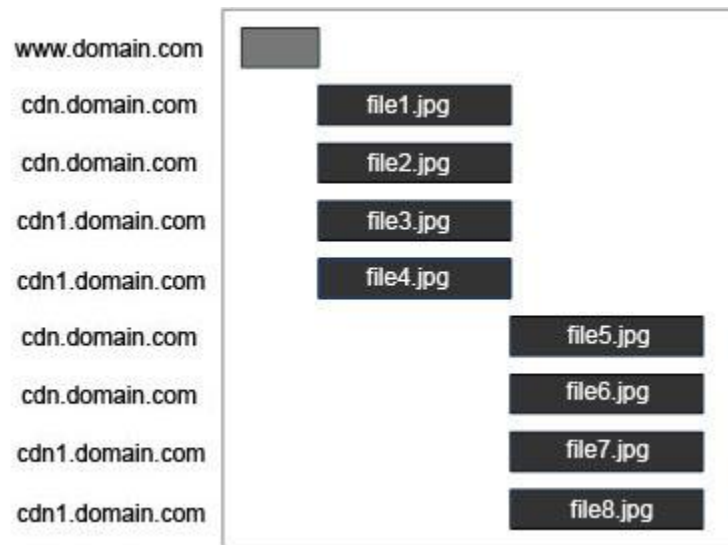
- 1. Optimize Images**
- 2. Use CDN**
- 3. Reduce http requests**
- 4. Reduce page weight**

- a. Optimize Images**
- b. Lazy Loading**
- c. Gzip components**
- d. Minify JS and CSS files**

Parallel downloads



- 1. Optimize Images**
- 2. Use CDN**
- 3. Reduce http requests**
- 4. Reduce page weight**
- 5. Use multiple domains for static assets**

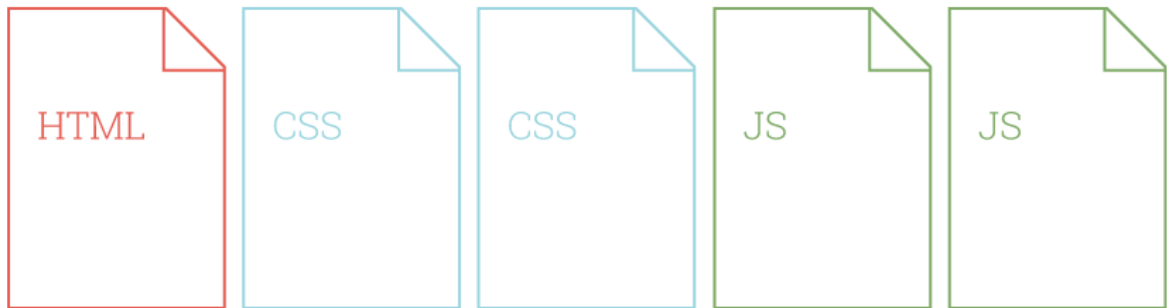
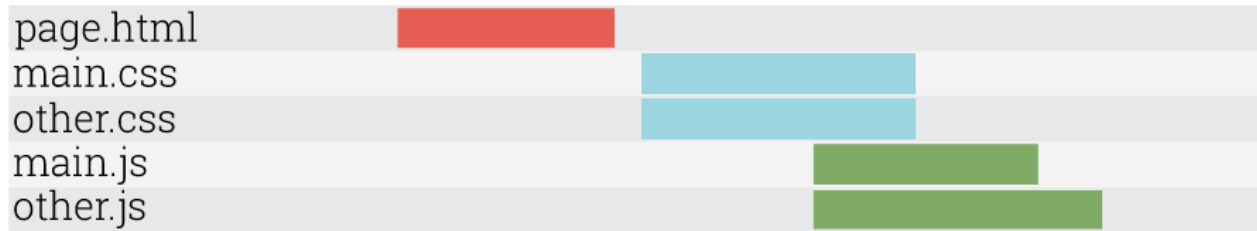


Cache

**The fastest http request
is the one not made**

- 1. Optimize Images**
- 2. Use CDN**
- 3. Reduce http requests**
- 4. Reduce page weight**
- 5. Use multiple domains for static assets**
- 6. Use far future header expires**

Rendering

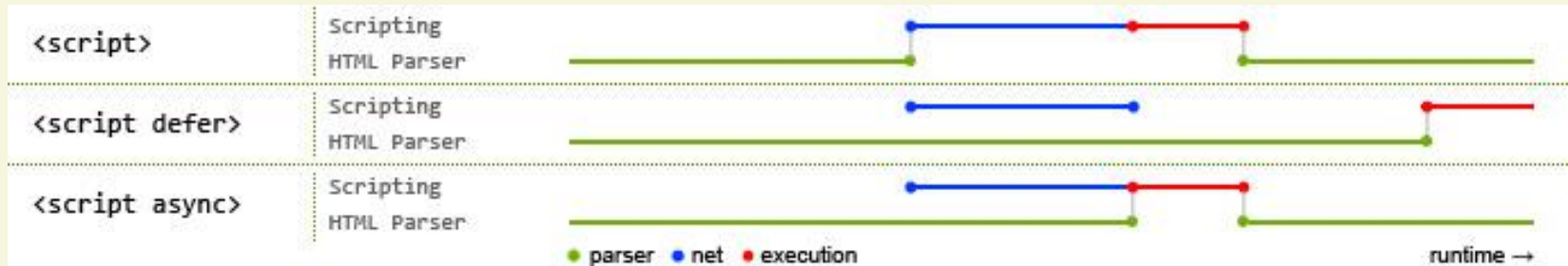


CSS blocks rendering

JS blocks HTML parser

- 1. Optimize Images**
- 2. Use CDN**
- 3. Reduce http requests**
- 4. Reduce page weight**
- 5. Use multiple domains for static assets**
- 6. Use far future header expires**
- 7. Optimize critical rendering path**

- a. CSS file at the top
- b. JS file async or at the bottom



Initial view
(Critical)



Below fold
(Not critical)

**We need render
above the fold content!**

- a. ~~CSS file at the top~~**
- b. JS file async or at the bottom**

- ~~a. CSS file at the top~~
- a. Inline critical CSS at the top**
- b. JS file async or at the bottom**
- c. Load the rest of CSS at the bottom**

Unoptimized



Optimized



0s

0.5s

1.0s

1.5s

2.0s

Tools

- **Developer tools, Firebug**
- **Webpagetest.org**
- **PageSpeed**
- **YSlow**

Remember

**Speed is an important
feature of a webstore**

**Every design decision has
an impact on performance**

Follow best practices for page speed optimization

Thank you

A scenic view of a coastal town, likely Dubrovnik, with a statue on a rock in the foreground. The text "DEVELOPERS PARADISE" is overlaid in the center. The word "DEVELOPERS" is in white with an orange 'D', and "PARADISE" is in white. The background shows a blue sea, a town with white buildings, and green hills under a blue sky with clouds.

DEVELOPERS PARADISE

Lots of Code and Fun in the Sun