BOTTLENECK ANALYSIS

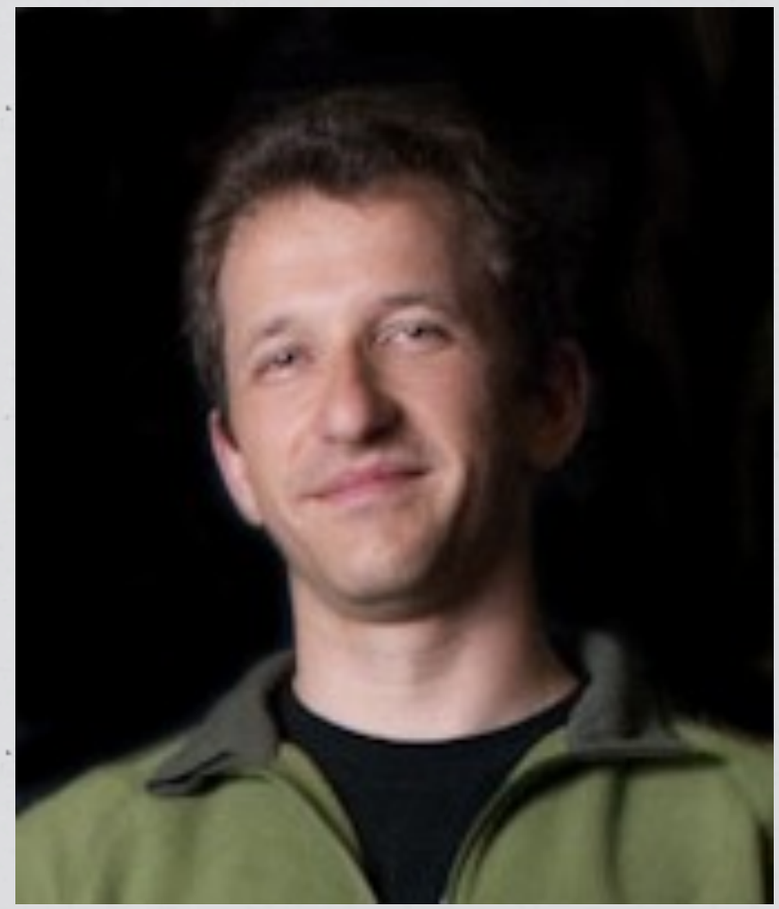
Ilia Alshanetsky - @iliaa

Me, Myself and I

PHP Core Developer

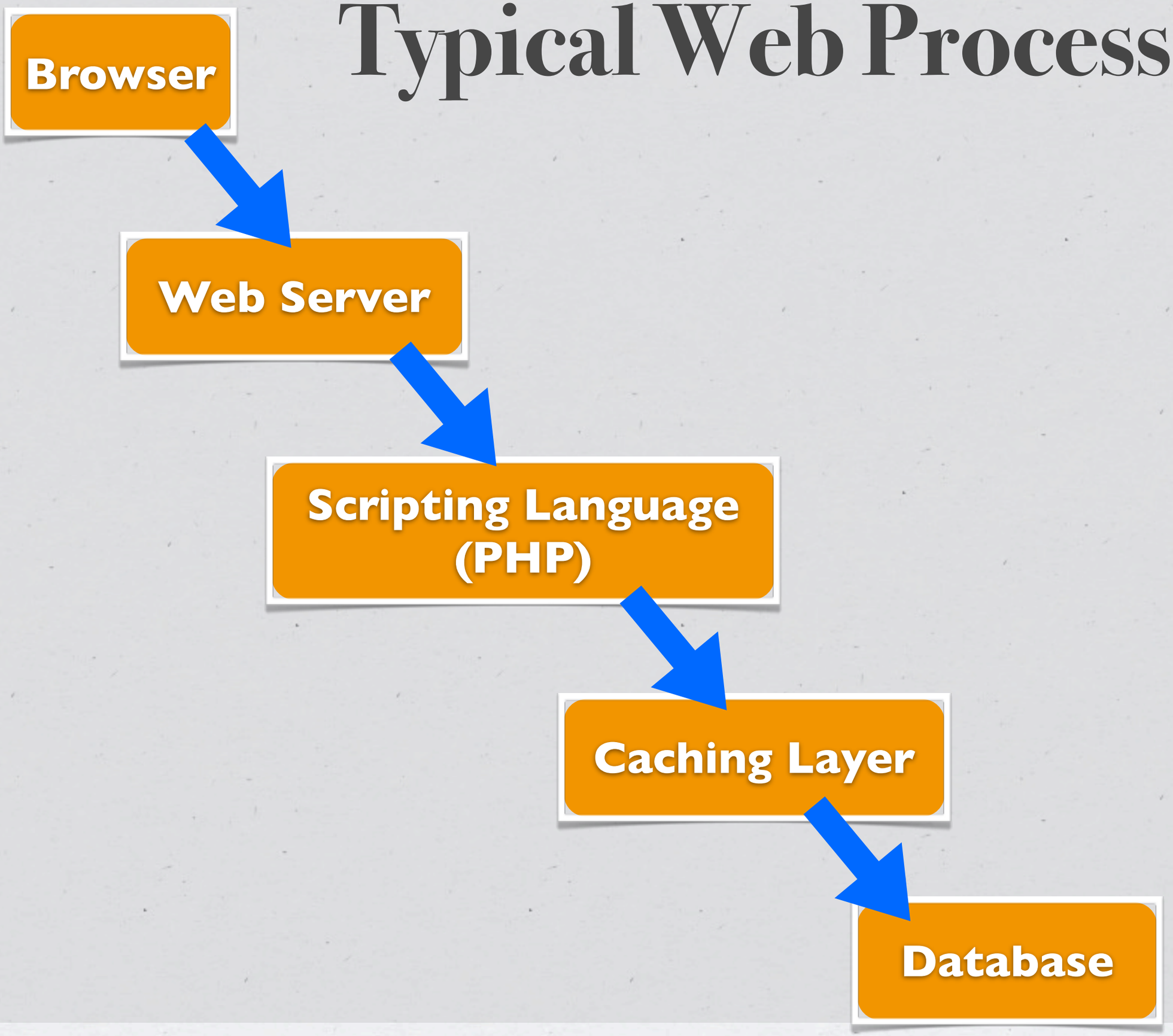
CIO at Centah Inc.

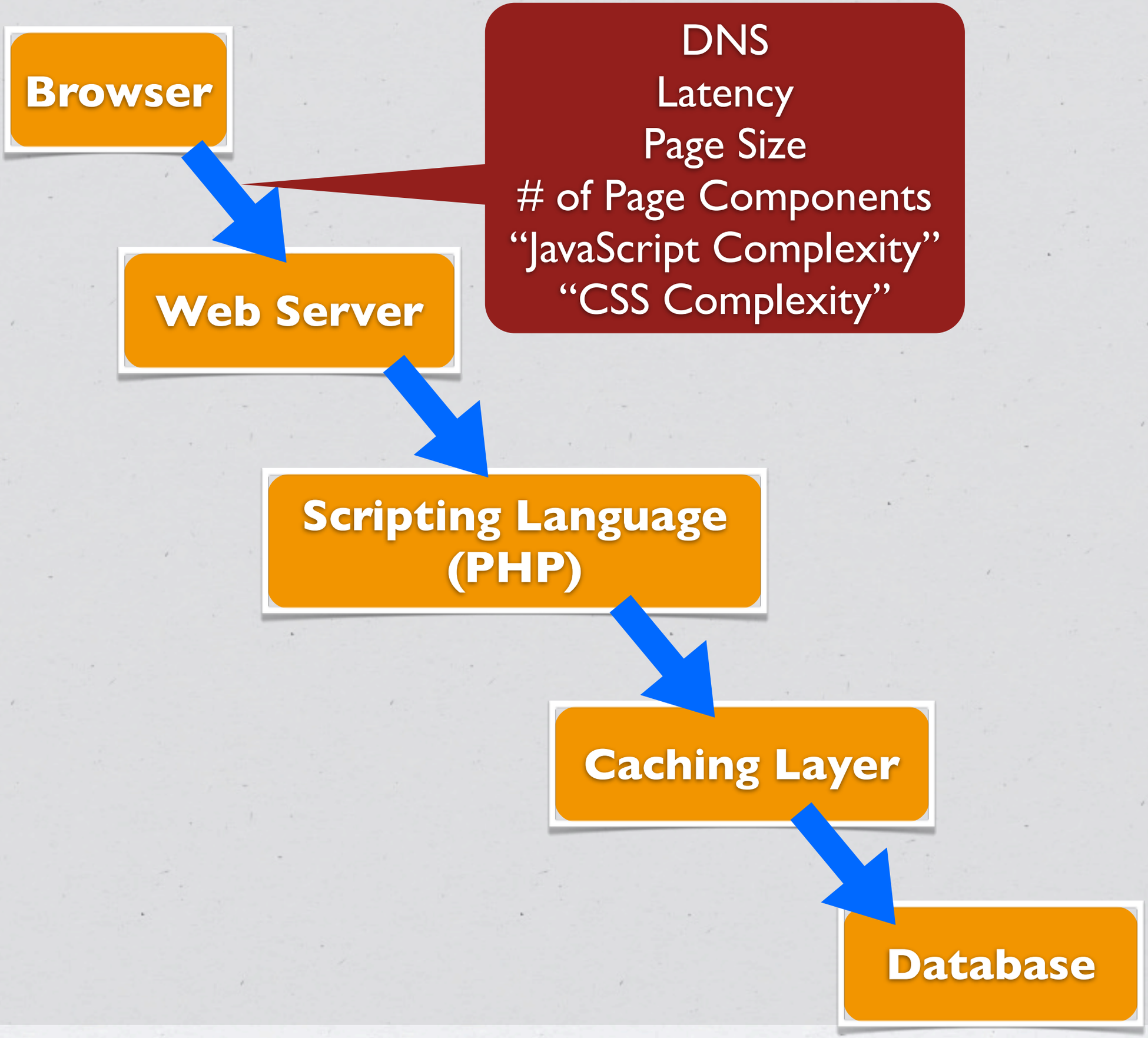
We are Hiring ;-)

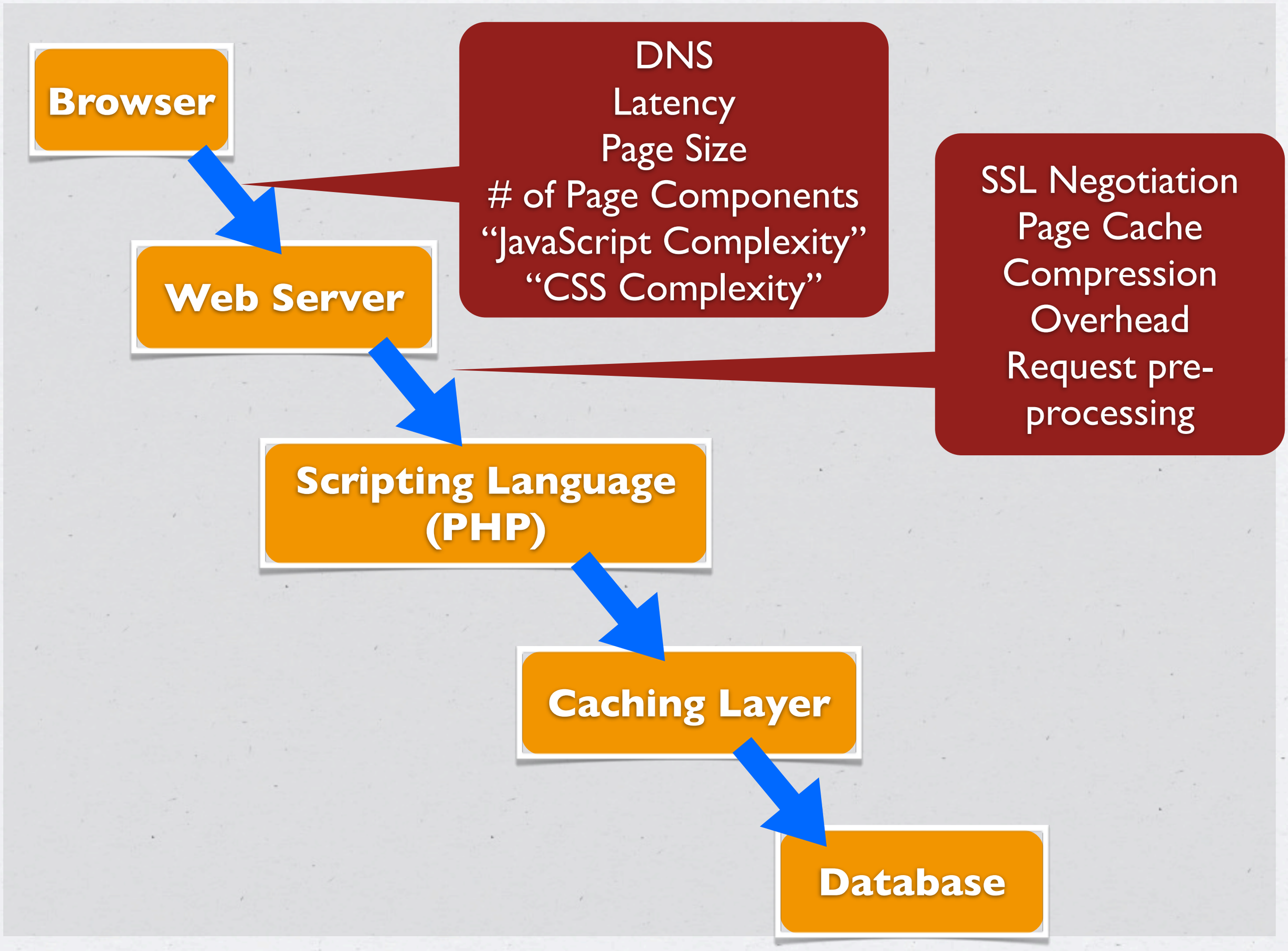


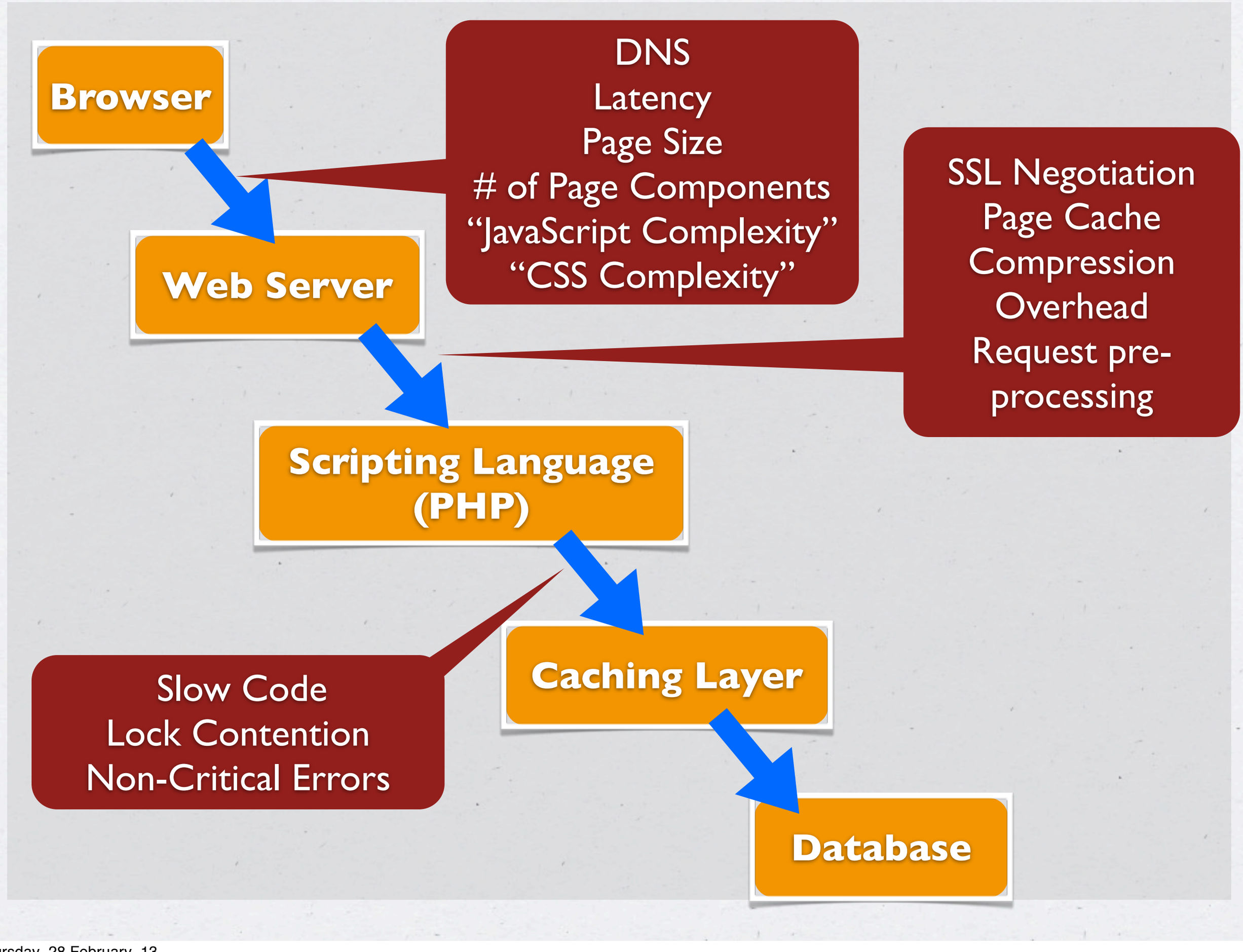
What Causes Bottlenecks?

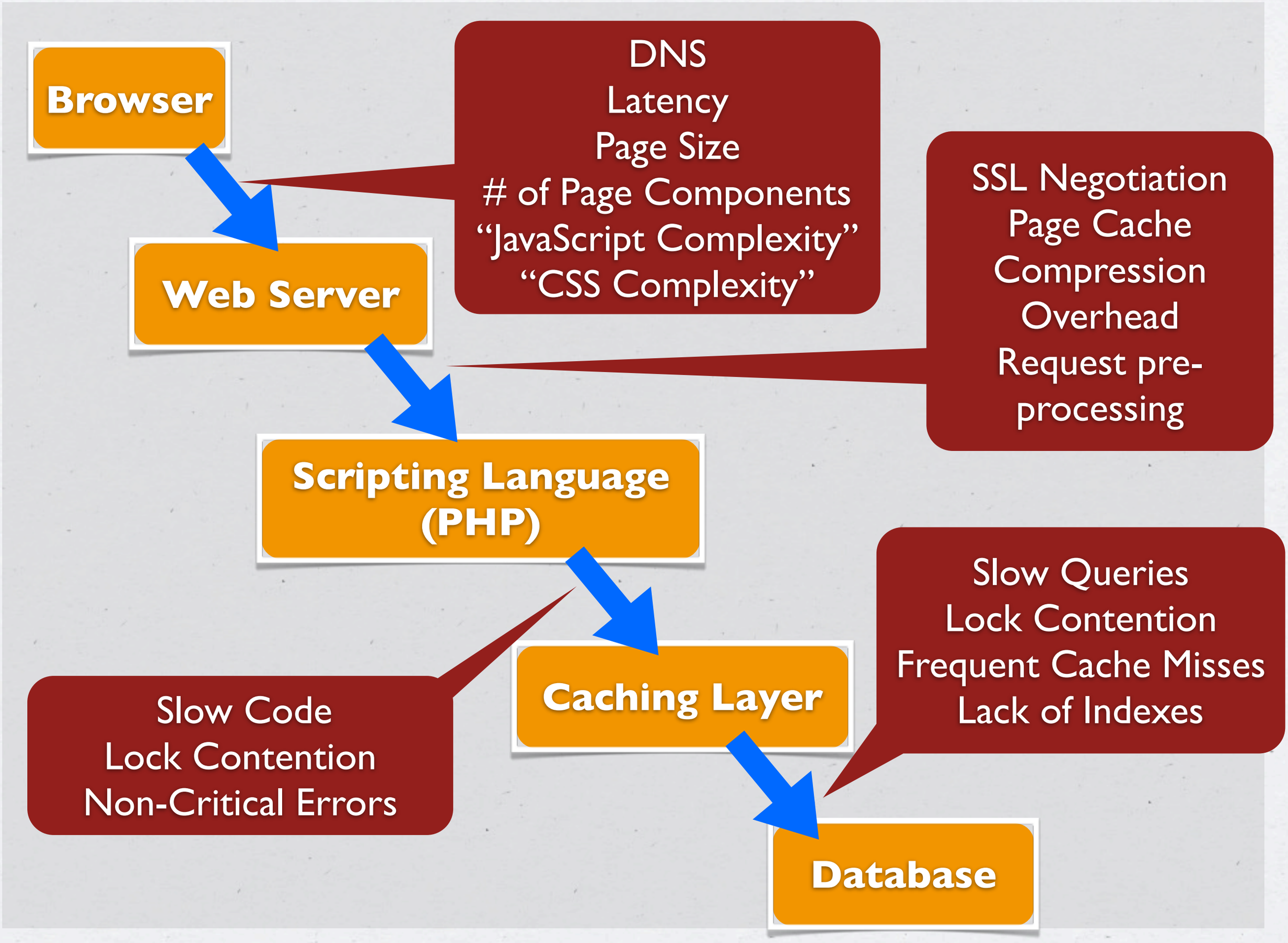
Typical Web Process




Browser
DNS
Latency
Page Size
# of Page Components
“JavaScript Complexity”
“CSS Complexity”
Web Server
Scripting Language (PHP)
Caching Layer
Database


Browser
Web Server
Scripting Language (PHP)
Caching Layer
Database
DNS
Latency
Page Size
# of Page Components
“JavaScript Complexity”
“CSS Complexity”
SSL Negotiation
Page Cache
Compression
Overhead
Request pre-processing


Browser
Web Server
Scripting Language (PHP)
Caching Layer
Database
DNS
Latency
Page Size
# of Page Components
“JavaScript Complexity”
“CSS Complexity”
SSL Negotiation
Page Cache
Compression
Overhead
Request pre-processing
Slow Code
Lock Contention
Non-Critical Errors


Browser
Web Server
Scripting Language (PHP)
Caching Layer
Database
DNS
Latency
Page Size
# of Page Components
"JavaScript Complexity"
"CSS Complexity"
SSL Negotiation
Page Cache
Compression
Overhead
Request pre-processing
Slow Queries
Lock Contention
Frequent Cache Misses
Lack of Indexes
Slow Code
Lock Contention
Non-Critical Errors

A LOT OF POSSIBLE BOTTLENECKS

User Perception = Reality

* If the end result "appears" slow to the user, it will be deemed *SLOW* even if dynamic page generation is super fast...

```
$ ab -vv -n1000 http://localhost/index.php
```

```
Server Software:        Apache/2.2.22
Server Hostname:        localhost
Server Port:            80

Document Length:        10259 bytes

Complete requests:      1000
Failed requests:        0
Broken pipe errors:     0
Requests per second:    2087.68 [#/sec] (mean)
Time per request:       0.48 [ms] (mean)
```

User Perception = Reality

* If the end result "appears" slow to the user, it will be deemed *SLOW* even if dynamic page generation is super fast...

```
$ ab -vv -n1000 http://localhost/index.php
```

```
Server Software:        Apache/2.2.22
Server Hostname:        localhost
Server Port:            80

Document Length:        10259 bytes

Complete requests:      1000
Failed requests:        0
Broken pipe errors:     0
Requests per second:    2087.68 [#/sec] (mean)
Time per request:       0.48 [ms] (mean)
```

PROFILING THE USER EXPERIENCE

BROWSER TEST

Using Chrome

Name / Path	Method	Type	Initiator	Size / Content	Time / Latency	Timeline (624ms · 936ms · 1.25s · 1.56s · 1.87s)
http://php.net/	GET	text/html	Other	49.28KB / 48.95KB	1.61s / 406ms	
site.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	19.85KB / 19.64KB	939ms / 183ms	
phpnet.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	1.19KB / 1000B	381ms / 198ms	
print.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	710B / 496B	541ms / 370ms	
userprefs.js static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB / 2.54KB	380ms / 378ms	
php.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:3 Parser	2.67KB / 2.46KB	184ms / 170ms	
small_submit_white.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:7 Parser	280B / 67B	401ms / 203ms	
leftbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	422B / 207B	412ms / 203ms	
rightbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	392B / 178B	362ms / 184ms	184ms

BROWSER TEST

Using Chrome

★ ★ ★ ★ ★ ★ ★ ★

Name Path	Method	Type	Initiator	Size Content	Time Latency	Timeline
http://php.net/	GET	text/html	Other	49.28KB 48.95KB	1.61s 406ms	406ms — 1.20s
site.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	19.85KB 19.64KB	939 183	
phpnet.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	1.19KB 1000B	381 198	
print.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	710B 496B	541 370	
userprefs.js static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB 2.54KB	380ms 378ms	
php.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:3 Parser	2.67KB 2.46KB	184ms 170ms	
small_submit_white.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:7 Parser	280B 67B	401ms 203ms	
leftbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	422B 207B	412ms 203ms	
rightbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	392B 178B	362ms 184ms	184ms

624ms 936ms 1.25s 1.56s 1.87s

DNS Lookup 0
Connecting 206ms
Sending 0
Waiting 199ms
Receiving 1.20s

BROWSER TEST

Using Chrome

Eeek! 1/5 of a second just to connect!

Name / Path				Size / Content	Time / Latency	Timeline
http://php.net/				49.28KB	1.61s	406ms – 1.20s
site.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	19.85KB 19.64KB	939 1	
phpnet.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	1.19KB 1000B	381 198	
print.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	710B 496B	541 370	
userprefs.js static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB 2.54KB	380ms 378ms	
php.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:3 Parser	2.67KB 2.46KB	184ms 170ms	
small_submit_white.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:7 Parser	280B 67B	401ms 203ms	
leftbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	422B 207B	412ms 203ms	
rightbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	392B 178B	362ms 184ms	184ms

Timeline: 624ms, 936ms, 1.25s, 1.56s, 1.87s

DNS Lookup 0
Connecting 206ms
Sending 0
Waiting 199ms
Receiving 1.20s

BROWSER TEST

Using Chrome

Eeek! 1/5 of a second just to connect!

Page speed is more or less ok

Name / Path	Method	Type	Initiator	Size / Content	Time / Latency
http://php.net/				49.28KB	1.61s
				42.05KB	406ms
site.css static.php.net/www.php.net/s	GET	text/css	http://php.net/:1	19.85KB 19.64KB	939 1
phpnet.css static.php.net/www.php.net/				1.19KB 1000B	381 198
print.css static.php.net/www.php.net/				710B 496B	5 370
userprefs.js static.php.net/www.php.net			Parser	2.76KB 2.54KB	380ms 378ms
php.gif static.php.net/www.php.net/im	GET	image/gif	http://php.net/:3 Parser	2.67KB 2.46KB	184ms 170ms
small_submit_white.gif static.php.net/www.php.net/im	GET	image/gif	http://php.net/:7 Parser	280B 67B	401ms 203ms
leftbar.png static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	422B 207B	412ms 203ms
rightbar.png static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	392B 178B	362ms 184ms

Timeline: 624ms, 936ms, 1.25s, 1.56s, 1.87s

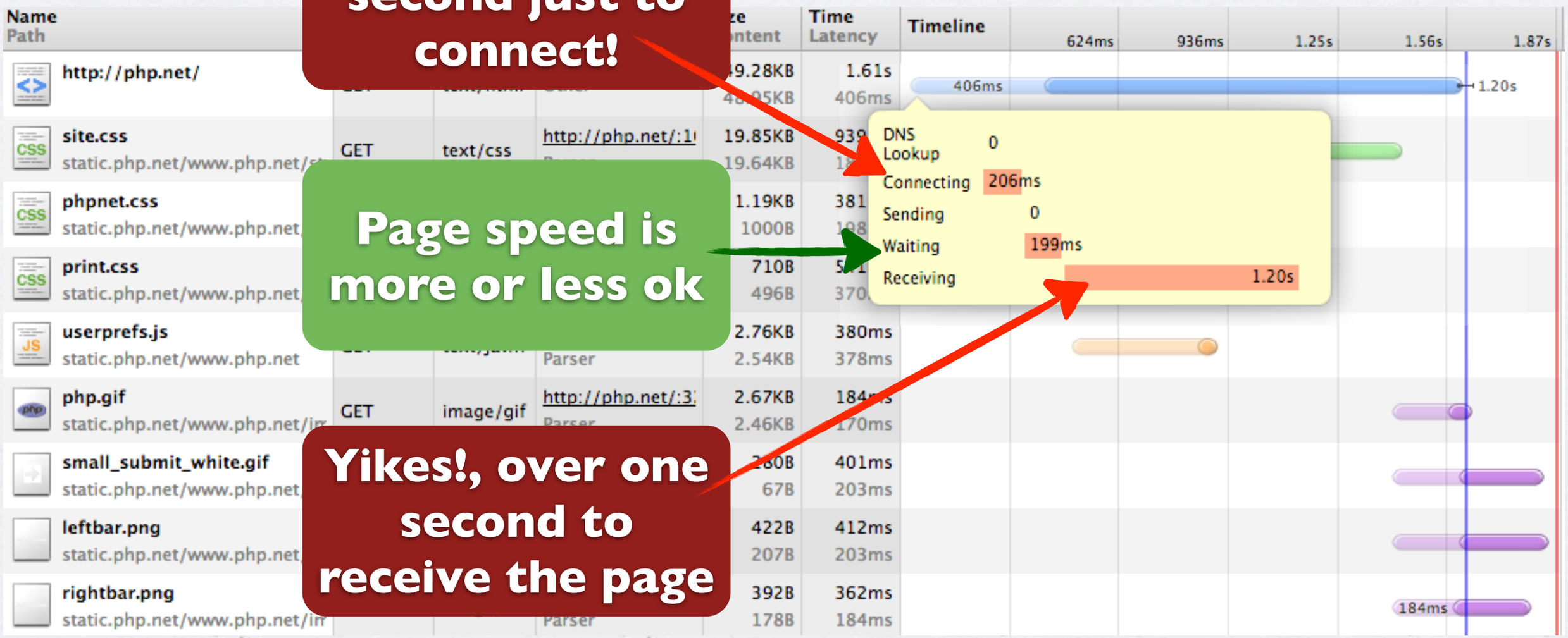
406ms — 1.20s

DNS Lookup	0
Connecting	206ms
Sending	0
Waiting	199ms
Receiving	1.20s

184ms

BROWSER TEST

Using Chrome



BROWSER TEST

Using Chrome

Name / Path	Method	Type	Initiator	Size / Content	Time / Latency	Timeline
http://php.net/	GET	text/html	Other	49.28KB / 48.95KB	1.61s / 406ms	
site.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	19.85KB / 19.64KB	939ms / 183ms	
phpnet.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	1.19KB / 1000B	381ms / 198ms	
print.css static.php.net/www.php.net/st	GET	text/css	http://php.net/:1 Parser	710B / 496B	541ms / 370ms	
userprefs.js static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB / 2.54KB	380ms / 378ms	
php.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:3 Parser	2.67KB / 2.46KB	184ms / 170ms	
small_submit_white.gif static.php.net/www.php.net/in	GET	image/gif	http://php.net/:7 Parser	280B / 67B	401ms / 203ms	
leftbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	422B / 207B	412ms / 203ms	
rightbar.png static.php.net/www.php.net/in	GET	image/...	http://php.net/:7 Parser	392B / 178B	362ms / 184ms	

Timeline: 624ms, 936ms, 1.25s, 1.56s, 1.87s

170ms 14ms

Blocking 0
Sending 1ms
Waiting 167ms
Receiving 14ms

184ms

BROWSER TEST

Using Chrome

Name / Path	Method	Type	Initiator	Size / Content	Time / Latency	Timeline (624ms, 936ms, 1.25s, 1.56s, 1.87s)
http://php.net/	GET	text/html	Other	49.28KB / 48.95KB	1.61s / 406ms	
site.css — static.php.net/www.php.net/s				9.85KB / .64KB	939ms / 183ms	
phpnet.css — static.php.net/www.php.net/s				.19KB / 000B	381ms / 198ms	
print.css — static.php.net/www.php.net/s				710B / 96B	541ms / 370ms	
userprefs.js — static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB / 2.54KB	380ms / 37[illegible]	
php.gif — static.php.net/www.php.net/im	GET	image/gif	http://php.net/:3 Parser	2.67KB / 2.46KB	184ms / 170ms	
small_submit_white.gif — static.php.net/www.php.net/im	GET	image/gif	http://php.net/:7 Parser	280B / 67B	401ms / 203ms	
leftbar.png — static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	422B / 207B	412ms / 203ms	
rightbar.png — static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	392B / 178B	362ms / 184ms	

162ms to send process static request!

170ms 14ms

Blocking	0
Sending	1ms
Waiting	167ms
Receiving	14ms

184ms

BROWSER TEST

Using Chrome

Name / Path	Method	Type	Initiator	Size / Content	Time / Latency	Timeline
http://php.net/	GET	text/html	Other	49.28KB / 48.95KB	1.61s / 406ms	406ms, 1.20s
site.css static.php.net/www.php.net/s				9.85KB / .64KB	939 / 183	
phpnet.css static.php.net/www.php.net/s				.19KB / .000B	381 / 198	
print.css static.php.net/www.php.net/s				710B / 96B	541 / 370	
userprefs.js static.php.net/www.php.net	GET	text/jav...	http://php.net/:2 Parser	2.76KB / 2.54KB	380ms / 378ms	
php.gif static.php.net/www.php.net/im	GET	image/gif	http://php.net/:3 Parser	2.67KB / 2.46KB	184ms / 170ms	170ms, 14ms
small_submit_white.gif static.php.net/www.php.net/im	GET	image/gif	http://php.net/:7 Parser	280B / 67B	401ms / 203ms	
leftbar.png static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	422B / 207B	412ms / 203ms	
rightbar.png static.php.net/www.php.net/im	GET	image/...	http://php.net/:7 Parser	392B / 178B	362ms / 184ms	184ms

DNS Lookup 0
Connecting 206ms
Sending 0
Waiting 199ms
Receiving 1.20s

Blocking 0
Sending 1ms
Waiting 167ms
Receiving 14ms

162ms to send process static request!

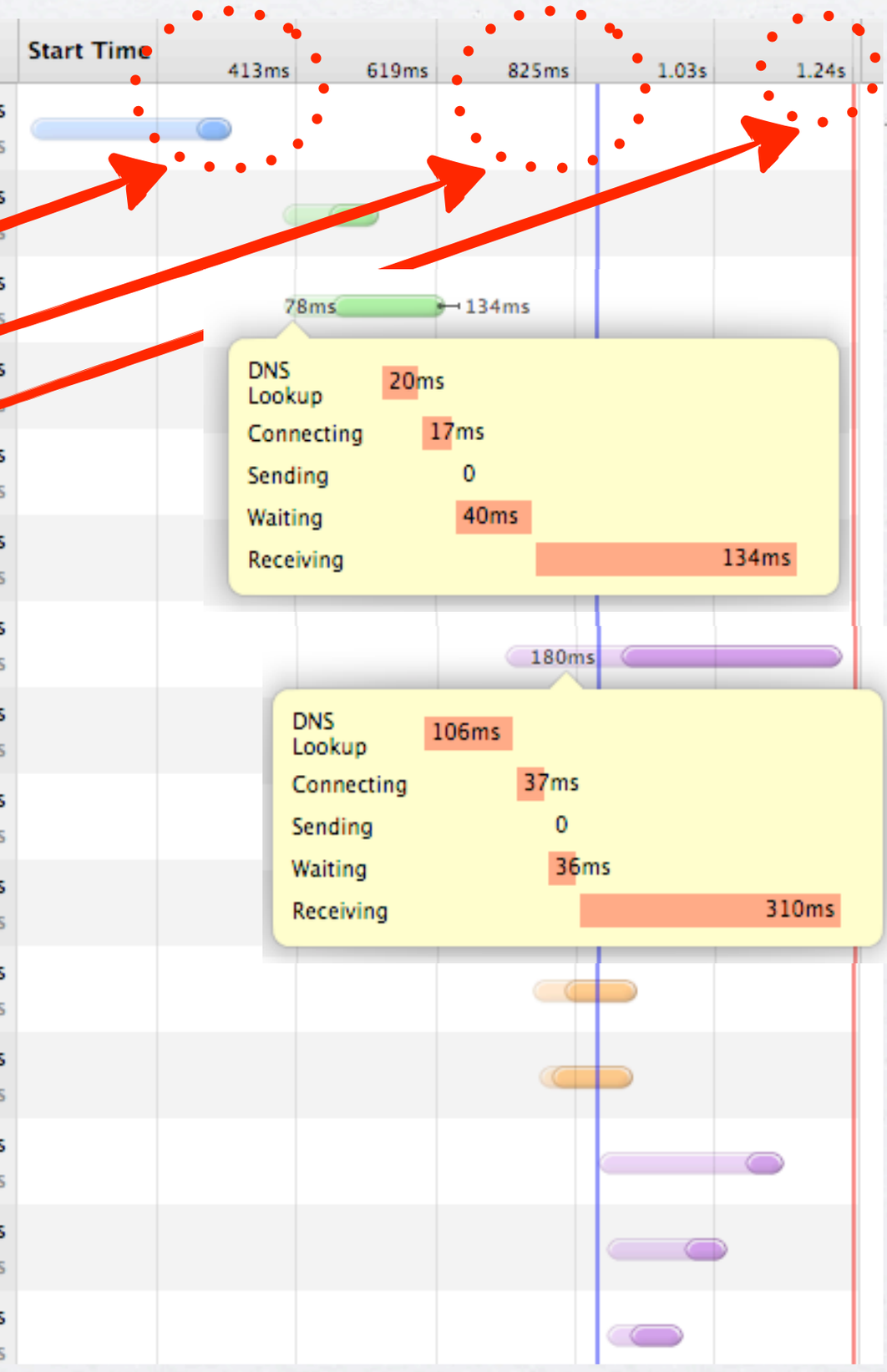
Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Start Time
http://www.facebook.com/	GET	200 OK	text/html	Other	12.40KB 37.19KB	275ms 261ms	
7MO6Ggjtnfq.css static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	text/css	http://www.faceb Parser	8.01KB 32.38KB	109ms 75ms	
4svSxh30U-4.css static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	text/css	http://www.faceb Parser	43.44KB 267.40KB	212ms 78ms	
yOXsARv5tTT.css static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	text/css	http://www.faceb Parser	37.09KB 93.44KB	198ms 79ms	
u8iA3kXb8Y1.css static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	text/css	http://www.faceb Parser	579B 103B	80ms 80ms	
HxfnMiPVR11.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	http://www.faceb Parser	17.49KB 50.88KB	161ms 78ms	
427933800596648_142098 dragon.ak.fbcdn.net/cfs-ak-a	GET	200 OK	image/…	http://www.faceb Parser	98.73KB 98.40KB	490ms 180ms	
LrYC68IJN6a.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	48.77KB 188.33KB	354ms 57ms	
PO4Yiug--P3.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	10.34KB 30.11KB	123ms 62ms	
-eoXHEHd51l.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	5.21KB 13.10KB	118ms 46ms	
zrd-GRF4xel.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	34.95KB 106.94KB	120ms 49ms	
NgnOOGEz7u1.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	35.70KB 114.09KB	108ms 21ms	
GsNJNwuI-UM.gif static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	917B 522B	248ms 230ms	
427933800596648_194983 dragon.ak.fbcdn.net/cfs-ak-s	GET	200 OK	image/…	http://www.faceb Parser	6.25KB 5.92KB	151ms 123ms	
2-E5vtt9MKh.png static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	19.71KB 19.33KB	78ms 38ms	

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Start Time 413ms 619ms 825ms 1.03s 1.24s
					12.40KB 37.19KB	275ms 261ms	
					8.01KB 32.38KB	109ms 75ms	
					43.44KB 267.40KB	212ms 78ms	
					37.09KB 93.44KB	198ms 79ms	
					579B 103B	80ms 80ms	
static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	Parser	17.49KB 50.88KB	161ms 78ms	
427933800596648_142098 dragon.ak.fbcdn.net/cfs-ak-a	GET	200 OK	image/…	http://www.faceb Parser	98.73KB 98.40KB	490ms 180ms	
LrYC68IJN6a.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	48.77KB 188.33KB	354ms 57ms	
PO4Yiug--P3.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	10.34KB 30.11KB	123ms 62ms	
-eoXHEHd51l.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	5.21KB 13.10KB	118ms 46ms	
zrd-GRF4xel.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	34.95KB 106.94KB	120ms 49ms	
NgnOOGEz7u1.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	35.70KB 114.09KB	108ms 21ms	
GsNJNwuI-UM.gif static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	917B 522B	248ms 230ms	
427933800596648_194983 dragon.ak.fbcdn.net/cfs-ak-s	GET	200 OK	image/…	http://www.faceb Parser	6.25KB 5.92KB	151ms 123ms	
2-E5vtt9MKh.png static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	19.71KB 19.33KB	78ms 38ms	

Backend is done in < 400ms, but loads only after 800ms and takes 1.2 seconds to fully appear for the user

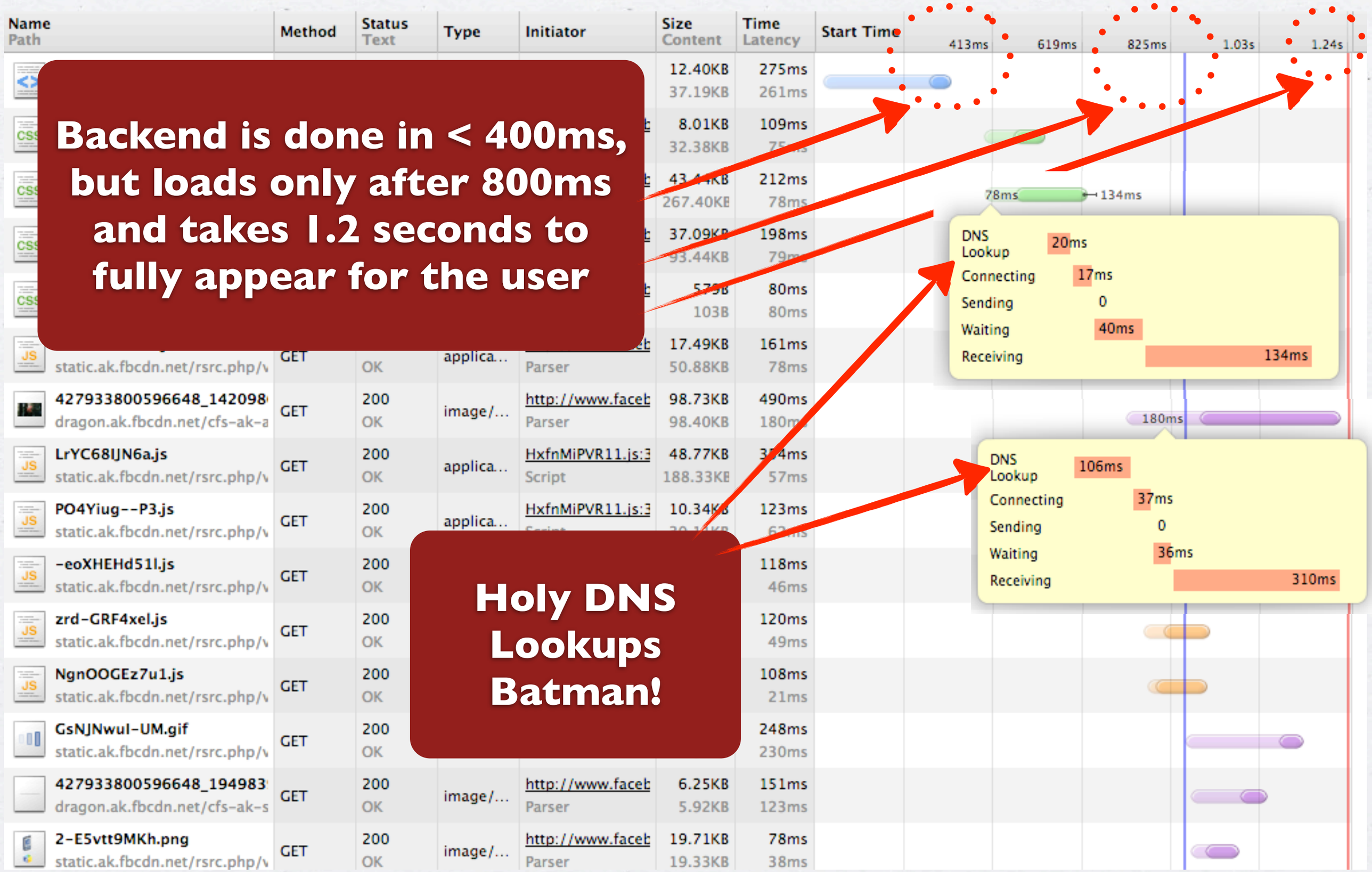
Backend is done in < 400ms, but loads only after 800ms and takes 1.2 seconds to fully appear for the user

Name / Path	Method	Status / Text	Type	Initiator	Size / Content	Time / Latency
					12.40KB / 37.19KB	275ms / 261ms
					8.01KB / 32.38KB	109ms / 75ms
					43.44KB / 267.40KB	212ms / 78ms
					37.09KB / 93.44KB	198ms / 79ms
					579B / 103B	80ms / 80ms
static.ak.fbcdn.net/rsrc.php/v	GET	OK	applica…	Parser	17.49KB / 50.88KB	161ms / 78ms
427933800596648_142098 dragon.ak.fbcdn.net/cfs-ak-a	GET	200 OK	image/…	http://www.faceb Parser	98.73KB / 98.40KB	490ms / 180ms
LrYC68IJN6a.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	48.77KB / 188.33KB	354ms / 57ms
PO4Yiug--P3.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	10.34KB / 30.11KB	123ms / 62ms
-eoXHEHd51l.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	5.21KB / 13.10KB	118ms / 46ms
zrd-GRF4xel.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	34.95KB / 106.94KB	120ms / 49ms
NgnOOGEz7u1.js static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	applica…	HxfnMiPVR11.js:3 Script	35.70KB / 114.09KB	108ms / 21ms
GsNJNwuI-UM.gif static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	917B / 522B	248ms / 230ms
427933800596648_194983 dragon.ak.fbcdn.net/cfs-ak-s	GET	200 OK	image/…	http://www.faceb Parser	6.25KB / 5.92KB	151ms / 123ms
2-E5vtt9MKh.png static.ak.fbcdn.net/rsrc.php/v	GET	200 OK	image/…	http://www.faceb Parser	19.71KB / 19.33KB	78ms / 38ms



Backend is done in < 400ms, but loads only after 800ms and takes 1.2 seconds to fully appear for the user

Holy DNS Lookups Batman!



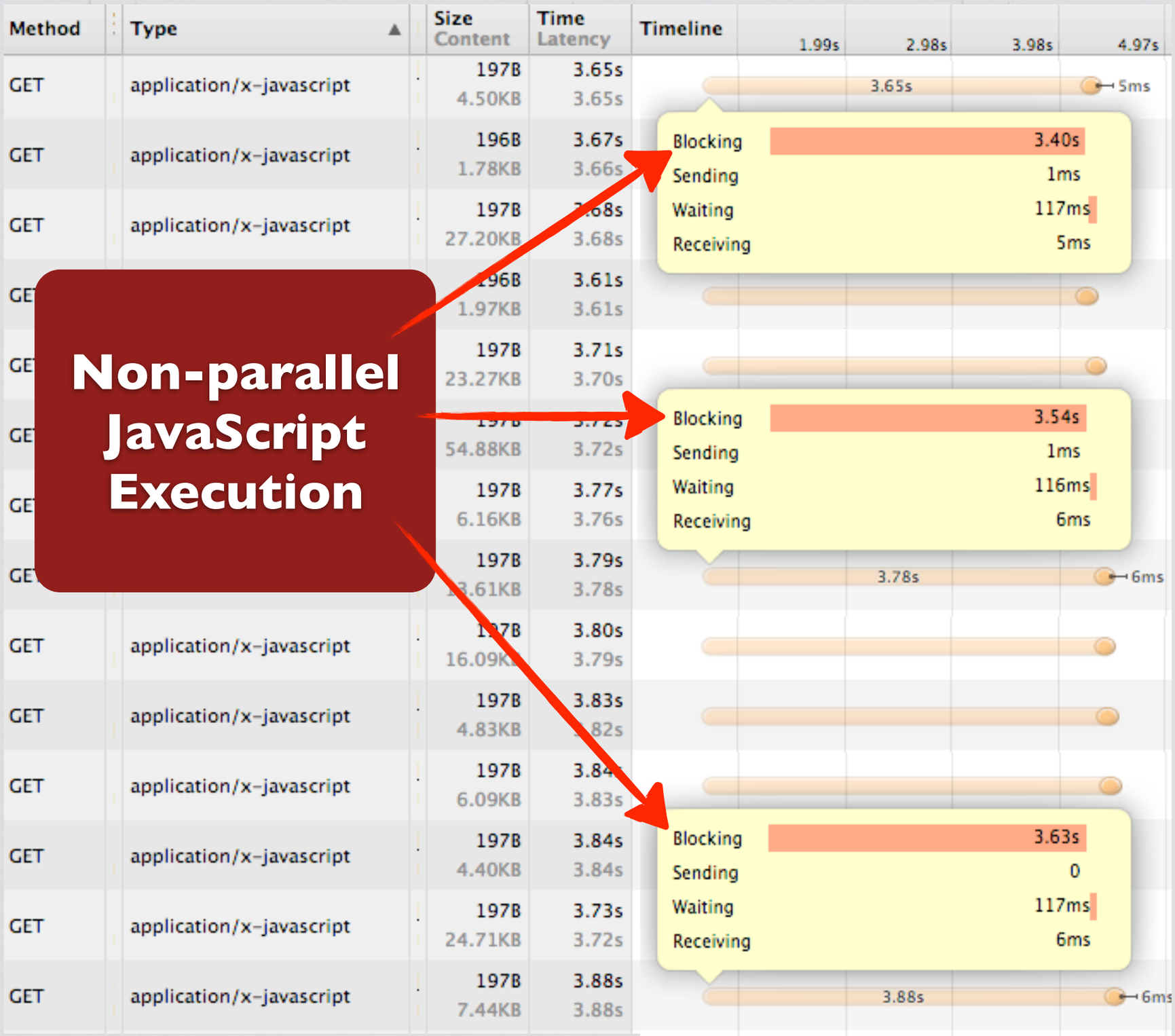
Method	Type	Size Content	Time Latency	Timeline 1.99s 2.98s 3.98s 4.97s
GET	application/x-javascript	197B 4.50KB	3.65s 3.65s	3.65s 5ms
GET	application/x-javascript	196B 1.78KB	3.67s 3.66s	
GET	application/x-javascript	197B 27.20KB	3.68s 3.68s	
GET	application/x-javascript	196B 1.97KB	3.61s 3.61s	
GET	application/x-javascript	197B 23.27KB	3.71s 3.70s	
GET	application/x-javascript	197B 54.88KB	3.72s 3.72s	
GET	application/x-javascript	197B 6.16KB	3.77s 3.76s	
GET	application/x-javascript	197B 13.61KB	3.79s 3.78s	3.78s 6ms
GET	application/x-javascript	197B 16.09KB	3.80s 3.79s	
GET	application/x-javascript	197B 4.83KB	3.83s 3.82s	
GET	application/x-javascript	197B 6.09KB	3.84s 3.83s	
GET	application/x-javascript	197B 4.40KB	3.84s 3.84s	
GET	application/x-javascript	197B 24.71KB	3.73s 3.72s	
GET	application/x-javascript	197B 7.44KB	3.88s 3.88s	3.88s 6ms

Blocking 3.40s
Sending 1ms
Waiting 117ms
Receiving 5ms

Blocking 3.54s
Sending 1ms
Waiting 116ms
Receiving 6ms

Blocking 3.63s
Sending 0
Waiting 117ms
Receiving 6ms

Non-parallel JavaScript Execution



Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline 1.56s 2.34s 3.12s 3.90s
paypal.com paypal.com	GET	301 Moved P	Pending	Other	154B 0B	491ms 490ms	
www.paypal.com	GET	200 OK	text/h...	https://paypal.c Redirect	8.67KB 29.44KB	2.07s 1.06s	
807f2587596b01fc7fff23 www.paypalobjects.com/eb	GET	200 OK	text/css	https://www.pa Parser	13.72KB 83.26KB	273ms 269ms	
40db0c074183048f12bf5 www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	6.21KB 11.61KB	216ms 215ms	
9e652a410df73ec249943 www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	12.91KB 36.35KB	231ms 230ms	
4abadeffed5ad75c4e261e www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	82.97KB 208.01KI	382ms 264ms	
pp_jscode_080706.js www.paypalobjects.com/js	GET	200 OK	applic...	https://www.pa Parser	26.49KB 56.15KB	317ms 265ms	
pa.js www.paypalobjects.com/pa	GET	200 OK	applic...	https://www.pa Parser	12.20KB 35.82KB	317ms 268ms	
adscout.php amch.questionmarket.com	GET	(pendi...	Pending	https://www.pa Parser	13B 0B	72ms 0.0 days	
data:image/png;base...	GET	Success	image...	http://amch.qu Redirect	0B 67B	4ms 0	
homepage-sell.png www.paypalobjects.com/w	GET	200 OK	image...	https://www.pa Parser	16.19KB 15.85KB	464ms 396ms	

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
paypal.com paypal.com	GET	301 Moved P	Pending	Other	154B 0B	491ms 490ms	490ms 1ms
www.paypal.com	GET	200 OK	text/h...	https://paypal.c Redirect	8.67KB 29.44KB	2.0 1.0	
807f2587596b01fc7fff23 www.paypalobjects.com/eb	GET	200 OK	text/css	https://www.pa Parser	13.72KB 83.26KB	273 269	
40db0c074183048f12bf5 www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	6.21KB 11.61KB	216 215ms	
9e652a410df73ec249943 www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	12.91KB 36.35KB	231ms 230ms	
4abadeffed5ad75c4e261e www.paypalobjects.com/eb	GET	200 OK	applic...	https://www.pa Parser	82.97KB 208.01KI	382ms 264ms	
pp_jscode_080706.js www.paypalobjects.com/js	GET	200 OK	applic...	https://www.pa Parser	26.49KB 56.15KB	317ms 265ms	
pa.js www.paypalobjects.com/pa	GET	200 OK	applic...	https://www.pa Parser	12.20KB 35.82KB	317ms 268ms	
adscout.php amch.questionmarket.com	GET	(pendi...	Pending	https://www.pa Parser	13B 0B	72ms 0.0 days	
data:image/png;base...	GET	Success	image...	http://amch.qu Redirect	0B 67B	4ms 0	
homepage-sell.png www.paypalobjects.com/w	GET	200 OK	image...	https://www.pa Parser	16.19KB 15.85KB	464ms 396ms	

1.56s 2.34s 3.12s 3.90s

Connecting 306ms
SSL 379ms
Sending 0
Waiting 183ms
Receiving 1ms

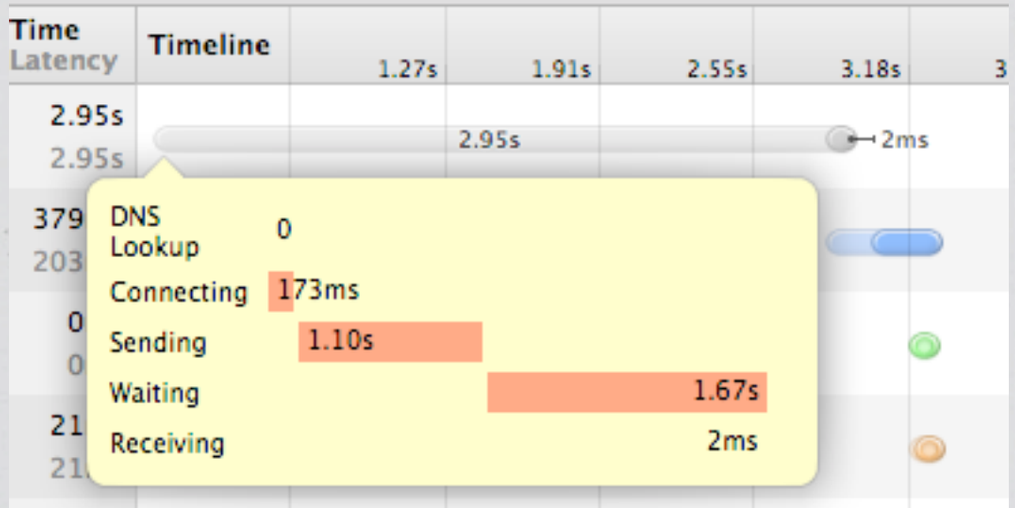
Name / Path	Method	Status / Text	Type	Initiator	Size / Content	Time / Latency	Timeline
paypal.com paypal.com	GET	301 Moved P	Pending	Other	154B 0B	491ms 490ms	
www.paypal.com	GET	200 OK	text/h...	https://paypal.c Redirect	8.67KB 29.44KB	2.07s	
807f2587596b01fc7fff23 www.paypalobjects.com/	GET	200	text/css	https://www.pa	13.72KB 83.26KB	273ms 269ms	
40db0c074183048 www.paypalobjects.				www.pa	6.21KB 11.61KB	216ms	
9e652a410df73ec www.paypalobjects.				www.pa	12.91KB 36.35KB	231ms 230ms	
4abadeffed5ad75c www.paypalobjects.				www.pa	82.97KB 208.01KI	382ms 264ms	
pp_jscode_080706 www.paypalobjects.				www.pa	26.49KB 56.15KB	317ms 265ms	
pa.js www.paypalobjects.com/pa	GET	200 OK	applic...	https://www.pa Parser	12.20KB 35.82KB	317ms 268ms	
adscout.php amch.questionmarket.com	GET	(pendi...	Pending	https://www.pa Parser	13B 0B	72ms 0.0 days	
data:image/png;base...	GET	Success	image...	http://amch.qu Redirect	0B 67B	4ms 0	
homepage-sell.png www.paypalobjects.com/w	GET	200 OK	image...	https://www.pa Parser	16.19KB 15.85KB	464ms 396ms	

1.56s 2.34s 3.12s 3.90s

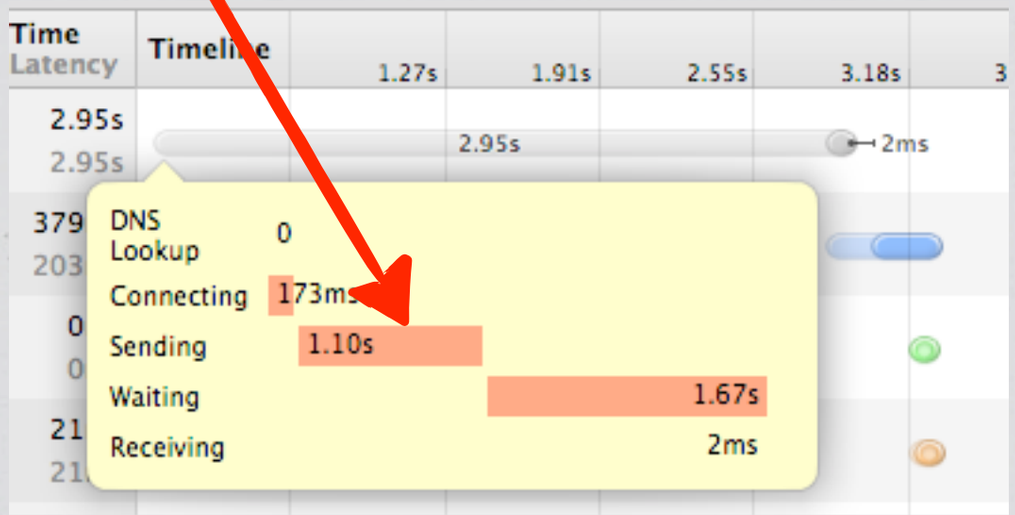
SSL 379ms

Connecting 190ms
SSL 119ms
Sending 1ms
Waiting 870ms
Receiving 1.00s

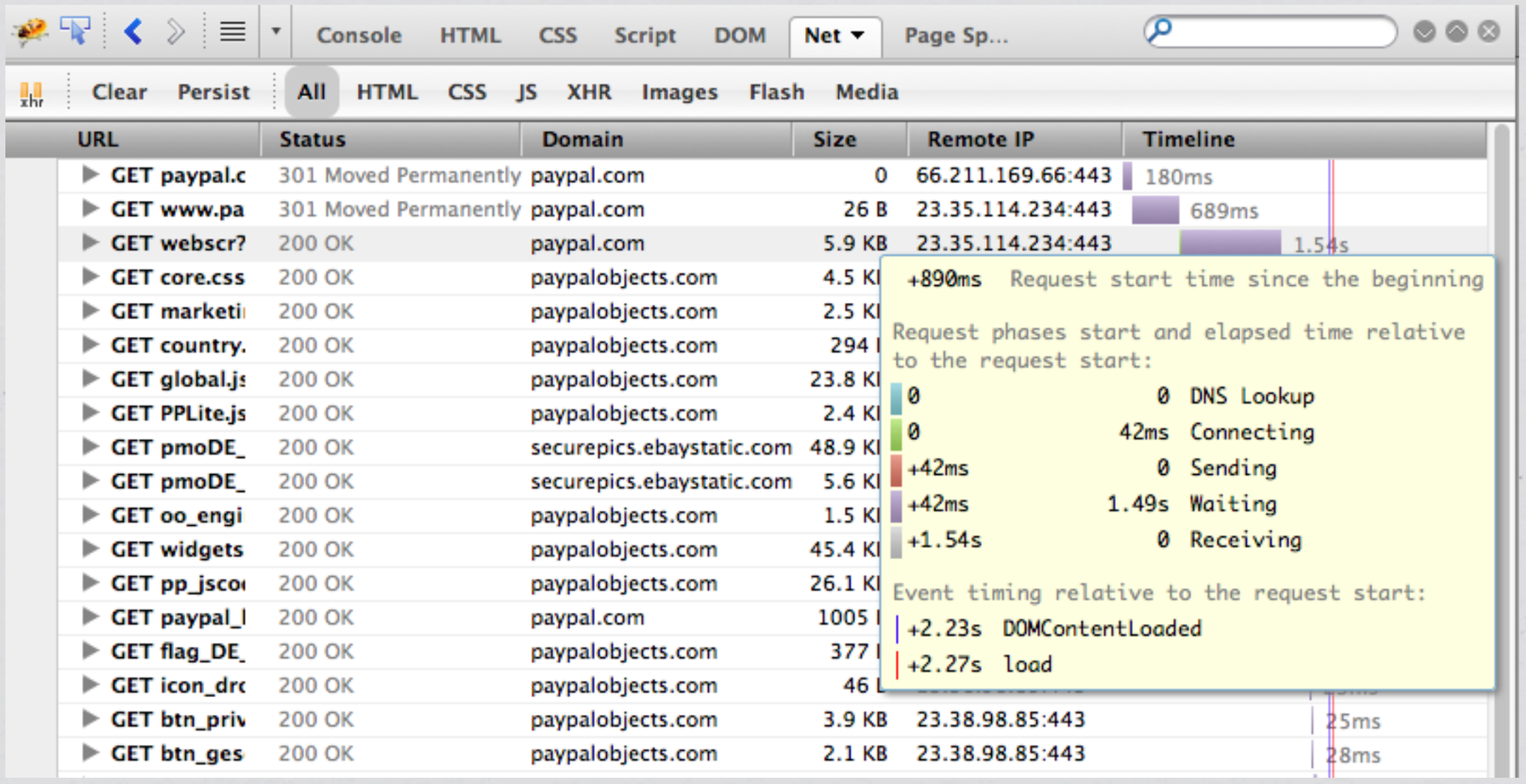
SSL Bottleneck worth 1/2 a second!


Time
Latency
Timeline
1.27s
1.91s
2.55s
3.18s
3
2.95s
2.95s
2.95s
2ms
379
203
0
0
21
21
DNS Lookup
0
Connecting
173ms
Sending
1.10s
Waiting
1.67s
Receiving
2ms

Many users have upstream speeds that are just a tad faster than dial-up, watch your cookies!



Firebug in Firefox



LOCAL TESTS ARE OFTEN NOT ENOUGH

Boomerang

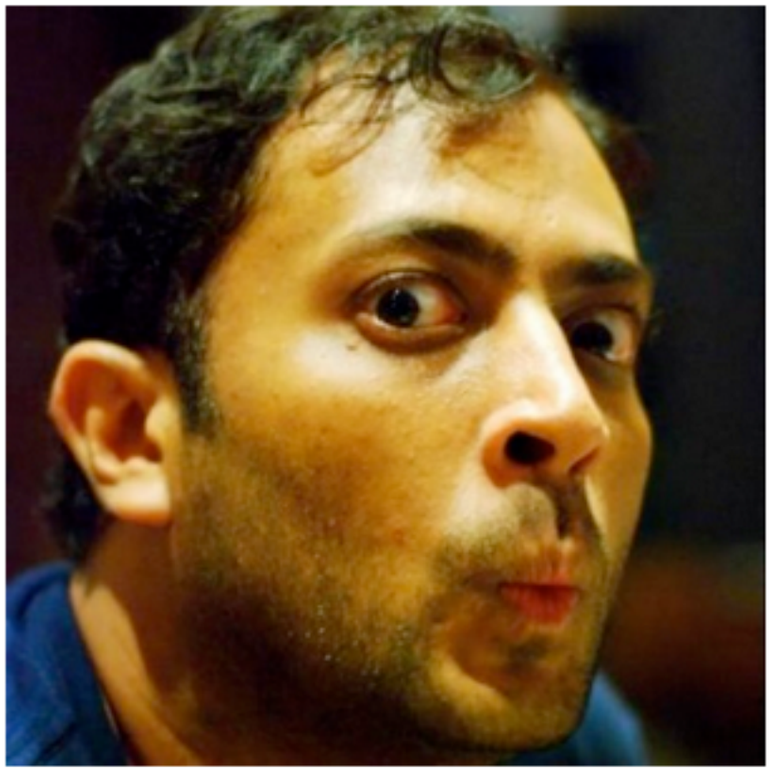
* Strangely enough, not designed by an Australian

* A JavaScript library designed to measure speed of a website as experienced by the user

* Gives you back data to analyze!

* Yeah, you can use it in production!

Boomerang



Philip Tellis

* Strangely enough, not designed by an Australian
* A JavaScript library designed to measure speed of a website as experienced by the user
* Gives you back data to analyze!
* Yeah, you can use it in production!

https://github.com/lognormal/boomerang

Initializing Boomerang

```
<script src="/boomerang.js" type="text/javascript">
</script>
<script type="text/javascript">
BOOMR.init({
        beacon_url: "http://yoursite.com/pixel.gif"
});
</script>
```

Initializing Boomerang

```
<script src="/boomerang.js" type="text/javascript">
</script>
<script type="text/javascript">
BOOMR.init({
        beacon_url: "http://yoursite.com/pixel.gif"
});
</script>
```

Tiny gif file, which will be requested along with logged data, passed via GET

Output *(from Apache access log)*

pixel.gif?v=0.9

&t_done=316

&u=http://test.com/sample.php

Boomerang Version

User Perceived Page load time in Milliseconds (1/1000 of a second)

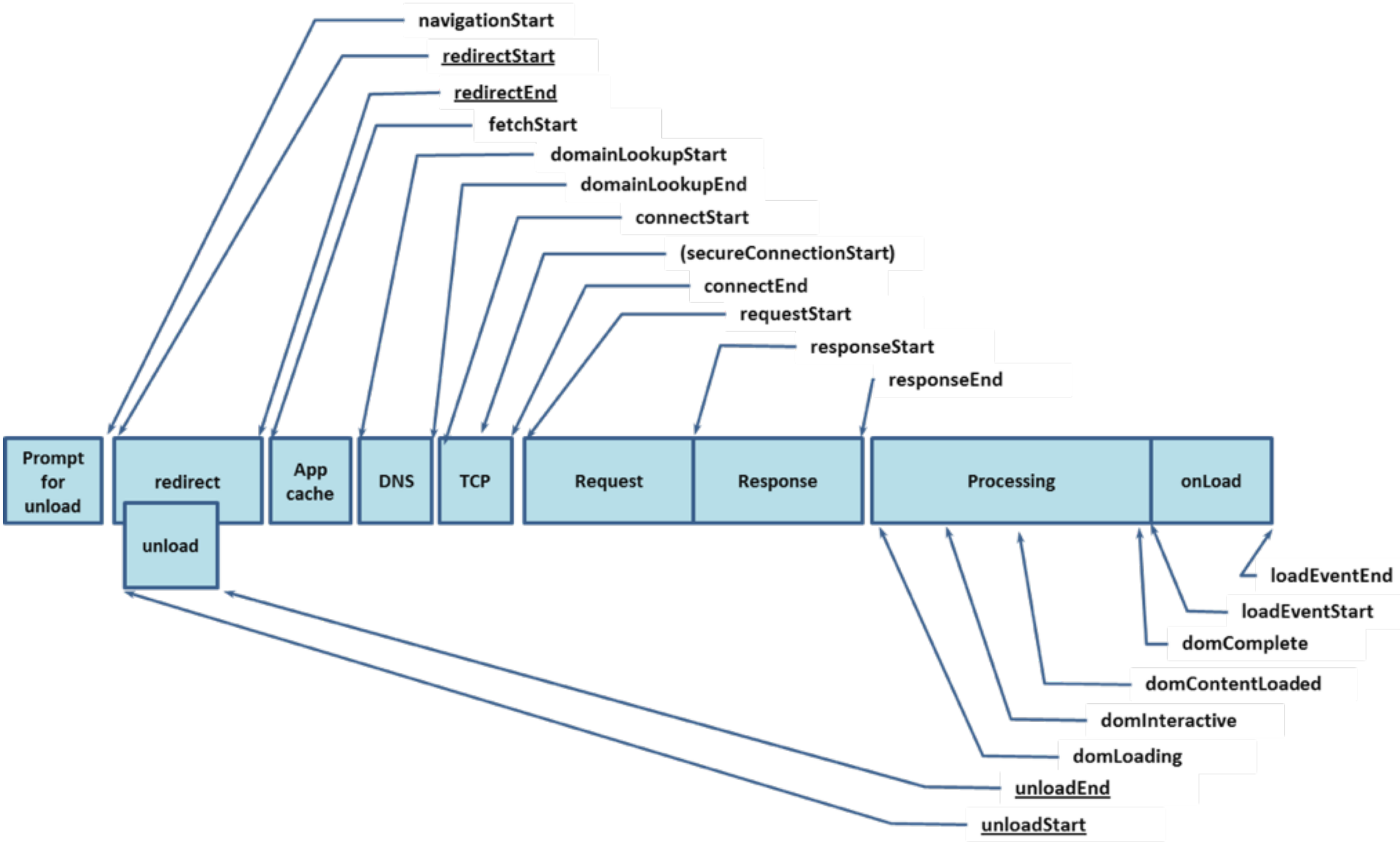
Requested URL (in url-encoded form)

W3C Navigation Timing API

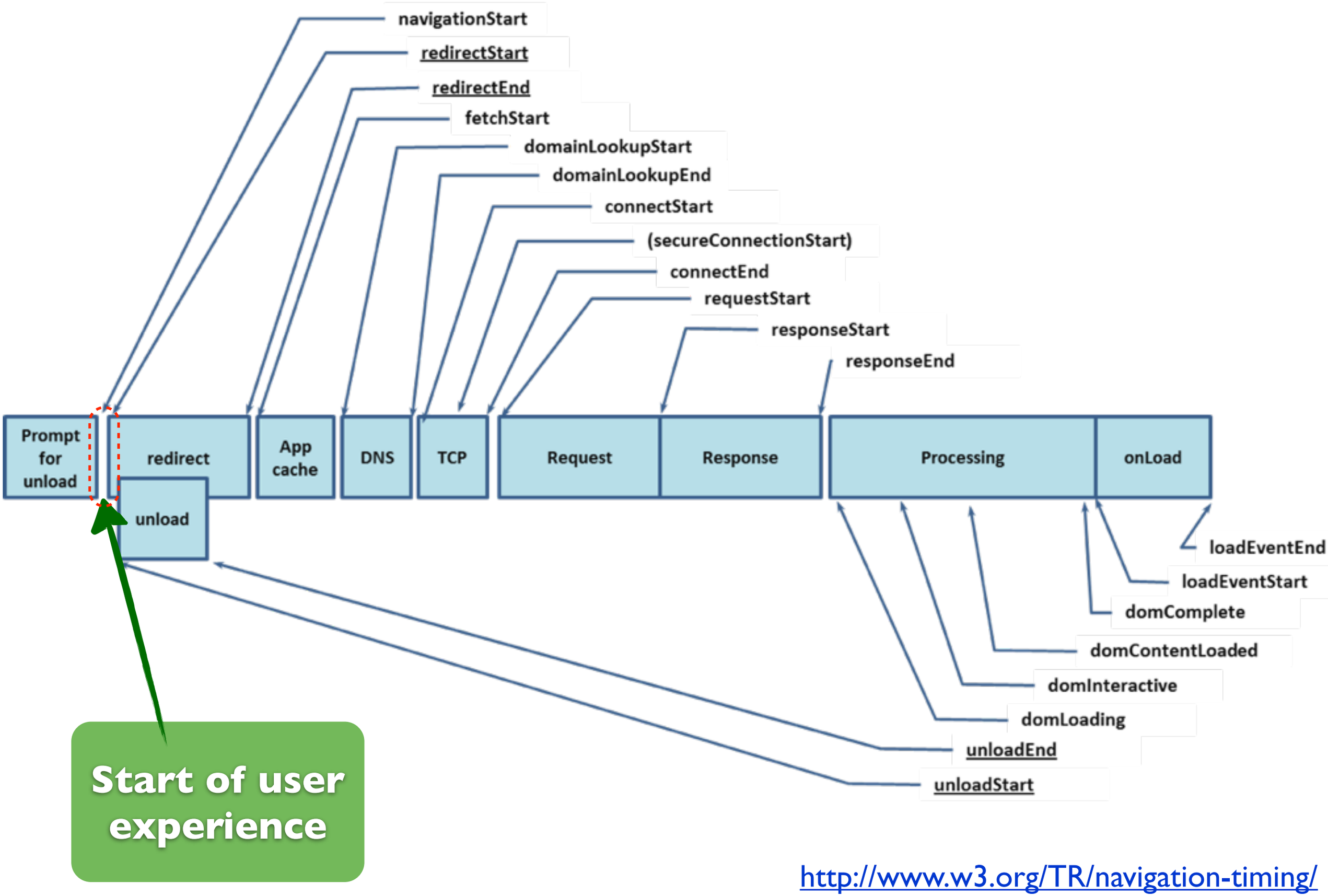
Access to web browsers' internal timing data, available for:

Chrome 6.0+ , IE 9.0+, Firefox 7+ * *

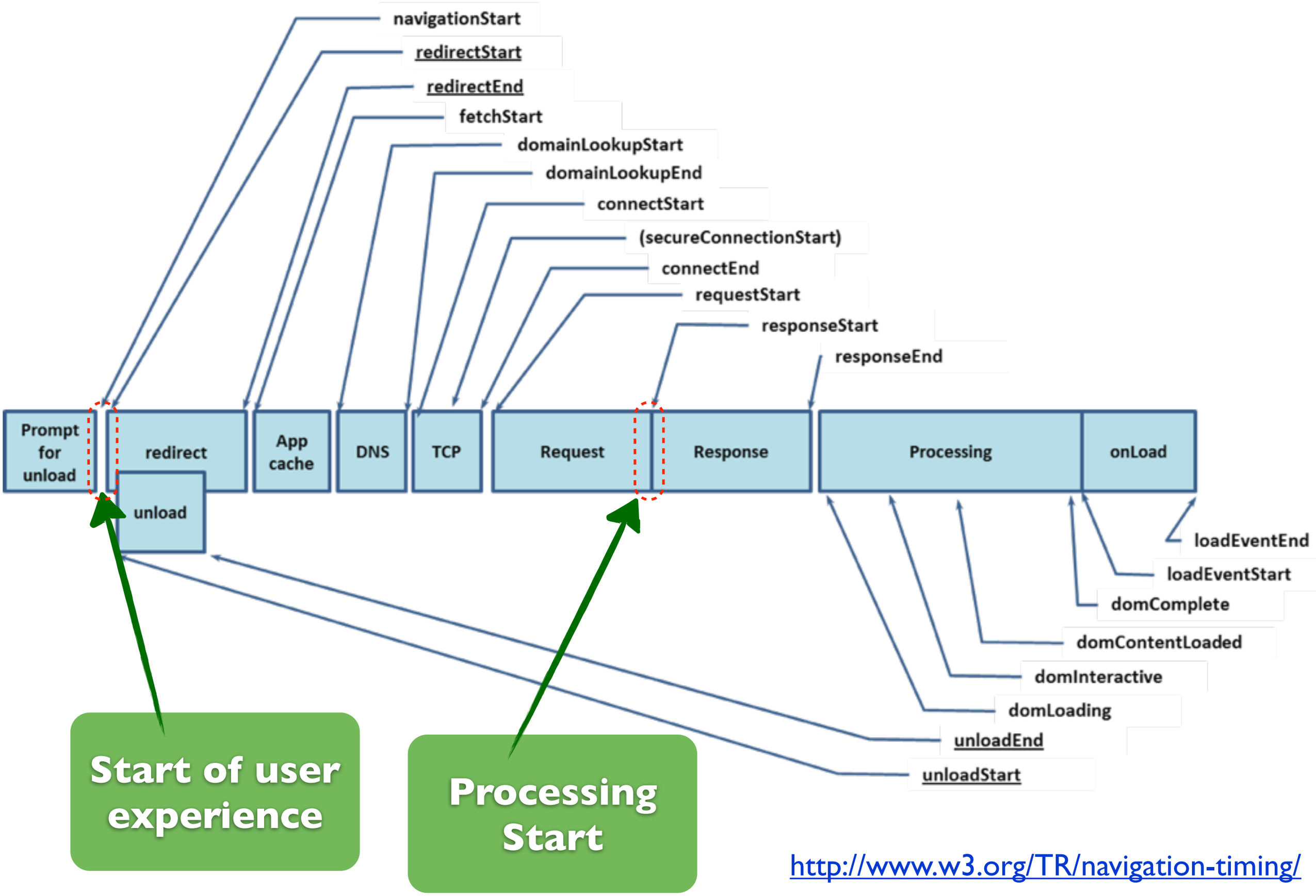
```
$ git clone https://github.com/lognormal/boomerang.git
$ cd boomerang
$ make PLUGINS=navtiming.js
$ scp boomerang.js you@server:webdir/
```



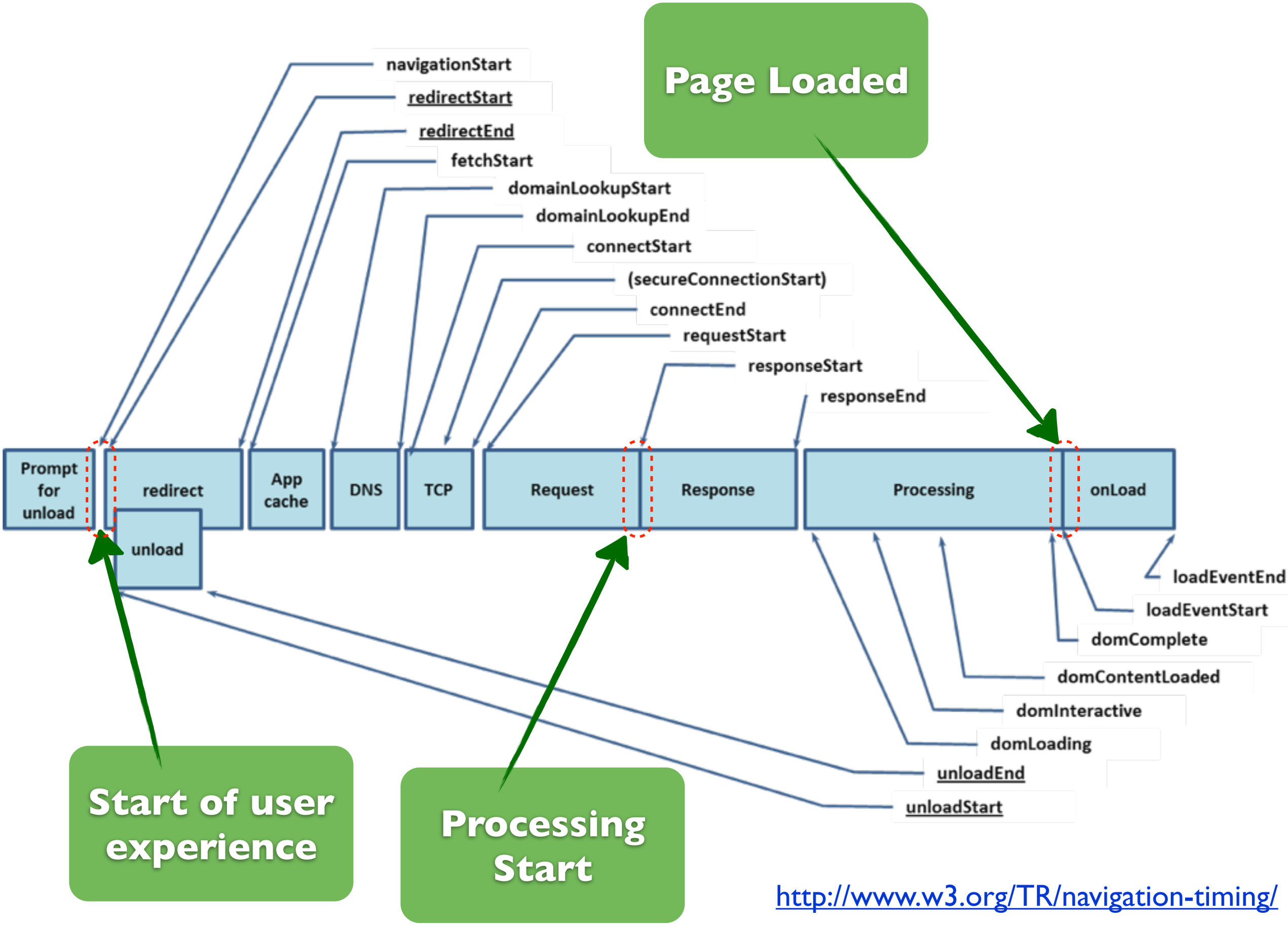
http://www.w3.org/TR/navigation-timing/



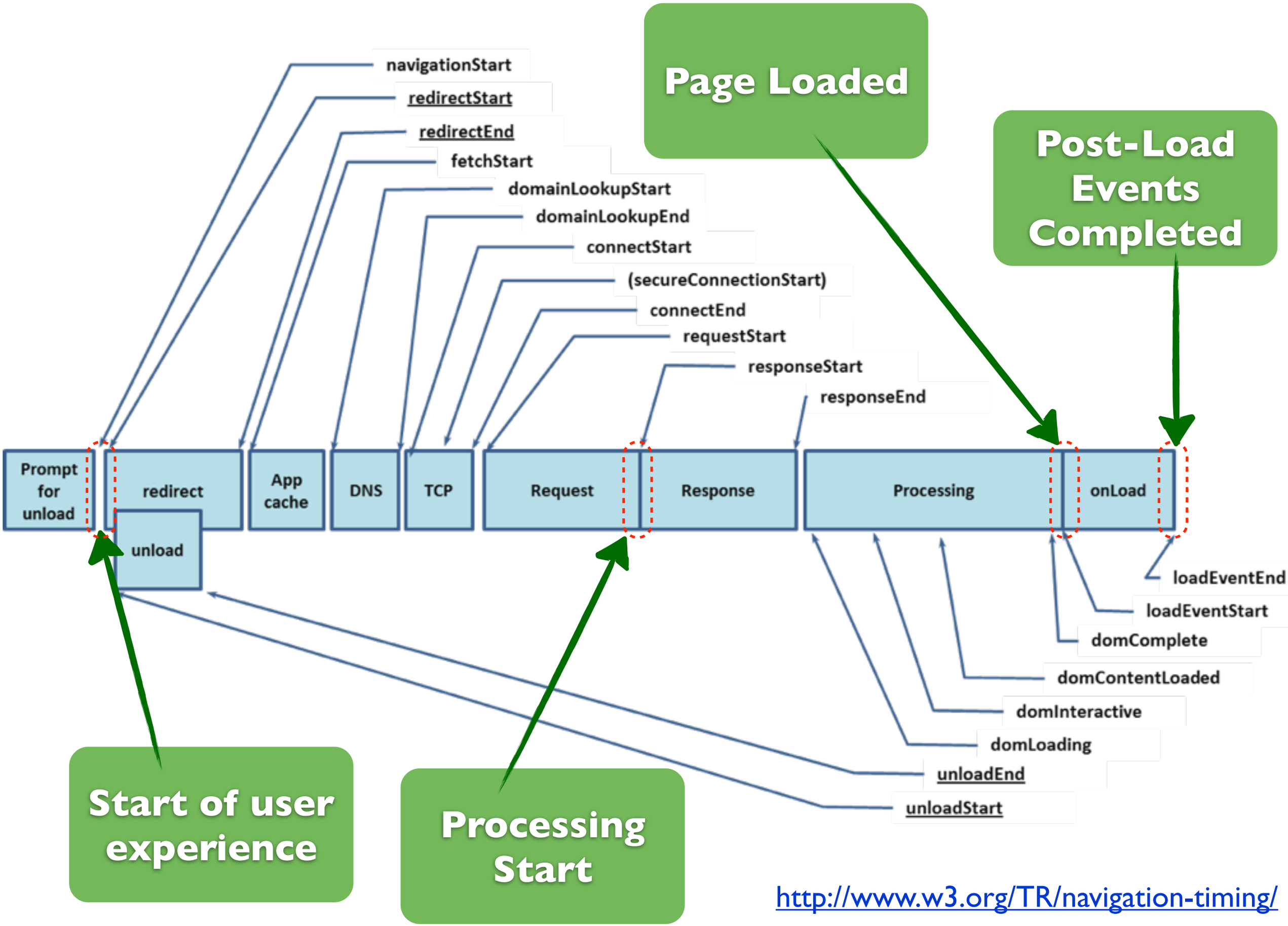
http://www.w3.org/TR/navigation-timing/



http://www.w3.org/TR/navigation-timing/



http://www.w3.org/TR/navigation-timing/



http://www.w3.org/TR/navigation-timing/

Output

Key	Data Source	Sample Value (in Msec)
nt_red_cnt	window.performance.navigation.redirectCount	0 *(counter, not a timer!)*
nt_nav_type	window.performance.navigation.type	1 *(counter, not a timer!)*
nt_nav_st	window.performance.timing.navigationStart	1350247373629
nt_red_st	window.performance.timing.redirectStart	0
nt_red_end	window.performance.timing.redirectEnd	0
nt_fet_st	window.performance.timing.fetchStart	1350247373629
nt_dns_st	window.performance.timing.domainLookupStart	1350247373629
nt_dns_end	window.performance.timing.domainLookupEnd	1350247373629
nt_con_st	window.performance.timing.connectStart	1350247373633
nt_con_end	window.performance.timing.connectEnd	1350247373734
nt_req_st	window.performance.timing.requestStart	1350247373734
nt_res_st	window.performance.timing.responseStart	1350247373854
nt_res_end	window.performance.timing.responseEnd	1350247373854
nt_domloading	window.performance.timing.domLoading	1350247373860
nt_domint	window.performance.timing.domInteractive	1350247374082
nt_domcontloaded_st	window.performance.timing.domContentLoadedStart	1350247374082
nt_domcontloaded_end	window.performance.timing.domContentLoadedEnd	1350247374083
nt_domcomp	window.performance.timing.domComplete	1350247374083
nt_load_st	window.performance.timing.loadEventStart	1350247374083
nt_load_end	window.performance.timing.loadEventEnd	1350247374084
nt_unload_st	window.performance.timing.unloadEventStart	1350247373855
nt_unload_end	window.performance.timing.unloadEventEnd	1350247373855

Can I do this myself?

```
<html>
<head>
<script type="text/javascript">
var PageTimerStart = new Date().getTime();
window.onload = function() {
     new Image().src = '/pixel.gif?page_load=' +
          (new Date().getTime() - PageTimerStart) +
          '&url=' + escape(window.location.href);
};
</script>
               /* rest of your code ... */
```

Output Processing

```
awk -F = '
BEGIN { sum=0; total=0; }
/page_load=/ { total++; sum += sprintf("%f\n",$2); }
END { print strftime("%Y-%m-%d"),": ",total,sum/total; }
 ' /var/log/apache2/access_log
```

Will return # of requests made
and their average speed

SEARCHING FOR WEB SERVER BOTTLENECKS

Using Apache Bench

```
Usage: /usr/sbin/ab [options] url
Options are:

-n  Number of requests to perform
-c  Number of multiple requests to make
-C  Add cookie, eg. 'Apache=1234. (repeatable)
-t  Seconds to max. wait for responses
-A  Add Basic WWW Authentication, the attributes
      are a colon separated username and password
-i  Use HEAD instead of GET
```

Understanding Output

```
Concurrency Level:      10
Time taken for tests:   18.054 seconds
Complete requests:      1000
Failed requests:        0
Write errors:           0
Total transferred:      494000 bytes
HTML transferred:       0 bytes
Requests per second:    55.39 [#/sec] (mean)
Time per request:       180.543 [ms] (mean)
Time per request:       18.054 [ms]
(mean, across all  concurrent requests)
Transfer rate:          26.72 [Kbytes/sec]
```

Understanding Output

Connection Times (Msec)

	min	mean[+/-sd]		median	max
Connect	5	7	0.9	6	12
Processing	76	174	38.0	173	320
Waiting	76	173	38.0	173	320
Total	82	180	38.1	179	327

Understanding the Output

```
Percentage of the requests served
within a certain time (Msec)
  50%    179
  66%    194
  75%    202
  80%    208
  90%    224
  95%    244
  98%    271
  99%    291
 100%    327 (longest request)
```

TESTING WEB SERVER FOR BOTTLENECKS

Static Tests

Request a static 10kb JavaScript, HTML and CSS files

Static Tests

Request a static 10kb JavaScript, HTML and CSS files

✓ Is the response size the same as file size? *(compression is missing)*

Static Tests

Request a static 10kb JavaScript, HTML and CSS files

- ✓ Is the response size the same as file size? *(compression is missing)*
- ✓ Do req/sec max out before the network does? *(if not, likely server overhead)*

Static Tests

Request a static 10kb JavaScript, HTML and CSS files

- ✓ Is the response size the same as file size? *(compression is missing)*
- ✓ Do req/sec max out before the network does?
  (if not, likely server overhead)
- ✓ Are there fluctuations in Connection Times?
  (concurrency issues, time to check for pre-processing directives)

Static Tests

Request a static 10kb JavaScript, HTML and CSS files

- ✓ Is the response size the same as file size? *(compression is missing)*
- ✓ Do req/sec max out before the network does?
  (if not, likely server overhead)
- ✓ Are there fluctuations in Connection Times?
  (concurrency issues, time to check for pre-processing directives)
- ✓ Does disk IO spike during test execution?
  (enable in memory caching of frequently requested static files)

Network IO

Request a static page that is roughly equivalent in size to your average output

Network IO

Request a static page that is roughly equivalent in size to your average output

- ✓ Does the network max-out before anticipated # of req/sec?
  (consider getting bigger network pipe or enabling compression)

Network IO

Request a static page that is roughly equivalent in size to your average output

- ✓ Does the network max-out before anticipated # of req/sec?
  (consider getting bigger network pipe or enabling compression)
- ✓ Do req/sec max out before anticipated # of req/sec?
  (check your web-server & TCP/IP kernel buffer sizes)

Scripting Language Overhead

Add a .php extension to your static page from the previous test and the test

Scripting Language Overhead

Add a .php extension to your static page from the previous test and the test

✓ Is there more than 2x speed difference?
(you may need an op-code cache)

Scripting Language Overhead

Add a .php extension to your static page from the previous test and the test

- ✓ Is there more than 2x speed difference?
  (you may need an op-code cache)
- ✓ What if op-code cache is already enabled?
  (consider different SAPI)

SSL Overhead

Request a static SSL page

✓ Is there > 100ms difference in loading speed between SSL and non-SSL Speed?

➡ *SSL SessionCache*

➡ *TCP/IP Congestion control settings in kernel*

➡ *GEO-Aware request routing*

Compression Overhead

Request a dynamically compressed page

✓ Is there more then 10-20ms speed difference? *(consider using reverse proxy to cache compressed output)*

SEARCHING FOR PHP BOTTLENECKS

XHProf

Light weight PHP profiler designed for use in production environment.

➡ Aggregates historical data runs

➡ Web UI

➡ Can be configured for In-Production sampling

http://pecl.php.net/xhprof

http://github.com/preinheimer/xhprof

Starting to Profile

```
;; Pre-pended to every PHP script (init)
auto_prepend_file = /xhprof/external/header.php

include_once __DIR__ . '/xhprof_lib/config.php');
include_once __DIR__ . '/xhprof_lib/utils/xhprof_lib.php';
include_once __DIR__ . '/xhprof_lib/utils/xhprof_runs.php';
xhprof_enable(XHPROF_FLAGS_CPU + XHPROF_FLAGS_MEMORY);

;; Appended to every PHP script (store)
auto_append_file = /xhprof/external/footer.php


$xhprof_data = xhprof_disable();
$xhprof_runs = new XHProfRuns_Default();
$xhprof_runs->save_run($xhprof_data, 'YourAppName', null, $_xhprof);
```

Starting to Profile

```
;; Pre-pended to every PHP script (init)
auto_prepend_file = /xhprof/external/header.php

include_once __DIR__ . '/xhprof_lib/config.php');
include_once __DIR__ . '/xhprof_lib/utils/xhprof_lib.php';
include_once __DIR__ . '/xhprof_lib/utils/xhprof_runs.php';
xhprof_enable(XHPROF_FLAGS_CPU + XHPROF_FLAGS_MEMORY);

;; Appended to every PHP script (store)
auto_append_file = /xhprof/external/footer.php


$xhprof_data = xhprof_disable();
$xhprof_runs = new XHProfRuns_Default();
$xhprof_runs->save_run($xhprof_data, 'YourAppName', null, $_xhprof);
```

Configuration for MySQL DB where to log data

Starting to Profile

```
;; Pre-pended to every PHP script (init)
auto_prepend_file = /xhprof/external/header.php

include_once __DIR__ . '/xhprof_lib/config.php');
include_once __DIR__ . '/xhprof_lib/utils/xhprof_lib.php';
include_once __DIR__ . '/xhprof_lib/utils/xhprof_runs.php';
xhprof_enable(XHPROF_FLAGS_CPU + XHPROF_FLAGS_MEMORY);

;; Appended to every PHP script (store)
auto_append_file = /xhprof/external/footer.php


$xhprof_data = xhprof_disable();
$xhprof_runs = new XHProfRuns_Default();
$xhprof_runs->save_run($xhprof_data, 'YourAppName', null, $_xhprof);
```

Configuration for MySQL DB where to log data

Not needed in recent version

OUTPUT

View Callgraph

Stat	Exact URL	Similar URLs
Count	38740	38740
Min Wall Time	21.1150 ms	21.1150 ms
Max Wall Time	44.0095 s	44.0095 s
Avg Wall Time	1.3765 s	1.3765 s
95% Wall Time	3.1266 s	3.1266 s
Display run Incl. Wall Time (microsec)	**584,869 microsecs**	
Min CPU Ticks	9.9980 ms	9.9980 ms
Max CPU Ticks	16.7735 s	16.7735 s
Avg CPU Ticks	614.3510 ms	614.3510 ms
95% CPU Ticks	893.8640 ms	893.8640 ms
Display run Incl. CPU (microsecs)	**553,917 microsecs**	
Min Peak Memory Usage	5,594,576 bytes	5,594,576 bytes
Max Peak Memory Usage	220,900,512 bytes	220,900,512 bytes
Avg Peak Memory Usage	172,595,582 bytes	172,595,582 bytes
95% Peak Memory Usage	217,540,480 bytes	217,540,480 bytes
Display run Incl. PeakMemUse (bytes)	**195,438,544 bytes**	
Number of Function Calls:	**999**	
Perform Delta:		Delta

View Callgraph

# of Historical Runs

Stat	Exact URL	Similar URLs
Count	38740	38740
Min Wall Time	21.1150 ms	21.1150 ms
Max Wall Time	44.0095 s	44.0095 s
Avg Wall Time	1.3765 s	1.3765 s
95% Wall Time	3.1266 s	3.1266 s
Display run Incl. Wall Time (microsec)	**584,869 microsecs**	
Min CPU Ticks	9.9980 ms	9.9980 ms
Max CPU Ticks	16.7735 s	16.7735 s
Avg CPU Ticks	614.3510 ms	614.3510 ms
95% CPU Ticks	893.8640 ms	893.8640 ms
Display run Incl. CPU (microsecs)	**553,917 microsecs**	
Min Peak Memory Usage	5,594,576 bytes	5,594,576 bytes
Max Peak Memory Usage	220,900,512 bytes	220,900,512 bytes
Avg Peak Memory Usage	172,595,582 bytes	172,595,582 bytes
95% Peak Memory Usage	217,540,480 bytes	217,540,480 bytes
Display run Incl. PeakMemUse (bytes)	**195,438,544 bytes**	
Number of Function Calls:	**999**	
Perform Delta:		Delta

View Callgraph

# of Historical Runs

Stat	Exact URL	Similar URLs
Count	38740	38740
Min Wall Time	21.1150 ms	21.1150 ms
Max Wall Time	44.0095 s	44.0095 s
Avg Wall Time	1.3765 s	1.3765 s
95% Wall Time	3.1266 s	3.1266 s
Display run Incl. Wall Time (microsec)	**584,869 microsecs**	
Min CPU Ticks	9.9980 ms	9.9980 ms
Max CPU Ticks	16.7735 s	16.7735 s
Avg CPU Ticks	614.3510 ms	614.3510 ms
95% CPU Ticks	893.8640 ms	893.8640 ms
Display run Incl. CPU (microsecs)	**553,917 microsecs**	
Min Peak Memory Usage	5,594,576 bytes	5,594,576 bytes
Max Peak Memory Usage	220,900,512 bytes	220,900,512 bytes
Avg Peak Memory Usage	172,595,582 bytes	172,595,582 bytes
95% Peak Memory Usage	217,540,480 bytes	217,540,480 bytes
Display run Incl. PeakMemUse (bytes)	**195,438,544 bytes**	
Number of Function Calls:	**999**	
Perform Delta:		Delta

Memory Usage (Max, Min, Avg, 95%)

View Callgraph

Stat	Exact URL	Similar URLs
Count	38740	38740
Min Wall Time	21.1150 ms	
Max Wall Time	44.0095 s	
Avg Wall Time	1.3765 s	
95% Wall Time	3.1266 s	
Display run Incl. Wall Time (microsec)	**584,869 microsecs**	
Min CPU Ticks	9.9980 ms	9.9980 ms
Max CPU Ticks	16.7735 s	16.7735 s
Avg CPU Ticks	614.3510 ms	614.3510 ms
95% CPU Ticks	893.8640 ms	893.8640 ms
Display run Incl. CPU (microsecs)	**553,917 microsecs**	
Min Peak Memory Usage	5,594,576 bytes	5,594,576 bytes
Max Peak Memory Usage	220,900,512 bytes	220,900,512 bytes
Avg Peak Memory Usage	172,595,582 bytes	172,595,582 bytes
95% Peak Memory Usage	217,540,480 bytes	217,540,480 bytes
Display run Incl. PeakMemUse (bytes)	**195,438,544 bytes**	
Number of Function Calls:	**999**	
Perform Delta:		Delta

# of Historical Runs

Execution Time (Max, Min, Avg, 95%)

Memory Usage (Max, Min, Avg, 95%)

View Callgraph

Stat	Exact URL	Similar URLs
Count	38740	38740
Min Wall Time	21.1150 ms	
Max Wall Time	44.0095 s	
Avg Wall Time	1.3765 s	
95% Wall Time	3.1266 s	
Display run Incl. Wall Time (microsec)	**584,869 microsecs**	
Min CPU Ticks	9.9980 ms	9.9980 ms
Max CPU Ticks	16.7735 s	16.7735 s
Avg CPU Ticks	614.3510 ms	614.3510 ms
95% CPU Ticks	893.8640 ms	893.8640 ms
Display run Incl. CPU (microsecs)	**553,917 microsecs**	
Min Peak Memory Usage	5,594,576 bytes	5,594,576 bytes
Max Peak Memory Usage	220,900,512 bytes	220,900,512 bytes
Avg Peak Memory Usage	172,595,582 bytes	172,595,582 bytes
95% Peak Memory Usage	217,540,480 bytes	217,540,480 bytes
Display run Incl. PeakMemUse (bytes)	**195,438,544 bytes**	
Number of Function Calls:	**999**	
Perform Delta:		Delta

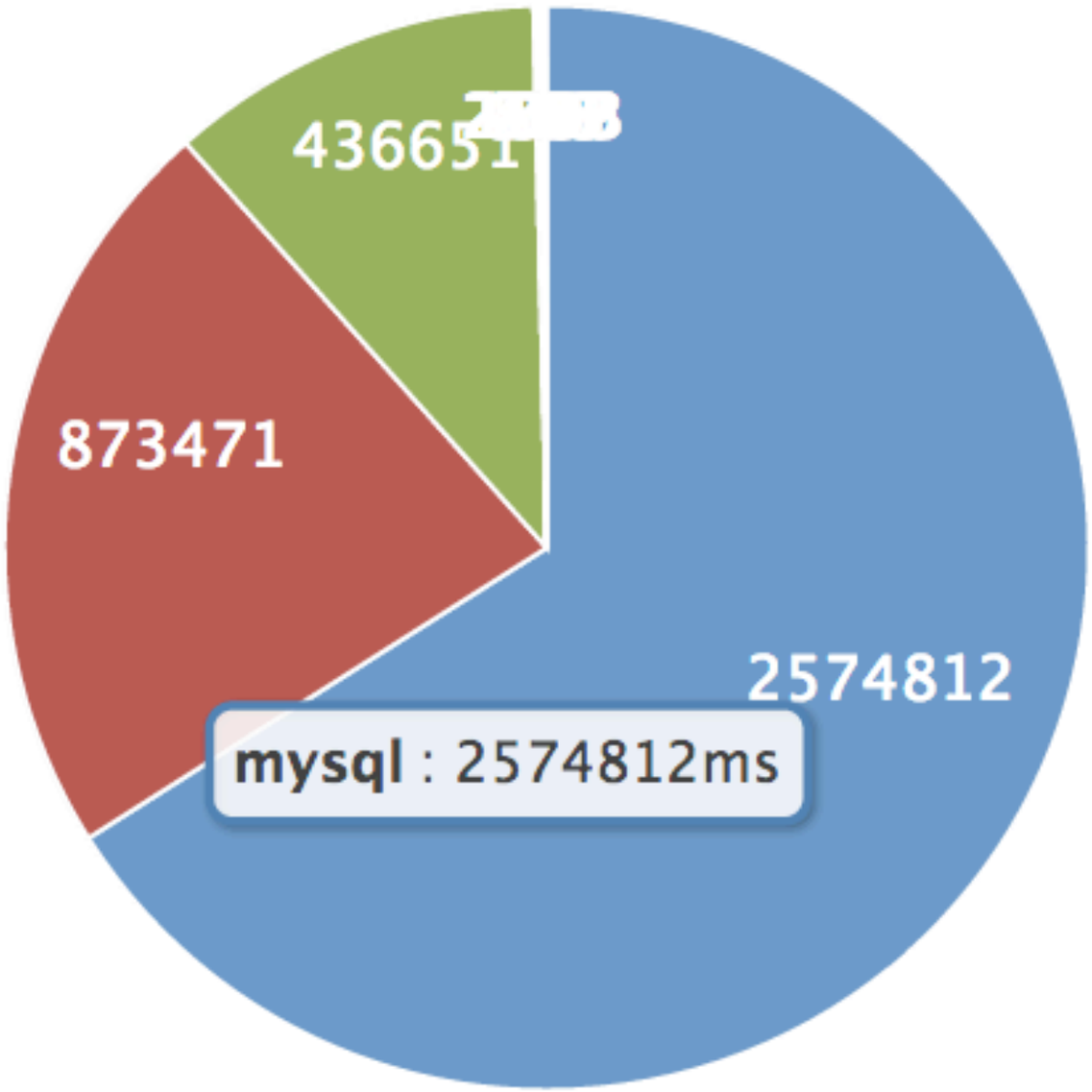
of Historical Runs

Execution Time (Max, Min, Avg, 95%)

Memory Usage (Max, Min, Avg, 95%)

CPU Time (Max, Min, Avg, 95%)

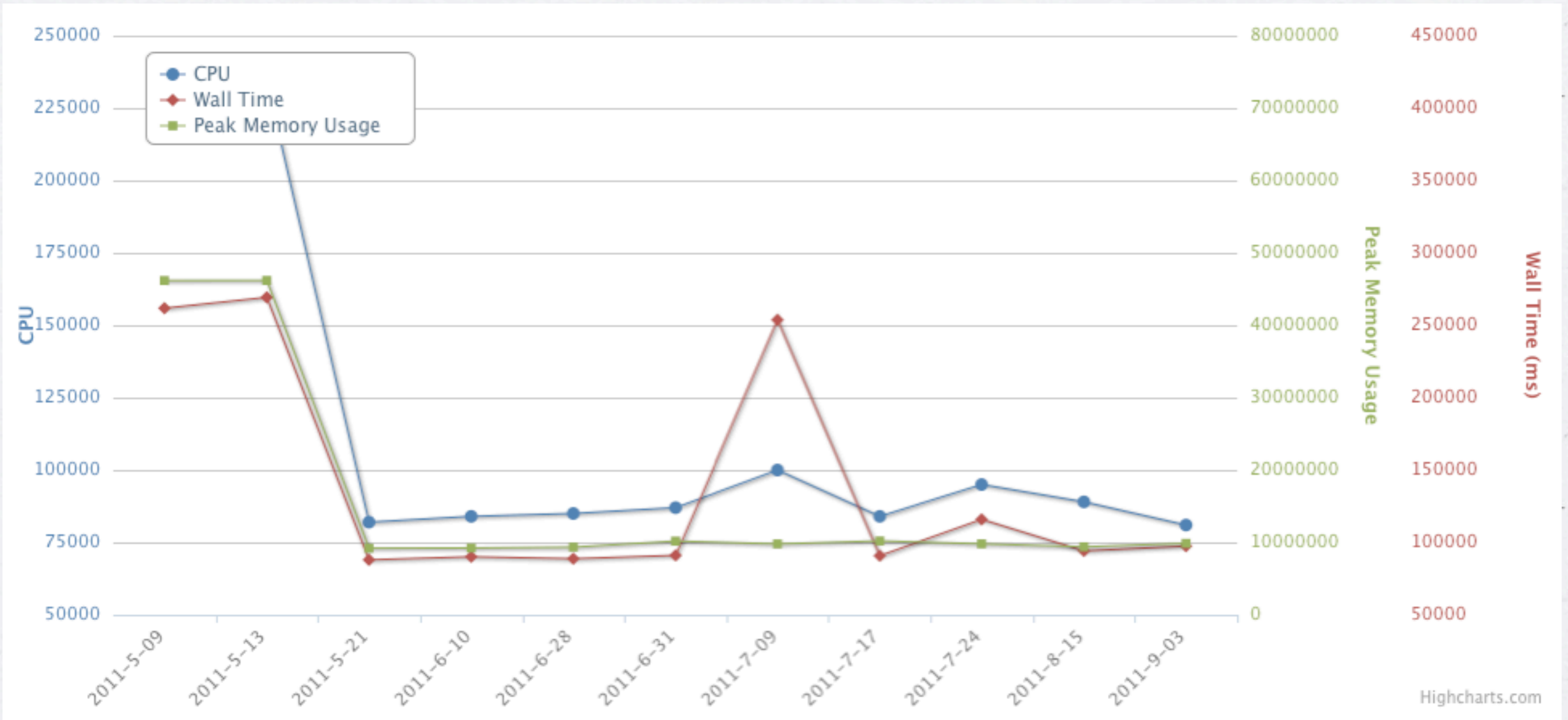
Expensive Calls by Exclusive Wall Time



- mysql
- xhprof_render_image
- stream_get_contents
- xhprof_generate_dot_...
- xhprof_parse_parent_...
- xhprof_compute_flat_...
- xhprof_compute_inclu...
- proc_open
- unserialize
- xhprof_get_children_...
- explode
- gzuncompress
- xhprof_get_content_b...
- XHProfRuns_Default::...
- Other

Boring per-call stats

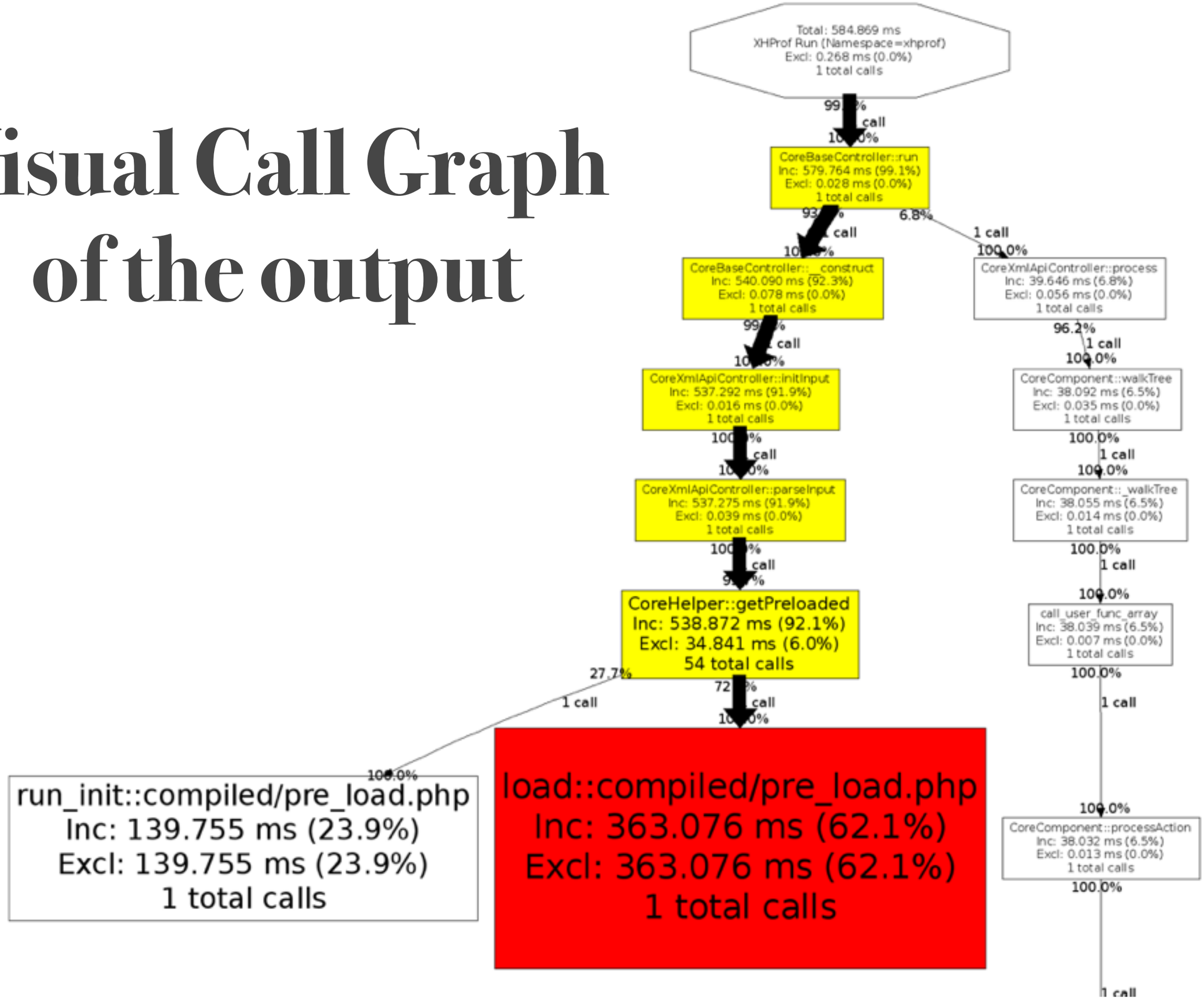
Function	Call Count	Wall Time	CPU	Memory Usage	Peak Memory Usage	Exclusive Wall Time	Exclusive CPU	Exclusive Memory Usage	Exclusive Peak Memory Usage
main()	1	584869	553917	98249760	195438544	268	0	4672	5072
CoreBaseController::run	1	579764	548917	93411472	190604032	28	0	4224	2880
CoreBaseController::__construct	1	540090	540918	92969872	190601152	78	0	4208	1944
CoreHelper::getPreloaded	54	538872	538918	92610840	190135488	34841	35995	-112592632	1936
CoreXmlApiController::initInput	1	537292	538918	92262488	189667824	16	0	352	0
CoreXmlApiController::parseInput	1	537275	538918	92260544	189667824	39	0	2304	0
load::compiled/pre_load.php	1	363076	363944	112610688	189667824	363076	363944	112610688	189667824
run_init::compiled/pre_load.php	1	139755	138979	92251736	0	139755	138979	92251736	0


CPU
Wall Time
Peak Memory Usage
250000
225000
200000
175000
150000
125000
100000
75000
50000
80000000
70000000
60000000
50000000
40000000
30000000
20000000
10000000
0
450000
400000
350000
300000
250000
200000
150000
100000
50000
CPU
Peak Memory Usage
Wall Time (ms)
2011-5-09
2011-5-13
2011-5-21
2011-6-10
2011-6-28
2011-6-31
2011-7-09
2011-7-17
2011-7-24
2011-8-15
2011-9-03
Highcharts.com

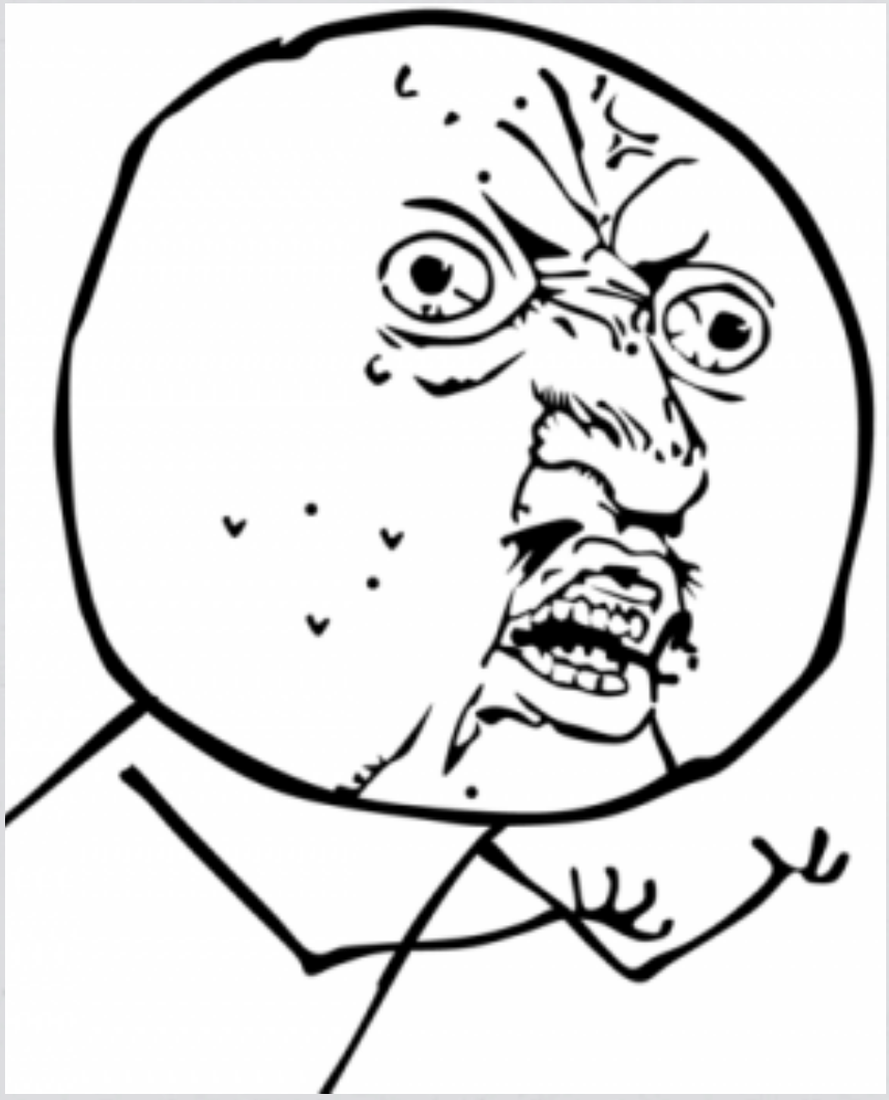

I DON'T ALWAYS CHECK IN MY CODE
BUT WHEN I DO, IT'S BROKEN
memegenerator.net


CPU
Wall Time
Peak Memory Usage
250000
225000
200000
175000
150000
125000
100000
75000
50000
80000000
70000000
60000000
50000000
40000000
30000000
20000000
10000000
0
450000
400000
350000
300000
250000
200000
150000
100000
50000
CPU
Peak Memory Usage
Wall Time (ms)
2011-5-09
2011-5-13
2011-5-21
2011-6-10
2011-6-28
2011-6-31
2011-7-09
2011-7-17
2011-7-24
2011-8-15
2011-9-03
Highcharts.com

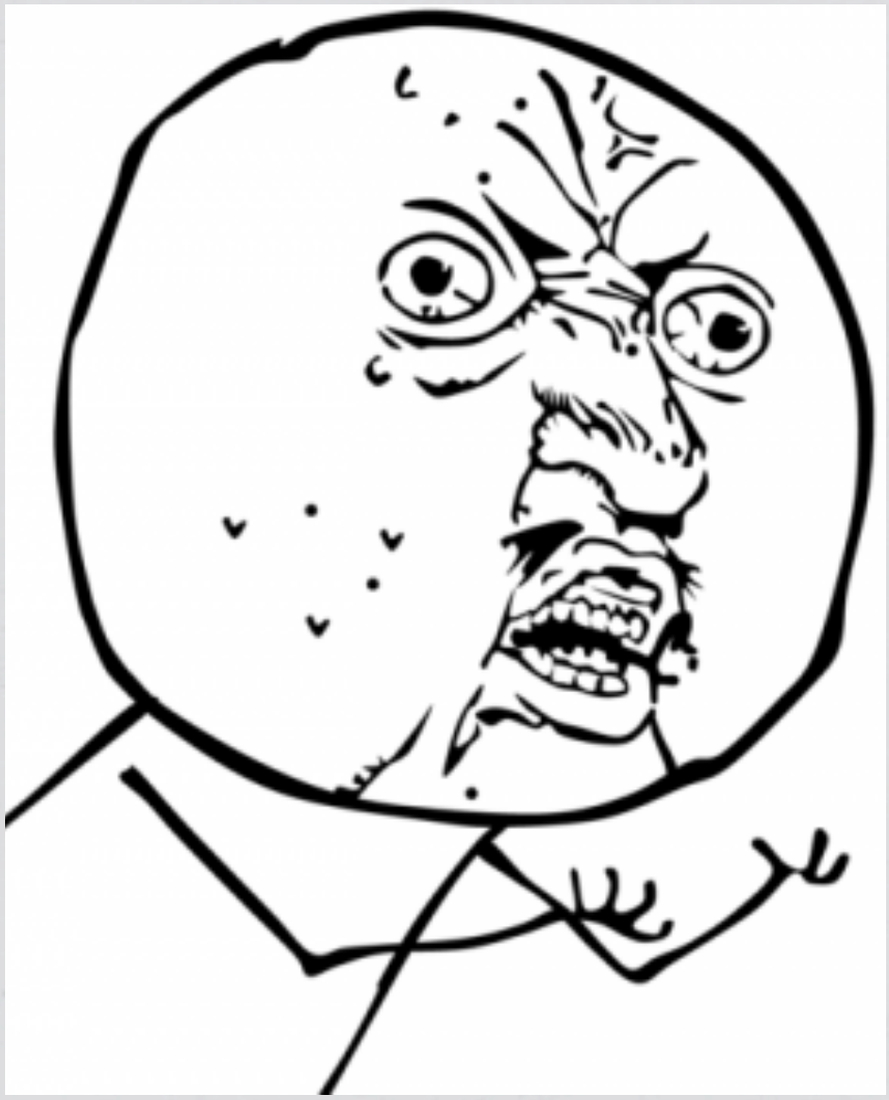
Visual Call Graph of the output



Profiling can cause a Bottleneck



Profiling can cause a Bottleneck



SOLUTION:

Targeted profiling of production environment:

- ➡ Statistical Sampling
- ➡ Pre-Identify Targets
- ➡ Replay-Log profiling

PRE-IDENTIFYING PROFILING TARGET

Apache:

```
LogFormat "{your usual stuff} %D %I %O" common
```

PRE-IDENTIFYING PROFILING TARGET

Apache:

```
LogFormat "{your usual stuff} %D %I %O" common
```

Processing Time in Msec

PRE-IDENTIFYING PROFILING TARGET

Apache:

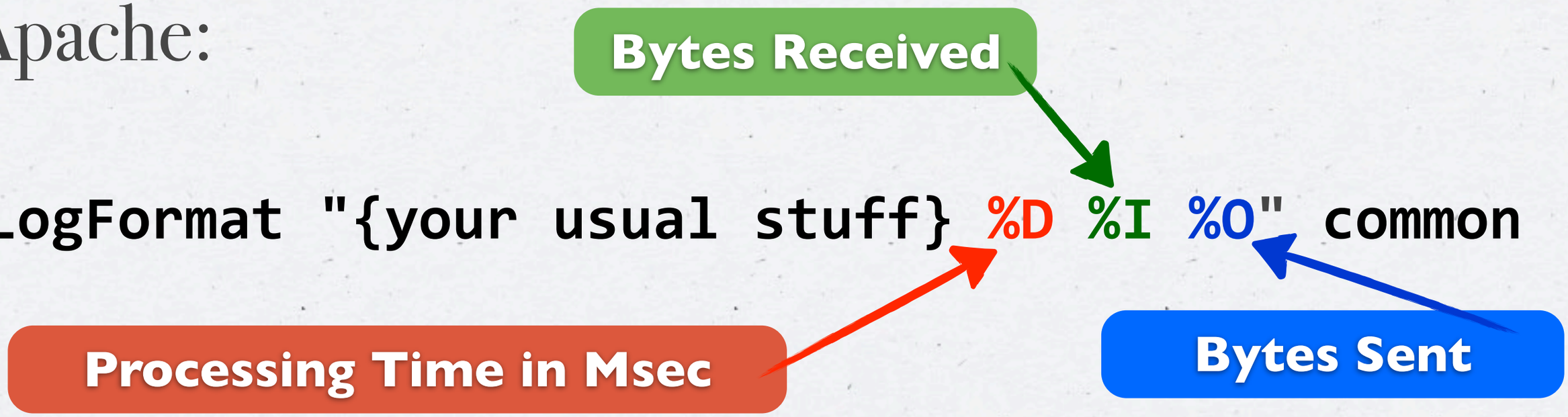
Bytes Received

```
LogFormat "{your usual stuff} %D %I %O" common
```

Processing Time in Msec

PRE-IDENTIFYING PROFILING TARGET

Apache:



PRE-IDENTIFYING PROFILING TARGET

Apache:

Bytes Received

```
LogFormat "{your usual stuff} %D %I %O" common
```

Processing Time in Msec

Bytes Sent

PRE-IDENTIFYING PROFILING TARGET

Nginx:

```
log_format combined ‘{your usual stuff}’
                    ‘$bytes_sent’
                    ‘$request_length’
                    ‘$request_time’
;
```

Apache:

```
LogFormat "{your usual stuff} %D %I %O" common
```

Bytes Received

Processing Time in Msec

Bytes Sent

PRE-IDENTIFYING PROFILING TARGET

Nginx:

```
log_format combined ‘{your usual stuff}’
        ‘$bytes_sent’
        ‘$request_length’
        ‘$request_time’
;
```

Processing Time in Msec *(Nginx 0.5.9+)*

Apache:

Bytes Received

```
LogFormat "{your usual stuff} %D %I %O" common
```

Processing Time in Msec

Bytes Sent

PRE-IDENTIFYING PROFILING TARGET

Nginx:

```
log_format combined ‘{your usual stuff}’
                    ‘$bytes_sent’
                    ‘$request_length’
                    ‘$request_time’
                    ;
```

Bytes Received

Processing Time in Msec *(Nginx 0.5.9+)*

Apache:

Bytes Received

```
LogFormat "{your usual stuff} %D %I %O" common
```

Processing Time in Msec

Bytes Sent

PRE-IDENTIFYING PROFILING TARGET

Nginx:

```
log_format combined ‘{your usual stuff}’
                    ‘$bytes_sent’
                    ‘$request_length’
                    ‘$request_time’
;
```

Bytes Sent

Bytes Received

Processing Time in Msec *(Nginx 0.5.9+)*

There are no 'Harmless' Errors

```php
function a() { if ($a) {} }
function b() { if (empty($a)) {} }

error_reporting(E_ERROR);

$s = microtime(1);
for ($i = 0; $i < 100000; $i++) a();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.0585

$s = microtime(1);
for ($i = 0; $i < 100000; $i++) b();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.0125
```

```php
function a() { if ($a) {} }
function b() { if (empty($a)) {} }

error_reporting(E_ALL|E_NOTICE);
ini_set("display_errors", 0);
ini_set("log_errors", 1);
ini_set("error_log", "/tmp/test");

$s = microtime(1);
for ($i = 0; $i < 1000; $i++) a();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.1503

$s = microtime(1);
for ($i = 0; $i < 1000; $i++) b();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.0002
```

```php
function a() { if ($a) {} }
function b() { if (empty($a)) {} }

error_reporting(E_ALL|E_NOTICE);
ini_set("display_errors", 0);
ini_set("log_errors", 1);
ini_set("error_log", "/tmp/test");

$s = microtime(1);
for ($i = 0; $i < 1000; $i++) a();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.1503

$s = microtime(1);
for ($i = 0; $i < 1000; $i++) b();
echo "Time Taken: " . (microtime(1) - $s) . "\n";
// Time Taken: 0.0002
```

Mere 1,000 executions causes 1/6 of a second overhead!

FIX YOUR ERRORS!

AS AN EXTRA INCENTIVE,

Windows

Windows crashed again. I am the Blue Screen of Death. No one hears your screams.

* Press any key to terminate the application.
* Press CTRL+ALT+DEL again to restart your computer. You will lose any usaved data in all applications.

Press any key to continue _

MAKE ALL ERRORS FATAL

* The BSOD is a trademark of the Microsoft Corporation.

CACHING BOTTLENECKS

Memcached Stats

```
STAT uptime 929318

...

STAT rusage_user_seconds 27

STAT rusage_user_microseconds 135874

STAT rusage_system_seconds 68

STAT rusage_system_microseconds 513584

...

STAT bytes_read 494845674

STAT bytes_written 2751269102
```

Memcached Stats

```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microsecon
STAT rusage_system_seconds
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```

bytes_read + bytes_written

uptime

Memcached Stats

```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microsecon
STAT rusage_system_seconds
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```

bytes_read + bytes_written

uptime

~= Bandwidth?

Memcached Stats

```
STAT uptime 929318

...

STAT rusage_user_seconds 27

STAT rusage_user_microseconds 135874

STAT rusage_system_seconds 68

STAT rusage_system_microseconds 513584

...

STAT bytes_read 494845674

STAT bytes_written 2751269102
```

Memcached Stats

```
STAT uptime 929318

...

STAT rusage_user_seconds 27

STAT rusage_user_microseconds 135874

STAT rusage_system_seconds 68

STAT rusage_system_microseconds 513584

...

STAT bytes_read 494845674

STAT bytes_written 2751269102
```

rusage_system / uptime
> 0.01 ?

Memcached Stats

```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microseconds 135874
STAT rusage_system_seconds 68
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```

rusage_system / uptime > 0.01 ?

WTF are you doing?

Memcached Stats

```
STAT uptime 929318

...

STAT rusage_user_seconds 27

STAT rusage_user_microseconds 135874

STAT rusage_system_seconds 68

STAT rusage_system_microseconds 513584

...

STAT bytes_read 494845674

STAT bytes_written 2751269102
```

Memcached Stats

```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microseconds 135874
STAT rusage_system_seconds 68
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```

rusage_user / rusage_system > 0.5 ?

Memcached Stats

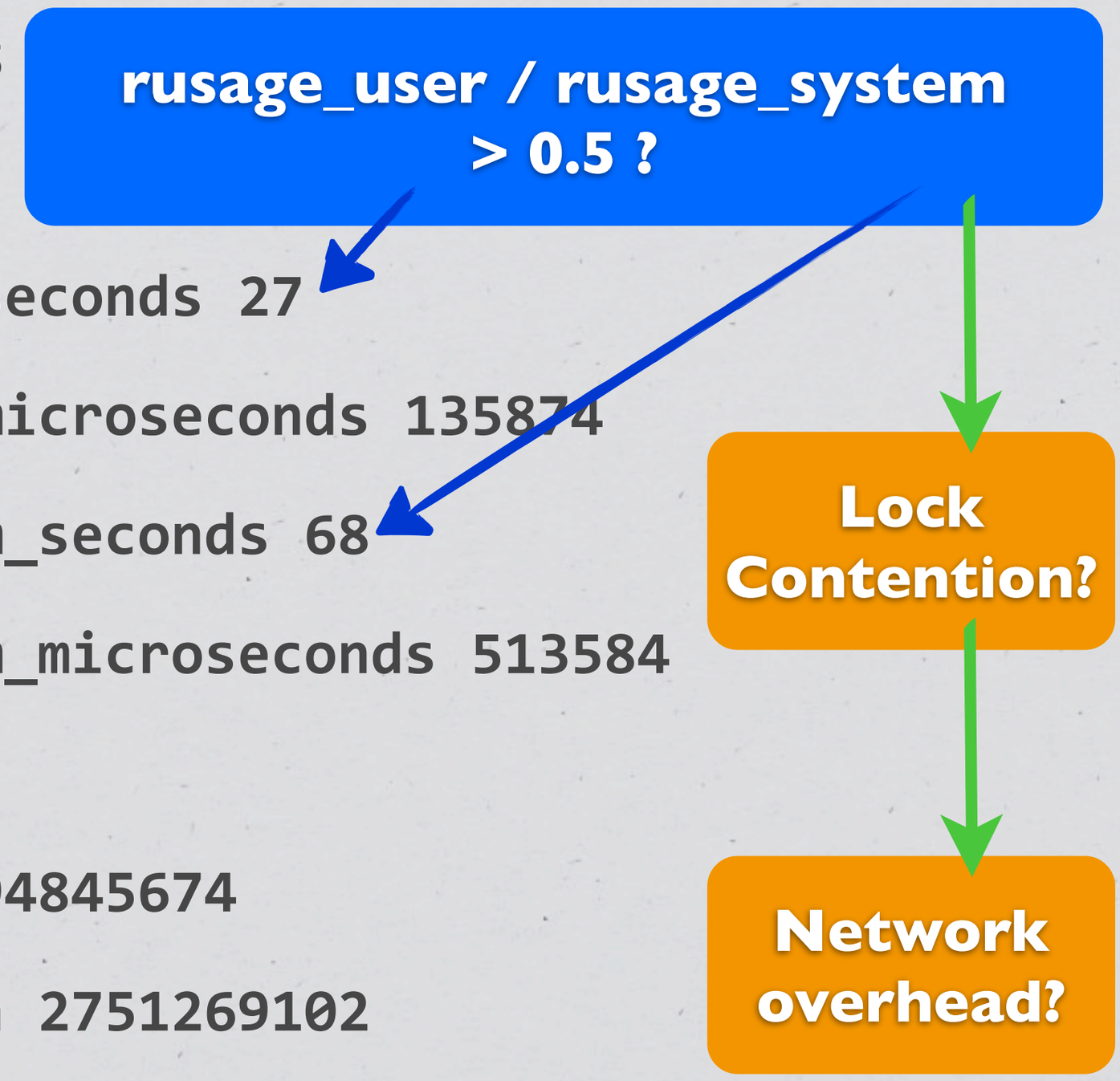
```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microseconds 135874
STAT rusage_system_seconds 68
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```

rusage_user / rusage_system > 0.5 ?

Lock Contention?

Memcached Stats

```
STAT uptime 929318
...
STAT rusage_user_seconds 27
STAT rusage_user_microseconds 135874
STAT rusage_system_seconds 68
STAT rusage_system_microseconds 513584
...
STAT bytes_read 494845674
STAT bytes_written 2751269102
```



Memcached Stats

```
STAT total_connections 111098
...
STAT cmd_set 156043
STAT get_hits 1356088
STAT get_misses 128824
STAT evictions 0
```

Memcached Stats

Have you heard of Persistent Connections?

```
STAT total_connections 111098
...
STAT cmd_set 156043
STAT get_hits 1356088
STAT get_misses 128824
STAT evictions 0
```

Memcached Stats

Have you heard of Persistent Connections?

get_misses / get_hits > 0.1 ?

```
STAT total_connections 111098

...

STAT cmd_set 156043

STAT get_hits 1356088

STAT get_misses 128824

STAT evictions 0
```

Memcached Stats

Have you heard of Persistent Connections?

```
STAT total_connections 111098

...

STAT cmd_set 156043

STAT get_hits 1356088

STAT get_misses 128824

STAT evictions 0
```

get_misses / get_hits > 0.1 ?

Evaluate your cache usage...

Memcached Stats

Have you heard of Persistent Connections?

```
STAT total_connections 111098

...

STAT cmd_set 156043

STAT get_hits 1356088

STAT get_misses 128824

STAT evictions 0
```

get_misses / get_hits > 0.1 ?

Evaluate your cache usage...

evictions / cmd_set > 0.1 ?

Memcached Stats

Have you heard of Persistent Connections?

```
STAT total_connections 111098

...

STAT cmd_set 156043

STAT get_hits 1356088

STAT get_misses 128824

STAT evictions 0
```

get_misses / get_hits > 0.1 ?

Evaluate your cache usage...

evictions / cmd_set > 0.1 ?

Give cache more RAM

DATABASE BOTTLENECKS

AVOID ACCIDENTS

We have gone

1

days since last

‘drop database `production``’

KEEP YOUR COWORKERS SAFE

Continuous Monitoring for MySQL

Make a point to log & analyze “bad” things!

```
--log-slow-queries=/var/log/TheBadBda.log

long_query_time = 1
```

Continuous Monitoring for MySQL

Make a point to log & analyze “bad” things!

```
--log-slow-queries=/var/log/TheBadBda.log

long_query_time = 1
```

Continuous Monitoring for MySQL

Make a point to log & analyze “bad” things!

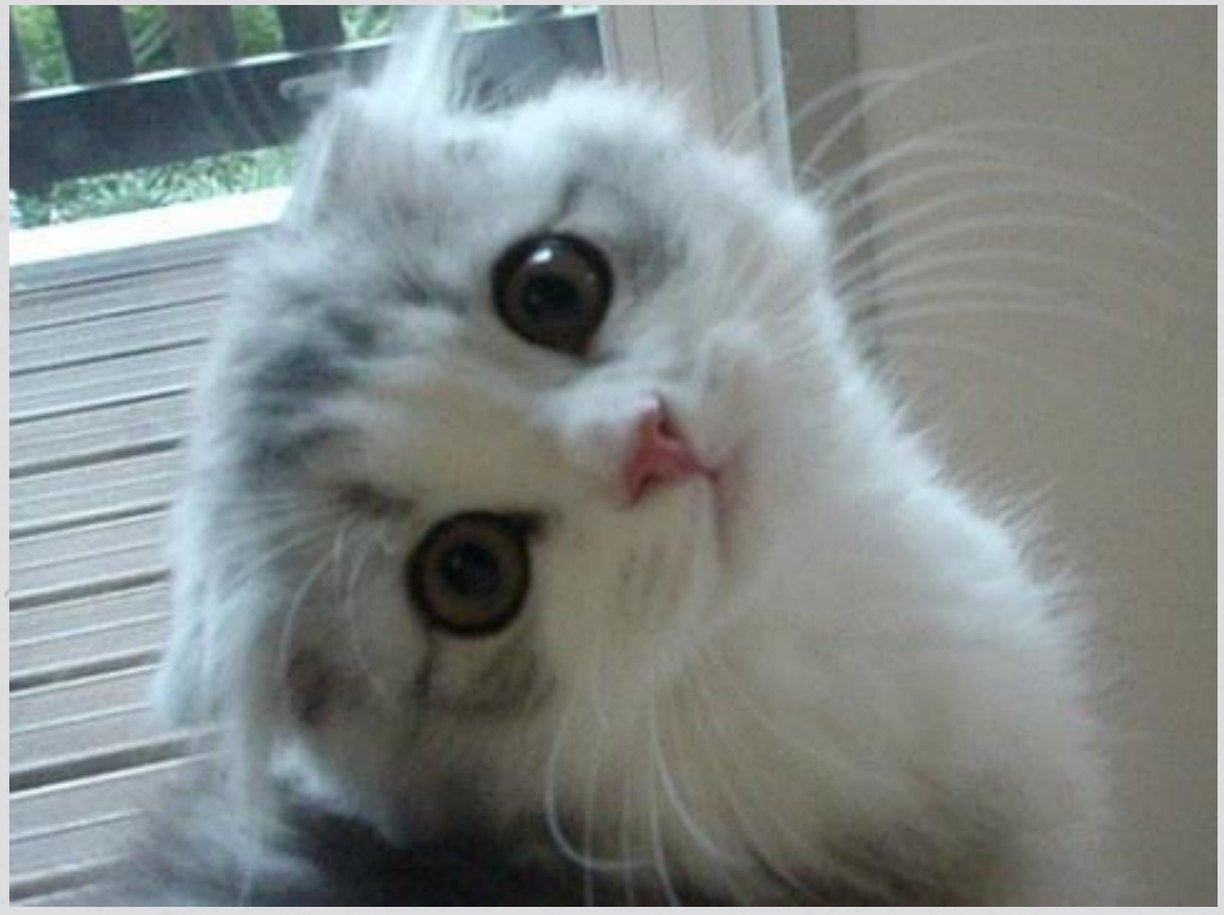
```
--log-slow-queries=/var/log/TheBadBda.log

long_query_time = 1
```

Because 10 second default is insane!

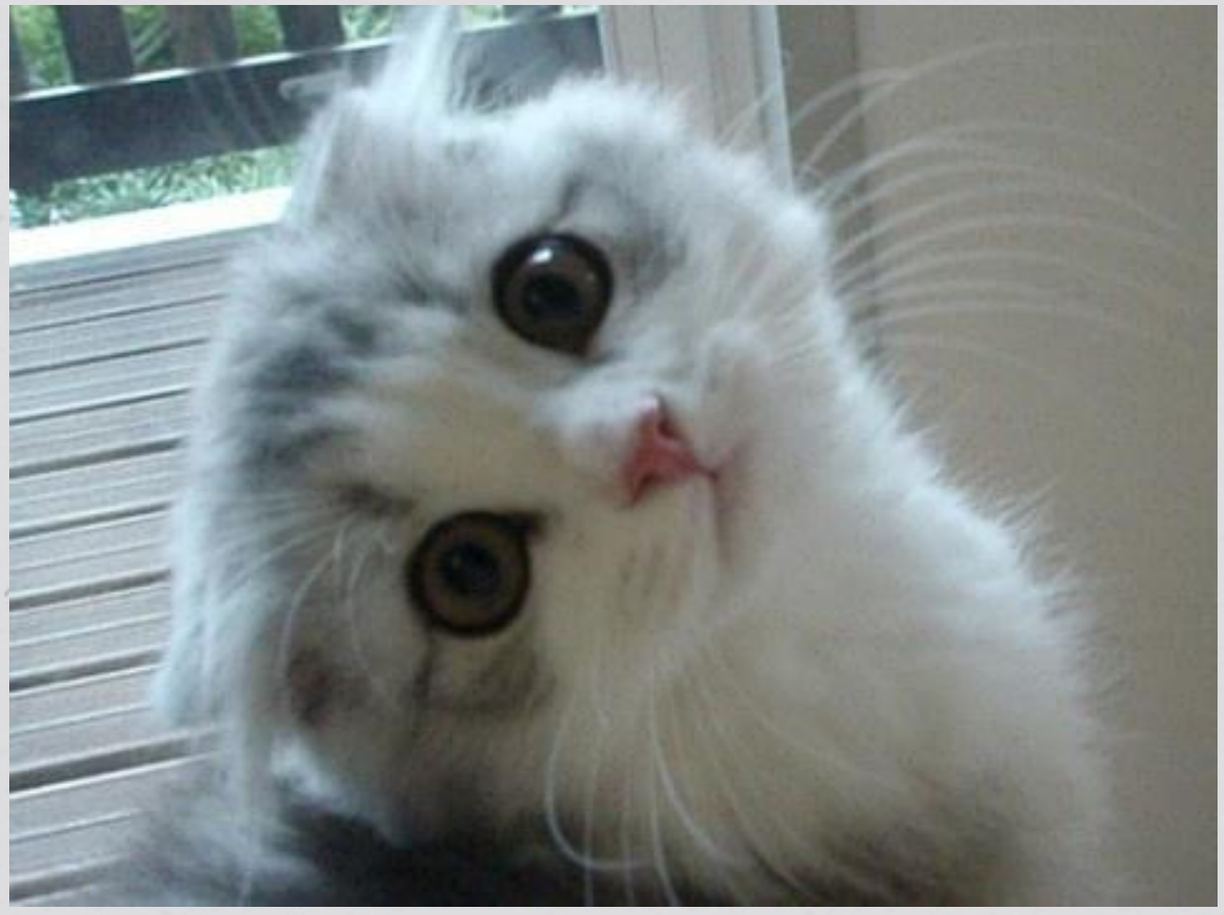
```
--log-queries-not-using-indexes
```

SHOW STATUS;



Variable_name	Value
Bytes_received	183
Bytes_sent	15693
Connections	83875
Created_tmp_files	7
Created_tmp_tables	1
Flush_commands	1
Handler_read_rnd_next	275
Handler_write	274
Innodb_buffer_pool_pages_data	44
Innodb_buffer_pool_pages_free	980
Innodb_buffer_pool_pages_total	1024
Innodb_buffer_pool_read_ahead_rnd	1
Innodb_buffer_pool_read_requests	269
Innodb_buffer_pool_reads	21
Innodb_data_fsyncs	3
Innodb_data_read	2904064
Innodb_data_reads	36
Innodb_data_writes	3
Innodb_data_written	1536
Innodb_log_writes	1
Innodb_os_log_fsyncs	3
Innodb_os_log_written	512
Innodb_page_size	16384
Innodb_pages_read	44
Key_blocks_unused	106733
Key_blocks_used	2615
Key_read_requests	638215
Key_reads	287
Key_write_requests	232608
Key_writes	3146
Qcache_free_blocks	16
Qcache_free_memory	66960032
Qcache_hits	753
Qcache_inserts	188
Qcache_not_cached	5584
Qcache_queries_in_cache	94
Qcache_total_blocks	219
Queries	145454
Questions	6
Select_scan	1
Table_locks_immediate	460

SHOW STATUS;

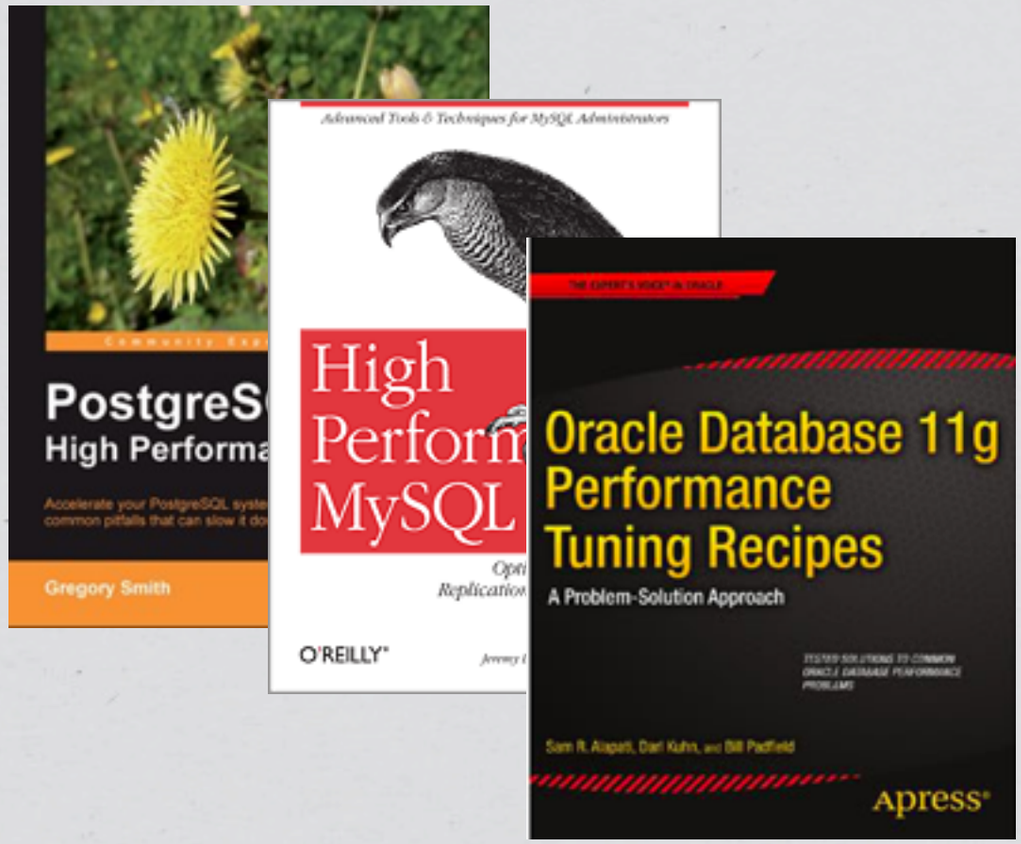


Variable_name	Value
Bytes_received	183
Bytes_sent	15693
Connections	83875
Created_tmp_files	7
Created_tmp_tables	1
Flush_commands	1
Handler_read_rnd_next	275
Handler_write	274
Innodb_buffer_pool_pages_data	44
Innodb_buffer_pool_pages_free	980
Innodb_buffer_pool_pages_total	1024
Innodb_buffer_pool_read_ahead_rnd	1
Innodb_buffer_pool_read_requests	269
Innodb_buffer_pool_reads	21
Innodb_data_fsyncs	3
Innodb_data_read	2904064

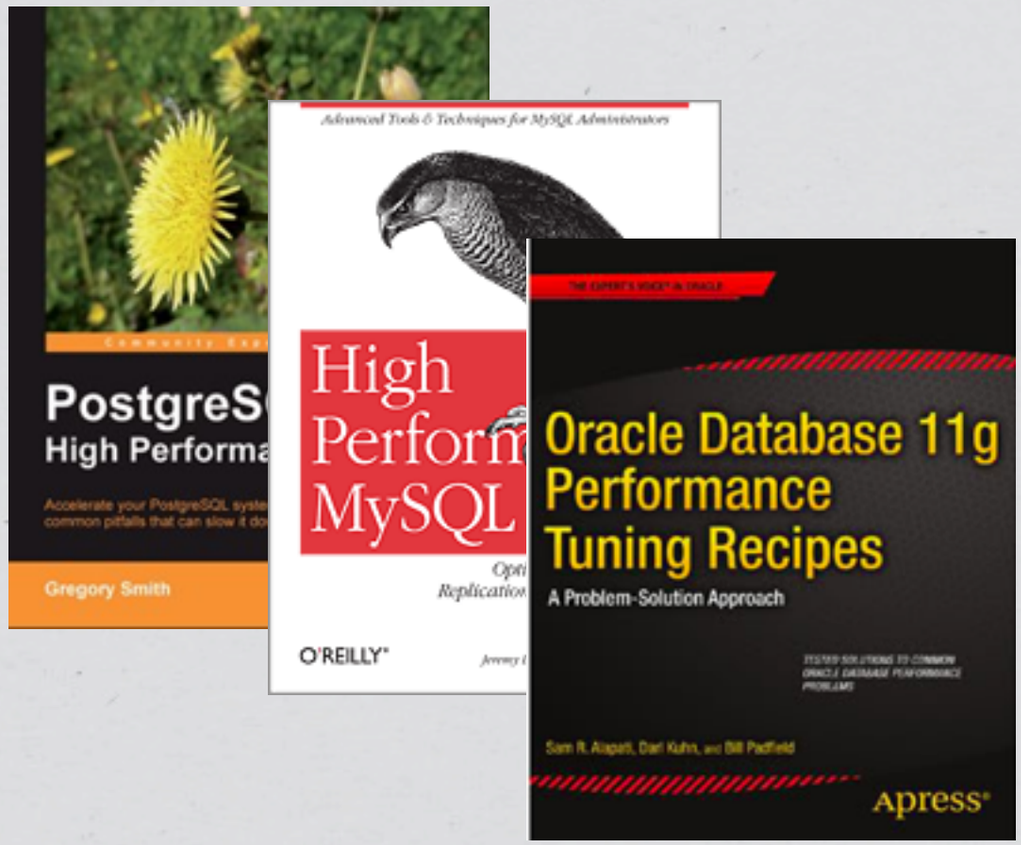
> 300 data points

Innodb_pages_read	44
Key_blocks_unused	106733
Key_blocks_used	2615
Key_read_requests	638215
Key_reads	287
Key_write_requests	232608
Key_writes	3146
Qcache_free_blocks	16
Qcache_free_memory	66960032
Qcache_hits	753
Qcache_inserts	188
Qcache_not_cached	5584
Qcache_queries_in_cache	94
Qcache_total_blocks	219
Queries	145454
Questions	6
Select_scan	1
Table_locks_immediate	460

You'll need some help…

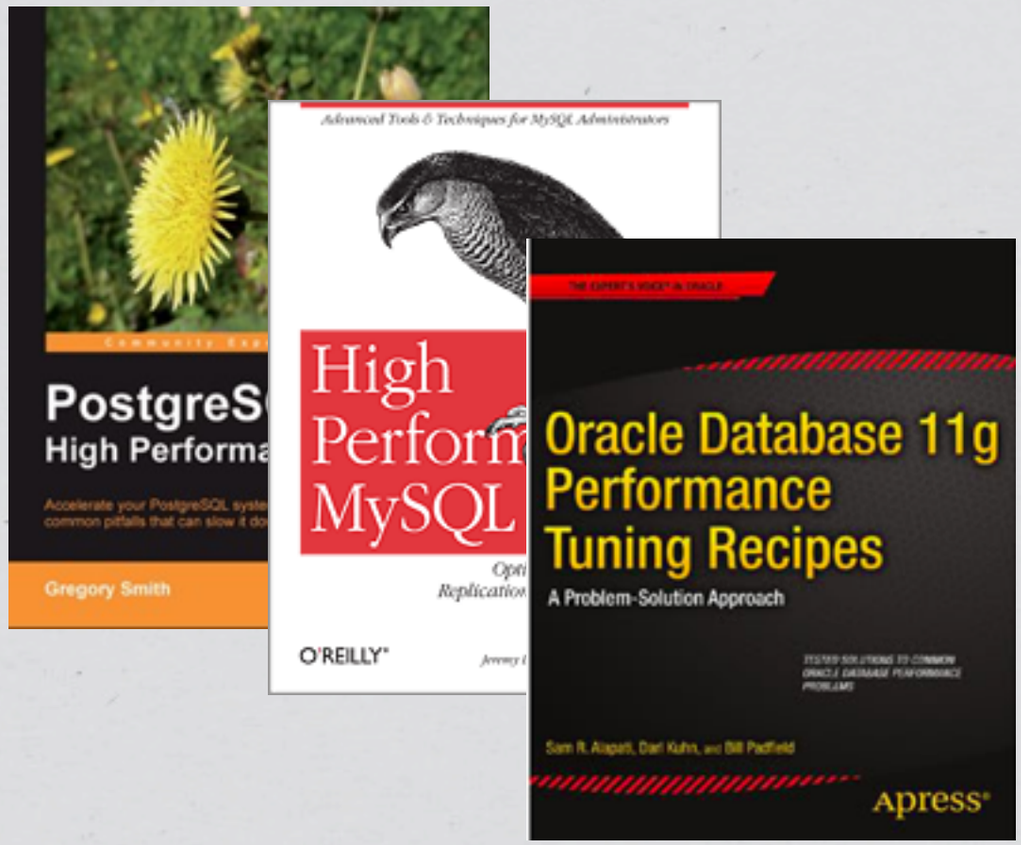


You'll need some help…



Read some books

You'll need some help…



Read some books

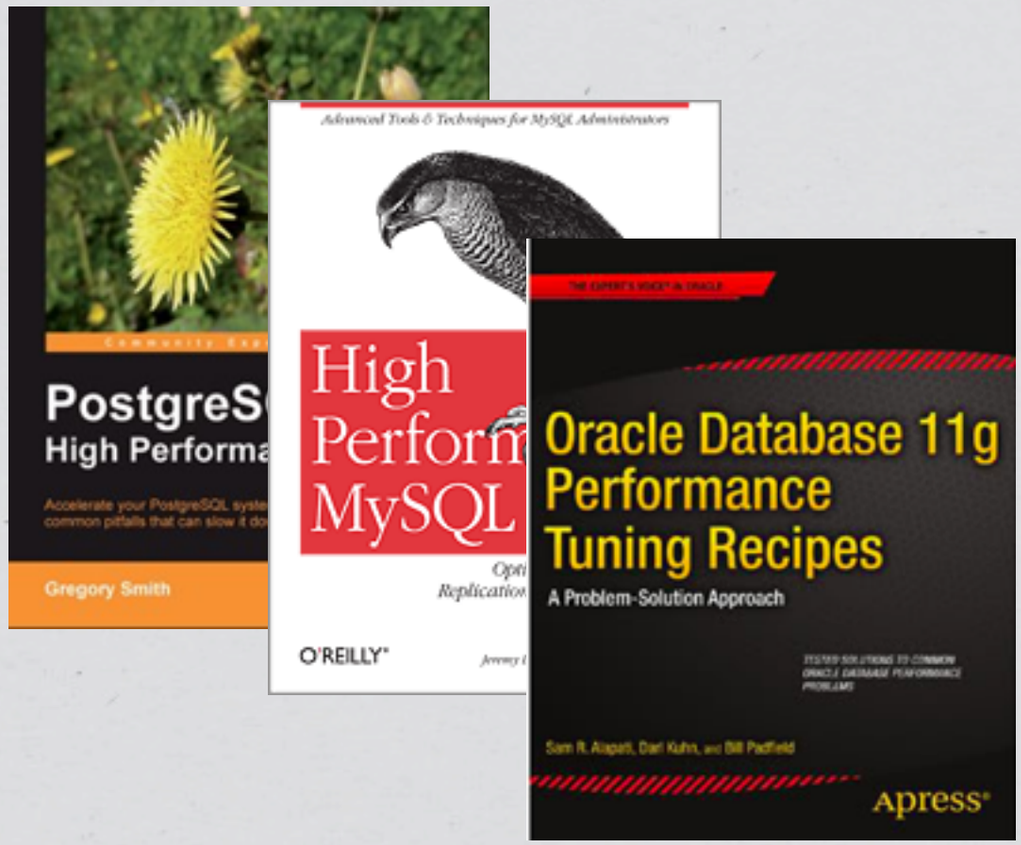
Subscribe to a few blogs

http://www.mysqlperformanceblog.com/

http://planet.postgresql.org/

http://planet.mysql.com/

You'll need some help…



And most of all, good luck!

Read some books

Subscribe to a few blogs

http://www.mysqlperformanceblog.com/

http://planet.postgresql.org/

http://planet.mysql.com/

Watch Your IO

vmstat - Standard Linux IO monitoring tool,

```
$ vmstat
procs -----------memory---------- ---swap-- -----io---- -system-- ----cpu----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa
 1  0 162256 583868 2336676 26893784    0    0     6   192    0    0  0  0 99  0
```

Machine readable ;-)

Watch Your IO

Saidar - http://www.i-scream.org/libstatgrab/

```
Hostname  : s3              Uptime : 375d 20:35:19     Date : 2012-10-15 08:56:44

Load 1    :    0.24   CPU Idle  :  68.06%  Running   :      1   Zombie    :      0
Load 5    :    0.35   CPU System:  12.04%  Sleeping  :    124   Total     :    125
Load 15   :    0.33   CPU User  :  19.90%  Stopped   :      0   No. Users :      1

Mem Total :    5028M  Swap Total:    2000M  Mem Used  : 91.26%  Paging in :      6
Mem Used  :    4588M  Swap Used :   90152K  Swap Used :  4.40%  Paging out:    590
Mem Free  :     439M  Swap Free :    1912M  Total Used: 66.54%

Disk Name       Read          Write          Network Interface         rx         tx
sda            6144B           590K          lo                        0B         0B
                                             eth0                      0B         0B
Total          6144B           590K          eth1                  35056B     59118B

                                             Mount Point             Free       Used
                                             /                     51373M     62.85%
```

For the rest of us...

Slides: http://ilia.ws

@iliaa

THANK YOU FOR LISTENING

Please leave feedback @
https://joind.in/7919