HTTP/2 Stream Priority in ATS

ATS Spring 2016 Summit Masaori Koshiba



Agenda

- 1. Summary of Stream Priority Specs
- 2. Algorithms and Implementations
- 3. Performance
- 4. Issues / Questions / Discussions

1. Summary of Stream Priority Spec

HTTP/2 Streams are Multiplexed

A single HTTP/2 connection can contain multiple streams.

Server

HEADERS #3

HEADERS #5

HEADERS #3

HEADERS #5

DATA #3

DATA #5

DATA #3

DATA #5

Which stream should be proceed first?

Stream Priority

DependencyTree



Response

HEADERS #3

DATA #3

DATA #3

HEADERS #5

DATA #5

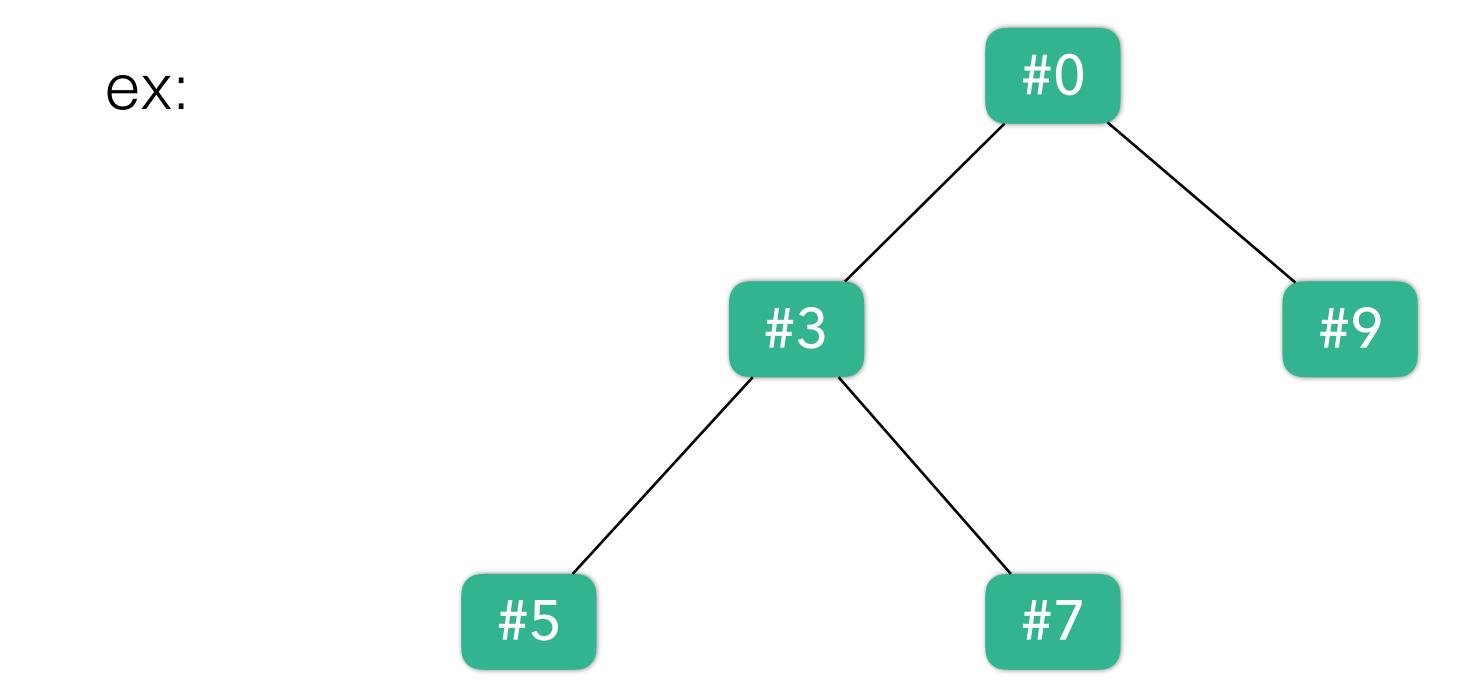
DATA #5

Stream Priority Specs

- Sender (client) can "advice" Stream Priority by sending HEADERS or PRIORITY frame.
- The payload contains
 - Stream Dependency
 - Exclusive Flag
 - Weight

Stream Priority Specs - Stream Dependency

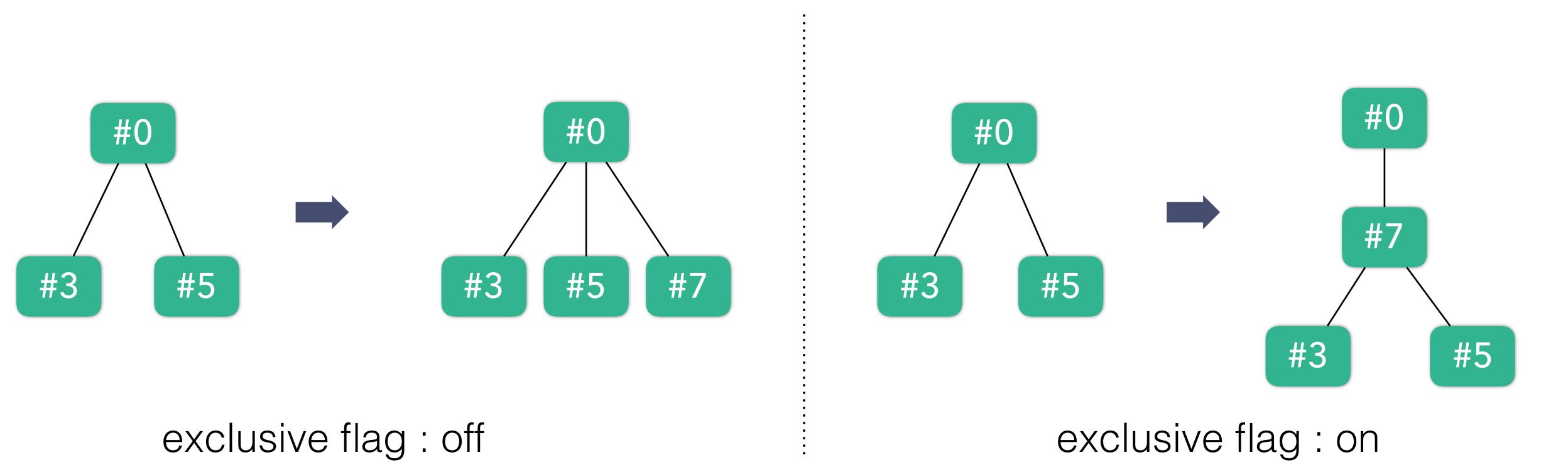
- Each stream can depend on another stream.
 - In default, stream depends on stream #0.



Stream Priority Specs - Exclusive Flag

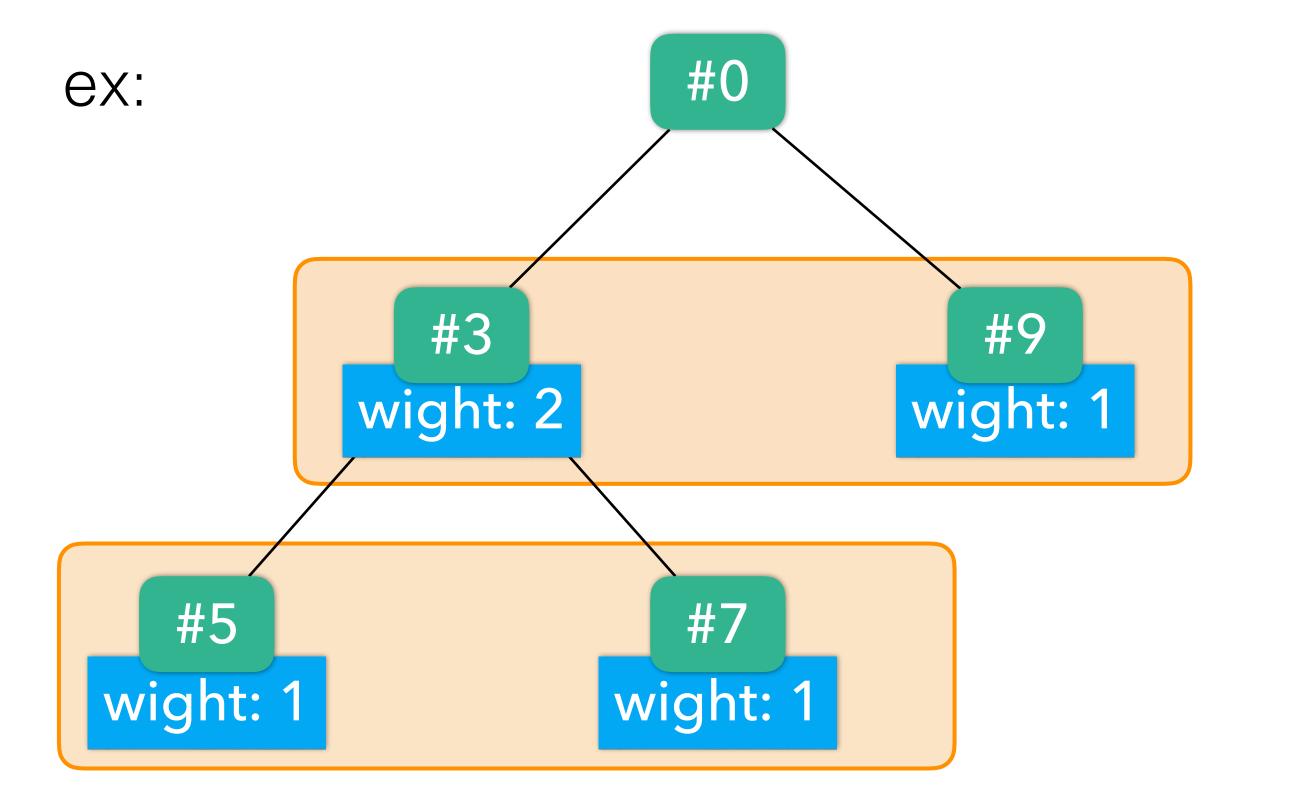
An exclusive flag allows for the insertion of a new level of dependencies.

Add new stream #7 to the tree

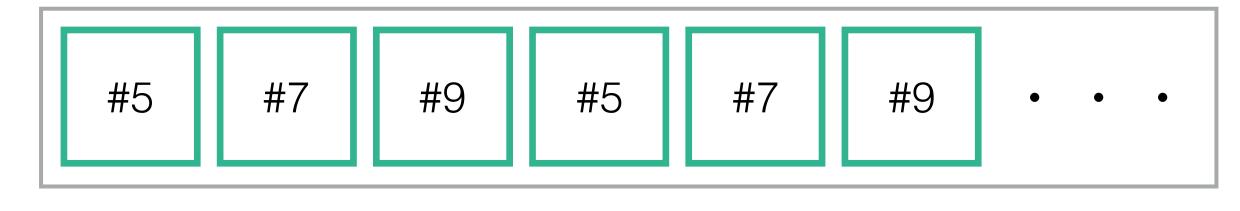


Stream Priority Specs - Weight

• Streams with the same parent SHOULD be allocated resources proportionally based on their weight.



Response Data Frame Queue

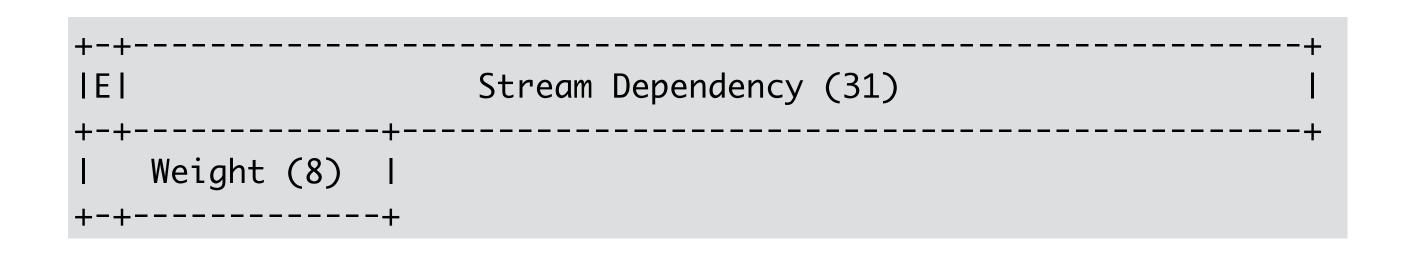


HEADERS Frame

If the PRIORITY flag is set, HEADERS frame contains E, Stream Dependency, Weight fields.

HEADERS frames can be sent on a stream in the "idle", "reserved (local)", "open", or "half-closed (remote)" state.

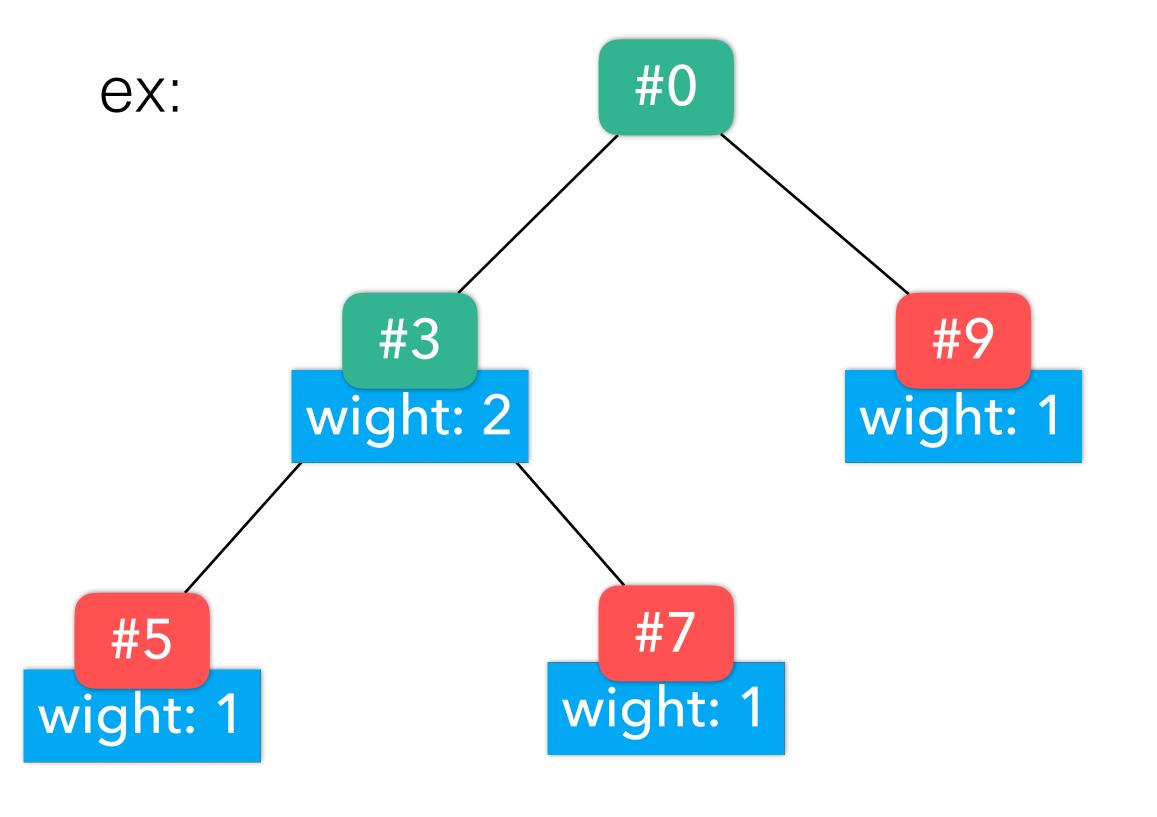
PRIORITY Frame



The PRIORITY frame specifies the sender-advised priority of a stream.

It can be sent in any stream state.

"Active" and "Inactive" Streams

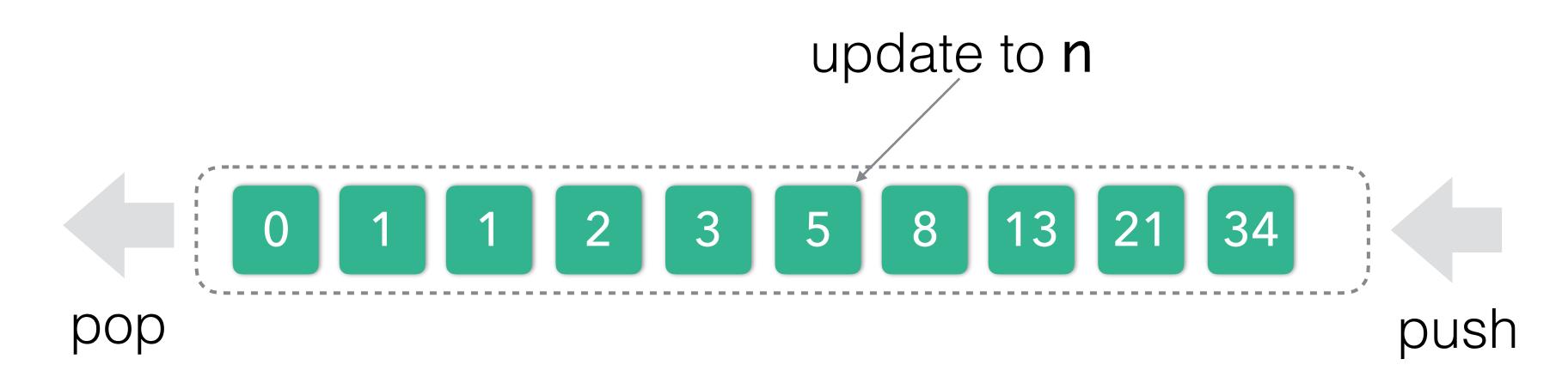


 If stream #3 is not used for requests and responses, it's OK.

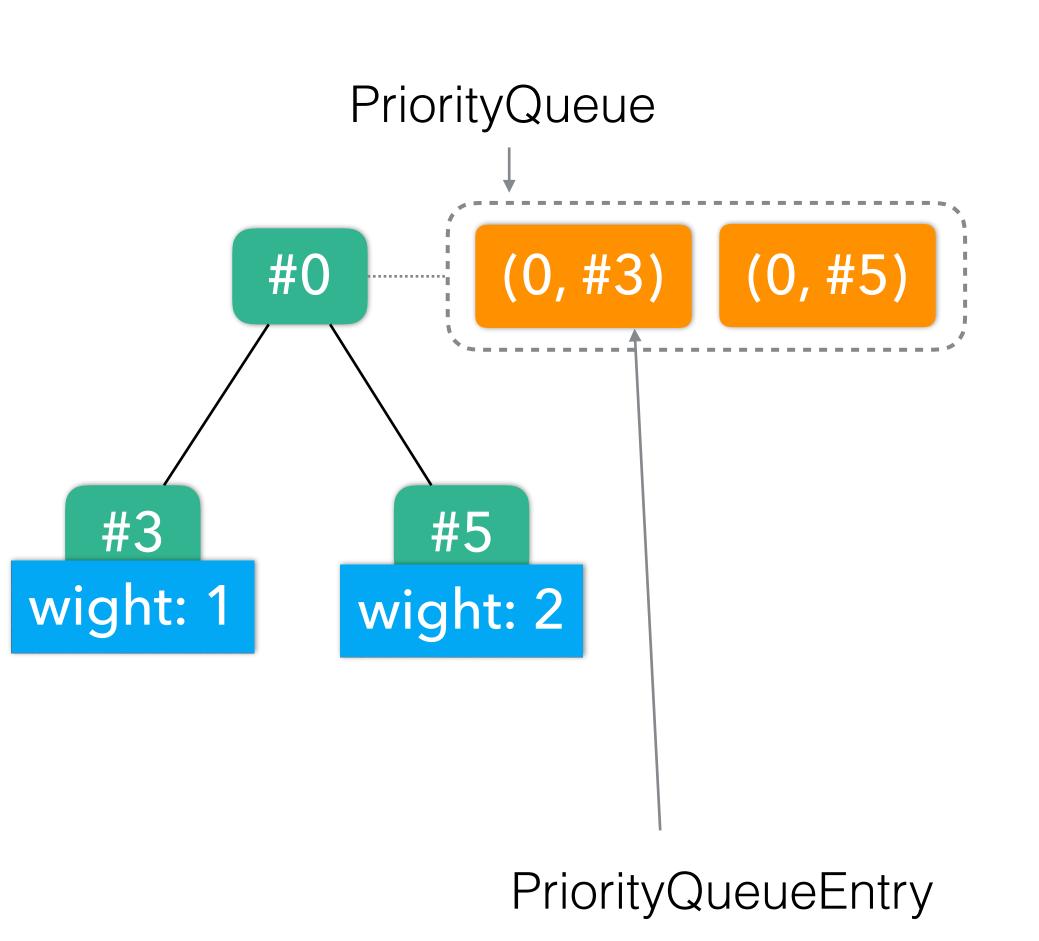
2. Algorithms and Implementations

lib/ts/PriorityQueue.h

- Using Simple Binary Heap (default: min heap)
- Interfaces are little bit similar to std::priority_queue (top/pop/push)
- But we can update point of entry



Weighted Fair Queuing

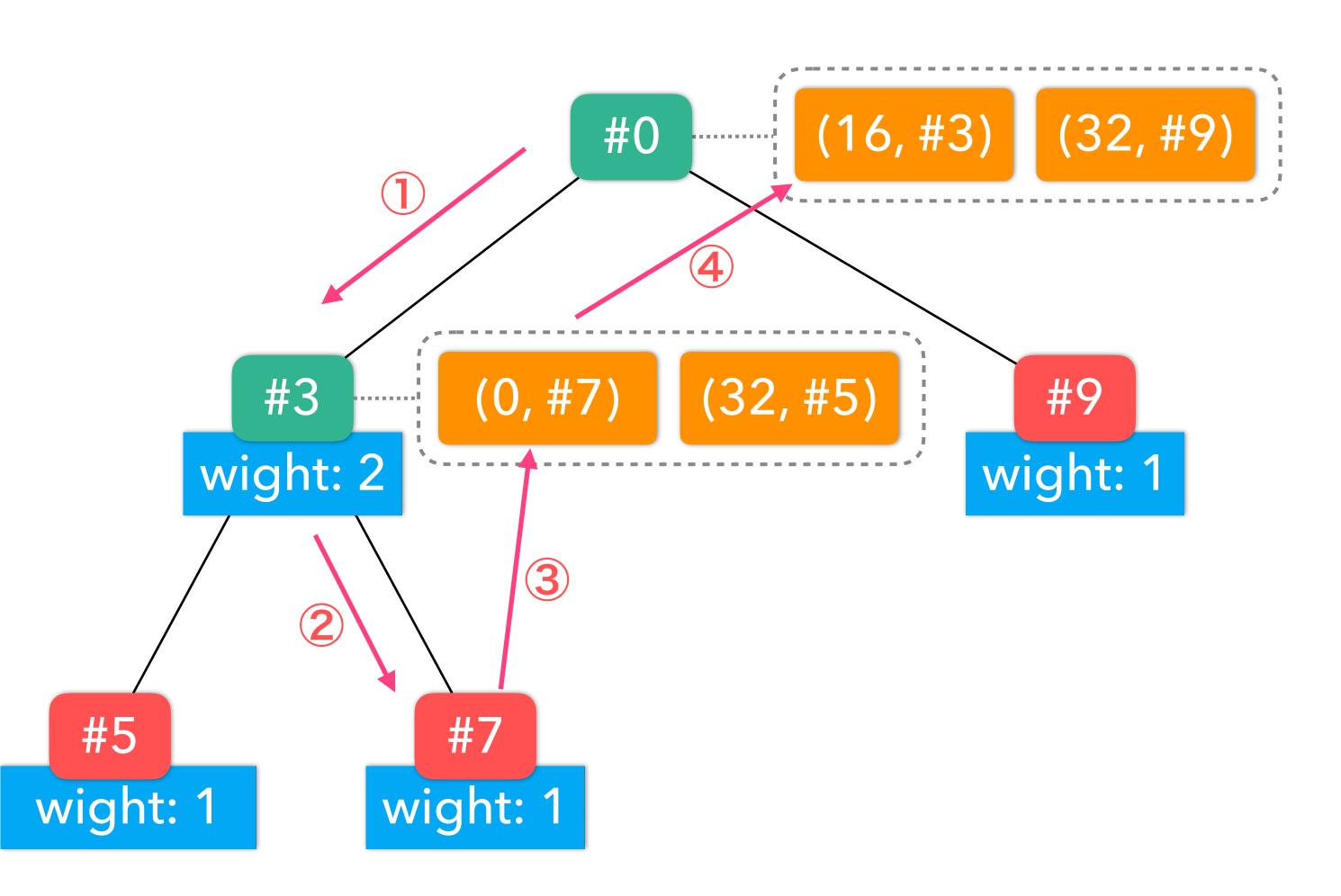


- Parent node has PriorityQueue.
- When a child node is ready, an entry for the node is pushed to PriorityQueue.
- When schedule streams, top entry of PriorityQueue is popped, sent response body and update point with below calculation.

```
point += sent_length * K / weight
```

(point, *stream) (K is constant number)

Dependency Tree



- Intermediate node(#3) has PriorityQueue.
- Start from #0, get top node of PriorityQueue in each level, Stop when get active node.
- Update point of entries in PriorityQueue recursively.

Notes

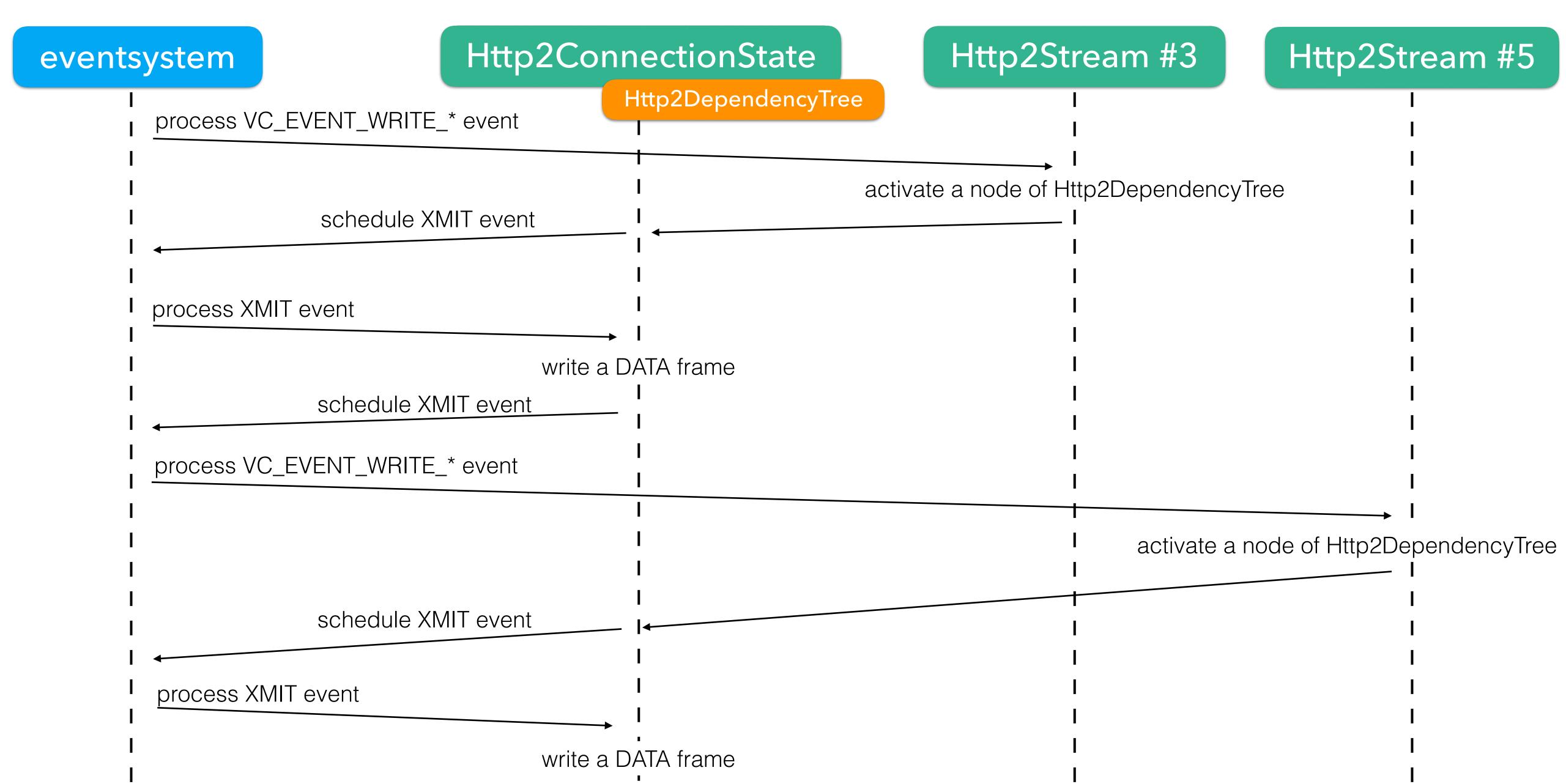
- The algorithm of scheduling streams using WFQ is came from h2o project and introduced by Kazu Yamamoto (warp) and Tatsuhiro Tsujikawa (nghttp2) at ATS Meetup in Tokyo 2015.
 - Server Implementations of HTTP/2 Priority
 - HTTP/2 priority implementation in nghttp2

ATS specific notes

- Currently only DATA frames are scheduled.
 - Almost all of HEADERS frames could be put in one frame (16KB)

- After every one DATA frame is written in buffer, an event is scheduled and go back to event-system.
 - To check if there are any new active node is available.

Scheduling DATA frames

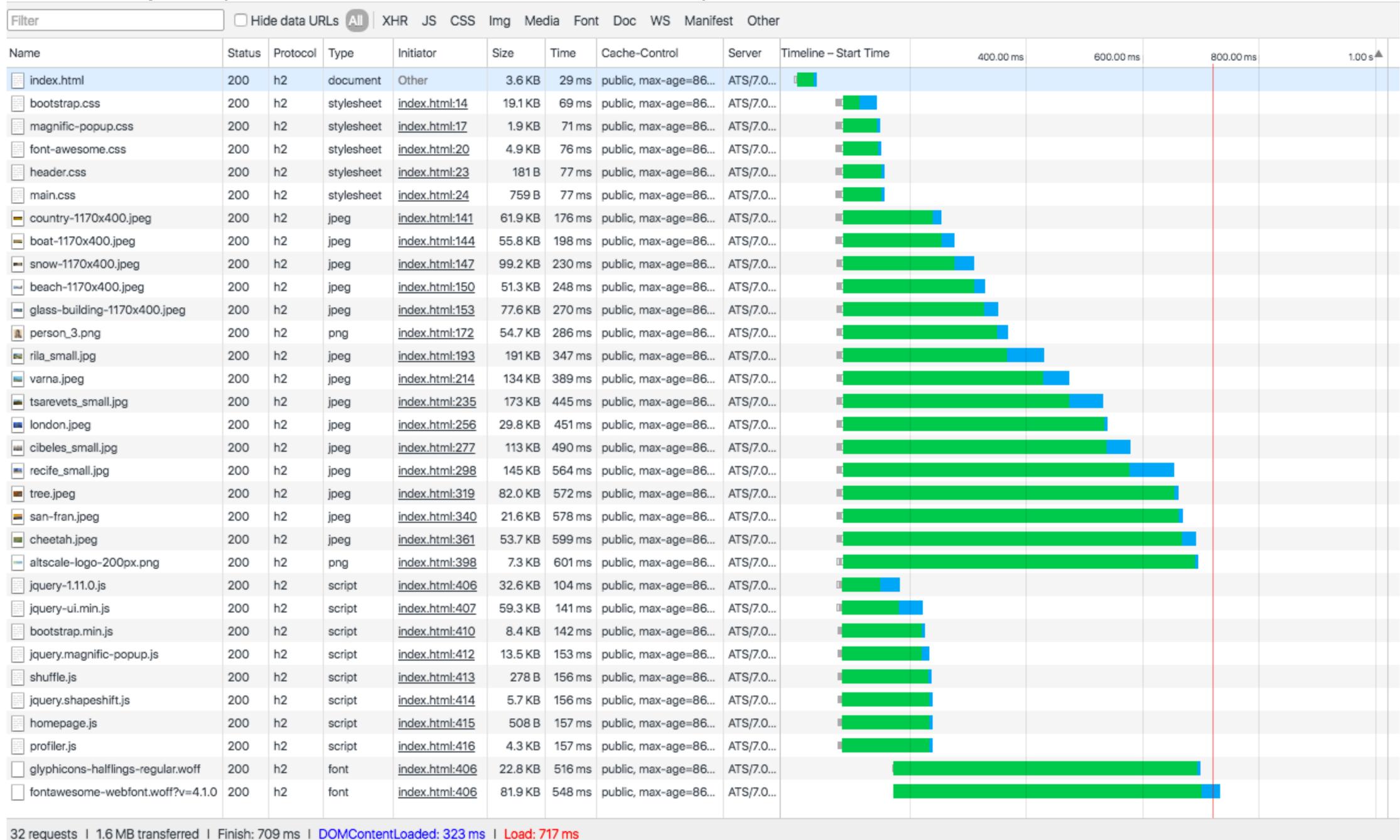


3. Performance Tests

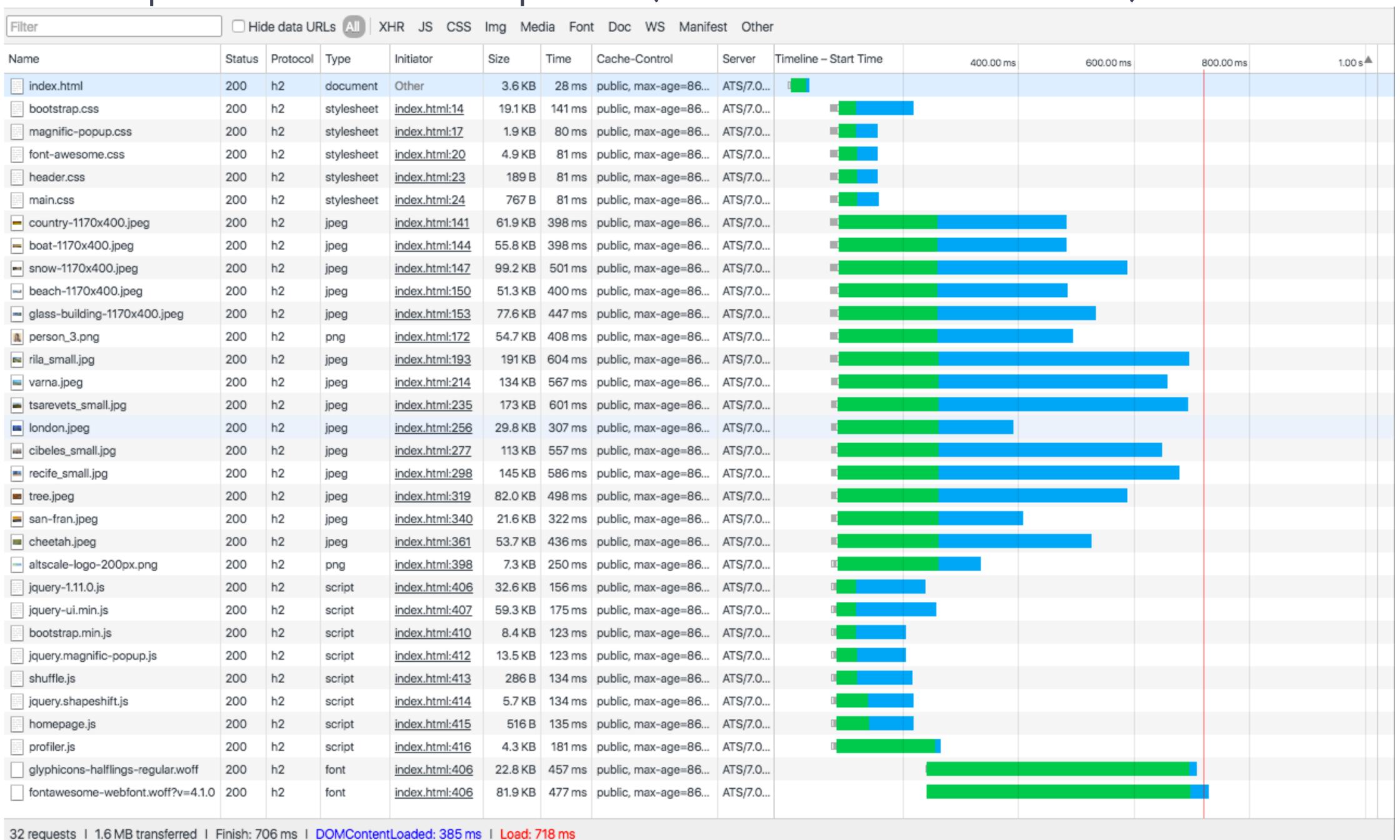
Performance Test

- Origin Server: HTTP2Rulez! is served from Apache
 - https://github.com/ipeychev/http2rulez.com
 - All contents are cacheable
 - Cache-Control: public, max-age=86400
- ATS: master(v7.0.0-pre) vs TS-3535 patch
- Client: Chrome 50.0.2661.86 / FireFox 46.0.1
 - Browser-Cache is disabled

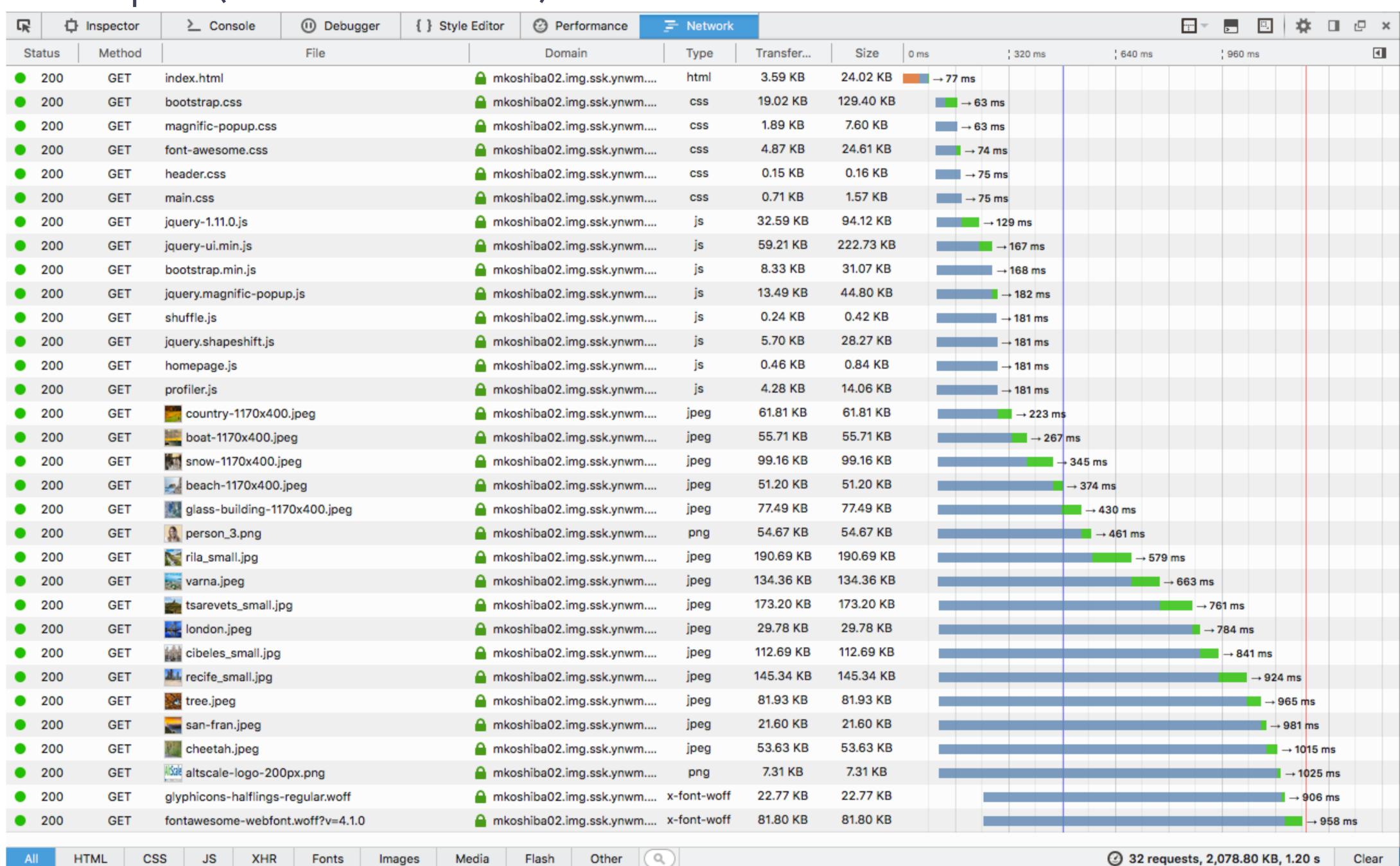
ATS-v7.0.0-pre(Chrome 50.0.2661.94)



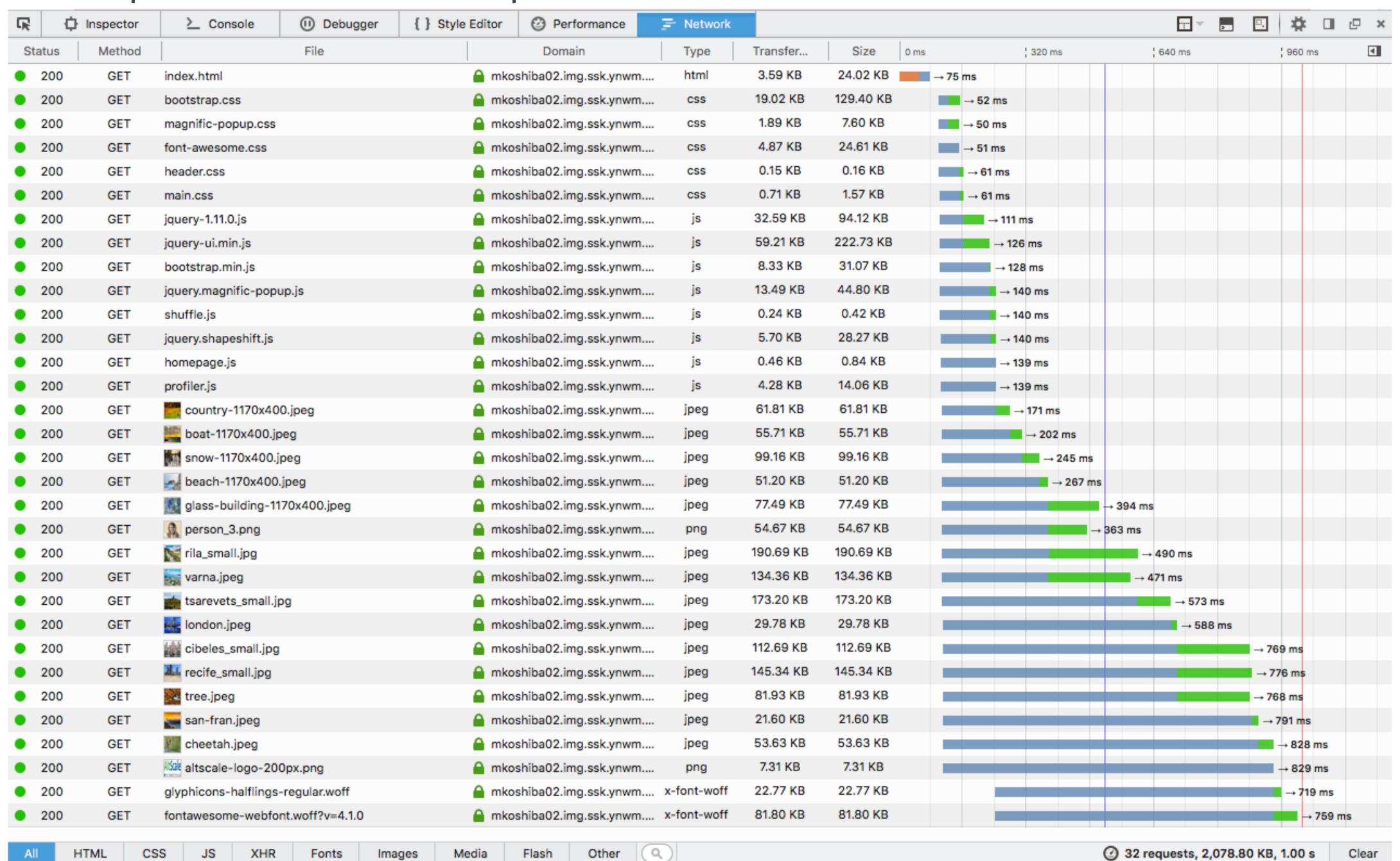
ATS-v7.0.0-pre with TS-3535 patch (Chrome 50.0.2661.94)



ATS-v7.0.0-pre (FireFox 46.0.1)



ATS-v7.0.0-pre with TS-3535 patch (FireFox 46.0.1)



ATS-v7.0.0-pre

- TTFB (Time To First Byte) is increased gradually
 - TS returns response body as much as possible

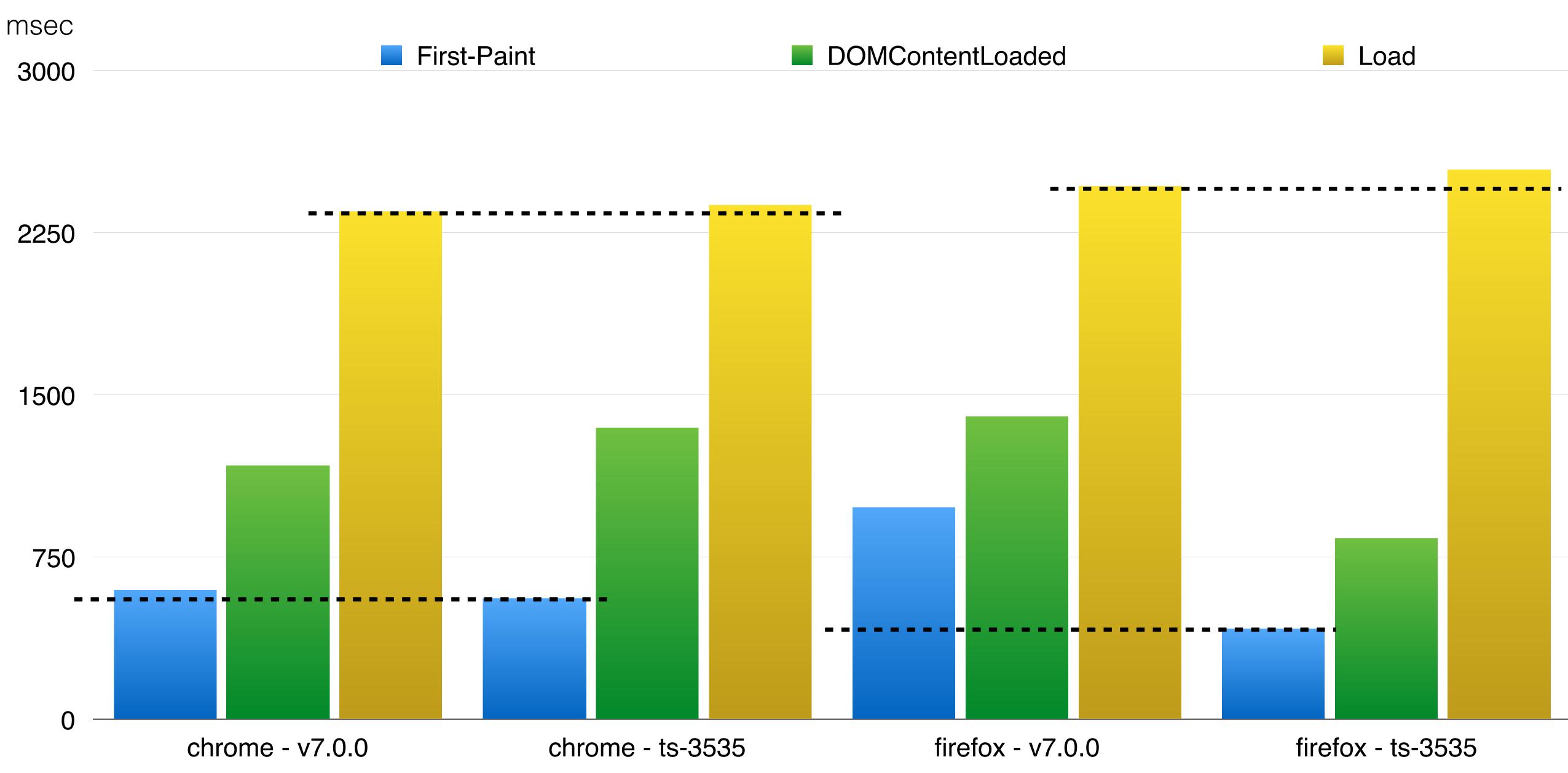
ATS-v7.0.0-pre with TS-3535

- Total duration is almost same (over heads looks little)
- TTFB (Time To First Byte) is not increased gradually
 - Content Download duration is increased

First-Paint Time Benchmark

- Server Side: sudo to qdisc replace dev eth0 root netem delay 100ms
- firstPaint.js from Browsertime
 (https://github.com/tobli/browsertime)
 - Chrome has a API to get First-Paint time but FireFox doesn't https://bugzilla.mozilla.org/show_bug.cgi?id=1210906
- Average of 10 requests / responses

First-Paint Time (latency: 100msec)



Conclusion

- In some cases, Stream Priority Feature might improve User Experiences by decreasing First-Paint time.
 - Note that it's trade-off (overheads are there)
- We need more test and performance improve.

4. Issues / Questions / Discussions

Issues / Questions / Discussions

Designs

• Is there any more good design of scheduling DATA frames?

Policies

- Should TS ignore client's advice when it is looks bad?
 - And how to detect it?



Appendix

- Understanding HTTP/2 prioritization
 https://speakerdeck.com/summerwind/2-prioritization
- Server Implementations of HTTP/2 Priority
 http://www.mew.org/~kazu/material/2015-http2-priority2.pdf
- HTTP/2 priority implementation in nghttp2 https://goo.gl/3sSHXJ
- http2rulez!
 https://github.com/ipeychev/http2rulez.com
- Browsertime
 http://browsertime.net