ATS + HTTP/2 Meetup in Tokyo

Nov 6, 2015

Masaori Koshiba

Refactoring HTTP/2 Component of ATS

Motivation

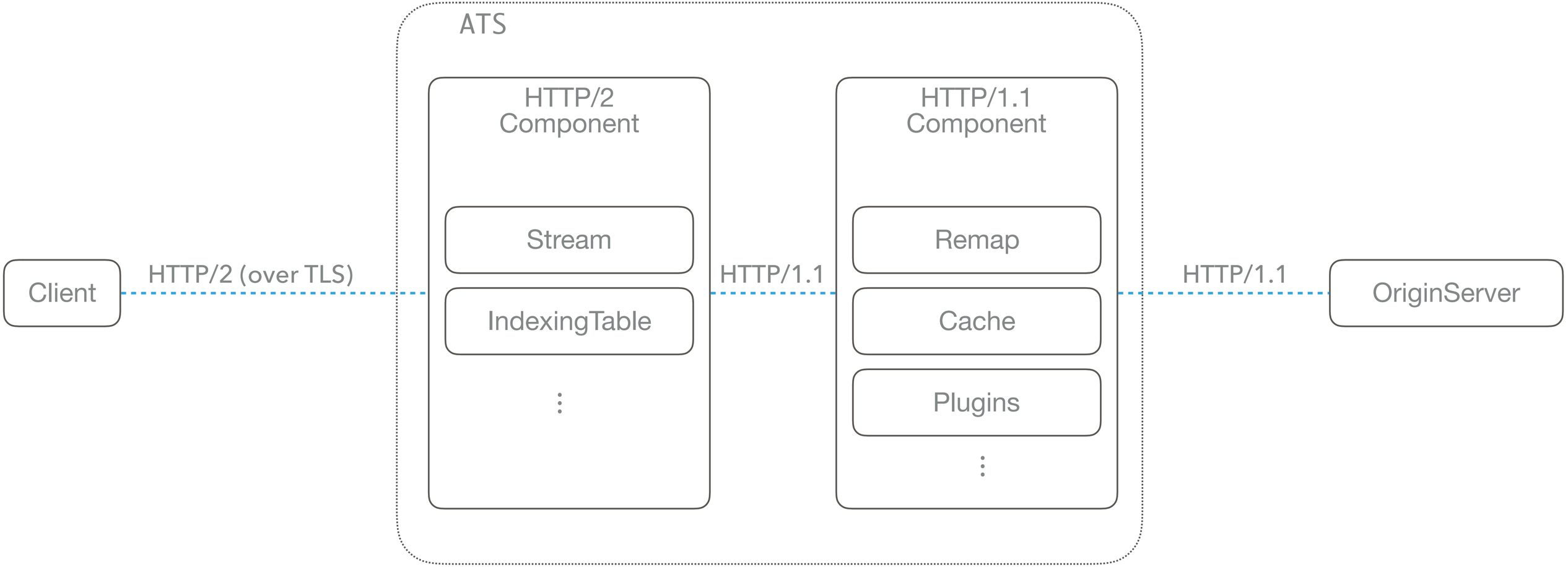
- We need to implement more HTTP/2 features
  - Stream Priority
  - Server Push
  - Back posting via HTTP/2
- But can we implement those features on current design?

Agenda

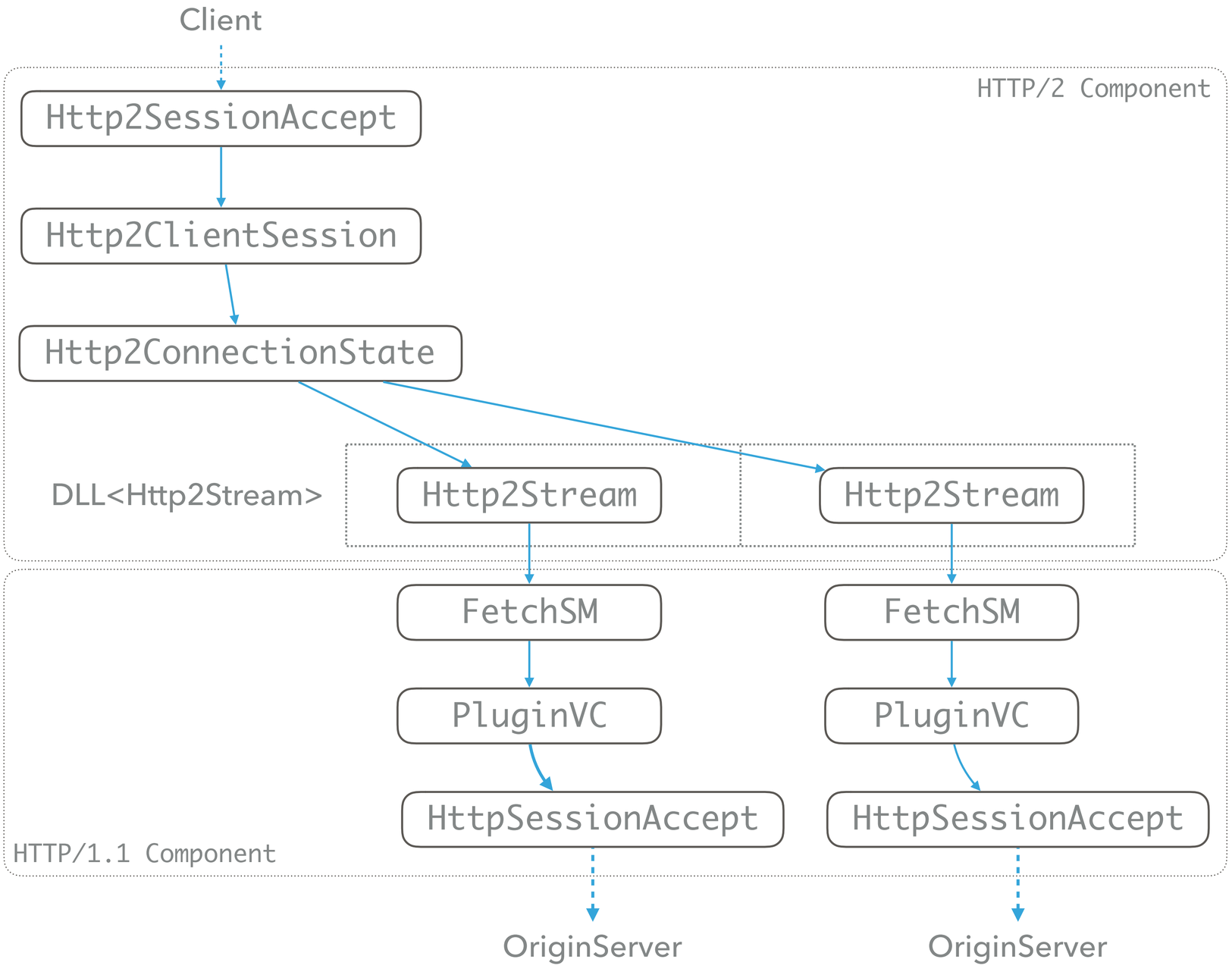
- Current Design Overview (v6.0.0)
- Refactoring Plans
- HTTP/2 and HPACK

Current Design Overview

Current Design Overview

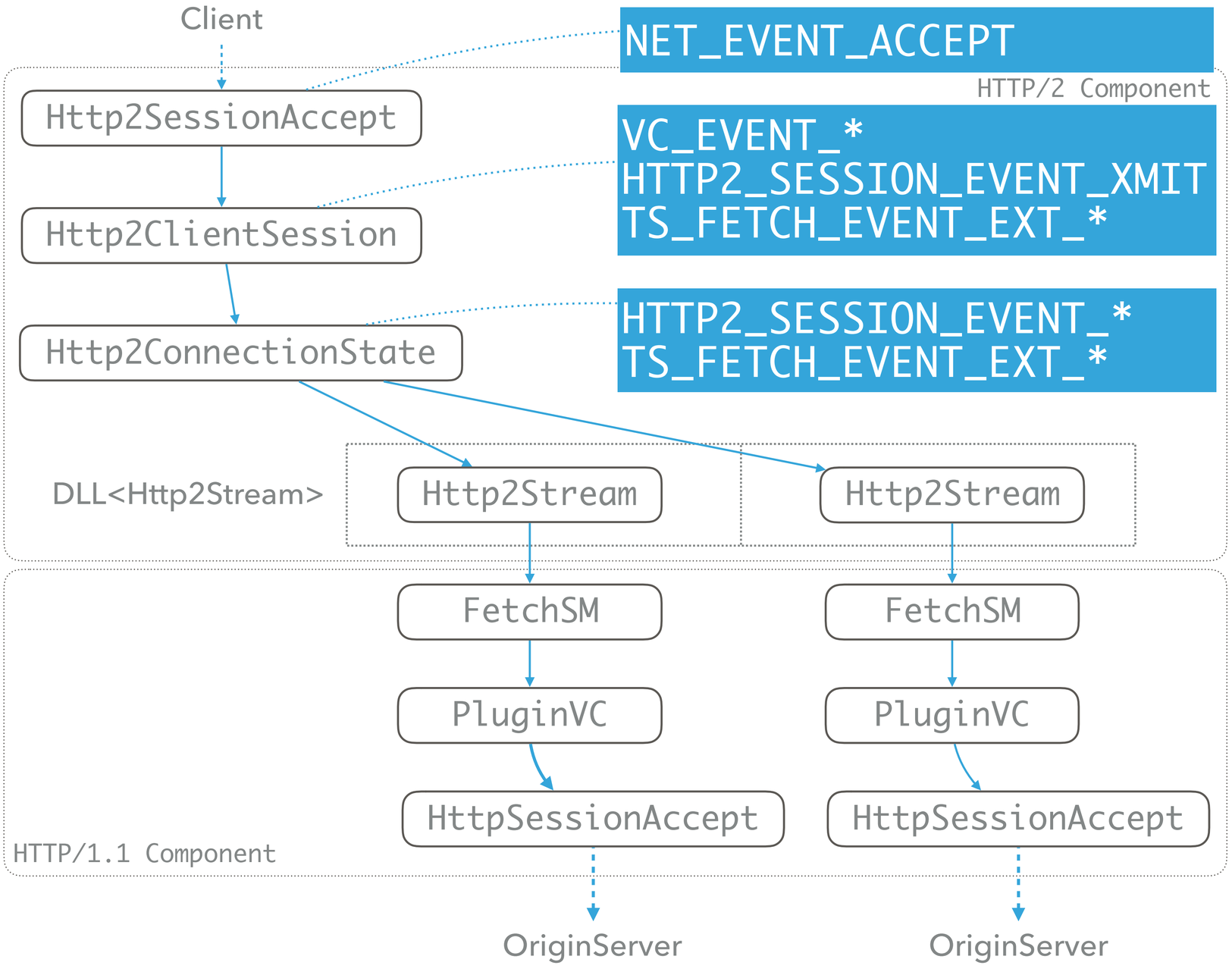


Current Design Overview - HTTP/2 Component - Requests



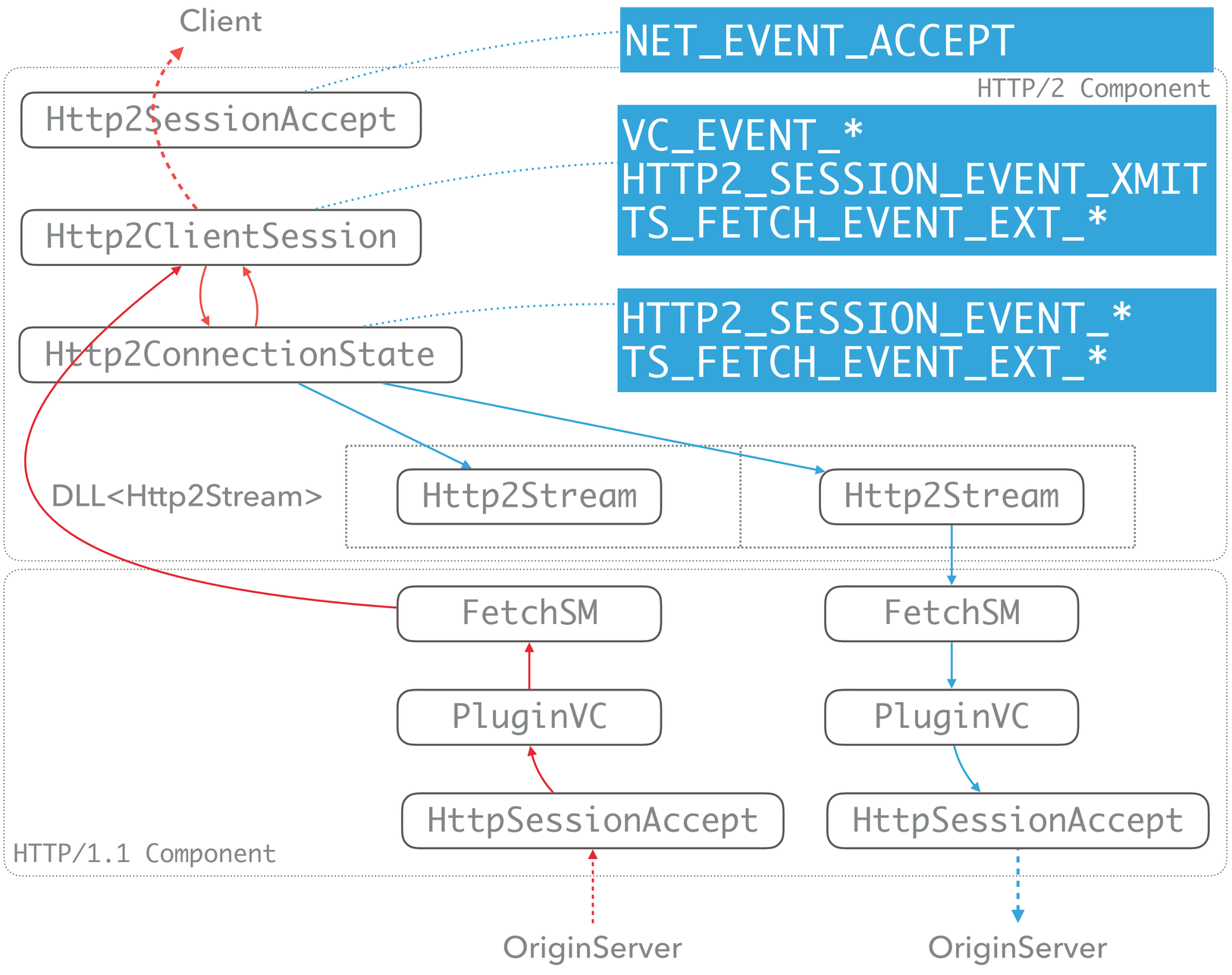
- Http2ConnectionState has Double Link List of Http2Stream.
- Http2Stream isn't continuation.
- FetchSM & PluginVC are bridges to HTTP/1.1 component.
  Those come from SPDY Plugin.

Current Design Overview - HTTP/2 Component - EVENT Handling



- Http2ClientSession and Http2ConnectionState handle “TS_FETCH_EVENT_EXT_*” events.

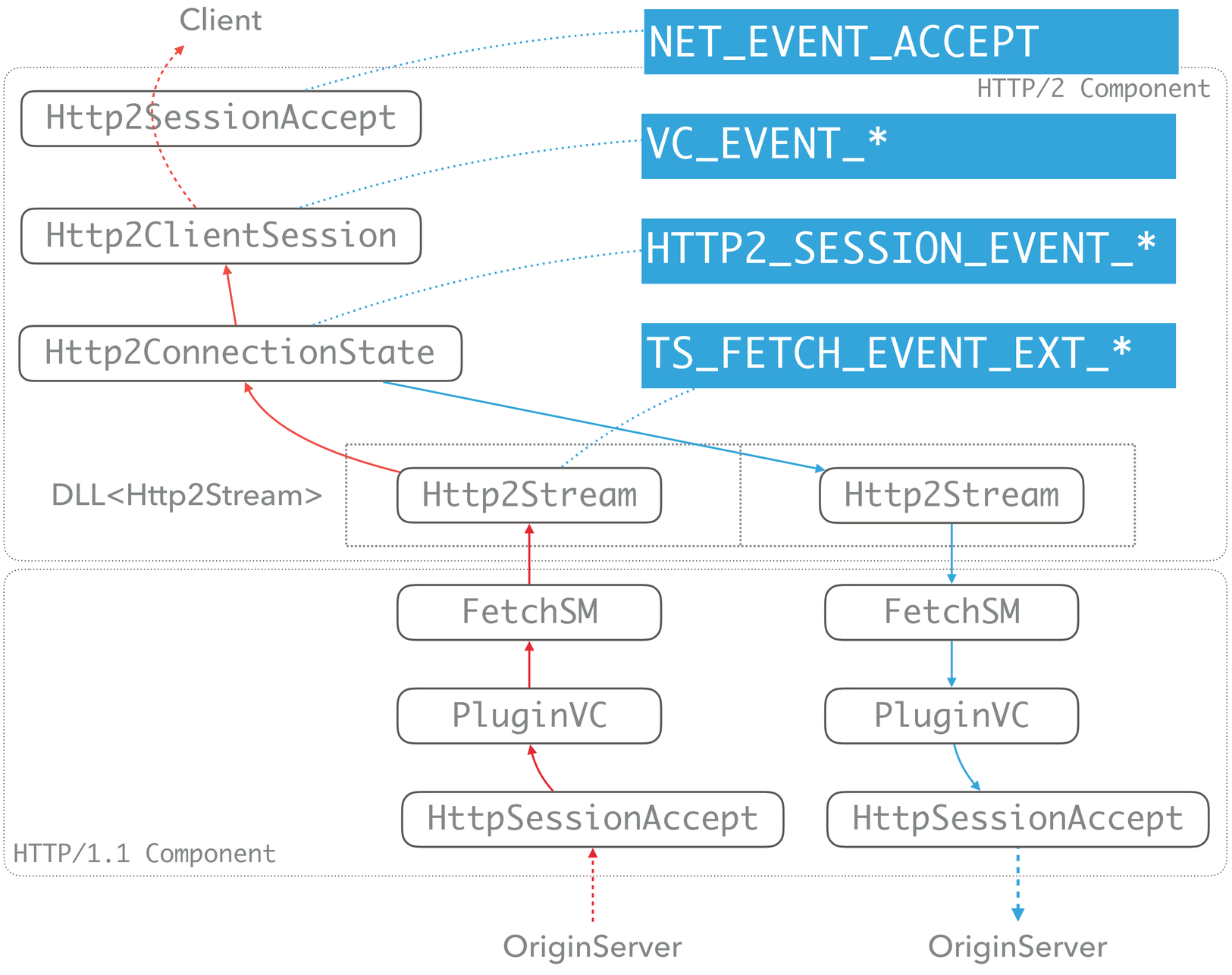
Current Design Overview - HTTP/2 Component - Responses



- Http2ClientSession and Http2ConnectionState handle events from FetchSM.
- Events from FetchSM are issued to Http2ClientSession once.
- Http2Stream isn't continuation, thus Http2Stream can't work in parallel.

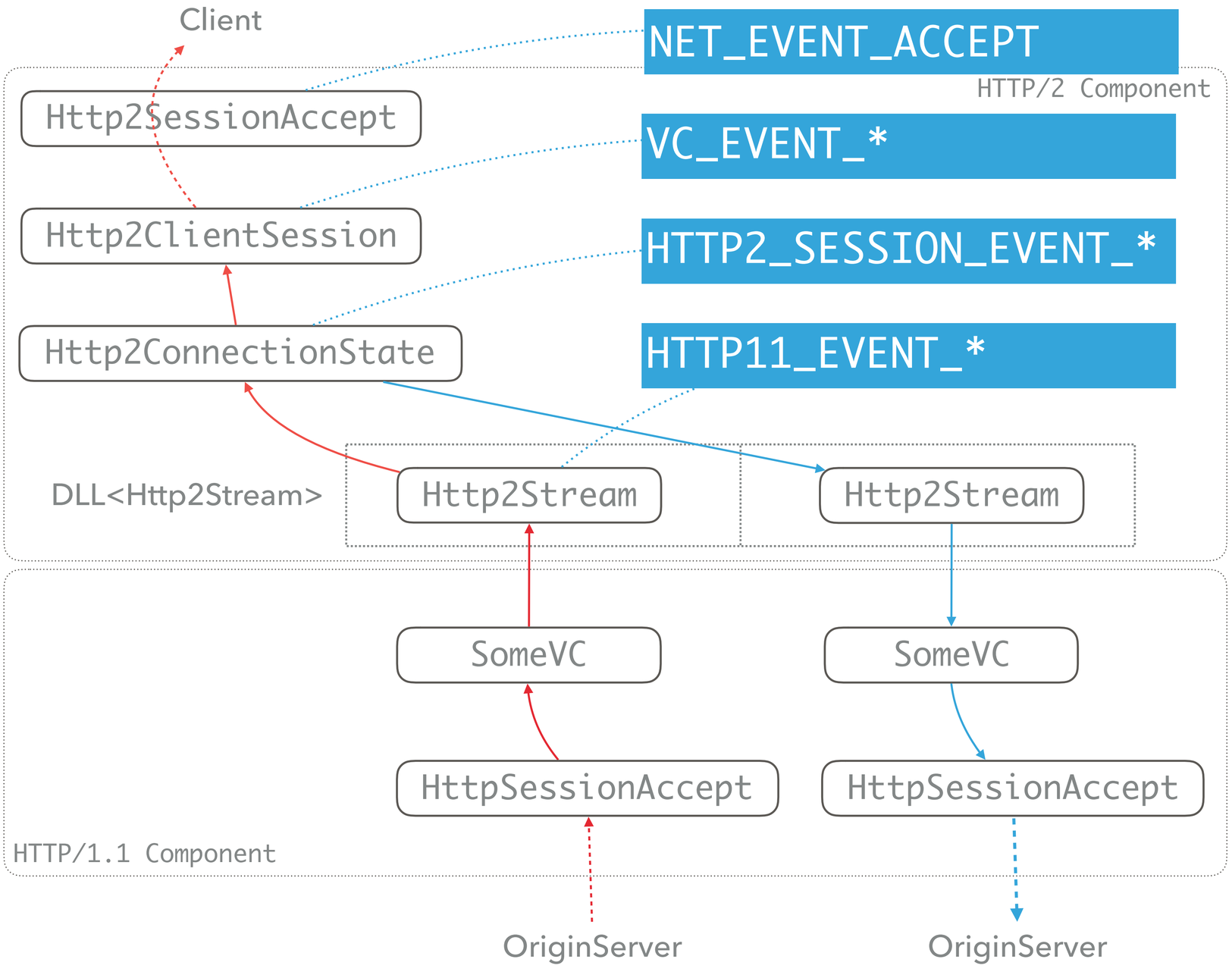
Refactoring Plans

Refactoring Plan - HTTP/2 Component - Phase 1



- Clarify events and handlers.
- Make Http2Stream Continuation.
- Http2ConnectionState parse only header of frames. Allocate new Http2Stream or put payloads to existed Http2Stream.
- Add scheduler in Http2ConnectionState to handle Stream Priority.

Refactoring Plan - HTTP/2 Component - Phase 2



- Replace FetchSM & PluginVC to some Virtual Connection to avoid overheads of *memcpy()*.

HTTP/2 and HPACK

HTTP/2 & HPACK

- In some cases, HTTP/2 and HPACK are mixed.
  - ex. http2_decode_header_blocks()
    - This function is HPACK implementation. But there are CONNECTION header checking. This check comes from HTTP/2. This makes adding unit test difficult.