HTTP/2 Semantics over QUIC

draft-shade-quic-http2-mapping-00

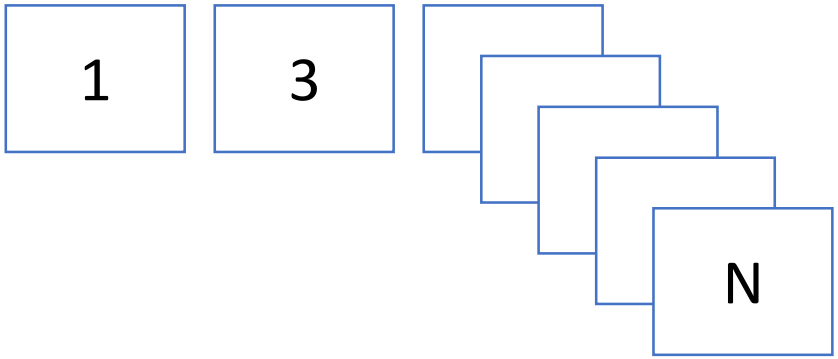
Current State

- Documents current state of QUIC protocol as deployed by Google
- Reflects iterative evolution from HTTP/2 over TCP to QUIC
- Minor process stuff to add
  - Security Considerations, IANA, etc.
  - Interaction with HTTP/2 registries, if any

Connection Negotiation

- HTTP/QUIC detected by use of Alt-Svc
  - Alt-Svc: quic=“:443”;v=“32,33”
  - New “v” parameter for Alt-Svc defined to carry version negotiation hints
- Negotiation of HTTP/QUIC versus foo/QUIC not yet specified
  - More general QUIC discussion to be had here
  - Could reasonably be implied by UDP port number assignments
  - Could be negotiated by ALPN token in TLS handshake

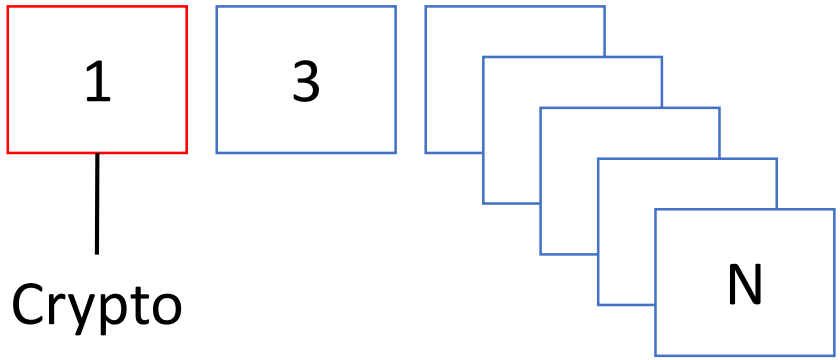
Connection Structure



QUIC streams

Connection Structure

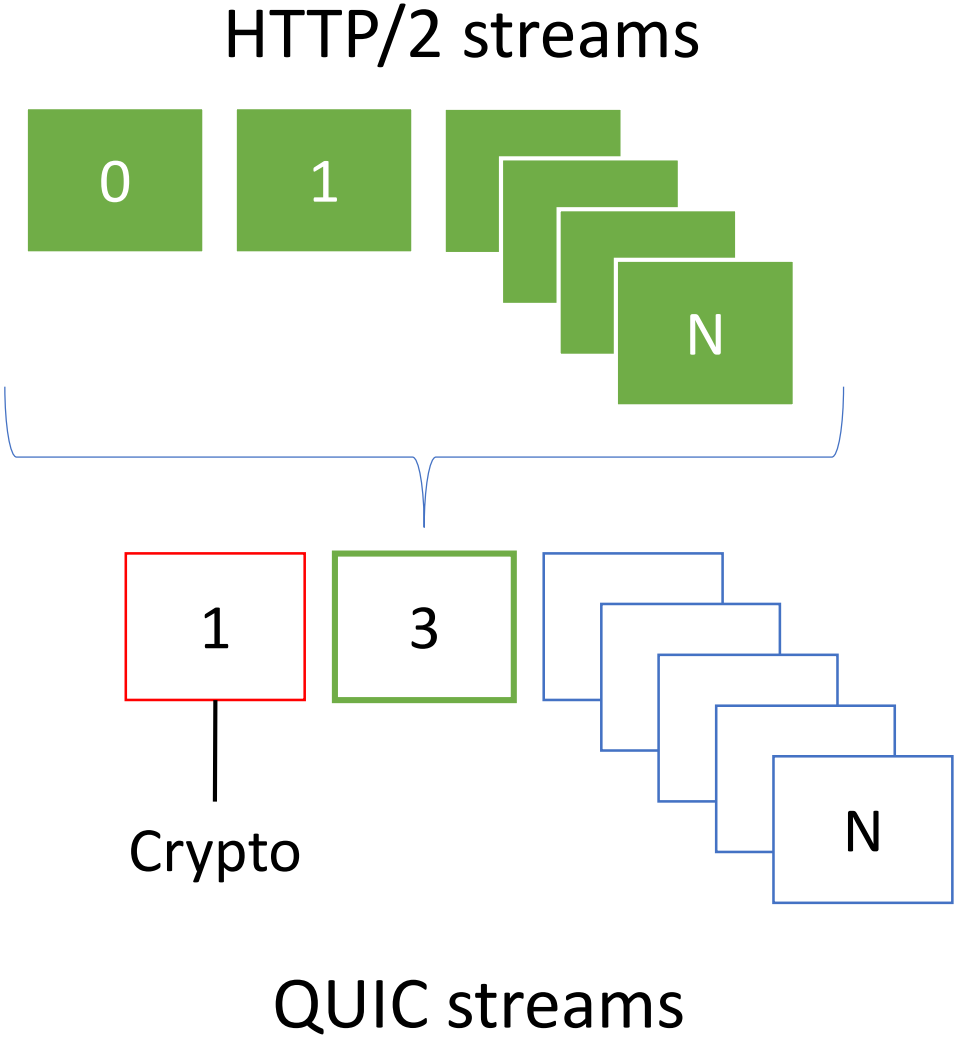
- Stream 1 reserved for crypto
  - Core QUIC spec



QUIC streams

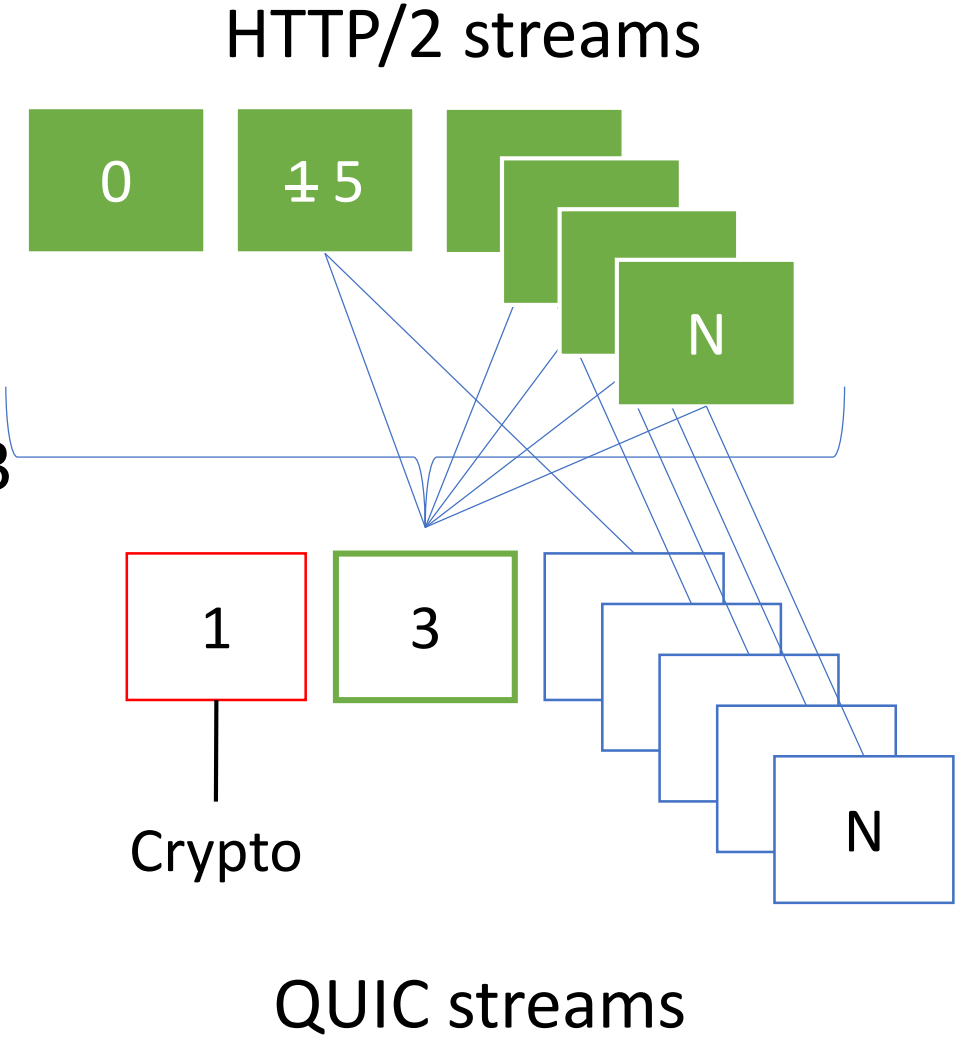
Connection Structure

- Stream 1 reserved for crypto
- Stream 3 reserved for abridged HTTP/2 session
  - Reflects migration path from TCP to QUIC
  - Functionality added to QUIC is removed from HTTP/2
    - PING
    - GOAWAY
    - Flow Control



Connection Structure

- Stream 1 reserved for crypto
- Stream 3 reserved for abridged HTTP/2 session
- HTTP/2 streams straddle QUIC Stream 3 *and* another QUIC stream
  - H2 Stream 0 is only on QUIC Stream 3
  - Other QUIC streams replace DATA frames
  - All other frames (HPACK) on QUIC Stream 3



A Fork in the Road

HTTP/2 over QUIC

Pro:

- Reuse existing HTTP/2 framing code
- H2 extensions will (probably) work unchanged

Con:

- Double-mux
- Head-of-line blocking on HPACK

Fresh HTTP Mapping over QUIC

Pro:

- Leave streams to QUIC
- Simplifies stream management

Con:

- New framing required
- HLB-avoiding header compression is a hard problem
  - HPACK => QPACK?

Some other possible routes

- Split DATA and Other-Stuff streams
  - 2 QUIC streams per HTTP transaction
    - One carries header blocks, other carries body
    - TBD: Which stream carries PUSH_PROMISE?
    - TBD: Extensions?
  - Still needs framing within each stream
- Stream per frame type
  - Protects extensions from ordering requirements amongst their own frames
  - Doesn’t solve cross-type ordering requirements
    - SETTINGS
    - HPACK