

# Multipath extension for QUIC

## draft-ietf-quic-multipath-03

QUIC meeting @ IETF-115 London  
Yanmei Liu, Yunfei Ma, Quentin De Coninck,  
Olivier Bonaventure, Christian Huitema, Mirja Kühlewind

# Diff from -02 to -03

Mostly editorials and minor changes

- Ack delay and RTT considerations for single packet number space (#125, #131)
- Implementation considerations on keep-alive (#144, #148)
- Remove MP\_CONNECTION\_ERROR error code (#136, #141)
- Purely editorials: #138, #142, #143, #147

# Ack delay and RTT considerations for SPNS (#125, #131)

Issue with generation of RTT sample as defined by QUIC-RECOVERY

- Some paths' RTT might not be updated if ACKs mostly come from other ones
- Only update if largest acknowledged increases over the connection, sample might be sum of two different paths' one-way delay

Two solutions:

- Either use timestamp extension
- Or adopt specific practices at both endhosts
  - Write path-specific "Largest Acknowledged" (depends on the path used to send ACK)
  - Accept samples when "Largest Acknowledged" increases on a per-path basis

# Implementation considerations on keep-alive (#144, #148)

The QUIC specification defines an optional keep alive process, see Section 5.3 of [QUIC-TRANSPORT]. Implementations of the multipath extension should map this keep alive process to a number of paths. Some applications may wish to ensure that one path remains active, while others could prefer to have two or more active paths during the connection lifetime. **Different applications will likely require different strategies.** Once the implementation has decided which paths to keep alive, it can do so by sending Ping frames on each of these paths before the idle timeout expires.

# Open Issues

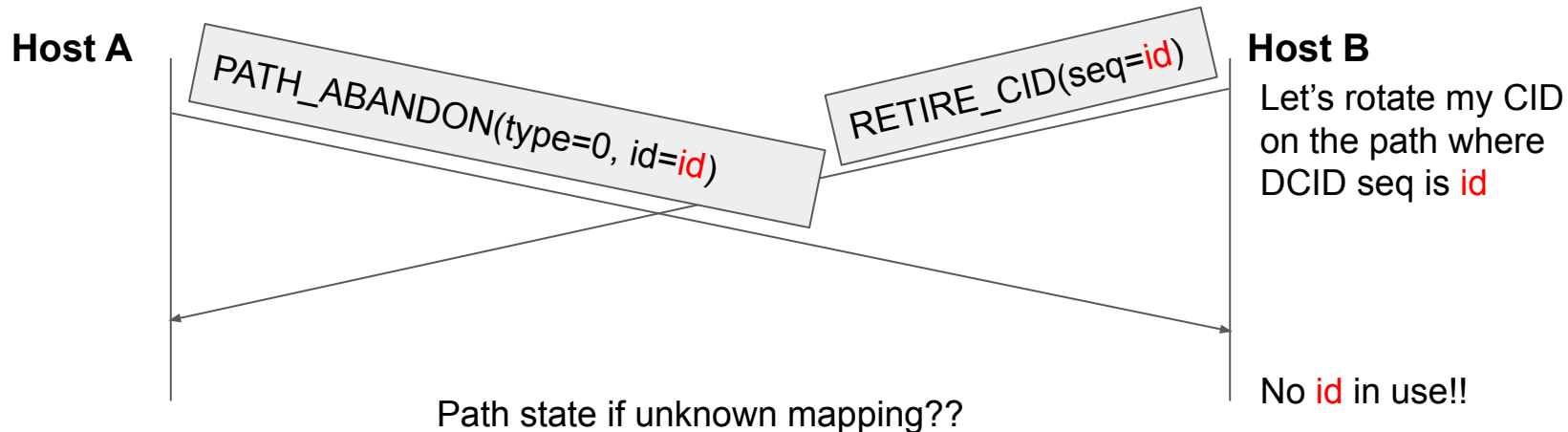
## New Issues:

- How should a receiver process a PATH\_ABANDON having an unknown Path Identifier? #137
- [Do we want to support zero-length connection ID and therefore single packet number space?](#) (PR #149, issues #151& #152)

## Left open after -02:

- [Should servers be allowed to open new paths?](#) #47
- [Sending non-probe packets before path validation complete](#) #50
- Can we be more specific about RTT computation? #132

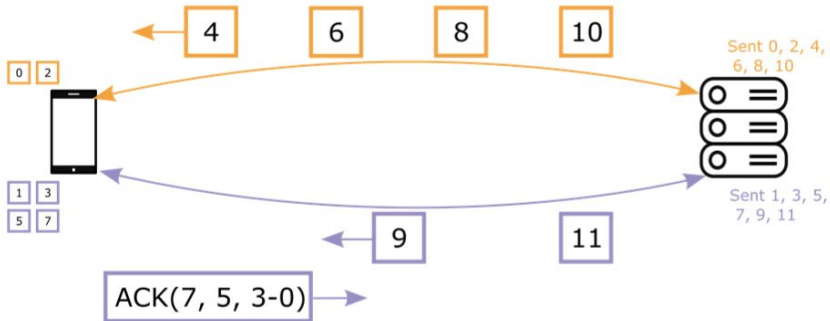
## How should a receiver process a PATH\_ABANDON having an unknown Path Identifier? #137



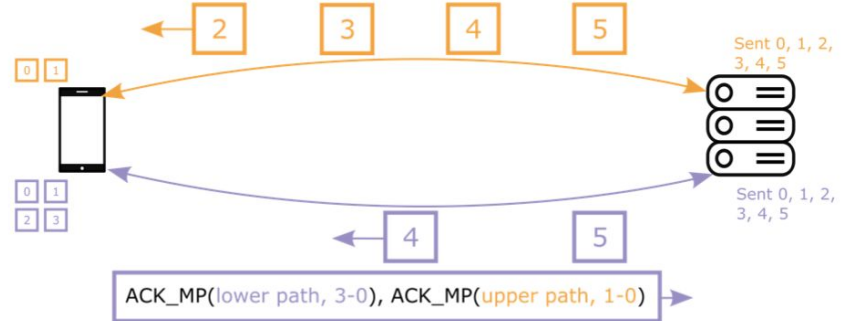
### Current proposal:

- Sender of `RETIRE_CID` should keep track of `id` being mapped to the path for some time
  - Receiver of `PATH_ABANDON` can then map `id` to the right path
- If receiver of `PATH_ABANDON` cannot map `id`, close the connection with `MP_PROTOCOL_VIOLATION` error

# Reminder: SPNS vs. MPNS



(a) Single packet number space case.



(b) Multiple packet number spaces case.

**Figure 1: An example of a 12-packets data transfer where the server follows a round-robin scheduling strategy to send packets. Here, the client acknowledges received packets with a single packet sent on the lower path.**

## Reminder: [First shot at a unified proposal #103](#)

- **Mandatory** support of multiple packet number spaces (and ACK\_MP frame), if multipath extension is negotiated
- Support for zero-length CID (at sender-side) is **optional**
  - Receivers of packets without CID (zero-length) simply set the PN space ID to 0 in ACK\_MP
  - Senders that **use multiple paths and send packets without CID** have to implement additional logic for
    - Loss and congestion handling (section 9.1.2)
    - ACK delay calculation (section 9.1.3)
    - ECN handling (section 9.1.4)
  - Or can only send data on one path at a time

Client SCID	Server SCID	What
long	long	Multiple number spaces
NULL	long	Multiple number spaces on client side (one per CID), single space on server side
long	NULL	Multiple number spaces on server side (one per CID), single space on client side
NULL	NULL	Single number space on each side



# Evaluation Reports between MPNS and SPNS

Key findings (from both reports):

- With #131 changes, RTT estimations in SPNS can be as good as in MPNS
- Even without losses, SPNS can experience large ACK holes
  - Limiting the number of advertised ACK ranges decreases SPNS performance
- Without packet number duplication heuristic change, SPNS receiver may not acknowledge all received packets
- On large transfers, SPNS performance is lower than MPNS
  - Acknowledgement issues, spurious losses

## Question #1

Is there any use-case that requires zero-length CID support in basic multipath?

## Question #2 (related to #149)

Should we keep support of SPNS in the basic multipath draft?

1. Keep the draft as it. SPNS (zero-length CID) support is optional, and there are heuristics for endpoints that do not want to support simultaneous multipath usage with SPNS
2. Use MPNS only
  - a. Keep support for zero-length CID only on the first/initial path (path ID 0). This path may be migrated (i.e., a different 5-tuple seen without CID may be considered as a migration of the path without CID) – see #149
  - b. Remove support of zero-length CID (SPNS) from base draft. A future draft may extend the base one to provide support for zero-length CID

# Next steps

Overall MPQUIC draft starts being stable

- Finalize discussion on remaining open issues
- Further interop testing

# Backup

Old slides on unsolved issues