

Unreliable Datagram Extension to QUIC

draft-pauly-quic-datagram-00

Tommy Pauly, Eric Kinnear, David Schinazi

QUIC

IETF 103, November 2018, Bangkok

Why unreliable QUIC frames?



Several application use cases take advantage of unreliable data

QUIC can provide functionality beyond DTLS or direct UDP

We have an extension mechanism in QUIC—let's try using it!

Advantages over other protocols



Share a single crypto handshake between reliable stream data and unreliable datagrams

Inherit features of the QUIC crypto handshake that may not be present in DTLS (**faster retransmission, transport parameters**)

Adds ability to **acknowledge** datagrams

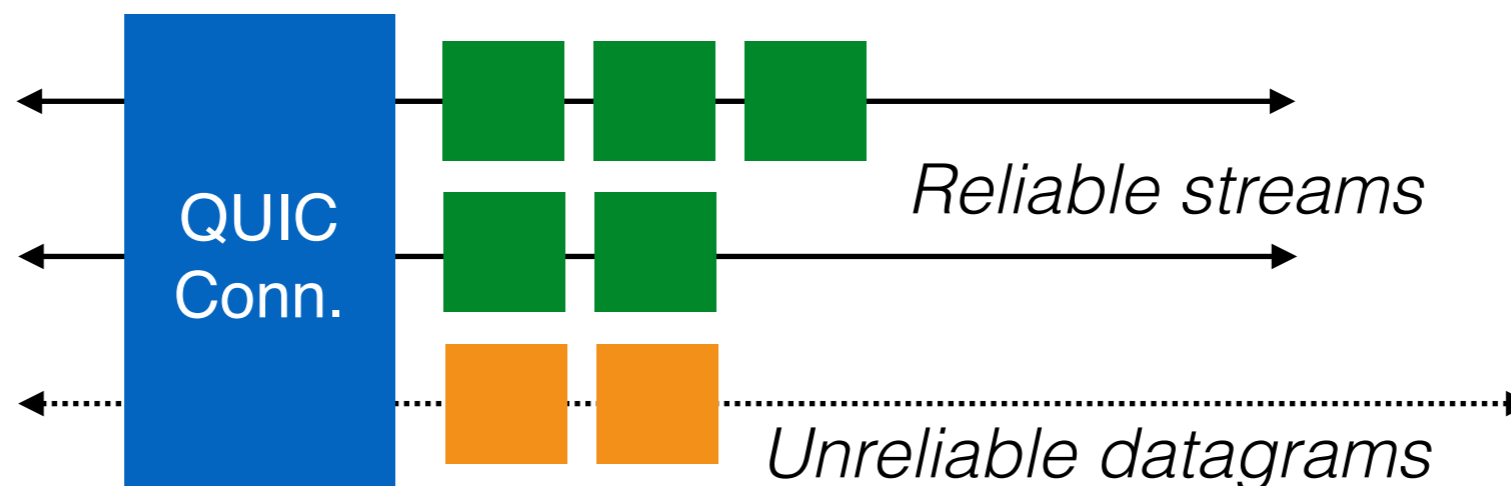


Example use cases

Applications that need to maintain both a reliable control stream and an unreliable frame flow

Audio/Video Streaming, Gaming, etc.

VPN-style tunneling of IP packets over a QUIC connection



Design decisions



Should DATAGRAM frames be acked?

Yes, as this adds more potential for sender to gauge loss on the network

Acknowledgements do not guarantee that the peer application actually processed datagram

Design decisions



Do DATAGRAM frames contribute to connection data limits?

Current draft said yes, on the basis that it seems to violate the overall data limit otherwise

List discussion highlighted that correctly interpreting flow control for datagrams adds too much complexity

Any reasons not to remove this flow control?



Design decisions

Can there be multiple flows (ID space) of DATAGRAM frames?

No, as applications can add identifiers within frames themselves. IDs are more important for STREAM frames where there is per-stream flow control.

Next steps



Feedback welcome and encouraged!

Not intended for initial QUIC version (don't want to disrupt the schedule)

Seems like a good test of extension mechanism for new frame types

