

Track Alias

Issues #350 and #597

What the heck is a Track Alias anyways?

1. To function correctly, MoQT requires *at minimum*, a track identifier that is unique per hop(session)
 - a. Used to correlate received objects to tracks/subscriptions
2. Some *systems* that want to run over MoQT already have a track identifier that is **more unique** than per-hop
 - a. For example, an application-specific globally unique identifier, or a hash of the track name
 - b. These systems require this ID to be carried with track data for use by intermediate nodes
 - i. eg: for load balancing, routing, logging
3. MoQ Transport's Track Alias feature allows applications choose their own identifier

Status Quo

1. Subscriber chooses initial Track Alias
2. Publisher can reject with `RETRY_TRACK_ALIAS` and a value it likes better
 - a. Adds 1-RTT to `SUBSCRIBE` for clients without foreknowledge
 - b. My understanding is that no one intends to use this mechanism. Either:
 - i. Application has no need of this
 - ii. The subscribers have out-of-band knowledge of the preferred alias.

How did we get here?

- Original MoQT draft had the Track Alias (then called Track ID) chosen by the publisher
- Concerns about buffering objects with unknown Track Alias *before* OK arrives
- October 2023 -> current compromise
- For a while, Objects had both Track Alias and Subscribe ID while we figured out dedup
 - Current draft only has Alias for SUBSCRIBE (no dups) and Subscribe ID for FETCH (dups allowed)
Note: Issue #350 was reopened as an objection to this

Why should we talk about this?

1. Stumbling block for newcomers to the protocol
 - a. At best, we've done a poor job articulating what this is and why it's in MoQT
2. Complaints and issues raised to chairs by implementers since the compromise
 - a. Alias doesn't provide proven value while adding non-zero complexity
 - b. The current compromise is awkward and requires a round-trip
 - c. Alias creates edge cases with overlapping subscribes to the same track

Why shouldn't we talk about this?

The design in MoQT will not win any beauty contests but there are no functional concerns

Reopening this design will extend the time it takes to reach WGLC

Question 1: Do you want to talk about this?

No = Stick with the status quo, end presentation and move to next topic

Yes = Consider alternatives, including

- A proposal that retains the feature but reduces the warts
- Your alternative here

Track Alias Alternate Proposal Goals

1. Remove 1-RTT time-to-first-object penalty for tracks with publisher chosen Track Alias
2. Simplify subscriber logic
3. Specify a design that everyone implements
4. Enable possible “pub-before-sub” use cases

Track Alias Alternate Proposal

1. Remove the Track Alias field from SUBSCRIBE and the RETRY_TRACK_ALIAS error code
2. Require Track Alias to be odd and Subscribe ID to be even
3. Make Track Alias an exclusively Publisher-chosen *optional* field in SUBSCRIBE_OK
4. Replace the Track Alias field in Object and Subscribe ID field in SUBSCRIBE_DONE with `Track Identifier`. A Track Identifier can be either a Subscribe ID or a Track Alias, determined by LSB.
5. Add a new message ALIAS_ACK, which acknowledges to the publisher that the subscriber has received the Subscribe ID/Track Alias mapping in SUBSCRIBE_OK. The subscriber MUST send this message if it receives an OK with a Track Alias.
6. If a subscriber receives an Object with an unknown Track Identifier, it MAY drop it.

Example Message Flow

S->P: SUBSCRIBE (sid=0)

P->S: SUBSCRIBE_OK (sid=0, alias=1)

P->S: OBJECT 1 (identifier=0)

P->S: OBJECT 2 (identifier=0)

S->P: ALIAS_ACK (sid=0)

P->S: OBJECT 3 (identifier=1)

P->S: SUBSCRIBE_ID (identifier=1) // identifier=0 is also fine

Sketch of Pub-Before-Sub

```
PUBLISH
{
  Type (i) = 0xTBD1,
  Length (i),
  Track Alias (i),
  Track Namespace (tuple),
  Track Name Length (i),
  Track Name (..),
  Group Order (8),
  ContentExists (8),
  [Largest Group ID (i)],
  [Largest Object ID (i)],
  Number of Parameters (i),
  Subscribe Parameters (..) ...
}
```

```
PUBLISH_OK
{
  Type (i) = 0xTBD2,
  Length (i),
  Track Alias (i)
  Subscribe ID (i),
  Subscriber Priority (8),
  Number of Parameters (i),
  Subscribe Parameters (..) ...
}
```

This is not part of the current proposal, just a sketch of how this could work

Discussion

At the end of discussion, I'd like a show of hands

1. Should the wg keep the Status Quo or make a change?
2. If we make a change, should we use something along the lines of the alternate proposal?