ACK and ACK_ECN

QUIC Interim, New York, September 2018

Issues and Drawbacks with current design

- ACK_ECN increases in size as the number of marks increase over a connection
- No indication of what packets were marked
  - Creates an edge case with ACK loss where sum of marks is larger than number of packets acked. (Fixed issue #1481)
- Separate ACK and ACK_ECN frames duplicates work

Original PR #1372

QUIC

Option 1: Status Quo

3 packet counters for ECT(0), ECT(1), and ECN-CE

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        ECT(0) Count (i)                     ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        ECT(1) Count (i)                     ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        ECN-CE Count (i)                     ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

QUIC

Option 2: Add a 'type' to each ACK block

2 bits from each ACK block taken to indicate type.

Cons: Inefficient for AccECN style of marking

Lose 2 bits of ack or gap length (64 -> 16)

Issue#1439, PR#1706

Option 3: Bitvector

A bit vector indicates the ECN marks of acked packets

One ACKed packet consumes 2 bits of the bit vector:

    non-ECT, ECT(0), ECT(1), ECN-CE

Con: ACK loss causes missed marks unless bitvector is large

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Bitvector Length (i)     |        Bitvector (*)        ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

QUIC

Option 4: Optimize Status Quo

Size can be reduced by using an epoch encoding/decoding because ECN marks always increase

Merge 2 ACK frames by indicating the least significant bit of the type indicates whether the ECN section is present

Cons: Doesn't remove edge case with ACK loss

Wraparound is more complex to implement

QUIC