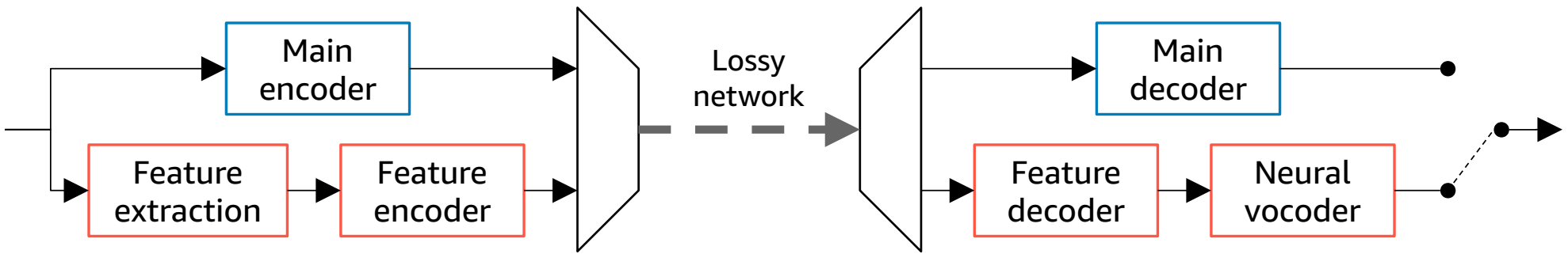
# Deep Redundancy for the Opus Codec
## `draft-valin-opus-dred-05`

Presenter: Jean-Marc Valin
jmvalin@jmvalin.ca

IETF 119

# DRED Recap

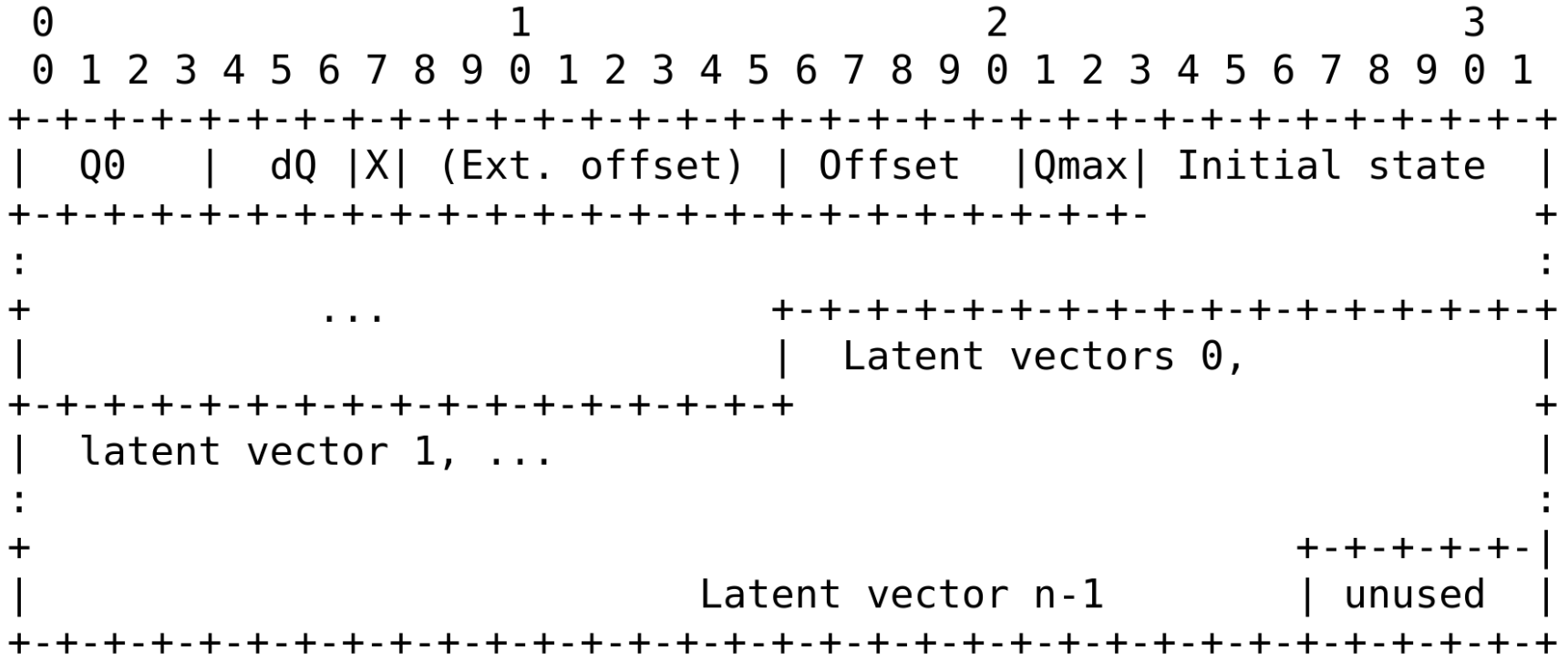
- Code large amounts of redundant audio in Opus packet
  - Use DNN to maximize compression
  - Can code 1 second per 20-ms packet (50x)



# Changes Since IETF 118

- Added extended offset
  - Optional extra byte signals up to 20 second offset
  - Used to trim silence (more efficient, better DTX)
  - Potential use in SFU just after switch
- Added Qmax field
  - Makes it possible to cap the quantizer
  - Variable size
  - Costs one bit when unused

# Format


0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Q0
dQ
X
(Ext. offset)
Offset
Qmax
Initial state
...
Latent vectors 0,
latent vector 1, ...
Latent vector n-1
unused

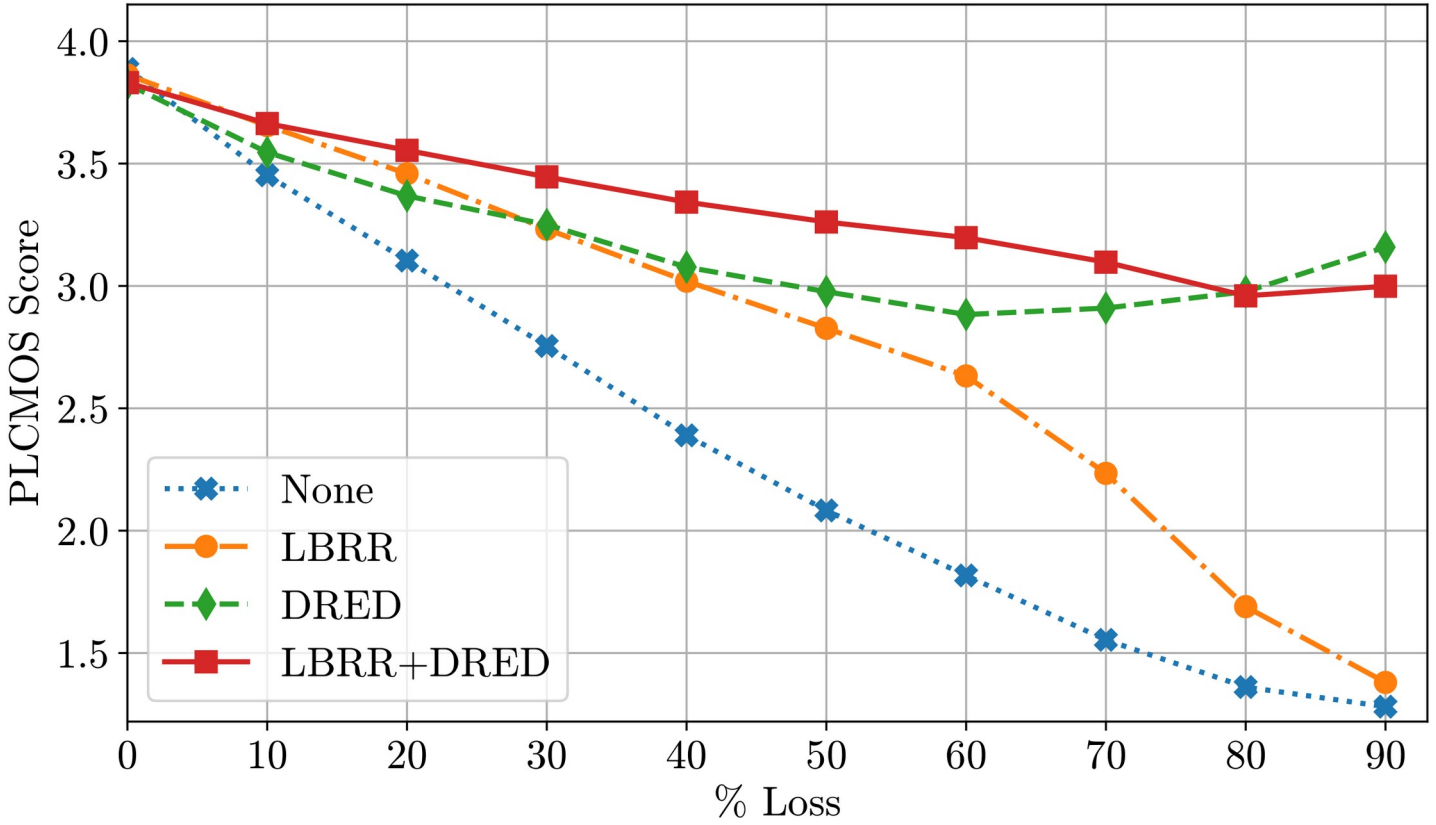
# Normative Aspects

- Normative spec for bits-to-feature decoder
  - All decoder weights are frozen
    - Publish as simple binary format
  - Definition of the acoustic features
    - How do we specify neural pitch estimator?
- Encoder is left unspecified
- Minimal constraints on vocoder

# Implementation Update

- Changes landed in new Opus main branch
- Released latest implementation in Opus 1.5
  - https://www.opus-codec.org/demo/opus-1.5/
  - DRED disabled by default
- WebRTC patch set with DRED support
  - https://github.com/xiph/webrtc-opus-ng/

# Results


PLCMOS Score
4.0
3.5
3.0
2.5
2.0
1.5
0
10
20
30
40
50
60
70
80
90
% Loss
None
LBRR
DRED
LBRR+DRED

# Open Questions

- Should there be a maximum duration allowed?
  - Technically we could do up to ~10 minutes
  - Proposal: no hard limit, since receiver can ignore the rest
- What are the lowest and highest useful bitrates?
  - Currently support 10 to 100 kb/s for 1 second redundancy
  - Equivalent to 200 b/s to 2 kb/s effective bitrate