



# The Opus Audio Codec

---

**Jean-Marc Valin**

**Koen Vos,**

**Gregory Maxwell, and**

**Timothy B. Terriberry**



# Outline



- **Introduction and Motivation**
- Opus Design
  - SILK
  - CELT
- Results
- WebRTC



# Lossy Audio Codecs



- Two common types:
  - Speech/communication (G.72x, GSM, AMR, Speex)
    - Low delay (15-30 ms)
    - Low sampling rate (8 kHz to 16 kHz): limited fidelity
    - No support for music
  - General purpose (MP3, AAC, Vorbis)
    - High sampling rates (44.1 kHz or higher)
    - "CD-quality" music
    - High-delay (> 100 ms)
  - We want both: high fidelity with *very* low delay



# Coding Latency



- Low delay is critical to live interaction
  - Prevents collisions during conversation
  - Reduce need for echo cancellation
    - Good for small, embedded devices without much CPU
  - Higher sense of presence
  - Allows synchronization for live music
    - Need less than 25 ms *total* delay (Carôt 2006)
    - Equivalent to sitting 8 m apart (farther requires a conductor)
- Lower delay in the codec increases range
  - 1 ms = 200 km in fiber

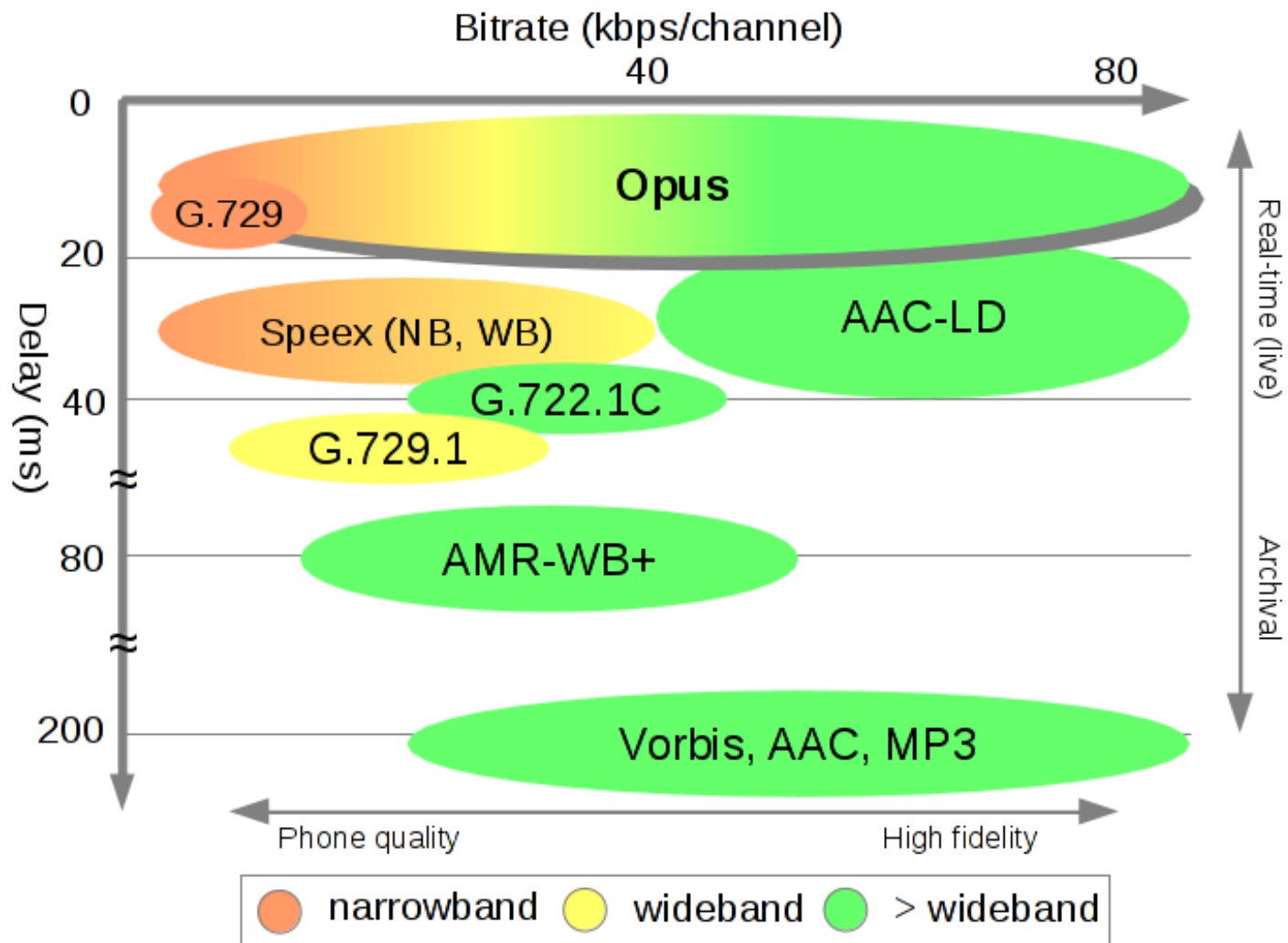
High delay  
(~250 ms)

Low delay  
(~15 ms)



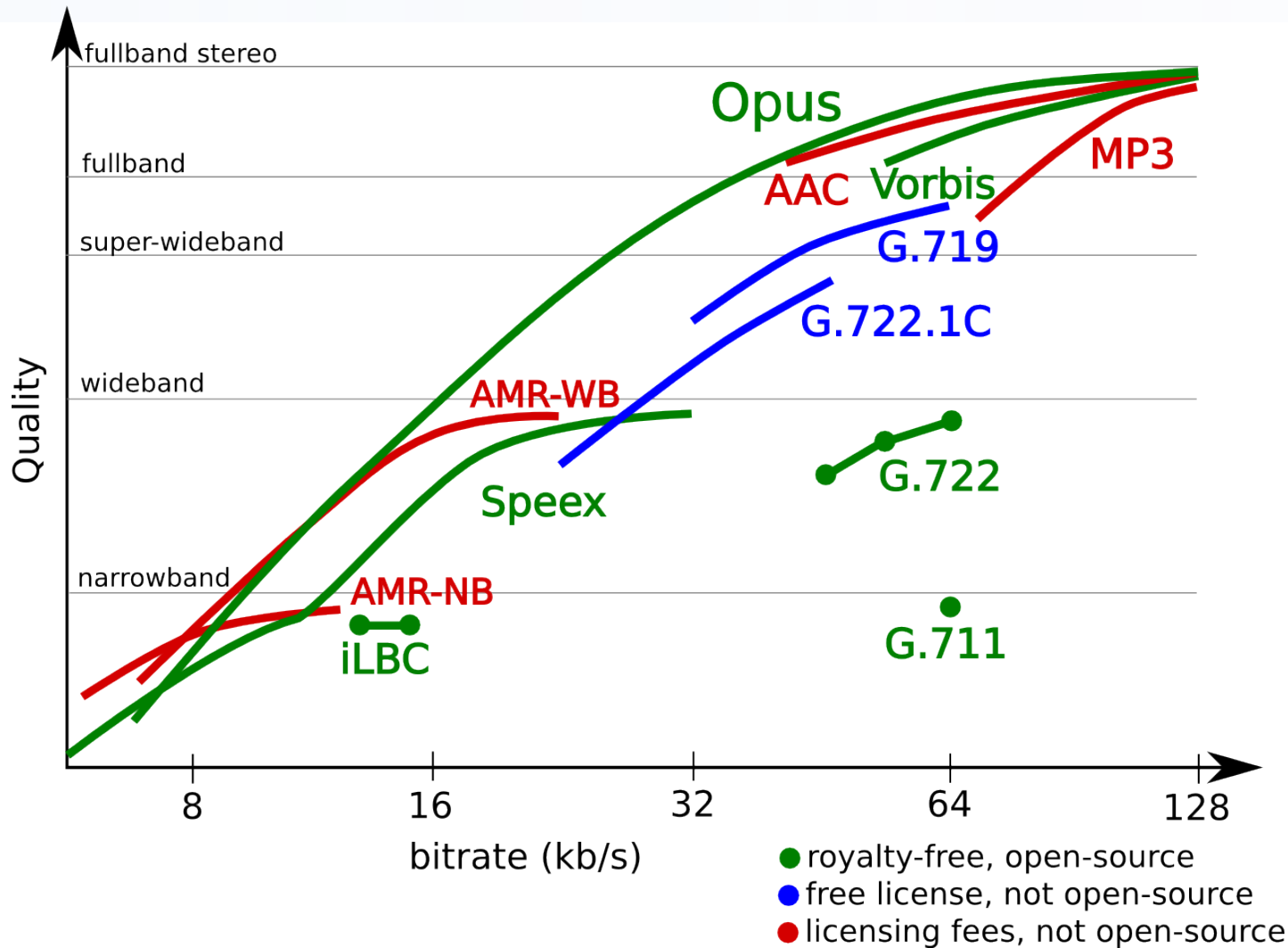


# Opus vs. the Competition: Latency





# Opus vs. the Competition: Quality





# Opus Features



- Sampling rate: 8...48 kHz (narrowband to fullband)
- Bitrates: 6...510 kbps
- Frame sizes: 2.5...20 ms
- Mono and stereo support
- Speech and music support
- Seamless switching between all of the above
- Combine multiple streams for up to 255 channels
- It just works for everything



Adaptive sweep: 8...64 kbps



# Outline



- Introduction
- **Opus Design**
  - SILK
  - CELT
- Results
- WebRTC





# Opus Characteristics



- Standardized by the IETF (RFC 6716)
  - First free, state-of-the-art audio codec standardized
- Built out of two separate codecs
  - SILK: a *linear prediction* (speech) codec
    - In-development by Skype (now Microsoft) since Jan. 2007
  - CELT: an *MDCT* (music) codec
    - In-development by Xiph since November 2007
  - Both were modified a *lot* to form Opus
    - Standardization saw contributions from Mozilla, Microsoft (Skype), Xiph, Broadcom, Octasic, Google, etc.



# History



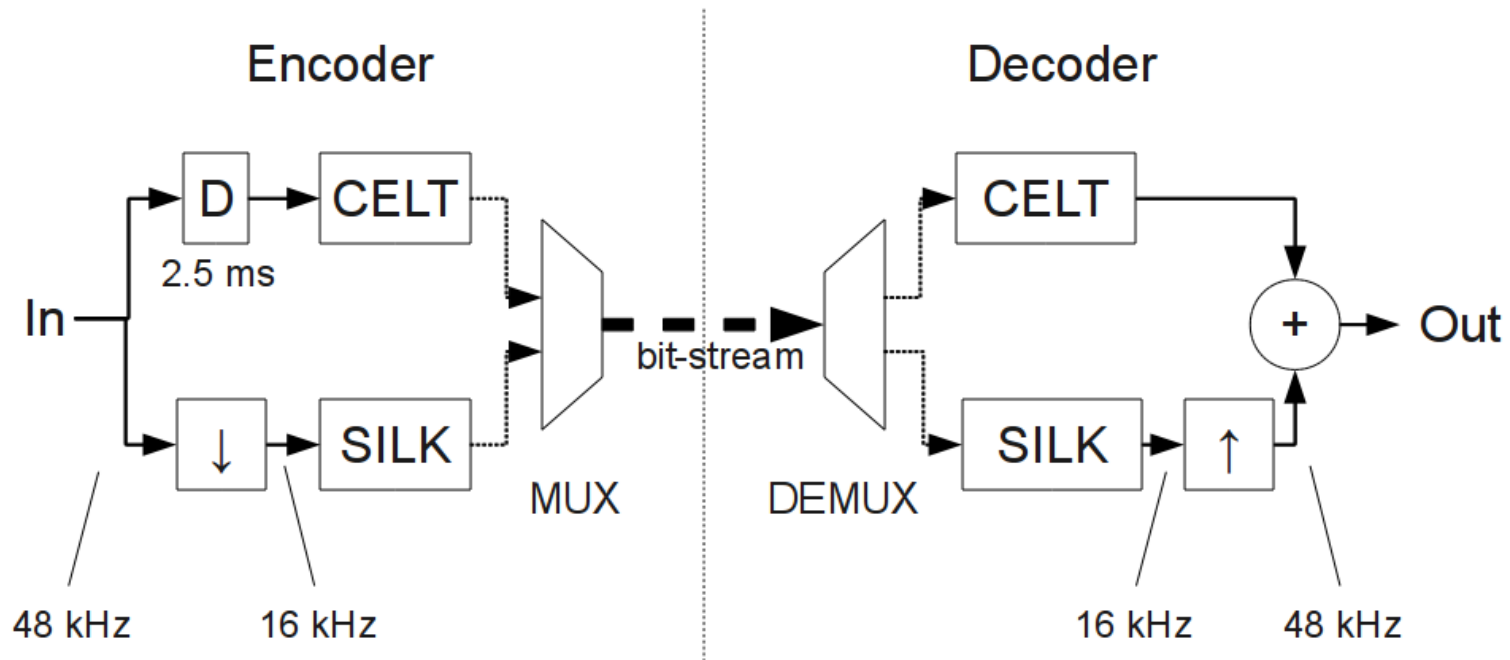
- Jan. 2007: SILK project started at Skype
- Nov. 2007: CELT project gets started
- Mar. 2009: Skype asks IETF to create a codec WG
- Feb. 2010: WG created
- Jul. 2010: First prototype of SILK+CELT hybrid codec
- ~Dec 2011: Opus surpasses Vorbis and HE-AAC
- Jan. 2012: last bit-stream changes, 2<sup>nd</sup> WGLC
- Sep. 2012: Opus becomes RFC 6716



# Opus Operating Modes



- **SILK-only:** Narrowband (NB), Mediumband (MB) or Wideband (WB) speech
- **Hybrid:** Super-wideband (SWB) or Fullband (FB) speech
- **CELT-only:** NB to FB music





# Outline



- Introduction
- Opus Design
  - **SILK**
    - Linear Prediction
    - Short-term Prediction (LPCs)
    - Long-term Prediction (LTP)
  - CELT
- Results
- WebRTC



# SILK



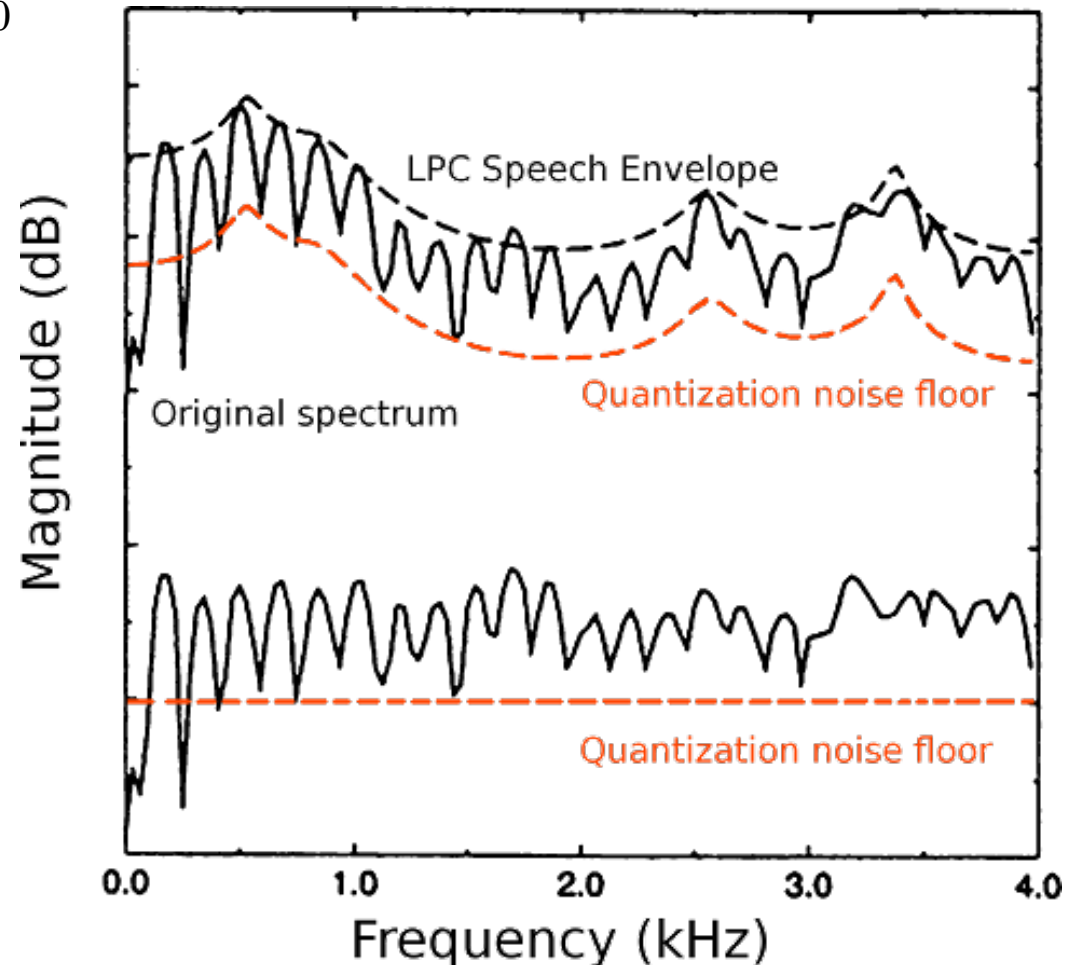
- Linear prediction
  - Short-term prediction via a linear IIR filter
    - 10 or 16 coefficients (for NB or MB / WB respectively)
    - Good for speech: filter coefficients directly related to cross-sectional area of human vocal tract
  - Long-term prediction via a “pitch” filter
    - Good for “periodic” signals from 55.6 Hz to 500 Hz
- Variable bitrate
  - Quantization level controls rate indirectly
  - Range (arithmetic) coding with fixed probabilities



# Linear Prediction



- IIR filter:  $y[i] = x[i] + \sum_{k=0}^{D-1} a[k]y[i-k-1]$
- Analysis “whitens” a signal
- Quantization (lossy compression) adds noise
- Synthesis “shapes” the noise the same as the spectrum

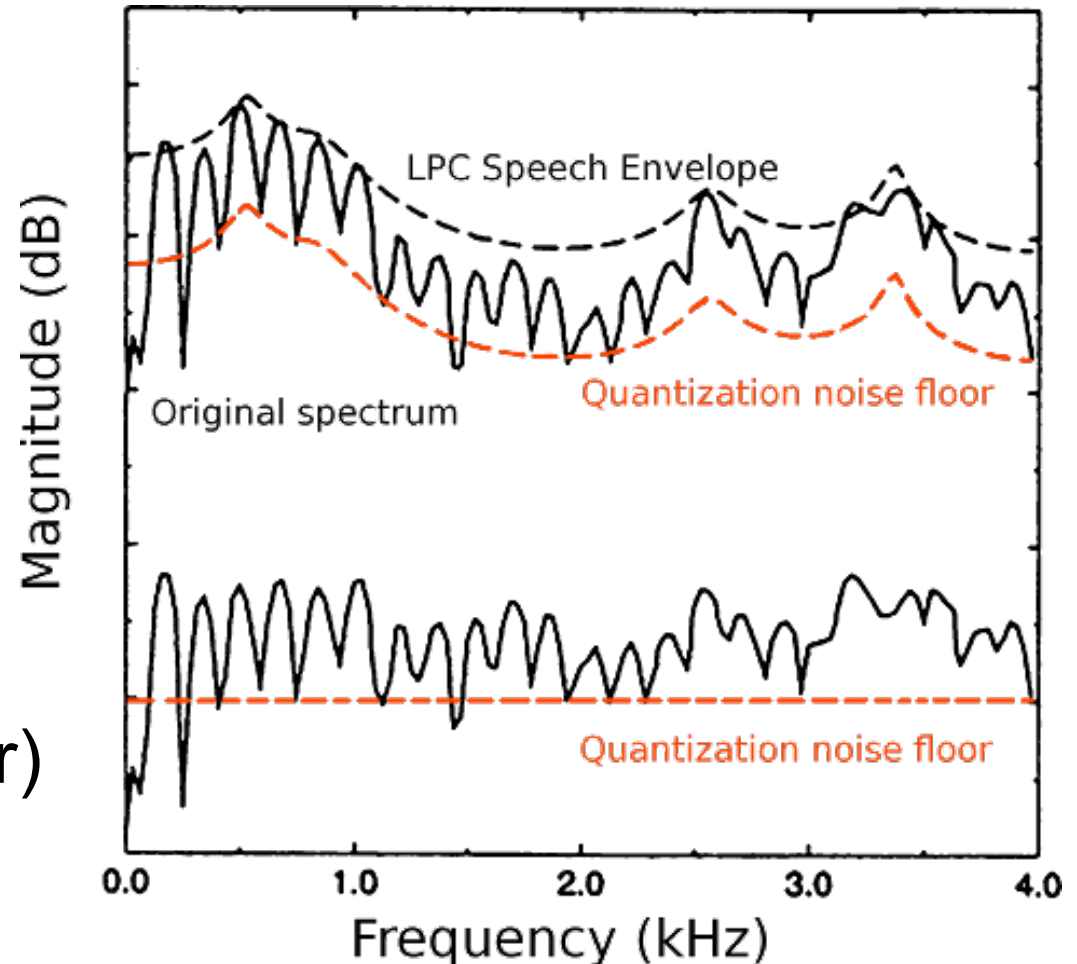




# Linear Prediction



- SILK: different analysis and synthesis filters
- De-emphasizes spectral valleys
  - Distortion least noticeable there
  - Reduces entropy (distance between signal and noise floor)
    - Uses fewer bits





# Handling Packet Loss



- LTP uses decoded signal from previous frames (up to 18 ms back)
  - Packet loss causes mis-prediction in future frames
- SILK: Artificially scale down previous frames
  - Uses less prediction (more bits) for the first period
    - But *only* affects the first pitch period
  - Amount depends on packet loss: signaled in bitstream (1.5 bits on average)





# Excitation



- Pyramid Vector Quantization (Fischer 1986)

$$S(N, K) = \{ \mathbf{y} \in \mathbb{Z}^N : \sum_{i=1}^N |y_i| = K \}$$

- $N$  fixed at 16,  $K$  between 0 and 16 (signaled)
  - For higher rates, add 1 bit per coefficient, uncoded
- PVQ ideal for Laplace-distributed data, LPC residuals more Gaussian
  - Recursively split vector in half
  - Trained probabilities to signal how many pulses in each
- Random quantization bias added to each value
  - Highly biases sign of coefficient, less sparse at same rate



# Outline



- Introduction
- Opus Design
  - SILK
  - **CELT**
    - "Lapped Transform"
    - "Constrained Energy"
    - Coding Band Shape
    - Psychoacoustics
- Results
- WebRTC



# CELT: "Constrained Energy Lapped Transform"



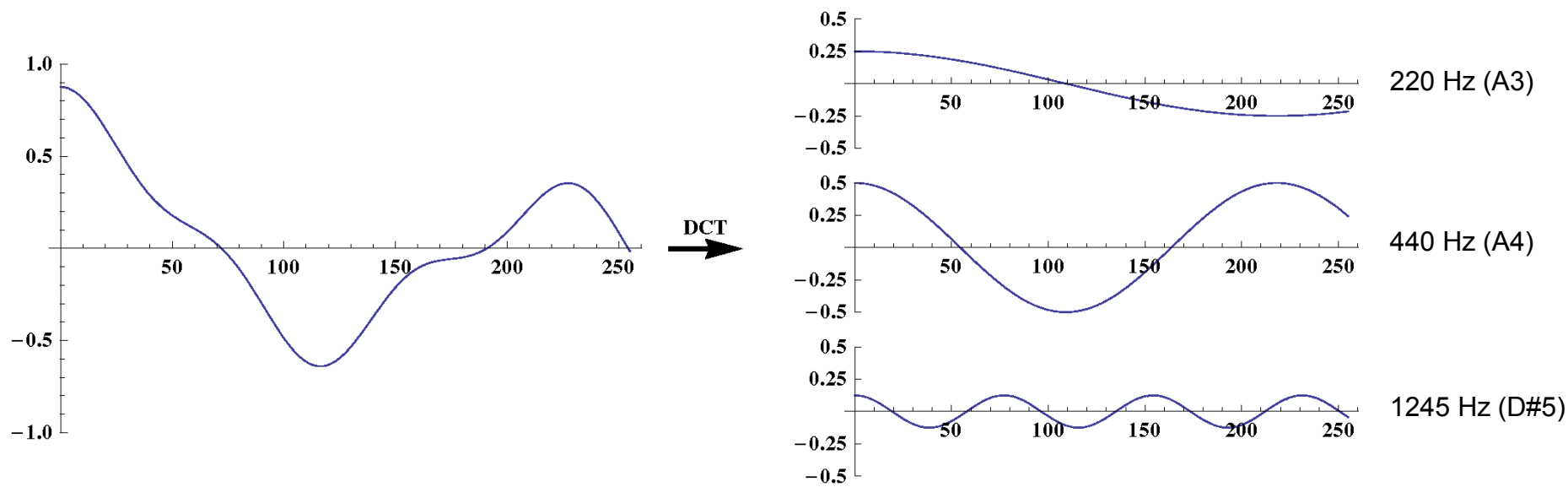
- Transform codec (MDCT, like MP3, Vorbis)
  - Short windows → poor frequency resolution
- *Explicitly code energy of each band of the signal*
  - Coarse shape of sound preserved no matter what
- Code remaining details using algebraic VQ
- Useful for 40 kbps and above
  - Not good for low bitrate speech



# "Lapped Transform" Time-Frequency Duality



- *Any* signal can be represented as a weighted sum of cosine curves with different frequencies
- The Discrete Cosine Transform (DCT) computes the weights for each frequency



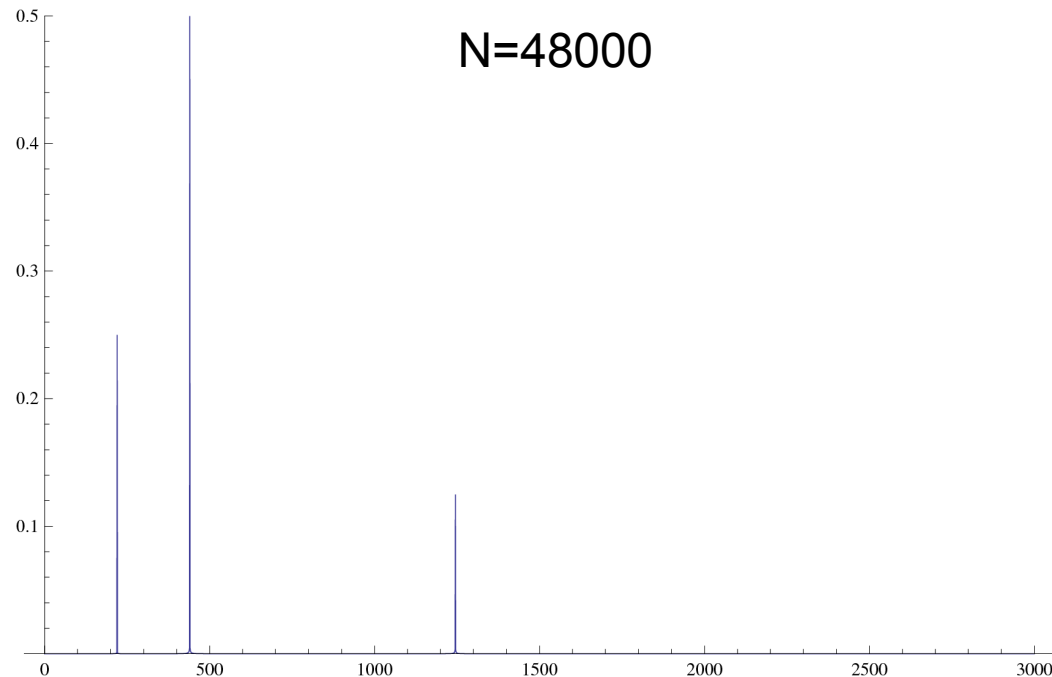


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (harder to compress)



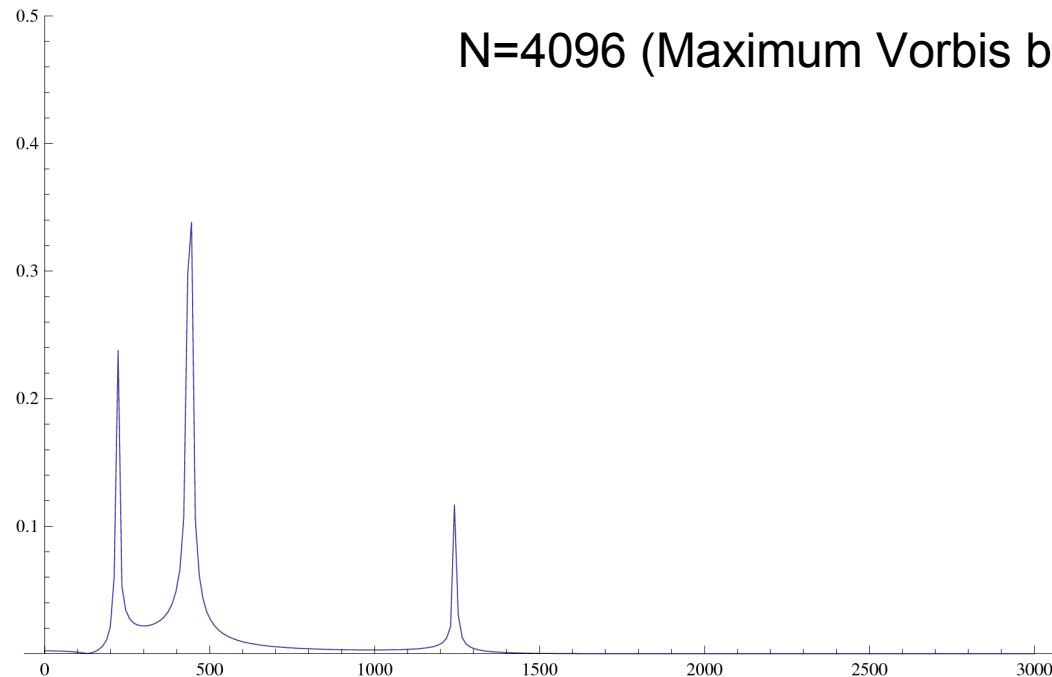


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (harder to compress)



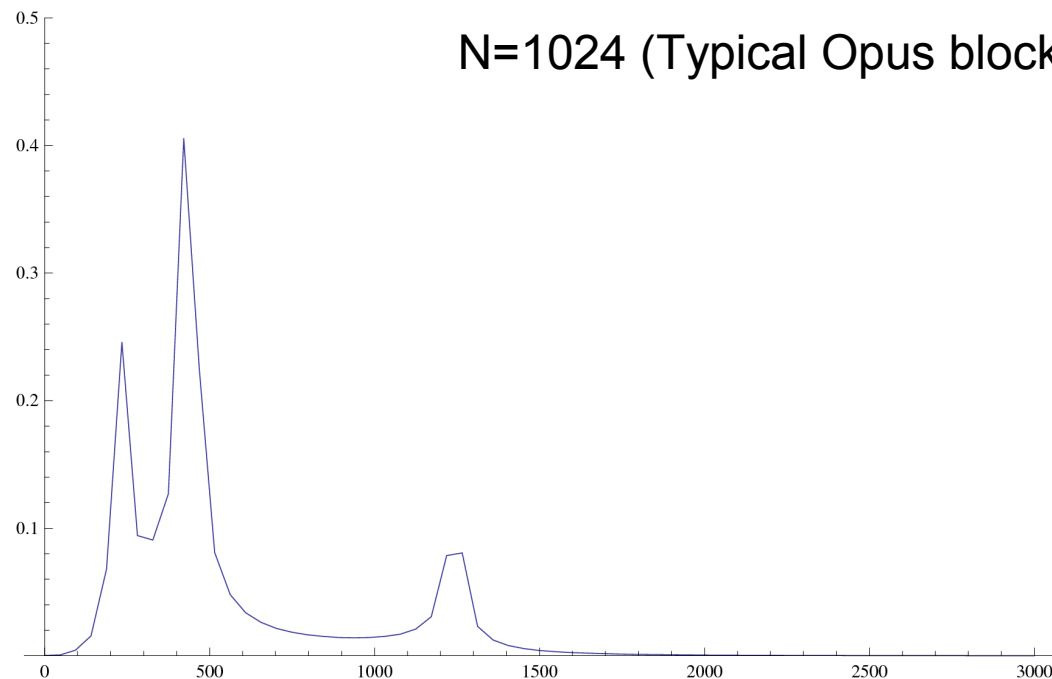


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (harder to compress)



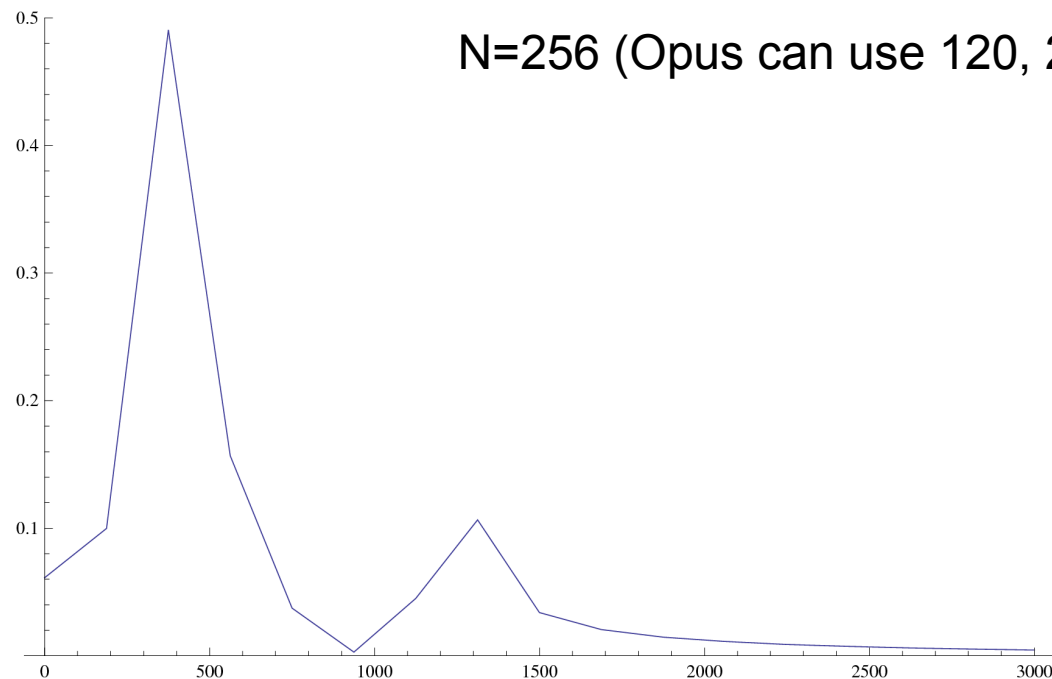


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (harder to compress)





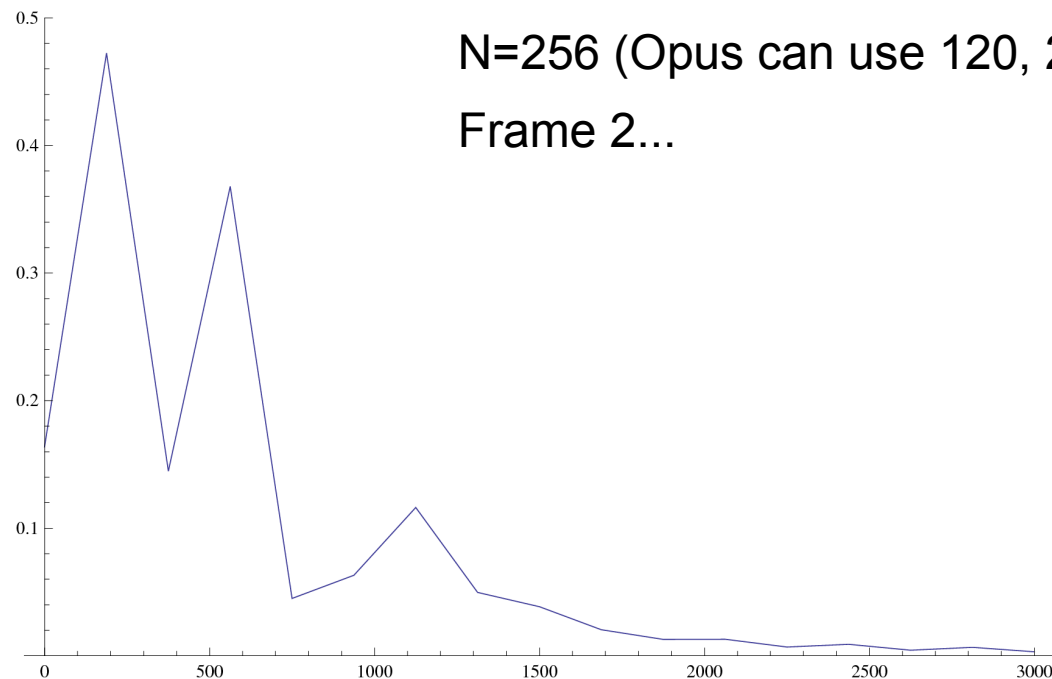


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (unstable over time)



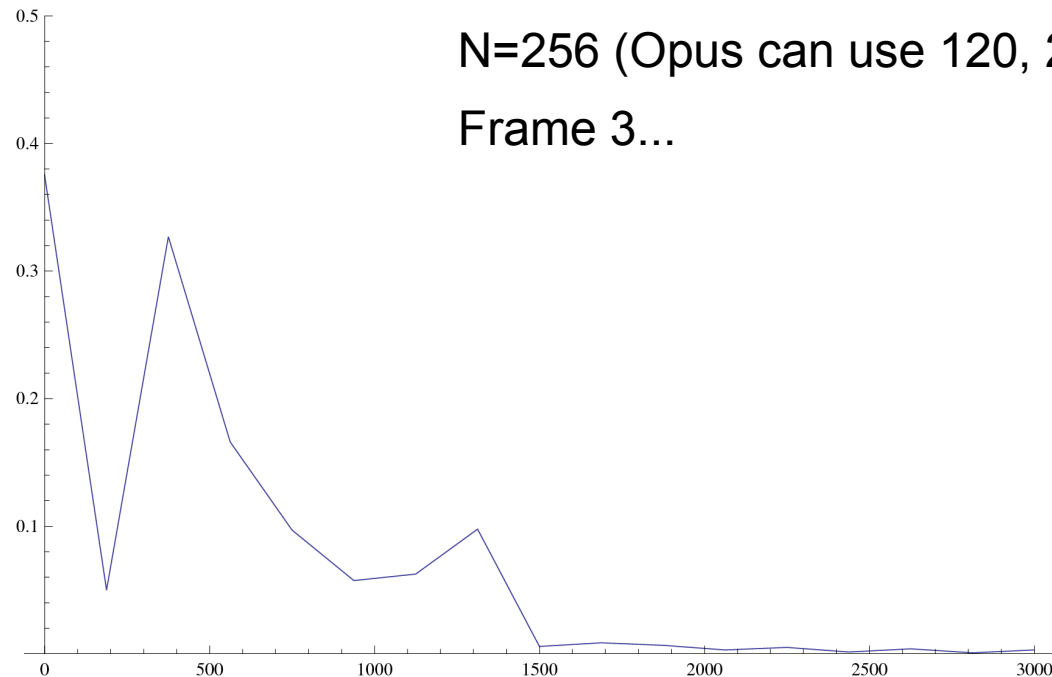


# "Lapped Transform"

## Discrete Cosine Transform



- The "Discrete" in DCT means we're restricted to a finite number of frequencies
  - As the transform size gets smaller, energy "leaks" into nearby frequencies (unstable over time)

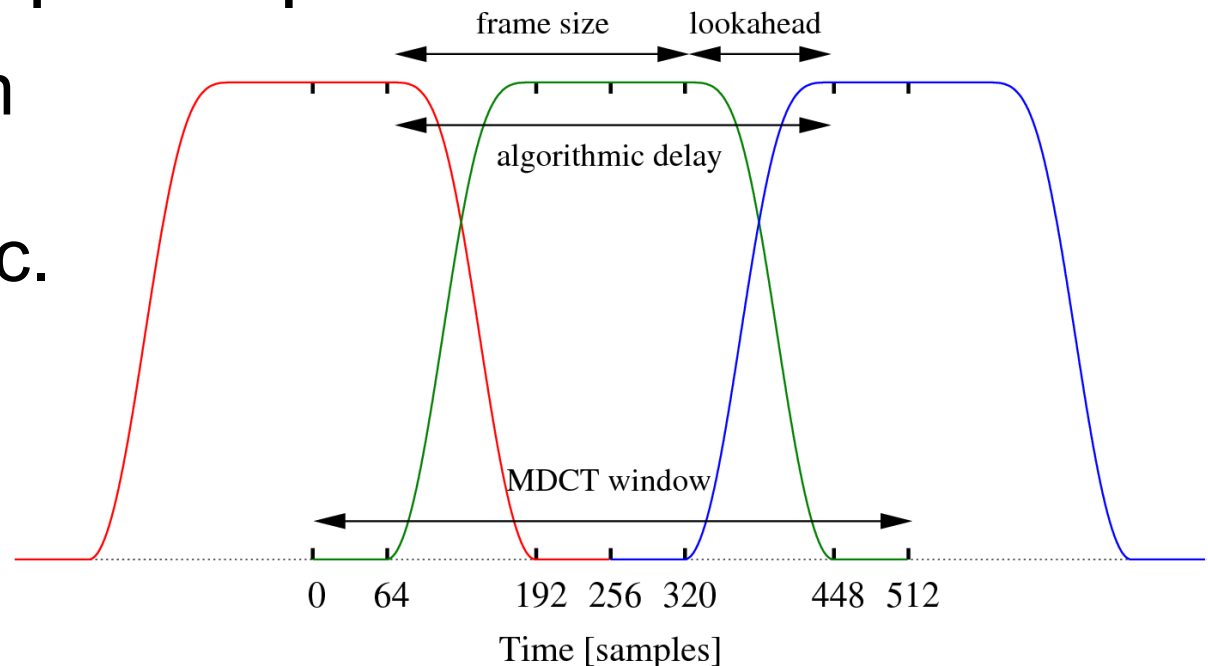




# "Lapped Transform" Modified DCT



- The normal DCT causes coding artifacts (sharp discontinuities) between blocks, easily audible
- The "Modified" DCT (MDCT) uses a decaying window to overlap multiple blocks
  - Same transform used in MP3, Vorbis, AAC, etc.
  - But with much smaller blocks, less overlap

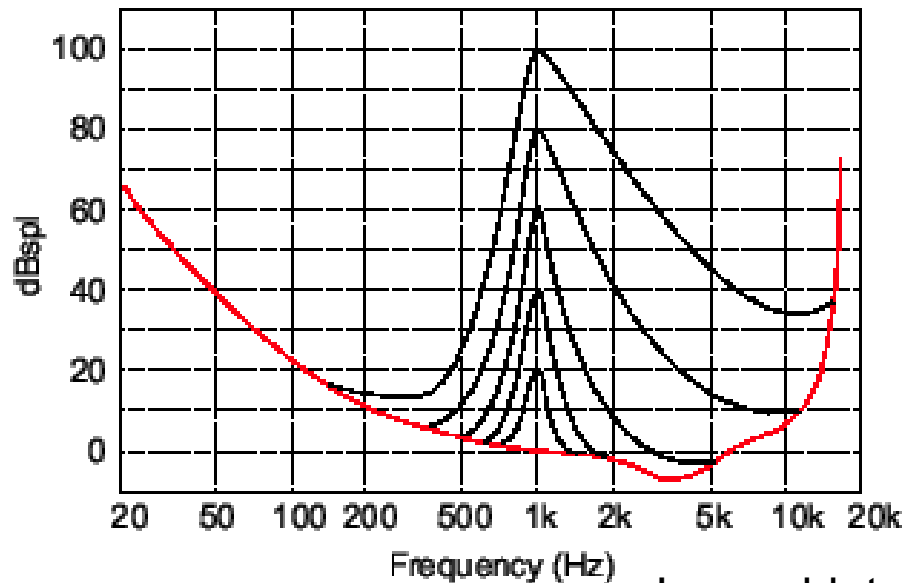




# "Constrained Energy" Critical Bands



- The human ear can hear about 25 distinct "critical bands" in the frequency domain
  - Psychoacoustic masking within a band is much stronger than between bands



Threshold of detection in the presence of masker at 1kHz with a bandwidth of 1 critical band and various levels.

Image blatantly stolen from  
<http://www.tonmeister.ca/main/textbook/node331.html>

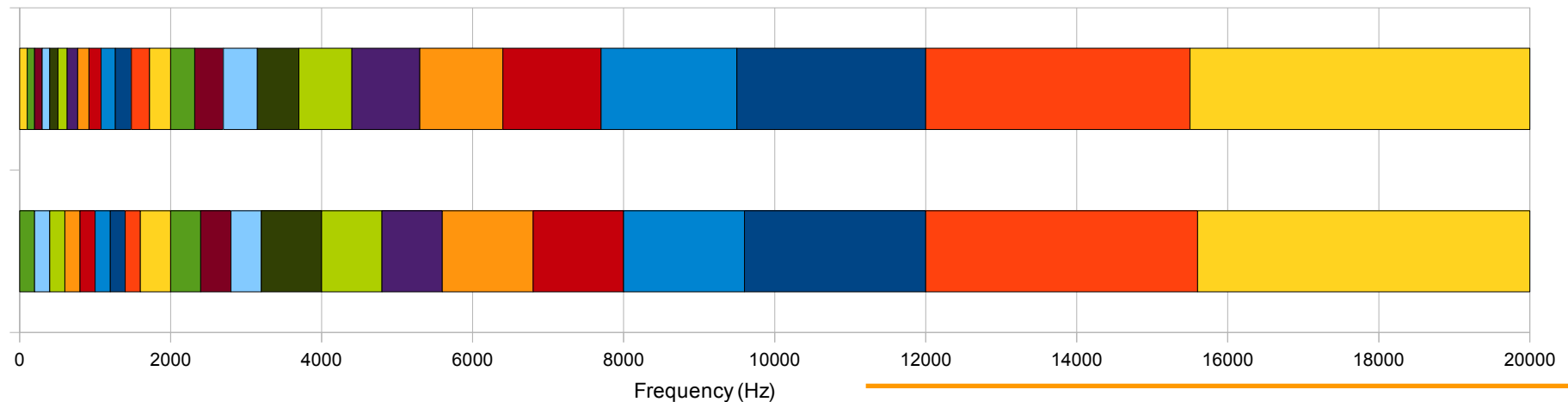


# "Constrained Energy" Critical Bands



- Group MDCT coefficients into bands approximating the critical bands (Bark scale)
  - Band layout the same for all frame sizes
    - Need at least 1 coefficient for 120 sample frames
    - Corresponds to 8 coefficients for 960 sample frames
  - Insufficient frequency resolution for all the bands

Bark Scale vs. CELT





# "Constrained Energy" Coding Band Energy



- Most important psychoacoustic lesson learned from Vorbis:


*Preserve the energy in each band*

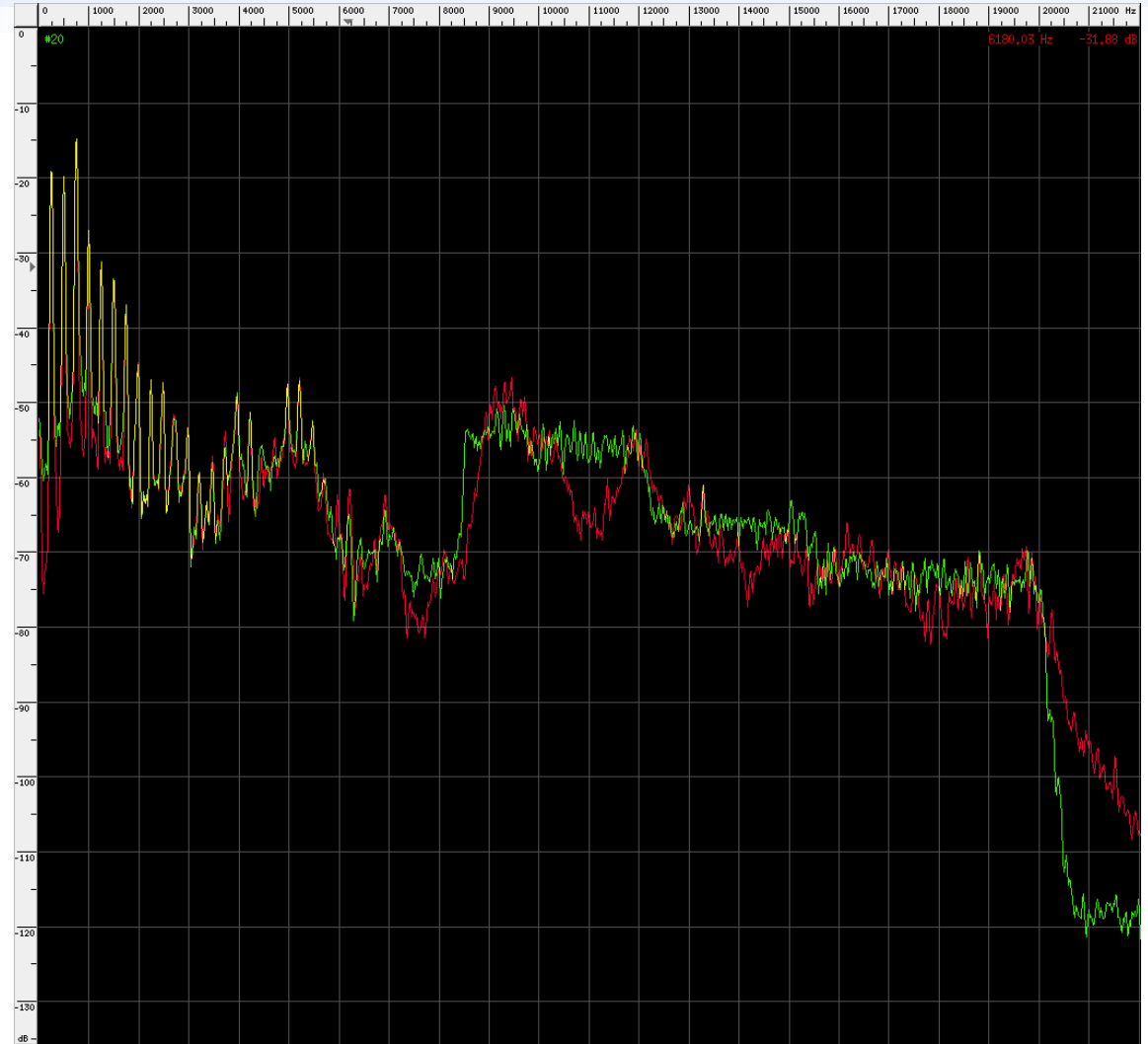
- Vorbis does this implicitly with its "floor curve"
- CELT codes the energy explicitly
  - Coarse energy (6 dB resolution), predicted from previous frame and from previous band
  - Fine energy, improves resolution where we have available bits, not predicted



# "Constrained Energy" Coding Band Energy



- CELT (green) vs original (red)
  - Notice the quantization between 8.5 kHz and 12 kHz
  - Speech is intelligible using coarse energy alone (~9 kbps for 5.3 ms frame sizes) 





# Coding Band Shape



- After normalizing, each band is represented by an  $N$ -dimensional unit vector
  - Point on an  $N$ -dimensional sphere
  - Describes "shape" of energy within the band
- CELT uses *algebraic* vector quantization
  - Have lots of codebooks (# dims, bitrates)
  - Very large codebooks (exponential in # of dims)
    - 50 dims at 0.6 bits/dim is over 1 billion codebook entries
  - But we're coding uniformly distributed unit vectors





# Coding Band Shape

# Algebraic Vector Quantization

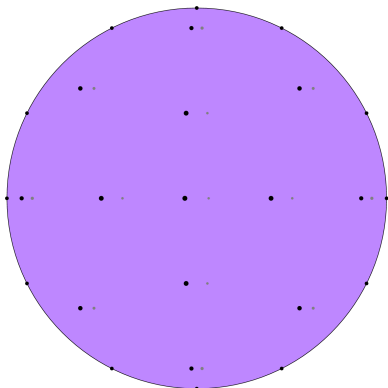
- Use a regularly structured, algebraic codebook: Pyramid Vector Quantization (Fischer, 1986)
  - We want evenly distributed points on a sphere
    - Don't know how to do that for arbitrary dimension
  - Use evenly distributed points on a pyramid instead
- For  $N$  dimensional vector, allocate  $K$  "pulses"
- Codebook: normalized vectors with integer coordinates whose magnitudes sum to  $K$

$$S(N, K) = \left\{ \frac{\mathbf{y}}{\|\mathbf{y}\|} \in \mathbb{Z}^N : \sum_{i=1}^N |y_i| = K \right\}$$

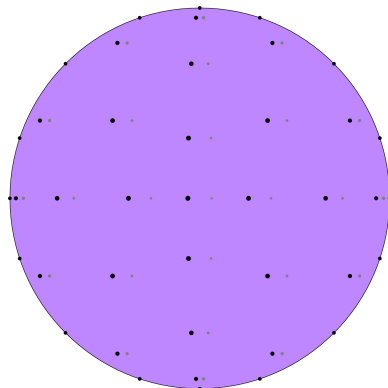


# Coding Band Shape

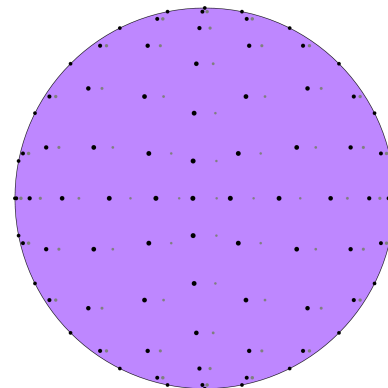
## N=3 at Various Rates



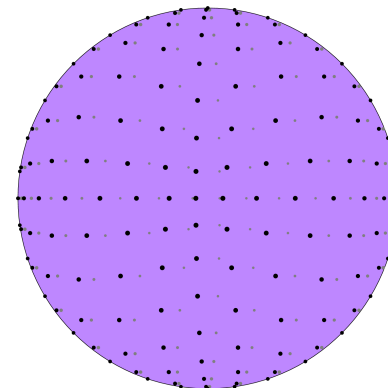
5.25 bits (K=3)



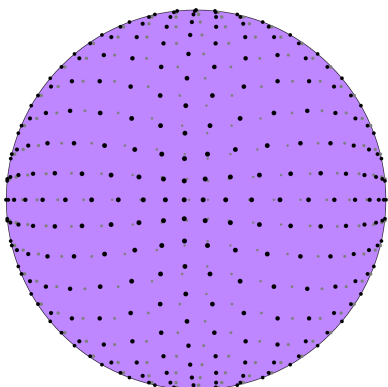
6.04 bits (K=4)



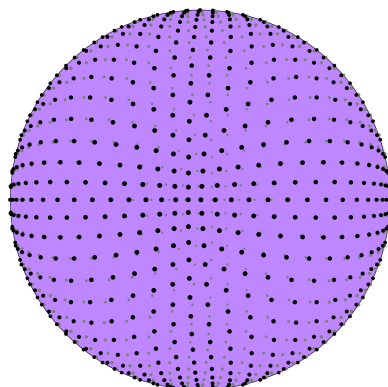
7.19 bits (K=6)



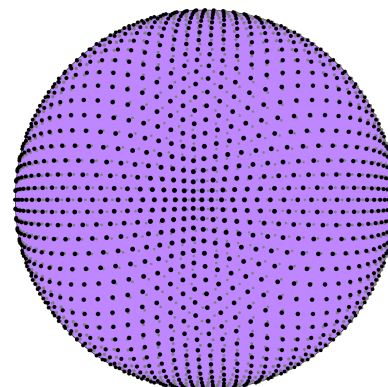
8.01 bits (K=8)



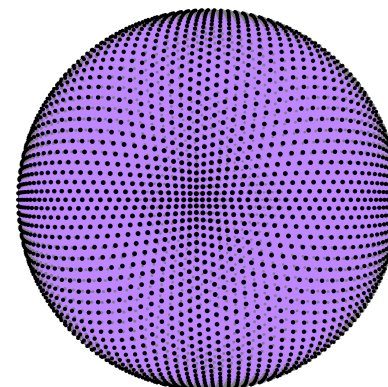
8.92 bits (K=11)



10.00 bits (K=16)



11.05 bits (K=23)



12.00 bits (K=32)



# Coding Band Shape

## Pyramid Vector Quantization



- PVQ codebook has a fast enumeration algorithm
  - Converts between vector and integer codebook index
  - $O(N+K)$  (lookup table, muls) or simpler  $O(NK)$  (adds)
  - Latter great for embedded processors, often faster
- Fast codebook search algorithm:  $O(N \cdot \min(N, K))$ 
  - Divide by  $L_1$  norm, round down: at least  $K-N$  pulses
  - Place remaining pulses (at most  $N$ ) one at a time
- Codebooks larger than 32 bits
  - Split the vector in half and code each half separately



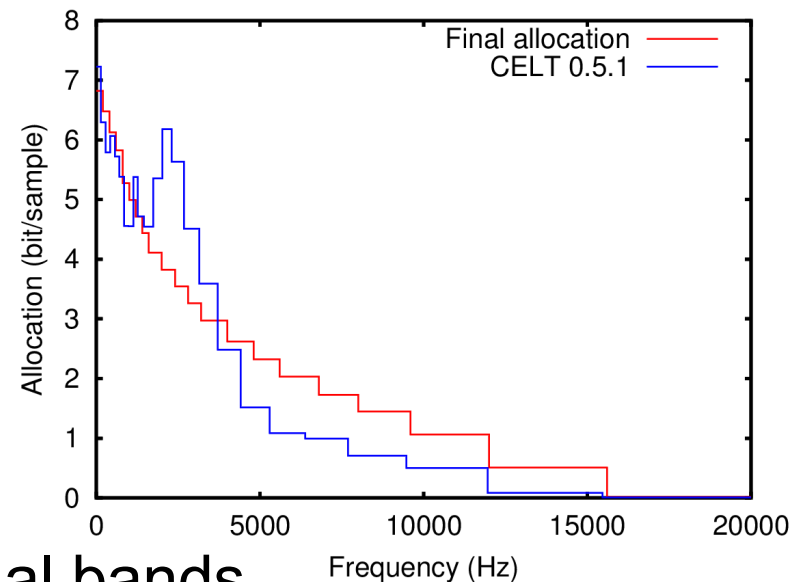
# Psychoacoustics

## Rate Allocation



- Encoder decides final bitrate early on
  - Right after coarse energy and side information
  - Can change from packet to packet, to adapt to network conditions
- Allocation between bands mostly static
  - Roughly constant *signal-to-mask ratio*
  - Two knobs available:
    - Boost: Gives more bits to individual bands
    - Tilt: shifts bits from LF to HF

Average Allocation @ 64 kbps





# Psychoacoustics

## Stereo Coupling



- Code separate energy for each channel
- Convert to mid-side in normalized domain
  - Safe, cannot cause cross-talk or bad artifacts
  - Split into separate  $M$  and  $S$  signals using normal split mechanism (incl. rate allocation)
- Intensity stereo:
  - Skip side channel in all bands past certain index

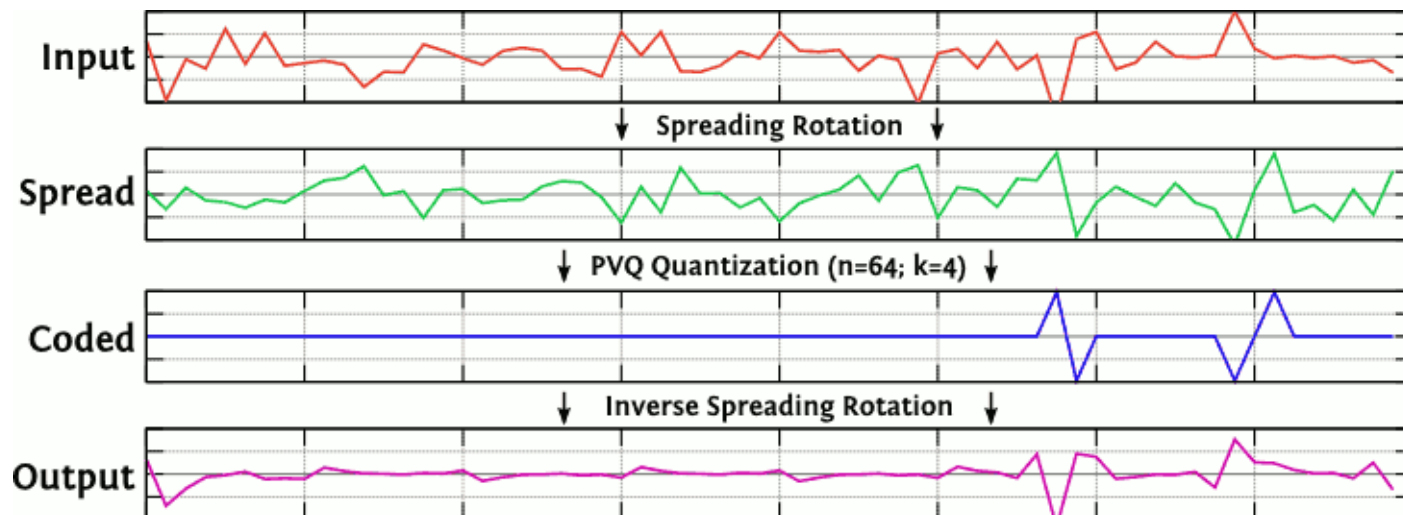


# Psychoacoustics

## Avoiding Birdie Artifacts



- Small  $K \rightarrow$  sparse spectrum after quantization
  - Produces tonal “tweets” in the HF
- CELT: Use pre-rotation and post-rotation to spread the spectrum (make it “rougher”)
  - Completely automatic (no per-band signaling)





# Psychoacoustics

## Folding



- When rate in a band is *too* low, code nothing
  - Hard threshold at 3/16ths bits per coefficient
    - Better than coding an extremely sparse spectrum
  - Encoder can choose to skip additional bands
- Still need to preserve energy
  - *Spectral folding*: copy previous coefficients
    - Gives correct temporal envelope
- Also used on just part of a band when splitting



# Psychoacoustics

## Transients (pre-echo)



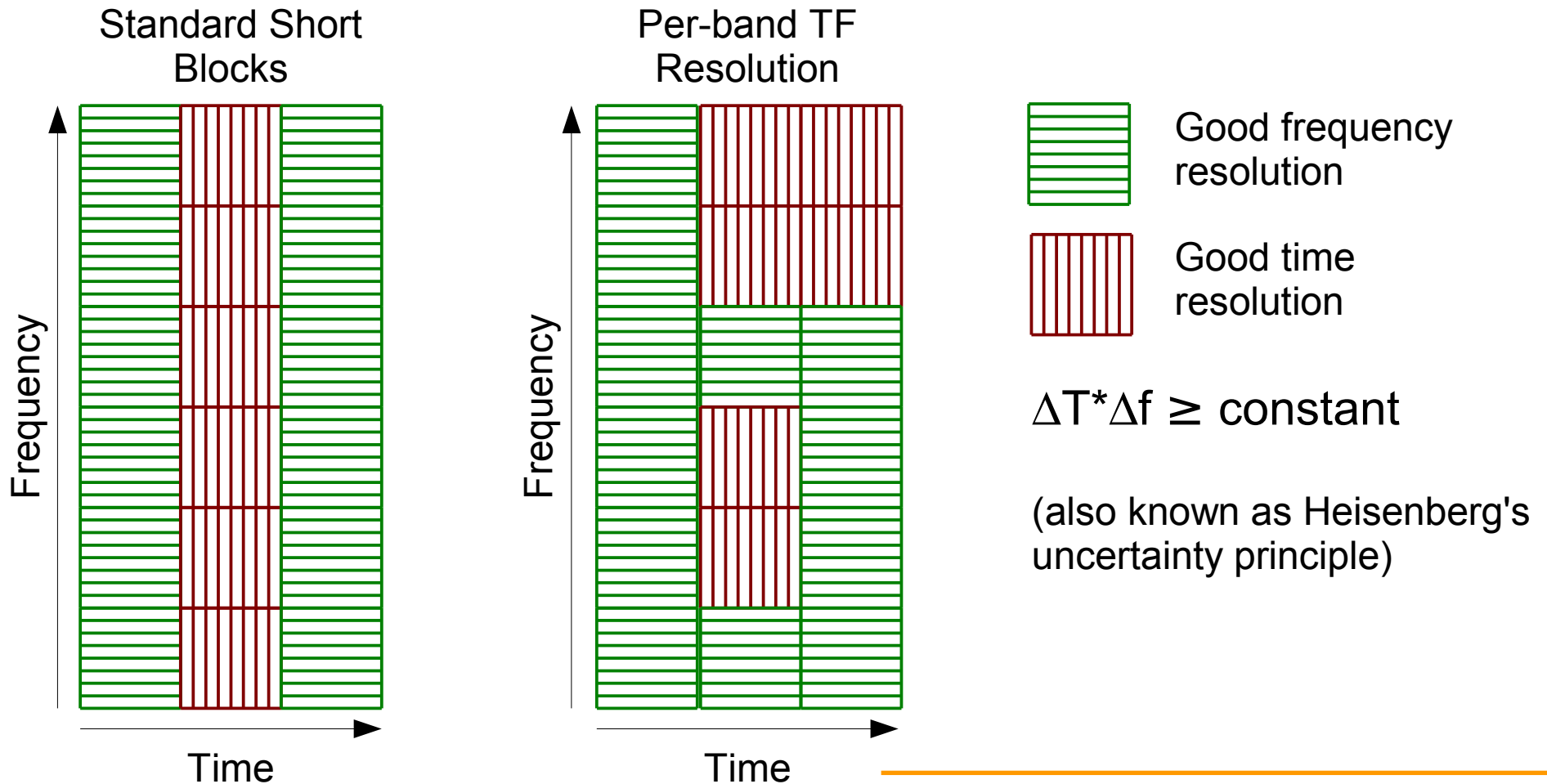
- Quant. error spreads over whole MDCT window
  - Can hear noise before an attack: pre-echo
- Split a frame into smaller MDCT windows (“short blocks”)
  - Interleave results and code as normal
    - Still code one energy value per band for all MDCTs
- Simultaneous tones and transients?
  - CELT: Use adaptive time-frequency resolution





# Psychoacoustics

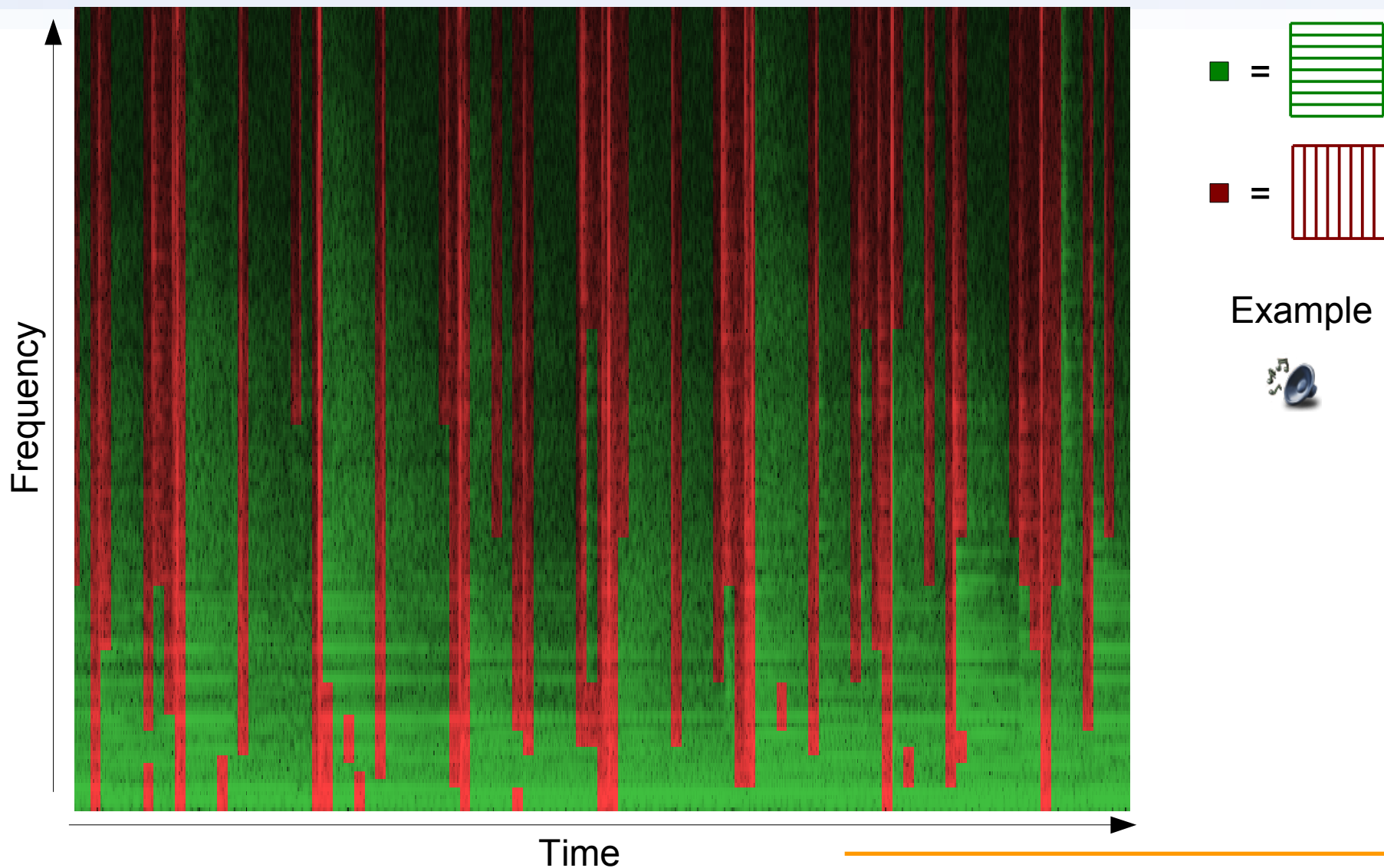
## Time-Frequency Resolution





# Psychoacoustics

## T-F Resolution Example



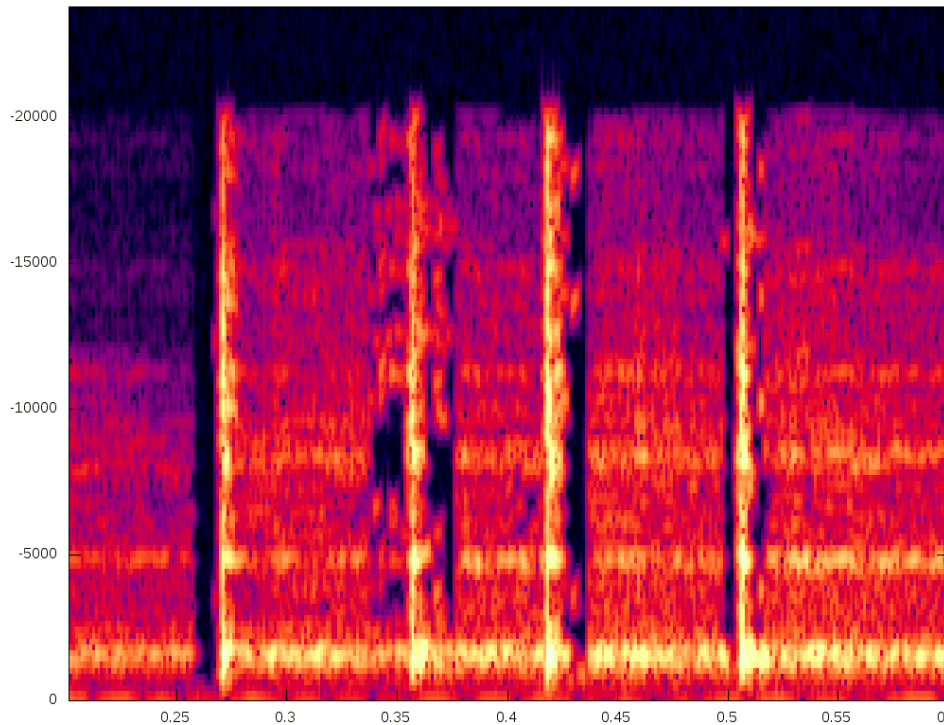


# Anti-Collapse

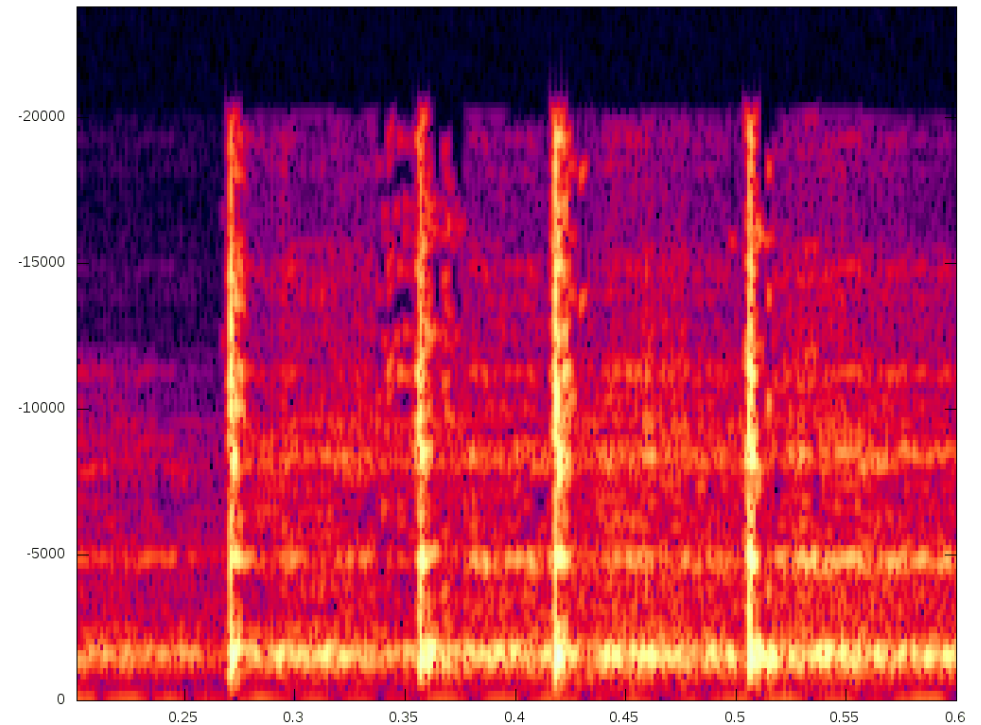


- Pre-echo avoidance can cause collapse
  - Solution: fill holes with noise

No anti-collapse



With anti-collapse





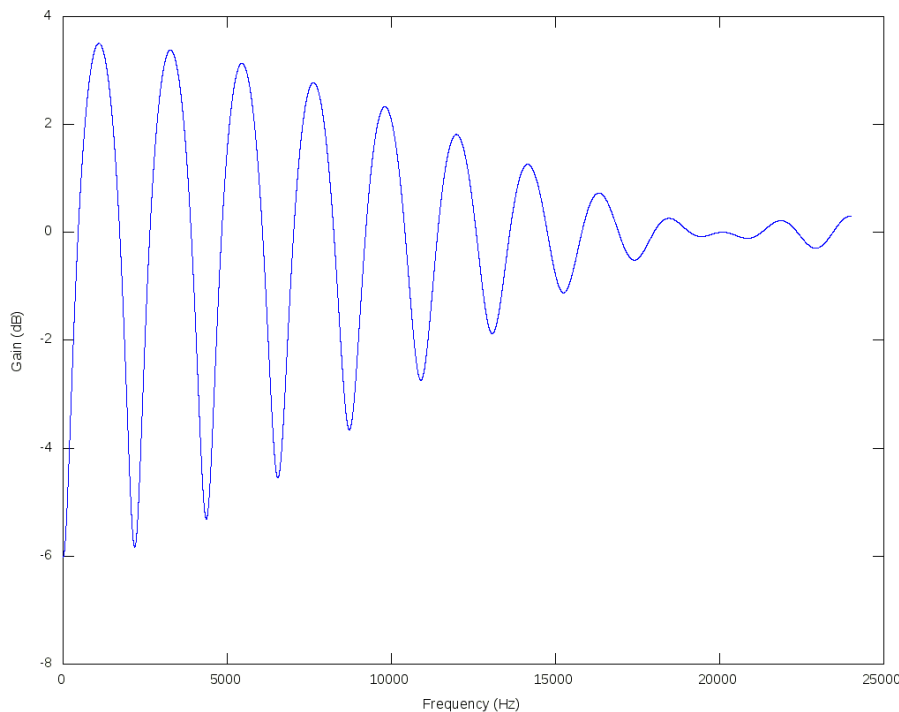
# Psychoacoustics

## Pitch Prefilter/Postfilter

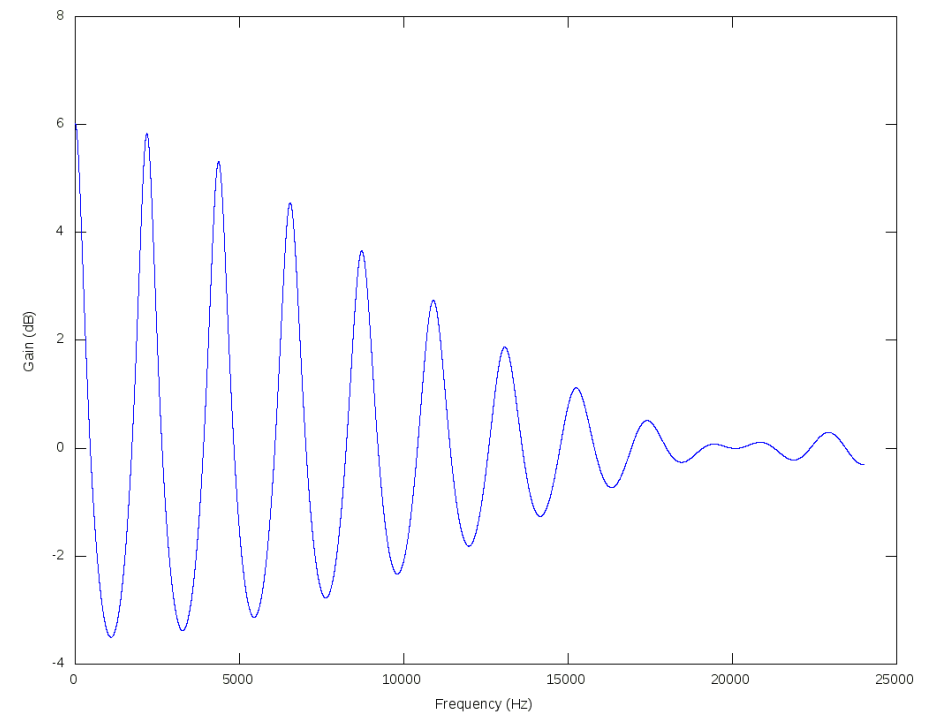


- Shapes quant. noise (like SILK's LPC filter), but for harmonic signals (like SILK's LTP filter)
  - Contributed by Broadcom

Prefilter



Postfilter





# Transitions



- Want to switch modes on the fly
- Don't want to create *glitches* when we switch
  - All modes can change frame sizes without issue
  - CELT can change audio bandwidth or mono/stereo without issue
    - The MDCT window overlap smooths the transition
  - SILK can change mono/stereo with encoder help
    - Narrow or enlarge stereo image slowly in frame before
- How about everything else?
  - 5 ms “Redundant” CELT frames smooth transition



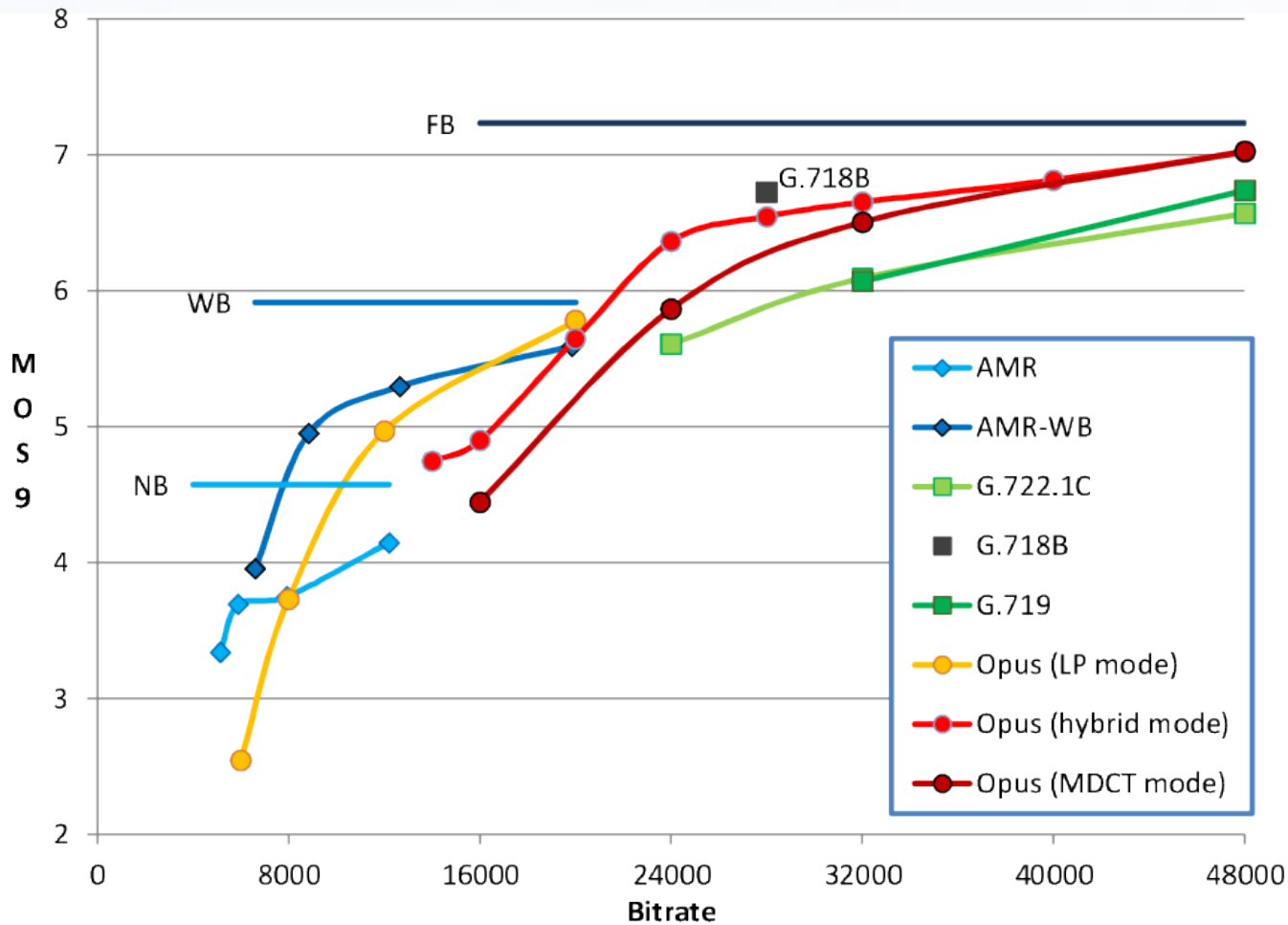
# Outline



- Introduction
- Opus Design
  - SILK
  - CELT
- **Results**
- WebRTC



# Opus Speech Quality



Anssi Rämö, Henri Toukoma, "Voice Quality Characterization of IETF Opus Codec", *Proc. Interspeech*, 2011.

See IETF proceedings for more listening test results:

<http://www.ietf.org/proceedings/82/slides/codec-1.pdf>  
<http://www.ietf.org/proceedings/80/slides/codec-5.pdf>



- Narrowband tests (English+Mandarin)
  - Opus clearly better than Speex and iLBC
  - Opus better than AMR-NB at 12 kb/s
- Wideband/fullband tests (English+Mandarin)
  - Opus clearly better than Speex, G.722.1, G.719
  - Opus better than AMR-WB at 20 kb/s
- Opus clearly better than MP3, inconclusive with AAC-LC
- No transcoding issues with AMR-NB/AMR-WB

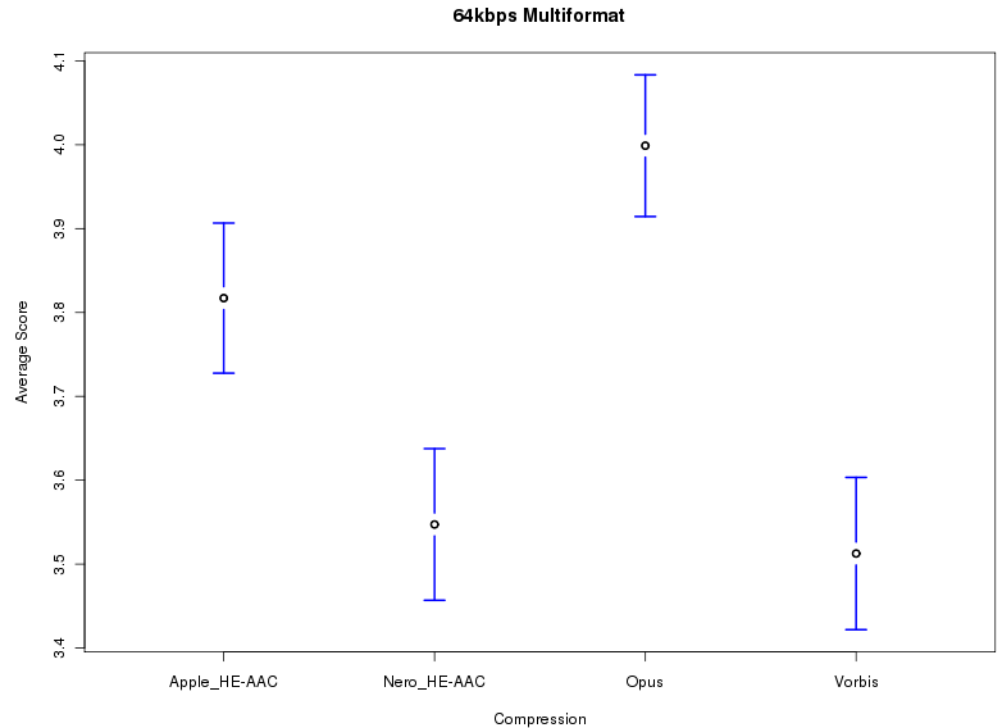




# Opus Music Quality



- 64 kb/s stereo music  
ABC/HR listening  
test by Hydrogen  
Audio



|              | Sample 01 | 02     | 03     | 04    | 05     | 06    | 07    | 08    | 09    | 10    | 11    | 12    | 13    | 14    | 15    | 16    | 17     | 18    | 19     | 20    | 21     | 22     | 23    | 24    | 25    | 26    | 27    | 28    | 29    | 30    |       |
|--------------|-----------|--------|--------|-------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|-------|--------|-------|--------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Opus         | Red       | Red    | Green  | Green | Green  | Green | Green | Green | Green | Green | Green | Green | Green | Green | Green | Green | Green  | Green | Green  | Red   | Yellow | Green  | Green | Green | Green | Green | Green | Green | Green | Green | Green |
| Apple HE-AAC | Green     | Green  | Yellow | Green | Yellow | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Yellow | Green | Yellow | Green | Green  | Red    | Red   | Red   | Red   | Red   | Green | Green | Green | Green | Green |
| Nero HE-AAC  | Green     | Green  | Red    | Red   | Red    | Green | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red   | Red    | Red   | Red    | Red   | Yellow | Yellow | Red   | Red   | Red   | Red   | Red   | Green | Green | Green | Green |
| Vorbis       | Red       | Yellow | Red    | Red   | Yellow | Red   | Green | Green | Green | Green | Green | Green | Green | Green | Green | Green | Green  | Green | Green  | Green | Green  | Green  | Green | Green | Green | Green | Green | Green | Green | Green |       |



# Current development



- Tools
  - Ogg encoder/decoder
  - libopusfile library
- Quality improvements
  - Better tuning of encoder decisions
  - Improved unconstrained VBR
  - Automatic speech/music detection



# Outline



- Introduction
- Opus Design
  - SILK
  - CELT
- Results
- **WebRTC**



# WebRTC Audio



- draft-ietf-rtcweb-audio
- Mandatory to implement (MTI) audio codecs
  - RFC 6716 (Opus)
  - G.711 ( $\mu$ -law/A-law)
- Audio level
  - AGC should adjust to -19 dBm0
  - Equivalent to RMS 2600 for 16-bit samples
- Should have AEC



# Opus over RTP



- Opus does not *require* signalling
  - All information is in-band
  - SDP parameters only express preferences
- Always signaled as `opus/48000/2`
- Bandwidth and stereo depend on the bit-rate
- Capabilities signaled as `stereo=` and `maxplaybackrate=` for receiver, `sprop-stereo=` and `sprop-maxcapture=` for sender



# And What About Video?



- WebRTC controversy: H.264 vs. VP8
  - No resolution in sight
- Attempt to create a new video codec WG
  - BoF held in last IETF meeting (Atlanta)
  - “Strong consensus” in favor of creating a WG
  - More discussion needed on charter
  - To be continued



# Questions?