Delivering Now and the future

Outputs for Today and tomorrow

Jeff I. Greenberg - Jeff @ JgreenbergConsulting.com

Yes, I can teach your group (or consult on compression or other video needs)

Jeff @ J Greenberg Consulting. com

@Filmgeek

Survey questions - NLE

- Adobe
- Apple
- Avid
- Sony
- Smoke
- Other

Survey - Compression

- Adobe Media Encoder
- Compressor
- Squeeze
- Episode
- Other (?)

Outputs

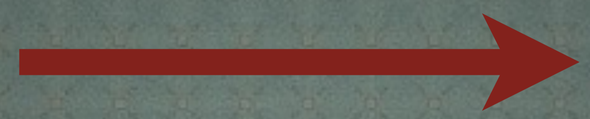
- “Master” (what format)
- Archival?
- Hardware - disc
- Hardware - h.264 device
- Online

Session intention

- Future proof our work
- Make it easy to be **flexible**
- **Do the work today and grab & go for the future**

A word on data rates

Data rates make my eyes fuzzy



Unfuzz filter

How much data per second

Large data rate
HUGE FILE
less compression

Low data rate smaller FILE MORE compression

At some point, files look too compressed

Golden Rules of Compression

Compress only once

probably the most important rule, and yeah, we're going to break it

- Compress only once
- Good, Fast, Small, Pick 2
- GIGO
- TETO
- VBR, except when you want CBR
- Scale down the video
- Super small? Drop frames
- Always obey hardware specs
- Shoot Progressive
- Consider shooting 24p
- Noise Reduction
- Watermark
- Prefer standard video sizing
- Normalized Audio

Golden rules

Good, Fast, Small - Pick 2

- Good & Fast - Super large, Post Codec, **Master**
- Small & Fast - Medium sized, CBR, watermarked **client approval** copy
- Good & Small - **small as possible** file

GIGO & TETO

VBR except when CBR

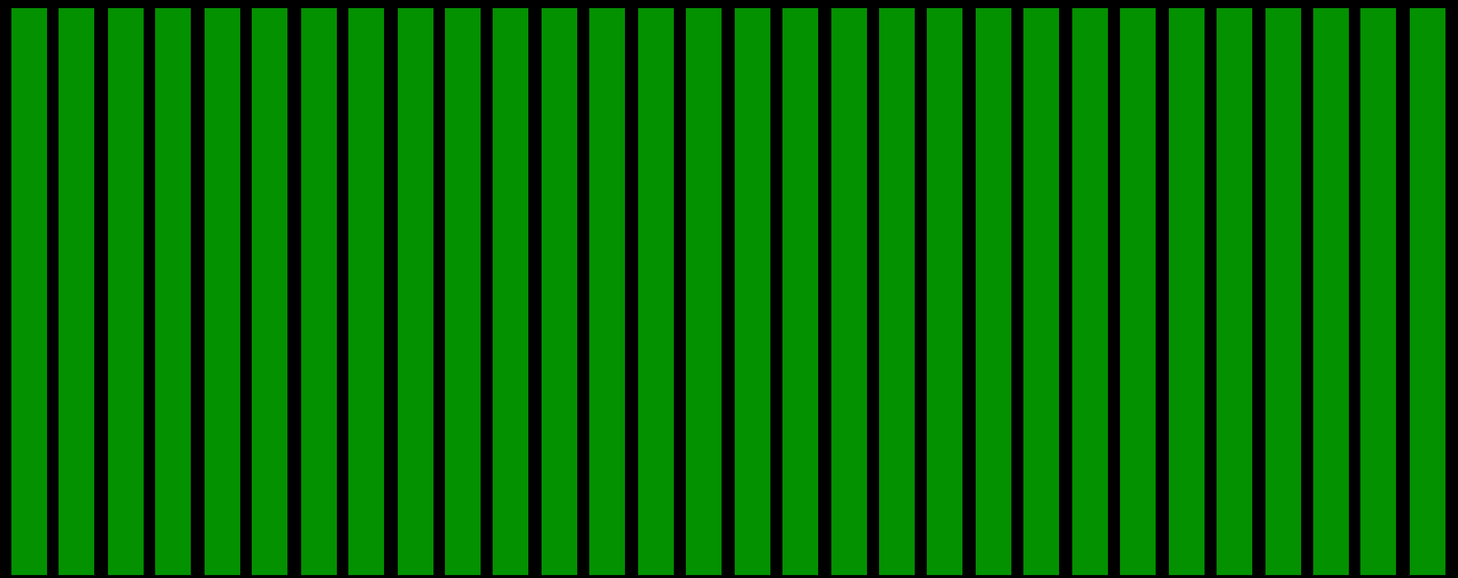
Which is where?

- CBR is usually camera/speed
- VBR can be meant for smaller (processing intensive)
- Or for a mild compromise (especially at large data rates)

Constant Bit rate

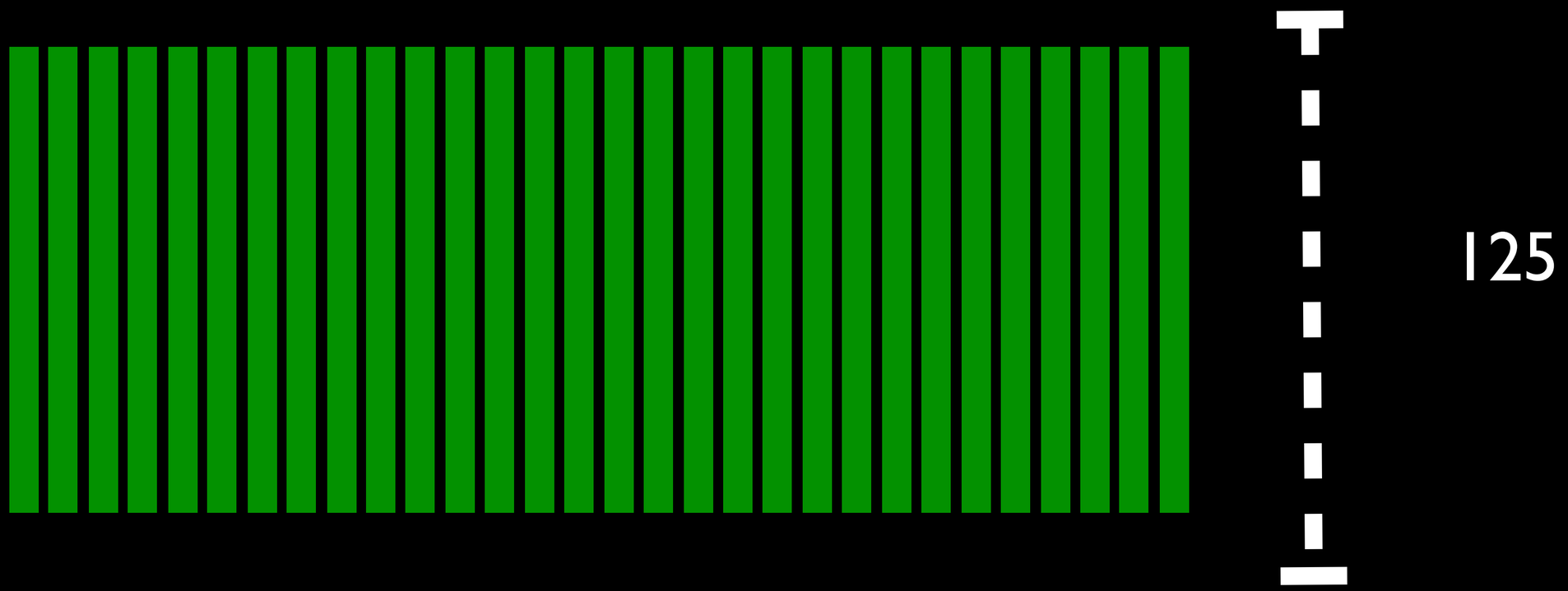
Constant Bit Rate

(every frame gets the same amount of data)



Constant Bit Rate

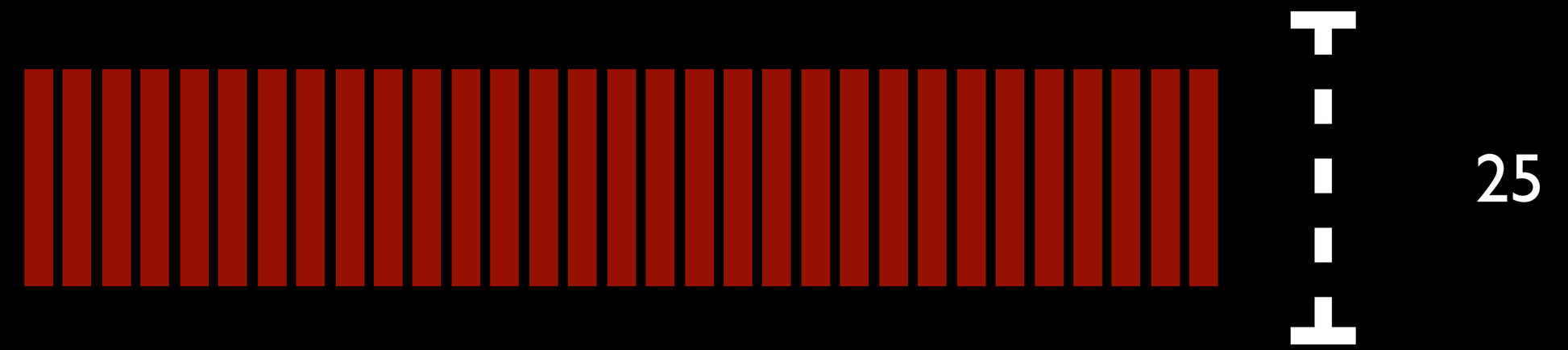
(every frame gets the same amount of data)



Uncompressed = easy for camera
(no compression/decompression)

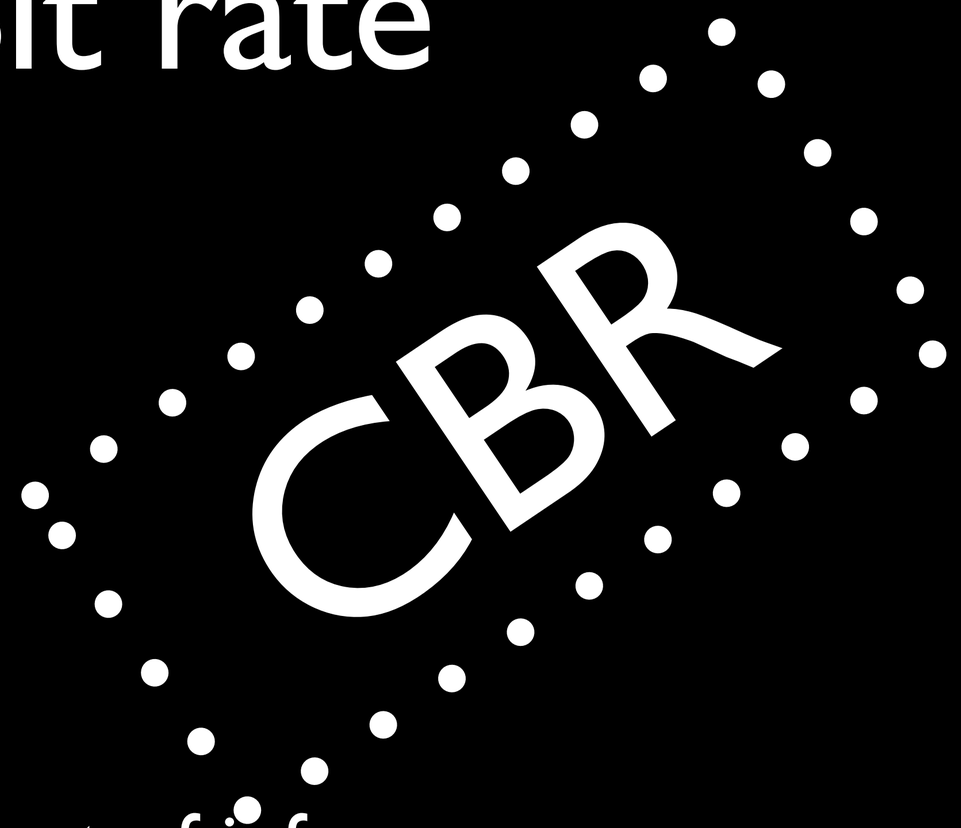
Constant Bit Rate

(every frame gets the same amount of data)



DV = each frame gets JPEG Compression.

Constant Bit rate



- How cameras work
- Fast, fast, fast
- Don’t make small files
- Every frame gets the same amount of info
- ex: 3000 units / 30 frames = 100 units per frame

Variable Bit Rate

Variable Bit Rates

or

Robbing Peter to pay Paul

How low can we go?

- At some data rate, the quality becomes unacceptable.

- Unique for any type of footage

- *but some footage has lesser demands*

Data

Complex material

Cherry Blossoms falling in DC

Simple material

Talking heads

What if we could steal data for where it's needed?

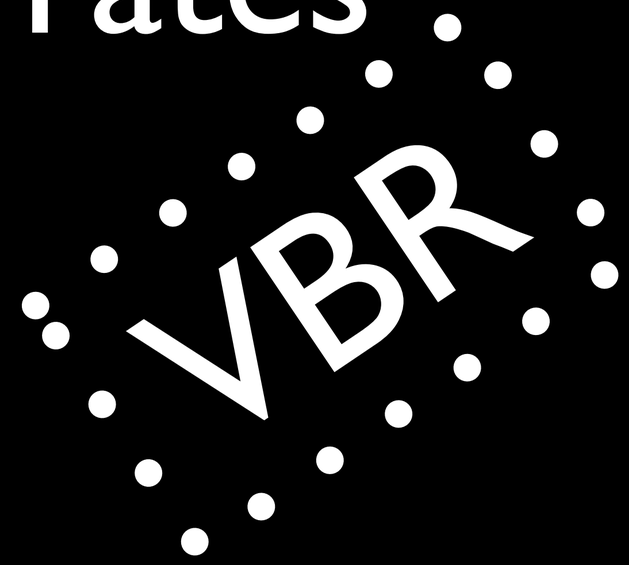
Complex material

Cherry Blossoms falling in DC

Simple material

Talking heads

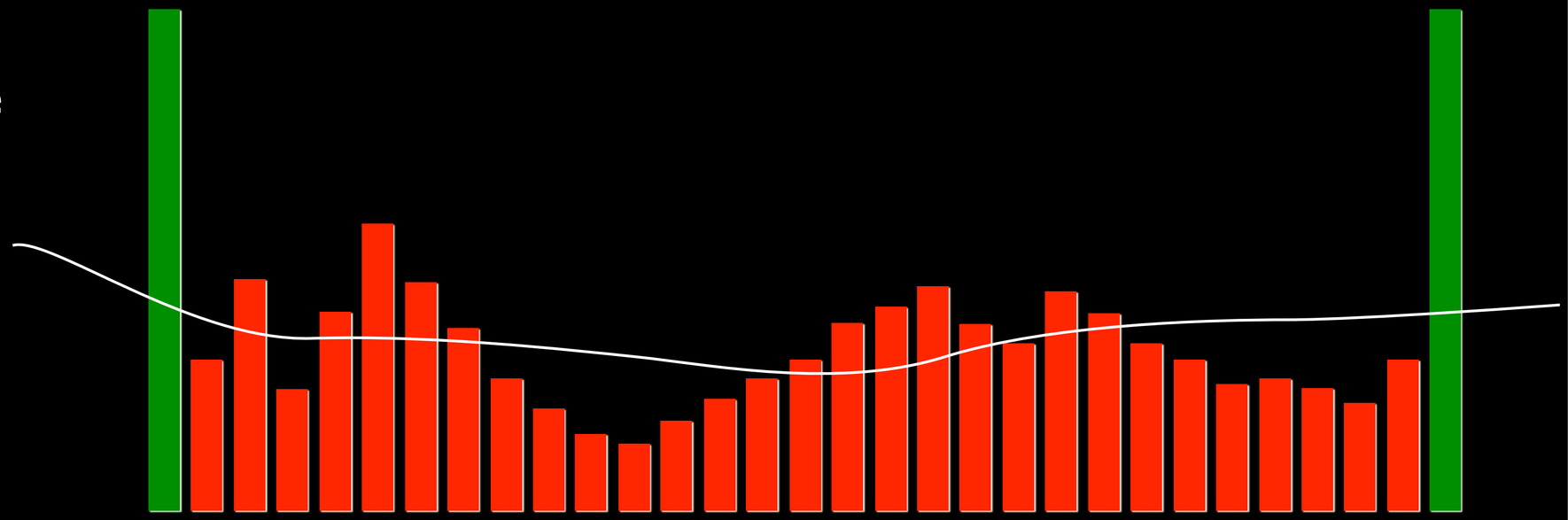
Variable data rates



Focus on the average

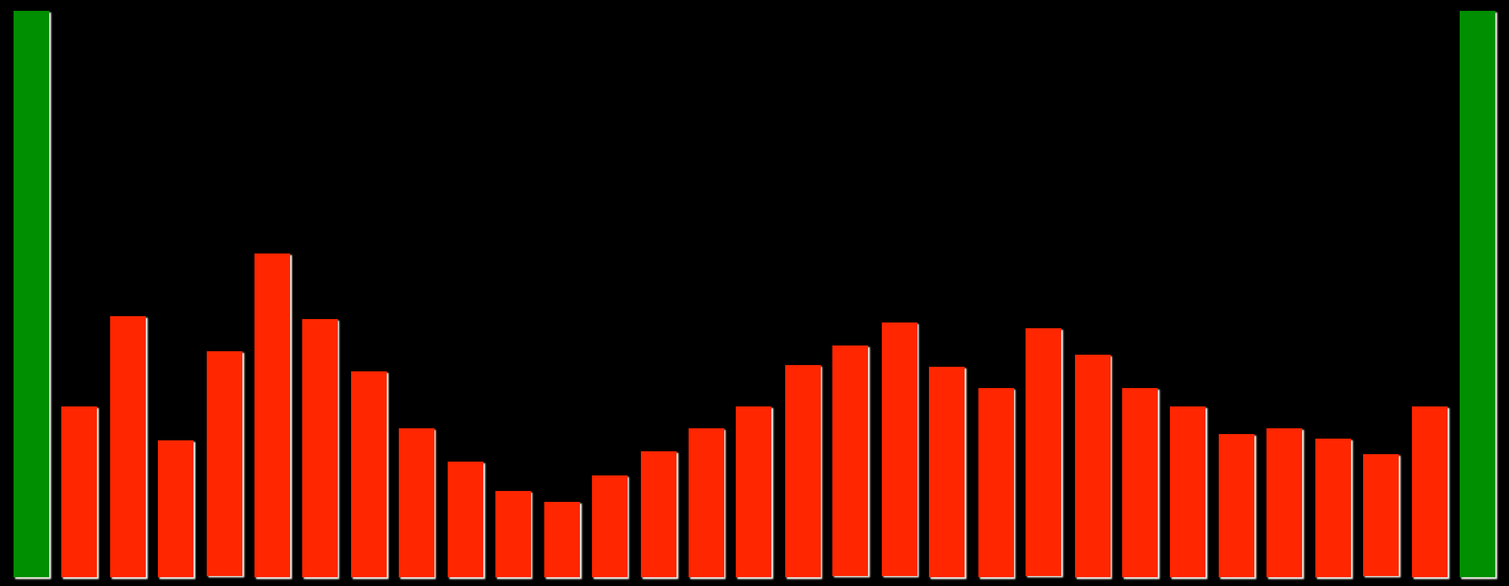
CBR or VBR, same data rate = same size file

Difference is where the data goes



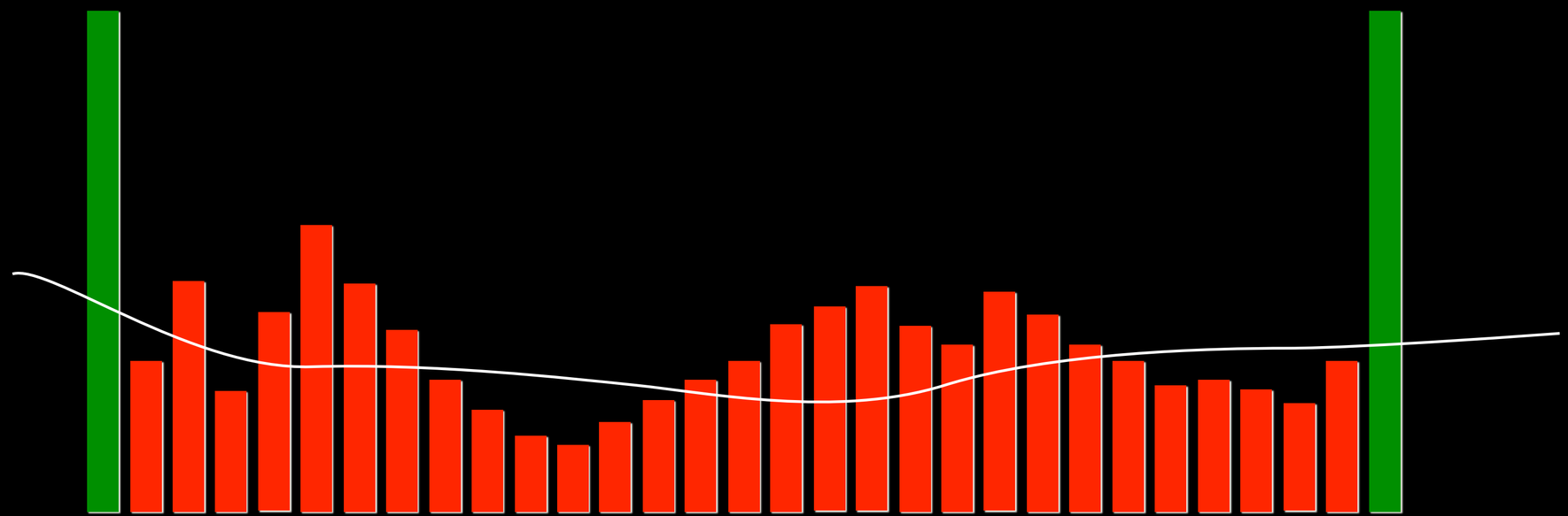
Lousy for editing

- Has to compute since last full frame
- “Why” you shouldn’t use formats like DVD for sources

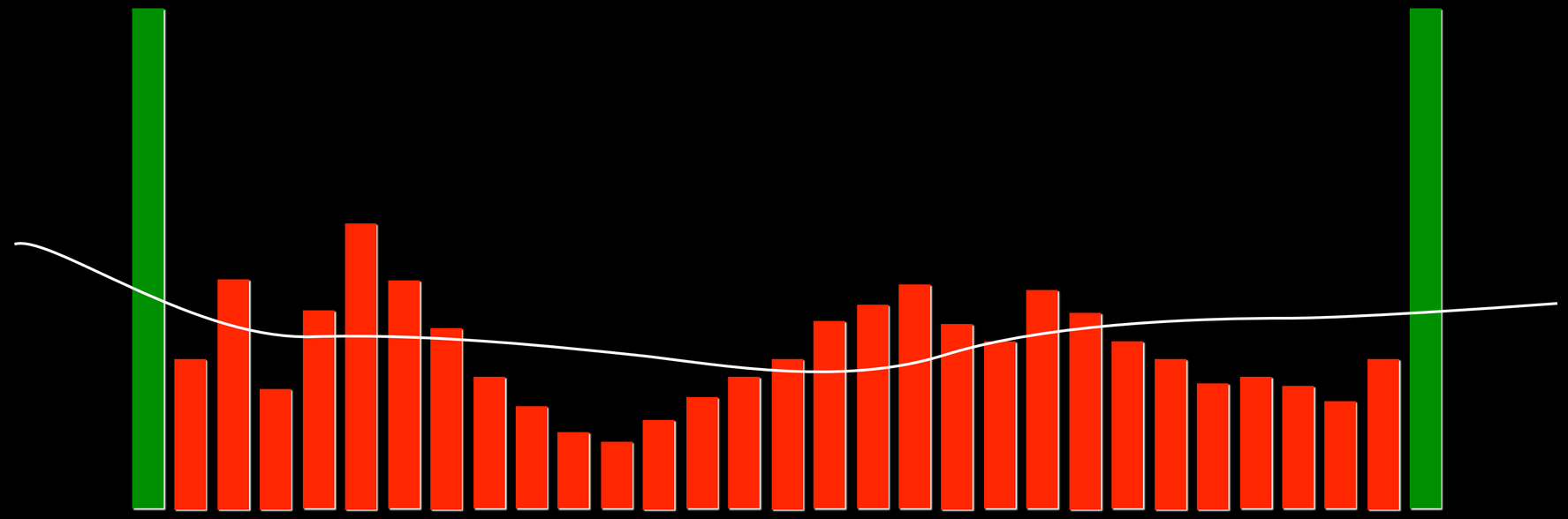


Focus on the average

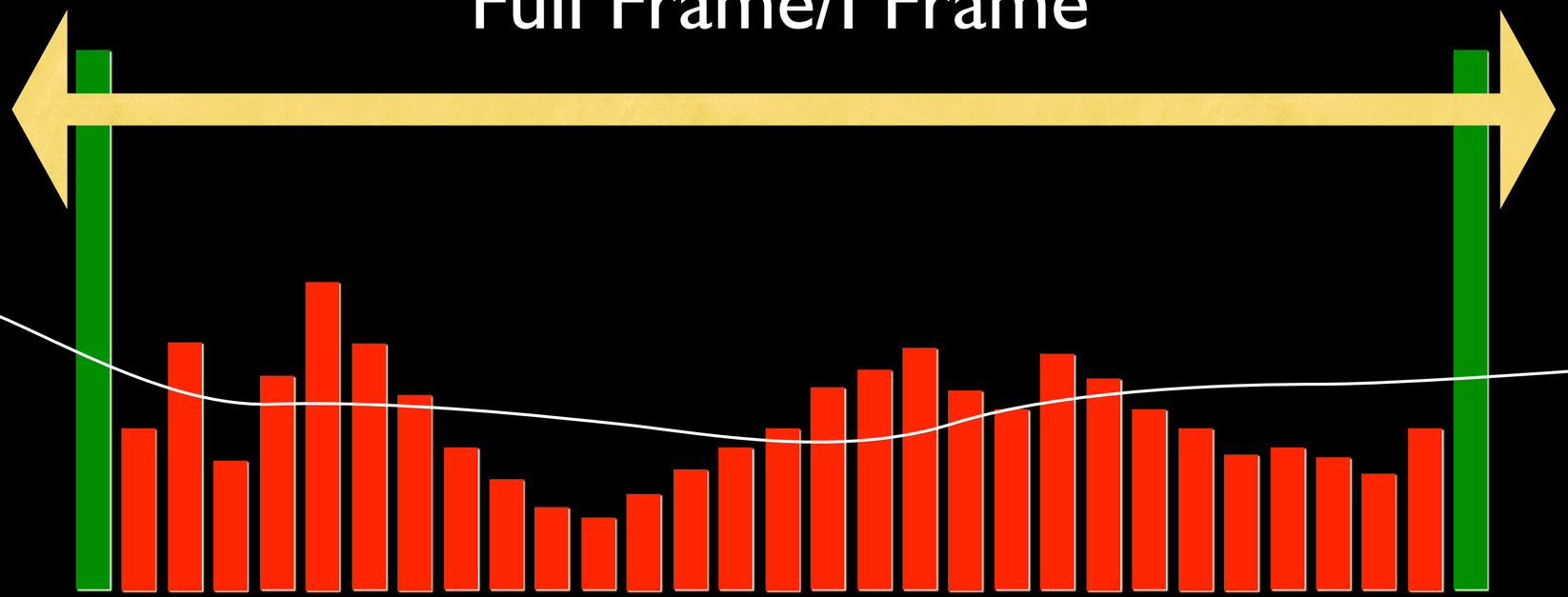
- A VBR or CBR file with the same data rate = same file size



Distribution codecs throw out tons of information

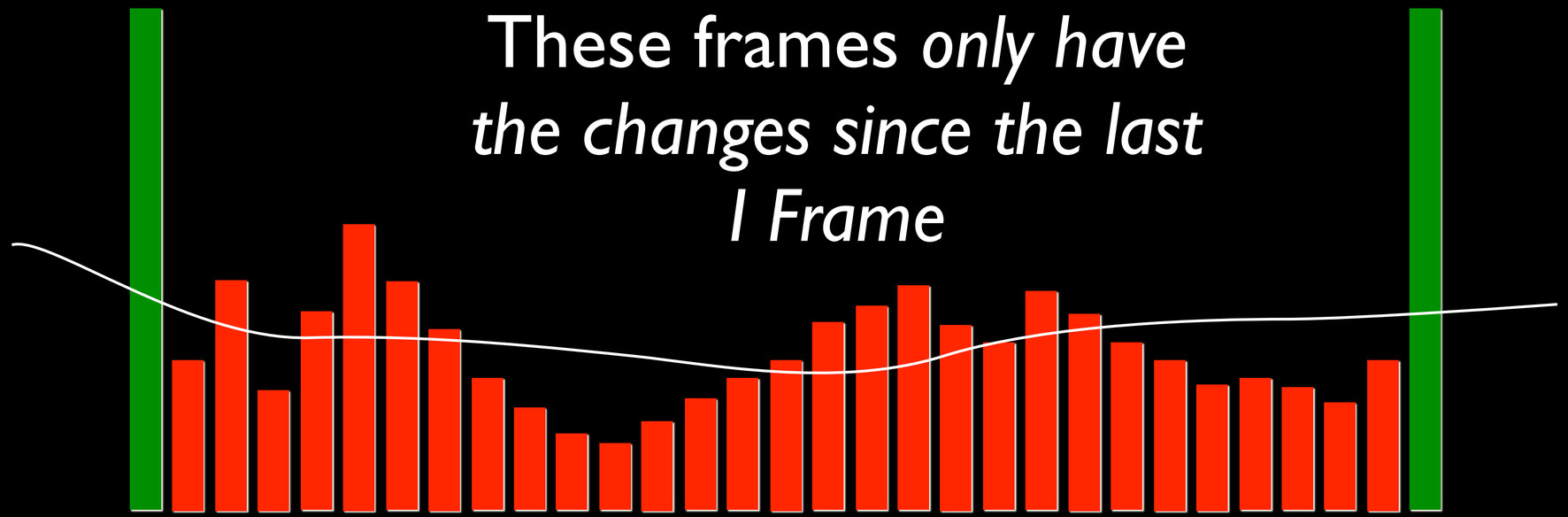


A “Group” of pictures



Delta Frames

B + P Frames



Multiple Passes =

moreAnalysis

Types of Data Rates

- Constant
- 1 Pass VBR
- 2 Pass VBR

- Which is fastest? Slowest?

Scale down your video/ reduce frames

- Methods for smallest video files
- 1/2 your HD width x height = 1/4 size file
- 1/2 your frame rate? = 1/2 your **data rate**

Obey Hardware rules

- DVD/Blu Ray
- Hardware h.264 players (iPhone, Android, Apple TV, Google TV)

Progressive...24p

- Despite Peter Jackson
- Progressive > Interlaced
- Deinterlacing means lower resolution
- Less frames = ease of online delivery/less data (20% less)

Noise Reduction

- Detail are bad **especially** with small files/low data rates
- (Visual) Noise Reduction tools
- Essentially complex (editing, detail, camera) is harder to compress

Watermark everything except the final deliverable

Prefer standard video sizes

Normalize your audio

Today

Minimum Suggestion

- Universal Master
- h.264 copies
- V. Large for Online/upload
- Device specific for common hardware
- As small as you can get and keep quality

Best possible master

- Match Frame size of camera
- Post Codec
- As **little damage** to original file

Rule Exception:

Compress only once

Done right, transcodes do minimal damage

Opinion: Post vs. Camera

- Camera codecs are **pure**
- But, we render, add graphics, color correct etc.
- Post codecs have the breathing space that camera codecs don't
- ~140 mb/s (dnxhd 145, ProRes 422) = 1 min = 1 GB

Why h264?

- Today - a standard (non proprietary) format that's flexible for less than SD to HD +

Large online hosted

- ~ 20 mb/s (will work well right now with nearly every
- Just under 150 megs /min

Device specs

- Rules of compression - obey hardware specs
- h.264
- Profiles
- Levels

Example: iPad

- High Profile
- Level 4.1

Profiles are (mostly) color space

- Hi Profile = 4:2:0 color space
- 8 bpp
- other specs about how a file is encoded

Levels are Frame + Data Rate

- 4.1
- 50 mb/s
- 720 (68fps)
- 1080 (30 fps)
- 2048 x 1024 (30fps) (2k)

Pay attention L+P

- Really sensitive in Android
- iOS is a little simpler (less fragmentation)
- Much easier to search for level + profile than individual specs
- Each L+P must support everything below)

Yes, you can overlap this with the online upload

Tiniest one

- Easy to transport.
- How aggressive do you want to be?
- SD data rate ~ .5-2 mb/s (1 min = 8 mb @ 1mb/s)
- 720 data rate ~ 2-4 mb/s (1 min = 24 mb @ 3mb/s)
- 1080 data rate ~3-6 mb/s (1 min = 32 mb @ 3 mb/s)

Do you want to be very agressive?

- Halve the frame size - **half the data rate**
- Halve the frame rate - **half the data rate**
- **Yes you can do both** - but probably won't play on devices

Building “Smart”

- Watch Folders (droplets)

- Make sure to **NAME appropriately**

The Future

Archival & h.264

JPEG 2k?

- Library of congress.
- Large files
- Like a post codec/ based on JPEG

h.265

- 1080 at **half the data rates**
- More complex
  - harder on computers
  - most hardware devices don't have the chips
- 2k, 3k, 4k

Feature support in particular profiles

Feature	CBP	BP	XP	MP	ProHiP	HiP	Hi10P	Hi422P	Hi444PP
Chroma formats	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0/4:2:2	4:2:0/4:2:2/4:4:4
Sample depths (bits)	8	8	8	8	8	8	8 to 10	8 to 10	8 to 14
Flexible macroblock ordering (FMO)	No	Yes	Yes	No	No	No	No	No	No
Arbitrary slice ordering (ASO)	No	Yes	Yes	No	No	No	No	No	No
Redundant slices (RS)	No	Yes	Yes	No	No	No	No	No	No
Data Partitioning	No	No	Yes	No	No	No	No	No	No
SI and SP slices	No	No	Yes	No	No	No	No	No	No
Interlaced coding (PicAFF, MBAFF)	No	No	Yes	Yes	No	Yes	Yes	Yes	Yes
B slices	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CABAC entropy coding	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes
8×8 vs. 4×4 transform adaptivity	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Quantization scaling matrices	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Separate C_b and C_r QP control	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Monochrome (4:0:0)	No	No	No	No	Yes	Yes	Yes	Yes	Yes
Separate color plane coding	No	No	No	No	No	No	No	No	Yes
Predictive lossless coding	No	No	No	No	No	No	No	No	Yes

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames)
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	128×96@30.9 (8) 176×144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	128×96@30.9 (8) 176×144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	176×144@30.3 (9) 320×240@10.0 (3) 352×288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	320×240@20.0 (7) 352×288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	320×240@36.0 (7) 352×288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	320×240@36.0 (7) 352×288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352×480@30.0 (7) 352×576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	352×480@30.7(10) 352×576@25.6 (7) 720×480@15.0 (6) 720×576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	352×480@61.4 (12) 352×576@51.1 (10) 720×480@30.0 (6) 720×576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	720×480@80.0 (13) 720×576@66.7 (11) 1280×720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280×720@60.0 (5) 1,280×1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	1,280×720@68.3 (9) 1,920×1,080@30.1 (4) 2,048×1,024@30.0 (4)

Bonus materials - wikipedia page on h.264 - https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC