Seeking is hard

Ralph Giles

Xiph.org Foundation
for Open Multimedia

linux.conf.au 2008 Melbourne Australia
Seeking is hard
Let’s talk about Ogg
what’s Ogg?
Ogg is a project
Free and Open Multimedia
talking about
container format
AVI
not DivX
MPEG (stream)
MPEG (stream) not h.264
Ogg
Ogg
not Vorbis
Ogg
not Theora
what is Ogg?
Ogg is a dodge
Ogg is clever
Ogg is dodgy
a better mousetrap
a different mousetrap
Containers
Containers

- Frame individual elements
- Random access
- Overhead
- Robustness
Media containers

- playback (sync)
- random access (seek)
- serial access (stream)
- editing (cut and paste)
Ogg does three

- playback (sync)
- random access (seek)
- serial access (stream)
Ogg does six

- sync
- seek
- stream
- frame
- robustness
- overhead
Sync

- Capture and restart
- multiplex with fixed or variable spacing
Packets vs Pages

- Codec defines packets
- Packets have arbitrary length
- Packets are divided, and the segments are packed into Pages for muxing.
Packets
packet length coding

- segments are 255 bytes
- last segment is < 255
- \(46 \Rightarrow 46\)
- \(510 \Rightarrow 255, 255, 0\)
- \(1024 \Rightarrow 255, 255, 255, 255, 4\)
- OggS magic
- start and end page flags
- sequence numbers
- “granulepos” timestamp
- “lacing” encoding of packet boundaries
- packing and continuation
- CRC
Packets vs. Pages

- Packets can be any length
- No internal framing is necessary
Packets vs Pages

- Timestamps on Pages
- Data lost in Page units
- Pages limited to < 64 KB
Multiplex
Multiplex
Chaining

$ cat alpha.ogv beta.ogv > gamma.ogv
Profiles

Stream Mapping

codecs  Ogg
Good

- Flexible
- Overhead bounded at 2%
- Single pass
- Robust
Bad

- Timestamp interpretation
- Flexible
- expects you to write real seek code
Bad

- CRC over the wrong stuff
- CRC expensive
- bitrate variance and buffering
Fixes
Fixes

Annodex Project
Fixes

- ‘Skeleton’ track with metadata
- codec media-type
- granulepos mapping
Skeleton

everything a muxer needs
(if it’s there)
Fixes?

- Alter CRC behaviour
- Alter packing rules
Seeking
time $\Rightarrow$ byte offset
Seeking is hard
Seeking is hard

In all formats
Seeking Methods
Implicit
Implicit

great for uncompressed data!
calculate from the bitrate
time $\propto$ bytes
If only bitrates were constant!
Seek table
Seek table?

AVI, Quicktime MOV
Seek table

Doesn’t stream!
Two pass
Seek table
Could be wrong!
Or missing
Unreliable
Only an optimization
Timestamps
Timestamps

Are harder to parse in Ogg
Get you in the right region
It gets worse
keyframes
pre-roll
setup data
Even with Timestamps

- Keyframes (prerequisites)
- lapped blocks (preroll)
- low bitrate streams (subtitles)
- program segments
Seeking

sick of it yet?
Seeking

- Use average bitrate
- Use a seek table
- Use timestamps
Seeking in Ogg

- Use average bitrate
- Build a seek table
- Use timestamps
To seek in Ogg

- Bisection search for the closest page with the timestamp prior to the seek point.
- Start decoding there.
- Discard frames until the seek point.
Edge cases

- spanning continued packets
- ignore this and you can seek faster
To seek in Ogg

- For each track:
  - Find the latest page marked with a timestamp before the seek point
  - Start decoding at the earliest of these
Thanks
Thanks

- Conrad Parker, Shane Stephens
- Mike Smith, thomasvs
- Christopher Montgomery
- everyone who complained
Questions?
Thank you!