

Stateless Codec Userspace Support

Hans Verkuil

Cisco Systems Norway

Why?

- Support for stateless codecs is/will be merged for 4.20. To use stateless codecs you need userspace support to parse the meta data from the bitstream (decoder) or add the meta data to a bitstream (encoder).
- What is the easiest way to add support for such codecs to the main applications (ffmpeg, gstreamer, vlc)?

Requirements for Stateless Decoder Support

- Parse bitstream for the various codecs.
- Fill in the required V4L2 codec controls in the Request based on the parsed metadata.
- Queue the buffer with the bitstream data to the Request.
- Queue the request and wait for it to complete.
- Read back the any updated metadata.

Requirements for Stateless Encoder Support

- Pass raw video frame to the Request.
- Fill in the required V4L2 codec controls in the Request in order to perform the compression.
- Queue the request and wait for it to complete.
- Read the V4L2 controls with the meta data from the completed request.
- Combine that with the dequeued buffer containing the compressed video bitstream to provide a proper bitstream.

Bit Parsing

- It turns out that this is very specific to the application: each application has their own implementation with different characteristics (error correction/hiding).
- No point in adding yet another parser, just leave it to the applications.
- Open question: is there any benefit to provide simple parsers that parse just enough to be able to test HW codecs in e.g. v4l2-ctl/compliance?

libva

- All main applications support libva: developed by Intel to provide access to their codec accelerators.
- A libva backend was written for the stateless cedrus decoder was developed by Bootlin and that worked well.
- Suggestion: create libva backends for stateless codecs as this is the easiest way to get support for the main applications up and running.
- My preference is to maintain them on linuxtv.org, so they can easily be kept in sync with the kernel.
- It is expected that apps like gstreamer will eventually replace these libva backends with direct V4L2 support to avoid this extra libva layer.