VVC/H.266 in GStreamer

Carlos Bentzen

2025-10-23

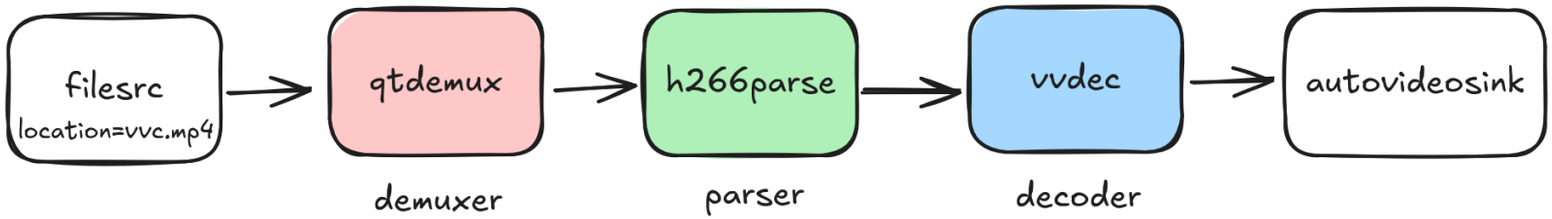
Versatile Video Coding (VVC/H.266)

- TLDR: the successor to HEVC/H.265
- Promises 50% bitrate savings over HEVC

VVC in GStreamer

Basic playback & remuxing support was added in GStreamer 1.26

```
gst-play-1.0 vvc.mp4
```



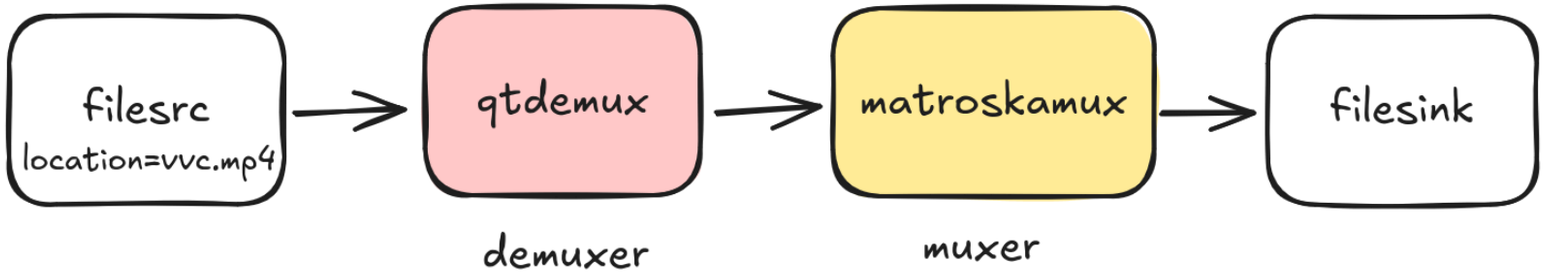
igalia

(De)muxing

Support added for H.266 to multiple muxers & demuxers:

- **qtmux** & **qtdemux**: ISOBMFF/QuickTime/MP4
  - TODO: open MR with VVC support for Rust (f)mp4 muxers
- **matroskamux** & **matroskademux**: Matroska (MKV)
- **mpegtsmux** & **tsdemux**: MPEG Transport Streams

Remuxing



Why do we need `h266parse` in the second case though?

igalia

Parsing

- **Why?** To handle the multiple alignment and stream formats
- Let's look at `h266parse` src pad caps:

```
video/x-h266
            parsed: true
     stream-format: { vvc1, vvi1, byte-stream }
         alignment: { au, nal }
```


igalia

Parsing

Stream formats:

- `byte-stream` (aka Annex-B): NAL units separated by 0x000001 or 0x00000001 prefix
- `vvc1` & `vvi`: length-prefixed NAL units
  - `vvc1` carries parameter sets over container-level metadata as `codec_data`
  - `vvi1` can carry them in the bitstream too

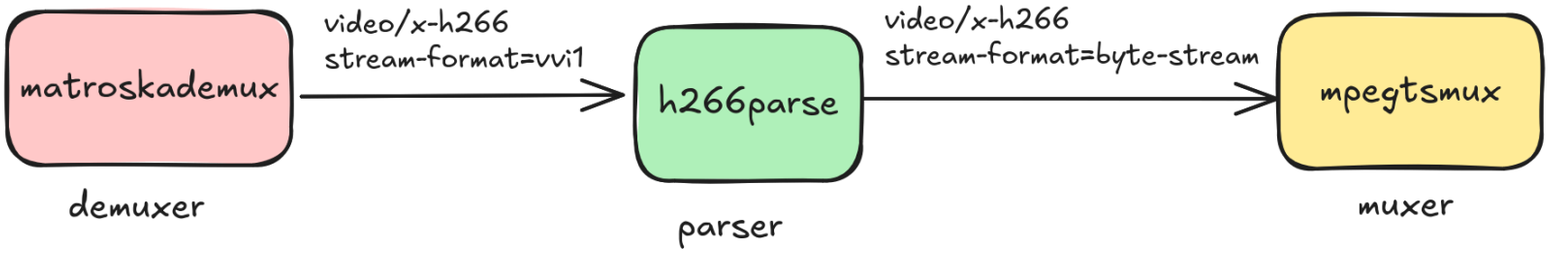
Parsing

Alignment:

- NAL units are the basic unit of data in H.266 streams
- Access unit (AU) is the smallest decodable unit for a complete video frame, which may contain multiple NALs.
- **Note**: we might need to add Picture Unit (PU) for multi-layer support

Parsing

- `h266parse` and parser library implemented by Intel, supporting only `byte-stream`
- Support for `vvc1` and `vvi1` added right afterwards



igalia

Decoding

- **avdec_h266**: FFmpeg's implementation since 7.1
- **vvdec**: VVdeC implementation from Fraunhofer
  - Implement via Rust bindings in gst-plugins-rs
  - TODO: handle latency query
- **vah266dec**: VAAPI implementation for Intel Lunar Lake & Panther Lake CPUs
  - Implemented by Intel with support only for `byte-stream`
  - Support added for `vvi1` and `vvc1` (to be released in 1.28)

igalia

WebKit ❤️ GStreamer

- Hardware-accelerated VVC playback in WebKitGTK via VAAPI

igalia

Ongoing & Future work

- **vvenc**: uses VVenC to implement an encoder element
  - MR open
  - Not suitable for real-time, mostly useful for VOD
- Support VVC in Rust (f)mp4 muxers
- **rtph266(de)pay**: RTP support
  - There is a C implementation, but upstream would prefer to have it in Rust
- Multilayer support:
  - Check out the talk *"VVC/H.266 Alpha Channel support in GStreamer"*!
- Hardware encoding?
- More hardware decoders?

igalia

Thank you!

Questions?

igalia