

# HLS/LL-HLS and DASH playback

Jan Schmidt [jan@centricular.com](mailto:jan@centricular.com)

A Coruña, 26th September 2023

# Introduction

- What are HLS/LL-HLS and DASH?
- HTTP-based adaptive streaming
- A Manifest/Playlist plus media segments
- Encode different quality levels
- Client(s) choose which level to load
- Live content vs VOD
- MSS

# HLS/LL-HLS

- Introduced by Apple in 2009
- Playlists based on m3u8. Multivariant + Media
- MPEG-TS media fragments for video/audio, later fMP4
- Can support raw audio + ID3 header
- WebVTT subtitles

```
#EXTM3U
#EXT-X-TARGETDURATION:10
#EXT-X-VERSION:3
#EXTINF:9.009,
http://media.example.com/first.ts
#EXTINF:9.009,
http://media.example.com/second.ts
#EXTINF:3.003,
http://media.example.com/third.ts
#EXT-X-ENDLIST
```

# DASH

- Released by MPEG in 2012
- XML based manifest
- Fragmented MP4 media segments

```
<MPD>
  ...
  <Period start="PT0.0S">
    <AdaptationSet contentType="video" segmentAlignment="true" bitstreamSwitching="true" frameRate="15/2">
      <Representation id="0" mimeType="video/mp4" codecs="avc1.4d0020" bandwidth="2000000" width="1280" height="960" frameRate="15/2">
        <SegmentTemplate timescale="15360" initialization="init-stream$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s" startNumber="22732">
          <SegmentTimeline>
            <S t="1047462831" d="30639" />
            <S d="61438" />
            <S d="30709" />
            <S d="61457" />
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```

# History in GStreamer

- Playback since 2013 in hlsdemux, dashdemux, mssdemux
- Production via hlssink
  - Now hlssink2 and hlssink3 (rust)
- Talking about playback today

# adaptivedemux base-class

- Introduced in 2014
- Merged the common code for managing downloads, refreshing manifests/playlists
- Subclasses responsible for manifest/playlist parsing and deciding which media fragments to download

# Problems

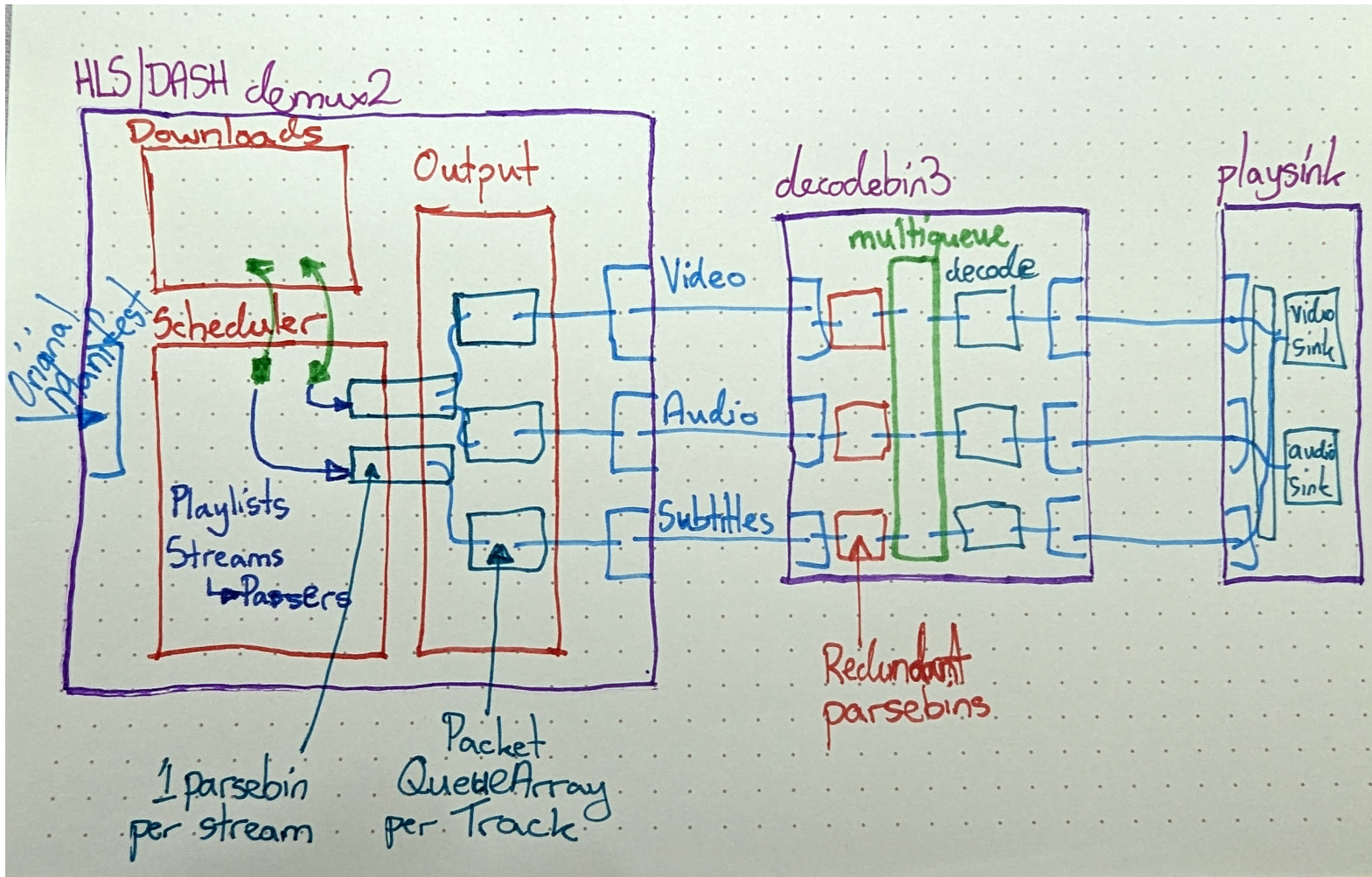
- Synchronous playlist updates in various places.
- Poor support for alternate renditions
  - Download all streams and wait for NOT-LINKED
- Poor buffering management
  - Buffering done downstream in decodebin2
- Problems with complex stream setups
  - HLS streams can be change format w/ quality levels
- 2 threads per stream - for download and output

# New architecture

- Designed for playbin3
  - Supports `GstStreamCollection` and `SELECT_STREAMS`
- Internalise buffering and parsing
  - Fine grained buffering and output control
  - Faster track switching
  - Quirks for common stream faults
- Almost no queues / buffering downstream during playback
- Reduce the threads to 3: Scheduler/Parser, Download and Output
- Parallelise downloads where possible

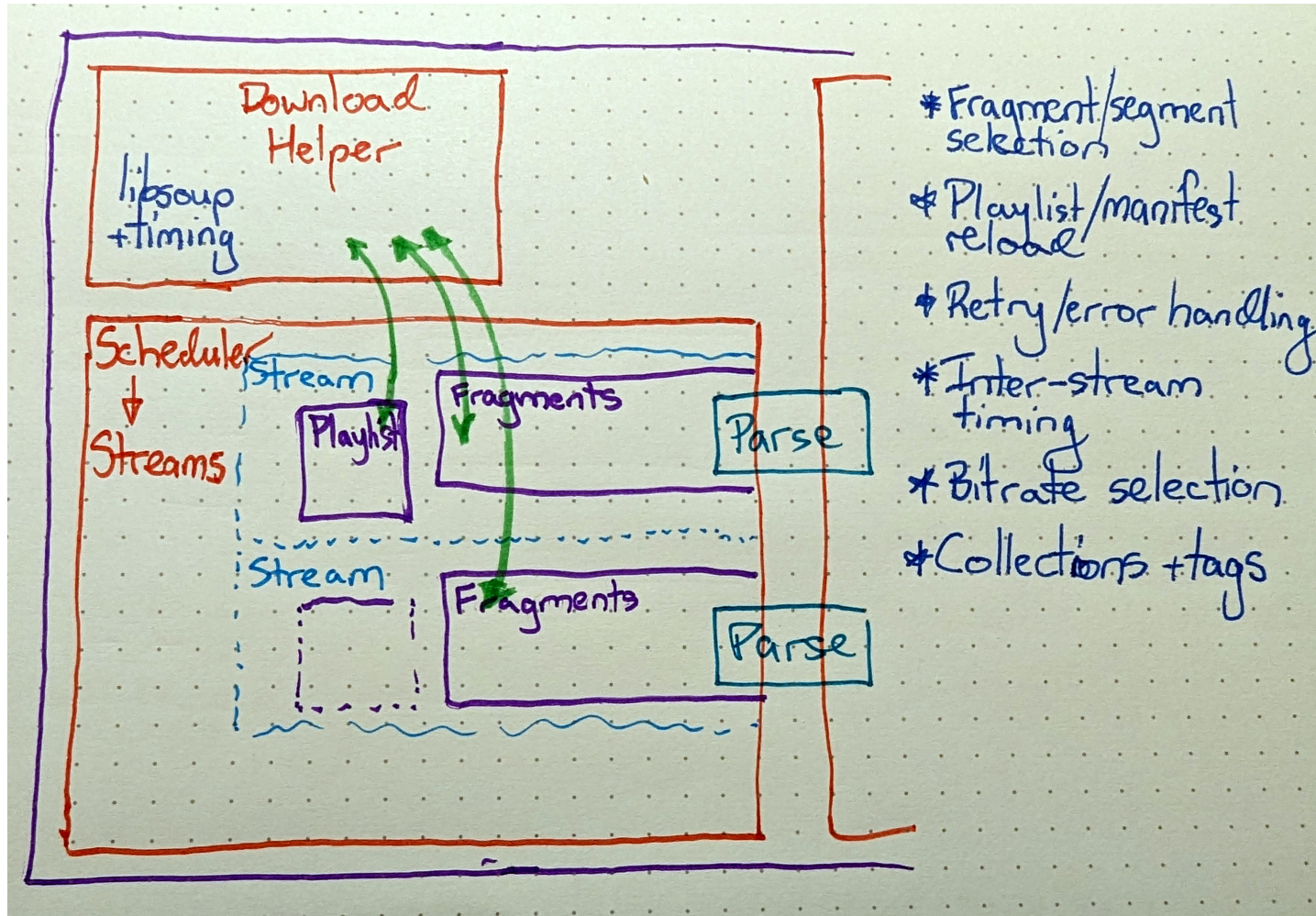


# Design

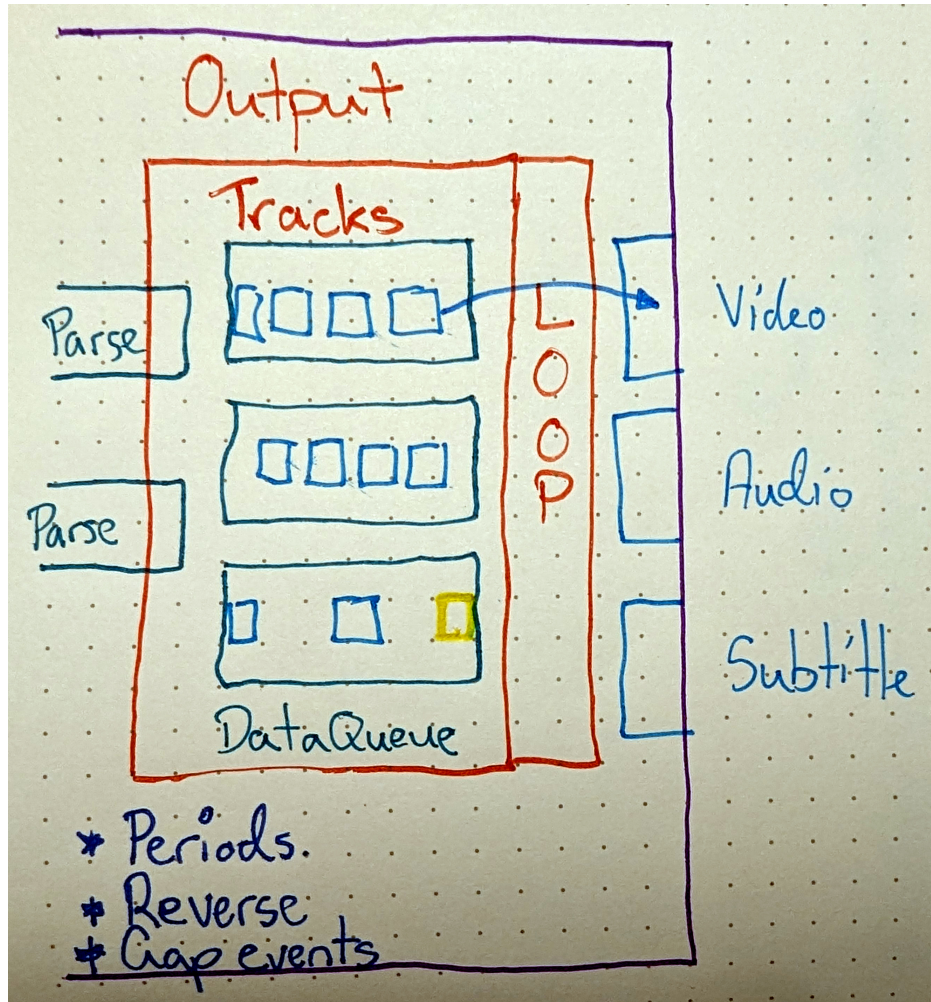




# Design - Scheduler



# Design - Output Loop



# New architecture benefits

- Much better inter-stream alignment logic
- Fewer threads, simpler locking logic
- Faster track/language selection switching
- Better buffering
- Ability to support LL-HLS

# LL-HLS

- Goal: Reduce end-to-end latency to  $< 10$  seconds

# LL-HLS

```
#EXTM3U
#EXT-X-TARGETDURATION:4
...
#EXTINF:4.00008,
fileSequence269.mp4
#EXTINF:4.00008,
fileSequence270.mp4
#EXT-X-PART:DURATION=2.00004, INDEPENDENT=YES, URI="filePart271.0.mp4"
#EXT-X-PART:DURATION=2.00004, URI="filePart271.1.mp4"
#EXTINF:4.00008,
fileSequence271.mp4
#EXT-X-PART:DURATION=2.00004, INDEPENDENT=YES, URI="filePart272.0.mp4"
#EXT-X-PART:DURATION=0.50001, URI="filePart272.1.mp4"
#EXTINF:2.50005,
fileSequence272.mp4
#EXT-X-DISCONTINUITY
#EXT-X-PART:DURATION=2.00004, INDEPENDENT=YES, URI="midRoll273.0.mp4"
#EXT-X-PART:DURATION=2.00004, URI="midRoll273.1.mp4"
#EXTINF:4.00008,
midRoll273.mp4
#EXT-X-PART:DURATION=2.00004, INDEPENDENT=YES, URI="midRoll274.0.mp4"
#EXT-X-PRELOAD-HINT:TYPE=PART, URI="midRoll274.1.mp4"
#EXT-X-RENDITION-REPORT:URI="/1M/LL-HLS.m3u8", LAST-MSN=274, LAST-PART=1
```

# LL-HLS

- Publishing smaller "Partial Segments" at the live edge
  - EXT-X-PART
- Preload mechanism
  - EXT-X-PRELOAD-HINT
- Shrinking playlist updates using Delta Playlists
  - EXT-X-SERVER-CONTROL: CAN-SKIP-UNTIL, HOLD-BACK=X
- Putting some inter-stream alignment hints in playlists
  - EXT-X-RENDITION-REPORT



# Demos