static void _f_do_barnacle_install_properties(GObjectClass *gobject_class)
{
    GParamSpec *pspec;
    /* Party code attribute */
    pspec = g_param_spec_uint64
        (F_DO_BARNACLE_CODE,
         "Barnacle code.",
         "Barnacle code",
         0,
         G_MAXUINT64,
         G_MAXUINT64 /* default value */,
         G_PARAM_READABLE |
         G_PARAM_WRITABLE |
         G_PARAM_PRIVATE);
    g_object_class_install_property (gobject_class,
    F_DO_BARNACLE_PROP_CODE,
    pspec);
}

Enrique Ocaña González
eocanha@igalia.com
Motivation for MSE

- HTML5 video tag:
  
  `<video src="movie.mp4" type="video/mp4" />

- Improvement: Blob URI

- Adaptive streaming and time shift limitations

- Solution: Media Source Extensions (MSE)
  
  [https://w3c.github.io/media-source/](https://w3c.github.io/media-source/)

- JavaScript can generate and feed video data

- More control on the player state
How to use MSE

- API for JavaScript:
  - HTMLMediaElement, Blob
  - MediaSource, SourceBuffer
  - {Audio, Video, Text}Track

```html
<video id="video"></video>

<script>
  var video = document.getElementById('video');
  var ms = new MediaSource();

  ms.addEventListener('sourceopen', function() {
    var videoSb = ms.addSourceBuffer('video/mp4; codecs="avc1.640028"');
    var audioSb = ms.addSourceBuffer('audio/mp4; codecs="mp4a.40.2"');
    videoSb.appendData(...);
    audioSb.appendData(...);
  });

  var blobUrl = URL.createObjectURL(ms);
  video.src = blobUrl;
</script>
```
Implementation: 1st attempt

PlayBin

URIDecodeBin

WebKitMediaSrc

Other streams (in theory)

AppSrc → TypeFind → QtDemux → MultiQueue → Parser

AppSrc → TypeFind → QtDemux → MultiQueue → Parser

encoded video

encoded audio

encoded video

encoded audio

DecodeBin

WebKitVideoSink

Encoded audio

WebKitVideoSink

Encoded video

AudioSink

Raw audio

Raw video
Implementation: 1\textsuperscript{st} attempt

- Single pipeline model based on PlayBin
- WebKitMediaSrc:
  - Multiple streams $\leftrightarrow$ SourceBuffers
  - Buffer stealing and reinjection

- Limited to H.264 + audio mpeg
- PlaySink
- Limitations:
  - appendBuffer and playback not independent
  - Append out of order
  - Seeks
Implementation: 2\textsuperscript{nd} attempt

- Split pipeline model
- Append pipeline (N instances, one per SourceBuffer)

- State machine to control append lifecycle and coordinate reporting of data and events to the upper SourceBuffer layer
- Playback pipeline (1 instance)
QtDemux & appendBuffer quirks

- appendBuffer accepts binary data
  - No clue about timestamps, blind demuxing
  - No problem with sequential appends
  - Problem with out-of-order segment appends
  - QtDemux must honor TFDT atom:
    - Bug 754230: `qtdemux`: Use the tfdt decode time on byte streams when it's significantly different than the time in the last sample
    - Bug 780410: `qtdemux`: distinguish TFDT with value 0 from no TFDT at all
- How to detect the end of append processing
  - Poor way: Probes and timeouts (we didn't know anything better yet)
  - Right way: AppSrc in AppendPipeline emits `need-data`
Adaptive streaming quirks

- Basically a change in caps on the fly
- In AppendPipeline:
  - QtDemux removes/creates pads → Reattach
  - Multiple tracks → Match track ids somehow?
- In PlaybackPipeline:
  - Flush & reenqueue (WebKit optimization to get best quality ASAP)
    - Video stream
      - Get current position
      - Insert flush start/stop events by hand
      - Abuse `gst_base_src_new_seamless_segment()` to insert new segment on AppSrc.
      - Use a `segmentFixerProbe` to realign the segment after AppSrc
    - Audio stream
      - No flush, just rely on the decoder/sink to skip inappropriate buffers
Seek quirks

- Coordinate WebKit and GStreamer layers, asynchronously
- Wait until data for target position is buffered
- `gst_element_seek(playbin)`
- `WebKitMediaSrc: seek-data` and `need-data` on all the AppSrcs (all the streams)
- MediaSource seek → provideMediaData
Final result...
Future work

- Bugfixing after multiplatform code changes
- Webm (vp9/opus) support
- Multiple streams per SourceBuffer
- Improve w3c tests passrate
Source code

• WebKitGTK+
  https://github.com/WebKit/webkit

• WPE customization for Raspberry Pi 2
  https://github.com/WebPlatformForEmbedded/WPEWebKit
Thank you!

Questions?