

Improving Accessibility Services using WebAssembly

POCs for Enhancing Subtitling and Sign Language

Rafael Bermúdez Guijo
Product Owner Research Team at
3Cat

12-03-2025



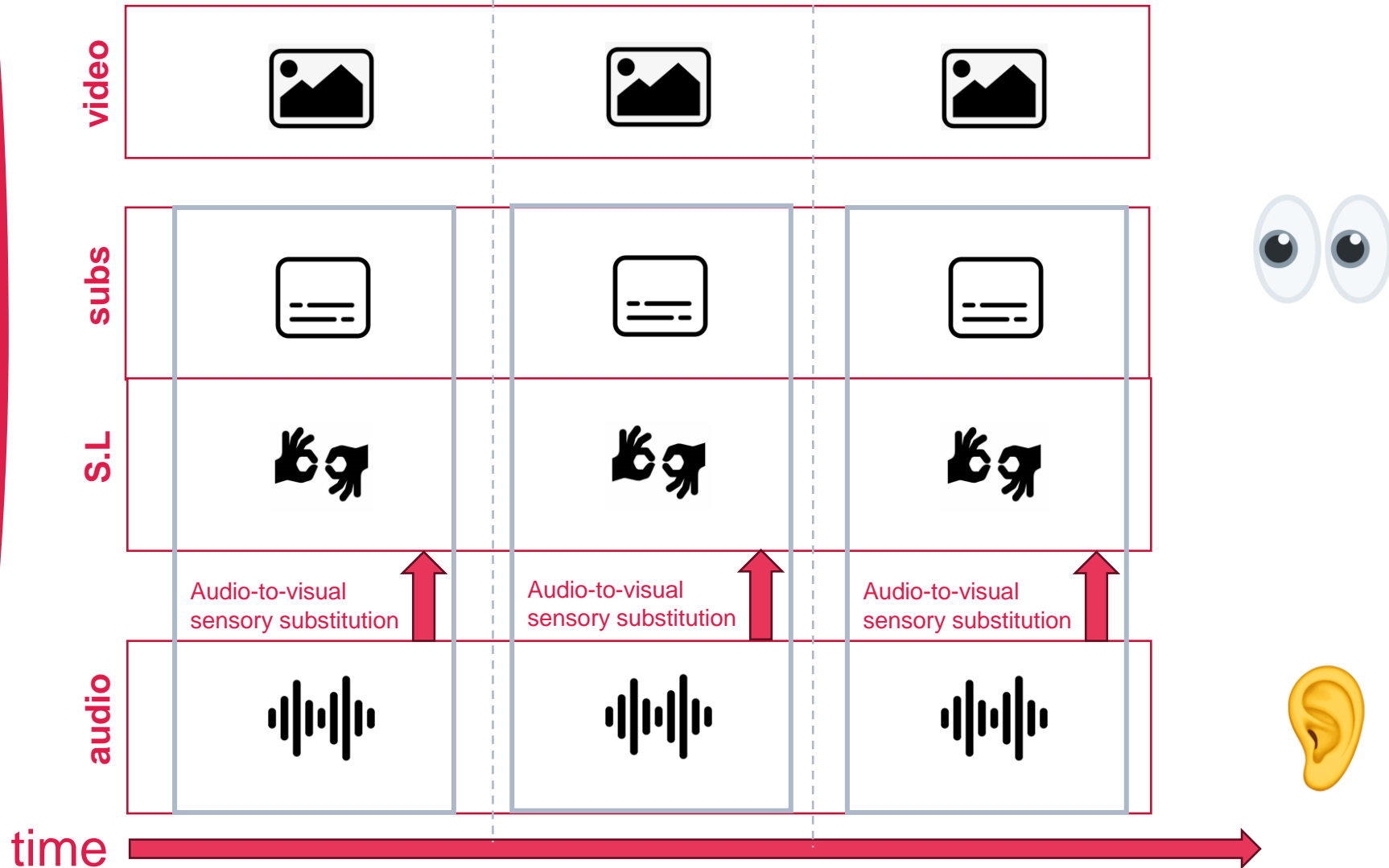
**Why
accessibility
needs
WebAssembly**



Why does accessibility need Wasm?

- ☐ It's cool
- ☐ We need to perform CPU-intense tasks
- ☐ We need to perform time-constrained tasks
- ☐ All of the above

Why accessibility needs WebAssembly — Synchronization Is Key

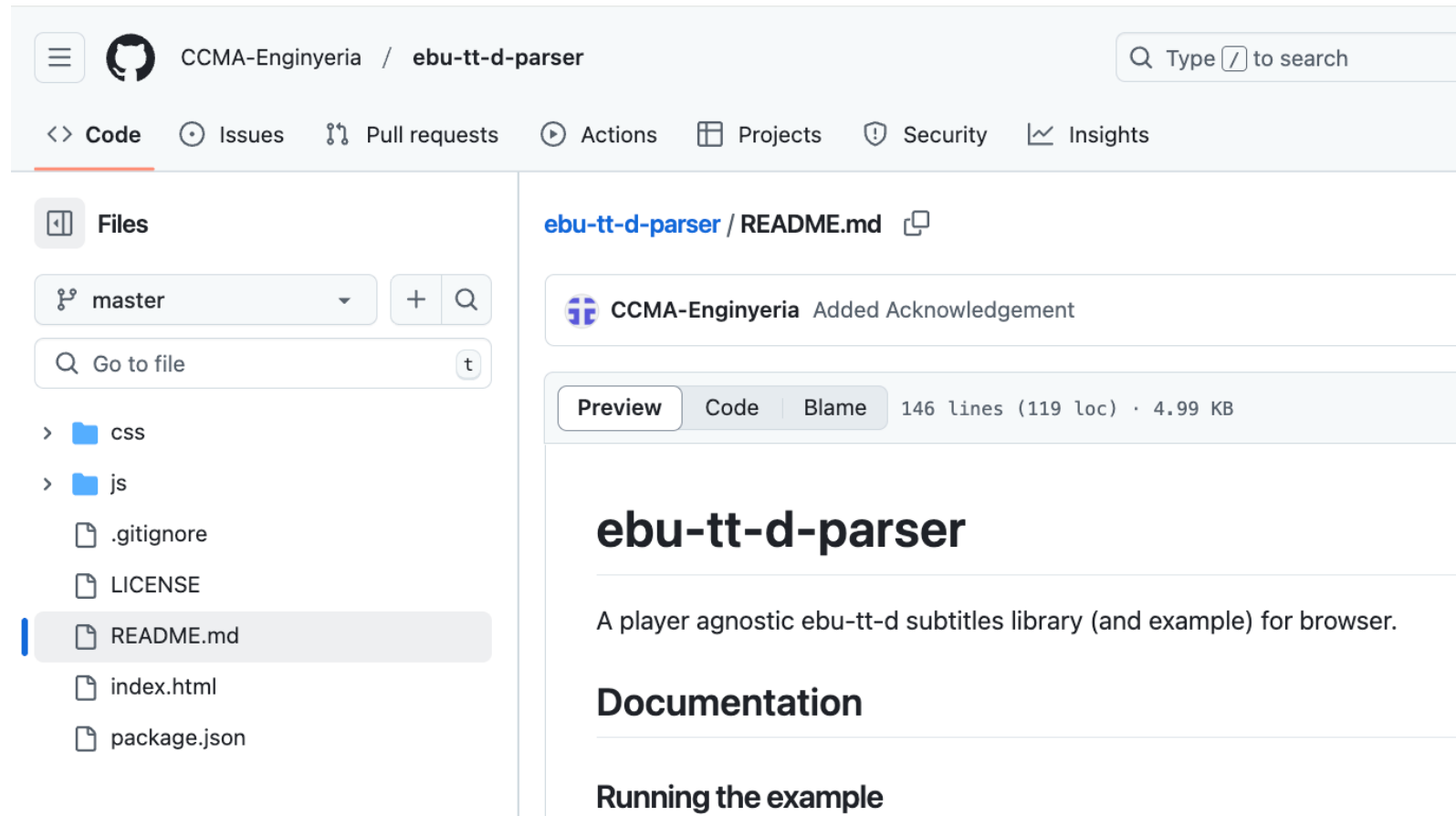


Subtitling with Wasm

```
subtitles.xml ×  ⓘ README.md  <> comparison.html M  ≡ Proof of Concepts & Demonstrations Un

> subtitles.xml
<tt:tt xmlns:tt="http://www.w3.org/ns/ttml"
  <tt:body>
    <tt:div>
      <tt:p xml:id="p3" region="r3" begin="00:00:07.400" end="00:00:09.880">
        <tt:span style="ss2">un "dating" on pots trobar l'amor,</tt:span>
        <tt:br />
        <tt:span style="ss2">o no.</tt:span>
      </tt:p>
      <tt:p xml:id="p4" region="r3" begin="00:00:09.960" end="00:00:12.080">
        <tt:span style="ss2">A cada capítol un o una protagonista</tt:span>
      </tt:p>
      <tt:p xml:id="p5" region="r3" begin="00:00:12.160" end="00:00:15.800">
        <tt:span style="ss2">vindrà buscant l'amor</tt:span>
        <tt:br />
        <tt:span style="ss2">i tindrà l'oportunitat de trobar-lo</tt:span>
      </tt:p>
      <tt:p xml:id="p6" region="r3" begin="00:00:15.880" end="00:00:18.520">
        <tt:span style="ss2">amb una de les quatre persones interessades, però, espòiler!,</tt:span>
      </tt:p>
      <tt:p xml:id="p7" region="r3" begin="00:00:18.600" end="00:00:22.160">
        <tt:span style="ss2">n'hi ha dues que no volen l'amor, volen els diners.</tt:span>
      </tt:p>
      <tt:p xml:id="p8" region="r3" begin="00:00:22.240" end="00:00:25.120">
        <tt:span style="ss2">Convèncer algú a contrarellotge</tt:span>
        <tt:br />
        <tt:span style="ss2">no és gens fàcil.</tt:span>
      </tt:p>
      <tt:p xml:id="p9" region="r3" begin="00:00:25.200" end="00:00:27.920">
```

Subtitling with JS



CCMA-Enginyeria / ebu-tt-d-parser

Code Issues Pull requests Actions Projects Security Insights

Files

master + 🔍

Go to file t

- > css
- > js
- .gitignore
- LICENSE
- README.md**
- index.html
- package.json

ebu-tt-d-parser / README.md 📄

CCMA-Enginyeria Added Acknowledgement

Preview Code Blame 146 lines (119 loc) · 4.99 KB

ebu-tt-d-parser

A player agnostic ebu-tt-d subtitles library (and example) for browser.

Documentation

Running the example

Subtitling with JS — The Results



- It ran on all devices, but on low-end devices, the user experience was poor—**subtitles** could **go out of sync** and sometimes **overlap**.

Subtitling with WebAssembly — Does It Make Sense?

- ✓ **High CPU consumption** for parsing big subtitles files
- ✓ **Precise timing** is critical for displaying subtitles in the correct timecode

Subtitling with WebAssembly – Compiling to WebAssembly



hbbtv-wasm-use-cases / subtitles / js / subtitles.js

Code Blame 734 lines (707 loc) · 21.8 KB

```
3  var SubtitlesPlugin = function (el, playlist) {
4    this.el = el;
5    this.defaultStyles = null;
6    //this.playlist = playlist;
7
8    this.refresh = _.bind(this.refresh, this);
9    this.unload = _.bind(this.unload, this);
10   this.clearAndRefresh = _.bind(this.clearAndRefresh, this);
11
12   //this.playlist.addEvent('fullscreen', this.refresh);
13   //this.playlist.addEvent('destroy', this.unload);
14   //this.playlist.addEvent('seeking', this.clearAndRefresh);
15   //this.playlist.addEvent('seeked', this.clearAndRefresh);
16   this.setDefaultPersonalization();
17 };
```



hbbtv-wasm-use-cases / subtitles / src / subtitles.rs

Code Blame 551 lines (510 loc) · 17.6 KB

```
226
227  pub struct Subtitles {
228      pub tt: Option<TT>,
229      pub cuepoints: cuepoints::Cuepoints,
230      pub cuepoint_to_subtitles_action: HashMap<String
231      pub styles_index: HashMap<String, usize>,
232      pub region_index: HashMap<String, usize>,
233      pub default_styles: String,
234      pub tt_root_config: TTRootConfig,
235      pub element_size: ElementSize,
236  }
237
```

- ✓ Better memory management
- ✓ Better error handling
- ✓ Static Typing (personal preference)

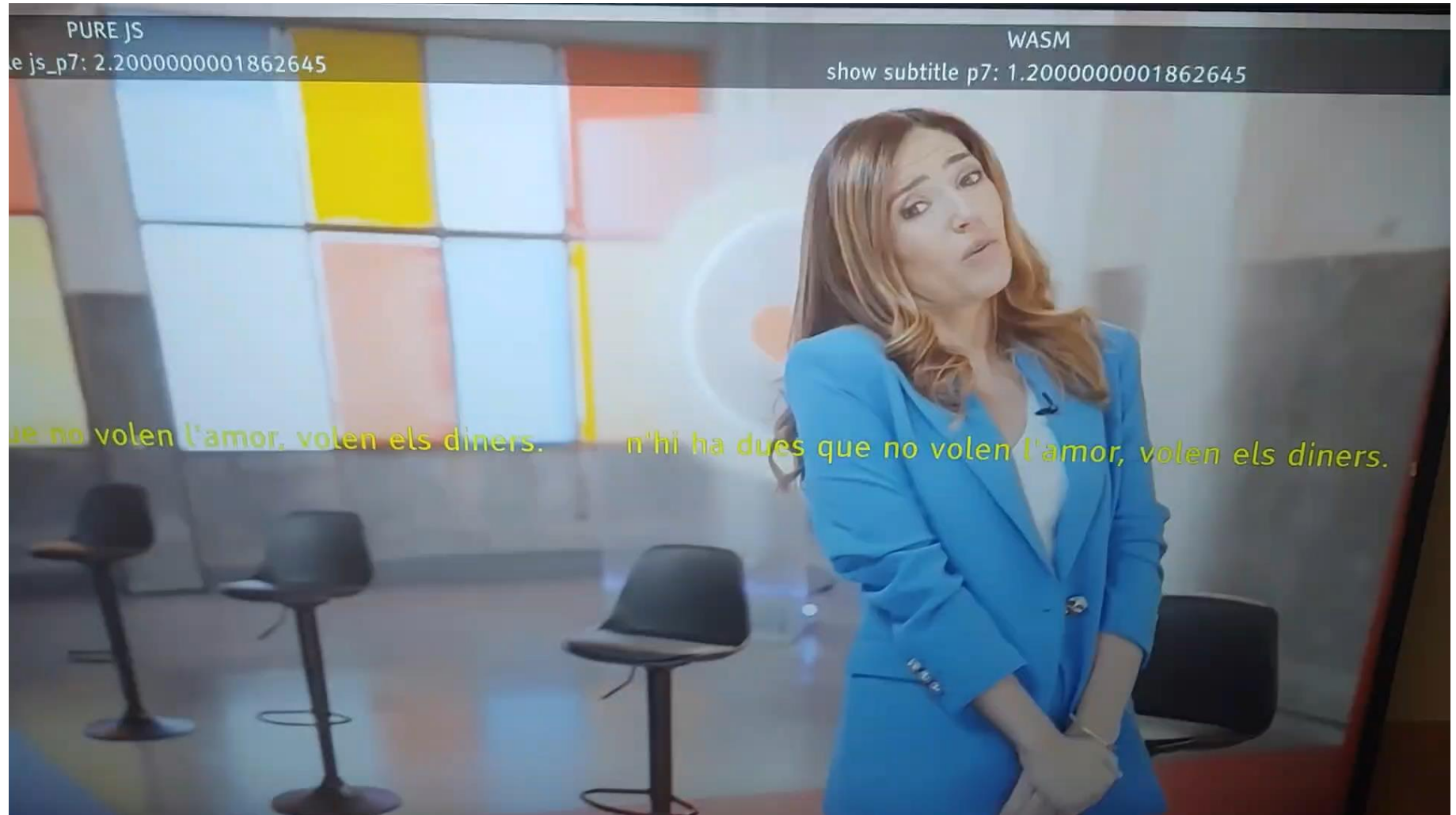
JS vs Wasm — Testing on desktop browser



JS vs Wasm — Performance on desktop browser

| ACTION | MEAN TIME IN JS (ms) | MEAN TIME IN WASM (ms) | IMPROVEMENT (%) |
|------------------------|-------------------------|---------------------------|-----------------|
| EBU-TT-D PARSING | 70,20 | 37,20 | 47% |
| SUBTITLES RENDERING | 0,36865 | 0,13766 | 62,6% |

JS vs Wasm — Testing on hbbtv device



JS vs Wasm — Performance on hbbtv device

| ACTION | MEAN TIME IN JS (ms) | MEAN TIME IN WASM (ms) | IMPROVEMENT (%) |
|------------------------|-------------------------|---------------------------|-----------------|
| EBU-TT-D PARSING | 1650,30 | 231,04 | 86% |
| SUBTITLES RENDERING | 3,80 | 1,06 | 72% |

Sign Language with WASM

3cat



Best Sign Language Service

- The best way to integrate sign language is by playing a **second video**.
- But with **only one decoder available**, the second video must use **software decoding**.

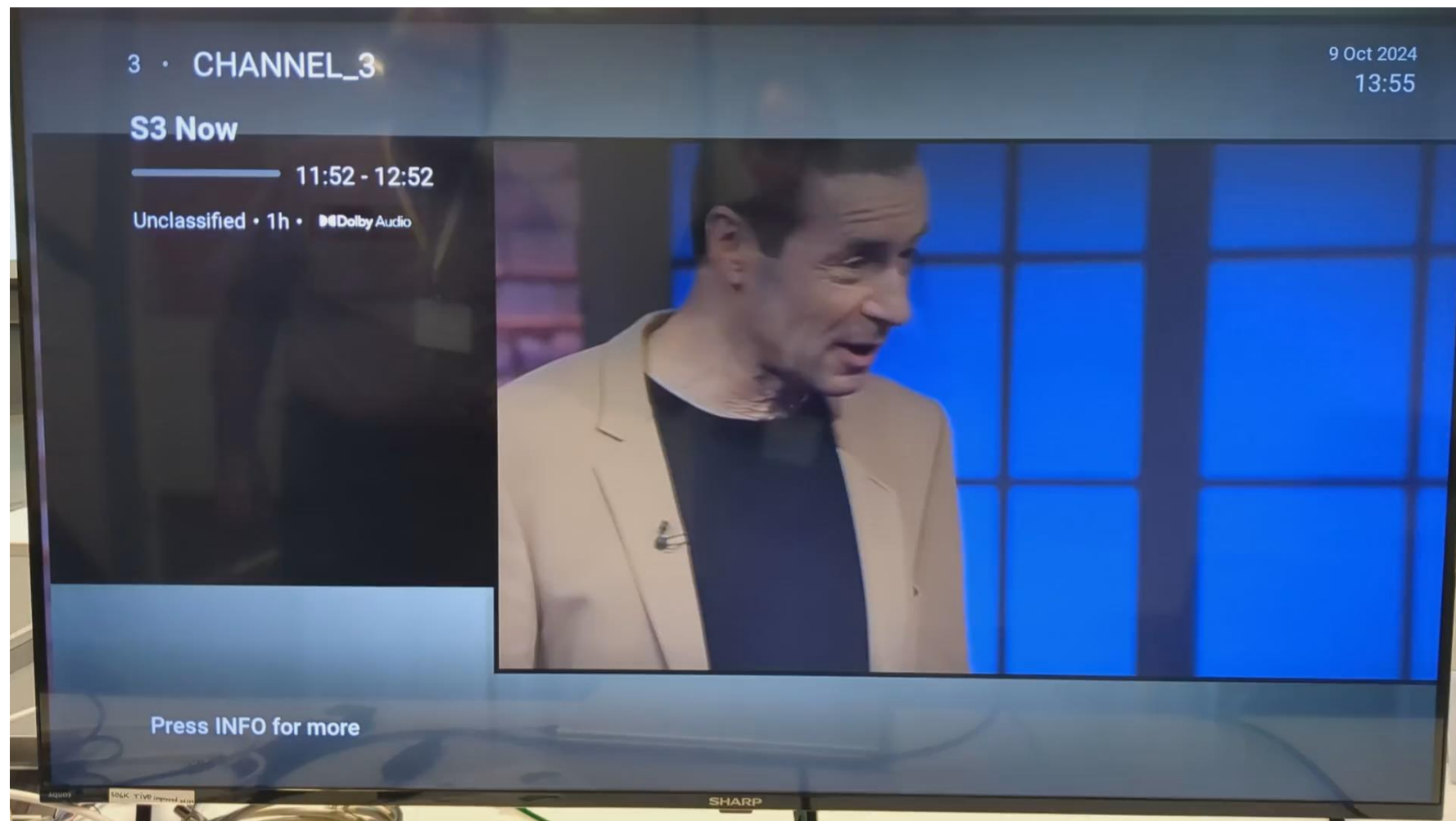
WebAssembly for software decoding — Does It Make Sense?

- ✓ **High CPU consumption** when decoding each frame
- ✓ **Strict timing requirements** to maintain frame rate
- ✓ **Leverage existing codecs** written in low-level languages

WebAssembly for software decoding — First test on HbbTv



POC for Plugfest Berlin 2024



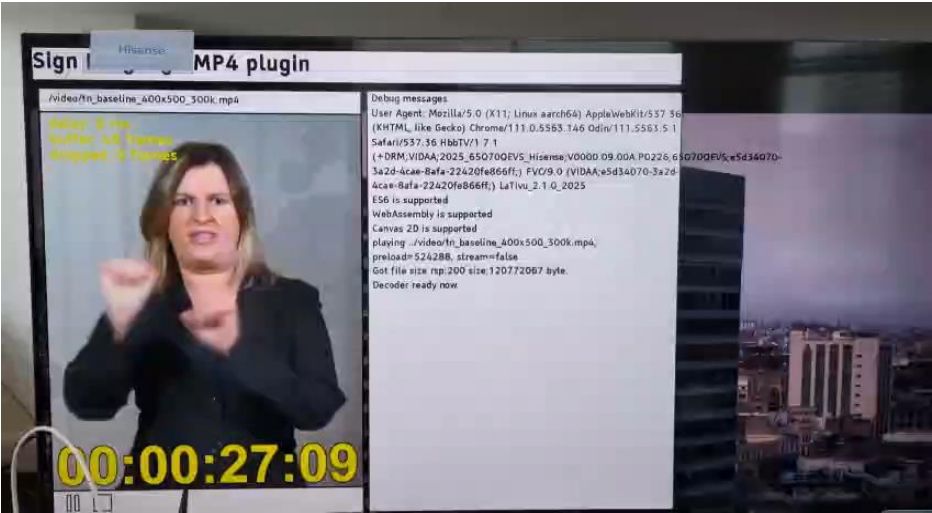
POC for Plugfest Berlin 2024 — Results

| Device | Runs Without Issues | Runs With Issues | Does not run |
|--------|---------------------|------------------|--------------|
| 1 | X | | |
| 2 | X | | |
| 3 | X | | |
| 4 | X | | |
| 5 | | X | |
| 6 | X | | |
| 7 | X | | |

POC for Plugfest London 2025



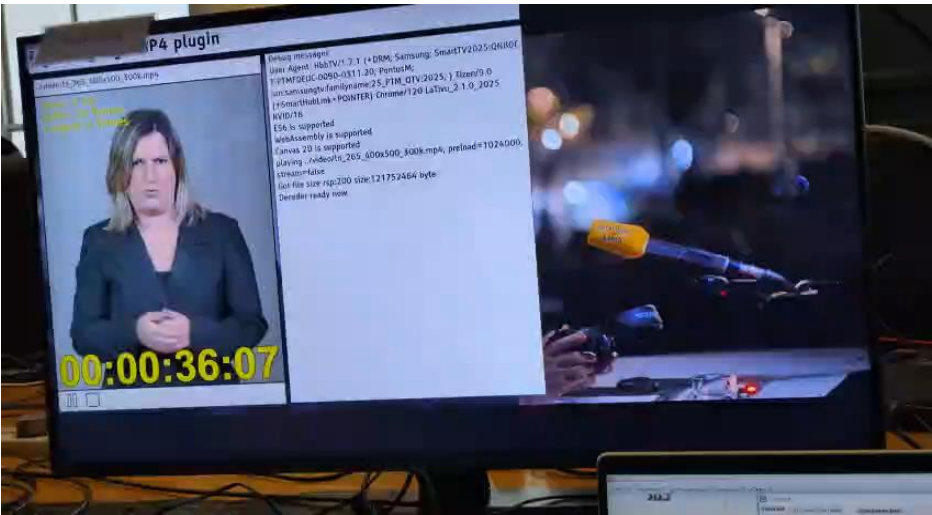
POC for Plugfest London 2025 — H264



| | |
|------------|-----------|
| Format | MP4 |
| Codec | H264 |
| Profile | Baseline |
| Resolution | 400 x 500 |
| Bitrate | 300kbps |

| | Zero Drops | Some Drops | A Lot of Drops | Does not run |
|-----------|------------|------------|----------------|--------------|
| # Devices | 5 | 3 | 1 | 0 |

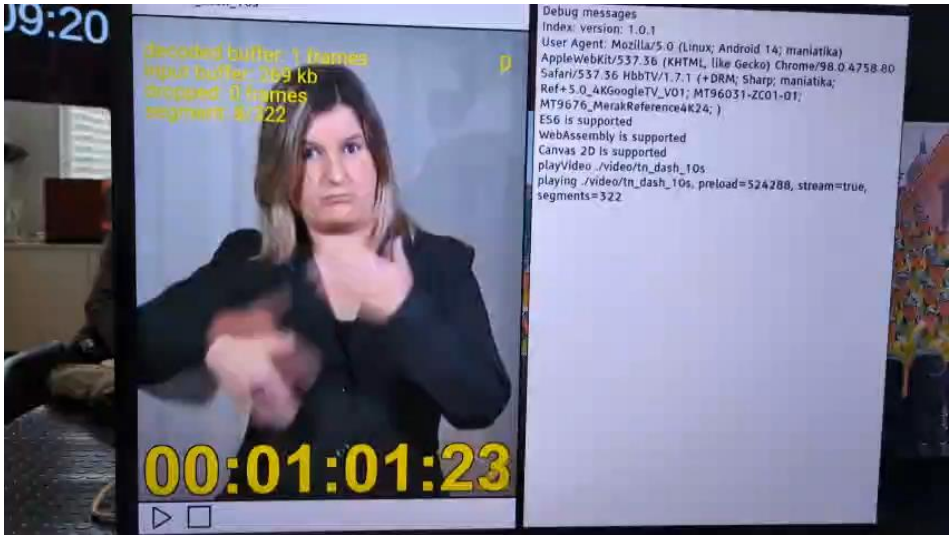
POC for Plugfest London 2025 — H265



| | |
|------------|-----------|
| Format | MP4 |
| Codec | H265 |
| Profile | Baseline |
| Resolution | 400 x 500 |
| Bitrate | 300kbps |

| | Zero Drops | Some Drops | A Lot of Drops | Does not run |
|-----------|------------|------------|----------------|--------------|
| # Devices | 3 | 4 | 2 | 0 |

POC for Plugfest London 2025 — DASH



| | |
|------------|-----------|
| Format | MPEG-DASH |
| Codec | H264 |
| Profile | Baseline |
| Resolution | 400 x 500 |
| Bitrate | 300kbps |

| | Zero Drops | Some Drops | A Lot of Drops | Does not run |
|-----------|------------|------------|----------------|--------------|
| # Devices | 6 | 3 | 0 | 0 |

Why does accessibility need Wasm?

- ☐ It's cool
- ☐ We need to perform CPU-intense tasks
- ☐ We need to perform time-constrained tasks
- ☒ All of the above

Conclusions

- WebAssembly could help us improve our services, especially on low-end devices.
- WebAssembly will enable us to create amazing new services.
- Start now—but with caution! Some devices in the market already support WebAssembly.

**Thank you for your
attention**

