Making Passthrough Audio Just Work™ On Linux

Arun Raghavan
Collabora

Some hardware can decode compressed audio

A/V receivers
Bluetooth headsets
Embedded SoCs

More efficient
Saves power
Multichannel (S/PDIF)

There's a problem ...

We measure timing in bytes

Codecs don't work like that

Enter IEC 61937 (or 958. or 60958.) Fixed # of samples per frame

State Of The Art

😐 ALSA passthrough with mplayer/XBMC/VLC

Bluetooth

☹ SoCs vendors just NIH

~ Windows 7

~ OS X

How we want to do it ...

✔ **Reuse existing API**

✔ Handle dynamic changes

✔ **Single point of config, if any**

❌ **Handle fallback path in lower layers**

How we did it ...

Extend PulseAudio API

```
struct pa_format_info {
    /* Represents a PCM/compressed format. */

    /* Used while connecting a stream, and */
    /* to query devices.                   */
};
```

`pa_stream_new_extended()`

Format negotiation

client ⟶ requested formats ⟶ server

client ⟵ selected format ⟵ server

Notification on format change

Device formats API

The rest of the stack ...

GStreamer

pulsesink
somethingsink
playbin2 (go go slomo!)

BlueZ

Formats UI
(courtesy SuperColin)

When can I have it?

What's left

Shiny UI
(and decide good policy)

HDMI + ELD

BlueZ – use Media API

ALSA passthrough
(Pierre and Vinod's talk)

Questions?

Thank you!