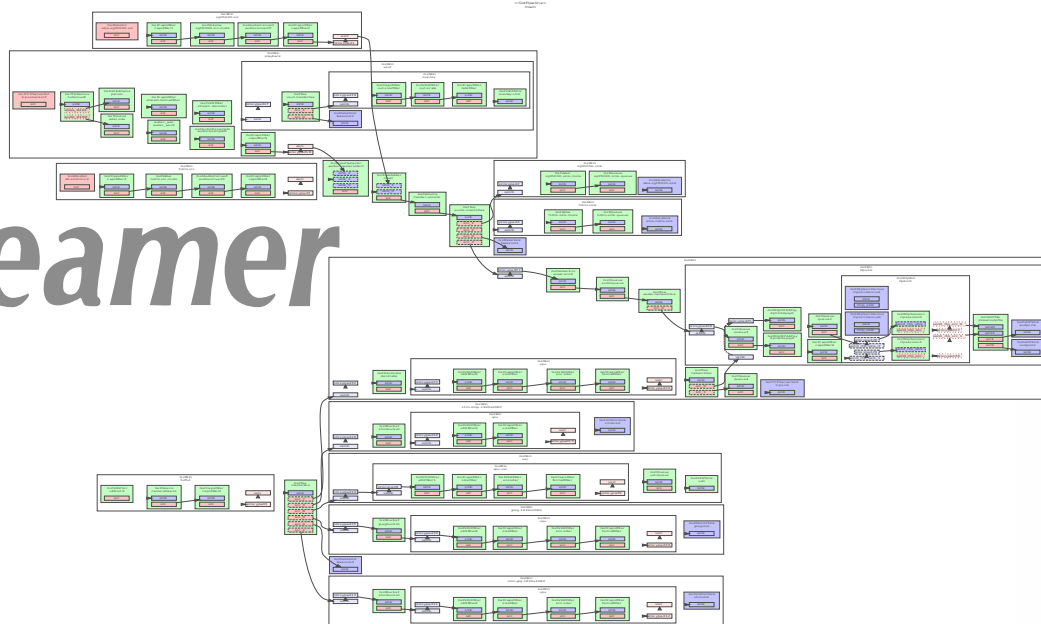# Multimedia on Embedded Linux
Pushing GStreamer to its Limits

GStreamer Conference 2013

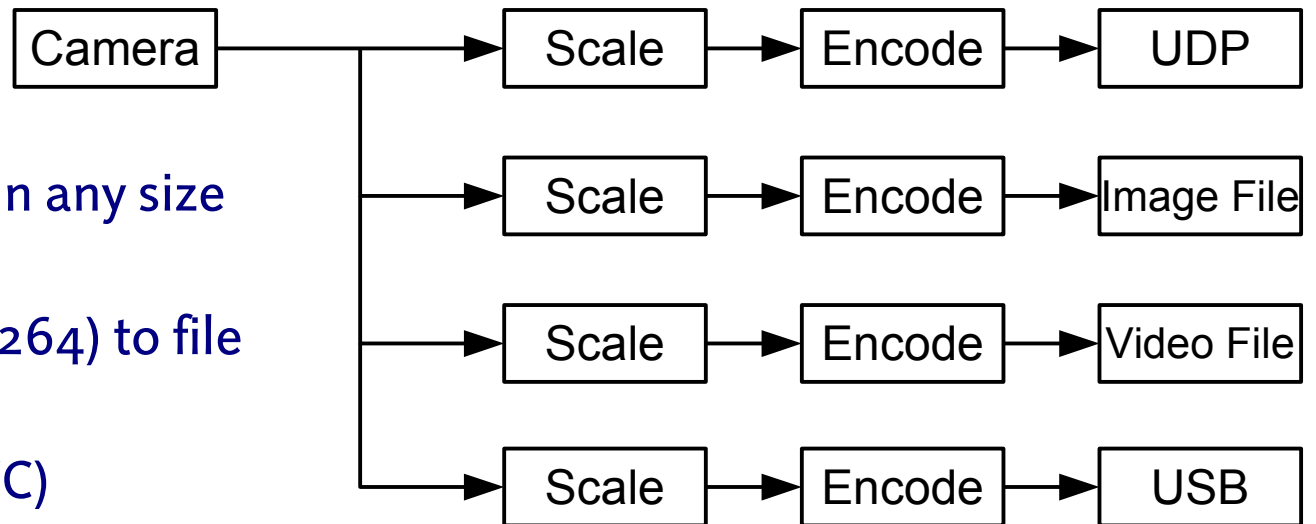**Michael Olbrich <m.olbrich@pengutronix.de>**

Who am I?

- Michael Olbrich

- Linux developer at Pengutronix since 2008
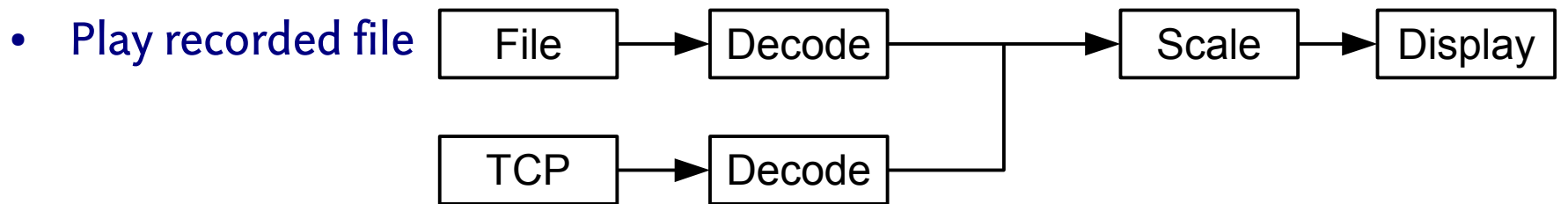
- GStreamer development since 2011

Pengutronix.

The Project

- FullHD Camera @ 30 fps

- Network Stream: H.264/MJPG 720p and smaller

- JPEG Snapshots in any size

- Record Video (H.264) to file

- Write to USB (UVC)

- Play recorded file

- Play H.264 network stream

```
Camera ──┬──────▶ Scale ──▶ Encode ──▶ UDP
         ├──────▶ Scale ──▶ Encode ──▶ Image File
         ├──────▶ Scale ──▶ Encode ──▶ Video File
         └──────▶ Scale ──▶ Encode ──▶ USB

File ──▶ Decode ──┬──▶ Scale ──▶ Display
TCP ──▶ Decode ───┘
```
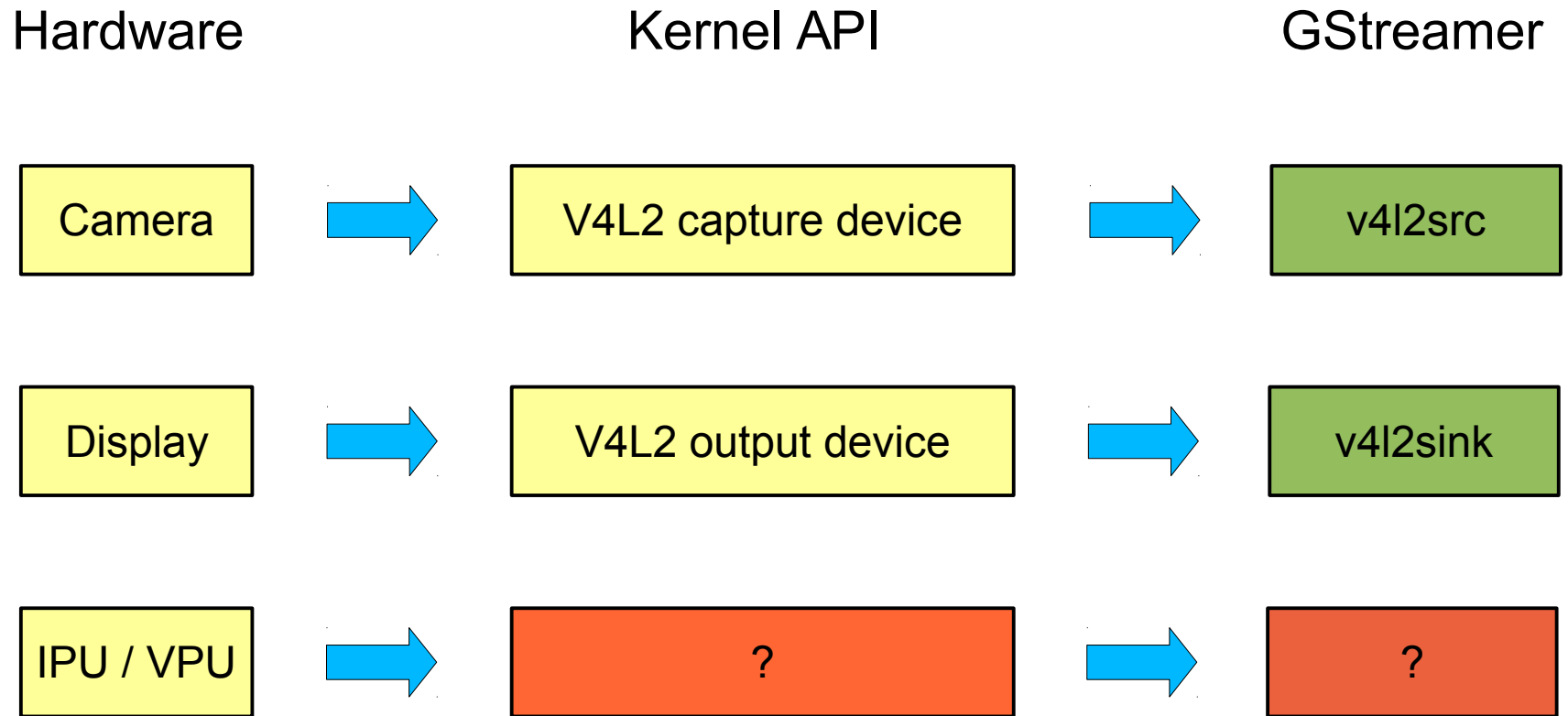
Pengutronix

i.MX53 Hardware Features

- i.MX53: 1 GHz Cortex A8

- Camera interface

- Display interface

- IPU: scaling & color space conversion

- VPU: H.264 & JPEG en-/decoding

Pengutronix

Using the Hardware in GStreamer

Hardware		Kernel API		GStreamer
Camera	→	V4L2 capture device	→	v4l2src
Display	→	V4L2 output device	→	v4l2sink
IPU / VPU	→	?	→	?

Pengutronix

Performance - Software

```
gst-launch-1.0 v4l2src device=/dev/video0 !
        video/x-raw,width=1920,height=1080 !
        videoscale !
        video/x-raw,width=1280,height=720 !
        fakesink sync=false
```
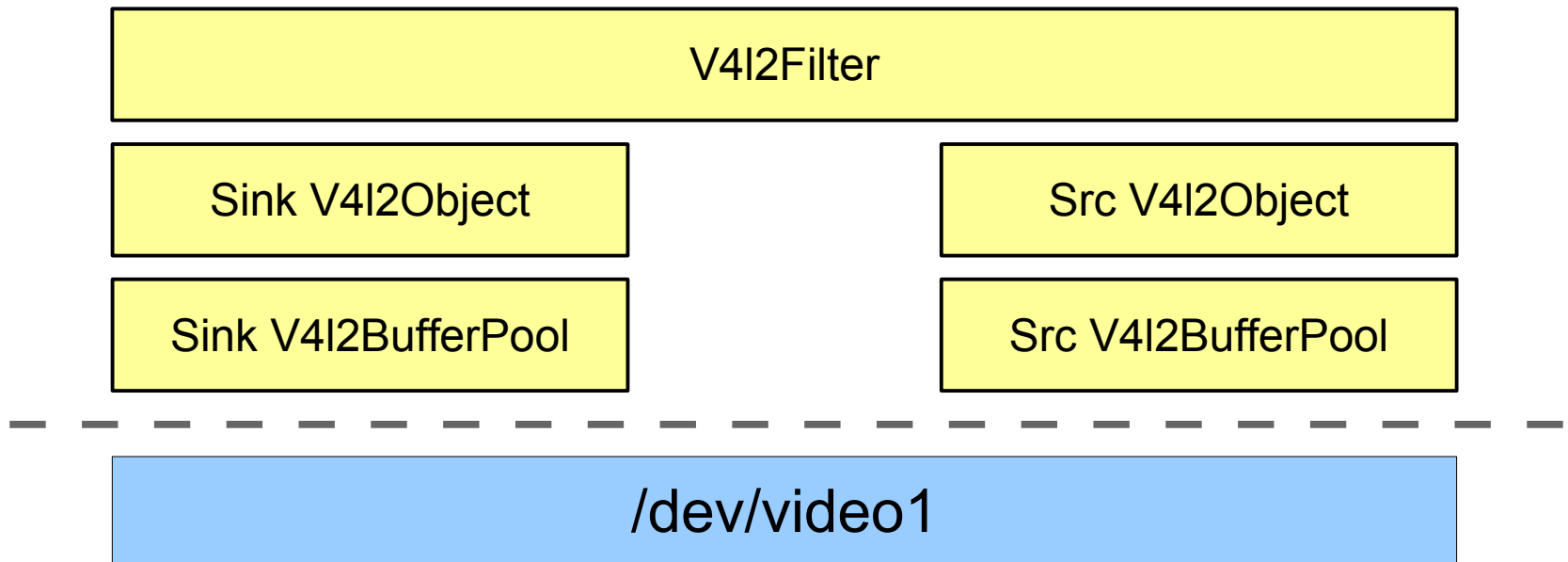
V4L2 mem2mem and GStreamer

- Existing: Freescale en-/decoder plugin

  - GStreamer 0.10 only

  - Custom Kernel API (this will never be mainline)

- V4L2 mem2mem devices

  - V4L2 capture and output in one device

  - One GStreamer plugin to rule them all

**Pengutronix**

GstV4l2Filter

V4l2Filter – Details

- No GstBaseTransform, GstVideoDecoder, GstVideoEncoder

- Best effort caps negotiation

- QOS inspired by GstVideoDecoder

- Basic event and meta data handling

- No good latency calculation

Pengutronix

Performance - Hardware

```
gst-launch-1.0 v4l2src device=/dev/video0 !
    video/x-raw,width=1920,height=1080 !
    v4l2filter device=/dev/video1 !
    video/x-raw,width=1280,height=720 !
    fakesink sync=false
```
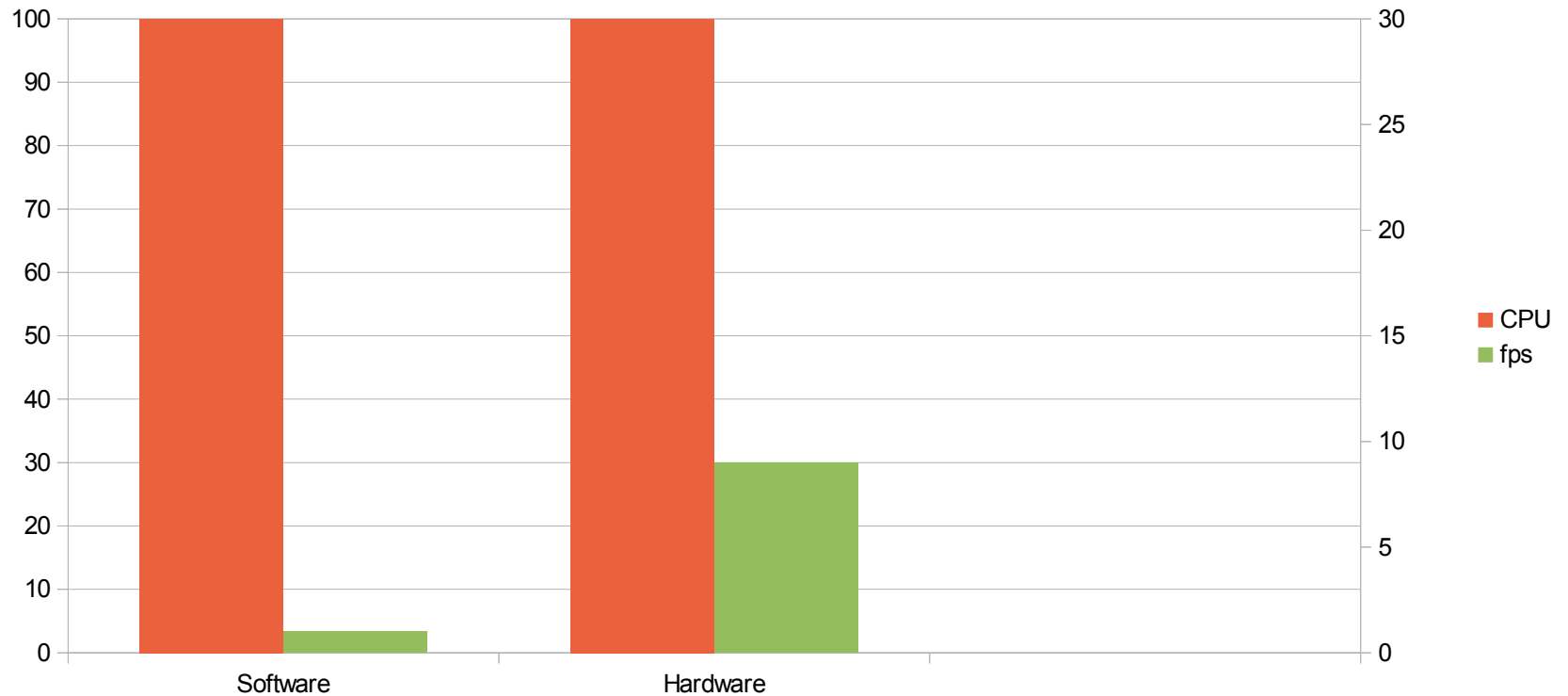
Zero-Copy with GStreamer and V4L2

- V4L2 supports 'userptr' and 'dmabuf'

- dmabuf has a cleaner API

- dmabuf should work with other plugins in the future

  - Wayland

  - OpenGL (Mesa)

Pengutronix

Zero-Copy with GStreamer and V4L2

- Improve dmabuf GstMemory and GstAllocator

- Create buffers with dmabuf memory in GstV4l2BufferPool
  - Needs a kernel patch to get writable buffers

- Use foreign dmabuf buffers instead of copying the content
  - Ugly hack – GstV4l2BufferPool is not designed to handle foreign buffers
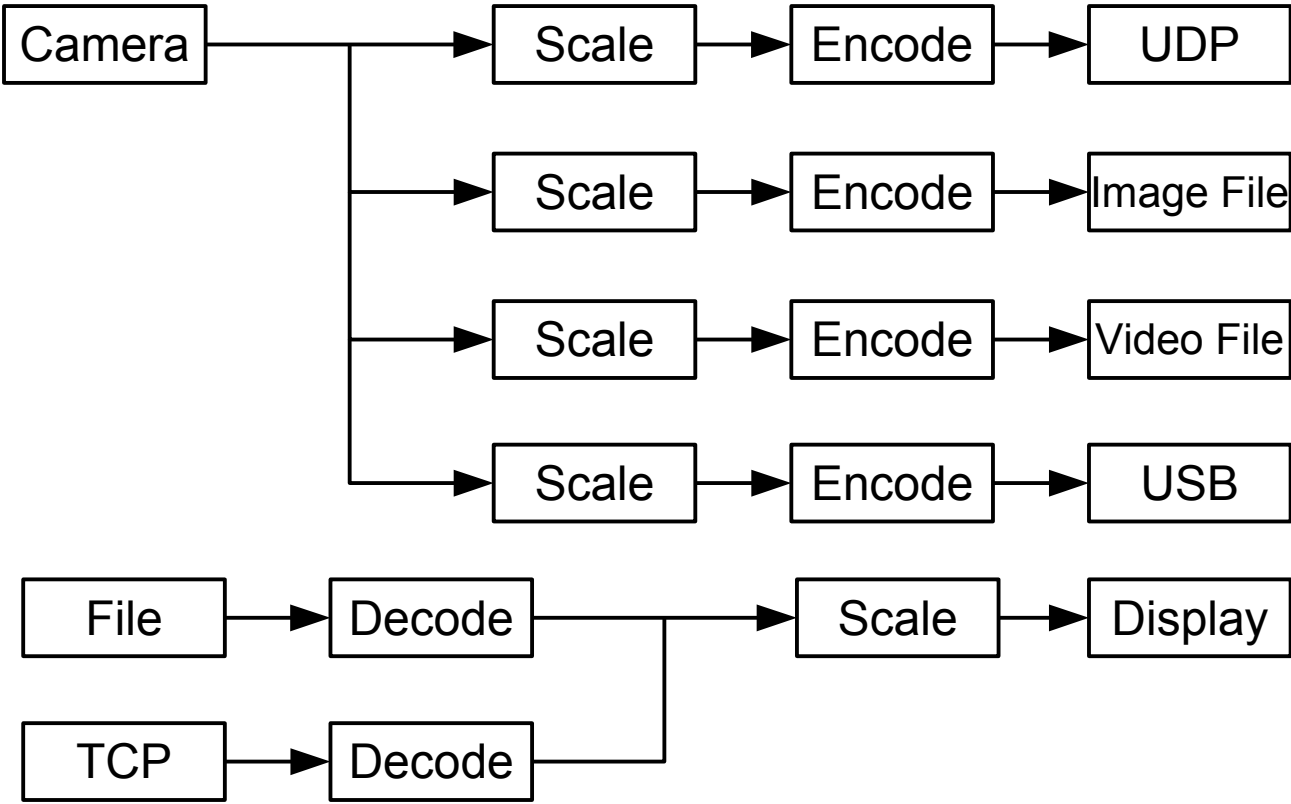
Pengutronix

Performance – Hardware (zero-copy)

```
gst-launch-1.0 v4l2src device=/dev/video0 !
        video/x-raw,width=1920,height=1080 !
        v4l2filter device=/dev/video1 !
        video/x-raw,width=1280,height=720 !
        fakesink sync=false
```
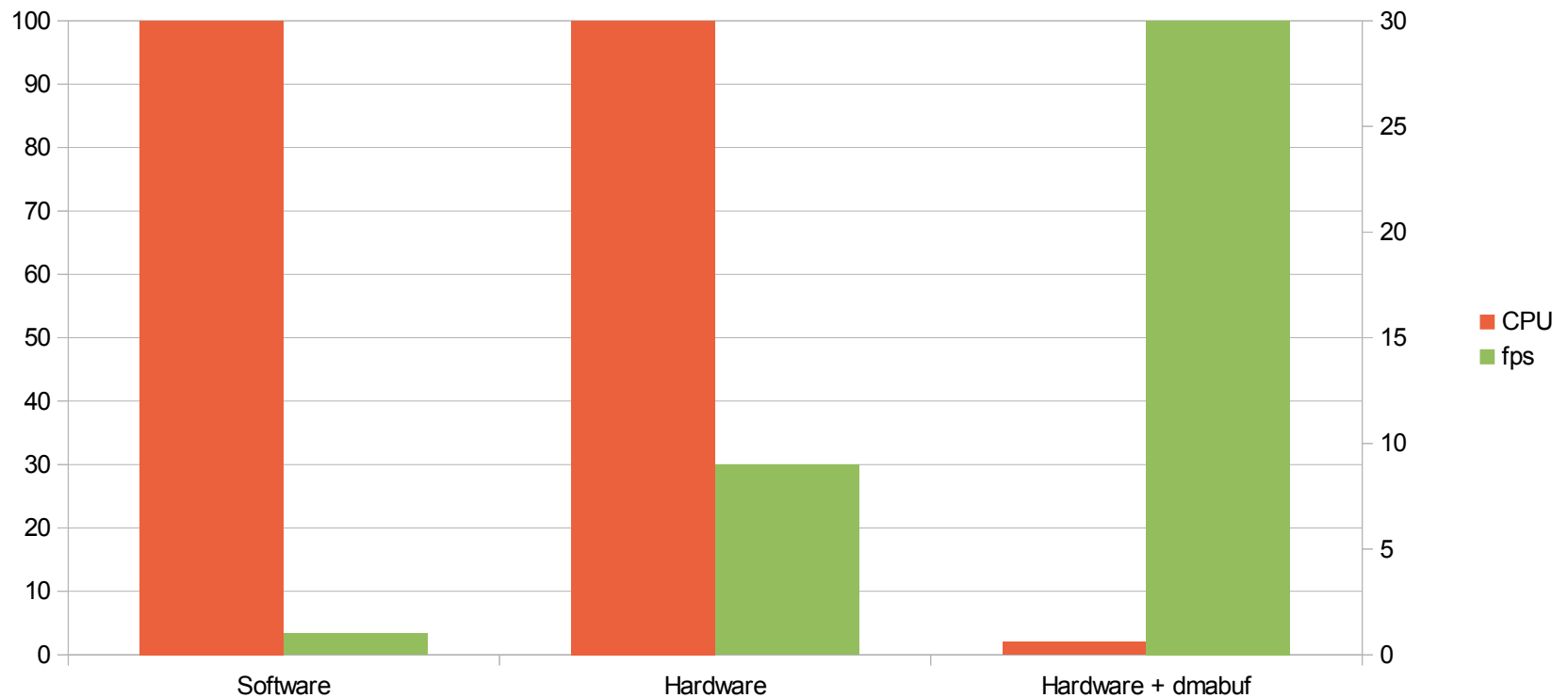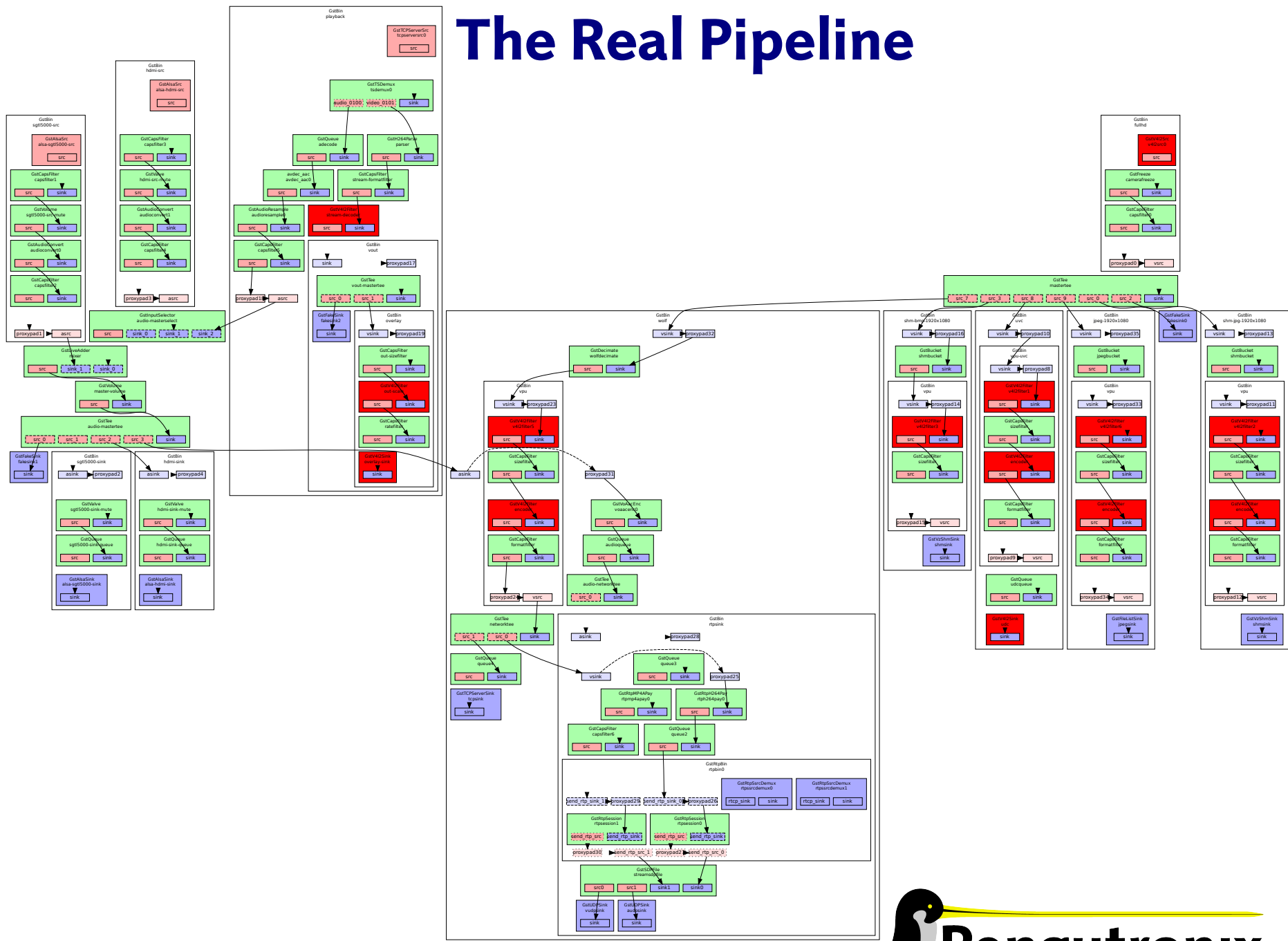
Dynamic Pipeline Changes

Dynamic Pipeline Changes

- Allocate new buffers as needed

    - V4L2 now has VIDIOC_CREATE_BUFS to create extra buffers

- Improve reconfiguring V4L2 elements

    - Ignore unchanged caps

    - Release buffers when stopping the pool

- Improved encoded buffer handling

- Improved property handling

- Correct V4L2 element shutdown

    ....

Pengutronix

The Real Pipeline

What's Next?

- Get things into upstream GStreamer

    - This requires some more cleanup

- Make it more robust

- Improve dynamic pipeline configuration

- New V4L2 APIs needed?

Pengutronix

Questions?

Contact:

m.olbrich@pengutronix.de

Code:

http://git-public.pengutronix.de/git-public/gst-plugins-good.git

Pengutronix

Thank You!

Pengutronix.