



COLLABORA

# From GL to your Encoder

## Zero-Copy Made Possible

**@ndufresne**

**Nicolas Dufresne**

[nicolas@collabora.com](mailto:nicolas@collabora.com)

**Open First**





COLLABORA

# Who am I ?

# Embedded Open Source Linux Developer

- Develops **CODECs** drivers and API in **Linux Media**
- All around contributor to **GStreamer** (with a Linux focus)
- Helps with the GStreamer project **maintenance**
- A month from **15<sup>th</sup>** year as Consultant at **Collabora**





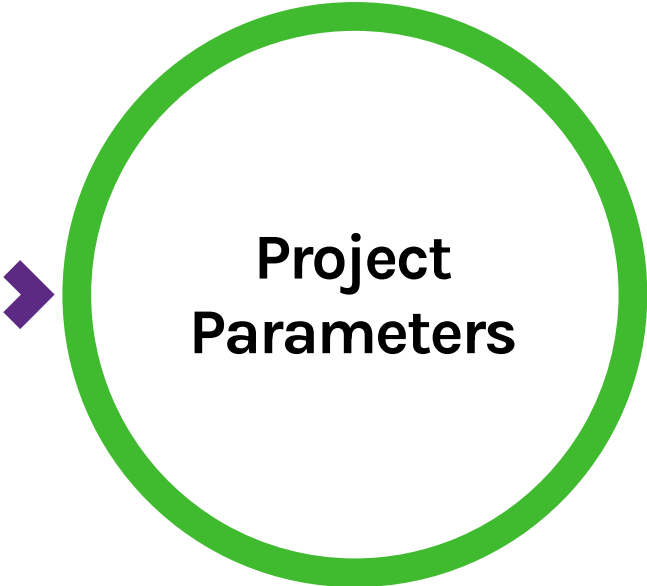
COLLABORA

# A Journey as Embedded OSS Engineering Consultant



COLLABORA

# It all started with a challenge



## Project Parameters

- NXP i.MX6 quad plus SoC
- 1280x800 Touchscreen Display
- Running QT6 Application
- Running mainline Linux and GStreamer





## Project Goal

“Remote render a running QT6 application using the RDP protocol using H.264 compression.”





## Initial Plan

Use GStreamer to capture and compress,  
FreeRDP to transmit.

```
"qml6glsrc -> v4l2h264enc -> appsink/FreeRDP"
```





## Customer Statement

“We managed to get **something** running, but only with **software color conversion** and software **encoder**, we can live with lower framerate, but this is too slow.”





COLLABORA

# Plan A

# The GStreamer Pipeline

```
qml6glsrc
```

Using EGLFS, so no known alternatives

QML renders upside down

```
! glvideoflip video-direction=auto
```

To produce NV12 for the encoder

```
! glcolorconvert
```

```
! gldownload
```

MR 6792: import DMABufs from downstream pool

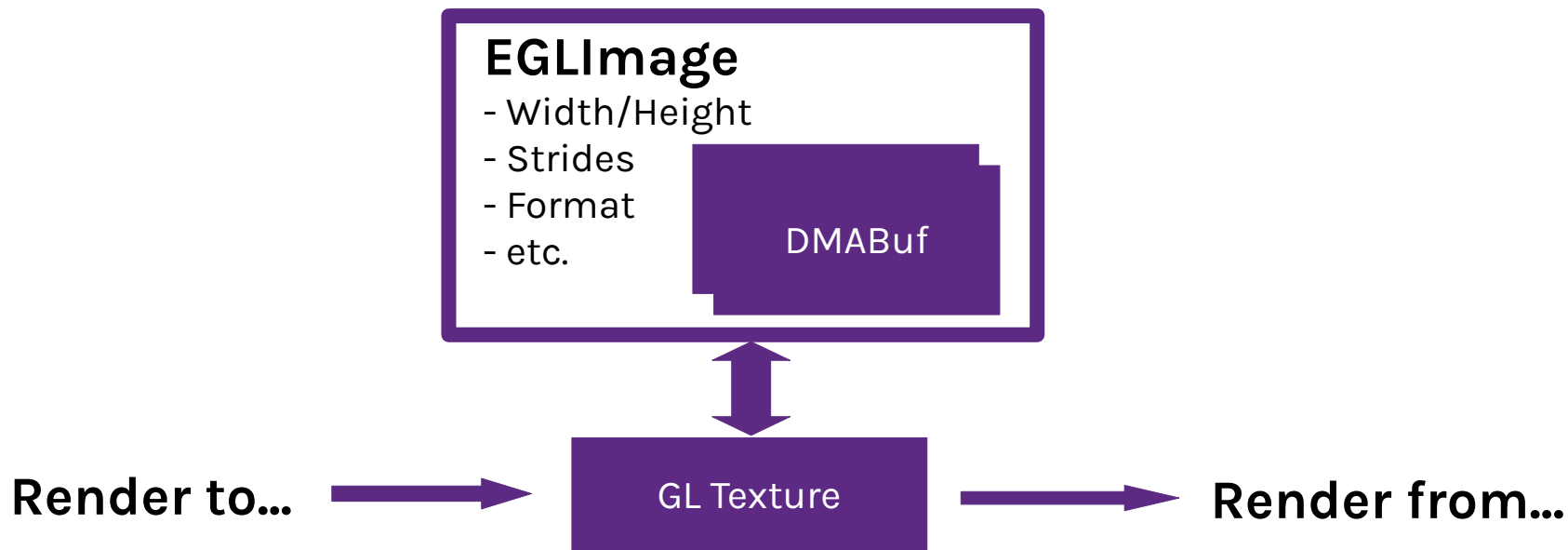
```
! v4l2h264enc
```

Add DMA\_DRM support to be compatible with MR 6792

```
! appsink
```



# Linux DMABuf in OpenGL





## Needed API

- `EGL_EXT_image_dma_buf_import_modifiers`
- `EglQueryDmaBufModifiersEXT()` (2019)
- `external_only`
- `DMA_DRM` negotiation with `memory:DMABuf` caps feature





**We have a plan, but will it work?**



## Setup

- Prepare a Yocto tree and hook it to CI
- Configure the Linux Kernel
- Prepare GStreamer Patches





## Prototyping Stage 1

- Fork and edit the qml6src example
- Test the part that are suppose to work already.
- Test MR 6792 on target  
gltestsrc ! gldownload ! kmssink
- Test use case without MR 6792,  
**should work** slowly at least.







**First Set of  
Issues**

- qmgl6src **leaks**
- qmgl6src has 2-3 frames of **delay**
- qmgl6src only works on **wayland**
- v4l2h264enc needs a **framerate**
- glcolorconvert **fails** to produce **NV12**
- **Memory bandwidth** is maxed





**Bandwidth**

qml6glsrc

Copies the GL Back Buffer

! glvideoflip

Render Pass

! glColorconvert

Render Pass

! gldownload

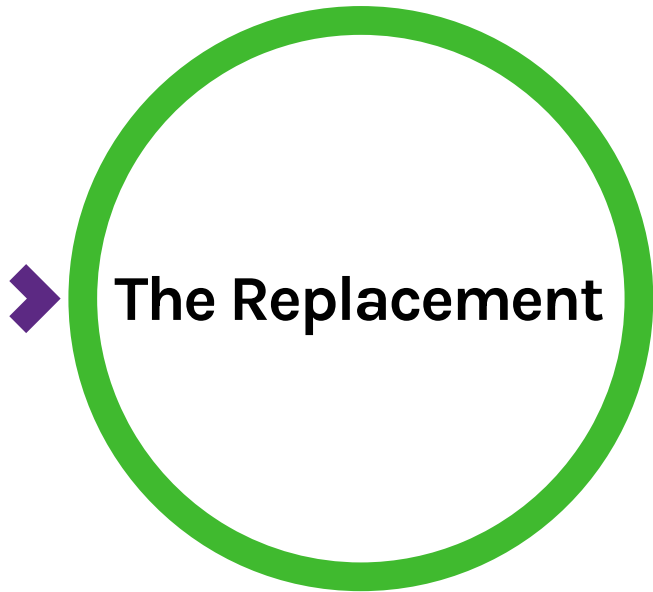
! videoconvert

Needs a replacement

! v4l2h264enc

! appsink





## The Replacement

- NXP IPUv3 Scaler and **Converter**
- Available through **v4l2convert**
- Does **vertical flips** in one pass
- Its a **staging driver** of average quality



# Plan B

# The Final GStreamer Pipeline

```
qm16g1src  
    ! gldownload  
    ! v4l2convert  
    ! v4l2h264enc  
    ! appsink
```

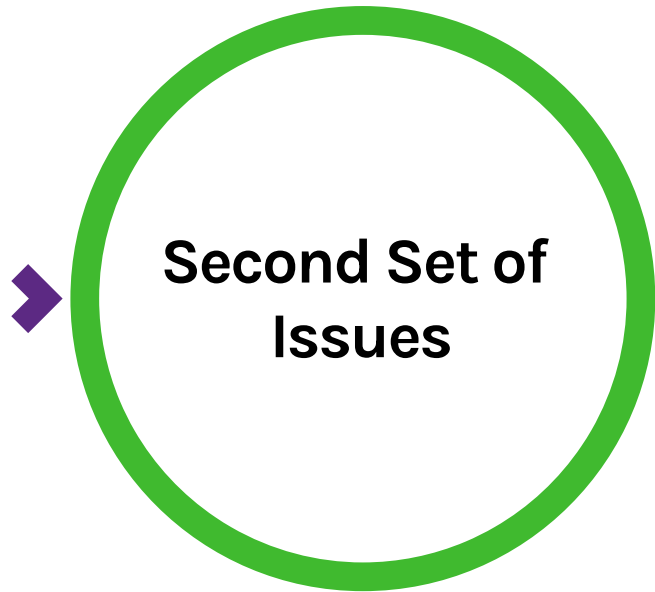




## Prototyping Stage 2

- Try the new pipeline without MR6792
- Test MR6792 with `qml6glsrc`  
`qml6glsrc ! gldownload ! kmssink`





## Second Set of Issues

- qm1gl6src dit not use downstream pool, **breaking zero-copy** with MR6792
- IPUv3 converter is missing **RGBA** pixel format
- IPUv3 fails to apply wanted **colorspace**
- Overall **performance** seems better even without zero-copy



COLLABORA

# The implementation





**Work!**

- !7286 qt6: glwindow: Don't leak previously rendered buffer
- !7290 qmlgl6src: Fix crash when use-default-fbo is not set
- !7321 qt6glwindow: Fallback to GL\_RGB on CopyTexImage2D error
- !7351 qml6glsrc: Reduce capture **delay**
- !7352 v4l2: encoder: Add **dynamic framerate** support
- !7540 v4l2: Various fixes and improvement
- Loads of **V4L2 backports** fixing useful stuff (also landed in 1.24)
- Draft of **DMA\_DRM** in **video4linux2** plugin (MR soon)
- **Imx-csc-scaler** Linux driver support for RGBA (coming soon)
- **Imx-csc-scaler** Linux driver support for colorspace (coming soon)
- One gsteglimage patch remaining for EGLFS (needs more work)





COLLABORA

# The results !



**Thank you!**



COLLABORA

**Open First**