libcamerasrc:
Introduction and usage of libcamera’s Gstreamer element

Gstreamer Conference 2023
A Coruña, Spain

Umang Jain
umang.jain@ideasonboard.com

Hello!

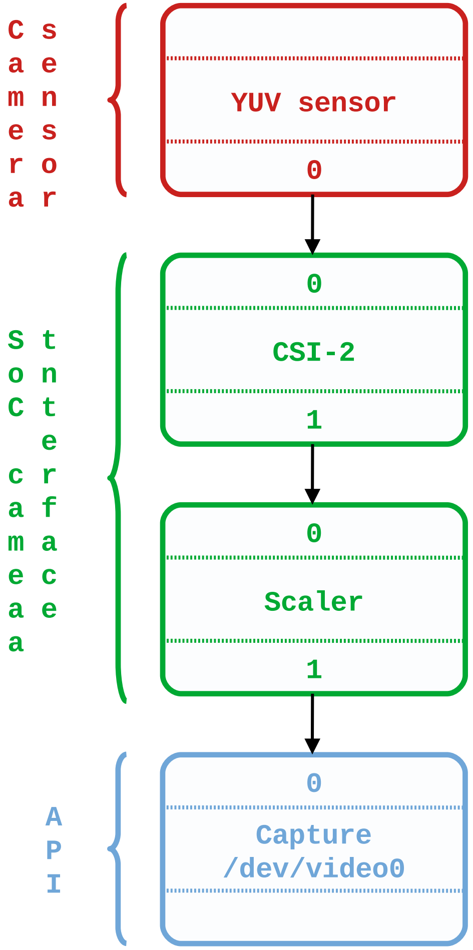
And welcome...

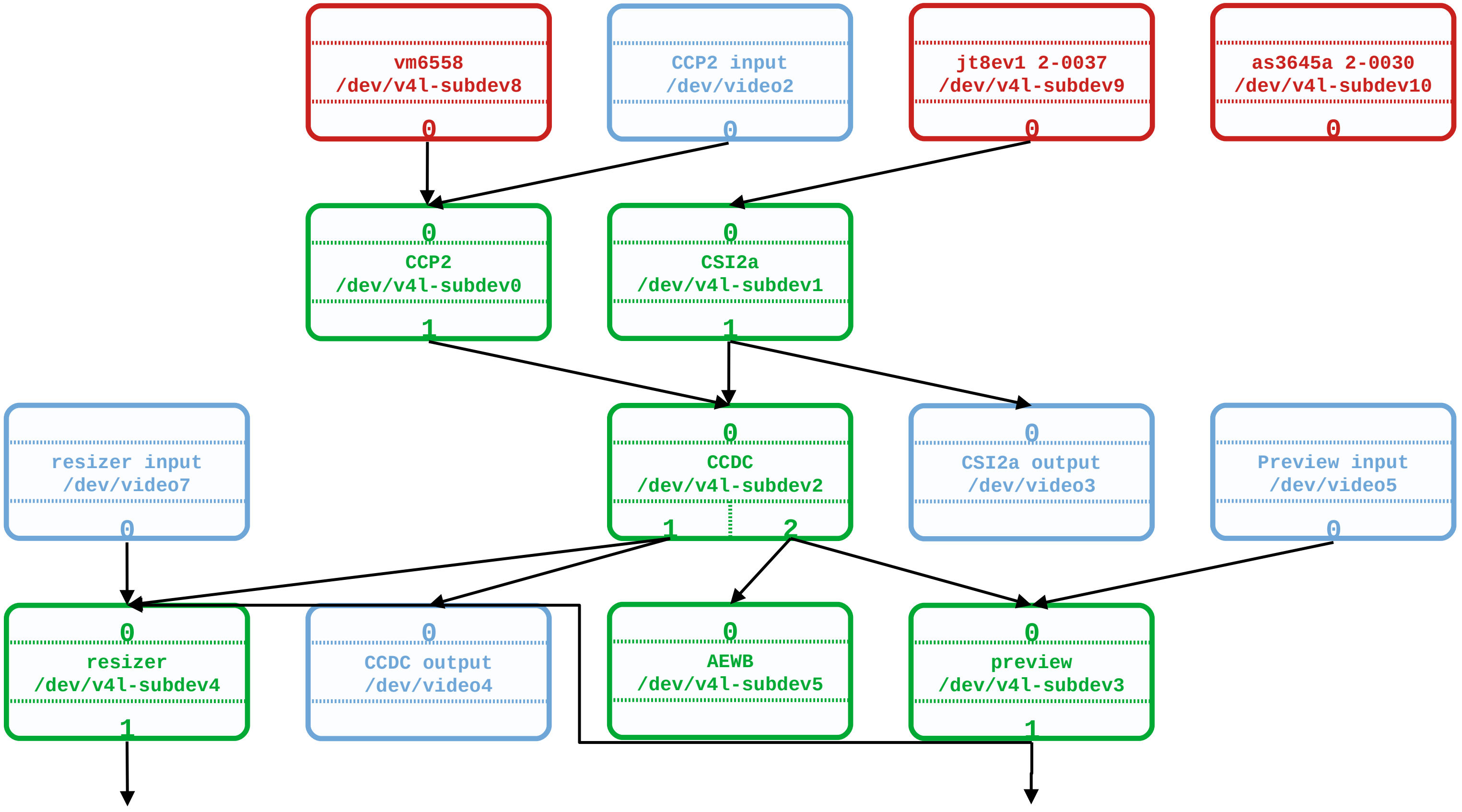
I’m Umang Jain
- Desktop turned Embedded developer
- Contributor to GNOME, OSTree, flatpak
- Mostly embedded, mainline linux and libcamera these days...

```
   _
+-/ \-+
| (o) |
+-----+
```

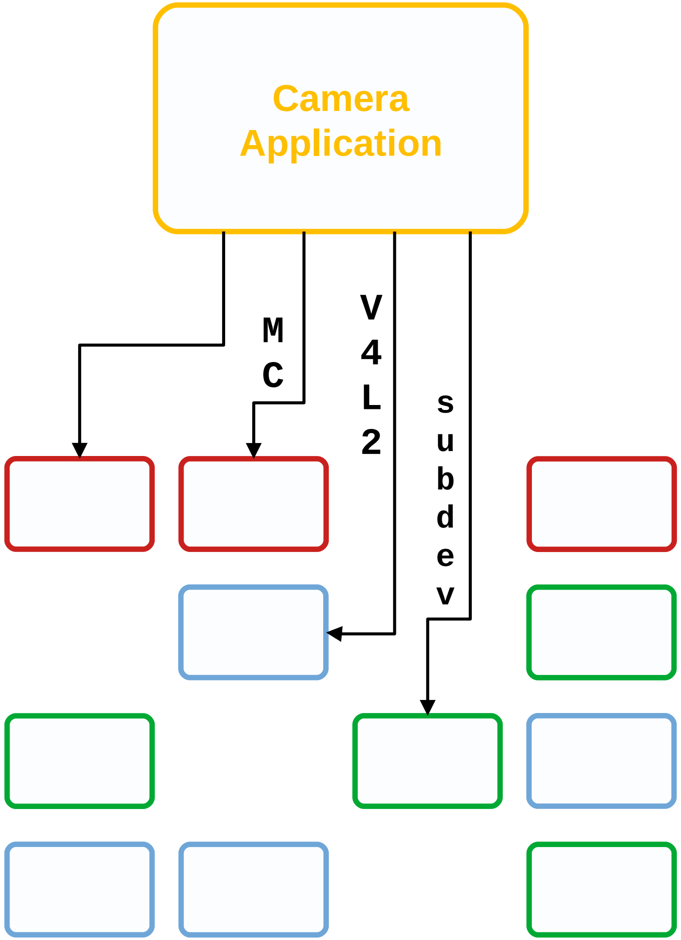
- **libcamera quick-intro**
- **libcamerasrc**
- **Camera capability discovery**
- **Controls handling**
- **Inter-operability**
- **Q+A**

Cameras are Complex

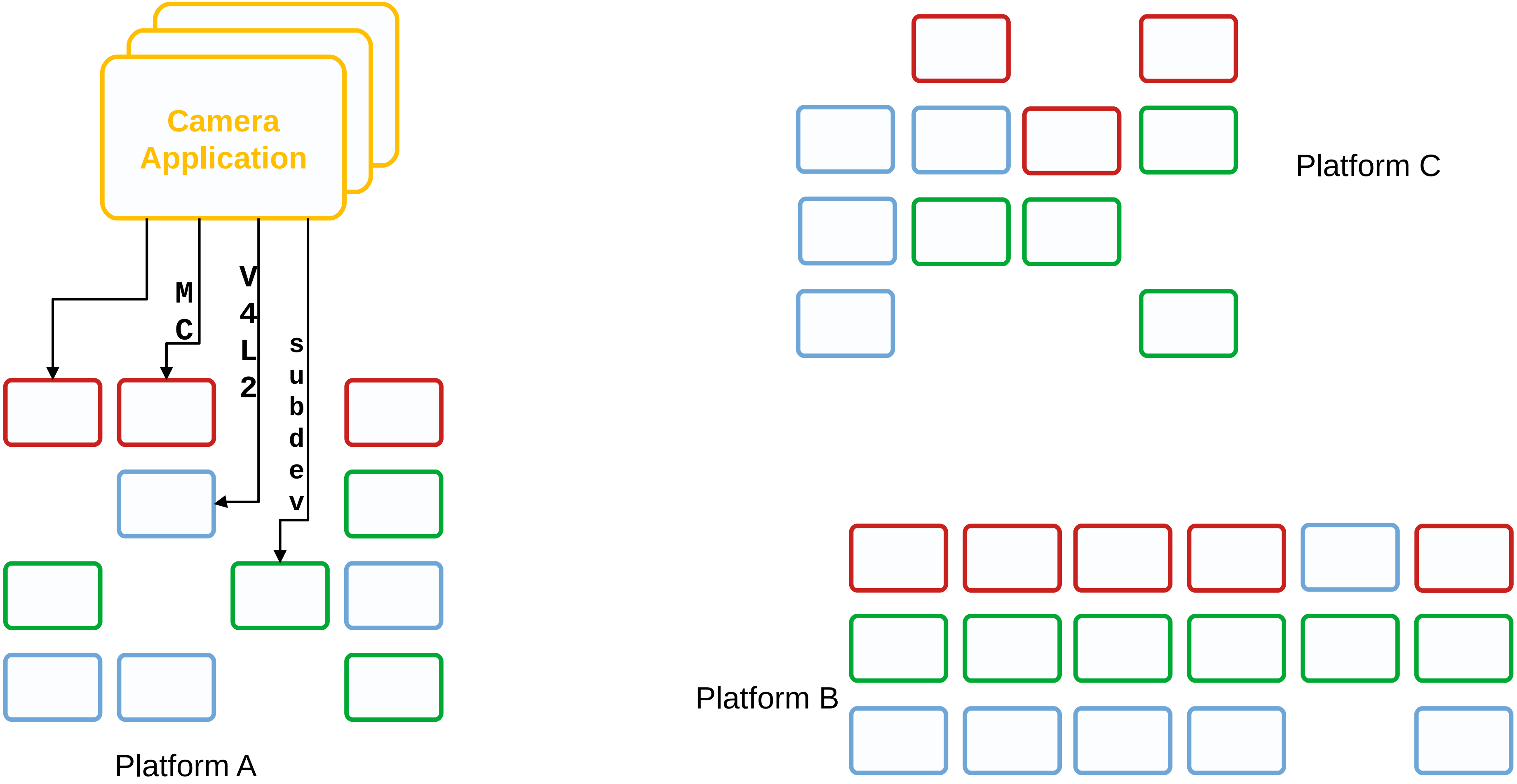




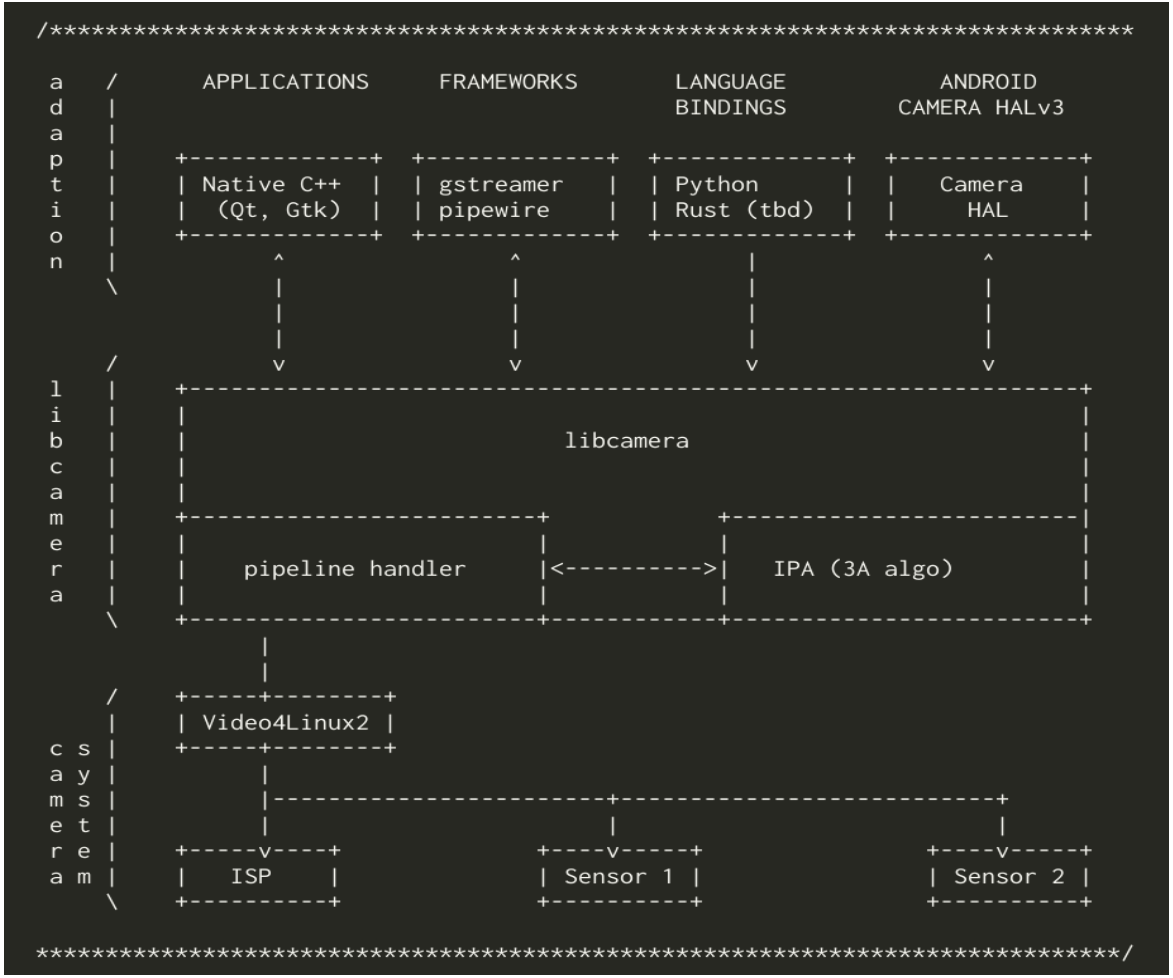
OMAP3 Camera in Nokia N900 - 2009



Applications can manage those complexities …



But it doesn’t scale

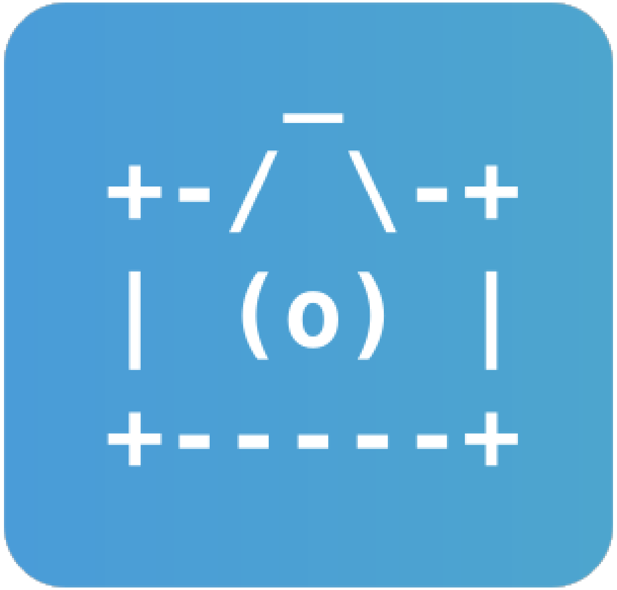


```
/******************************************************************************

 a   /      APPLICATIONS     FRAMEWORKS       LANGUAGE           ANDROID
 d   |                                        BINDINGS        CAMERA HALv3
 a   |
 p   |    +-------------+  +-------------+  +-------------+  +-------------+
 t   |    | Native C++  |  | gstreamer   |  | Python      |  |   Camera    |
 i   |    |  (Qt, Gtk)  |  | pipewire    |  | Rust (tbd)  |  |     HAL     |
 o   |    +-------------+  +-------------+  +-------------+  +-------------+
 n   |           ^                ^                |                ^
     \           |                |                |                |
                 |                |                |                |
                 |                |                |                |
     /           v                v                v                v
 l   |    +----------------------------------------------------------------+
 i   |    |                                                                |
 b   |    |                           libcamera                            |
 c   |    |                                                                |
 a   |    |                                                                |
 m   |    +-------------------------+            +-------------------------|
 e   |    |                         |            |                         |
 r   |    |    pipeline handler     |<---------->|   IPA (3A algo)         |
 a   |    |                         |            |                         |
     \    +-------------------------+------------+-------------------------+
                |
                |
     /    +-----+--------+
     |    | Video4Linux2 |
 c s |    +-----+--------+
 a y |          |
 m s |          |------------------------+---------------------------+
 e t |          |                        |                           |
 r e |    +-----v----+              +----v-----+                +----v-----+
 a m |    |   ISP    |              | Sensor 1 |                | Sensor 2 |
     \    +----------+              +----------+                +----------+

******************************************************************************/
```

libcamera fills that gap

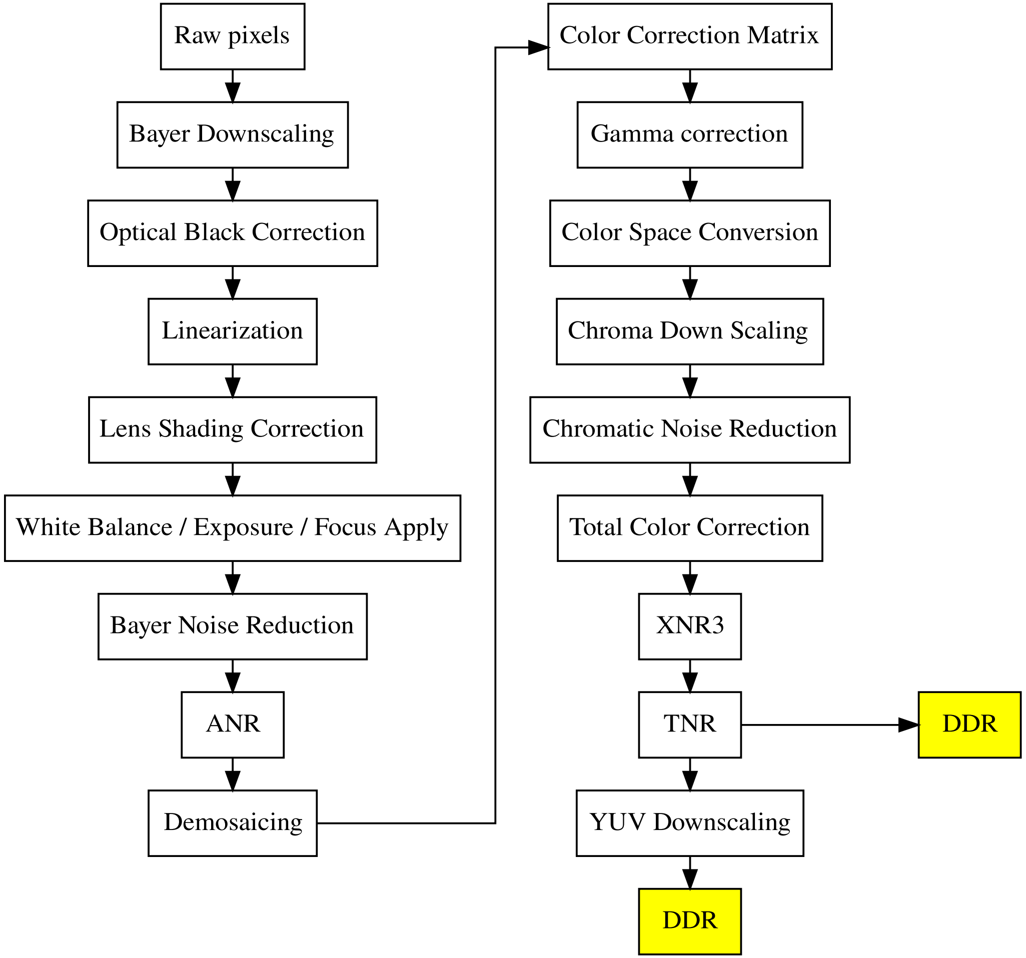


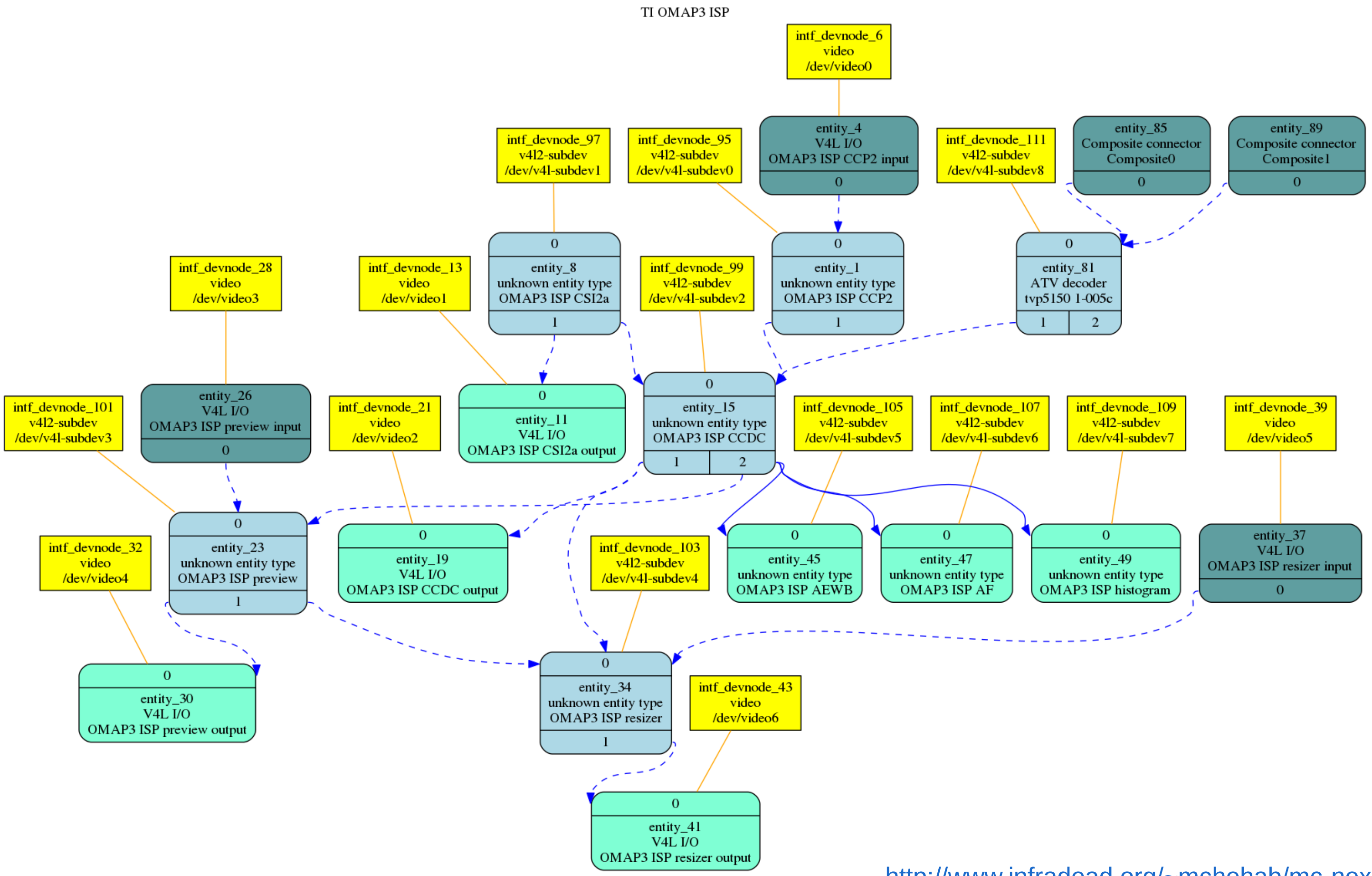
Complex Camera Workshop
Jun 2018, Tokyo

https://www.linuxtv.org/news.php?entry=2018-06-04-3.mchehab

libcamera fills that gap

IPU3 ImgU block diagram




TI OMAP3 ISP
intf_devnode_6 video /dev/video0
entity_4 V4L I/O OMAP3 ISP CCP2 input 0
intf_devnode_97 v4l2-subdev /dev/v4l-subdev1
intf_devnode_95 v4l2-subdev /dev/v4l-subdev0
intf_devnode_111 v4l2-subdev /dev/v4l-subdev8
entity_85 Composite connector Composite0 0
entity_89 Composite connector Composite1 0
0 entity_8 unknown entity type OMAP3 ISP CSI2a 1
0 entity_1 unknown entity type OMAP3 ISP CCP2 1
0 entity_81 ATV decoder tvp5150 1-005c 1 2
intf_devnode_28 video /dev/video3
intf_devnode_13 video /dev/video1
intf_devnode_99 v4l2-subdev /dev/v4l-subdev2
entity_26 V4L I/O OMAP3 ISP preview input 0
0 entity_11 V4L I/O OMAP3 ISP CSI2a output
0 entity_15 unknown entity type OMAP3 ISP CCDC 1 2
intf_devnode_101 v4l2-subdev /dev/v4l-subdev3
intf_devnode_21 video /dev/video2
intf_devnode_105 v4l2-subdev /dev/v4l-subdev5
intf_devnode_107 v4l2-subdev /dev/v4l-subdev6
intf_devnode_109 v4l2-subdev /dev/v4l-subdev7
intf_devnode_39 video /dev/video5
0 entity_23 unknown entity type OMAP3 ISP preview 1
0 entity_19 V4L I/O OMAP3 ISP CCDC output
0 entity_45 unknown entity type OMAP3 ISP AEWB
0 entity_47 unknown entity type OMAP3 ISP AF
0 entity_49 unknown entity type OMAP3 ISP histogram
entity_37 V4L I/O OMAP3 ISP resizer input 0
intf_devnode_32 video /dev/video4
intf_devnode_103 v4l2-subdev /dev/v4l-subdev4
0 entity_30 V4L I/O OMAP3 ISP preview output
0 entity_34 unknown entity type OMAP3 ISP resizer 1
intf_devnode_43 video /dev/video6
0 entity_41 V4L I/O OMAP3 ISP resizer output

http://www.infradead.org/~mchehab/mc-next-gen/omap3-igepv2-with-tvp5150.png

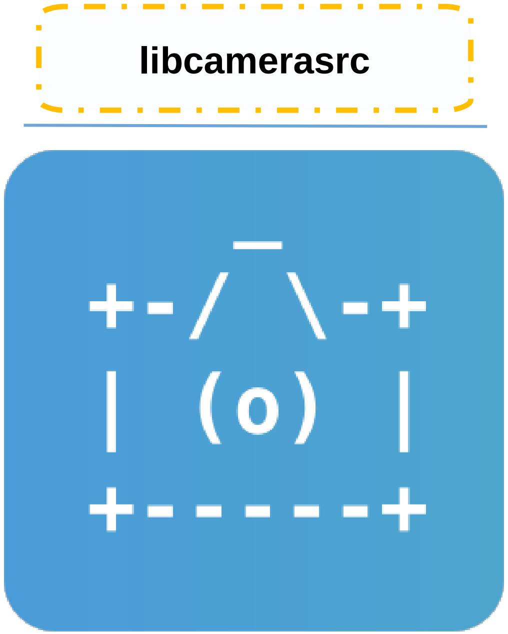
TI OMAP3 ISP

```
   _
+-/ \-+
| (o) |
+-----+
```

- libcamera quick-intro
- **libcamerasrc**
- Camera capability discovery
- Controls handling
- Inter-operability
- Q+A

libcamerasrc

- libcamera based video capture source element for gstreamer framework
- Uses the libcamera public API but currently lives in libcamera tree https://git.libcamera.org/libcamera/libcamera.git/tree/src/gstreamer



```
    _
+-/ \-+
| (o) |
+-----+
```

- libcamera quick-intro
- libcamerasrc
- **Camera Capability discovery**
- Controls handling
- Inter-operability
- Q+A

libcamera exposes the formats and sizes available for configuration

-
  ```
  Pixelformat: NV21 (64x64)-(4656x3496)/(+2,+2)
  160x120
  240x160
  320x240
  ….
  3840x2400
  ```

-
  ```
  Pixelformat: YUV420 (64x64)-(4656x3496)/(+2,+2)
  160x120
  240x160
  320x240
  …
  3840x2400
  ```

- Many more…

Camera capability discovery

Camera capability discovery

- Exposed formats with size-ranges are converted to libcamerasrc’s “filter” caps for its source pad (can also be probed by gst-device-monitor)

-
  ```
  gst-launch-1.0
    libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
    ! video/x-raw,width=1920,height=1080 ! fakesink
  ```

Peer caps suggests the capabilities intended to be used by libcamerasrc

- Query the downstream peer caps against the filter caps of srcpad
- `caps = gst_pad_peer_query_caps(srcpad, filter);`

```
gst-launch-1.0
  libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
  ! video/x-raw,width=1920,height=1080 ! fakesink
```

Camera capability discovery

Peer caps suggests the capabilities intended to be used by libcamerasrc

- Query the downstream peer caps against the filter caps of srcpad
- `caps = gst_pad_peer_query_caps(srcpad, filter);`

```
gst-launch-1.0
  libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
  ! video/x-raw,width=1920,height=1080 ! fakesink
```

- libcamerasrc will configure the camera according to the queried caps

Camera capability discovery

If queried caps are subset of filter, one achieves fully functioning pipeline

-
  ```
  gst-launch-1.0
    libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
    ! video/x-raw,width=1920,height=1080 ! fakesink
  ```

Camera capability discovery

But wait! what about framerate or colorimetry

-
  ```
  gst-launch-1.0
    libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
    ! video/x-raw,format=NV12,width=1920,height=1080,framerate=30/1
    ! fakesink
  ```

Camera capability discovery

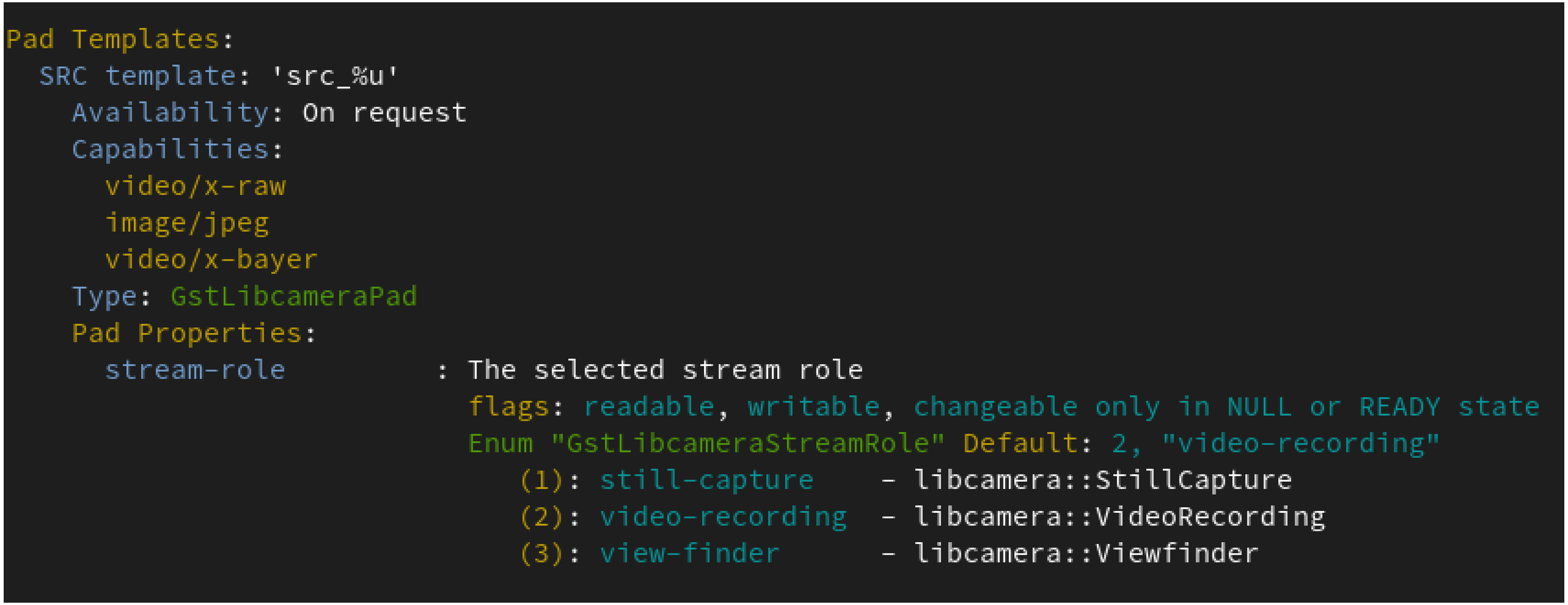
Framerate and colorimetry

```
gst-launch-1.0
  libcamerasrc camera-name="/base/soc/i2c0mux/i2c@1/imx519@1a"
  ! video/x-raw,format=NV12,width=1920,height=1080,framerate=30/1
  ! fakesink
```

- framerate deviates from strict caps query
- Validated after configuration and clamped, if out of bounds
- Colorimetry also might be adjusted internally by libcamera according to stream configuration which might fail pipeline negotiation

libcamera supports multistream cameras

- Request pads for more than one stream



```
Pad Templates:
  SRC template: 'src_%u'
    Availability: On request
    Capabilities:
      video/x-raw
      image/jpeg
      video/x-bayer
    Type: GstLibcameraPad
    Pad Properties:
      stream-role         : The selected stream role
                            flags: readable, writable, changeable only in NULL or READY state
                            Enum "GstLibcameraStreamRole" Default: 2, "video-recording"
                               (1): still-capture    - libcamera::StillCapture
                               (2): video-recording  - libcamera::VideoRecording
                               (3): view-finder      - libcamera::Viewfinder
```

libcamera supports multistream cameras

-
  ```
  gst-launch-1.0 libcamerasrc camera-name="cam" name=src src.src ! \
  queue ! videoconvert ! autovideosink src.src_0 ! Queue ! \
  videoconvert ! autovideosink
  ```

```
   _
+-/ \-+
| (o) |
+-----+
```

- libcamera quick-intro
- Libcamerasrc
- Camera Capability discovery
- **Controls handling**
- Inter-operability
- Q+A

Controls support (Work-in-progress)

```
gst-launch-1.0 libcamerasrc auto-focus-mode=on \
  ! 'video/x-raw,format=RGB,framerate=30/1' ! queue ! kmssink

gst-launch-1.0 libcamerasrc contrast=9 \
    ! 'video/x-raw,format=RGB,framerate=30/1' ! queue ! kmssink

gst-launch-1.0 libcamerasrc saturation=5 \
   ! 'video/x-raw,format=RGB,framerate=30/1' ! queue ! kmssink
```

Controls handling

Controls support (Work-in-progress)

- High Priority and under discussion
- WIP plumbing for gluing controls support by Nicolas Dufresne [dev branch]
- There are dedicated efforts from libcamera, gstreamer, and pipewire all working together here

libcamera get-together/hackfest after Kernel-Recipes Sept 30, Paris

Controls handling

Gaps in controls handling

- libcamera only exposes controls supported on the camera whereas the gstreamer controls are expected to be element properties

  ```
  gst-launch-1.0 libcamerasrc auto-focus-mode=on \
     ! 'video/x-raw,format=RGB,framerate=30/1' ! queue ! kmssink
  ```

  What if the camera does not have any autofocus capabilites…?!

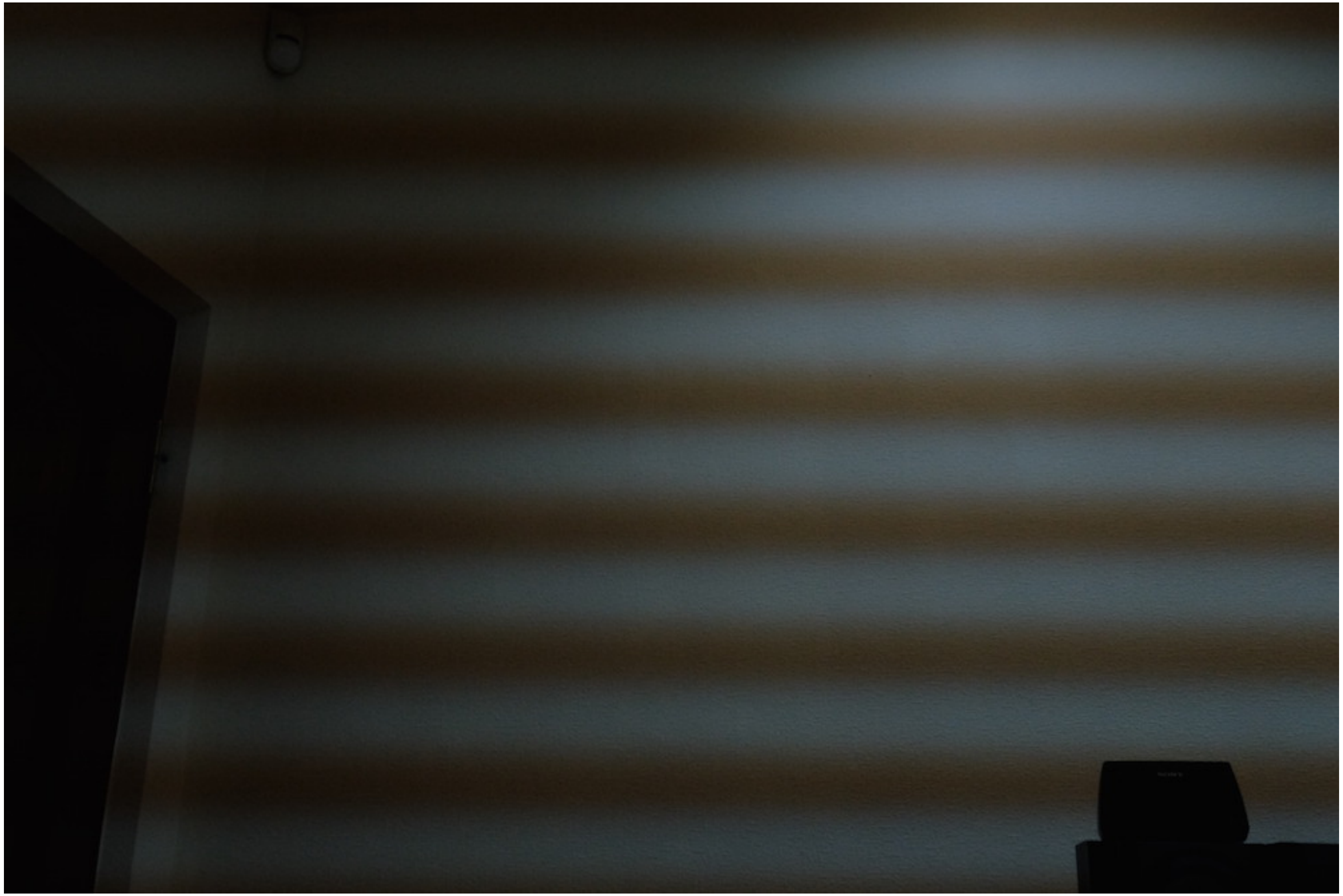
- Control binding and/or restrictions

Controls in libcamera

- Controls the camera and also frame metadata reporting
- YAML definition - `$libcamera/src/libcamera/control_ids.yaml`
- Case of flicker controls

Case of flicker avoidance

- Detect it
- Manually set it



Controls handling

Per frame controls

> *“The library shall support controlling capture parameters for each stream on a per-frame basis, on a best effort basis based on the capabilities of the hardware and underlying software stack (including kernel drivers and firmware). It shall apply capture parameters to the frame they target, and report the value of the parameters that have effectively been used for each captured frame.”*

```
   _
+-/ \-+
| (o) |
+-----+
```

- libcamera quick-intro
- Libcamerasrc
- Camera Capability discovery
- Controls handling
- **Inter-operability**
- Q+A

JPEG Network Streamer

```
gst-launch-1.0 libcamerasrc !
  video/x-raw,colorimetry=bt709,format=NV12,width=1280,height=720 \
  framerate=30/1 ! \
  jpegenc ! multipartmux ! tcpserversink host=0.0.0.0 port=5000
```

JPEG Network Receiver

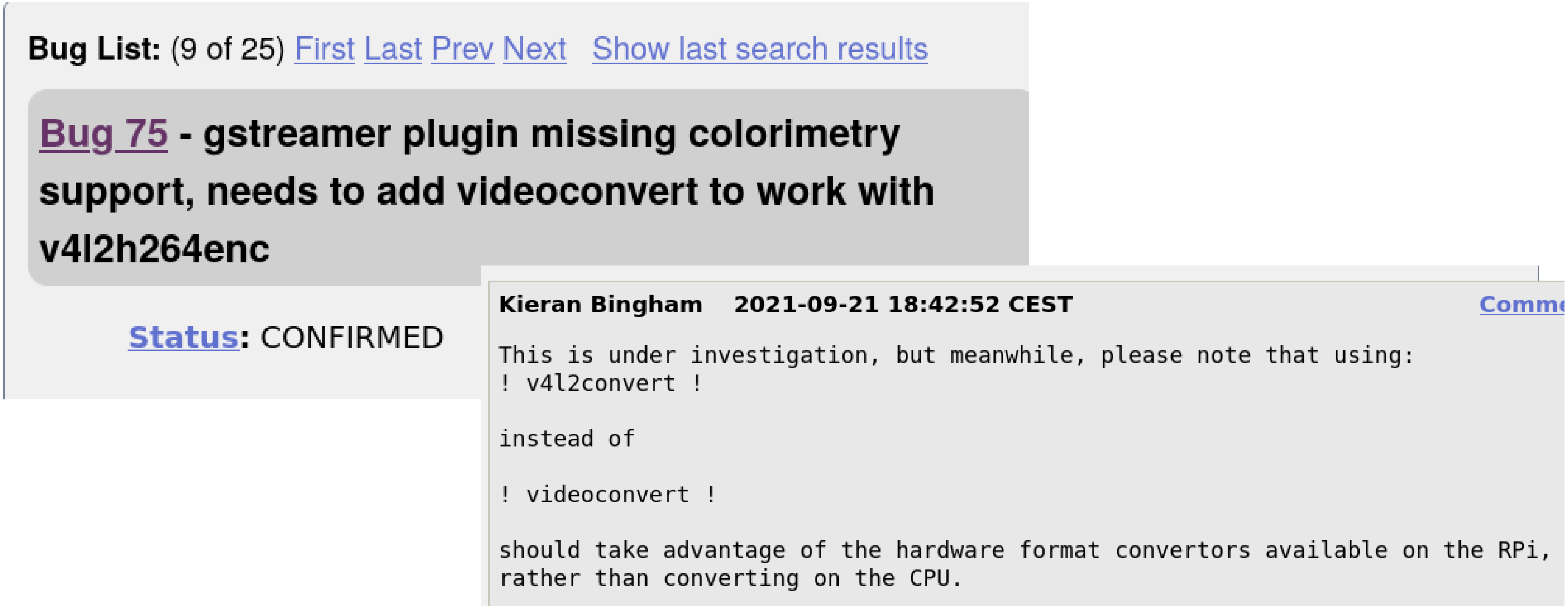
```
gst-launch-1.0 tcpclientsrc host=$DEVICE_IP port=5000 ! \
      multipartdemux ! jpegdec ! autovideosink
```

Inter-operability

With v4l2h264enc and matroskamux

```
gst-launch-1.0 -vvvv libcamerasrc ! \
    video/x-raw,width=1280,height=720,format=NV12,colorimetry=bt601,\
    framerate=30/1,interlace-mode=progressive \
    ! v4l2h264enc extra-controls="controls,repeat_sequence_header=1"\
    ! 'video/x-h264,level=(string)4' \
    ! h264parse ! matroskamux ! filesink location=foo.mkv
```

Encoders support improved overtime...



Bug List: (9 of 25) First Last Prev Next Show last search results

Bug 75 - gstreamer plugin missing colorimetry support, needs to add videoconvert to work with v4l2h264enc

Status: CONFIRMED

Kieran Bingham 2021-09-21 18:42:52 CEST Comme

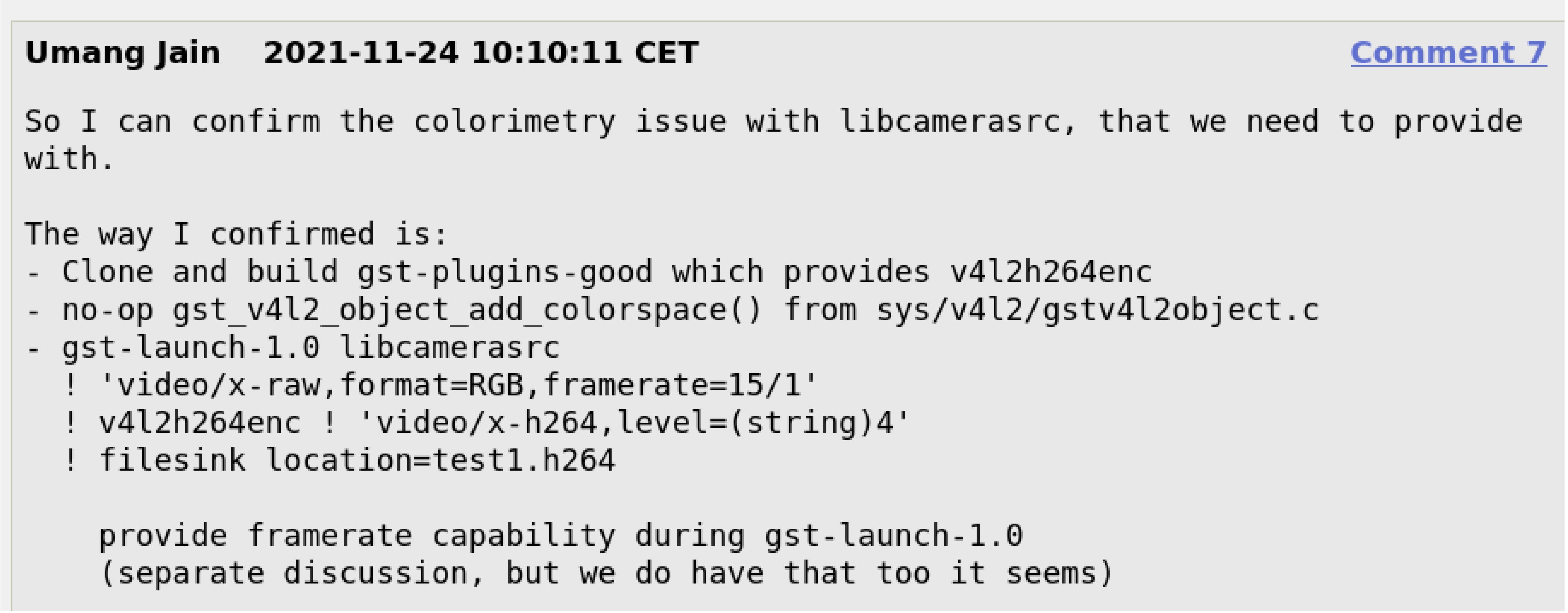
```
This is under investigation, but meanwhile, please note that using:
! v4l2convert !

instead of

! videoconvert !

should take advantage of the hardware format convertors available on the RPi,
rather than converting on the CPU.
```

Encoders support improved overtime...



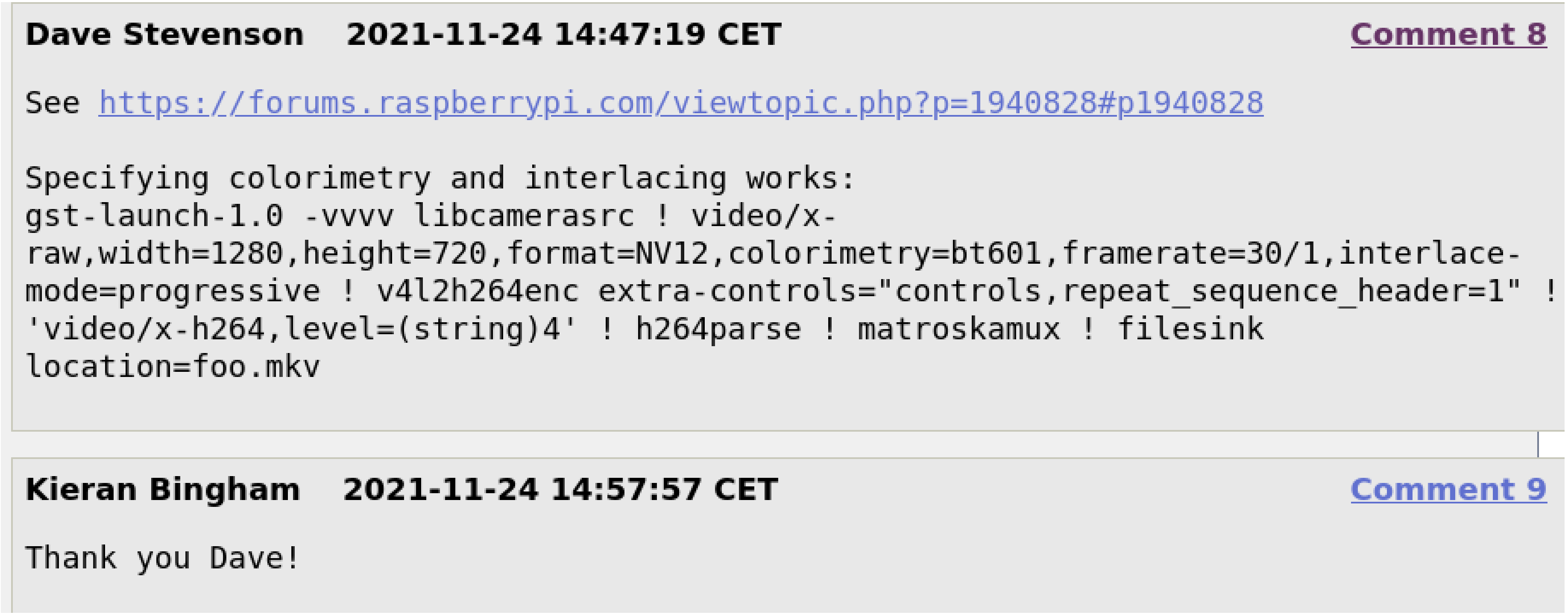
Umang Jain 2021-11-24 10:10:11 CET Comment 7

```
So I can confirm the colorimetry issue with libcamerasrc, that we need to provide
with.

The way I confirmed is:
- Clone and build gst-plugins-good which provides v4l2h264enc
- no-op gst_v4l2_object_add_colorspace() from sys/v4l2/gstv4l2object.c
- gst-launch-1.0 libcamerasrc
  ! 'video/x-raw,format=RGB,framerate=15/1'
  ! v4l2h264enc ! 'video/x-h264,level=(string)4'
  ! filesink location=test1.h264

    provide framerate capability during gst-launch-1.0
    (separate discussion, but we do have that too it seems)
```

Encoders support improved overtime...



Dave Stevenson 2021-11-24 14:47:19 CET Comment 8

See https://forums.raspberrypi.com/viewtopic.php?p=1940828#p1940828

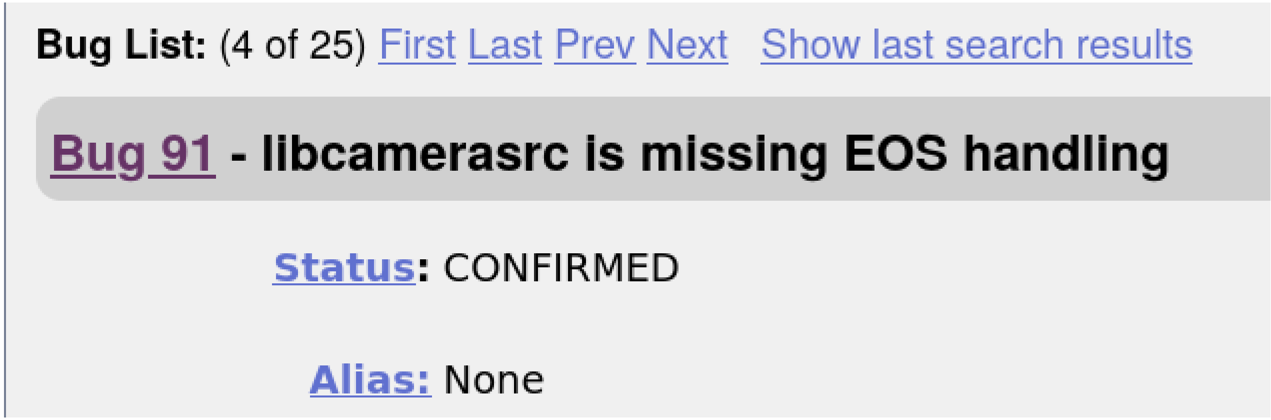
```
Specifying colorimetry and interlacing works:
gst-launch-1.0 -vvvv libcamerasrc ! video/x-
raw,width=1280,height=720,format=NV12,colorimetry=bt601,framerate=30/1,interlace-
mode=progressive ! v4l2h264enc extra-controls="controls,repeat_sequence_header=1" !
'video/x-h264,level=(string)4' ! h264parse ! matroskamux ! filesink
location=foo.mkv
```

Kieran Bingham 2021-11-24 14:57:57 CET Comment 9

Thank you Dave!

Inter-operability

There might still rough corners and inter-operability issues...



Bug List: (4 of 25) First Last Prev Next Show last search results

Bug 91 - libcamerasrc is missing EOS handling

Status: CONFIRMED

Alias: None

But with community support, it is bound to improve by testing more use cases...

Thank you!

```
    _
+-/ \-+
| (o) |
+-----+
```

libcamera

Suggestions/Feedback/Questions?

#libcamera on OFTC.net
https://webchat.oftc.net/?channels=libcamera
https://matrix.to/#/#_oftc_#libcamera:matrix.org