Kernel CAM

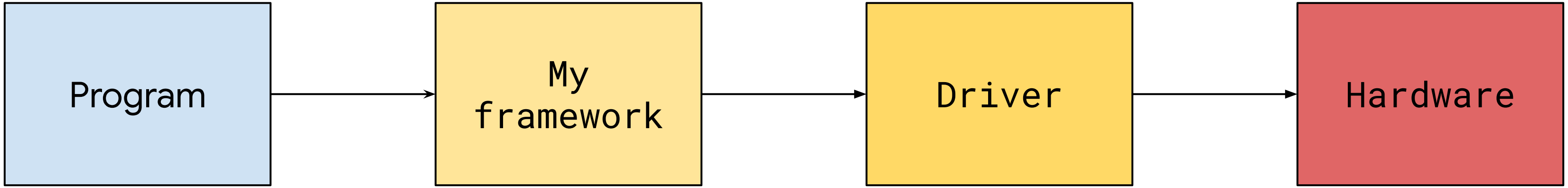
Rethinking the kernel camera framework

Sergey Senozhatsky, Fei Shao, Yunke Cao, Tomasz Figa, Ricardo Ribalda

Paris, 1st June 2022

Once upon a time

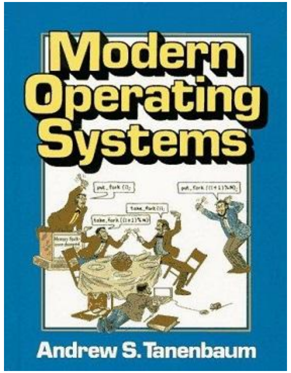
Operating system (for driver developers)



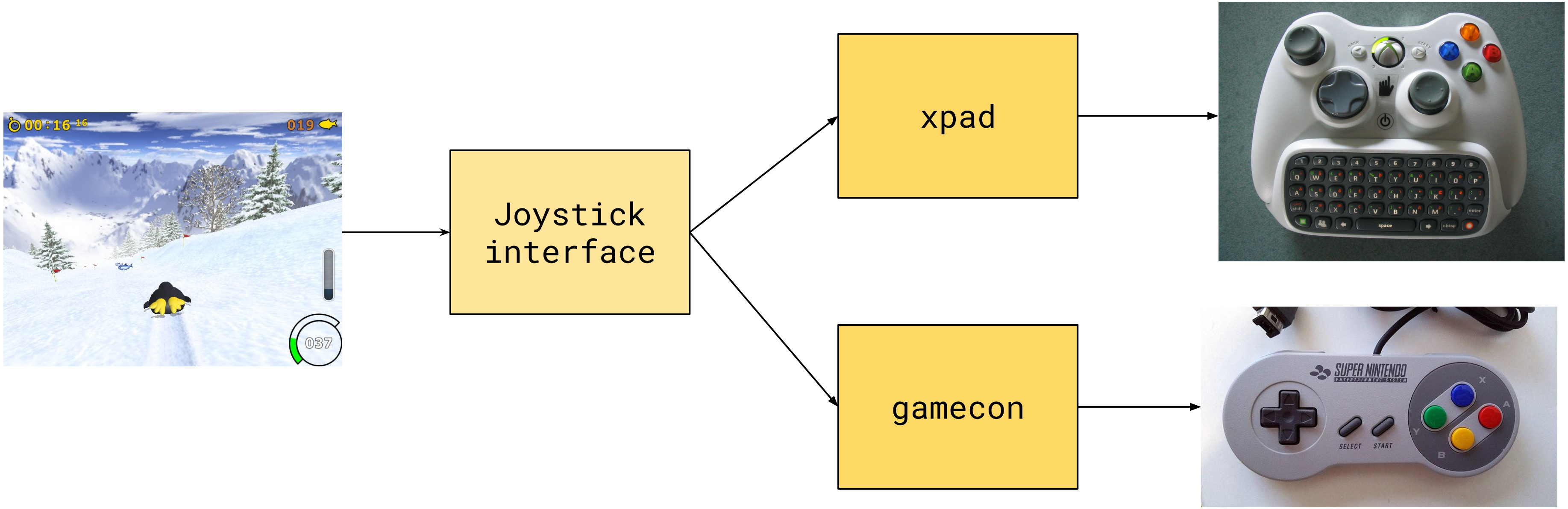

```
open(/dev/my_device)
read(fd)
ioctl(fd,IOC_,... )
```

```
my_init()
my_alloc()
my_read_from()
```

```
inb()
outb()
```



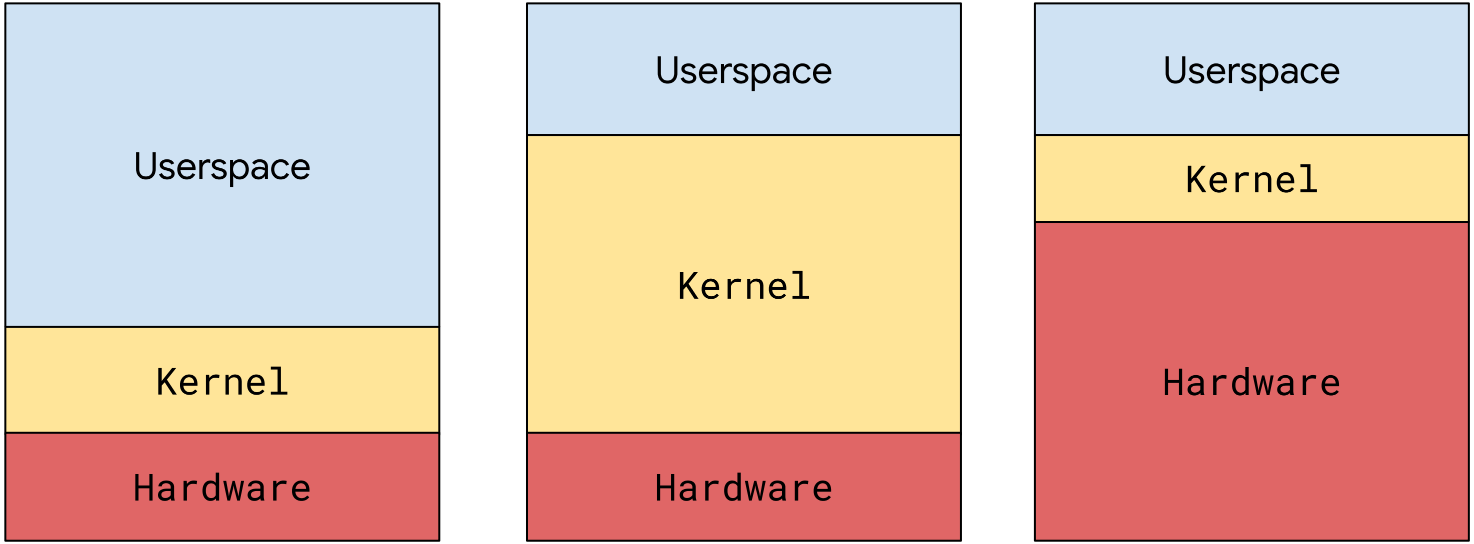
Operating system (for driver developers)



Cameras are special

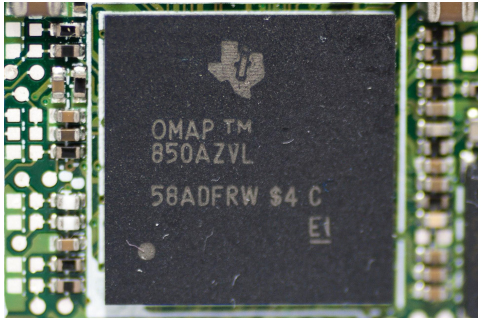
Great diversity of output

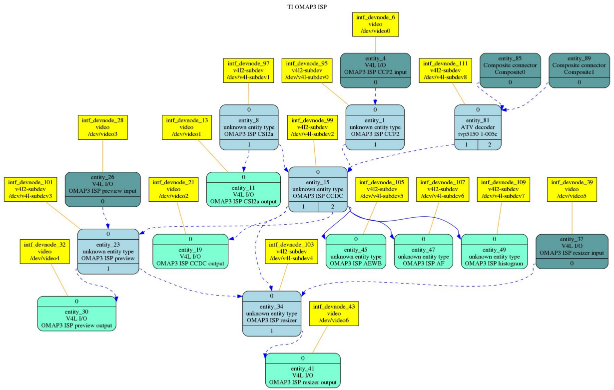
- There are more than 200 video formats
  - `grep -c "^#define V4L2_PIX_FMT" include/uapi/linux/videodev2.h ->` **202!**



Multiple ways to do the same thing

- Image is too dark?
  - Exposure time
  - Analog Gain
  - Digital Gain

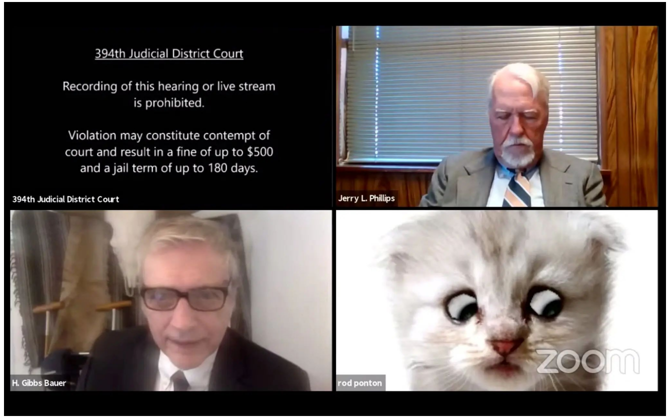


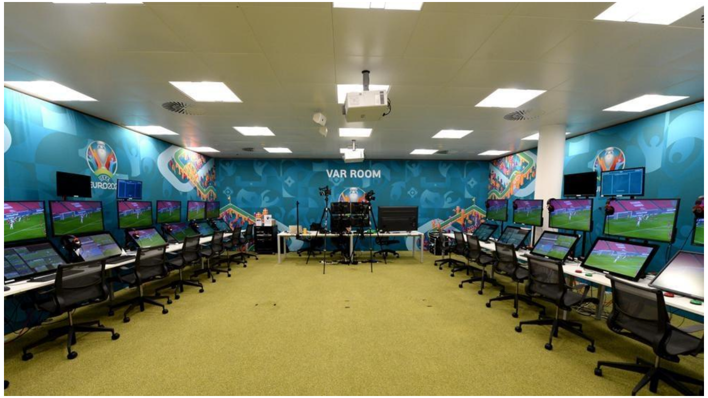


- Do you want a 640x480 image from a 24Mpix (6000*400)?:
  - Cropping
  - Binning
  - Interpolate

Do it fast or don't do it

- Big data rates
  - `1080p RGB32@60fps -> 474 MiB/sec!`
- Low latency
- Most of the time handled by other hardware
  - GPU
  - TPU
  - CODEC





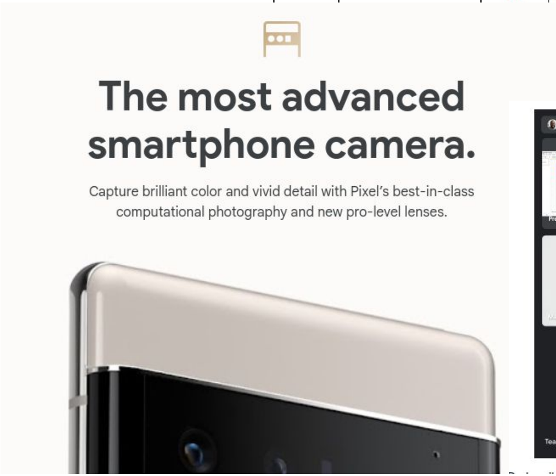
Cameras drive the consumer market

- Work from home/anywhere
- E-learning
- Deciding factor for purchase

International Journal of Managing Value and Supply Chains (IJMVSC) Vol.5, No. 2, June 2014

Table 5- Naming of Factors

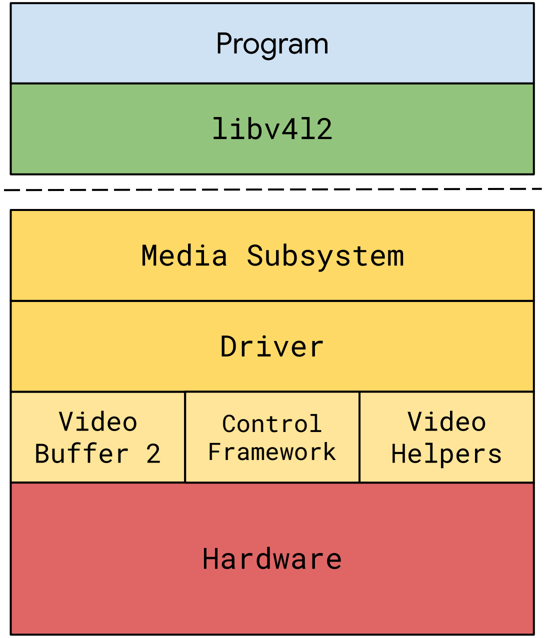
Factor no.	Name of dimension	Item no	variables	Factor loading
F1	**Physical attributes**	1	Camera and video	.827
		2	Bluetooth	.802
		3	Multimedia option	.800
		4	Touch screen	.775
		5	Memory capacity	.772
		6	Color display	.763
		7	Attractive color	.753
		8	Model/style	.684
		9	New features	.684
		10	Design of the phone	.669
		11	Appearance	.608
			Web browser	.597
			Brand value/quality	.504



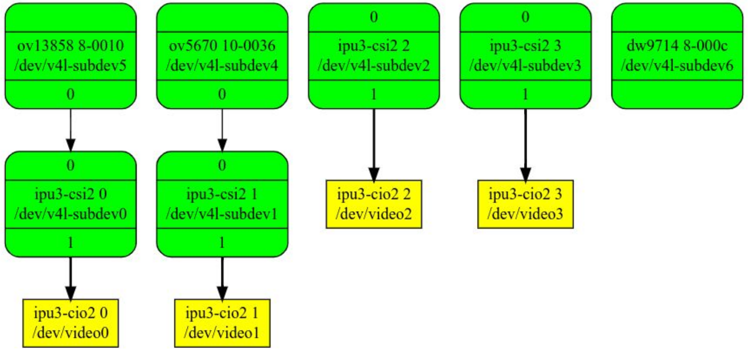
Cameras in Linux today

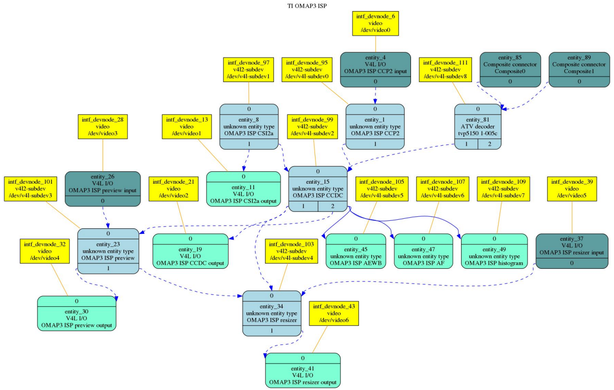
Video4Linux 2 (non media controller)

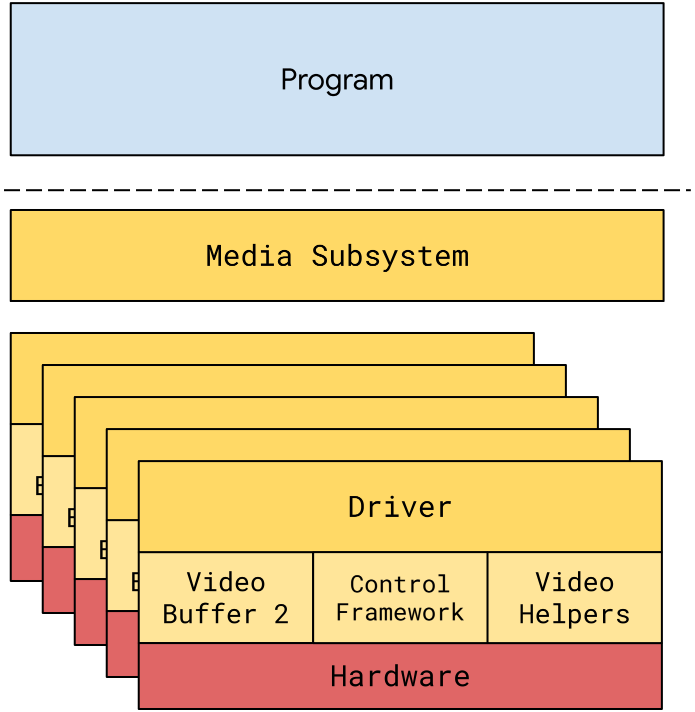
- Output can be converted via software to 4 standard formats.
- The driver makes most of the decisions for us.
- Hardware produces decent images by default.



Video4Linux 2 (media controller)

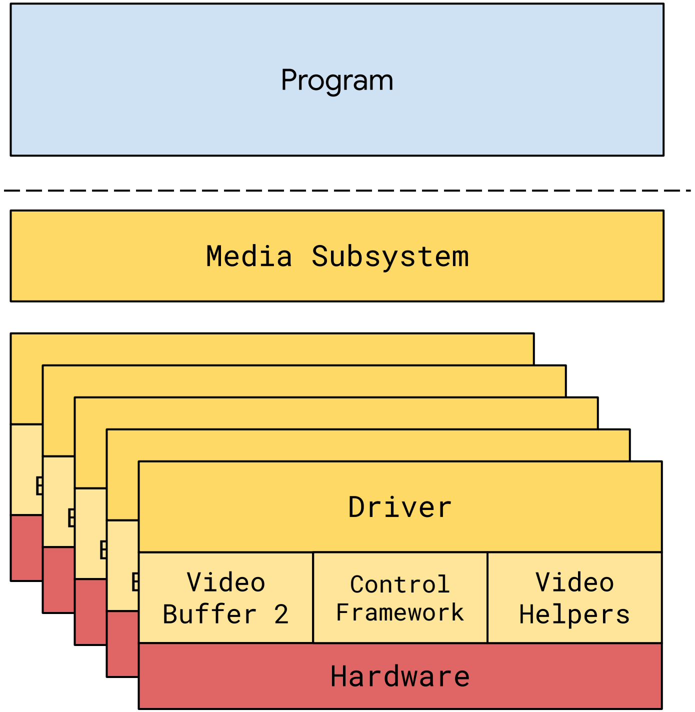






Video4Linux 2 (post media controller)

- Software stack needs to know about the internals of the hardware.
- All the decisions are taken by the software stack.
- A lot of Image tuning before we can use the images.



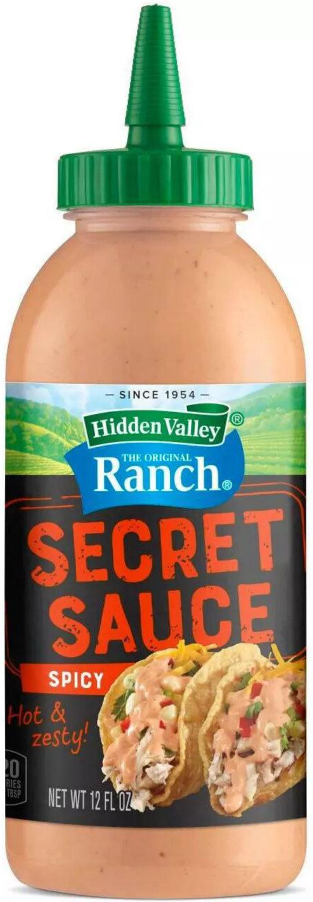
Libcamera

```
-------------------< libcamera Public API >---------------------------
          ^                                      ^
          |                                      |
          v                                      v
  +-------------+  +-------------------------------------------------+
  |   Camera    |  |  Camera Device                                  |
  |   Devices   |  | +---------------------------------------------+ |
  |   Manager   |  | | Device-Agnostic                             | |
  +-------------+  | |                                             | |
          ^        | |                       +---------------------+ |
          |        | |                       |   ~~~~~~~~~~~~~~~~~~~~  |
          |        | |                       |  {  +-------------+  } |
          |        | |                       |  }  |////Image////|  { |
          |        | |                       | <-> |/Processing//|  } |
          |        | |                       |  }  |/Algorithms//|  { |
          |        | |                       |  {  +-------------+  } |
          |        | |                       |   ~~~~~~~~~~~~~~~~~~~~  |
          |        | |                       | ====================== |
          |        | |                       |     +-------------+    |
          |        | |                       |     |//Pipeline///|    |
          |        | |                       | <-> |///Handler///|    |
          |        | |                       |     |/////////////|    |
          |        | +-----------------------+     +-------------+    |
          |        |                                 Device-Specific |
          |        +-------------------------------------------------+
          |                     ^                        ^
          |                     |                        |
          v                     v                        v
  +--------------------------------------------------------------------+
  | Helpers and Support Classes                                        |
  | +-------------+  +-------------+  +-------------+  +-------------+ |
  | |  MC & V4L2  |  |   Buffers   |  | Sandboxing  |  |   Plugins   | |
  | |   Support   |  |  Allocator  |  |     IPC     |  |   Manager   | |
  | +-------------+  +-------------+  +-------------+  +-------------+ |
  | +-------------+  +-------------+                                   |
  | |  Pipeline   |  |     ...     |                                   |
  | |   Runner    |  |             |                                   |
  | +-------------+  +-------------+                                   |
  +--------------------------------------------------------------------+
```

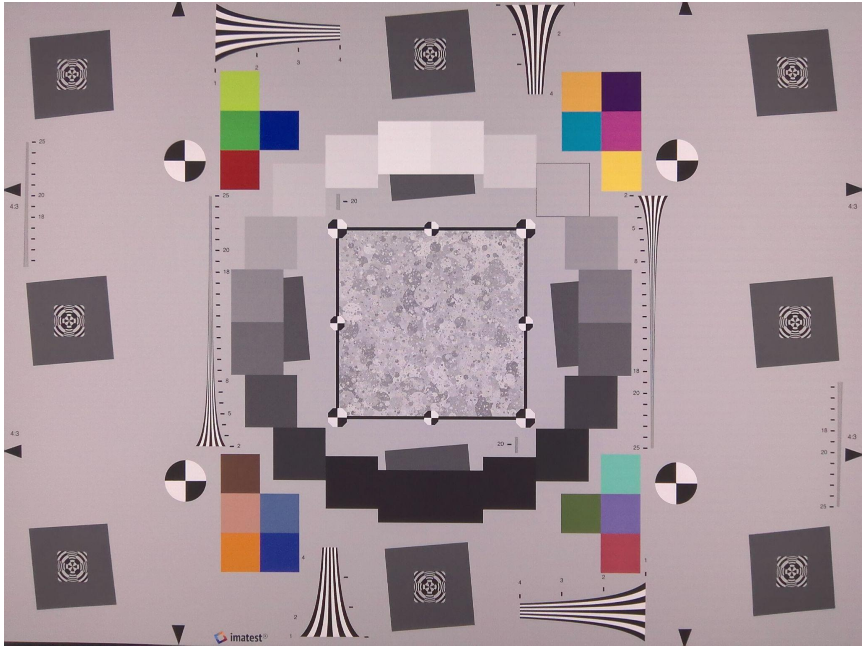
```
/// Device-Specific Components
~~~ Sandboxing
```

Libcamera

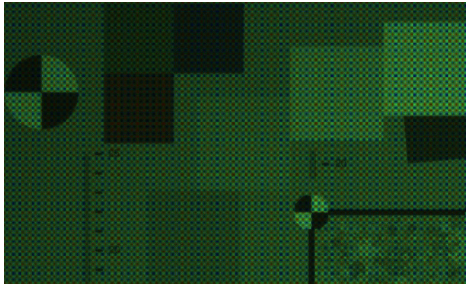
```
-------------------< libcamera Public API >--------------------------
         ^                                    ^
         |                                    |
         v                                    v
  +-------------+   +--------------------------------------------------+
  |   Camera    |   |  Camera Device                                   |
  |   Devices   |   |  +--------------------------------------------+  |
  |   Manager   |   |  | Device-Agnostic                            |  |
  +-------------+   |  |                                            |  |
         ^          |  |                       +--------------------+  |
         |          |  |                       | ~~~~~~~~~~~~~~~~~~~~~ |
         |          |  |                       |  {  +-------------+  } |
         |          |  |                       |  }  |////Image////|  { |
         |          |  |                       | <-> |/Processing//|  } |
         |          |  |                       |  }  |/Algorithms//|  { |
         |          |  |                       |  {  +-------------+  } |
         |          |  |                       | ~~~~~~~~~~~~~~~~~~~~~ |
         |          |  |                       | ======================= |
         |          |  |                       |     +-------------+    |
         |          |  |                       |     |//Pipeline///|    |
         |          |  |                       | <-> |///Handler///|    |
         |          |  |                       |     |/////////////|    |
         |          |  +-----------------------+     +-------------+    |
         |          |                                  Device-Specific  |
         |          +--------------------------------------------------+
         |                            ^                        ^
         |                            |                        |
         v                            v                        v
+--------------------------------------------------------------------+
| Helpers and Support Classes                                        |
| +-------------+  +-------------+  +-------------+  +-------------+ |
| | MC & V4L2   |  |   Buffers   |  | Sandboxing  |  |   Plugins   | |
| |  Support    |  |  Allocator  |  |     IPC     |  |   Manager   | |
| +-------------+  +-------------+  +-------------+  +-------------+ |
| +-------------+  +-------------+                                   |
| |  Pipeline   |  |     ...     |                                   |
| |   Runner    |  |             |                                   |
| +-------------+  +-------------+                                   |
+--------------------------------------------------------------------+
```

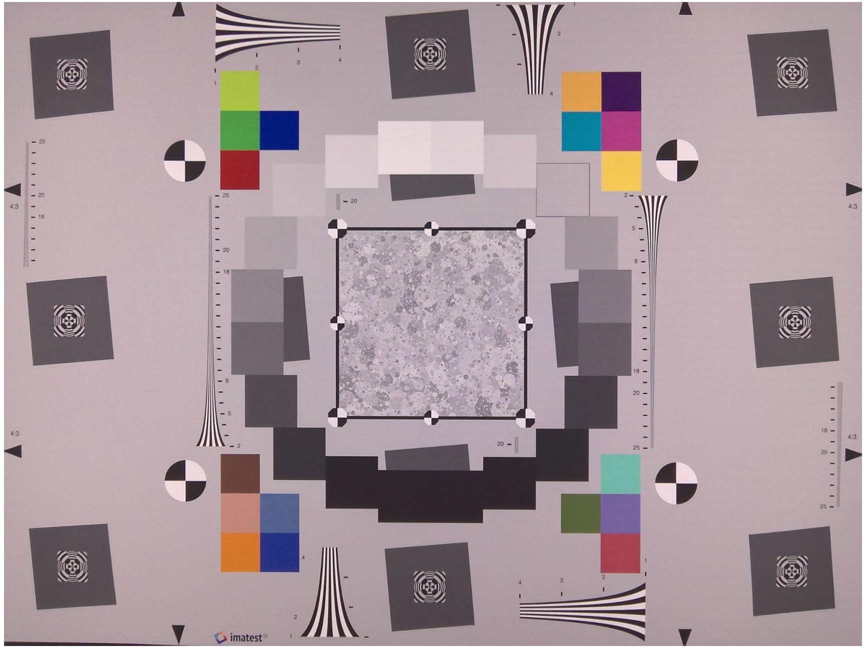
/// Device-Sp
~~~ Sandboxin


~~~

Secret Sauce

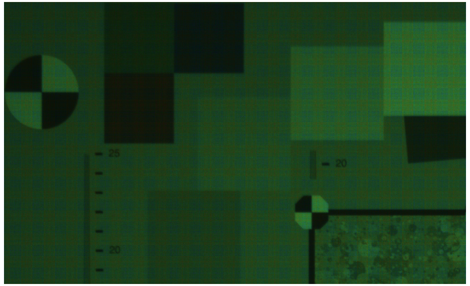


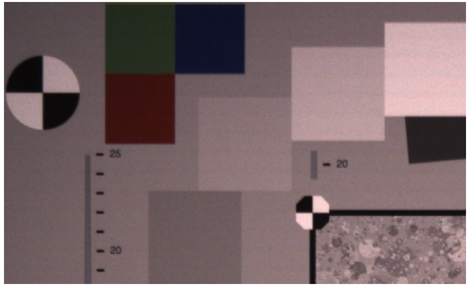
Secret Sauce

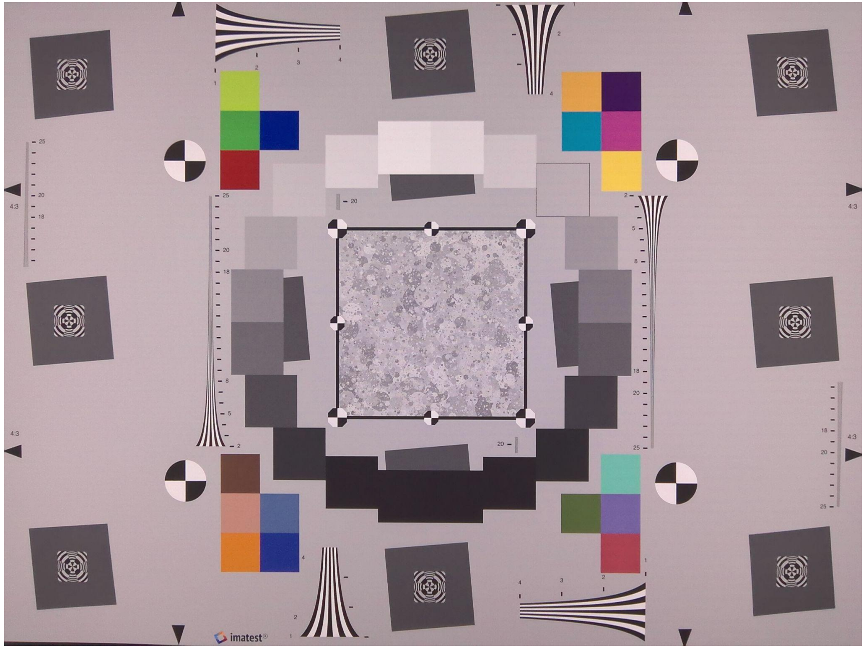




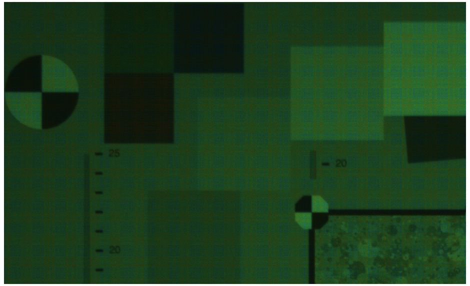
Secret Sauce

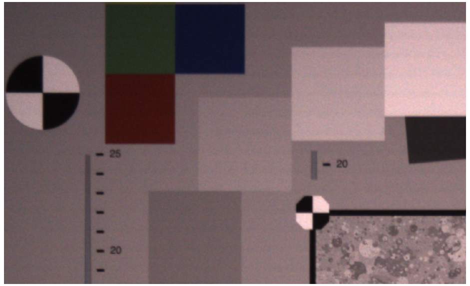


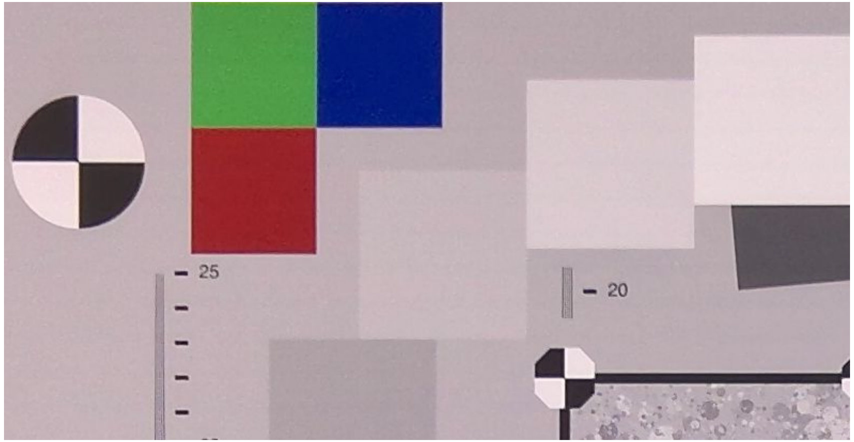


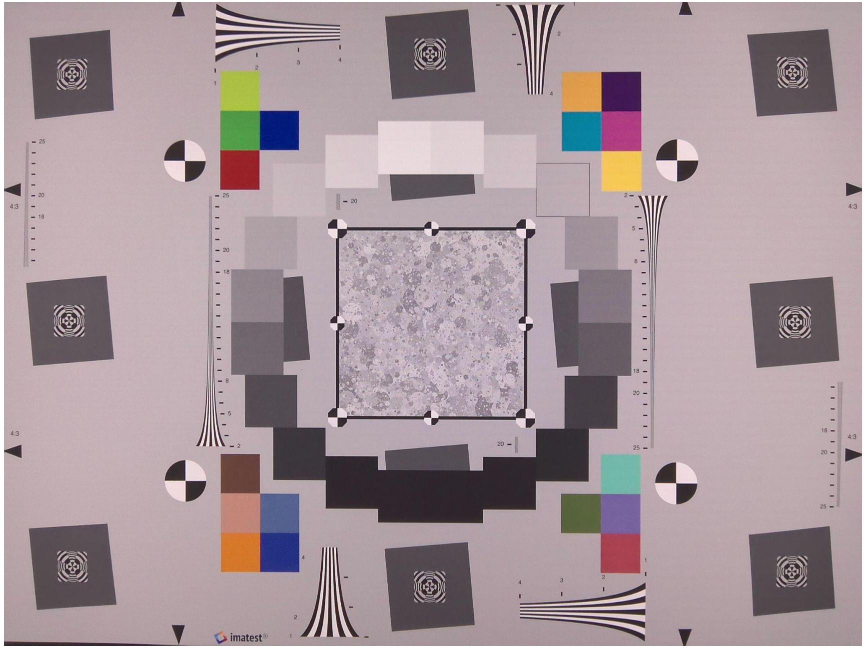


Secret Sauce

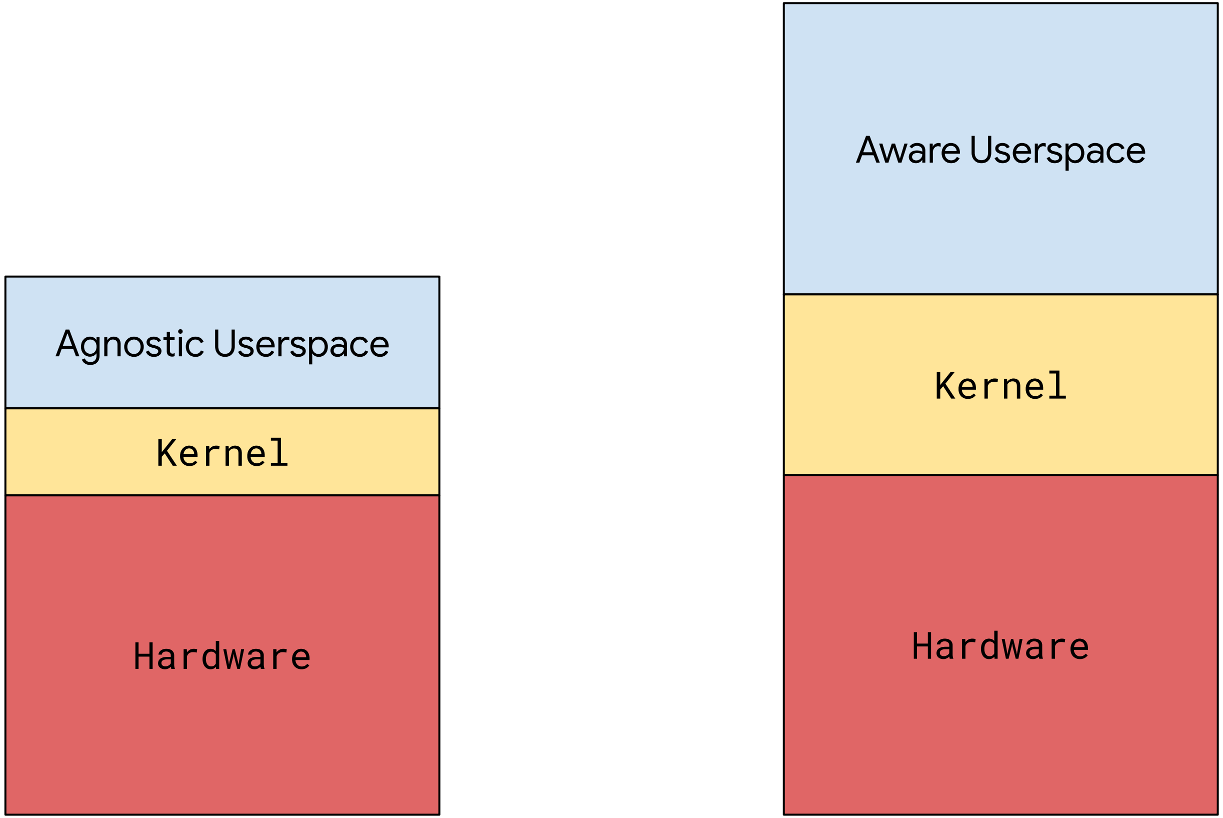








Recap



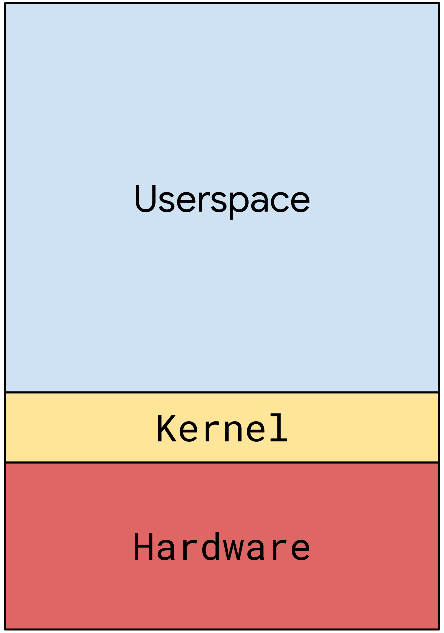
V4L2

Media Controller

Kernel CAM

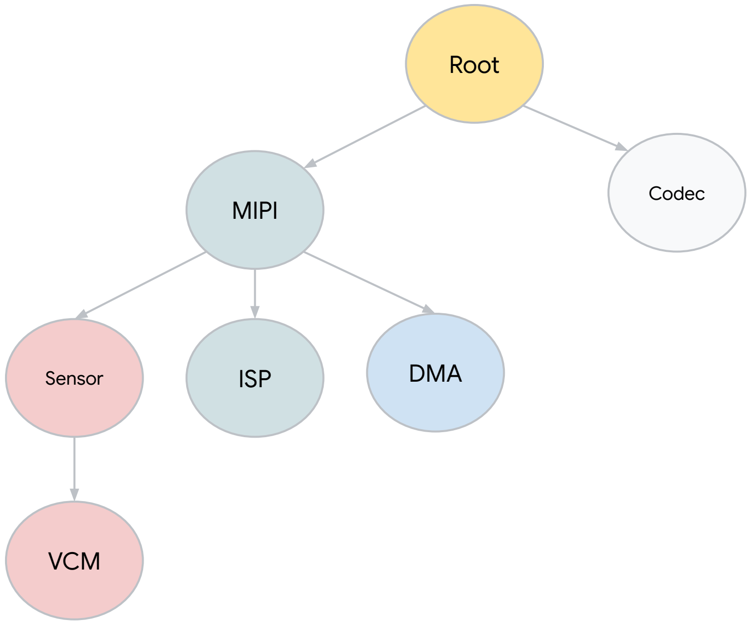
What is CAM?

- New kernel subsystem.
- No media abstractions.
- Fluid collaboration between userspace and other subsystems.
- Two components:
  - Entities
  - Operations



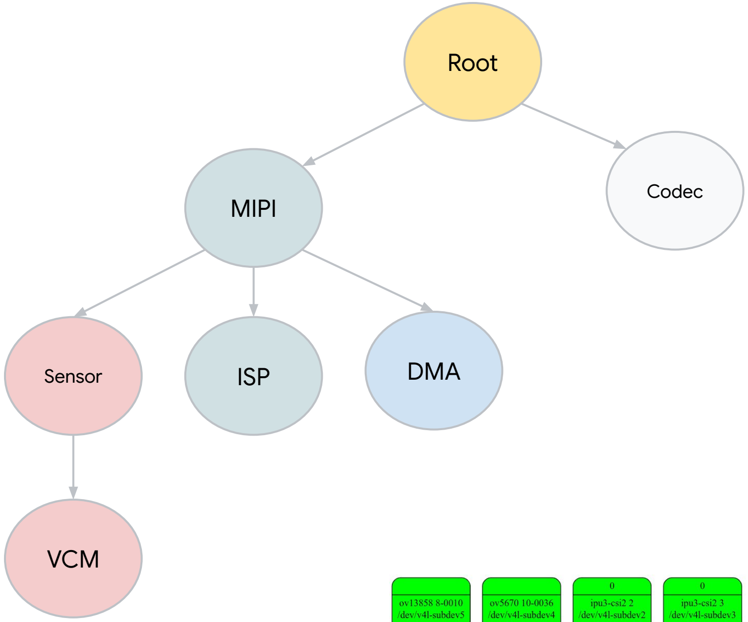
KCAM components: Entities

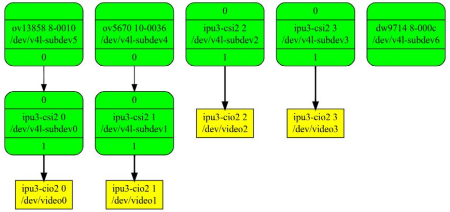
- Organized in a tree, based on the hardware architecture.
- Single register-set.
- Can throw events.



KCAM components: Entities

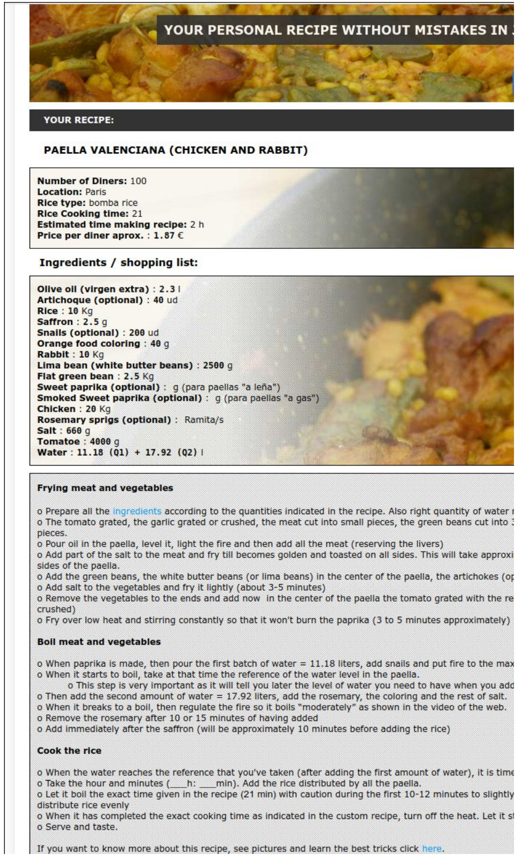
- Organized in a tree, based on the hardware architecture.
- Single register-set.
- Can throw events.





KCAM components: Operations

- Read and/or write to an entity
  - regmap
  - parameter buffers
- Can depend on:
  - Events
  - Other operation
  - Fence
- Can create a fence



YOUR PERSONAL RECIPE WITHOUT MISTAKES IN

YOUR RECIPE:

PAELLA VALENCIANA (CHICKEN AND RABBIT)

Number of Diners: 100
Location: Paris
Rice type: bomba rice
Rice Cooking time: 21
Estimated time making recipe: 2 h
Price per diner aprox. : 1.87 €

Ingredients / shopping list:

Olive oil (virgen extra) : 2.3 l
Artichoque (optional) : 40 ud
Rice : 10 Kg
Saffron : 2.5 g
Snails (optional) : 200 ud
Orange food coloring : 40 g
Rabbit : 10 Kg
Lima bean (white butter beans) : 2500 g
Flat green bean : 2.5 Kg
Sweet paprika (optional) : g (para paellas "a leña")
Smoked Sweet paprika (optional) : g (para paellas "a gas")
Chicken : 20 Kg
Rosemary sprigs (optional) : Ramita/s
Salt : 660 g
Tomatoe : 4000 g
Water : 11.18 (Q1) + 17.92 (Q2) l

Frying meat and vegetables

o Prepare all the ingredients according to the quantities indicated in the recipe. Also right quantity of water
o The tomato grated, the garlic grated or crushed, the meat cut into small pieces, the green beans cut into
pieces.
o Pour oil in the paella, level it, light the fire and then add all the meat (reserving the livers)
o Add part of the salt to the meat and fry till becomes golden and toasted on all sides. This will take approxi
sides of the paella.
o Add the green beans, the white butter beans (or lima beans) in the center of the paella, the artichokes (op
o Add salt to the vegetables and fry it lightly (about 3-5 minutes)
o Remove the vegetables to the ends and add now in the center of the paella the tomato grated with the re
crushed)
o Fry over low heat and stirring constantly so that it won't burn the paprika (3 to 5 minutes approximately)

Boil meat and vegetables

o When paprika is made, then pour the first batch of water = 11.18 liters, add snails and put fire to the max
o When it starts to boil, take at that time the reference of the water level in the paella.
o This step is very important as it will tell you later the level of water you need to have when you add
o Then add the second amount of water = 17.92 liters, add the rosemary, the coloring and the rest of salt.
o When it breaks to a boil, then regulate the fire so it boils "moderately" as shown in the video of the web.
o Remove the rosemary after 10 or 15 minutes of having added
o Add immediately after the saffron (will be approximately 10 minutes before adding the rice)

Cook the rice

o When the water reaches the reference that you've taken (after adding the first amount of water), it is time
o Take the hour and minutes (___h: ___min). Add the rice distributed by all the paella.
o Let it boil the exact time given in the recipe (21 min) with caution during the first 10-12 minutes to slightly
distribute rice evenly
o When it has completed the exact cooking time as indicated in the custom recipe, turn off the heat. Let it st
o Serve and taste.

If you want to know more about this recipe, see pictures and learn the best tricks click here.

http://www.paella10.com/calculadora/paellas/paella-valenciana/

KCAM components: Operations

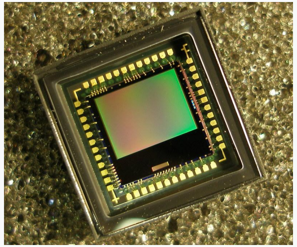
fence_in

ioctl

Operation List
Configure Sensor
Trigger image
Init ISP
Start DMA

ready

signal



Circular buffer

fence_out

Show me the code!!

```
#define CAM_MAX_DEPENDENCIES            8
struct cam_operation_add {
        __u32                   id;
        __u32                   fence_out;
        __u32                   flags;
        /*
         * Pre-execution dependencies list and dependency
execution mode
         */
        __u32                   mode;
        struct cam_dependency deps[CAM_MAX_DEPENDENCIES];
        /*
         * Execution context specific data (if any)
         */
        __u64                   delay_ns;
        __u64                   rd_wr_list;
        __u32                   entity;
} __attribute__((packed));
```

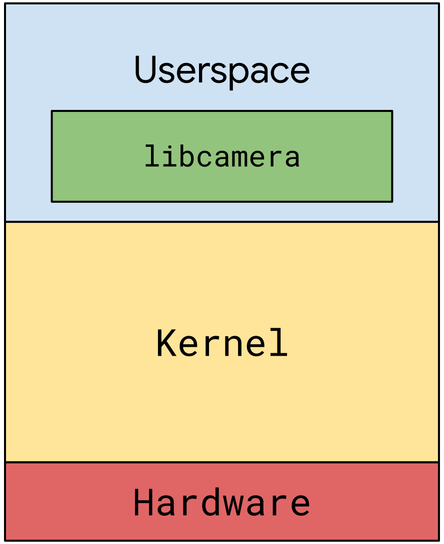
```
enum cam_dependency_type {
        CAM_DEPENDENCY_NONE,
        CAM_DEPENDENCY_OP,
        CAM_DEPENDENCY_EVENT,
        CAM_DEPENDENCY_FENCE_IN,
};

struct cam_dependency {
        __u32           type;
        __u32           id;
} __attribute__((packed));

enum cam_dependency_mode {
        CAM_DEPENDENCY_WEAK_ORDER,
        CAM_DEPENDENCY_STRICT_ORDER,
};
```

Testing (lots of)

- Software testing:
  - kunit
  - libkc
  - vcam
  - error injection
- Hardware testing
  - ChromeOS test infra
  - human test

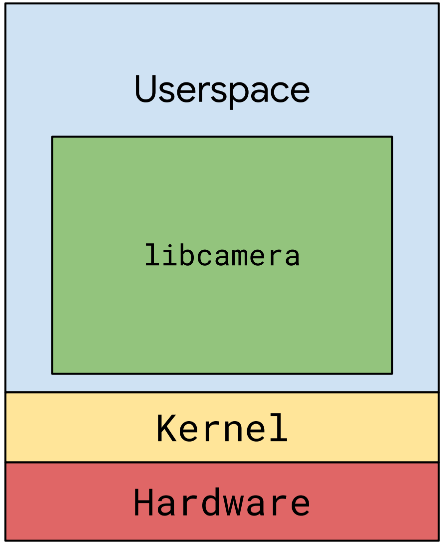
Comparison

Media
Controller



- Limitless functionality
- Fast upstream
- Small drivers
- Operations vs Streams

Kernel CAM



Google

Governance

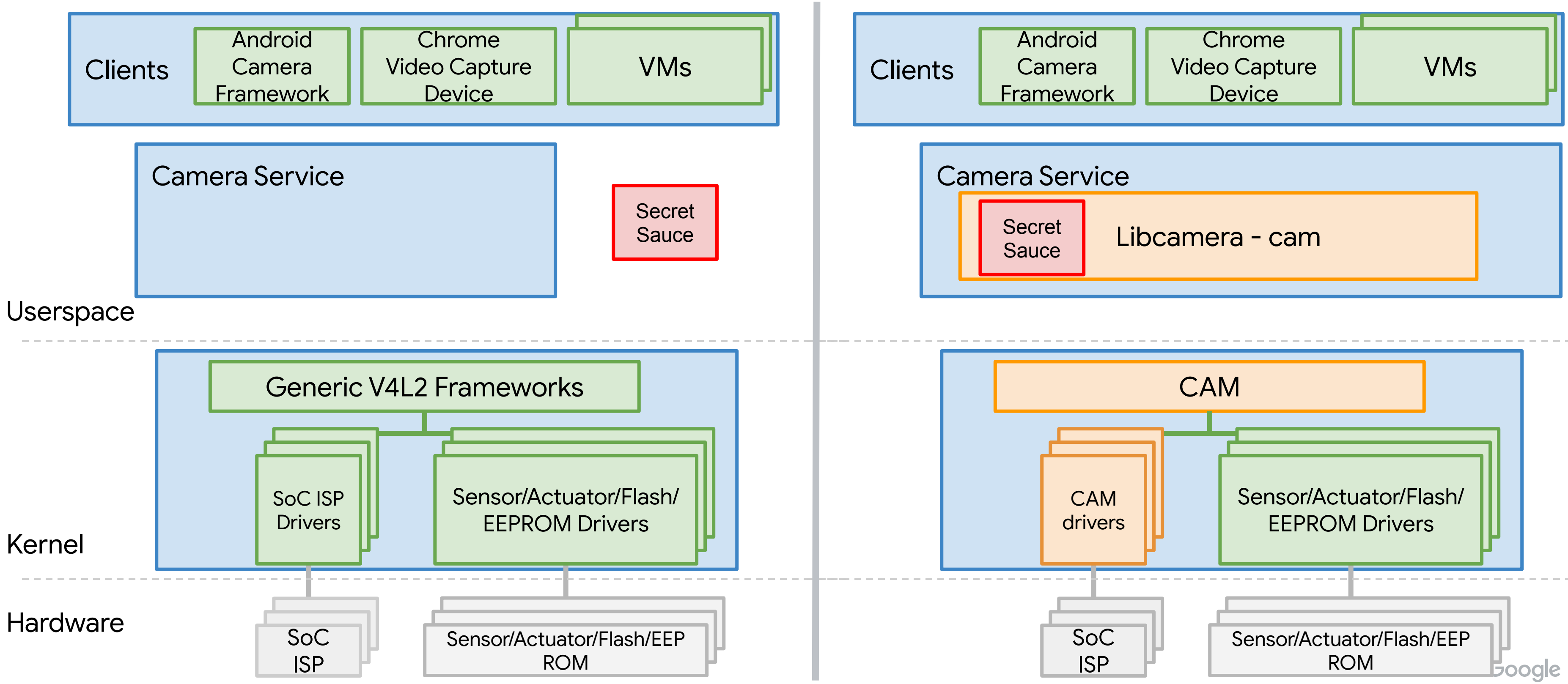
- There must be an open source stack before a driver is merged.
  - Similar to DRM model.
- Compliance-test with minimum requirements.

Google

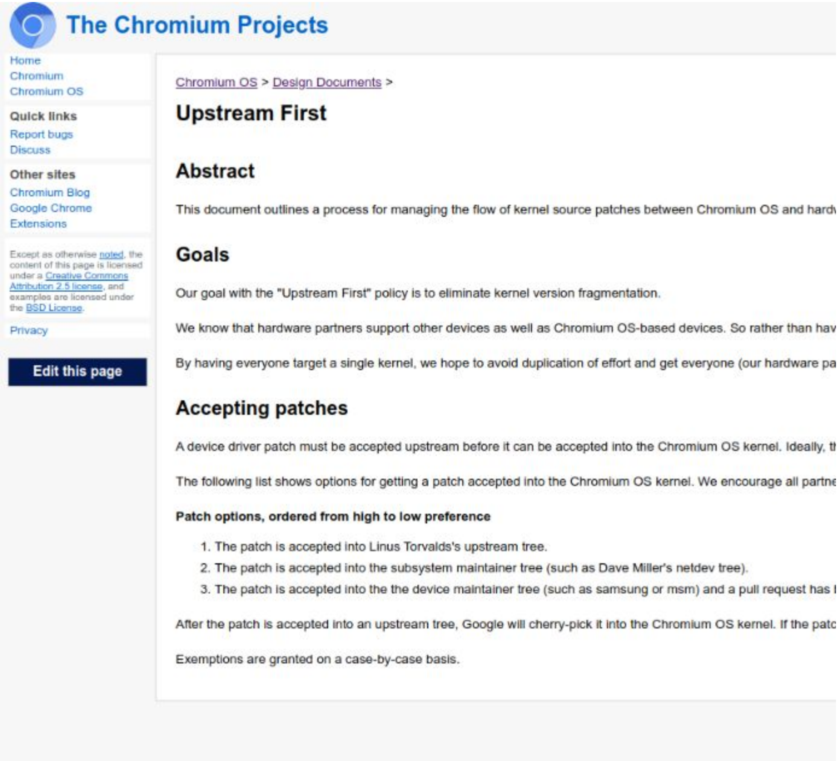
Future

Benchmark the stack

chromeOS



Land upstream



The Chromium Projects

Home
Chromium
Chromium OS

Quick links
Report bugs
Discuss

Other sites
Chromium Blog
Google Chrome
Extensions

Privacy

Edit this page

Chromium OS > Design Documents >

Upstream First

Abstract

This document outlines a process for managing the flow of kernel source patches between Chromium OS and hard

Goals

Our goal with the "Upstream First" policy is to eliminate kernel version fragmentation.

We know that hardware partners support other devices as well as Chromium OS-based devices. So rather than hav

By having everyone target a single kernel, we hope to avoid duplication of effort and get everyone (our hardware pa

Accepting patches

A device driver patch must be accepted upstream before it can be accepted into the Chromium OS kernel. Ideally, t

The following list shows options for getting a patch accepted into the Chromium OS kernel. We encourage all partne

Patch options, ordered from high to low preference

1. The patch is accepted into Linus Torvalds's upstream tree.
2. The patch is accepted into the subsystem maintainer tree (such as Dave Miller's netdev tree).
3. The patch is accepted into the the device maintainer tree (such as samsung or msm) and a pull request has

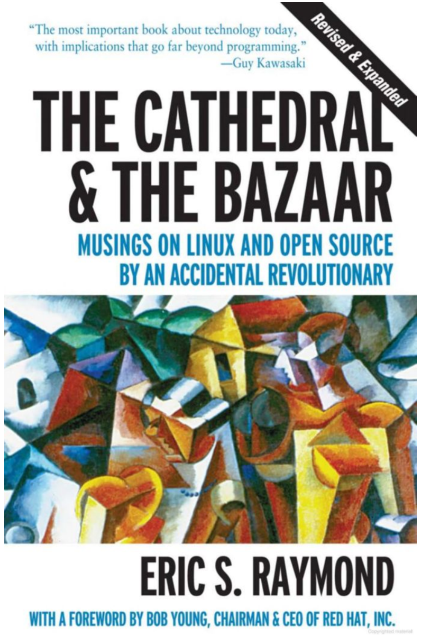
After the patch is accepted into an upstream tree, Google will cherry-pick it into the Chromium OS kernel. If the patc

Exemptions are granted on a case-by-case basis.

Contributing

How to contribute

- Show me the code!
  - https://chromium.googlesource.com/chromiumos/third_party/kernel/+/refs/heads/kcam
- Mailing List
  - kernel-cam@chromium.org
- Bugs:
  - WIP
- Join the team
  - https://crosjobs.page.link/event



Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

37

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

38

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

42

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

46

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

50

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

51

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

54

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

55

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

56

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

57

ribalda@chromium.org

https://crosjobs.page.link/event

Google

Thank you!

ribalda@chromium.org

https://crosjobs.page.link/event

Thank you!

0xffffffff

ribalda@chromium.org

https://crosjobs.page.link/event