Display and Kernel Review and Future

Kernel Recipes 2013
Paris

Laurent Pinchart
laurent.pinchart@ideasonboard.com


oxygen


Desktop
This folder is empty
Softpedia Linux (softpedia) on softpedia-Virtual
kubuntu
Search:
Favorites
Web Browser
Personal Information Manager
System Settings
File Manager
Instant Messenger Contacts
Audio Player
Favorites
Applications
Computer
Recently Used
Leave
oxygen

display /

/ graphics /

/ video

display / graphics / video

format

memory / deep pipeline

device / CPU

format
memory / deep pipeline
device / CPU

rotation /

/ scaling /

/ composing

rotation
scaling
composing

X11 /

/ Wayland /

/ MIR /

/ SDL /

/ DirectFB /

/ Raw API

X11
Wayland
SDL
DirectFB
Raw API

Problem - Stack

DRM /

/ FBDEV /

/ V4L2

DRM
FBDEV
V4L2

Problem - API

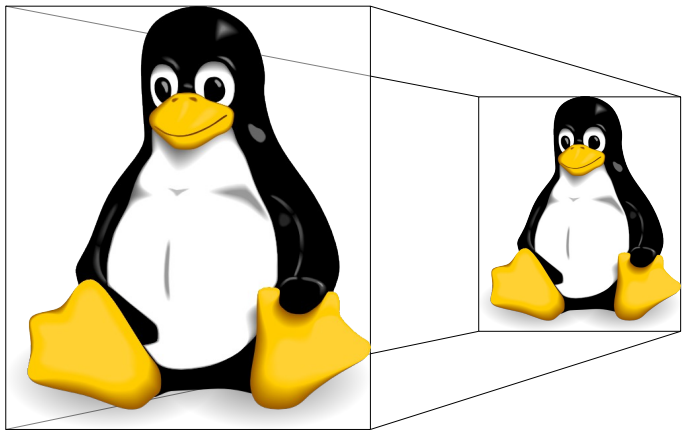
Frame Buffer

Display – Scanout

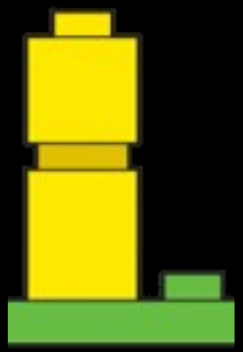
CRTC

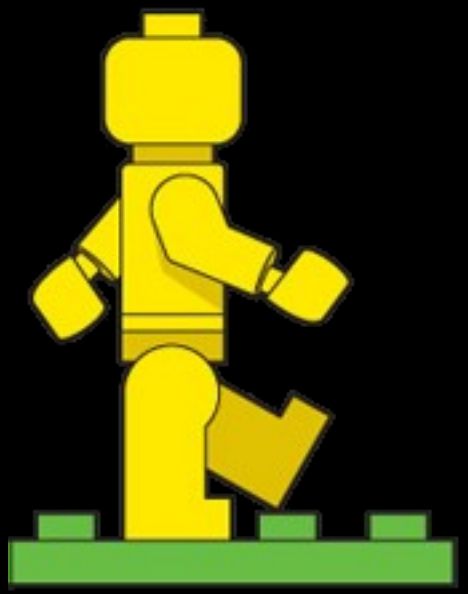
Composition

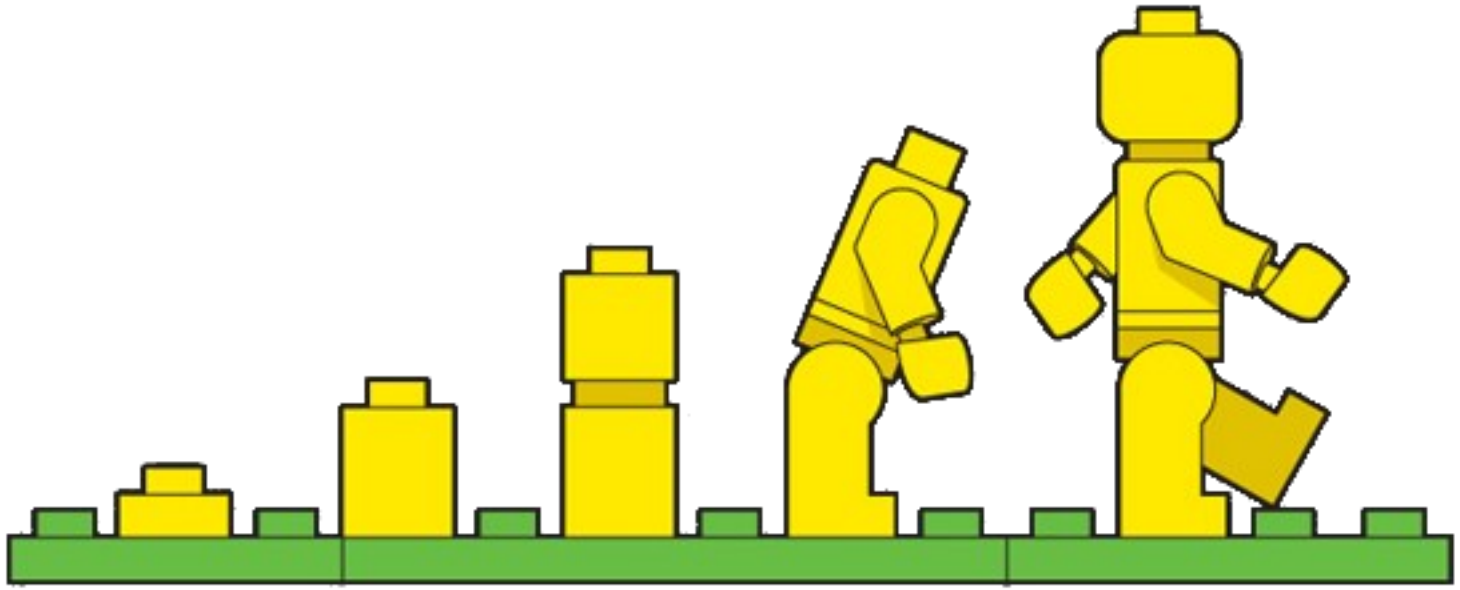
Plane(s)



Display – Composition

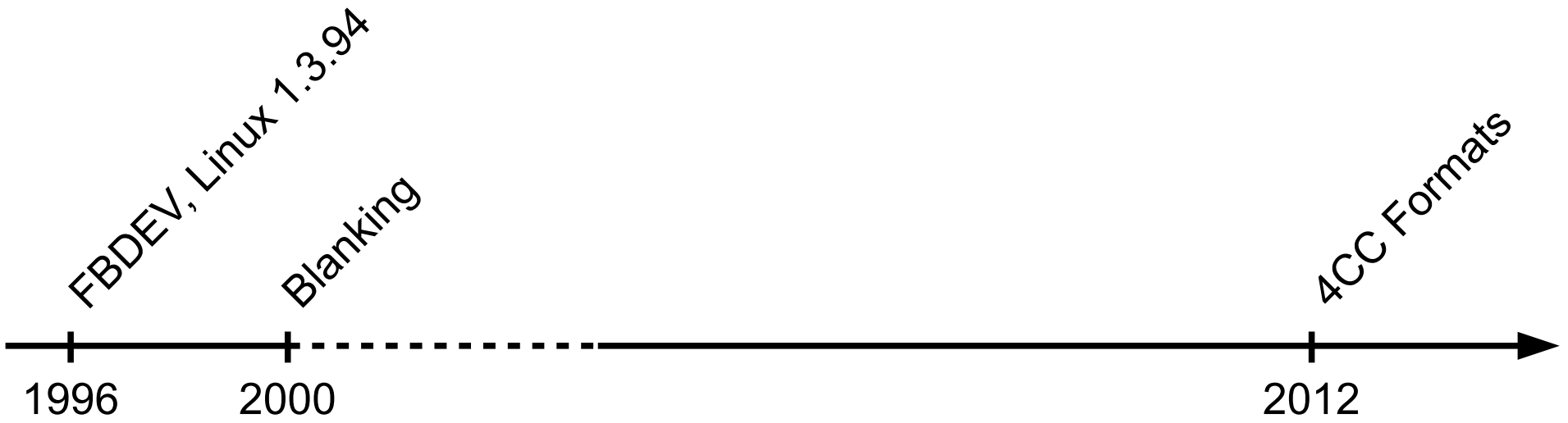




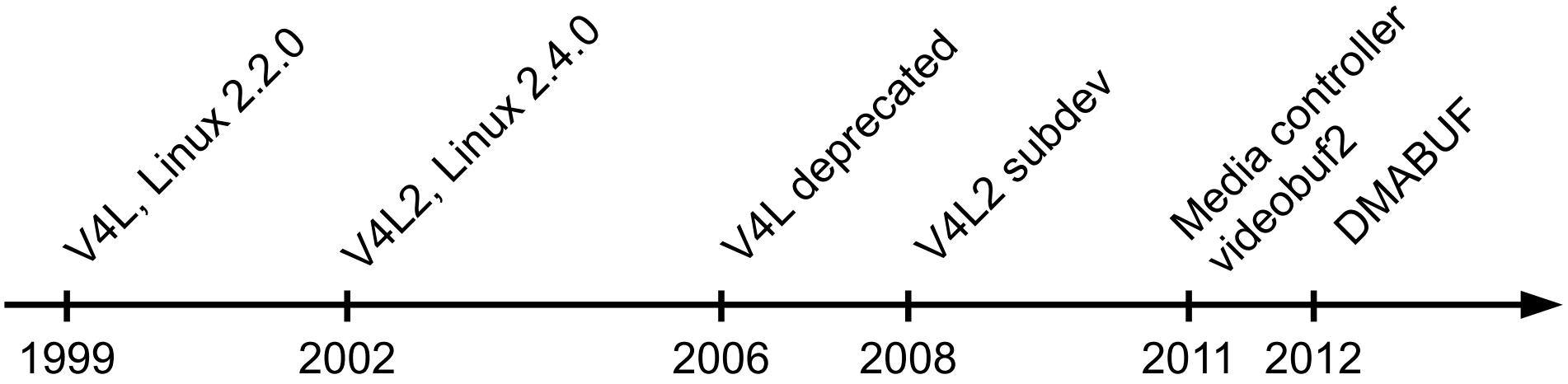


IDEAS ON BOARD

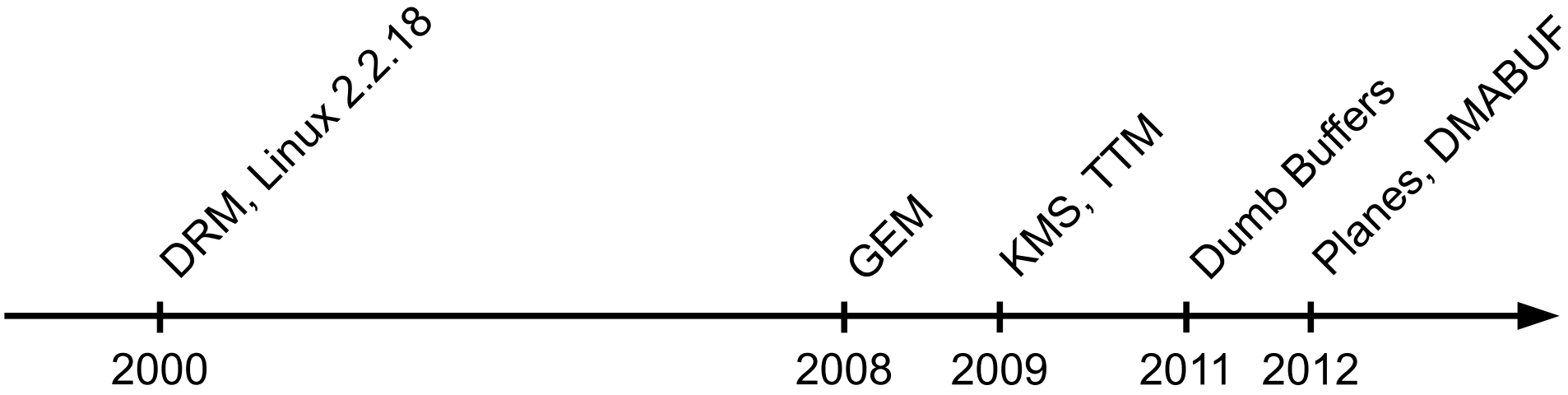
Origins



Origins – FBDEV



Origins – V4L2



Origins – DRM/KMS


IT TOOK A LOT OF WORK, BUT THIS LATEST LINUX PATCH ENABLES SUPPORT FOR MACHINES WITH 4,096 CPUs, UP FROM THE OLD LIMIT OF 1,024.
DO YOU HAVE SUPPORT FOR SMOOTH FULL-SCREEN FLASH VIDEO YET?
NO, BUT WHO USES THAT?


IDEAS
ON BOARD

Features

	DRM	FB	V4L2
Dynamic Allocation	Yes	No	Yes
Multiple Buffers	Yes	panning	Yes
Import	dmabuf	No	userptr
Export	dmabuf mmap	mmap	mmap

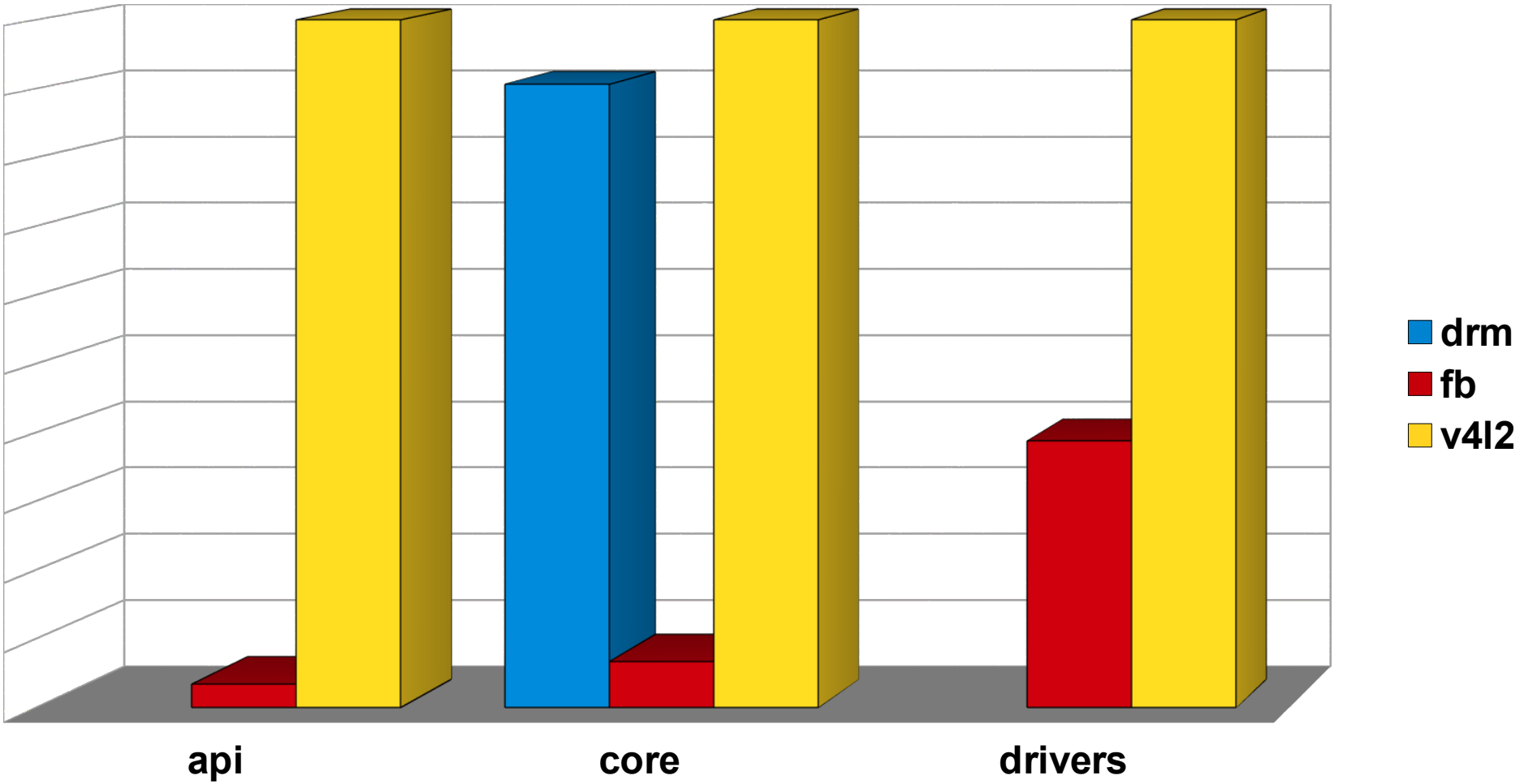
Memory Management

	DRM	FB	V4L2
Formats	4CC	RGB 4CC	4CC
Enumeration	Planes	No	Yes
Negotiation	No	No	Yes
Atomicity	Yes	No	No

Mode Setting

	DRM	FB	V4L2
Overlays	Yes	No	Yes
Rotation	Yes	No	Yes
Scaling	Yes	No	Yes
Cropping/Panning	Yes	Yes	Yes

Transformations



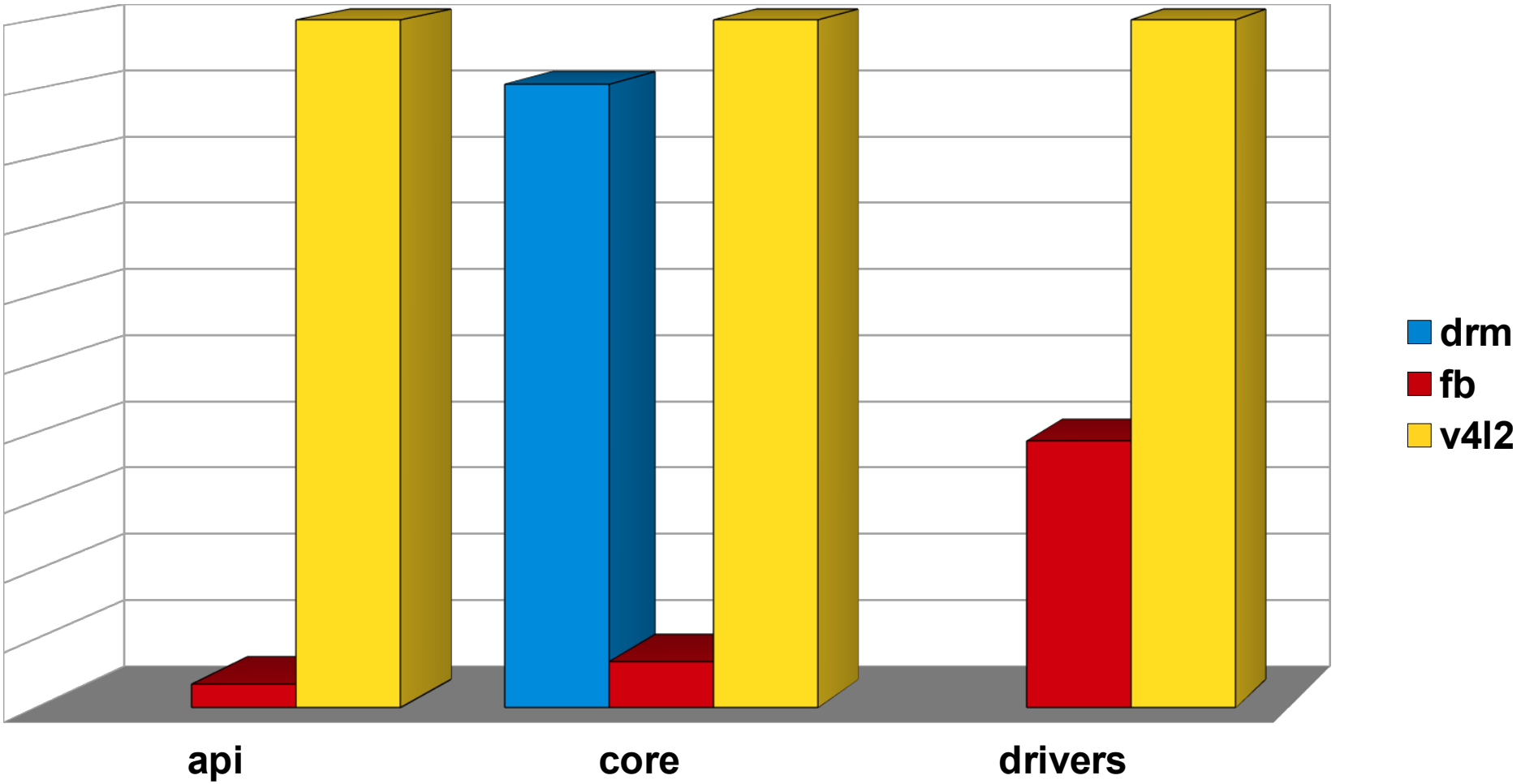
Documentation

“ The DRM core exports several interfaces to applications, generally intended to be used through corresponding libdrm wrapper functions. In addition, drivers export device-specific interfaces for use by userspace drivers & device-aware applications through ioctls and sysfs files.

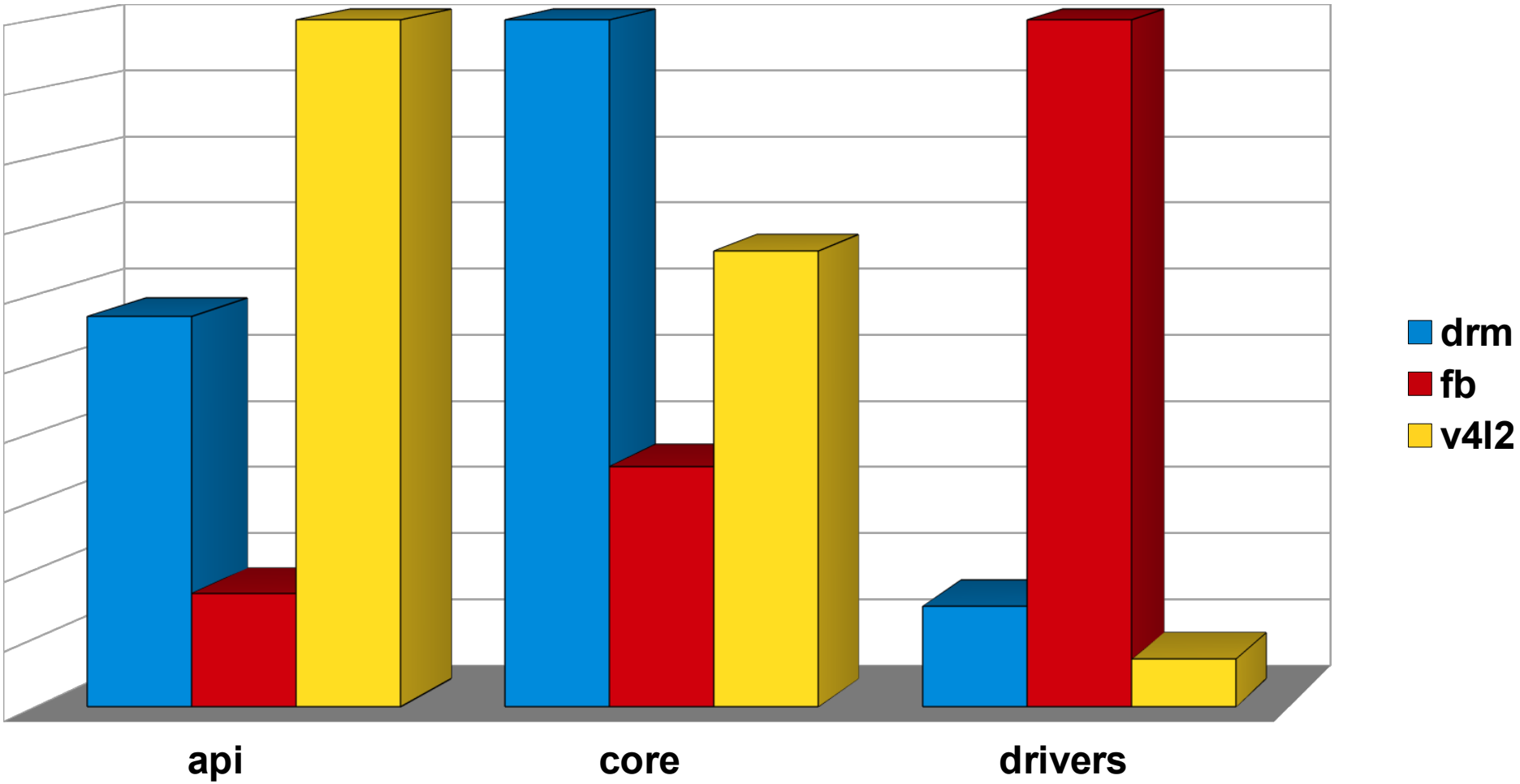
External interfaces include: memory mapping, context management, DMA operations, AGP management, vblank control, fence management, memory management, and output management.

Cover generic ioctls and sysfs layout here. We only need high-level info, since man pages should cover the rest. ”

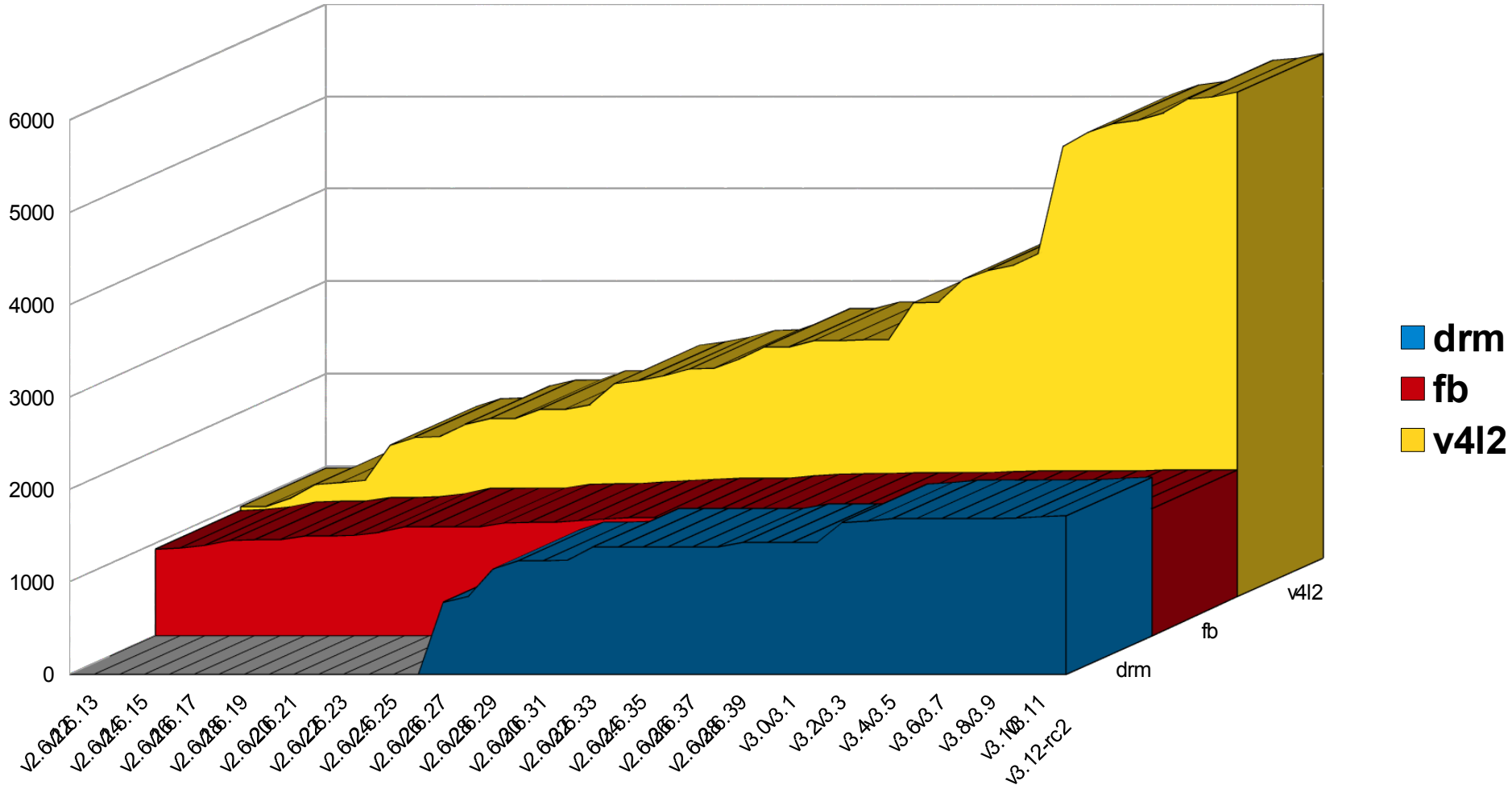
DRM API Documentation



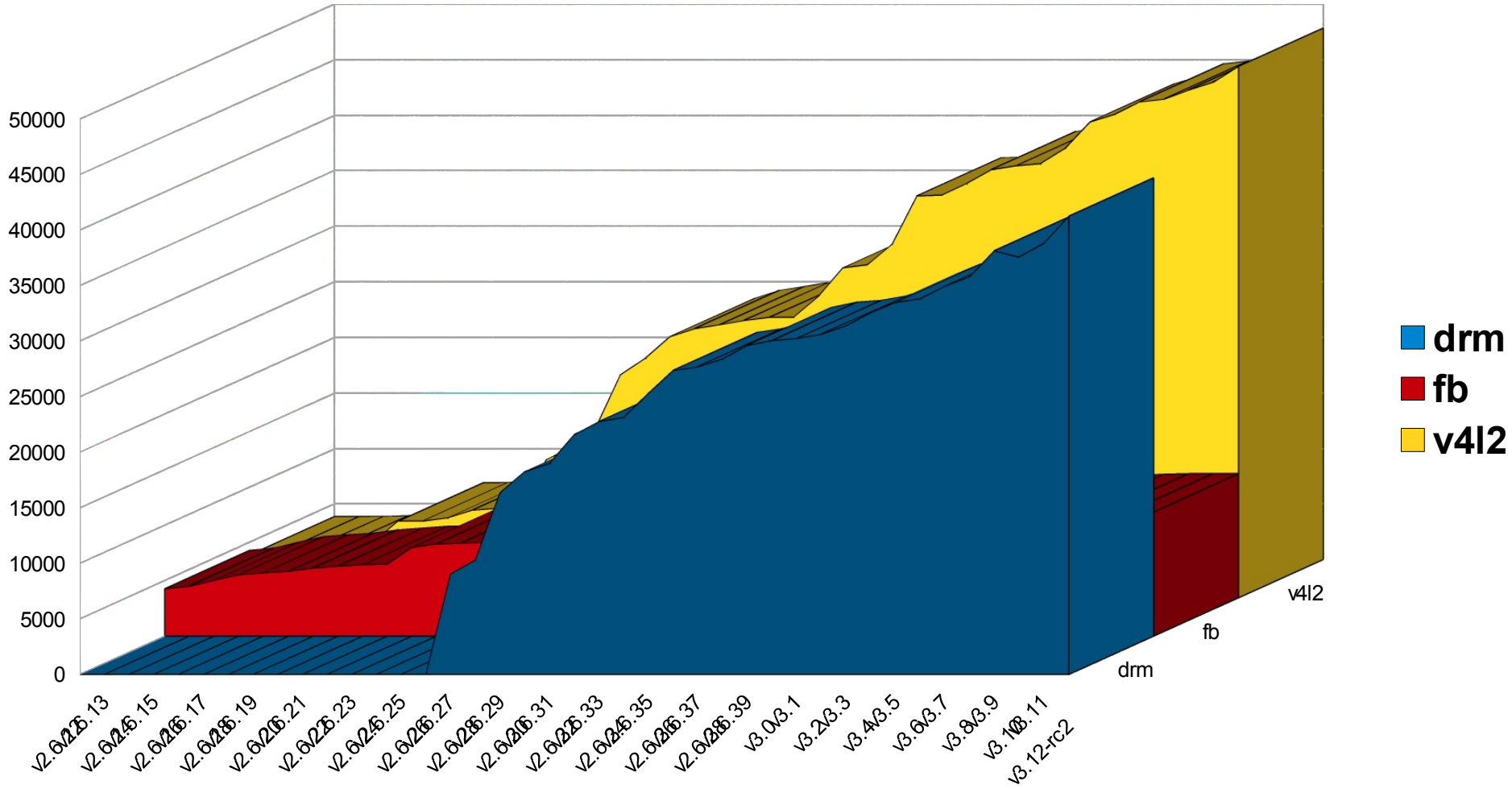
Documentation



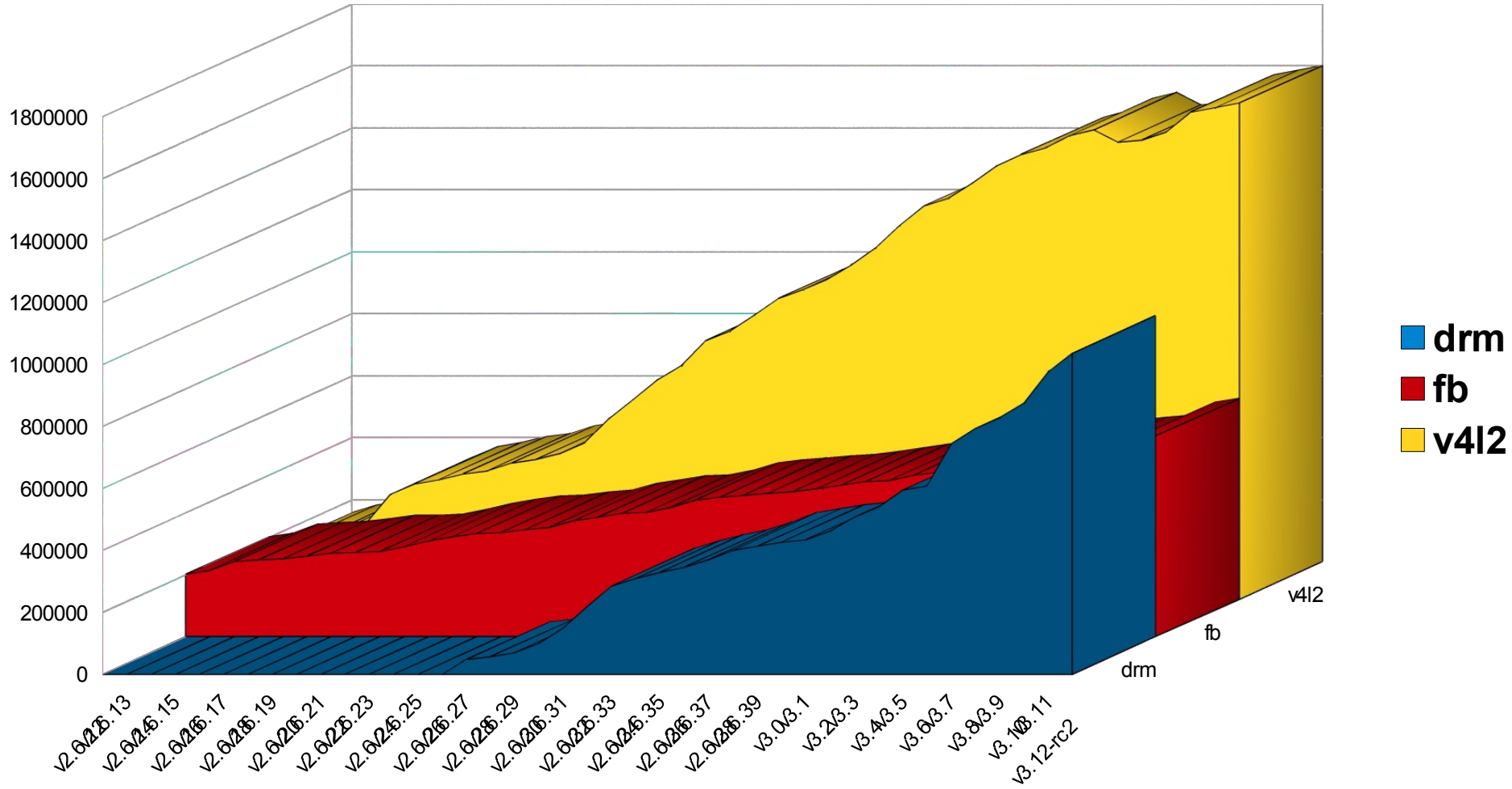
Code Size



Cumulative Changes - API

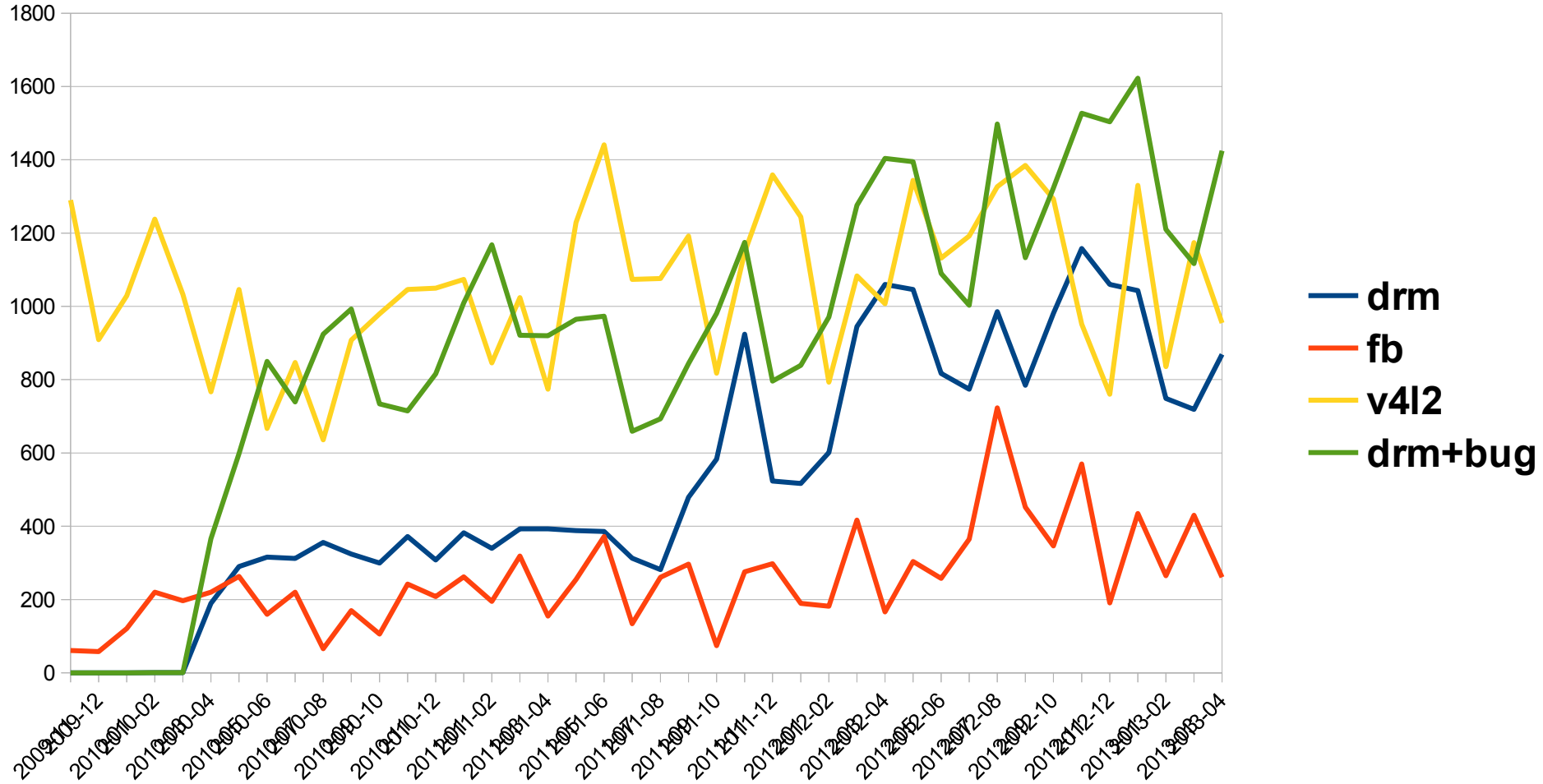


Cumulative Changes - Core

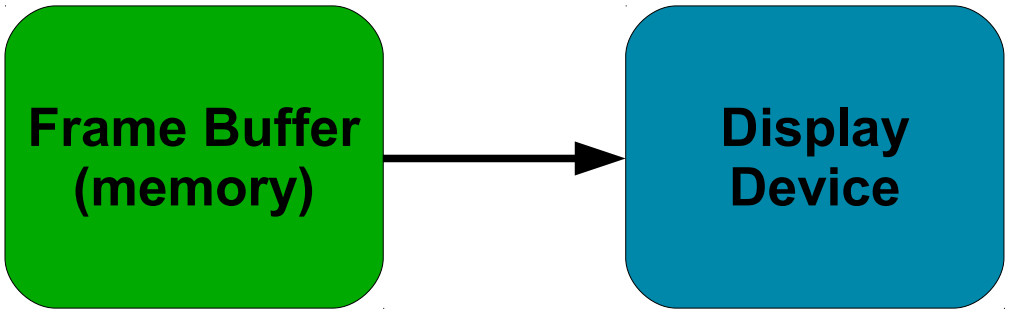


IDEAS ON BOARD

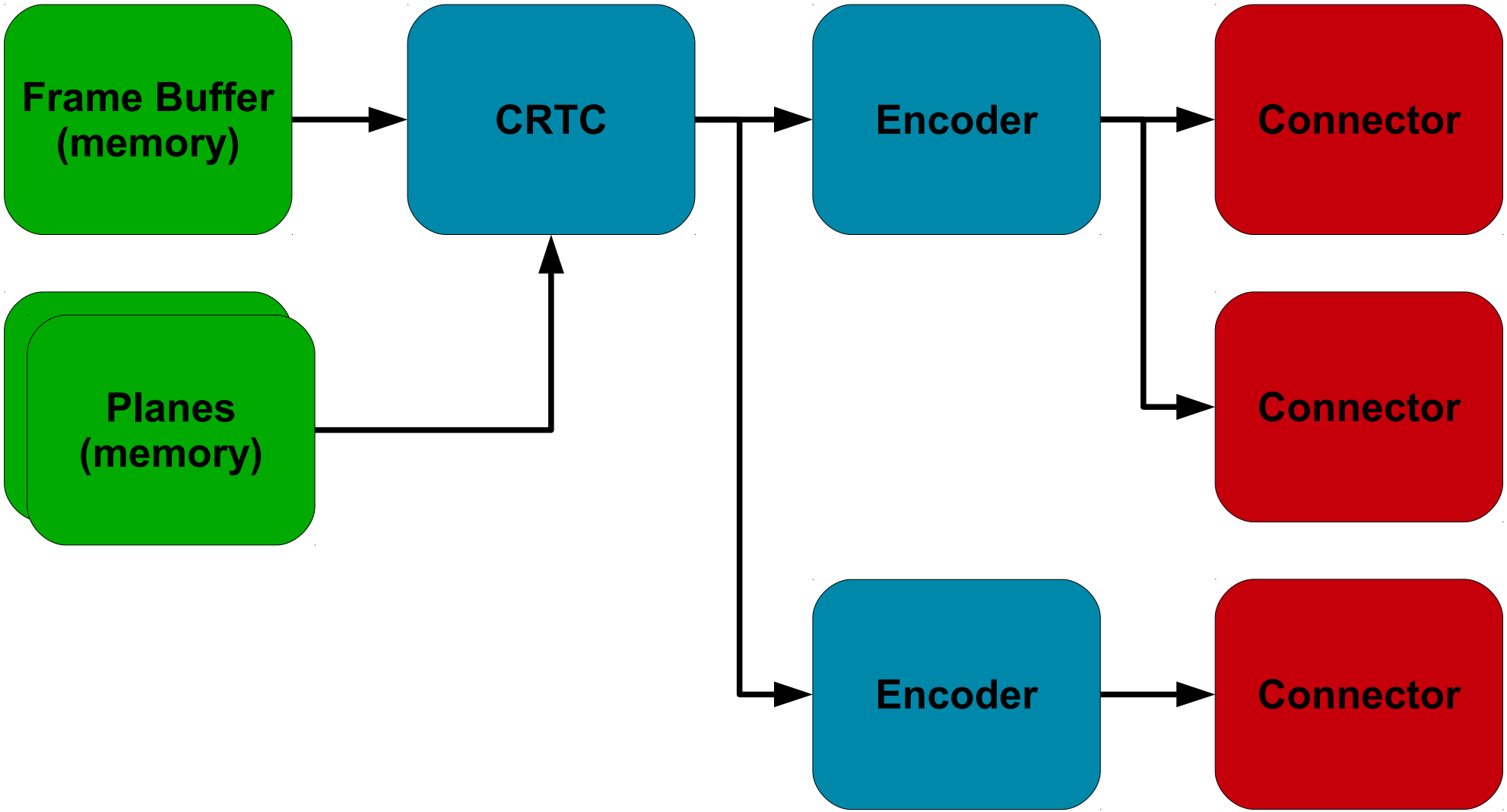
Cumulative Changes - Drivers



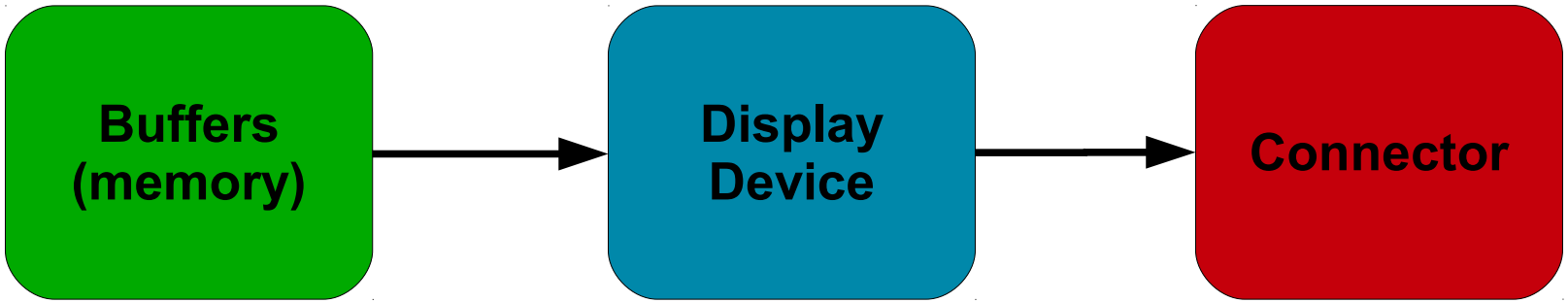
Mailing List Traffic



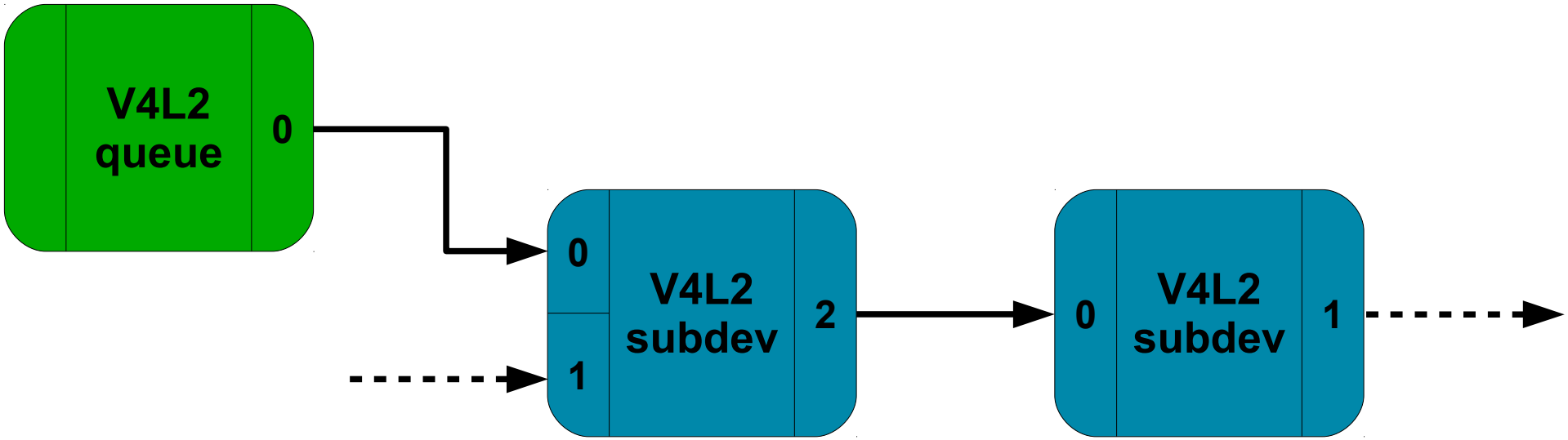
Device Model – FBDEV



Device Model – DRM/KMS

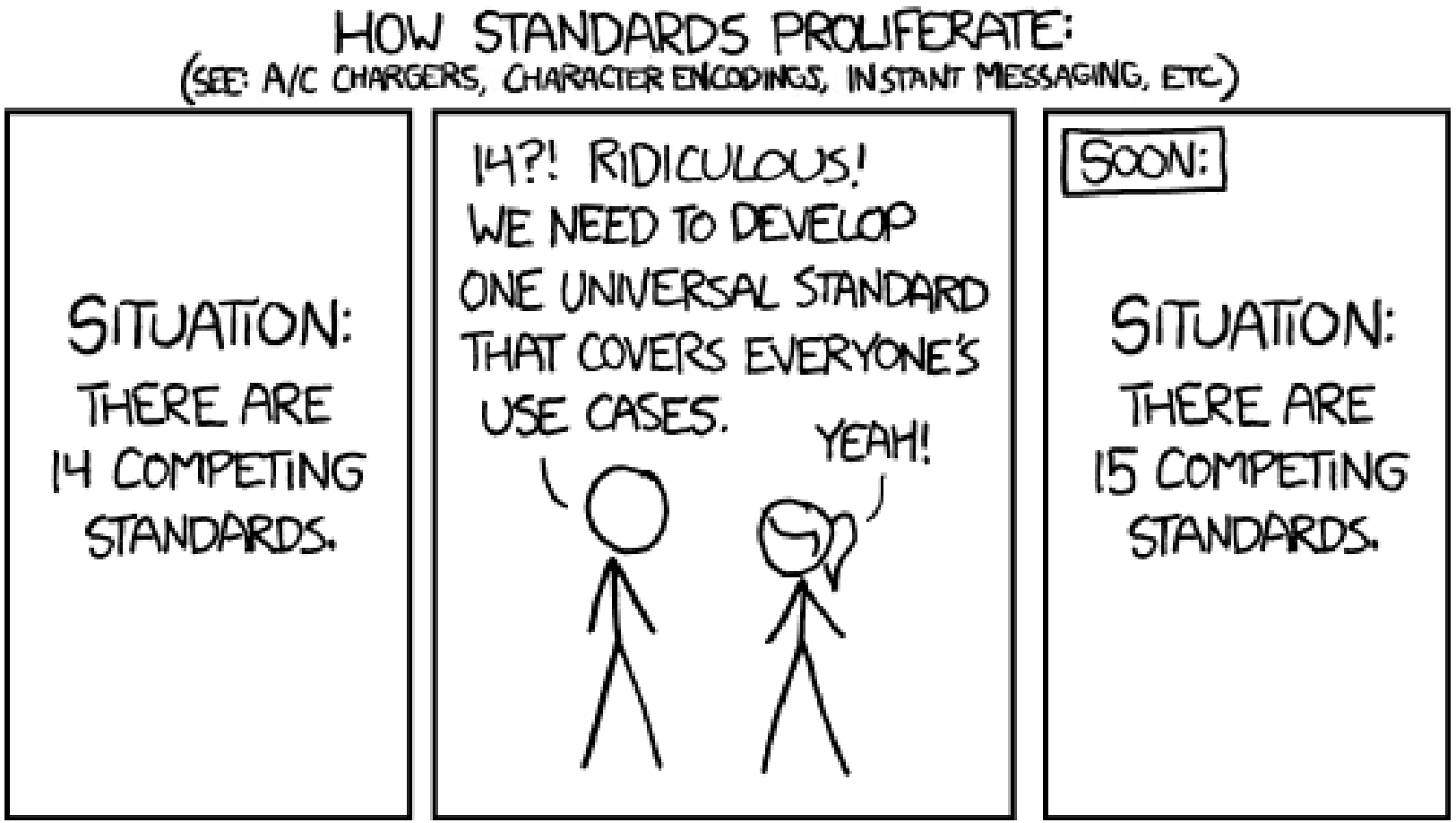


Device Model – V4L2



Device Model – V4L2/MC

Device Model – V4L2/MC


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

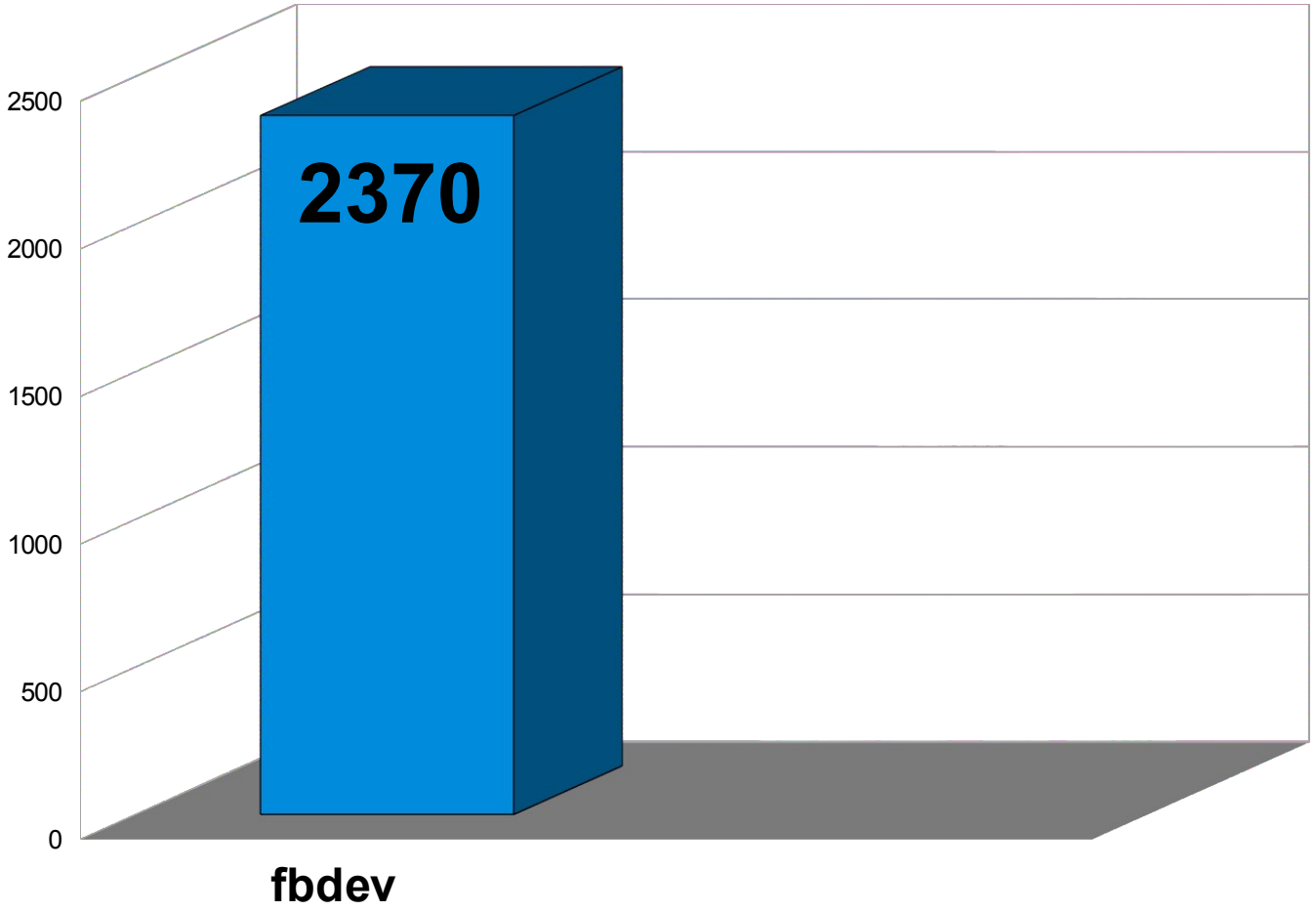
Use Cases

Use Cases - FBDEV

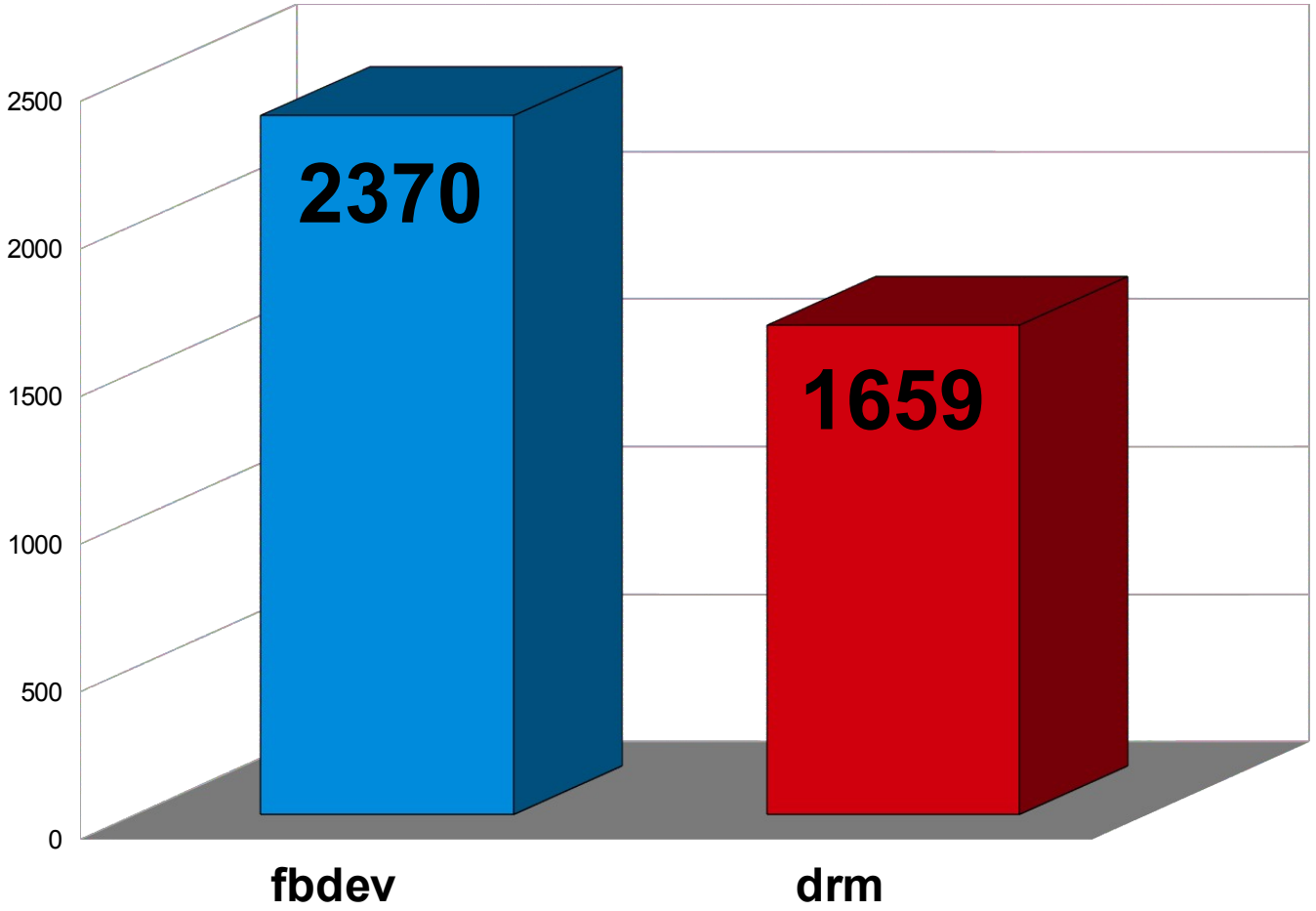
(that's it...)

Video

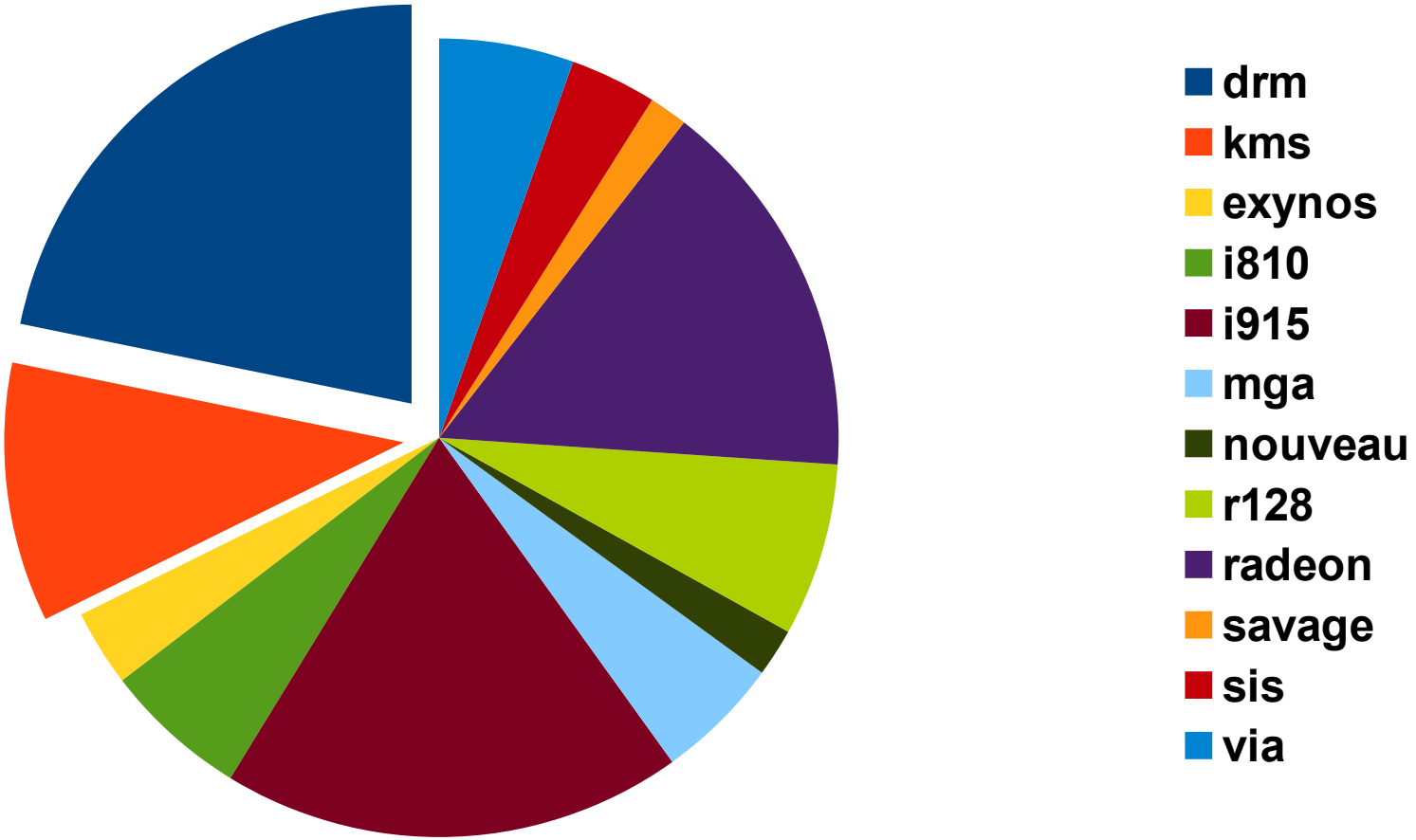
Everything else



FB vs. DRM - sloccount

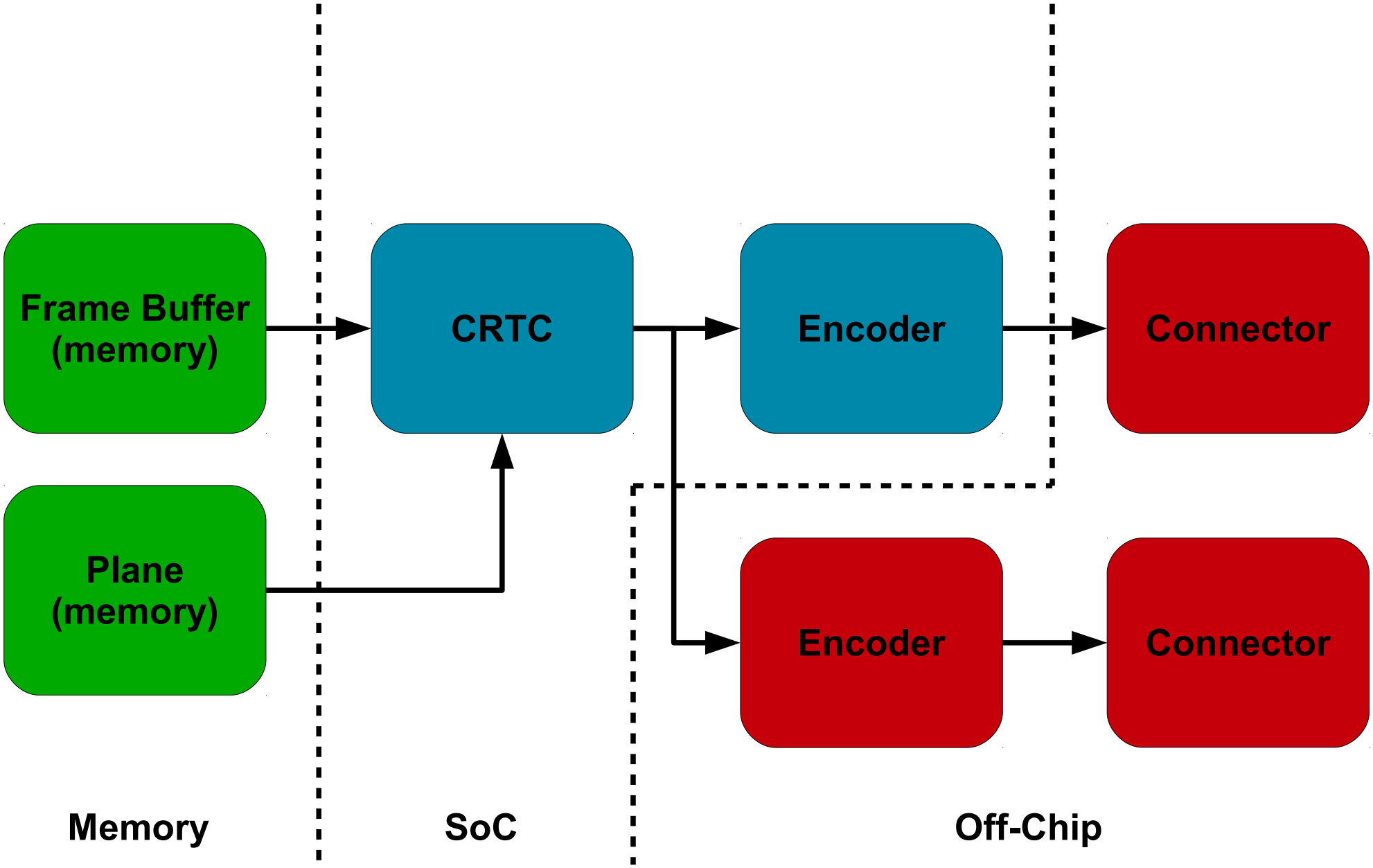


FB vs. DRM - sloccount

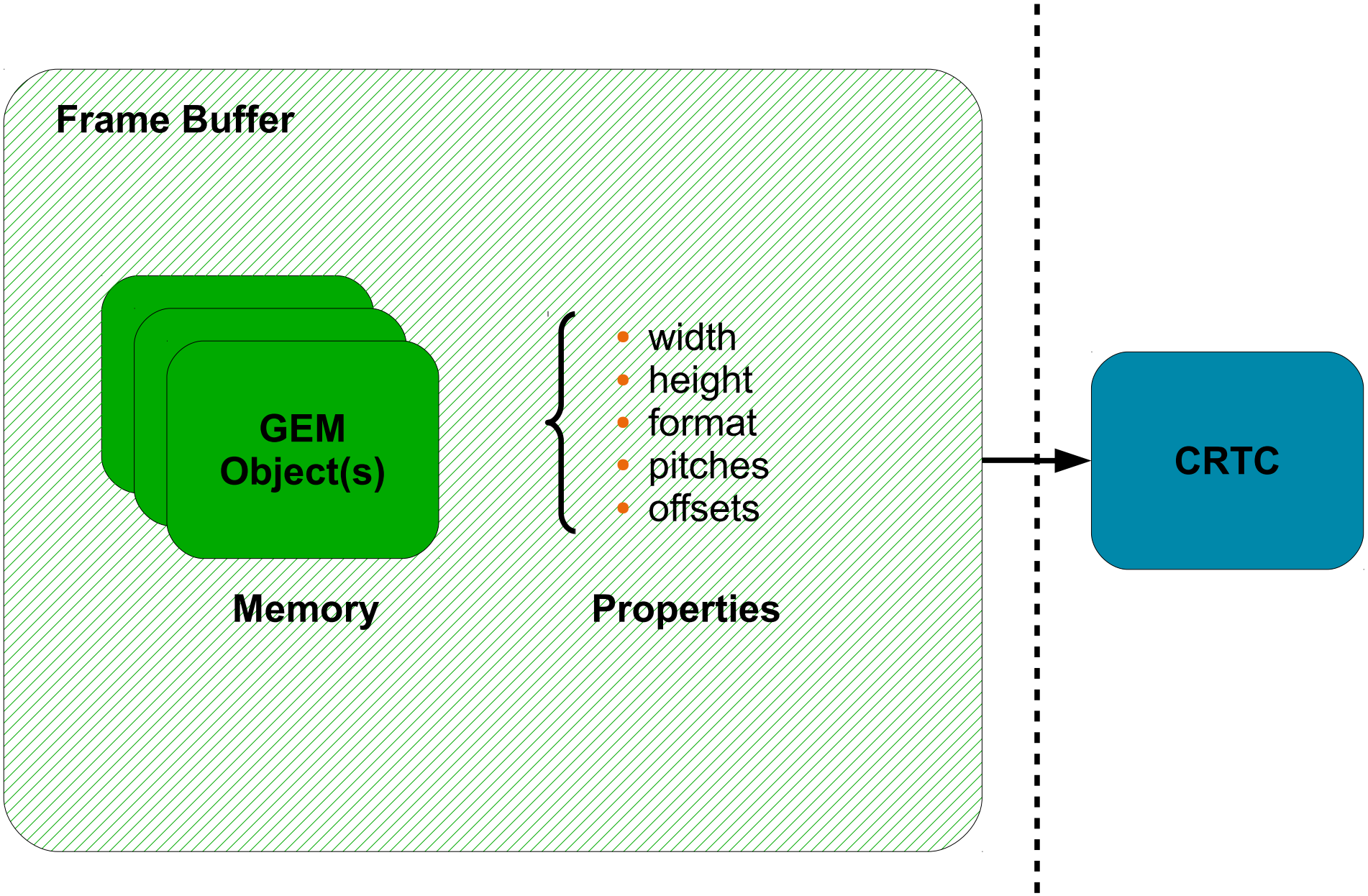

drm
kms
exynos
i810
i915
mga
nouveau
r128
radeon
savage
sis
via


IDEAS
ON BOARD

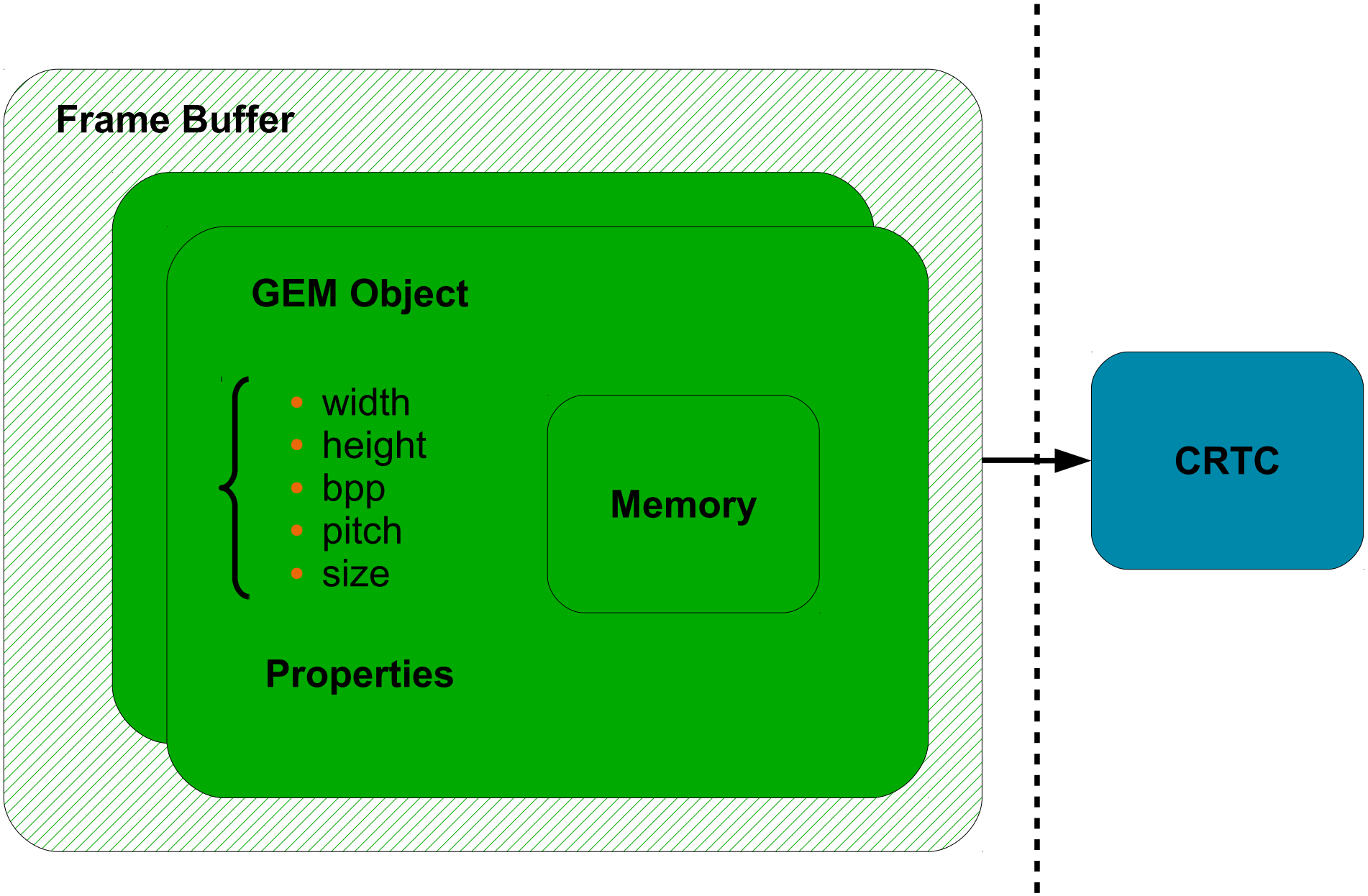
DRM/KMS API



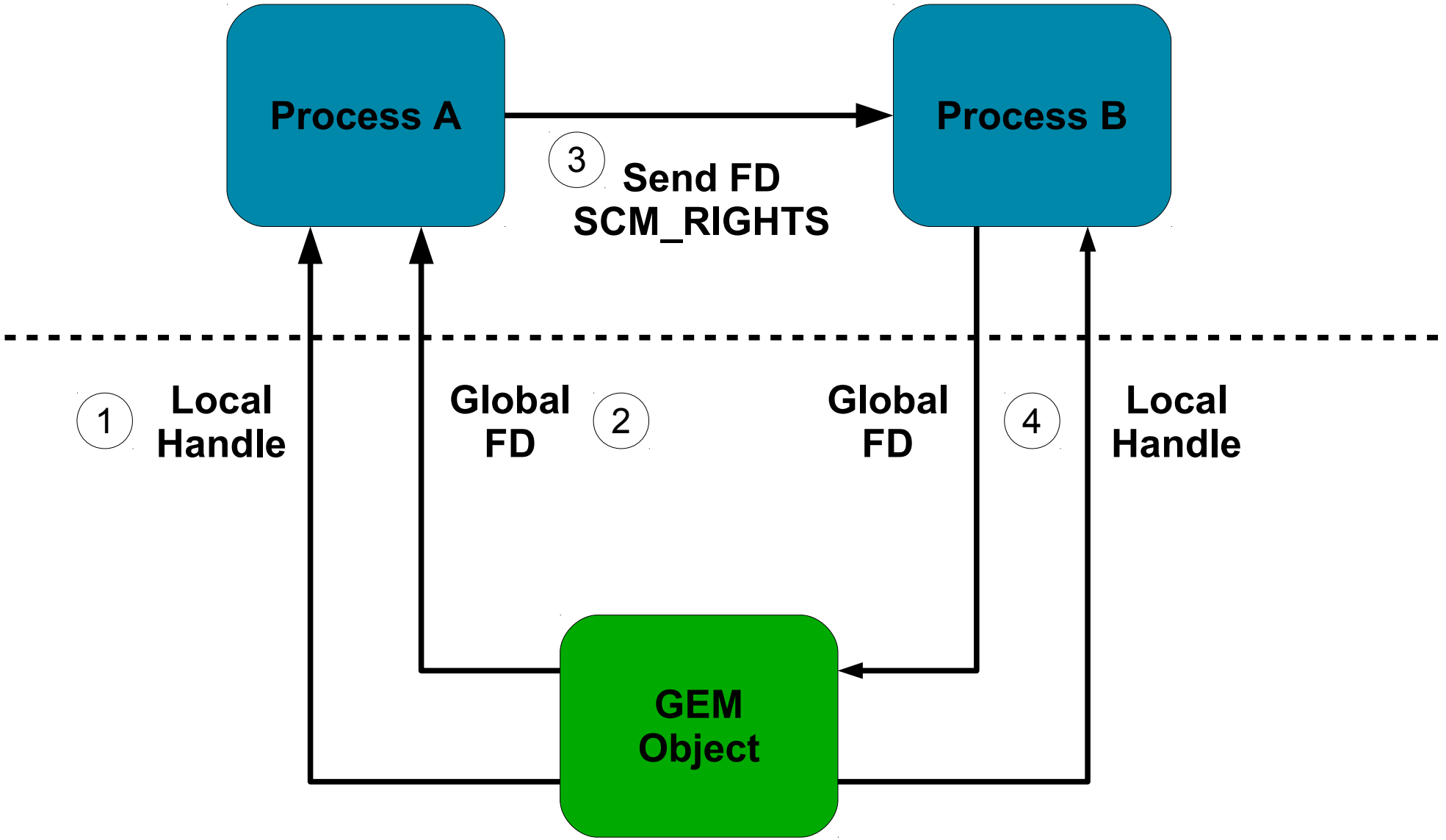
KMS - Device Model



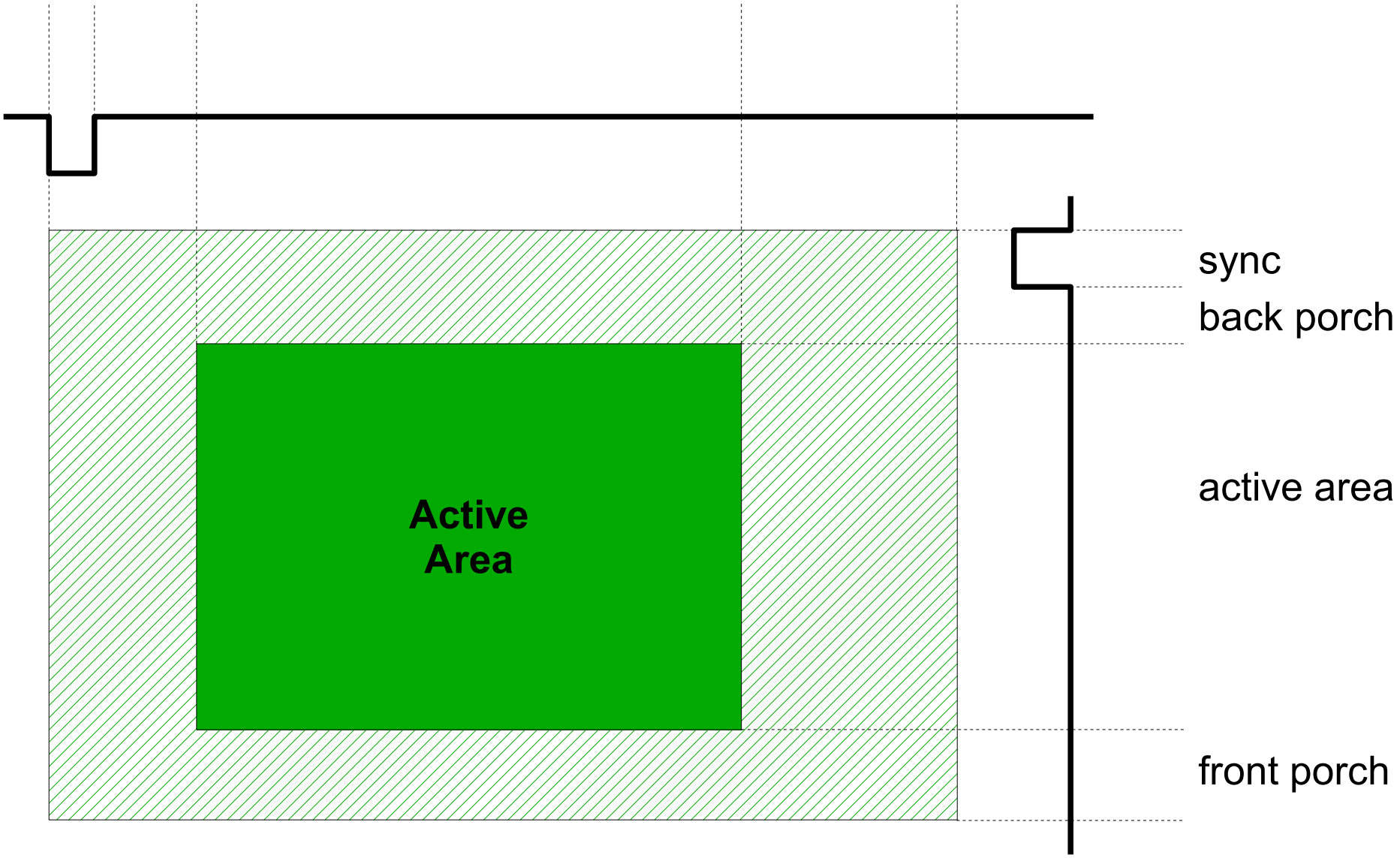
KMS – Frame Buffer



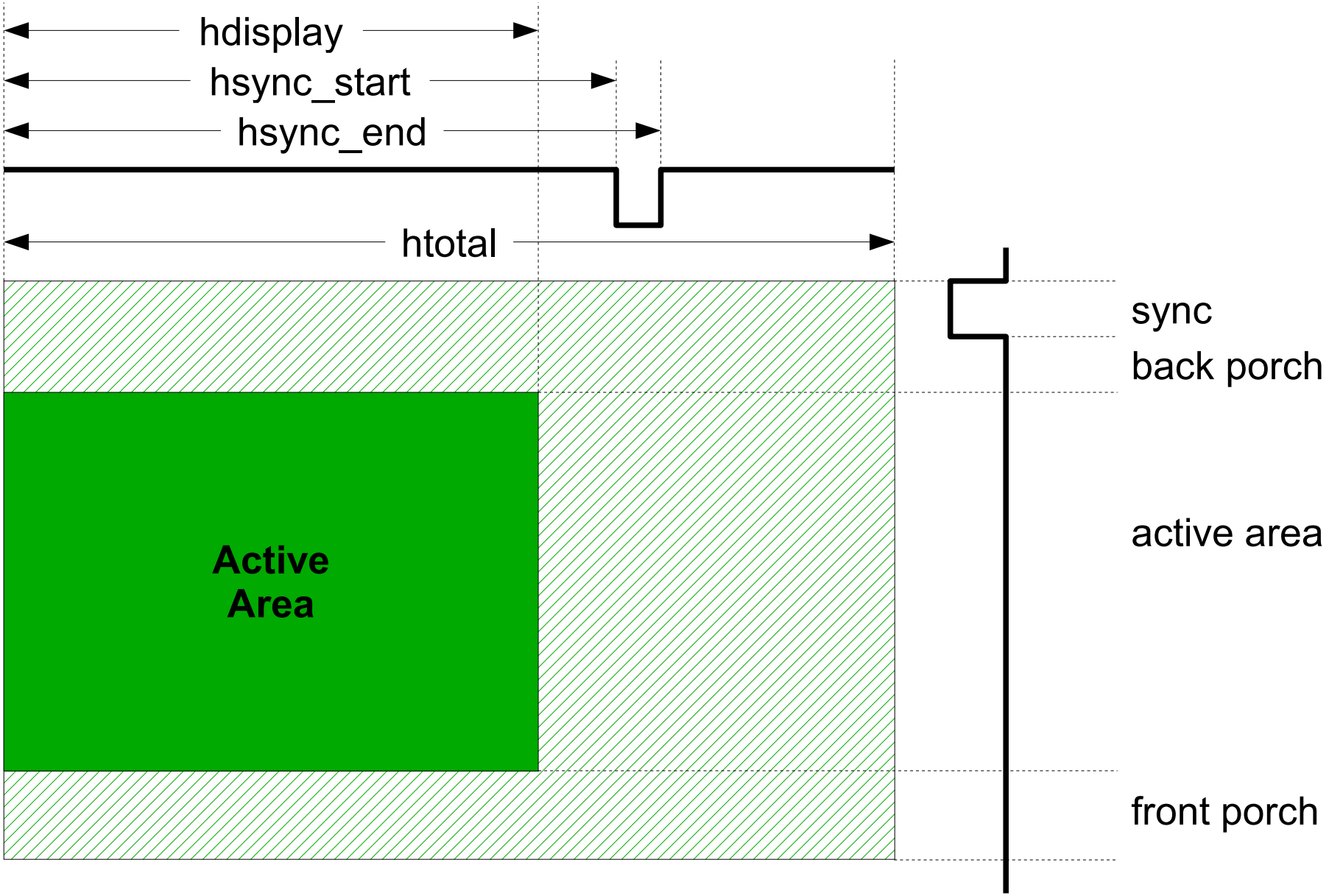
DRM/KMS – GEM Object


Process A
Process B
3
Send FD
SCM_RIGHTS
1
Local
Handle
Global
FD
2
Global
FD
4
Local
Handle
GEM
Object

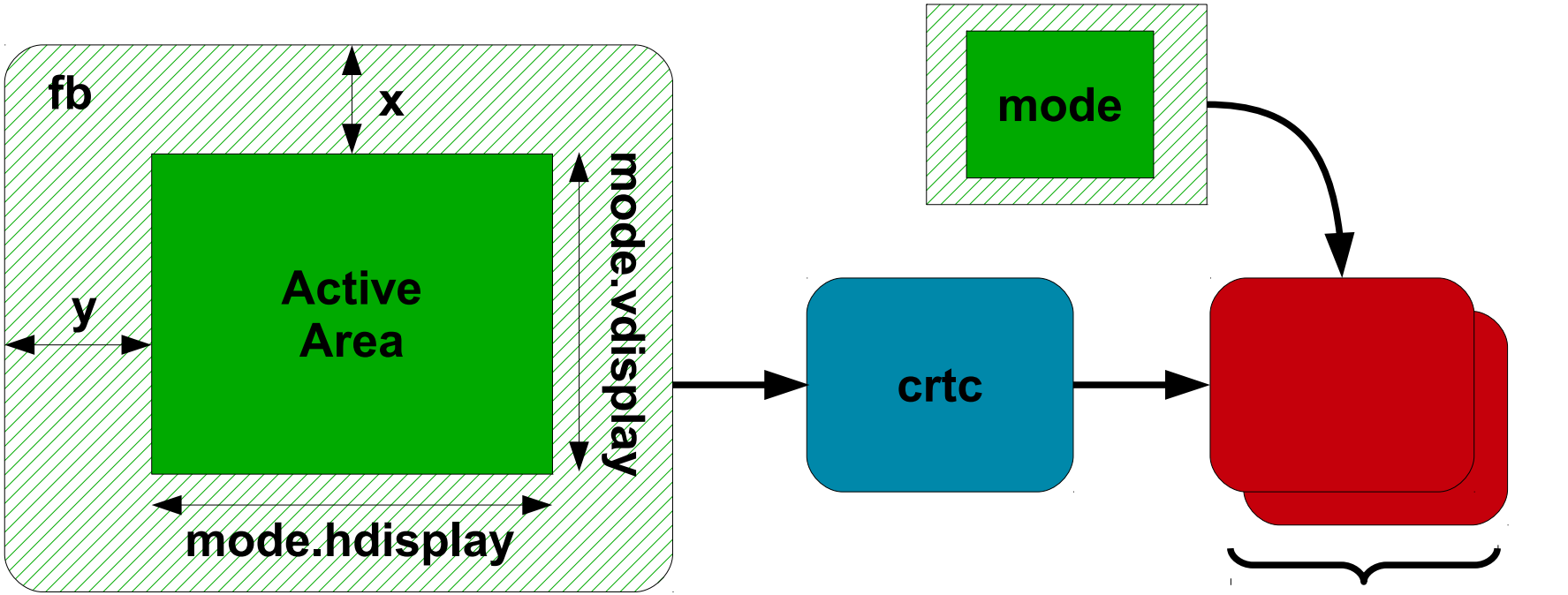
DRM – Handles



KMS – Modes (1/2)


hdisplay
hsync_start
hsync_end
htotal
Active
Area
sync
back porch
active area
front porch

KMS – Modes (2/2)




```
struct drm_mode_set {
    struct drm_framebuffer *fb;
    struct drm_crtc *crtc;
    struct drm_display_mode *mode;
    uint32_t x;
    uint32_t y;
    struct drm_connector **connectors;
    size_t num_connectors;
};
```


KMS – Mode Setting

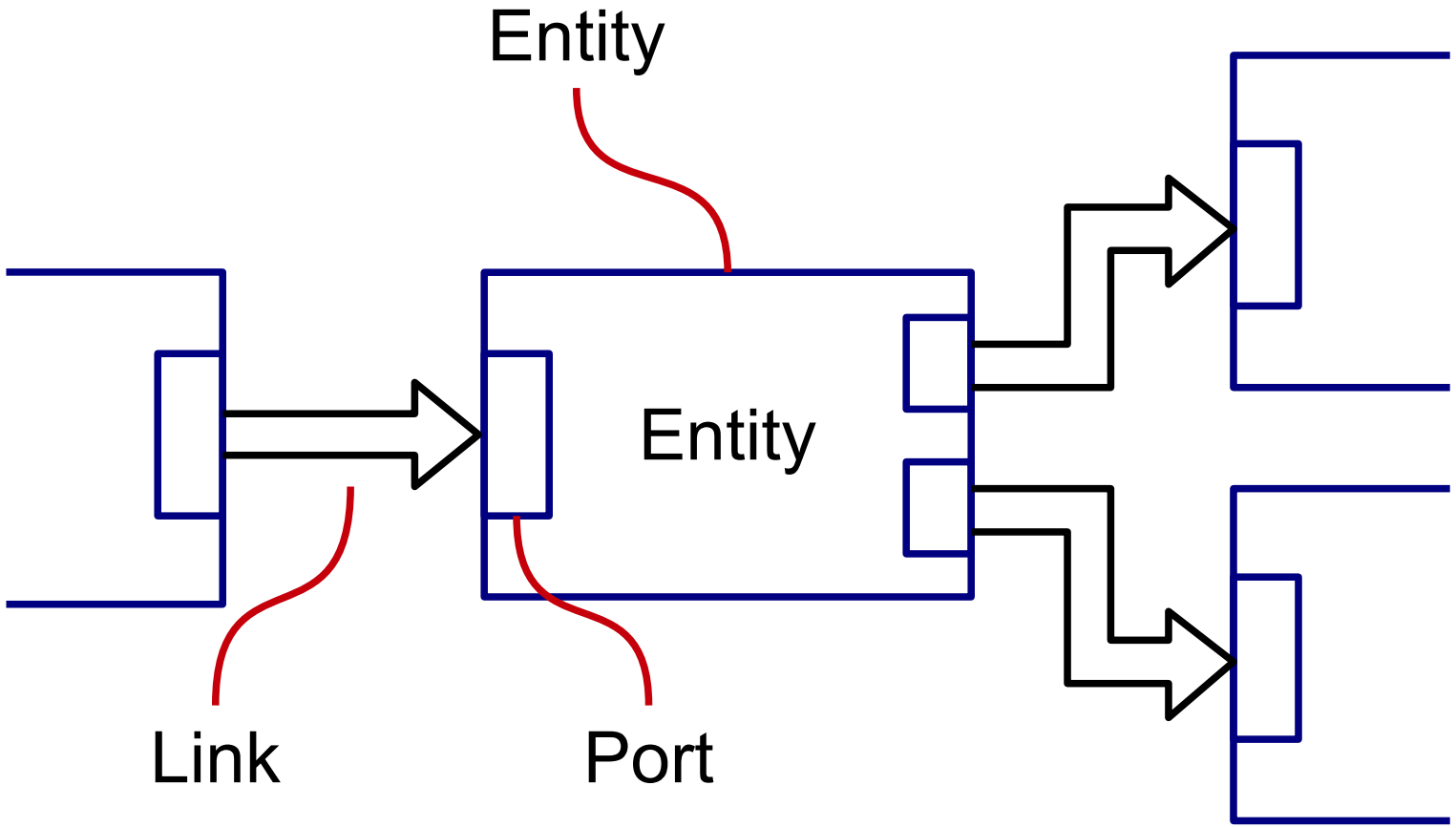
IDEAS ON BOARD

Source: http://www.flickr.com/photos/buckaroobay/3721809183/

Common Display Framework

http://lwn.net/Articles/512363/

WIP – Display Framework

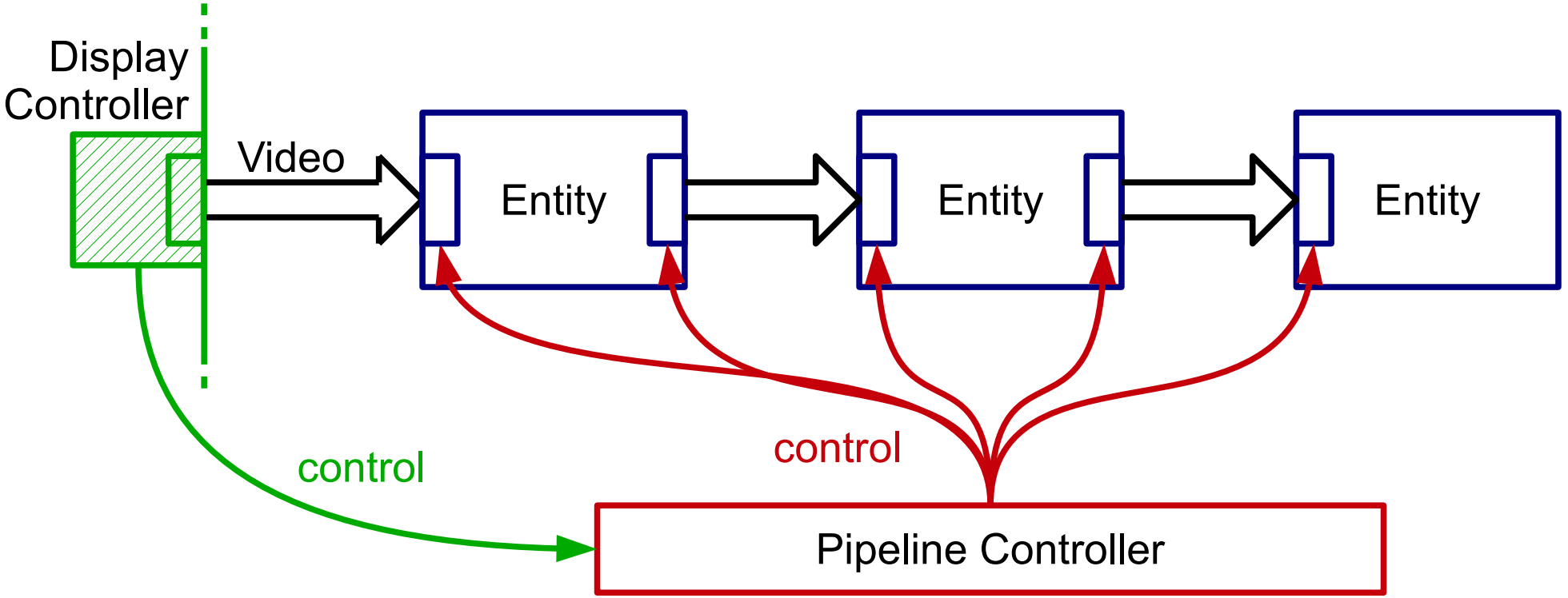


CDF - Entity Model

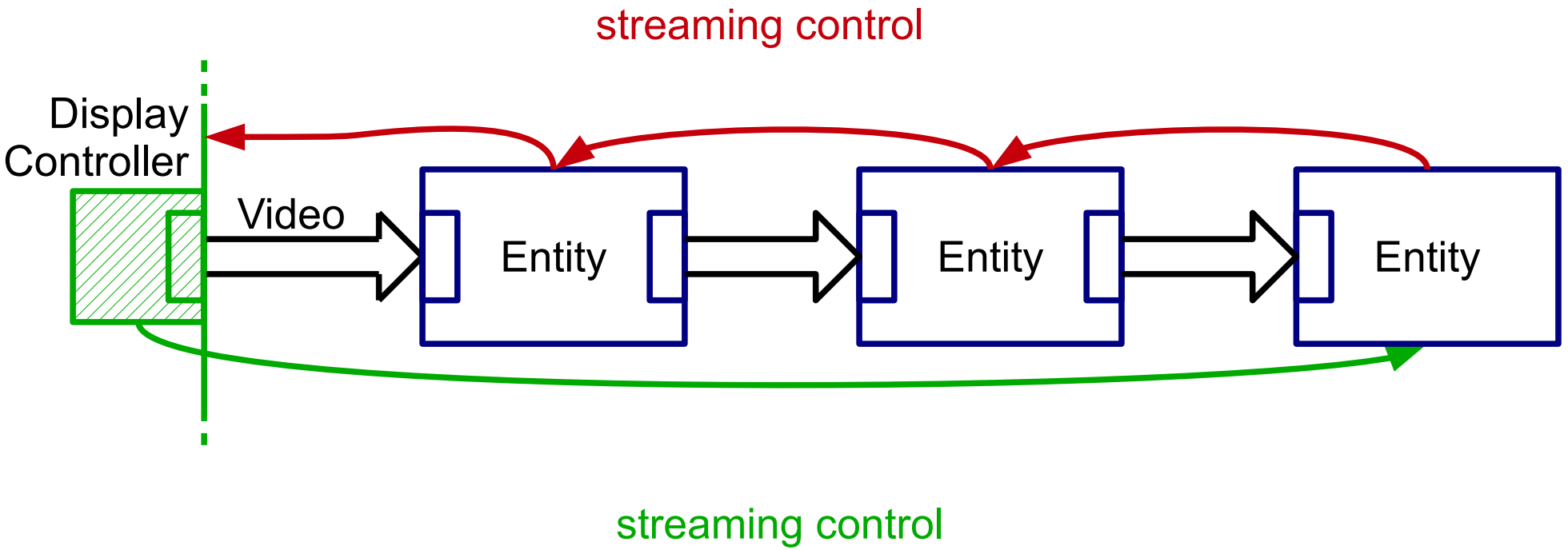
```
hdmi_encoder {
    ports {
        #address-cells = <1>;
        #size-cells = <0>;

        port@0 {
            hdmi_input: endpoint@0 {
                remote = <&display_output>;
            };
        };
        port@1 {
            endpoint@0 { ... };
            endpoint@1 { ... };
        };
    };
};
```

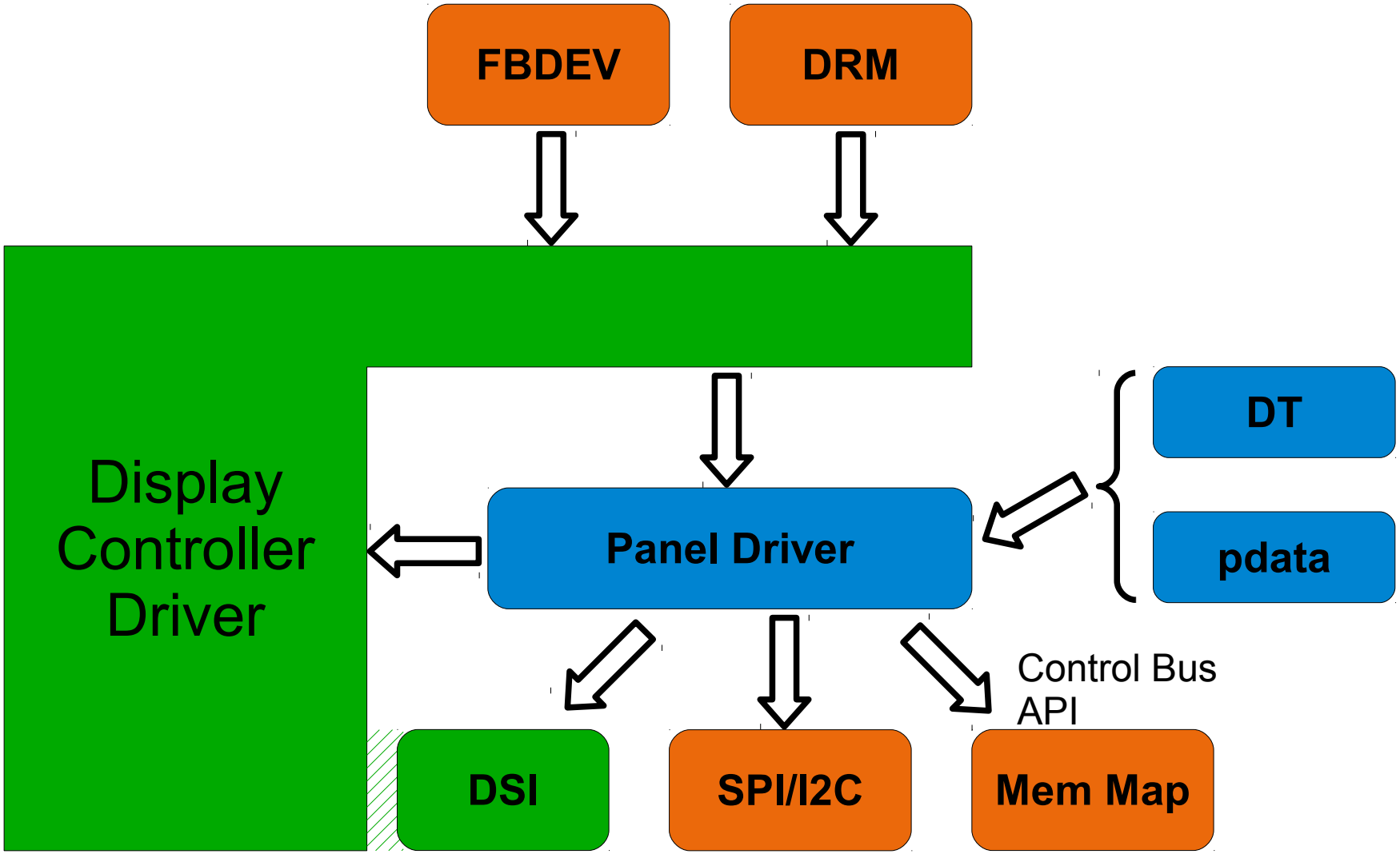
CDF - Device Tree



CDF - Configuration Model



CDF - Streaming Control



CDF - Integration

- dri-devel@listsfreedesktop.org
- ~~linux-fbdev@vger.kernel.org~~
- linux-media@vger.kernel.org

- laurent.pinchart@ideasonboard.com

Contact

merci.