DRM/KMS, FB and V4L2: How to Select a Graphics and Video API

Embedded Linux Conference Europe 2012

Laurent Pinchart
laurent.pinchart@ideasonboard.com
Personal opinion
Flame war possible
Handle with care

Disclaimer
Problem Definition
Problem Definition
display / graphics / video
format
memory / deep pipeline
device / CPU
rotation
scaling
composing
X11
Wayland
DirectFB
Raw API
DRM
FBDEV
V4L2
Origins
Origins – DRM/KMS

- DRM, Linux 2.2.18 (2000)
- GEM (2008)
- KMS, TTM (2009)
- Dumb Buffers, Planes, DMABUF (2011, 2012)
Origins – FBDEV

FBDEV, Linux 1.3.94
Blanking
1996

2000

2012

4CC Formats
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?
<table>
<thead>
<tr>
<th></th>
<th>DRM</th>
<th>FB</th>
<th>V4L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dynamic Allocation</strong></td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Multiple Buffers</strong></td>
<td>Yes</td>
<td>panning</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Import</strong></td>
<td>dmabuf</td>
<td>No</td>
<td>userptr</td>
</tr>
<tr>
<td><strong>Export</strong></td>
<td>dmabuf</td>
<td>mmap</td>
<td>mmap</td>
</tr>
</tbody>
</table>

Memory Management
<table>
<thead>
<tr>
<th></th>
<th>DRM</th>
<th>FB</th>
<th>V4L2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formats</strong></td>
<td>4CC</td>
<td>RGB 4CC</td>
<td>4CC</td>
</tr>
<tr>
<td><strong>Enumeration</strong></td>
<td>Planes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Negotiation</strong></td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Atomicity</strong></td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

**Mode Setting**
<table>
<thead>
<tr>
<th>Transformations</th>
<th>DRM</th>
<th>FB</th>
<th>V4L2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlays</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Rotation</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Scaling</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Cropping/Panning</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
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.
Documentation
Code Size
Cumulative Changes - Drivers
Mailing List Traffic

- drm
- fb
- v4l2
- drm+bug
Device Model – FBDEV

- Frame Buffer (memory)
- Display Device
Device Model – V4L2
Use Cases

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.
FB vs. DRM - sloccount

2370
FB vs. DRM - sloccount

- fbdev: 2370
- drm: 1659
Use Cases - FBDEV
(that's it...)
Video
Everything else
KMS – Device Model

- Frame Buffer (memory)
- Plane (memory)
- CRTC
- Encoder
- Connector

Memory → SoC → Off-Chip
KMS – Frame Buffer

Frame Buffer

GEM Object(s)

Memory

Properties

- width
- height
- format
- pitches
- offsets

CRTC
Frame Buffer

GEM Object

- width
- height
- bpp
- pitch
- size

Memory

CRTC

DRM/KMS – GEM Object
DRM – Handles

1. Local Handle
2. Global FD
3. Send FD SCM_RIGHTS
4. Local Handle
The Future
NEXT EXIT
panel & backlight

http://lwn.net/Articles/512363/
- dri-devel@lists.freedesktop.org
- linux-fbdev@vger.kernel.org
- linux-media@vger.kernel.org
- laurent.pinchart@ideasonboard.com
thx.