Kernel backlight API

Presented by
Hans de Goede
Senior Software Engineer, Red Hat

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported license.
Todays Topics

- 2 problems with the current API
- Adding a backlight property to drm connectors
Causes
Causes

- Windows 8 ready laptops often have broken acpi-video backlight control, so since 3.16 we will use the native backlight interface on these.
- But native interfaces, behave different then firmware interfaces.
- This causes problems for userspace, which expects the kernel to offer a consistent backlight API.
Current API issues
Current API issue 1

- With firmware interfaces 0 means lowest possible backlight settings
- With native interfaces 0 means backlight-off, and sometimes a whole range of values from 0 to x means off.
- Note this has recently been partially fixed for the intel driver with the commit titled:
  “drm/i915: respect the VBT minimum backlight brightness”
Solution 1

- Clearly document in Documentation/ABI/stable/sysfs-class-backlight that a brightness value of 0 means lowest possible brightness, and that only setting bl_power may actually turn the backlight off.

- And file bugs against / fix any drivers not honoring this.
Current API issue 2

- With firmware interfaces the brightness has “perceived brightness” as scale
- With native interfaces the brightness has electrical power (mW) as scale
- The user typically we want to set the perceived brightness, not the electrical power
Solution 2

- Add a new brightness_scale attribute which has a value of either “perceived brightness” or “electrical power” and let userspace further deal with this;

- Or solve this fully in the kernel? I've heard suggestions to map the actual hardware scale to a 0-100 value for userspace, this mapping could include a correction to make the 0-100 a perceived brightness scale
Adding a backlight property to drm connectors
It would be nice to have the backlight level as a property of the connector.

Problem is mapping a backlight interface to a connector.

There are some ideas to let userspace tell the kernel which backlight interface to use (through e.g. udev rules), but otherwise handle this in the kernel.

David Herrmann has already posted patches for this.