New brightness uAPI
Replacing /sys/class/backlight

Hans de Goede
Principal Engineer, Red Hat
Topics

- Problems with /sys/class/backlight
- New uAPI proposal
- Backlight handling technical debt
- Probe ordering issues
Problems with \texttt{/sys/class/backlight}
Long unsolved problem

- First gave a talk about this at XDC 2014
- Intel send a proposal to try and fix this in 2017:
  - https://lore.kernel.org/all/4b17ba08-39f3-57dd-5aad-d37d844b02c6@linux.intel.com/
Problems with current uAPI

- No way to map backlight sysfs device to display
- Often multiple backlight sysfs devices for a single display
- Leaving userspace to guess which one will work
- Brightness control requires root rights
- Meaning of value 0 is undefined
New uAPI proposal
New uAPI proposal

- Add new `display_brightness` and `display_brightness_max` properties on the drm connector object
- `display_brightness_max == 0` means brightness control is not supported
- `display_brightness_max` may change on hotplug events. E.g. plugging in a monitor which supports brightness control over DDC/CI
Backlight handling
technical debt
Technical debt

- Multiple laptop brightness control methods:
  - Directly by the GPU driver
  - Other driver(s) using `$random firmware interface(s)`

- Current approach:
  - Just register sysfs devices for all of them
  - Userspace picks which sysfs device to use based on the type (firmware → platform → native)
  - If kernel heuristics say the native device should be preferred, unregister the others when the native one registers
Probe ordering issues
The new uAPI requires 1 backlight dev (per panel)
The kernel heuristics cannot detect if the GPU driver will offer native control
The heuristics will only pick native when available
This causes the acpi_video backlight device to register before the native driver
To fix this acpi_video backlight registration must be delayed till after the GPU drivers are done probing
Sometimes the heuristics find no known brightness control method.

Then vendor (aka other) is returned.

The vendor (dell-laptop, asus-wmi, etc) drivers typically load much later than the GPU drivers.

This causes there to be no brightness control if userspace looks at the drm connectors early.

Userspace needs hotplug brightness control support for DCC/CI anyways, so the plan is to rely on this.
Questions / Discussion

hdegoede@redhat.com
https://github.com/jwrdegoede/