PCI Express Hotplug

\texttt{pciehp vs acpiphp}

- Originally designed as \textit{alternative} drivers that can bind to the same hardware and register “slots”.
- For “leaf” devices (hint: you can hotplug switches too).
- Ideally, “power on” operation in both should lead to the same thing (but it doesn’t now).
- Analogously for “power off”.
- It should be possible to handle hotplug events from both sources in the same way (\texttt{pciehp} adds some “bells and whistles”). Some BIOSes seem to expect us to do that.
- Currently still mutually exclusive (no technical reason).
ACPI-Based Hotplug Of Devices, CPUs, Memory, . . .

- Device object notifications (bus check, device check, eject).
- Can be generated for any device objects (regardless of the type, methods etc.).
- The handling may depend on device type (e.g. PCI is special), but the basic algorithms are analogous for all devices.
- It would make sense to have one generic ACPI hotplug notify handler for all device objects.
  
1. No duplicated code.
2. Reduced memory footprint.
3. No doubts which devices to install the handler for.

- Rescans resulting from bus check notifications may need to cross bus type boundaries anyway.

- We may need to handle _EJD generically.
PM Settings And Constraints

- What devices can be power-managed (/sys/devices/.../power/control).
- Resume latency tolerance (device PM QoS; needs to be cleaned up to get rid of the notifier mechanism which isn’t really necessary).
- Wakeup settings (device PM QoS flags, /sys/devices/.../power/wakeup). This needs to be consolidated in my opinion.
- Debug attributes (statistics and so on).
- How to describe dependencies between devices (beyond device hierarchy)?
- Do we need more settings?
Device Trees vs ACPI

- Alternative methods of providing the kernel with information needed to handle devices without native enumeration ("platform" devices).
- Device Trees can convey information that has no standard representation in ACPI.
- Some device drivers expect to have such information available (via the of_* interface).
- Those drivers may be needed on systems with ACPI.
Intel is a trademark of Intel Corporation in the U. S. and other countries. *Other names and brands may be claimed as the property of others. Copyright © 2013 Intel Corporation, All rights reserved.