Re: Add option to passively listen for PCIE hotplug events

Source: http://linux.derkeiler.com/Mailing−Lists/Kernel/2008−11/msg00910.html

- *From*: Alan Jenkins <aj504@xxxxxxxxxxxxx>
- *Date*: Tue, 4 Nov 2008 03:29:23 −0800 (PST)

On Nov 4, 5:10 am, Matthew Garrett <mj...@xxxxxxxxxxxxx> wrote:

> On Tue, Nov 04, 2008 at 02:07:00AM +0000, Matthew Garrett wrote:
>
> > On Tue, Nov 04, 2008 at 10:58:11AM +0900, Kenji Kaneshige wrote:
>
> > t_slot−>hpc_ops−>get_adapter_status(t_slot,
> > &value); /* Check if
> > slot is occupied */
> > − if (value && pciehp_force) {
> > + if (value && (pciehp_force ||
> > pciehp_passive)) {
> > rc = pciehp_enable_slot(t_slot);
> > if (rc) /* −ENODEV: shouldn't happen,
> > but deal with it */
> > value = 0;
>
> > This code no longer runs in the pciehp_passive case. However, by the
> > looks of it it still does in the resume case − that probably wants
> > fixing.
>
> Thinking about this − you said that the problem occurs because
> pciehp_force=1 causes it to try to enable an already enabled slot, and
> then tries to power down the slot as a result? It sounds like this code
> should actually be checking whether the return value is ENODEV or
> EINVAL, and in the latter case not powering the slot down. That sounds
> like a separate bugfix that I'll send later on.

I've tested pciehp with this patch on my EeePC, which as you say uses pcie hotplug to allow power savings when the wireless is not needed. Functionally it seems ok.

I see kernel log messages saying "Device already exists, cannot hot−add". I wonder whether this causes the _timing_ problems that I see?

The module takes 2−5 seconds to load (manually after boot, with pciehp_passive=1). Obviously this is not compatible with ambitious boot−time targets. Hot−"remove" works immediately, hot−"add" takes a few seconds, which seems reasonable.

But resuming from suspend to ram can now take 15−20 seconds. It seems the longer suspend time happens with the device "present"; it's about 5 seconds shorter with the device "removed", but still much longer than previously. There's more than one PCIE port, so the rest of the delay could be due to other ports which always have devices "present".

Will it help if I provide a dmesg (with printk.time=1)?

I think removing and re−inserting the module may also increase the delay... ow.

I just tested this and my EeePC failed to resume, just as if the delay had become infinite (or > 180 seconds). At this point the screen is black, not even a cursor... It might be significant that I removed the pciehp module while the device was "removed".

I will try the "don't suspend consoles" option and see if it sheds any light.

Thanks!
Alan
−−