

Re: Flames over -- Re: Which is simpler?

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-02/msg04551.html>

- *From:* Phillip Susi <psusi@xxxxxxxxxx>
 - *Date:* Mon, 13 Feb 2006 11:40:10 -0500
-

Kyle Moffett wrote:

This is true for software suspend, but not for hardware suspend (see the differences now?)
This is why the two are independent and should not be

No, that is not necessarily correct. Sometimes the ACPI bios can leave certain devices in a standby mode so they can wake the machine, but it does not have to, and often does not. Thus when suspended to ram, typically your usb hard drive and almost always your ide/sata/scsi drive will be completely powered off.

mashed together into one "Generic Suspend". Let me bring up the example of my PowerBook again. It's RAM is fully powered right now, running from battery, and it has another couple days of sleep-charge left before I have to worry about plugging it in again. When I open it, the firmware automatically powers up the CPU and other hardware and returns control to the OS. I can `_also_` trigger it to wake by leaving it closed and connecting an external VGA and USB (it wakes every time I connect a USB, but my suspend script puts it to sleep again if it's closed and has no external VGA).

Then your motherboard keeps the bus in a lower power state such that it is capable of causing a wake event when state changes. I'm also fairly sure that when such a wake event happens, there is no indication to the kernel about why it was woken up. Because of that, and the fact that not all systems even support such wake modes, the kernel must reprobe all hardware when it wakes up, and hopefully finds the same devices that were there when it went to sleep. It does this for both types of suspend.

and in either case, there is nothing running on the CPU to monitor device insertion/removal.

You don't need the CPU, just a good USB controller and hubs with low-power modes and such. The fact that plugging in a USB keyboard/mouse and a VGA monitor is enough to wake the system when properly configured should be proof enough.

That is not proof of anything other than the bus controller has the capability of generating a wake event. As I

Re: Flames over -- Re: Which is simpler?

said before, once woken up, the kernel must probe all hardware and if there is a new device, it will find it. The hardware only detected that something changed and thus, generated a wake even, it did not notice exactly what that change was and inform the kernel.

When the system is resumed the kernel decides if the hardware has changed the same way for either system: it probes the hardware to see if it is still there. There isn't anything special that monitors device insertion/removal while suspended to ram.

Sometimes not, but again, it depends on the hardware.

Again, always not since the hardware doesn't actually tell the kernel what happend, it just wakes it up.

Cheers,
Kyle Moffett

--

I have yet to see any problem, however complicated, which, when you looked at it in the right way, did not become still more complicated.

-- Poul Anderson

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>