

[patch 1/1] Hot plug CPU to support physical add of new processors (i386)

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-09/6492.html>

Natalie.Protasevich_at_unisys.com

Date: 09/21/05

To: akpm@osdl.org

Date: Tue, 20 Sep 2005 16:57:39 -0700

The patch allows physical bring-up of new processors (not initially present in the configuration) from facilities such as driver/utility implemented on a platform. The actual method of making processors available is up to the platform implementation.

Signed-off-by: Natalie Protasevich <Natalie.Protasevich@unisys.com>

```
---
 arch/i386/kernel/irq.c      |    8 ++++----
 arch/i386/kernel/mpparse.c |    6 +++---
 arch/i386/kernel/smpboot.c |    4 ++++
 3 files changed, 11 insertions(+), 7 deletions(-)
diff -puN arch/i386/kernel/mpparse.c-hotcpu-i386 arch/i386/kernel/mpparse.c
--- linux-2.6.14-rc1-mm1/arch/i386/kernel/mpparse.c-hotcpu-i386 2005-09-20 16:48:12.078408952 -07
+++ linux-2.6.14-rc1-mm1-root/arch/i386/kernel/mpparse.c        2005-09-20 16:50:09.516555640 -07
@@ -70,7 +70,7 @@ unsigned int def_to_bigsm = 0;
 /* Processor that is doing the boot up */
 unsigned int boot_cpu_physical_apicid = -1U;
 /* Internal processor count */
-static unsigned int __initdata num_processors;
+static unsigned int __devinitdata num_processors;

 /* Bitmask of physically existing CPUs */
 physid_mask_t phys_cpu_present_map;
@@ -120,7 +120,7 @@ static int MP_valid_apicid(int apicid, i
 }
 #endif

-static void __init MP_processor_info (struct mpc_config_processor *m)
+static void __devinit MP_processor_info (struct mpc_config_processor *m)
 {
     int ver, apicid;
     physid_mask_t phys_cpu;
@@ -835,7 +835,7 @@ void __init mp_register_lapic_address (
 }

-void __init mp_register_lapic (
+void __devinit mp_register_lapic (
     u8 id,
     u8 enabled)
 {
```

Linux-Kernel: [patch 1/1] Hot plug CPU to support physical add of new processors (i386)

```
diff -puN arch/i386/kernel/smpboot.c~hotcpu-i386 arch/i386/kernel/smpboot.c
--- linux-2.6.14-rc1-mm1/arch/i386/kernel/smpboot.c~hotcpu-i386 2005-09-20 16:48:12.113403632 -07
+++ linux-2.6.14-rc1-mm1-root/arch/i386/kernel/smpboot.c 2005-09-20 16:51:55.422455496 -07
@@ -88,7 +88,11 @@ EXPORT_SYMBOL(cpu_online_map);
 cpumask_t cpu_callin_map;
 cpumask_t cpu_callout_map;
 EXPORT_SYMBOL(cpu_callout_map);
+#ifdef CONFIG_HOTPLUG_CPU
+cpumask_t cpu_possible_map = CPU_MASK_ALL;
+#else
 cpumask_t cpu_possible_map;
+#endif
 EXPORT_SYMBOL(cpu_possible_map);
 static cpumask_t smp_commenced_mask;

diff -puN arch/i386/kernel/irq.c~hotcpu-i386 arch/i386/kernel/irq.c
--- linux-2.6.14-rc1-mm1/arch/i386/kernel/irq.c~hotcpu-i386 2005-09-20 16:48:12.148398312 -07
+++ linux-2.6.14-rc1-mm1-root/arch/i386/kernel/irq.c 2005-09-20 16:52:29.668249344 -0700
@@ -220,7 +220,7 @@ int show_interrupts(struct seq_file *p,

     if (i == 0) {
         seq_printf(p, "          ");
-        for_each_cpu(j)
+        for_each_online_cpu(j)
             seq_printf(p, "CPU%d          ", j);
         seq_putc(p, '\n');
     }
@@ -234,7 +234,7 @@ int show_interrupts(struct seq_file *p,
 #ifndef CONFIG_SMP
     seq_printf(p, "%10u ", kstat_irqs(i));
 #else
-    for_each_cpu(j)
+    for_each_online_cpu(j)
         seq_printf(p, "%10u ", kstat_cpu(j).irqs[i]);
 #endif
     seq_printf(p, " %14s", irq_desc[i].handler->typename);
@@ -248,12 +248,12 @@ skip:
     spin_unlock_irqrestore(&irq_desc[i].lock, flags);
 } else if (i == NR_IRQS) {
     seq_printf(p, "NMI: ");
-    for_each_cpu(j)
+    for_each_online_cpu(j)
         seq_printf(p, "%10u ", nmi_count(j));
     seq_putc(p, '\n');
 #ifdef CONFIG_X86_LOCAL_APIC
     seq_printf(p, "LOC: ");
-    for_each_cpu(j)
+    for_each_online_cpu(j)
         seq_printf(p, "%10u ",
             per_cpu(irq_stat, j).apic_timer_irqs);
     seq_putc(p, '\n');
-
-

```

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>