Subject: [PATCH 1/2] Consolidate bust_spinlocks()
Posted by adobriyan on Thu, 18 Jan 2007 11:09:39 GMT
View Forum Message <> Reply to Message

From: Kirill Korotaev <dev@sw.ru>

Part of long forgotten patch
 http://groups.google.com/group/fa.linux.kernel/msg/e98e941ce 1cf29f6?dmode=source
Since then, m32r grabbed two copies.

Signed-off-by: Alexey Dobriyan <adobriyan@openvz.org>
---

 arch/i386/mm/fault.c | 26 ----------
 arch/ia64/kernel/traps.c | 30 ----------
 arch/m32r/mm/fault-nommu.c | 26 ----------
 arch/m32r/mm/fault.c | 26 ----------
 arch/s390/mm/fault.c | 26 ----------
 arch/x86_64/mm/fault.c | 21 -------
 lib/Makefile | 4 ++--
 lib/bust_spinlocks.c | 2 +-
 8 files changed, 3 insertions(+), 158 deletions(-)

--- a/arch/i386/mm/fault.c
+++ b/arch/i386/mm/fault.c
@@ -60,32 +60,6 @@ static inline int notify_page_fault(enum
 }

 /*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
- oops_in_progress = 1;
- return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
- * OK, the message is on the console. Now we call printk()
- * without oops_in_progress set so that printk will give klogd
- * a poke. Hold onto your hats...

```
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
-/*
  * Return EIP plus the CS segment base.  The segment limit is also
  * adjusted, clamped to the kernel/user address space (whichever is
  * appropriate), and returned in *eip_limit.
--- a/arch/ia64/kernel/traps.c
+++ b/arch/ia64/kernel/traps.c
@@ -24,8 +24,6 @@ #include <asm/processor.h>
 #include <asm/uaccess.h>
 #include <asm/kdebug.h>

-extern spinlock_t timerlist_lock;
-
 fpswa_interface_t *fpswa_interface;
 EXPORT_SYMBOL(fpswa_interface);

@@ -53,34 +51,6 @@ trap_init (void)
   fpswa_interface = __va(ia64_boot_param->fpswa);
 }

-/*
- * Unlock any spinlocks which will prevent us from getting the message out (timerlist_lock
- * is acquired through the console unblank code)
- */
-void
-bust_spinlocks (int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk() without
-  * oops_in_progress set so that printk will give klogd a poke.  Hold onto
-  * your hats...
-  */
```

```
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
 void
 die (const char *str, struct pt_regs *regs, long err)
 {
--- a/arch/m32r/mm/fault-nommu.c
+++ b/arch/m32r/mm/fault-nommu.c
@@ -46,32 +46,6 @@ #define tlb_entry_i tlb_entry_i_dat[smp_
 #define tlb_entry_d tlb_entry_d_dat[smp_processor_id()]
 #endif

-/*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk()
-  * without oops_in_progress set so that printk will give klogd
-  * a poke.  Hold onto your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
 void do_BUG(const char *file, int line)
 {
  bust_spinlocks(1);
--- a/arch/m32r/mm/fault.c
+++ b/arch/m32r/mm/fault.c
@@ -49,32 +49,6 @@ #endif

 extern void init_tlb(void);
```

```
-/*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk()
-  * without oops_in_progress set so that printk will give klogd
-  * a poke.  Hold onto your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
  /*=============================================================== ============*
  * do_page_fault()
   *=============================================================== ===========*
--- a/arch/s390/mm/fault.c
+++ b/arch/s390/mm/fault.c
@@ -83,32 +83,6 @@ static inline int notify_page_fault(enum
 }
 #endif

-extern spinlock_t timerlist_lock;
-
-/*
- * Unlock any spinlocks which will prevent us from getting the
- * message out (timerlist_lock is acquired through the
- * console unblank code)
- */
-void bust_spinlocks(int yes)
-{
- if (yes) {
-  oops_in_progress = 1;
- } else {
```

```
-  int loglevel_save = console_loglevel;
-  console_unblank();
-  oops_in_progress = 0;
-  /*
-   * OK, the message is on the console.  Now we call printk()
-   * without oops_in_progress set so that printk will give klogd
-   * a poke.  Hold onto your hats...
-   */
-  console_loglevel = 15;
-  printk(" ");
-  console_loglevel = loglevel_save;
- }
-}
-
 /*
  * Check which address space is addressed by the access
  * register in S390_lowcore.exc_access_id.
--- a/arch/x86_64/mm/fault.c
+++ b/arch/x86_64/mm/fault.c
@@ -69,27 +69,6 @@ static inline int notify_page_fault(enum
  return atomic_notifier_call_chain(&notify_page_fault_chain, val, &args);
 }

-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
- if (yes) {
-  oops_in_progress = 1;
- } else {
-#ifdef CONFIG_VT
-  unblank_screen();
-#endif
-  oops_in_progress = 0;
-  /*
-   * OK, the message is on the console.  Now we call printk()
-   * without oops_in_progress set so that printk will give klogd
-   * a poke.  Hold onto your hats...
-   */
-  console_loglevel = 15;  /* NMI oopser may have shut the console up */
-  printk(" ");
-  console_loglevel = loglevel_save;
- }
-}
-
 /* Sometimes the CPU reports invalid exceptions on prefetch.
    Check that here and ignore.
    Opcode checker based on code by Richard Brunner */
--- a/lib/Makefile
```

```
+++ b/lib/Makefile
@@ -3,7 +3,7 @@ # Makefile for some libs needed in the k
 #

 lib-y := ctype.o string.o vsprintf.o cmdline.o \
-  bust_spinlocks.o rbtree.o radix-tree.o dump_stack.o \
+  rbtree.o radix-tree.o dump_stack.o \
    idr.o div64.o int_sqrt.o bitmap.o extable.o prio_tree.o \
    sha1.o irq_regs.o reciprocal_div.o

@@ -12,7 +12,7 @@ lib-$(CONFIG_SMP) += cpumask.o

 lib-y += kobject.o kref.o kobject_uevent.o klist.o

-obj-y += sort.o parser.o halfmd4.o debug_locks.o random32.o
+obj-y += sort.o parser.o halfmd4.o debug_locks.o random32.o bust_spinlocks.o

 ifeq ($(CONFIG_DEBUG_KOBJECT),y)
 CFLAGS_kobject.o += -DDEBUG
--- a/lib/bust_spinlocks.c
+++ b/lib/bust_spinlocks.c
@@ -14,7 +14,7 @@ #include <linux/wait.h>
 #include <linux/vt_kern.h>


-void bust_spinlocks(int yes)
+void __attribute__((weak)) bust_spinlocks(int yes)
 {
  if (yes) {
   oops_in_progress = 1;
```

Subject: Re: [PATCH 1/2] Consolidate bust_spinlocks()
Posted by Martin Schwidefsky on Thu, 18 Jan 2007 11:39:25 GMT
View Forum Message <> Reply to Message

On Thu, 2007-01-18 at 14:16 +0300, Alexey Dobriyan wrote:
> From: Kirill Korotaev <dev@sw.ru>
>
> Part of long forgotten patch
> http://groups.google.com/group/fa.linux.kernel/msg/e98e941ce 1cf29f6?dmode=source
> Since then, m32r grabbed two copies.
>
> Signed-off-by: Alexey Dobriyan <adobriyan@openvz.org>
> ---
>
> arch/i386/mm/fault.c | 26 --------------------------
> arch/ia64/kernel/traps.c | 30 ------------------------------

> arch/m32r/mm/fault-nommu.c | 26 ----------
> arch/m32r/mm/fault.c | 26 ----------
> arch/s390/mm/fault.c | 26 ----------
> arch/x86_64/mm/fault.c | 21 ----------
> lib/Makefile | 4 ++--
> lib/bust_spinlocks.c | 2 +-
> 8 files changed, 3 insertions(+), 158 deletions(-)

NACK for the s390 part. lib/bust_spinlocks.c does an unblank_screen if
CONFIG_VT is defined. That is not good enough for s390 because we do not
have CONFIG_VT nor unblank_screen but still require that console_unblank
is called.

--
blue skies,
Martin.

Martin Schwidefsky
Linux for zSeries Development & Services
IBM Deutschland Entwicklung GmbH

"Reality continues to ruin my life." - Calvin.

Subject: Re: [PATCH 1/2] Consolidate bust_spinlocks()
Posted by adobriyan on Thu, 18 Jan 2007 16:23:04 GMT
View Forum Message <> Reply to Message

On Thu, Jan 18, 2007 at 12:39:25PM +0100, Martin Schwidefsky wrote:
> NACK for the s390 part. lib/bust_spinlocks.c does an unblank_screen if
> CONFIG_VT is defined. That is not good enough for s390 because we do not
> have CONFIG_VT nor unblank_screen but still require that console_unblank
> is called.

Martin, are you OK with comments tweaking in s390 code?

[PATCH 1/2] Consolidate bust_spinlocks()

From: Kirill Korotaev <dev@sw.ru>

Part of long forgotten patch
http://groups.google.com/group/fa.linux.kernel/msg/e98e941ce 1cf29f6?dmode=source
Since then, m32r grabbed two copies.

Leave s390 copy because of important absence of CONFIG_VT, but remove
references to non-existent timerlist_lock. ia64 also loses timerlist_lock.

Signed-off-by: Alexey Dobriyan <adobriyan@openvz.org>

```
---

 arch/i386/mm/fault.c       |   26 -------------------------
 arch/ia64/kernel/traps.c   |   30 ------------------------------
 arch/m32r/mm/fault-nommu.c |   26 -------------------------
 arch/m32r/mm/fault.c       |   26 -------------------------
 arch/s390/mm/fault.c       |    4 +---
 arch/x86_64/mm/fault.c     |   21 ---------------------
 lib/Makefile               |    4 ++--
 lib/bust_spinlocks.c       |    2 +-
 8 files changed, 4 insertions(+), 135 deletions(-)

--- a/arch/i386/mm/fault.c
+++ b/arch/i386/mm/fault.c
@@ -60,32 +60,6 @@ static inline int notify_page_fault(enum
 }

 /*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk()
-  * without oops_in_progress set so that printk will give klogd
-  * a poke.  Hold onto your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
-/*
  * Return EIP plus the CS segment base.  The segment limit is also
  * adjusted, clamped to the kernel/user address space (whichever is
  * appropriate), and returned in *eip_limit.
--- a/arch/ia64/kernel/traps.c
```

```
+++ b/arch/ia64/kernel/traps.c
@@ -24,8 +24,6 @@ #include <asm/processor.h>
 #include <asm/uaccess.h>
 #include <asm/kdebug.h>

-extern spinlock_t timerlist_lock;
-
 fpswa_interface_t *fpswa_interface;
 EXPORT_SYMBOL(fpswa_interface);

@@ -53,34 +51,6 @@ trap_init (void)
   fpswa_interface = __va(ia64_boot_param->fpswa);
 }

-/*
- * Unlock any spinlocks which will prevent us from getting the message out (timerlist_lock
- * is acquired through the console unblank code)
- */
-void
-bust_spinlocks (int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk() without
-  * oops_in_progress set so that printk will give klogd a poke.  Hold onto
-  * your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
 void
 die (const char *str, struct pt_regs *regs, long err)
 {
--- a/arch/m32r/mm/fault-nommu.c
+++ b/arch/m32r/mm/fault-nommu.c
@@ -46,32 +46,6 @@ #define tlb_entry_i tlb_entry_i_dat[smp_
```

```
 #define tlb_entry_d tlb_entry_d_dat[smp_processor_id()]
 #endif

-/*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
-  return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk()
-  * without oops_in_progress set so that printk will give klogd
-  * a poke.  Hold onto your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
 void do_BUG(const char *file, int line)
 {
  bust_spinlocks(1);
--- a/arch/m32r/mm/fault.c
+++ b/arch/m32r/mm/fault.c
@@ -49,32 +49,6 @@ #endif

 extern void init_tlb(void);

-/*
- * Unlock any spinlocks which will prevent us from getting the
- * message out
- */
-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
-
- if (yes) {
-  oops_in_progress = 1;
```

```
-  return;
- }
-#ifdef CONFIG_VT
- unblank_screen();
-#endif
- oops_in_progress = 0;
- /*
-  * OK, the message is on the console.  Now we call printk()
-  * without oops_in_progress set so that printk will give klogd
-  * a poke.  Hold onto your hats...
-  */
- console_loglevel = 15;  /* NMI oopser may have shut the console up */
- printk(" ");
- console_loglevel = loglevel_save;
-}
-
  /*=============================================================== ============*
  * do_page_fault()
   *=============================================================== ===========*
--- a/arch/s390/mm/fault.c
+++ b/arch/s390/mm/fault.c
@@ -83,12 +83,10 @@ static inline int notify_page_fault(enum
 }
 #endif

-extern spinlock_t timerlist_lock;

 /*
  * Unlock any spinlocks which will prevent us from getting the
- * message out (timerlist_lock is acquired through the
- * console unblank code)
+ * message out.
  */
 void bust_spinlocks(int yes)
 {
--- a/arch/x86_64/mm/fault.c
+++ b/arch/x86_64/mm/fault.c
@@ -69,27 +69,6 @@ static inline int notify_page_fault(enum
  return atomic_notifier_call_chain(&notify_page_fault_chain, val, &args);
 }

-void bust_spinlocks(int yes)
-{
- int loglevel_save = console_loglevel;
- if (yes) {
-  oops_in_progress = 1;
- } else {
-#ifdef CONFIG_VT
```

```
-  unblank_screen();
-#endif
-  oops_in_progress = 0;
-  /*
-   * OK, the message is on the console.  Now we call printk()
-   * without oops_in_progress set so that printk will give klogd
-   * a poke.  Hold onto your hats...
-   */
-  console_loglevel = 15;  /* NMI oopser may have shut the console up */
-  printk(" ");
-  console_loglevel = loglevel_save;
- }
-}
-
 /* Sometimes the CPU reports invalid exceptions on prefetch.
    Check that here and ignore.
    Opcode checker based on code by Richard Brunner */
--- a/lib/Makefile
+++ b/lib/Makefile
@@ -3,7 +3,7 @@ # Makefile for some libs needed in the k
 #

 lib-y := ctype.o string.o vsprintf.o cmdline.o \
-  bust_spinlocks.o rbtree.o radix-tree.o dump_stack.o \
+  rbtree.o radix-tree.o dump_stack.o \
   idr.o div64.o int_sqrt.o bitmap.o extable.o prio_tree.o \
   sha1.o irq_regs.o reciprocal_div.o

@@ -12,7 +12,7 @@ lib-$(CONFIG_SMP) += cpumask.o

 lib-y += kobject.o kref.o kobject_uevent.o klist.o

-obj-y += sort.o parser.o halfmd4.o debug_locks.o random32.o
+obj-y += sort.o parser.o halfmd4.o debug_locks.o random32.o bust_spinlocks.o

 ifeq ($(CONFIG_DEBUG_KOBJECT),y)
 CFLAGS_kobject.o += -DDEBUG
--- a/lib/bust_spinlocks.c
+++ b/lib/bust_spinlocks.c
@@ -14,7 +14,7 @@ #include <linux/wait.h>
 #include <linux/vt_kern.h>


-void bust_spinlocks(int yes)
+void __attribute__((weak)) bust_spinlocks(int yes)
 {
  if (yes) {
   oops_in_progress = 1;
```

Subject: Re: [PATCH 1/2] Consolidate bust_spinlocks()
Posted by Martin Schwidefsky on Fri, 19 Jan 2007 09:27:31 GMT
View Forum Message <> Reply to Message

On Thu, 2007-01-18 at 19:26 +0300, Alexey Dobriyan wrote:
> Martin, are you OK with comments tweaking in s390 code?
> ---------------------------------------
> [PATCH 1/2] Consolidate bust_spinlocks()

Yes, this should be fine now.

--
blue skies,
  Martin.

Martin Schwidefsky
Linux for zSeries Development & Services
IBM Deutschland Entwicklung GmbH

"Reality continues to ruin my life." - Calvin.