

# [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2004-07/6301.html>

---

**From:** Matt Mackall (*mpm\_at\_selenic.com*)

**Date:** 07/31/04

Date: Fri, 30 Jul 2004 17:05:59 -0500

To: linux-kernel <linux-kernel@vger.kernel.org>, Andrew Morton <akpm@osdl.org>

This moves duplicate BUG, PAGE\_BUG, BUG\_ON, and WARN\_ON code to asm-generic and makes them slightly more consistent. This cleanup is also preparatory work for making BUG and WARN verbosity configurable.

Signed-off-by: Matt Mackall <mpm@selenic.com>

Index: mm/include/asm-alpha/bug.h

```
=====
--- mm.orig/include/asm-alpha/bug.h 2004-04-03 21:37:59.000000000 -0600
+++ mm/include/asm-alpha/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -9,15 +9,7 @@
 __asm__ __volatile__ ("call_pal %0 # bugchk\n\t".long %1\n\t.8byte %2" \
    :: "i" (PAL_bugchk), "i"(__LINE__), "i"(__FILE__))

-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
-#define PAGE_BUG(page) BUG()
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
- } while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>
```

#endif

Index: mm/include/asm-arm/bug.h

```
=====
--- mm.orig/include/asm-arm/bug.h 2004-04-03 21:36:15.000000000 -0600
+++ mm/include/asm-arm/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -8,23 +8,15 @@

/* give file/line information */
```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
#define BUG() __bug(__FILE__, __LINE__, NULL)
-#define PAGE_BUG(page) __bug(__FILE__, __LINE__, page)

#else

-/* these just cause an oops */
+/* this just causes an oops */
#define BUG() (*(int *)0 = 0)
-#define PAGE_BUG(page) (*(int *)0 = 0)

#endif

-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>
```

#endif

Index: mm/include/asm-arm26/bug.h

```
-----
--- mm.orig/include/asm-arm26/bug.h 2004-04-03 21:38:21.000000000 -0600
+++ mm/include/asm-arm26/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -5,26 +5,13 @@
```

```
#ifdef CONFIG_DEBUG_BUGVERBOSE
extern volatile void __bug(const char *file, int line, void *data);
-
-/* give file/line information */
#define BUG() __bug(__FILE__, __LINE__, NULL)
-#define PAGE_BUG(page) __bug(__FILE__, __LINE__, page)
-
#else
-
-/* these just cause an oops */
#define BUG() (*(int *)0 = 0)
-#define PAGE_BUG(page) (*(int *)0 = 0)
-
#endif

-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-cris/bug.h
=====
--- mm.orig/include/asm-cris/bug.h 2004-04-03 21:38:19.000000000 -0600
+++ mm/include/asm-cris/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,21 +1,4 @@
 #ifndef _CRIS_BUG_H
 #define _CRIS_BUG_H
-
-#define BUG() do { \
- printk("kernel BUG at %s:%d!\n", __FILE__, __LINE__); \
-} while (0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
-
+#include <asm-generic/bug.h>
#endif
Index: mm/include/asm-h8300/bug.h
=====
--- mm.orig/include/asm-h8300/bug.h 2004-04-03 21:36:13.000000000 -0600
+++ mm/include/asm-h8300/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,21 +1,4 @@
 #ifndef _H8300_BUG_H
 #define _H8300_BUG_H
-
-#define BUG() do { \
- printk("%s(%d): kernel BUG!\n", __FILE__, __LINE__); \
-} while (0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```

-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
-
+#include <asm-generic/bug.h>
#endif
Index: mm/include/asm-i386/bug.h
=====
--- mm.orig/include/asm-i386/bug.h 2004-04-03 21:36:25.000000000 -0600
+++ mm/include/asm-i386/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -9,7 +9,7 @@
 * undefined" opcode for parsing in the trap handler.
 */

-#if 1 /* Set to zero for a slightly smaller kernel */
+#ifdef CONFIG_DEBUG_BUGVERBOSE /* Set to zero for a slightly smaller kernel */
#define BUG() \
    __asm__ __volatile__( "ud2\n" \
        "\t.word %c0\n" \
@@ -19,17 +19,7 @@
#define BUG() __asm__ __volatile__("ud2\n")
#endif

-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-ia64/bug.h
=====
--- mm.orig/include/asm-ia64/bug.h 2004-04-03 21:36:15.000000000 -0600
+++ mm/include/asm-ia64/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -8,15 +8,8 @@
#endif
#define BUG() do { printk("kernel BUG at %s:%d!\n", __FILE__, __LINE__); ia64_abort(); } while (0)

-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)

```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
-
-#define PAGE_BUG(page) do { BUG(); } while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+/* should this BUG should be made generic? */
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>
```

#endif

Index: mm/include/asm-m68k/bug.h

```
----- mm.orig/include/asm-m68k/bug.h 2004-04-03 21:37:38.000000000 -0600
+++ mm/include/asm-m68k/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -21,20 +21,7 @@
 } while (0)
#endif
```

```
-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>
```

#endif

Index: mm/include/asm-m68knommu/bug.h

```
----- mm.orig/include/asm-m68knommu/bug.h 2004-04-03 21:36:26.000000000 -0600
+++ mm/include/asm-m68knommu/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,24 +1,4 @@
 #ifndef _M68KNOMMU_BUG_H
 #define _M68KNOMMU_BUG_H
-
-#define BUG() do { \
- printk("%s(%d): kernel BUG!\n", __FILE__, __LINE__); \
```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
-} while (0)
-
-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
-
+#include <asm-generic/bug.h>
#endif
Index: mm/include/asm-mips/bug.h
=====
--- mm.orig/include/asm-mips/bug.h 2004-04-03 21:37:37.000000000 -0600
+++ mm/include/asm-mips/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -7,14 +7,8 @@
do { \
    __asm__ __volatile__ ("break %0" :: "i" (BRK_BUG)); \
} while (0)
-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-#define PAGE_BUG(page) do { BUG(); } while (0)

-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-parisc/bug.h
=====
--- mm.orig/include/asm-parisc/bug.h 2004-04-03 21:37:37.000000000 -0600
+++ mm/include/asm-parisc/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,29 +1,4 @@
-#ifndef _PARISC_BUG_H
-#define _PARISC_BUG_H
-
-/*
- * Tell the user there is some problem.

```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
- */
-#define BUG() do { \
- extern void dump_stack(void); \
- printk("kernel BUG at %s:%d!\n", __FILE__, __LINE__); \
- dump_stack(); \
-} while (0)
-
-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
-
+#include <asm-generic/bug.h>
#endif
Index: mm/include/asm-ppc/bug.h
=====
--- mm.orig/include/asm-ppc/bug.h 2004-07-15 13:57:57.000000000 -0500
+++ mm/include/asm-ppc/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -35,8 +35,6 @@
     } \
 } while (0)

-#define PAGE_BUG(page) BUG()
-
#define WARN_ON(x) do { \
    if (!__builtin_constant_p(x) || (x)) { \
        __asm__ __volatile__(\
@@ -49,4 +47,9 @@
    } \
 } while (0)

+#define HAVE_ARCH_BUG
+#define HAVE_ARCH_BUG_ON
+#define HAVE_ARCH_WARN_ON
+#include <asm-generic/bug.h>
+
#endif
Index: mm/include/asm-ppc64/bug.h
=====
--- mm.orig/include/asm-ppc64/bug.h 2004-04-03 21:36:24.000000000 -0600
```

## Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
+++ mm/include/asm-ppc64/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -43,8 +43,6 @@
     "i" (__FUNCTION__)); \
} while (0)

-#define PAGE_BUG(page) do { BUG(); } while (0)
-
#define WARN_ON(x) do { \
    __asm__ __volatile__( \
        "1: tdnei %0,0\n" \
@@ -56,4 +54,10 @@
} while (0)

#endif
+
+#define HAVE_ARCH_BUG
+#define HAVE_ARCH_BUG_ON
+#define HAVE_ARCH_WARN_ON
+#include <asm-generic/bug.h>
+
#endif
Index: mm/include/asm-s390/bug.h
=====
--- mm.orig/include/asm-s390/bug.h 2004-04-03 21:37:38.000000000 -0600
+++ mm/include/asm-s390/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -6,22 +6,9 @@
#define BUG() do { \
    printk("kernel BUG at %s:%d!\n", __FILE__, __LINE__); \
    __asm__ __volatile__(".long 0"); \
-} while (0)
-
-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
} while (0)

+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>
+
#endif
```

## Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

Index: mm/include/asm-sh/bug.h

```
----- mm.orig/include/asm-sh/bug.h 2004-04-03 21:38:13.000000000 -0600
+++ mm/include/asm-sh/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -11,20 +11,7 @@
     asm volatile("nop"); \
 } while (0)

-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>

#endif
```

Index: mm/include/asm-sparc/bug.h

```
----- mm.orig/include/asm-sparc/bug.h 2004-07-28 14:52:44.000000000 -0500
+++ mm/include/asm-sparc/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -25,20 +25,7 @@
#define BUG() __bug_trap()
#endif

-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
```

[PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-sparc64/bug.h
=====
--- mm.orig/include/asm-sparc64/bug.h 2004-07-28 14:52:44.000000000 -0500
+++ mm/include/asm-sparc64/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,5 +1,3 @@
-/* $Id$ */
-
#ifdef _SPARC64_BUG_H
#define _SPARC64_BUG_H

@@ -15,20 +13,7 @@
#define BUG() __builtin_trap()
#endif

-#define BUG_ON(condition) do { \
- if (unlikely((condition)!=0)) \
- BUG(); \
-} while(0)
-
-#define PAGE_BUG(page) do { \
- BUG(); \
-} while (0)
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#define HAVE_ARCH_BUG
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-um/bug.h
=====
--- mm.orig/include/asm-um/bug.h 2004-04-03 21:37:06.000000000 -0600
+++ mm/include/asm-um/bug.h 2004-07-30 16:28:00.000000000 -0500
@@ -1,30 +1,4 @@
#ifdef __UM_BUG_H
#define __UM_BUG_H
-
-#ifndef __ASSEMBLY__
-
-#define BUG() do { \
- panic("kernel BUG at %s:%d!\n", __FILE__, __LINE__); \
-} while (0)
-
-#define BUG_ON(condition) do { \
```

## Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
- if (unlikely((condition)!=0)) \  
- BUG(); \  
-} while(0)  
-  
-#define PAGE_BUG(page) do { \  
- BUG(); \  
-} while (0)  
-  
-#define WARN_ON(condition) do { \  
- if (unlikely((condition)!=0)) { \  
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \  
- dump_stack(); \  
- } \  
-} while (0)  
-  
-extern int foo;  
-  
-#endif  
-  
+#include <asm-generic/bug.h>  
#endif  
Index: mm/include/asm-v850/bug.h
```

```
=====  
--- mm.orig/include/asm-v850/bug.h 2004-04-03 21:36:26.000000000 -0600  
+++ mm/include/asm-v850/bug.h 2004-07-30 16:28:00.000000000 -0500  
@@ -16,15 +16,7 @@  
  
extern void __bug (void) __attribute__ ((noreturn));  
#define BUG() __bug()  
-#define PAGE_BUG(page) __bug()  
-  
-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)  
-  
-#define WARN_ON(condition) do { \  
- if (unlikely((condition)!=0)) { \  
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \  
- dump_stack(); \  
- } \  
-} while (0)  
+#define HAVE_ARCH_BUG  
+#include <asm-generic/bug.h>  
  
#endif /* __V850_BUG_H */  
Index: mm/include/asm-x86_64/bug.h
```

```
=====  
--- mm.orig/include/asm-x86_64/bug.h 2004-04-03 21:38:22.000000000 -0600  
+++ mm/include/asm-x86_64/bug.h 2004-07-30 16:28:00.000000000 -0500  
@@ -6,27 +6,20 @@  
/*  
 * Tell the user there is some problem. The exception handler decodes  
 * this frame.
```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```

- */
-struct bug_frame {
- unsigned char ud2[2];
+ */
+struct bug_frame {
+ unsigned char ud2[2];
+     /* should use 32bit offset instead, but the assembler doesn't
- like it */
- char *filename;
- unsigned short line;
-} __attribute__((packed));
+ like it */
+ char *filename;
+ unsigned short line;
+} __attribute__((packed));

+#define HAVE_ARCH_BUG
#define BUG() \
    asm volatile("ud2 ; .quad %c1 ; .short %c0" :: \
        "i"(__LINE__), "i" (__stringify(KBUILD_BASENAME)))
-#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
-#define PAGE_BUG(page) BUG()
void out_of_line_bug(void);
-
-#define WARN_ON(condition) do { \
- if (unlikely((condition)!=0)) { \
- printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
- dump_stack(); \
- } \
-} while (0)
+#include <asm-generic/bug.h>

#endif
Index: mm/include/asm-generic/bug.h
=====
--- mm.orig/include/asm-generic/bug.h 2003-09-12 12:14:37.000000000 -0500
+++ mm/include/asm-generic/bug.h 2004-07-30 16:30:53.000000000 -0500
@@ -0,0 +1,34 @@
+#ifndef _ASM_GENERIC_BUG_H
+#define _ASM_GENERIC_BUG_H
+
+#include <linux/compiler.h>
+#include <linux/config.h>
+
+#ifndef HAVE_ARCH_BUG
+#define BUG() do { \
+ printk("kernel BUG at %s:%d!\n", __FILE__, __LINE__); \
+ panic("BUG!"); \
+} while (0)
+#endif
+

```

Linux-Kernel: [PATCH] move duplicate BUG and WARN\_ON bits to asm-generic

```
+#ifndef HAVE_ARCH_PAGE_BUG
+#define PAGE_BUG(page) do { \
+ printk("page BUG for page at %p\n", page); \
+ BUG(); \
+} while (0)
+#endif
+
+#ifndef HAVE_ARCH_BUG_ON
+#define BUG_ON(condition) do { if (unlikely((condition)!=0)) BUG(); } while(0)
+#endif
+
+#ifndef HAVE_ARCH_WARN_ON
+#define WARN_ON(condition) do { \
+ if (unlikely((condition)!=0)) { \
+ printk("Badness in %s at %s:%d\n", __FUNCTION__, __FILE__, __LINE__); \
+ dump_stack(); \
+ } \
+} while (0)
+#endif
+
+#endif
```

--  
Mathematics is the supreme nostalgia of our time.

-  
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/>