

```

*****
15052 Fri Feb 27 16:32:47 2015
new/usr/src/common/atomic/arm/atomic.s
atomic: remove old weak symbols
These symbols have been deprecated for a really long time on x86 and SPARC.
They exist as weak symbols to allow old binaries to still work. On ARM,
these old binaries never existed to begin with and so we don't have to
provide these legacy symbols.
*****
1 /*
2  * This file and its contents are supplied under the terms of the
3  * Common Development and Distribution License ("CDDL"), version 1.0.
4  * You may only use this file in accordance with the terms of version
5  * 1.0 of the CDDL.
6  *
7  * A full copy of the text of the CDDL should have accompanied this
8  * source. A copy of the CDDL is also available via the Internet at
9  * http://www.illumos.org/license/CDDL.
10 */

12 /*
13  * Copyright (c) 2013 Joyent, Inc. All rights reserved.
14 */

16     .file     "atomic.s"

18 /*
19  * Atomic Operatoins for 32-bit ARM. Note, that these require at least ARMv6K so
20  * as to have access to the non-word size LDREX and STREX.
21  */

23 #include <sys/asm_linkage.h>
24 #include <sys/atomic_impl.h>

26 /*
27  * XXX We probably want some kind of backoff built in to these routines at some
28  * point.
29  */

31 #if defined(_KERNEL)
32     /*
33      * Legacy kernel interfaces; they will go away (eventually).
34      */
35     ANSI_PRAGMA_WEAK2(cas8,atomic_cas_8,function)
36     ANSI_PRAGMA_WEAK2(cas32,atomic_cas_32,function)
37     ANSI_PRAGMA_WEAK2(cas64,atomic_cas_64,function)
38     ANSI_PRAGMA_WEAK2(caslong,atomic_cas_ulong,function)
39     ANSI_PRAGMA_WEAK2(casptr,atomic_cas_ptr,function)
40     ANSI_PRAGMA_WEAK2(atomic_and_long,atomic_and_ulong,function)
41     ANSI_PRAGMA_WEAK2(atomic_or_long,atomic_or_ulong,function)
42     ANSI_PRAGMA_WEAK2(swapl,atomic_swap_32,function)
43 #endif

31     /*
32      * NOTE: If atomic_inc_8 and atomic_inc_8_nv are ever
33      * separated, you need to also edit the libc arm platform
34      * specific mapfile and remove the NODYNSORT attribute
35      * from atomic_inc_8_nv.
36      */
37     ENTRY(atomic_inc_8)
38     ALTENTRY(atomic_inc_8_nv)
39     ALTENTRY(atomic_inc_uchar)
40     ALTENTRY(atomic_inc_uchar_nv)
41     mov     r1, #1
42     b       atomic_add_8
43     SET_SIZE(atomic_inc_uchar_nv)
unchanged portion omitted

```