

new/usr/src/cmd/localedef/messages.c

1

\*\*\*\*\*

2047 Fri Jul 11 11:42:51 2014

new/usr/src/cmd/localedef/messages.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */
```

```
17 /*
18  * LC_MESSAGES database generation routines for localedef.
19 */
```

```
21 #include <stdio.h>
22 #include <stdlib.h>
23 #include <errno.h>
24 #include <sys/types.h>
25 #include <string.h>
26 #include <unistd.h>
27 #include <alloca.h>
28 #include "localedef.h"
29 #include "parser.tab.h"
30 #include "lmessages.h"
```

```
32 static struct lc_messages msgs;
31 static struct lc_messages_T msgs;
```

```
34 void
35 init_messages(void)
36 {
37     (void) memset(&msgs, 0, sizeof (msgs));
38 }
```

unchanged\_portion\_omitted

new/usr/src/cmd/localedef/monetary.c

1

\*\*\*\*\*

4248 Fri Jul 11 11:42:51 2014

new/usr/src/cmd/localedef/monetary.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */
```

```
17 /*
18  * LC_MONETARY database generation routines for localedef.
19 */
```

```
21 #include <stdio.h>
22 #include <stdlib.h>
23 #include <errno.h>
24 #include <sys/types.h>
25 #include <string.h>
26 #include <unistd.h>
27 #include "localedef.h"
28 #include "parser.tab.h"
29 #include "lmonetary.h"
```

```
31 static struct lc_monetary mon;
30 static struct lc_monetary_T mon;
```

```
33 void
34 init_monetary(void)
35 {
36     (void) memset(&mon, 0, sizeof (mon));
37 }
```

unchanged\_portion\_omitted\_

new/usr/src/cmd/localedef/numeric.c

1

\*\*\*\*\*

1892 Fri Jul 11 11:42:51 2014

new/usr/src/cmd/localedef/numeric.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */
```

```
17 /*
18  * LC_NUMERIC database generation routines for localedef.
19 */
```

```
21 #include <stdio.h>
22 #include <stdlib.h>
23 #include <errno.h>
24 #include <sys/types.h>
25 #include <string.h>
26 #include <unistd.h>
27 #include "localedef.h"
28 #include "parser.tab.h"
29 #include "lnumeric.h"
```

```
31 static struct lc_numeric numeric;
30 static struct lc_numeric_T numeric;
```

```
33 void
34 init_numeric(void)
35 {
36     (void) memset(&numeric, 0, sizeof (numeric));
37 }
38
39 unchanged_portion_omitted_
```

new/usr/src/cmd/localedef/time.c

1

\*\*\*\*\*

4496 Fri Jul 11 11:42:51 2014

new/usr/src/cmd/localedef/time.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */
```

```
17 /*
18  * LC_TIME database generation routines for localedef.
19 */
```

```
21 #include <stdio.h>
22 #include <stdlib.h>
23 #include <errno.h>
24 #include <sys/types.h>
25 #include <string.h>
26 #include <unistd.h>
27 #include "localedef.h"
28 #include "parser.tab.h"
29 #include "timelocal.h"
```

```
31 struct lc_time tm;
30 struct lc_time_T tm;
```

```
33 void
34 init_time(void)
35 {
36     (void) memset(&tm, 0, sizeof (tm));
37 }
```

unchanged\_portion\_omitted\_

```

*****
1789 Fri Jul 11 11:42:52 2014
new/usr/src/cmd/mdb/common/kmdb/mapfile_skel
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /* BEGIN PROLOGUE */
22 /*
23  * Copyright (c) 2004, 2010, Oracle and/or its affiliates. All rights reserved.
24  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
25  */
26
27 /*
28  * This file is used to limit the symbols that are to be exported from the
29  * debugger. This ensures that dmods follow the module API.
30  *
31  * There is a special rule for generating the mapfile. If the symbol
32  * is not a function then the symbol, in the mapfile, must have the
33  * the token "variable" as the third symbol on the line - see the
34  * entries for __ctype and errno below.
35  */
36
37 $mapfile_version 2
38
39 SYMBOL_SCOPE {
40     global:
41
42     /* END PROLOGUE */
43     /* BEGIN EPILOGUE */
44     /*
45      * Secret additions to the module API
46      */
47
48     /* Implementation detail of the ctype macros */
49     __ctype; /* variable */
50     /* There should be only one - ours */
51     errno; /* variable */
52     isprint;
53     isalnum;
54     isalpha;
55     isgraph;
56     iscntrl;
57     isdigit;
58     isxdigit;

```

```

57         isupper;
58         islower;
59         ispunct;
60         isspace;
61
62         mdb_tgt_aread;
63         mdb_dis_create;
64         mdb_dis_destroy;
65
66         local:
67             *;
68     };

```

unchanged\_portion\_omitted

```

*****
1718 Fri Jul 11 11:42:52 2014
new/usr/src/cmd/mdb/common/libstand/ctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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  * CDDL HEADER START
7  *
8  * A full copy of the text of the CDDL should have accompanied this
9  * source. A copy of the CDDL is also available via the Internet at
10 * http://www.illumos.org/license/CDDL.
11 * The contents of this file are subject to the terms of the
12 * Common Development and Distribution License, Version 1.0 only
13 * (the "License"). You may not use this file except in compliance
14 * with the License.
15 *
16 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
17 * or http://www.opensolaris.org/os/licensing.
18 * See the License for the specific language governing permissions
19 * and limitations under the License.
20 *
21 * When distributing Covered Code, include this CDDL HEADER in each
22 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
23 * If applicable, add the following below this CDDL HEADER, with the
24 * fields enclosed by brackets "[]" replaced with your own identifying
25 * information: Portions Copyright [yyyy] [name of copyright owner]
26 *
27 * CDDL HEADER END
28 */
29
30 /*
31 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
32 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
33 * Use is subject to license terms.
34 */
35
36 /*
37 * ASCII versions of ctype character classification functions. This avoids
38 * pulling in the entire locale framework that is in libc.
39 */
40 #pragma ident "%Z%M% %I% %E% SMI"
41
42 int
43 isdigit(int c)
44 {
45     return ((c >= '0' && c <= '9') ? 1 : 0);
46 }
47 #include <ctype.h>
48
49 int
50 isupper(int c)
51 {
52     unsigned char __ctype[129] =
53     {
54         return ((c >= 'A' && c <= 'Z') ? 1 : 0);
55     }
56 }
57
58 int
59 islower(int c)

```

```

36 {
37     return ((c >= 'a' && c <= 'z') ? 1 : 0);
38 }
39
40 int
41 isspace(int c)
42 {
43     return (((c == ' ') || (c == '\t') || (c == '\r') || (c == '\n') ||
44             (c == '\v') || (c == '\f')) ? 1 : 0);
45 }
46
47 int
48 isxdigit(int c)
49 {
50     return ((isdigit(c) || (c >= 'A' && c <= 'F') ||
51             (c >= 'a' && c <= 'f')) ? 1 : 0);
52 }
53
54 int
55 isalpha(int c)
56 {
57     return ((isupper(c) || islower(c)) ? 1 : 0);
58 }
59
60 int
61 isalnum(int c)
62 {
63     return ((isalpha(c) || isdigit(c)) ? 1 : 0);
64 }
65
66 int
67 ispunct(int c)
68 {
69     return (((c >= '!' && (c <= '/')) ||
70             ((c >= ':' && (c <= '@')) ||
71             ((c >= '[' && (c <= '`')) ||
72             ((c >= '{' && (c <= '~'))));
73 }
74
75 int
76 iscntrl(int c)
77 {
78     return ((c < 0x20) || (c == 0x7f));
79 }
80
81 int
82 isprint(int c)
83 {
84     /*
85      * Almost the inverse of iscntrl, but be careful that c > 0x7f
86      * returns false for everything.
87      */
88     return ((c >= ' ') && (c <= '~'));
89 }
90
91 int
92 isgraph(int c)
93 {
94     /* isgraph is like is print, but excludes <space> */
95     return ((c >= ' ') && (c <= '~'));
96 }
97
98 0, /* EOF */
99 _C, _C, _C, _C, _C, _C, _C, _C,
100 _C, _S/_C, _S/_C, _S/_C, _S/_C, _S/_C, _C, _C,
101 _C, _C, _C, _C, _C, _C, _C, _C,

```

```
37  _C,  _C,  _C,  _C,  _C,  _C,  _C,  _C,
38  _S|_B, _P,  _P,  _P,  _P,  _P,  _P,  _P,
39  _P,  _P,  _P,  _P,  _P,  _P,  _P,  _P,
40  _N|_X, _N|_X, _N|_X, _N|_X, _N|_X, _N|_X, _N|_X, _N|_X,
41  _N|_X, _N|_X, _P,  _P,  _P,  _P,  _P,  _P,
42  _P,  _U|_X, _U|_X, _U|_X, _U|_X, _U|_X, _U|_X, _U|_X,
43  _U,  _U,  _U,  _U,  _U,  _U,  _U,  _U,
44  _U,  _U,  _U,  _U,  _U,  _U,  _U,  _U,
45  _U,  _U,  _U,  _P,  _P,  _P,  _P,  _P,
46  _P,  _L|_X, _L|_X, _L|_X, _L|_X, _L|_X, _L|_X, _L|_X,
47  _L,  _L,  _L,  _L,  _L,  _L,  _L,  _L,
48  _L,  _L,  _L,  _L,  _L,  _L,  _L,  _L,
49  _L,  _L,  _L,  _P,  _P,  _P,  _P,  _C,
50  };
```

```

*****
19617 Fri Jul 11 11:42:52 2014
new/usr/src/cmd/sgs/rtld/common/external.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright (c) 1992, 2010, Oracle and/or its affiliates. All rights reserved.
24  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
25  */
26
27 /*
28  * Implementation of all external interfaces between ld.so.1 and libc.
29  *
30  * This file started as a set of routines that provided synchronization and
31  * locking operations using calls to libthread. libthread has merged with libc
32  * under the Unified Process Model (UPM), and things have gotten a lot simpler.
33  * This file continues to establish and redirect various events within ld.so.1
34  * to interfaces within libc.
35  *
36  * Until libc is loaded and relocated, any external interfaces are captured
37  * locally. Each link-map list maintains its own set of external vectors, as
38  * each link-map list typically provides its own libc. Although this per-link-
39  * map list vectoring provides a degree of flexibility, there is a protocol
40  * expected when calling various libc interfaces.
41  *
42  * i. Any new alternative link-map list should call CI_THRINIT, and then call
43  * CI_TLS_MODADD to register any TLS for each object of that link-map list
44  * (this item is labeled i. as auditors can be the first objects loaded,
45  * and they exist on their own link-map list).
46  *
47  * ii. For the primary link-map list, CI_TLS_STATMOD must be called first to
48  * register any static TLS. This routine is called regardless of there
49  * being any TLS, as this routine also establishes the link-map list as the
50  * primary list and fixes the association of uberdata). CI_THRINIT should
51  * then be called.
52  *
53  * iii. Any objects added to an existing link-map list (primary or alternative)
54  * should call CI_TLS_MODADD to register any additional TLS.
55  *
56  * These events are established by:
57  *
58  * i. Typically, libc is loaded as part of the primary dependencies of any

```

```

59  * link-map list (since the Unified Process Model (UPM), libc can't be
60  * lazily loaded). To minimize the possibility of loading and registering
61  * objects, and then tearing them down (because of a relocation error),
62  * external vectors are established as part of load_completion(). This
63  * routine is called on completion of any operation that can cause objects
64  * to be loaded. This point of control insures the objects have been fully
65  * analyzed and relocated, and moved to their controlling link-map list.
66  * The external vectors are established prior to any .inits being fired.
67  *
68  * ii. Calls to CI_THRINIT, and CI_TLS_MODADD also occur as part of
69  * load_completion(). CI_THRINIT is only called once for each link-map
70  * control list.
71  *
72  * iii. Calls to CI_TLS_STATMOD, and CI_THRINIT occur for the primary link-map
73  * list in the final stages of setup().
74  *
75  * The interfaces provide by libc can be divided into two families. The first
76  * family consists of those interfaces that should be called from the link-map
77  * list. It's possible that these interfaces convey state concerning the
78  * link-map list they are part of:
79  *
80  * CI_ATEXIT
81  * CI_TLS_MODADD
82  * CI_TLS_MODREM
83  * CI_TLS_STATMOD
84  * CI_THRINIT
85  *
86  * The second family are global in nature, that is, the link-map list from
87  * which they are called provides no state information. In fact, for
88  * CI_BIND_GUARD, the calling link-map isn't even known. The link-map can only
89  * be deduced after ld.so.1's global lock has been obtained. Therefore, the
90  * following interfaces are also maintained as global:
91  *
92  * CI_LCMESSAGES
93  * CI_BIND_GUARD
94  * CI_BIND_CLEAR
95  * CI_THR_SELF
96  *
97  * Note, it is possible that these global interfaces are obtained from an
98  * alternative link-map list that gets torn down because of a processing
99  * failure (unlikely, because the link-map list components must be analyzed
100 * and relocated prior to load_completion(), but perhaps the tear down is still
101 * a possibility). Thus the global interfaces may have to be replaced. Once
102 * the interfaces have been obtained from the primary link-map, they can
103 * remain fixed, as the primary link-map isn't going to go anywhere.
104 *
105 * The last wrinkle in the puzzle is what happens if an alternative link-map
106 * is loaded with no libc dependency? In this case, the alternative objects
107 * can not call CI_THRINIT, can not be allowed to use TLS, and will not receive
108 * any atexit processing.
109 *
110 * The history of these external interfaces is defined by their version:
111 *
112 * TI_VERSION == 1
113 * Under this model libthread provided rw_rwlock/rw_unlock, through which
114 * all rt_mutex_lock/rt_mutex_unlock calls were vectored.
115 * Under libc/libthread these interfaces provided _sigon/_sigoff (unlike
116 * lwp/libthread that provided signal blocking via bind_guard/bind_clear).
117 *
118 * TI_VERSION == 2
119 * Under this model only libthreads bind_guard/bind_clear and thr_self
120 * interfaces were used. Both libthreads blocked signals under the
121 * bind_guard/bind_clear interfaces. Lower level locking is derived
122 * from internally bound_lwp interfaces. This removes recursive
123 * problems encountered when obtaining locking interfaces from libthread.
124 * The use of mutexes over reader/writer locks also enables the use of

```



```

125 *      condition variables for controlling thread concurrency (allows access
126 *      to objects only after their .init has completed).
127 *
128 * NOTE, the TI_VERSION indicated the ti_interface version number, where the
129 * ti_interface was a large vector of functions passed to both libc (to override
130 * the thread stub interfaces) and ld.so.1. ld.so.1 used only a small subset of
131 * these interfaces.
132 *
133 * CI_VERSION == 1
134 *   Introduced with CI_VERSION & CI_ATEXIT
135 *
136 * CI_VERSION == 2 (Solaris 8 update 2).
137 *   Added support for CI_LCMESSAGES
138 *
139 * CI_VERSION == 3 (Solaris 9).
140 *   Added the following versions to the CI table:
141 *
142 *       CI_BIND_GUARD, CI_BIND_CLEAR, CI_THR_SELF
143 *       CI_TLS_MODADD, CI_TLS_MOD_REMOVE, CI_TLS_STATMOD
144 *
145 *   This version introduced the DT_SUNW_RTLDINFO structure as a mechanism
146 *   to handshake with ld.so.1.
147 *
148 * CI_VERSION == 4 (Solaris 10).
149 *   Added the CI_THRINIT handshake as part of the libc/libthread unified
150 *   process model. libc now initializes the current thread pointer from
151 *   this interface (and no longer relies on the INITFIRST flag - which
152 *   others have started to camp out on).
153 *
154 * CI_VERSION == 5 (Solaris 11).
155 *   Use of "protected" references within libc, so that symbols are
156 *   pre-bound, and don't require ld.so.1 binding. This implementation
157 *   protects libc's critical regions from being vectored to auditors.
158 *
159 * CI_VERSION == 6 (Solaris 11).
160 *   Added the CI_CRITICAL handshake, to allow "mem*" family to be reexposed
161 *   as "global", and thus be redirected to auxiliary filters.
162 *
163 * Release summary:
164 *
165 *   Solaris 8      CI_ATEXIT via _ld_libc()
166 *                  TI_* via _ld_concurrency()
167 *
168 *   Solaris 9      CI_ATEXIT and CI_LCMESSAGES via _ld_libc()
169 *                  CI_* via RTLDINFO and _ld_libc() - new libthread
170 *                  TI_* via _ld_concurrency() - old libthread
171 *
172 *   Solaris 10     CI_ATEXIT and CI_LCMESSAGES via _ld_libc()
173 *                  CI_* via RTLDINFO and _ld_libc() - new libthread
174 */

176 #include <sys/debug.h>
177 #include <synch.h>
178 #include <signal.h>
179 #include <thread.h>
180 #include <synch.h>
181 #include <strings.h>
182 #include <stdio.h>
183 #include <debug.h>
184 #include <libc_int.h>
185 #include "_elf.h"
186 #include "_rtld.h"

188 /*
189 * This interface provides the unified process model communication between
190 * ld.so.1 and libc. This interface can be called a number of times:

```

```

191 *
192 * - Initially, this interface is called to process RTLDINFO. This data
193 *   structure is typically provided by libc, and contains the address of
194 *   libc interfaces that must be called to initialize threads information.
195 *
196 * - _ld_libc(), this interface can also be called by libc at process
197 *   initialization, after libc has been loaded and relocated, but before
198 *   control has been passed to any user code (.init's or main()). This
199 *   call provides additional libc interface information that ld.so.1 must
200 *   call during process execution.
201 *
202 * - _ld_libc() can also be called by libc during process execution to
203 *   re-establish interfaces such as the locale.
204 */
205 static void
206 get_lcinterface(Rt_map *lmp, Lc_interface *funcs)
207 {
208     int          threaded = 0, entry = 0, tag;
209     Lm_list      *lml;
210     Lc_desc      *lcp;

212     if ((lmp == NULL) || (funcs == NULL))
213         return;

215     /*
216      * Once the process is active, ensure we grab a lock.
217      */
218     if (rtld_flags & RT_FL_APPLIC)
219         entry = enter(0);

221     lml = LIST(lmp);
222     lcp = &lml->lm_lcs[0];

224     DBG_CALL(DBG_util_nl(lml, DBG_NL_STD));

226     for (tag = funcs->ci_tag; tag; tag = (++funcs->ci_tag) {
227         char *gpptr;
228         char *lpptr = funcs->ci_un.ci_ptr;

230         DBG_CALL(DBG_util_lcinterface(lmp, tag, lpptr));

232         if (tag >= CI_MAX)
233             continue;

235         /*
236          * Maintain all interfaces on a per-link-map basis. Note, for
237          * most interfaces, only the first interface is used for any
238          * link-map list. This prevents accidents with developers who
239          * manage to load two different versions of libc.
240          */
241         if ((lcp[tag].lc_lmp) &&
242             (tag != CI_LCMESSAGES) && (tag != CI_VERSION)) {
243             DBG_CALL(DBG_unused_lcinterface(lmp,
244                 lcp[tag].lc_lmp, tag));
245             continue;
246         }

248         lcp[tag].lc_un.lc_ptr = lpptr;
249         lcp[tag].lc_lmp = lmp;

251         gpptr = glcs[tag].lc_un.lc_ptr;

253         /*
254          * Process any interfaces that must be maintained on a global
255          * basis.
256          */

```

```

257     switch (tag) {
258     case CI_ATEXIT:
259         break;

261     case CI_LCMESSAGES:
262         /*
263          * At startup, ld.so.1 can establish a locale from one
264          * of the locale family of environment variables (see
265          * ld_str_env() and readenv_user()). During process
266          * execution the locale can also be changed by the user.
267          * This interface is called from libc should the locale
268          * be modified. Presently, only one global locale is
269          * maintained for all link-map lists, and only objects
270          * on the primary link-map may change this locale.
271          */
272         if ((lml->lm_flags & LML_FLG_BASELM) &&
273             ((gptr == NULL) || (strcmp(gptr, lpstr) != 0))) {
274             /*
275              * If we've obtained a message locale (typically
276              * supplied via libc's setlocale()), then
277              * register the locale for use in dgettext() so
278              * as to reestablish the locale for ld.so.1's
279              * messages.
280              */
281             if (gpstr) {
282                 free((void *)gpstr);
283                 rtld_flags |= RT_FL_NEWLOCALE;
284             }
285             glcs[tag].lc_un.lc_ptr = strdup(lpstr);

287             /*
288              * Clear any cached messages.
289              */
290             bzero(err_strs, sizeof (err_strs));
291             nosym_str = NULL;
292         }
293         break;

295     case CI_BIND_GUARD:
296     case CI_BIND_CLEAR:
297     case CI_THR_SELF:
298     case CI_CRITICAL:
299         /*
300          * If the global vector is unset, or this is the primary
301          * link-map, set the global vector.
302          */
303         if ((gpstr == NULL) || (lml->lm_flags & LML_FLG_BASELM))
304             glcs[tag].lc_un.lc_ptr = lpstr;

306         /* FALLTHROUGH */

308     case CI_TLS_MODADD:
309     case CI_TLS_MODREM:
310     case CI_TLS_STATMOD:
311     case CI_THRINIT:
312         threaded++;
313         break;

315     case CI_VERSION:
316         if ((rtld_flags2 & RT_FL2_RTLTLDSEEN) == 0) {
317             Aliste idx;
318             Lm_list *lml2;
319             int version;

321             rtld_flags2 |= RT_FL2_RTLTLDSEEN;

```

```

323         version = funcs->ci_un.ci_val;
324 #if defined(CI_V_FIVE)
325         if (version >= CI_V_FIVE) {
326             thr_flg_nolock = THR_FLG_NOLOCK;
327             thr_flg_reenter = THR_FLG_REENTER;
328         }
329 #endif
330         if (version < CI_V_FOUR)
331             break;

333         rtld_flags2 |= RT_FL2_UNIFPROC;

335         /*
336          * We might have seen an auditor which is not
337          * dependent on libc. Such an auditor's link
338          * map list has LML_FLG_HOLDLOCK set. This
339          * lock needs to be dropped. Refer to
340          * audit_setup() in audit.c.
341          */
342         if ((rtld_flags2 & RT_FL2_HASAUDIT) == 0)
343             break;

345         /*
346          * Yes, we did. Take care of them.
347          */
348         for (APLIST_TRAVERSE(dynlm_list, idx, lml2)) {
349             Rt_map *map = (Rt_map *)lml2->lm_head;

351             if (FLAGS(map) & FLG_RT_AUDIT) {
352                 lml2->lm_flags &=
353                     ~LML_FLG_HOLDLOCK;
354             }
355         }
356         break;
357     }

359     default:
360         break;
361     }
362 }

364     if (threaded) {
365         /*
366          * If a version of libc gives us only a subset of the TLS
367          * interfaces, it's confused and we discard the whole lot.
368          */
369         if ((lcp[CI_TLS_MODADD].lc_un.lc_func &&
370             lcp[CI_TLS_MODREM].lc_un.lc_func &&
371             lcp[CI_TLS_STATMOD].lc_un.lc_func) == NULL) {
372             lcp[CI_TLS_MODADD].lc_un.lc_func = NULL;
373             lcp[CI_TLS_MODREM].lc_un.lc_func = NULL;
374             lcp[CI_TLS_STATMOD].lc_un.lc_func = NULL;
375         }

377         /*
378          * Indicate that we're now thread capable.
379          */
380         if ((lml->lm_flags & LML_FLG_RTLDLM) == 0)
381             rtld_flags |= RT_FL_THREADS;
382     }

384     if (entry)
385         leave(lml, 0);
386 }

```

unchanged portion omitted

```
666 ssize_t
667 write(int fd, const void *buf, size_t size)
668 {
669     extern ssize_t __write(int, const void *, size_t);
670     return (__write(fd, buf, size));
671 }

673 /*
674  * ASCII versions of ctype character classification functions. This avoids
675  * pulling in the entire locale framework that is in libc.
676  */

678 int
679 isdigit(int c)
680 {
681     return ((c >= '0' && c <= '9') ? 1 : 0);
682 }

684 int
685 isupper(int c)
686 {
687     return ((c >= 'A' && c <= 'Z') ? 1 : 0);
688 }

690 int
691 islower(int c)
692 {
693     return ((c >= 'a' && c <= 'z') ? 1 : 0);
694 }

696 int
697 isspace(int c)
698 {
699     return (((c == ' ') || (c == '\t') || (c == '\r') || (c == '\n') ||
700             (c == '\v') || (c == '\f')) ? 1 : 0);
701 }

703 int
704 isxdigit(int c)
705 {
706     return ((isdigit(c) || (c >= 'A' && c <= 'F') ||
707             (c >= 'a' && c <= 'f')) ? 1 : 0);
708 }

710 int
711 isalpha(int c)
712 {
713     return ((isupper(c) || islower(c)) ? 1 : 0);
714 }

716 int
717 isalnum(int c)
718 {
719     return ((isalpha(c) || isdigit(c)) ? 1 : 0);
720 }

_____unchanged_portion_omitted_
```

```

*****
11016 Fri Jul 11 11:42:52 2014
new/usr/src/head/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright 2010 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
26 #
27 # head/Makefile
28 #
29 # include global definitions
30 include ../Makefile.master

32 sparc_HDRS=
33 i386_HDRS=      stack_unwind.h

35 # Headers are listed one per line so that TeamWare can auto-merge most changes

37 KRB5HDRS= mit_copyright.h mit-sipb-copyright.h

39 ATTRDB_HDRS=  secdb.h auth_attr.h exec_attr.h prof_attr.h user_attr.h \
40              auth_list.h

42 HDRS=        ${$(MACH)_HDRS}  ${$(ATTRDB_HDRS)} \
43              aio.h             \
44              alloca.h          \
45              apptrace.h        \
46              apptrace_impl.h  \
47              ar.h              \
48              archives.h        \
49              assert.h          \
50              atomic.h          \
51              attr.h            \
52              config_admin.h    \
53              cpio.h            \
54              crypt.h           \
55              ctype.h           \
56              deflt.h           \
57              devid.h           \
58              devmgmt.h         \

```

```

59          devpoll.h           \
60          dial.h              \
61          dirent.h           \
62          dlfcn.h            \
63          door.h             \
64          elf.h              \
65          err.h              \
66          errno.h            \
67          euc.h              \
68          exacct.h           \
69          exacct_impl.h      \
70          execinfo.h         \
71          fatal.h            \
72          fcntl.h            \
73          float.h            \
74          fmtmsg.h           \
75          fnmatch.h          \
76          ftw.h              \
77          gelf.h             \
78          getopt.h           \
79          getwidth.h         \
80          glob.h             \
81          grp.h              \
82          iconv.h            \
83          ieeefp.h           \
84          ifaddrs.h          \
85          inttypes.h         \
86          iso646.h           \
87          klpd.h             \
88          langinfo.h         \
89          lastlog.h          \
90          lber.h             \
91          ldap.h             \
92          libelf.h           \
93          libgen.h           \
94          libintl.h          \
95          libw.h             \
96          libzonecfg.h       \
97          limits.h           \
98          linenum.h          \
99          link.h             \
100         listen.h            \
101         locale.h            \
102         macros.h            \
103         malloc.h            \
104         mdmn_changelog.h    \
105         memory.h            \
106         meta.h              \
107         meta_runtime.h      \
108         metadyn.h           \
109         mon.h               \
110         monetary.h          \
111         mp.h                \
112         mqueue.h            \
113         nan.h               \
114         ndbm.h              \
115         ndpd.h              \
116         netconfig.h         \
117         netdb.h             \
118         netdir.h            \
119         nl_types.h          \
120         nlist.h             \
121         note.h              \
122         nsctl.h             \
123         nsswitch.h          \
124         nss_common.h        \

```

## new/usr/src/head/Makefile

```

125     nss_dbdefs.h  \
126     nss_netdir.h  \
127     paths.h       \
128     pcsample.h    \
129     pfmt.h        \
130     pkgdev.h      \
131     pkginfo.h     \
132     pkglocs.h     \
133     pkgstrct.h    \
134     pkgtrans.h    \
135     poll.h        \
136     port.h        \
137     priv.h        \
138     priv_utils.h  \
139     proc_service.h \
140     procfs.h      \
141     prof.h        \
142     project.h     \
143     pthread.h     \
144     pw.h          \
145     pwd.h         \
146     rctl.h        \
147     re_comp.h     \
148     regex.h       \
149     regexp.h      \
150     resolv.h      \
151     rje.h         \
152     rtld_db.h     \
153     sac.h         \
154     sched.h       \
155     schedctl.h    \
156     sdssc.h       \
157     search.h      \
158     semaphore.h   \
159     setjmp.h      \
160     sgtty.h       \
161     shadow.h      \
162     sginfo.h      \
163     signal.h      \
164     spawn.h       \
165     stdarg.h      \
166     stdbool.h     \
167     stddef.h      \
168     stdint.h      \
169     stdio.h       \
170     stdio_ext.h   \
171     stdio_tag.h   \
172     stdio_impl.h  \
173     stdlib.h      \
174     storclass.h   \
175     string.h      \
176     strings.h     \
177     stropts.h    \
178     synch.h       \
179     sysexits.h    \
180     syslog.h      \
181     syms.h        \
182     tar.h         \
183     termio.h      \
184     termios.h     \
185     thread.h      \
186     thread_db.h   \
187     time.h        \
188     tiuser.h      \
189     tzfile.h      \
190     ucontext.h    \

```

3

## new/usr/src/head/Makefile

```

191     ucred.h       \
192     ulimit.h      \
193     unistd.h      \
194     userdefs.h    \
195     ustat.h       \
196     utime.h       \
197     utmp.h        \
198     utmpx.h       \
199     valtools.h    \
200     values.h      \
201     varargs.h     \
202     wait.h        \
203     wchar.h       \
204     wchar_impl.h  \
205     wctype.h      \
206     widec.h       \
207     wordexp.h     \
208     xlocale.h    \
209     xti.h         \
210     xti_inet.h    \
211     zone.h        \
212 \
213 ISOHDRS = \
214     ctype_c99.h  \
215     ctype_iso.h   \
216     limits_iso.h  \
217     locale_iso.h  \
218     setjmp_iso.h  \
219     signal_iso.h  \
220     stdarg_c99.h  \
221     stdarg_iso.h  \
222     stddef_iso.h  \
223     stdio_c99.h   \
224     stdio_iso.h   \
225     stdlib_c99.h  \
226     stdlib_iso.h  \
227     string_iso.h  \
228     time_iso.h    \
229     wchar_c99.h   \
230     wctype_c99.h \
231     wctype_iso.h  \
232 \
233 ARPAHDRS = \
234     ftp.h         \
235     inet.h        \
236     nameser.h     \
237     telnet.h      \
238     tftp.h        \
239     nameser_compat.h \
240 \
241 AUDIOHDRS = \
242     au.h          \
243 \
244 UIIDHDRS = \
245     uuid.h        \
246 \
247 # rpcsvc headers which are just headers (not derived from a .x file)
248 RPCSVC_SRC_HDRS = \
249     bootparam.h   \
250     daemon_utils.h \
251     dbm.h          \
252     nis_db.h       \
253     nislib.h       \
254     svc_dg_priv.h \
255     yp_prot.h      \

```

4

## new/usr/src/head/Makefile

```

255     ypcInt.h      \
256     yppasswd.h   \
257     yppwd.h      \
258     rpc_sotypes.h

260 # rpcsvc headers which are generated from .x files
261 RPCSVC_GEN_HDRS = \
262     bootparam_prot.h \
263     mount.h          \
264     nfs_prot.h       \
265     nfs4_prot.h     \
266     nis.h            \
267     rex.h            \
268     rquota.h        \
269     rstat.h         \
270     rusers.h        \
271     rwall.h         \
272     spray.h         \
273     ufs_prot.h      \
274     nfs_acl.h

276 LVMPCHDRS = \
277     mndx.h mdiox.h meta_basic.h metad.h metamed.h metamd.h metacl.h

279 SYMHDRASSERT = $(ROOT)/usr/include/iso/assert_iso.h
280 SYMHDRERRNO  = $(ROOT)/usr/include/iso/errno_iso.h
281 SYMHDRFLOAT  = $(ROOT)/usr/include/iso/float_iso.h
282 SYMHDRISO646 = $(ROOT)/usr/include/iso/iso646_iso.h

284 RPCGENFLAGS = -C -h
285 rpcsvc/rwall.h := RPCGENFLAGS += -M
286 meta_basic.h := RPCGENFLAGS += -M
287 metad.h := RPCGENFLAGS += -M
288 metamed.h := RPCGENFLAGS += -M
289 mndx.h := RPCGENFLAGS += -M
290 mdiox.h := RPCGENFLAGS += -M
291 metamd.h := RPCGENFLAGS += -M
292 metacl.h := RPCGENFLAGS += -M

294 # rpcsvc rpcgen source (.x files)
295 #
296 # yp.x is an attempt at codifying what was hand coded in RPCL.
297 # Unfortunately it doesn't quite work. (The handcoded stuff isn't
298 # expressible in RPCL) This is due to the fact that YP was written
299 # before rpcgen existed. Hence, yp_prot.h cannot be derived from yp.x
300 #
301 # There is no '.h' for nis_object.x because it is included by nis.x and
302 # the resulting .h is nis.h.

304 RPCSVCPROTS = \
305     $(RPCSVC_GEN_HDRS:%.h=%.x)      nis_object.x      yp.x

307 LVMSVCPROTS = \
308     $(LVMPCHDRS:%.h=%.x)

310 RPCSVCHDRS= $(RPCSVC_SRC_HDRS) $(RPCSVC_GEN_HDRS)

312 PROTOHDRS=  dumprestore.h routed.h ripngd.h rwhod.h timed.h

314 ROOTHDRS= $(HDRS:%=$(ROOT)/usr/include/%) \
315     $(KRB5HDRS:%=$(ROOT)/usr/include/kerberos5/%) \
316     $(ISOHDRS:%=$(ROOT)/usr/include/iso/%) \
317     $(ARPAHDRS:%=$(ROOT)/usr/include/arpa/%) \
318     $(AUDIOHDRS:%=$(ROOT)/usr/include/audio/%) \
319     $(UIDHDRS:%=$(ROOT)/usr/include/uuid/%) \
320     $(RPCSVCHDRS:%=$(ROOT)/usr/include/rpcsvc/%) \

```

5

## new/usr/src/head/Makefile

```

321     $(RPCSVCPROTS:%=$(ROOT)/usr/include/rpcsvc/%) \
322     $(LVMPCHDRS:%=$(ROOT)/usr/include/%) \
323     $(PROTOHDRS:%=$(ROOT)/usr/include/protocols/%)

325 DIRS= iso arpa audio rpcsvc protocols security uuid kerberos5
326 ROOTDIRS= $(DIRS:%=$(ROOT)/usr/include/%)

328 SED= sed

330 # check files really don't exist
331 #
332 # should do something with the rpcsvc headers

334 iso/%.check:      iso/%.h
335     $(DOT_H_CHECK)

337 arpa/%.check:     arpa/%.h
338     $(DOT_H_CHECK)

340 audio/%.check:    audio/%.h
341     $(DOT_H_CHECK)

343 rpcsvc/%.check:   rpcsvc/%.h
344     $(DOT_H_CHECK)

346 rpcsvc/%.check:   rpcsvc/%.x
347     $(DOT_X_CHECK)

349 protocols/%.check: protocols/%.h
350     $(DOT_H_CHECK)

352 kerberos5/%.check: kerberos5/%.h
353     $(DOT_H_CHECK)

355 uuid/%.check:     uuid/%.h
356     $(DOT_H_CHECK)

358 # Note that the derived headers (rpcgen) are not checked at this time. These
359 # need work at the source level and rpcgen itself has a bug which causes a
360 # cstyle violation. Furthermore, there seems to be good reasons for the
361 # generated headers to not pass all of the hdrchk rules.
362 #
363 # Add the following to the CHECKHDRS list to activate the .x checks:
364 #     $(RPCSVCPROTS:%.x=rpcsvc/%.check) \
365 #
366 CHECKHDRS= $(HDRS:%.h=%.check) \
367     $(KRB5HDRS:%.h=kerberos5/%.check) \
368     $(ISOHDRS:%.h=iso/%.check) \
369     $(ARPAHDRS:%.h=arpa/%.check) \
370     $(AUDIOHDRS:%.h=audio/%.check) \
371     $(UIDHDRS:%.h=uuid/%.check) \
372     $(RPCSVC_SRC_HDRS:%.h=rpcsvc/%.check) \
373     $(PROTOHDRS:%.h=protocols/%.check)

375 # headers which won't quite meet the standards...
376 #
377 # assert.h is required by ansi-c to *not* be idempotent (section 4.1.2).
378 # Hence the trailing guard is not the last thing in the file nor can it
379 # be without playing silly games.

381 assert.check      := HDRCHK_TAIL = | grep -v "end guard wrong" | true

383 # install rules

385 $(ROOT)/usr/include/security/%: security/%
386     $(INS.file)

```

6



new/usr/src/head/ctype.h

1

```
*****
3390 Fri Jul 11 11:42:52 2014
new/usr/src/head/ctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*      Copyright (c) 1988 AT&T */
23 /*      All Rights Reserved      */

26 /*
27 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
28 * Use is subject to license terms.
29 */
30 /*
31 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
32 */

34 #ifndef _CTYPE_H
35 #define _CTYPE_H

34 #pragma ident      "%Z%M% %I%      %E% SMI"

37 #include <iso/ctype_iso.h>
37 #include <iso/ctype_c99.h>

39 /*
40 * Allow global visibility for symbols defined in
41 * C++ "std" namespace in <iso/ctype_iso.h>.
42 */
43 #if __cplusplus >= 199711L
44 using std::isalnum;
45 using std::isalpha;
46 using std::iscntrl;
47 using std::isdigit;
48 using std::isgraph;
49 using std::islower;
50 using std::isprint;
51 using std::ispunct;
52 using std::isspace;
53 using std::isupper;
54 using std::isxdigit;
55 using std::tolower;
```

new/usr/src/head/ctype.h

2

```
56 using std::toupper;
57 #if __cplusplus >= 201103L
58 using std::isblank;
59 #endif
60 #endif

62 #ifdef __cplusplus
63 extern "C" {
64 #endif

66 #if defined(__STDC__)

68 #if defined(__EXTENSIONS__) || \
69     ((!defined(_STRICT_STDC) && !defined(_POSIX_C_SOURCE)) || \
70     defined(_XOPEN_SOURCE))

72 extern int isascii(int);
73 extern int toascii(int);
74 extern int _tolower(int);
75 extern int _toupper(int);

77 #endif /* defined(__EXTENSIONS__) || ((!defined(_STRICT_STDC) ... */

79 #if !defined(__lint)

81 #if defined(__EXTENSIONS__) || \
82     ((!defined(_STRICT_STDC) && !defined(_POSIX_C_SOURCE)) || \
83     defined(_XOPEN_SOURCE)) || defined(__XPG4_CHAR_CLASS__)
84 #define isascii(c)      (((int)(c) & ~0177)
85 #define toascii(c)     ((int)(c) & 0177)
86 #define _toupper(c)    (toupper(c))
87 #define _tolower(c)    (tolower(c))
88 #if defined(__XPG4_CHAR_CLASS__) || defined(_XPG4)
84 #define _toupper(c)    (__trans_upper[(int)(c)])
85 #define _tolower(c)    (__trans_lower[(int)(c)])
86 #else
87 #define _toupper(c)    ((__ctype + 258)[(int)(c)])
88 #define _tolower(c)    ((__ctype + 258)[(int)(c)])
89 #endif /* defined(__XPG4_CHAR_CLASS__) || defined(_XPG4) */

89 #endif /* defined(__EXTENSIONS__) || ((!defined(_STRICT_STDC) ... */

91 #endif /* !defined(__lint) */

93 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)

95 #ifndef _LOCALE_T
96 #define _LOCALE_T
97 typedef struct locale *locale_t;
98 #endif

100 extern int isalnum_l(int, locale_t);
101 extern int isalpha_l(int, locale_t);
102 extern int isblank_l(int, locale_t);
103 extern int iscntrl_l(int, locale_t);
104 extern int isdigit_l(int, locale_t);
105 extern int isgraph_l(int, locale_t);
106 extern int islower_l(int, locale_t);
107 extern int isprint_l(int, locale_t);
108 extern int ispunct_l(int, locale_t);
109 extern int isspace_l(int, locale_t);
110 extern int isupper_l(int, locale_t);
111 extern int isxdigit_l(int, locale_t);

113 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */
```



```
115 #else /* defined(__STDC__) */
117 #if !defined(__lint)
119 #define isascii(c)      (!(((int)(c)) & ~0177))
120 #define _toupper(c)    (isascii(c) ? __trans_upper[(int)(c)] : toupper(c))
121 #define _tolower(c)    (isascii(c) ? __trans_lower[(int)(c)] : tolower(c))
100 #define _toupper(c)    ((_ctype + 258)[(int)(c)])
101 #define _tolower(c)    ((_ctype + 258)[(int)(c)])
122 #define toascii(c)     (((int)(c)) & 0177)
124 #endif /* !defined(__lint) */
126 #endif /* defined(__STDC__) */
128 #ifdef __cplusplus
129 }
_____unchanged_portion_omitted_____
```

new/usr/src/head/iso/ctype\_iso.h

1

```
*****
3834 Fri Jul 11 11:42:52 2014
new/usr/src/head/iso/ctype_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*      Copyright (c) 1988 AT&T */
23 /*      All Rights Reserved      */

26 /*
27 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
28 * Use is subject to license terms.
29 */
30 /*
31 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
32 */

34 /*
35 * An application should not include this header directly. Instead it
36 * should be included only through the inclusion of other Sun headers.
37 *
38 * The contents of this header is limited to identifiers specified in the
39 * C Standard. Any new identifiers specified in future amendments to the
40 * C Standard must be placed in this header. If these new identifiers
41 * are required to also be in the C++ Standard "std" namespace, then for
42 * anything other than macro definitions, corresponding "using" directives
43 * must also be added to <ctype.h>.
44 */

46 #ifndef _ISO_CTYPE_ISO_H
47 #define _ISO_CTYPE_ISO_H

46 #pragma ident      "%Z%M% %I%      %E% SMI"

49 #include <sys/feature_tests.h>

51 #ifdef __cplusplus
52 extern "C" {
53 #endif

55 #define _U      0x00000001      /* Upper case */
56 #define _L      0x00000002      /* Lower case */
```

new/usr/src/head/iso/ctype\_iso.h

2

```
57 #define _N      0x00000004      /* Numeral (digit) */
58 #define _S      0x00000008      /* Spacing character */
59 #define _P      0x00000010      /* Punctuation */
60 #define _C      0x00000020      /* Control character */
61 #define _B      0x00000040      /* Blank */
62 #define _X      0x00000080      /* hexadecimal digit */

64 #define _ISUPPER      _U
65 #define _ISLOWER      _L
66 #define _ISDIGIT      _N
67 #define _ISSPACE      _S
68 #define _ISPUNCT      _P
69 #define _ISCNTRL      _C
70 #define _ISBLANK      _B
71 #define _ISXDIGIT      _X
72 #define _ISGRAPH      0x00002000
73 #define _ISALPHA      0x00004000
74 #define _ISPRINT      0x00008000
75 #define _ISALNUM      (_ISALPHA | _ISDIGIT)

77 extern unsigned char      __ctype[];
78 extern unsigned int      *__ctype_mask;
79 extern int      *__trans_upper;
80 extern int      *__trans_lower;

82 #if defined(__STDC__)

84 #if __cplusplus >= 199711L
85 namespace std {
86 #endif
79 #if __cplusplus < 199711L /* Use inline functions instead for ANSI C++ */

88 /*
89 * These used to be macros, which while more efficient, precludes operation
90 * with thread specific locales. The old macros will still work, but new
91 * code compiles to use functions. This is specifically permitted by the
92 * various standards. Only _tolower and _toupper were required to be
93 * delivered in macro form.
94 */
95 extern int isalnum(int);
96 extern int isalpha(int);
97 extern int iscntrl(int);
98 extern int isdigit(int);
99 extern int isgraph(int);
100 extern int islower(int);
101 extern int isprint(int);
102 extern int ispunct(int);
103 extern int isspace(int);
104 extern int isupper(int);
105 extern int isxdigit(int);
106 #if defined(_XPG6) || defined(_STDC_C99) || !defined(_STRICT_SYMBOLS)
107 extern int isblank(int);

93 #endif /* __cplusplus < 199711L */

95 #if __cplusplus >= 199711L
96 namespace std {
108 #endif

110 extern int tolower(int);
111 extern int toupper(int);

113 #if __cplusplus >= 199711L
114 } /* end of namespace std */
115 #endif
```

```

106 extern unsigned char  __ctype[];
107 extern unsigned int   *__ctype_mask;
108 extern int             *__trans_upper;
109 extern int             *__trans_lower;

111 #if !defined(__lint)

113 #if __cplusplus >= 199711L
114 namespace std {

116 #if defined(__XPG4_CHAR_CLASS__) || defined(_XPG4)

118 inline int isalpha(int c) { return (__ctype_mask[c] & _ISALPHA); }
119 inline int isupper(int c) { return (__ctype_mask[c] & _ISUPPER); }
120 inline int islower(int c) { return (__ctype_mask[c] & _ISLOWER); }
121 inline int isdigit(int c) { return (__ctype_mask[c] & _ISDIGIT); }
122 inline int isxdigit(int c) { return (__ctype_mask[c] & _ISXDIGIT); }
123 inline int isalnum(int c) { return (__ctype_mask[c] & _ISALNUM); }
124 inline int isspace(int c) { return (__ctype_mask[c] & _ISSPACE); }
125 inline int ispunct(int c) { return (__ctype_mask[c] & _ISPUNCT); }
126 inline int isprint(int c) { return (__ctype_mask[c] & _ISPRINT); }
127 inline int isgraph(int c) { return (__ctype_mask[c] & _ISGRAPH); }
128 inline int iscntrl(int c) { return (__ctype_mask[c] & _ISCNTRL); }
129 #else
130 inline int isalpha(int c) { return ((__ctype + 1)[c] & (_U | _L)); }
131 inline int isupper(int c) { return ((__ctype + 1)[c] & _U); }
132 inline int islower(int c) { return ((__ctype + 1)[c] & _L); }
133 inline int isdigit(int c) { return ((__ctype + 1)[c] & _N); }
134 inline int isxdigit(int c) { return ((__ctype + 1)[c] & _X); }
135 inline int isalnum(int c) { return ((__ctype + 1)[c] & (_U | _L | _N)); }
136 inline int isspace(int c) { return ((__ctype + 1)[c] & _S); }
137 inline int ispunct(int c) { return ((__ctype + 1)[c] & _P); }
138 inline int isprint(int c) {
139     return ((__ctype + 1)[c] & (_P | _U | _L | _N | _B)); }
140 inline int isgraph(int c) { return ((__ctype + 1)[c] & (_P | _U | _L | _N)); }
141 inline int iscntrl(int c) { return ((__ctype + 1)[c] & _C); }
142 #endif /* defined(__XPG4_CHAR_CLASS__) || defined(_XPG4) */

144 } /* end of namespace std */

146 #else /* __cplusplus >= 199711L */

148 #if defined(__XPG4_CHAR_CLASS__) || defined(_XPG4)
149 #define isalpha(c)      (__ctype_mask[(int)(c)] & _ISALPHA)
150 #define isupper(c)     (__ctype_mask[(int)(c)] & _ISUPPER)
151 #define islower(c)     (__ctype_mask[(int)(c)] & _ISLOWER)
152 #define isdigit(c)     (__ctype_mask[(int)(c)] & _ISDIGIT)
153 #define isxdigit(c)   (__ctype_mask[(int)(c)] & _ISXDIGIT)
154 #define isalnum(c)    (__ctype_mask[(int)(c)] & _ISALNUM)
155 #define isspace(c)    (__ctype_mask[(int)(c)] & _ISSPACE)
156 #define ispunct(c)    (__ctype_mask[(int)(c)] & _ISPUNCT)
157 #define isprint(c)    (__ctype_mask[(int)(c)] & _ISPRINT)
158 #define isgraph(c)    (__ctype_mask[(int)(c)] & _ISGRAPH)
159 #define iscntrl(c)    (__ctype_mask[(int)(c)] & _ISCNTRL)
160 #else
161 #define isalpha(c)     ((__ctype + 1)[(int)(c)] & (_U | _L))
162 #define isupper(c)    ((__ctype + 1)[(int)(c)] & _U)
163 #define islower(c)    ((__ctype + 1)[(int)(c)] & _L)
164 #define isdigit(c)    ((__ctype + 1)[(int)(c)] & _N)
165 #define isxdigit(c)  ((__ctype + 1)[(int)(c)] & _X)
166 #define isalnum(c)   ((__ctype + 1)[(int)(c)] & (_U | _L | _N))
167 #define isspace(c)   ((__ctype + 1)[(int)(c)] & _S)
168 #define ispunct(c)   ((__ctype + 1)[(int)(c)] & _P)
169 #define isprint(c)   ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N | _B))
170 #define isgraph(c)   ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N))
171 #define iscntrl(c)   ((__ctype + 1)[(int)(c)] & _C)

```

```

173 #endif /* defined(__XPG4_CHAR_CLASS__) || defined(_XPG4) */

175 #endif /* __cplusplus >= 199711L */

177 #endif /* !defined(__lint) */

179 #else /* defined(__STDC__) */

181 extern unsigned char  __ctype[];

183 #if !defined(__lint)

185 #define isalpha(c)      ((__ctype + 1)[(int)(c)] & (_U | _L))
186 #define isupper(c)     ((__ctype + 1)[(int)(c)] & _U)
187 #define islower(c)     ((__ctype + 1)[(int)(c)] & _L)
188 #define isdigit(c)     ((__ctype + 1)[(int)(c)] & _N)
189 #define isxdigit(c)   ((__ctype + 1)[(int)(c)] & _X)
190 #define isalnum(c)    ((__ctype + 1)[(int)(c)] & (_U | _L | _N))
191 #define isspace(c)    ((__ctype + 1)[(int)(c)] & _S)
192 #define ispunct(c)    ((__ctype + 1)[(int)(c)] & _P)
193 #define isprint(c)    ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N | _B))
194 #define isgraph(c)    ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N))
195 #define iscntrl(c)    ((__ctype + 1)[(int)(c)] & _C)

197 #endif /* !defined(__lint) */

199 #endif /* defined(__STDC__) */

201 #ifdef __cplusplus
202 }
203 #endif

```

---

unchanged portion omitted

new/usr/src/head/iso/stdlib\_iso.h

1

```
*****
6072 Fri Jul 11 11:42:53 2014
new/usr/src/head/iso/stdlib_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
24  * Use is subject to license terms.
25 */
26 /*
27  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
28 */

30 /*      Copyright (c) 1988 AT&T */
31 /*      All Rights Reserved      */

34 /*
35  * An application should not include this header directly. Instead it
36  * should be included only through the inclusion of other Sun headers.
37  *
38  * The contents of this header is limited to identifiers specified in the
39  * C Standard. Any new identifiers specified in future amendments to the
40  * C Standard must be placed in this header. If these new identifiers
41  * are required to also be in the C++ Standard "std" namespace, then for
42  * anything other than macro definitions, corresponding "using" directives
43  * must also be added to <locale.h>.
44  */

46 #ifndef _ISO_STDLIB_ISO_H
47 #define _ISO_STDLIB_ISO_H

49 #include <sys/feature_tests.h>

51 #ifdef __cplusplus
52 extern "C" {
53 #endif

55 #if defined(__STDC__)
56 unsigned char __mb_cur_max(void);
53 extern unsigned char __ctype[];
54 #define MB_CUR_MAX __ctype[520]
```

new/usr/src/head/iso/stdlib\_iso.h

2

```
57 #else
58 unsigned char __mb_cur_max();
56 extern unsigned char __ctype[];
57 #define MB_CUR_MAX __ctype[520]
59 #endif
60 #ifndef MB_CUR_MAX
61 #define MB_CUR_MAX (__mb_cur_max())
62 #endif

64 #if __cplusplus >= 199711L
65 namespace std {
66 #endif

68 typedef struct {
69     int    quot;
70     int    rem;
71 } div_t;
_____ unchanged_portion_omitted_____
```

new/usr/src/head/iso/wctype\_iso.h

1

```
*****
4735 Fri Jul 11 11:42:53 2014
new/usr/src/head/iso/wctype_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*      wctype.h      1.13 89/11/02 SMI; JLE */
23 /*      from AT&T JAE 2.1      */
24 /*      definitions for international functions */

26 /*
27  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
28  * Use is subject to license terms.
29  */
30 /*
31  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
32  */

34 /*
35  * An application should not include this header directly. Instead it
36  * should be included only through the inclusion of other Sun headers.
37  *
38  * The contents of this header is limited to identifiers specified in the
39  * C Standard. Any new identifiers specified in future amendments to the
40  * C Standard must be placed in this header. If these new identifiers
41  * are required to also be in the C++ Standard "std" namespace, then for
42  * anything other than macro definitions, corresponding "using" directives
43  * must also be added to <wctype.h>.
44  */

46 #ifndef _ISO_WCTYPE_ISO_H
47 #define _ISO_WCTYPE_ISO_H

48 #pragma ident      "%Z%M% %I%      %E% SMI"

49 #include <sys/feature_tests.h>

51 #ifdef __cplusplus
52 extern "C" {
53 #endif

55 #if __cplusplus >= 199711L
56 namespace std {
```

new/usr/src/head/iso/wctype\_iso.h

2

```
57 #endif

59 #if !defined(_WINT_T) || __cplusplus >= 199711L
60 #define _WINT_T
61 #if defined(_LP64)
62 typedef int      wint_t;
63 #else
64 typedef long     wint_t;
65 #endif
66 #endif /* !defined(_WINT_T) || __cplusplus >= 199711L */

68 #if !defined(_WCTYPE_T) || __cplusplus >= 199711L
69 #define _WCTYPE_T
70 typedef int      wctype_t;
71 #endif

73 typedef unsigned int      wctrans_t;

75 /* not XPG4 and not XPG4v2 */
76 #if !defined(_XPG4) || defined(_XPG5)
77 #ifndef WEOF
78 #if __cplusplus >= 199711L
79 #define WEOF      ((std::wint_t)(-1))
80 #else
81 #define WEOF      ((wint_t)(-1))
82 #endif
83 #endif /* WEOF */
84 #endif /* not XPG4 and not XPG4v2 */

86 #ifdef __STDC__
87 extern int iswalnum(wint_t);
88 extern int iswalphabeta(wint_t);
89 extern int iswcntrl(wint_t);
90 extern int iswdigit(wint_t);
91 extern int iswgraph(wint_t);
92 extern int iswlower(wint_t);
93 extern int iswprint(wint_t);
94 extern int iswpunct(wint_t);
95 extern int iswspace(wint_t);
96 extern int iswupper(wint_t);
97 extern int iswxdigit(wint_t);

99 #if (__cplusplus >= 201103L) || defined(_STDC_C99) || defined(_XPG6) || \
100 !defined(_STRICT_SYMBOLS)
101 extern int iswblank(wint_t);
102 #endif

104 /* tow* also become functions */
105 extern wint_t towlower(wint_t);
106 extern wint_t towupper(wint_t);
107 extern wctrans_t wctrans(const char *);
108 extern wint_t towctrans(wint_t, wctrans_t);
109 extern int iswctype(wint_t, wctype_t);
110 extern wctype_t wctype(const char *);
111 #else /* __STDC__ */
112 #endif
112 extern int iswalnum();
113 extern int iswalphabeta();
114 extern int iswcntrl();
115 extern int iswdigit();
116 extern int iswgraph();
117 extern int iswlower();
118 extern int iswprint();
119 extern int iswpunct();
120 extern int iswspace();
121 extern int iswupper();
```

```
122 extern int iswxdigit();
123 /* tow* also become functions */
124 extern wint_t towlower();
125 extern wint_t towupper();
126 extern wctrans_t wctrans();
127 extern wint_t towctrans();
128 extern int iswctype();
129 extern wctype_t wctype();
130 #if defined(_XPG6) || !defined(_STRICT_SYMBOLS)
131 extern int iswblank();
132 #endif
133 #endif /* __STDC__ */

135 /* bit definition for character class */

137 #define _E1 0x00000100 /* phonogram (international use) */
138 #define _E2 0x00000200 /* ideogram (international use) */
139 #define _E3 0x00000400 /* English (international use) */
140 #define _E4 0x00000800 /* number (international use) */
141 #define _E5 0x00001000 /* special (international use) */
142 #define _E6 0x00002000 /* other characters (international use) */
143 #define _E7 0x00004000 /* reserved (international use) */
144 #define _E8 0x00008000 /* reserved (international use) */

146 #define _E9 0x00010000
147 #define _E10 0x00020000
148 #define _E11 0x00040000
149 #define _E12 0x00080000
150 #define _E13 0x00100000
151 #define _E14 0x00200000
152 #define _E15 0x00400000
153 #define _E16 0x00800000
154 #define _E17 0x01000000
155 #define _E18 0x02000000
156 #define _E19 0x04000000
157 #define _E20 0x08000000
158 #define _E21 0x10000000
159 #define _E22 0x20000000
160 #define _E23 0x40000000
161 #define _E24 0x80000000

163 #if __cplusplus >= 199711L
164 }
_____unchanged_portion_omitted_____
```

new/usr/src/head/langinfo.h

1

```
*****
4697 Fri Jul 11 11:42:53 2014
new/usr/src/head/langinfo.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License").  You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2003 Sun Microsystems, Inc.  All rights reserved.
24  * Use is subject to license terms.
25 */

27 /*      Copyright (c) 1988 AT&T */
28 /*      All Rights Reserved */

30 /*
31  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
32  */

35 #ifndef _LANGINFO_H
36 #define _LANGINFO_H

34 #pragma ident      "%Z%M% %I%      %E% SMI"      /* SVr4.0 1.2 */

38 #include <sys/feature_tests.h>
39 #include <nl_types.h>

41 #ifdef __cplusplus
42 extern "C" {
43 #endif

45 /*
46  * The seven days of the week in their full beauty
47  */

49 #define DAY_1      1      /* sunday */
50 #define DAY_2      2      /* monday */
51 #define DAY_3      3      /* tuesday */
52 #define DAY_4      4      /* wednesday */
53 #define DAY_5      5      /* thursday */
54 #define DAY_6      6      /* friday */
55 #define DAY_7      7      /* saturday */
```

new/usr/src/head/langinfo.h

2

```
57 /*
58  * The abbreviated seven days of the week
59  */

61 #define ABDAY_1      8      /* sun */
62 #define ABDAY_2      9      /* mon */
63 #define ABDAY_3      10     /* tue */
64 #define ABDAY_4      11     /* wed */
65 #define ABDAY_5      12     /* thu */
66 #define ABDAY_6      13     /* fri */
67 #define ABDAY_7      14     /* sat */

69 /*
70  * The full names of the twelve months...
71  */

73 #define MON_1      15     /* january */
74 #define MON_2      16     /* february */
75 #define MON_3      17     /* march */
76 #define MON_4      18     /* april */
77 #define MON_5      19     /* may */
78 #define MON_6      20     /* june */
79 #define MON_7      21     /* july */
80 #define MON_8      22     /* august */
81 #define MON_9      23     /* september */
82 #define MON_10     24     /* october */
83 #define MON_11     25     /* november */
84 #define MON_12     26     /* december */

86 /*
87  * ... and their abbreviated form
88  */

90 #define ABMON_1      27     /* jan */
91 #define ABMON_2      28     /* feb */
92 #define ABMON_3      29     /* mar */
93 #define ABMON_4      30     /* apr */
94 #define ABMON_5      31     /* may */
95 #define ABMON_6      32     /* jun */
96 #define ABMON_7      33     /* jul */
97 #define ABMON_8      34     /* aug */
98 #define ABMON_9      35     /* sep */
99 #define ABMON_10     36     /* oct */
100 #define ABMON_11     37     /* nov */
101 #define ABMON_12     38     /* dec */

103 /*
104  * plus some special strings you might need to know
105  */

107 #define RADIXCHAR 39      /* radix character */
108 #define THOUSEP   40      /* separator for thousand */
109 /* YESSTR and NOSTR marked as legacy in XPG5 and removed in SUSv3 */
110 #if !defined(_XPG6) || defined(__EXTENSIONS__)
111 #define YESSTR     41      /* affirmative response for yes/no queries */
112 #define NOSTR     42      /* negative response for yes/no queries */
113 #endif /* !defined(_XPG6) || defined(__EXTENSIONS__) */
114 #define CRNCYSTR  43      /* currency symbol */

116 /*
117  * Default string used to format date and time
118  * e.g. Sunday, August 24 21:08:38 MET 1986
119  */

121 #define D_T_FMT    44      /* string for formatting date and time */
122 #define D_FMT      45      /* date format */
```

```
123 #define T_FMT      46    /* time format */
124 #define AM_STR     47    /* am string */
125 #define PM_STR     48    /* pm string */

127 /*
128  * Additions for XPG4 (XSH4) Compliance
129  */

131 #define CODESET     49    /* codeset name */
132 #define T_FMT_AMPM 50    /* am or pm time format string */
133 #define ERA         51    /* era description segments */
134 #define ERA_D_FMT   52    /* era date format string */
135 #define ERA_D_T_FMT 53    /* era date and time format string */
136 #define ERA_T_FMT   54    /* era time format string */
137 #define ALT_DIGITS  55    /* alternative symbols for digits */
138 #define YESEXPR     56    /* affirmative response expression */
139 #define NOEXPR      57    /* negative response expression */
140 #define _DATE_FMT   58    /* strftime format for date(1) */

142 #if defined(__EXTENSIONS__) || !defined(_XOPEN_SOURCE)
143 #define MAXSTRMSG   58 /* Maximum number of strings in langinfo */
144 #endif /* defined(__EXTENSIONS__) || !defined(_XOPEN_SOURCE) */

146 /*
147  * and the definitions of functions langinfo(3C)
148  */
149 #if defined(__STDC__)
150 char *nl_langinfo(nl_item); /* get a string from the database */
151 #else
152 char *nl_langinfo(); /* get a string from the database */
153 #endif

155 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
156 #ifndef _LOCALE_T
157 #define _LOCALE_T
158 typedef struct locale *locale_t;
159 #endif

161 #if defined(__STDC__)
162 char *nl_langinfo_l(nl_item, locale_t);
163 #else
164 char *nl_langinfo_l();
165 #endif
166 #endif

168 #ifdef __cplusplus
169 }
    unchanged_portion_omitted

```



new/usr/src/head/locale.h

1

```
*****
3046 Fri Jul 11 11:42:53 2014
new/usr/src/head/locale.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
24  * Use is subject to license terms.
25 */
27 /*
28  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
29  *
30  * Portions of this file developed by Garrett D'Amore are licensed
31  * under the terms of the Common Development and Distribution License (CDDL)
32  * version 1.0 only. The use of subsequent versions of the License are
33  * is specifically prohibited unless those terms are not in conflict with
34  * version 1.0 of the License. You can find this license on-line at
35  * http://www.illumos.org/license/CDDL
36  */
38 #ifndef _LOCALE_H
39 #define _LOCALE_H
41 #include <iso/locale_iso.h>
43 #if (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
44     defined(__EXTENSIONS__)
45 #include <libintl.h>
46 #endif
48 /*
49  * Allow global visibility for symbols defined in
50  * C++ "std" namespace in <iso/locale_iso.h>.
51  */
52 #if __cplusplus >= 199711L
53 using std::lconv;
54 using std::setlocale;
55 using std::localeconv;
56 #endif
```

new/usr/src/head/locale.h

2

```
58 #ifdef __cplusplus
59 extern "C" {
60 #endif
62 #define _LastCategory LC_MESSAGES /* This must be last category */
64 #define _ValidCategory(c) \
65     (((int)(c) >= LC_CTYPE) && ((int)(c) <= _LastCategory) || \
66     ((int)c == LC_ALL))
69 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
71 /*
72  * These were added in POSIX 2008 as part of the newlocale() specification.
73  */
74 #define LC_CTYPE_MASK (1 << LC_CTYPE)
75 #define LC_NUMERIC_MASK (1 << LC_NUMERIC)
76 #define LC_TIME_MASK (1 << LC_TIME)
77 #define LC_COLLATE_MASK (1 << LC_COLLATE)
78 #define LC_MONETARY_MASK (1 << LC_MONETARY)
79 #define LC_MESSAGES_MASK (1 << LC_MESSAGES)
80 #define LC_ALL_MASK (0x3f)
82 #ifndef _LOCALE_T
83 #define _LOCALE_T
84 typedef struct locale *locale_t;
85 #endif
87 #if defined(__STDC__)
88 extern locale_t duplocale(locale_t);
89 extern void freelocale(locale_t);
90 extern locale_t newlocale(int, const char *, locale_t);
91 extern locale_t uselocale(locale_t);
92 #else /* __STDC__ */
93 extern locale_t duplocale();
94 extern void freelocale();
95 extern locale_t newlocale();
96 extern locale_t uselocale();
97 #endif /* __STDC__ */
99 #define LC_GLOBAL_LOCALE (__global_locale())
100 extern locale_t __global_locale(void);
102 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */
104 #ifdef __cplusplus
105 }
__unchanged_portion_omitted_
```

new/usr/src/head/monetary.h

1

```
*****
1626 Fri Jul 11 11:42:53 2014
new/usr/src/head/monetary.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright 2003 Sun Microsystems, Inc. All rights reserved.
24  * Use is subject to license terms.
25 */
26 /*
27  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
28 */

30 #ifndef _MONETARY_H
31 #define _MONETARY_H

32 #pragma ident "%Z%M% %I% %E% SMI"

33 #include <sys/feature_tests.h>
34 #include <sys/types.h>

36 #ifdef __cplusplus
37 extern "C" {
38 #endif

40 #if defined(__STDC__)
41 extern ssize_t strfmon(char *_RESTRICT_KYWD, size_t,
42                      const char *_RESTRICT_KYWD, ...);

44 #if defined(_XPG7) || (!defined(_STRICT_STRICT_SYMBOLS))

46 #ifndef _LOCALE_T
47 #define _LOCALE_T
48 typedef struct locale *locale_t;
49 #endif

51 extern ssize_t strfmon_l(char *_RESTRICT_KYWD, size_t, locale_t,
52                        const char *_RESTRICT_KYWD, ...);
53 #endif

55 #else
56 extern ssize_t strfmon();
```

new/usr/src/head/monetary.h

2

```
57 extern ssize_t strfmon_l();
58 #endif

60 #ifdef __cplusplus
61 }
_____ unchanged_portion_omitted_____
```

new/usr/src/head/string.h

1

```
*****
7182 Fri Jul 11 11:42:53 2014
new/usr/src/head/string.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
24  * Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
25  */
26
27 /*      Copyright (c) 1988 AT&T */
28 /*      All Rights Reserved      */
29
30 #ifndef _STRING_H
31 #define _STRING_H
32
33 #include <iso/string_iso.h>
34
35 /*
36  * Allow global visibility for symbols defined in
37  * C++ "std" namespace in <iso/string_iso.h>.
38  */
39 #if __cplusplus >= 199711L
40 using std::size_t;
41 using std::memchr;
42 using std::memcmp;
43 using std::memcpy;
44 using std::memmove;
45 using std::memset;
46 using std::strcat;
47 using std::strchr;
48 using std::strcmp;
49 using std::strcoll;
50 using std::strcpy;
51 using std::strcspn;
52 using std::strerror;
53 using std::strlen;
54 using std::strncat;
55 using std::strncpy;
56 using std::strncpy;
57 using std::strpbrk;
58 using std::strrchr;
```

new/usr/src/head/string.h

2

```
59 using std::strspn;
60 using std::strstr;
61 using std::strtok;
62 using std::strxfrm;
63 #endif
64
65 #ifdef __cplusplus
66 extern "C" {
67 #endif
68
69 #if defined(__STDC__)
70
71 #if defined(__EXTENSIONS__) || \
72     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
73     defined(_XPG6) || defined(_REENTRANT)
74 extern int strerror_r(int, char *, size_t);
75 #endif
76
77 #if defined(__EXTENSIONS__) || \
78     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
79     (_POSIX_C_SOURCE - 0 >= 199506L) || defined(_REENTRANT)
80 extern char *strtok_r(char *_RESTRICT_KYWD, const char *_RESTRICT_KYWD,
81                     char **_RESTRICT_KYWD);
82 #endif
83
84 #if defined(__EXTENSIONS__) || !defined(_STRICT_STDC) || \
85     defined(__XOPEN_OR_POSIX)
86 extern void *memcpy(void *_RESTRICT_KYWD, const void *_RESTRICT_KYWD,
87                  int, size_t);
88 #endif
89
90 #if !defined(_STRICT_SYMBOLS) || defined(_XPG7)
91
92 #if defined(__EXTENSIONS__) || \
93     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
94 /* || defined(_XPG7) */
95 extern int strcasecmp(const char *, const char *);
96 extern int strncasecmp(const char *, const char *, size_t);
97 extern char *stpcpy(char *_RESTRICT_KYWD, const char *_RESTRICT_KYWD);
98 extern char *stpncpy(char *_RESTRICT_KYWD, const char *_RESTRICT_KYWD, size_t);
99 extern char *strndup(const char *, size_t);
100 extern size_t strlen(const char *, size_t);
101 extern char *strsignal(int);
102
103 #ifndef _LOCALE_T
104 #define _LOCALE_T
105 typedef struct locale *locale_t;
106 #endif
107
108 extern int strcoll_l(const char *, const char *, locale_t);
109 extern size_t strxfrm_l(char *_RESTRICT_KYWD, const char *_RESTRICT_KYWD,
110                        size_t, locale_t);
111 extern int strcasecmp_l(const char *, const char *, locale_t);
112 extern int strncasecmp_l(const char *, const char *, size_t, locale_t);
113
114 #endif /* defined(_STRICT_SYMBOLS) || defined(_XPG7) */
115
116 #if !defined(_STRICT_SYMBOLS)
117
118 /* Note that some of these are also declared in strings.h for XPG4_2+ */
119 #if defined(__EXTENSIONS__) || \
120     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
121 extern int ucopy(const void *_RESTRICT_KYWD, void *_RESTRICT_KYWD, size_t);
122 extern int ucopystr(const void *_RESTRICT_KYWD, void *_RESTRICT_KYWD, size_t);
123 extern int ffs(int);
124 extern int ffs_l(long);
```

```

118 extern int ffsll(long long);
119 extern int fls(int);
120 extern int flsl(long);
121 extern int flsll(long long);
122 extern void *memmem(const void *, size_t, const void *, size_t);
123 extern char *strcasestr(const char *, const char *);
124 extern char *strnstr(const char *, const char *, size_t);
125 extern size_t strlcpy(char *, const char *, size_t);
126 extern size_t strlcat(char *, const char *, size_t);
127 extern char *strsep(char **stringp, const char *delim);
128 extern char *strchrnul(const char *, int);
129 extern char *strcasestr_l(const char *, const char *, locale_t);
130 extern int strcasecmp(const char *, const char *);
131 extern int strncasecmp(const char *, const char *, size_t);
132 #endif /* defined(__EXTENSIONS__)... */

134 #if defined(__EXTENSIONS__) || \
135     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
136     defined(_XPG4_2)
137 extern char *strdup(const char *);
138 #endif

140 #if defined(__EXTENSIONS__) || \
141     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
142 #if defined(__GNUC__)

144 /*
145  * gcc provides this inlining facility but Studio C does not.
146  * We should use it exclusively once Studio C also provides it.
147  */
148 extern void *__builtin_alloca(size_t);

150 #define strdupa(s) \
151     { \
152         __extension__ \
153         char *__str = (char *) (s); \
154         strcpy((char *) __builtin_alloca(strlen(__str) + 1), __str); \
155     }

157 #define strndupa(s, n) \
158     { \
159         __extension__ \
160         char *__str = (char *) (s); \
161         size_t __len = strlen(__str, (n)); \
162         (__str = strncpy((char *) __builtin_alloca(__len + 1), \
163             __str, __len), \
164             __str[__len] = '\0', __str); \
165     }

167 #else /* __GNUC__ */

169 #if defined(unix) /* excludes c99 */
170 /*
171  * Studio C currently can't do the gcc-style inlining,
172  * so we use thread-local storage instead.
173  */
174 extern void *__builtin_alloca(size_t);
175 extern __thread char *__strdupa_str;
176 extern __thread size_t __strdupa_len;

178 #define strdupa(s) \
179     (__strdupa_str = (char *) (s), \
180     strcpy((char *) __builtin_alloca(strlen(__strdupa_str) + 1), \
181         __strdupa_str))

183 #define strndupa(s, n) \

```

```

184     (__strdupa_str = (char *) (s), \
185     __strdupa_len = strlen(__strdupa_str, (n)), \
186     __strdupa_str = strncpy((char *) __builtin_alloca(__strdupa_len + 1), \
187         __strdupa_str, __strdupa_len), \
188     __strdupa_str[__strdupa_len] = '\0', __strdupa_str)
189 #endif /* unix */

191 #endif /* __GNUC__ */
192 #endif /* __EXTENSIONS__ ... */

194 #else /* __STDC__ */

196 #if defined(__EXTENSIONS__) || !defined(__XOPEN_OR_POSIX) || \
197     defined(_XPG6) || defined(_REENTRANT)
198 extern int strerror_r();
199 #endif

201 #if defined(__EXTENSIONS__) || !defined(__XOPEN_OR_POSIX) || \
202     (_POSIX_C_SOURCE - 0 >= 199506L) || defined(_REENTRANT)
203 extern char *strtok_r();
204 #endif

206 #if defined(__EXTENSIONS__) || !defined(_STRICT_STDC) || \
207     defined(__XOPEN_OR_POSIX)
208 extern void *memcpy();
209 #endif

211 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
212 #if defined(__EXTENSIONS__) || \
213     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
214     /* || defined(_XPG7) */
215 extern int strcasecmp();
216 extern int strncasecmp();
217 extern char *stpcpy();
218 extern char *stpncpy();
219 extern char *strndup();
220 extern size_t strlen();
221 #endif

223 #if !defined(_STRICT_SYMBOLS)
224 #if defined(__EXTENSIONS__) || \
225     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
226 extern int ucopy();
227 extern int ucopystr();
228 extern int ffs();
229 extern int ffsll();
230 extern int fls();
231 extern int flsll();
232 extern char *strcasestr();
233 extern char *strcasestr_l();
234 extern char *strnstr();
235 extern size_t strlcpy();
236 extern size_t strlcat();
237 extern char *strsep();
238 extern char *strchrnul();
239 #endif /* _STRICT_SYMBOLS */
240 #endif /* defined(__EXTENSIONS__) ... */

241 #if defined(__EXTENSIONS__) || !defined(__XOPEN_OR_POSIX) || defined(_XPG4_2)
242 extern char *strdup();
243 #endif

```

```
245 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
246 extern size_t strcoll_l();
247 extern size_t strxfrm_l();
248 #endif
250 #endif /* __STDC__ */
252 #ifdef __cplusplus
253 }
_____ unchanged_portion_omitted
```

new/usr/src/head/strings.h

1

```
*****
2905 Fri Jul 11 11:42:53 2014
new/usr/src/head/strings.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*
23  * Copyright (c) 1995, 1996, by Sun Microsystems, Inc.
24  * All rights reserved.
25 */
26 /*
27  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
28 */

30 #ifndef _STRINGS_H
31 #define _STRINGS_H

30 #pragma ident      "%Z%M% %I%      %E% SMI"

33 #include <sys/types.h>
34 #include <sys/feature_tests.h>

36 #if !defined(_XOPEN_SOURCE) || defined(__EXTENSIONS__)
37 #include <string.h>
38 #endif

40 #ifdef __cplusplus
41 extern "C" {
42 #endif

44 #if defined(__STDC__)

46 extern int bcmp(const void *, const void *, size_t);
47 extern void bcopy(const void *, void *, size_t);
48 extern void bzero(void *, size_t);

50 extern char *index(const char *, int);
51 extern char *rindex(const char *, int);

53 /*
54  * X/Open System Interfaces and Headers, Issue 4, Version 2, defines
55  * both <string.h> and <strings.h>. The namespace requirements
56  * do not permit the visibility of anything other than what is
```

new/usr/src/head/strings.h

2

```
57  * specifically defined for each of these headers. As a result,
58  * inclusion of <string.h> would result in declarations not allowed
59  * in <strings.h>, and making the following prototypes visible for
60  * anything other than X/Open UNIX Extension would result in
61  * conflicts with what is now in <string.h>.
62  */
63 #if defined(_XPG4_2) && !defined(__EXTENSIONS__)
64 extern int ffs(int);
65 extern int strcasecmp(const char *, const char *);
66 extern int strncasecmp(const char *, const char *, size_t);
67 #if defined(_XPG7)
68 #ifndef _LOCALE_T
69 #define _LOCALE_T
70 typedef struct locale *locale_t;
71 #endif
72 extern int strcasecmp_l(const char *, const char *, locale_t);
73 extern int strncasecmp_l(const char *, const char *, size_t, locale_t);
74 #endif /* defined(_XPG7) */
75 #endif /* defined(_XPG4_2) && !defined(__EXTENSIONS__) */

77 #else

79 extern int bcmp();
80 extern void bcopy();
81 extern void bzero();

83 extern char *index();
84 extern char *rindex();

86 #if defined(_XPG4_2) && !defined(__EXTENSIONS__)
87 extern int ffs();
88 extern int strcasecmp();
89 extern int strncasecmp();
90 #if defined(_XPG7)
91 extern int strcasecmp_l();
92 extern int strncasecmp_l();
93 #endif
94 #endif /* defined(_XPG4_2) && !defined(__EXTENSIONS__) */

96 #endif /* __STDC__ */

98 #ifdef __cplusplus
99 }
_____unchanged_portion_omitted_____
```

new/usr/src/head/time.h

1

```
*****
12091 Fri Jul 11 11:42:54 2014
new/usr/src/head/time.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*      Copyright (c) 1988 AT&T */
22 /*      All Rights Reserved      */

25 /*
26 * Copyright 2007 Sun Microsystems, Inc.  All rights reserved.
27 * Use is subject to license terms.
28 */
29 /*
30 * Copyright 2010 Nexenta Systems, Inc.  All rights reserved.
31 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
32 */

34 #ifndef _TIME_H
35 #define _TIME_H

37 #include <sys/feature_tests.h>
38 #include <iso/time_iso.h>
39 #if (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
40     (_POSIX_C_SOURCE > 2) || defined(__EXTENSIONS__)
41 #include <sys/types.h>
42 #include <sys/time_impl.h>
43 #endif /* (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) ... */

45 /*
46 * Allow global visibility for symbols defined in
47 * C++ "std" namespace in <iso/time_iso.h>.
48 */
49 #if __cplusplus >= 199711L
50 using std::size_t;
51 using std::clock_t;
52 using std::time_t;
53 using std::tm;
54 using std::asctime;
55 using std::clock;
56 using std::ctime;
57 using std::difftime;
58 using std::gmtime;
```

new/usr/src/head/time.h

2

```
59 using std::localtime;
60 using std::mktime;
61 using std::time;
62 using std::strftime;
63 #endif

65 #ifdef __cplusplus
66 extern "C" {
67 #endif

69 #ifndef _CLOCKID_T
70 #define _CLOCKID_T
71 typedef int    clockid_t;
72 #endif

74 #ifndef _TIMER_T
75 #define _TIMER_T
76 typedef int    timer_t;
77 #endif

79 #if defined(__STDC__)

81 #if defined(__EXTENSIONS__) || \
82     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
83     (_POSIX_C_SOURCE - 0 >= 199506L) || defined(_REENTRANT)
84 extern struct tm *gmtime_r(const time_t *_RESTRICT_KYWD,
85                          struct tm *_RESTRICT_KYWD);
86 extern struct tm *localtime_r(const time_t *_RESTRICT_KYWD,
87                              struct tm *_RESTRICT_KYWD);
88 #endif

90 #if (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
91     defined(_XPG4) || defined(__EXTENSIONS__)

93 #ifdef _STRPTIME_DONTZERO
94 #ifdef _PRAGMA_REDEFINE_EXTNAME
95 #pragma redefine_extname strftime __strftime_dontzero
96 #else /* _PRAGMA_REDEFINE_EXTNAME */
97 extern char *__strftime_dontzero(const char *, const char *, struct tm *);
98 #define strftime __strftime_dontzero
99 #endif /* _PRAGMA_REDEFINE_EXTNAME */
100 #endif /* _STRPTIME_DONTZERO */

102 extern char *strftime(const char *_RESTRICT_KYWD, const char *_RESTRICT_KYWD,
103                    struct tm *_RESTRICT_KYWD);

105 #endif /* (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))... */

107 #if (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
108     (_POSIX_C_SOURCE > 2) || defined(__EXTENSIONS__)
109 /*
110 * Neither X/Open nor POSIX allow the inclusion of <signal.h> for the
111 * definition of the sigevent structure.  Both require the inclusion
112 * of <signal.h> and <time.h> when using the timer_create() function.
113 * However, X/Open also specifies that the sigevent structure be defined
114 * in <time.h> as described in the header <signal.h>.  This prevents
115 * compiler warnings for applications that only include <time.h> and not
116 * also <signal.h>.  The signal union and the sigevent structure is
117 * therefore defined both here and in <sys/signinfo.h> which gets included
118 * via inclusion of <signal.h>.
119 */
120 #ifndef _SIGVAL
121 #define _SIGVAL
122 union sigval {
123     int    sival_int;    /* integer value */
124     void   *sival_ptr;  /* pointer value */
```

```
125 };
    unchanged_portion_omitted
361 #endif /* !__lint */
362 #endif /* __PRAGMA_REDEFINE_EXTNAME */
364 #else /* (_POSIX_C_SOURCE - 0 >= 199506L) || ... */
366 extern char *asctime_r();
367 extern char *ctime_r();
369 #endif /* (_POSIX_C_SOURCE - 0 >= 199506L) || ... */
371 #endif /* __STDC__ */
373 #endif /* defined(__EXTENSIONS__) || defined(_REENTRANT)... */
376 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
378 #ifndef _LOCALE_T
379 #define _LOCALE_T
380 typedef struct locale *locale_t;
381 #endif
383 #if defined(__STDC__)
384 extern size_t strftime_l(char *_RESTRICT_KYWD, size_t,
385     const char *_RESTRICT_KYWD, const struct tm *_RESTRICT_KYWD, locale_t);
386 #else /* __STDC__ */
387 extern size_t strftime_l();
388 #endif /* __STDC__ */
390 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */
392 #ifdef __cplusplus
393 }
    unchanged_portion_omitted
```



new/usr/src/head/wchar.h

1

```
*****
6376 Fri Jul 11 11:42:54 2014
new/usr/src/head/wchar.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
24  * Copyright (c) 1993, 2010, Oracle and/or its affiliates. All rights reserved.
25  */

27 #ifndef _WCHAR_H
28 #define _WCHAR_H

30 #include <sys/feature_tests.h>
31 #include <iso/wchar_iso.h>
32 #include <iso/wchar_c99.h>

34 /*
35  * Allow global visibility for symbols defined in
36  * C++ "std" namespace in <iso/wchar_iso.h>.
37  */
38 #if __cplusplus >= 199711L
39 using std::FILE;
40 using std::wint_t;
41 using std::clock_t;
42 using std::size_t;
43 using std::time_t;
44 using std::tm;
45 using std::mbstate_t;
46 using std::fgetwc;
47 using std::fgetw;
48 using std::fputwc;
49 using std::fputw;
50 using std::ungetwc;
51 using std::getwc;
52 using std::getwchar;
53 using std::putwc;
54 using std::putwchar;
55 using std::wcstod;
56 using std::wcstol;
57 using std::wcstoul;
58 using std::wscat;
```

new/usr/src/head/wchar.h

2

```
59 using std::wcschr;
60 using std::wcsncmp;
61 using std::wscoll;
62 using std::wscpy;
63 using std::wscspn;
64 using std::wcslen;
65 using std::wscncat;
66 using std::wscncmp;
67 using std::wscncpy;
68 using std::wscsbrk;
69 using std::wcsrchr;
70 using std::wcsspn;
71 using std::wcsxfrm;
72 using std::wcstok;
73 using std::wcsftime;
74 /* not XPG4 and not XPG4v2 */
75 #if (!defined(_XPG4) && !defined(_XPG4_2) || defined(_XPG5))
76 using std::btowc;
77 using std::fwprintf;
78 using std::fwscanf;
79 using std::fwide;
80 using std::mbsinit;
81 using std::mbrlen;
82 using std::mbrtowc;
83 using std::mbrtowcs;
84 using std::swprintf;
85 using std::swscanf;
86 using std::vfwprintf;
87 using std::vwprintf;
88 using std::vswprintf;
89 using std::wcr tomb;
90 using std::wcr tomb;
91 using std::wcr tomb;
92 using std::wctob;
93 using std::wmemchr;
94 using std::wmemcmp;
95 using std::wmemcpy;
96 using std::wmemmove;
97 using std::wmemset;
98 using std::wprintf;
99 using std::wscanf;
100 #endif /* not XPG4 and not XPG4v2 */
101 #endif /* __cplusplus >= 199711L */

103 #ifdef __cplusplus
104 extern "C" {
105 #endif

107 #if !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE) || defined(__EXTENSIONS__)
108 #if !defined(_WCTYPE_T) || __cplusplus >= 199711L
109 #define _WCTYPE_T
110 typedef int wctype_t;
111 #endif
112 #endif /* !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE)... */

114 /*
115  * XPG6 requires that va_list be defined as defined in <stdarg.h>,
116  * however, inclusion of <stdarg.h> breaks Standard C namespace.
117  */
118 #if defined(_XPG6) && !defined(_VA_LIST)
119 #define _VA_LIST
120 typedef __va_list va_list;
121 #endif /* defined(_XPG6) && !defined(_VA_LIST) */

123 #ifdef _STDC_
```

```

125 #if !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE) || defined(__EXTENSIONS__)
126 #if __cplusplus >= 199711L
127 namespace std {
128 #endif
129 extern int iswalpha(wint_t);
130 extern int iswupper(wint_t);
131 extern int iswlower(wint_t);
132 extern int iswdigit(wint_t);
133 extern int iswxdigit(wint_t);
134 extern int iswalnum(wint_t);
135 extern int iswspace(wint_t);
136 extern int iswpunct(wint_t);
137 extern int iswprint(wint_t);
138 extern int iswgraph(wint_t);
139 extern int iswcntrl(wint_t);
140 extern int iswctype(wint_t, wctype_t);
141 extern wint_t tolower(wint_t);
142 extern wint_t toupper(wint_t);
143 extern wchar_t *wcswcs(const wchar_t *, const wchar_t *);
144 extern int wcswidth(const wchar_t *, size_t);
145 extern int wcwidth(wchar_t);
146 extern wctype_t wctype(const char *);
147 #if __cplusplus >= 199711L
148 } /* namespace std */

150 using std::iswalpha;
151 using std::iswupper;
152 using std::iswlower;
153 using std::iswdigit;
154 using std::iswxdigit;
155 using std::iswalnum;
156 using std::iswspace;
157 using std::iswpunct;
158 using std::iswprint;
159 using std::iswgraph;
160 using std::iswcntrl;
161 using std::iswctype;
162 using std::tolower;
163 using std::toupper;
164 using std::wcswcs;
165 using std::wcswidth;
166 using std::wcwidth;
167 using std::wctype;
168 #endif
169 #endif /* !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE)... */

171 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)

173 #ifndef _LOCALE_T
174 #define _LOCALE_T
175 typedef struct locale *locale_t;
176 #endif

170 #if defined(__EXTENSIONS__) || \
171     (!defined(_STRICT_STDC) && !defined(_XOPEN_OR_POSIX))
172     /* || defined(_XPG7) */
173 extern wchar_t *wcsdup(const wchar_t *);
174 extern size_t wcsnlen(const wchar_t *, size_t);
175 extern wchar_t *wcpncpy(wchar_t *_RESTRIC_KYWD, const wchar_t *_RESTRIC_KYWD);
176 extern wchar_t *wcpncpy(wchar_t *_RESTRIC_KYWD, const wchar_t *_RESTRIC_KYWD,
177     size_t);
178 extern size_t wcsxfrm_l(wchar_t *_RESTRIC_KYWD, const wchar_t *_RESTRIC_KYWD,
179     size_t, locale_t);
180 extern int wcsxfrm_l(const wchar_t *, const wchar_t *, locale_t);
181 extern int wscoll_l(const wchar_t *, const wchar_t *, locale_t);
182 extern wchar_t *wcsdup(const wchar_t *);
183 extern int wscasecmp(const wchar_t *, const wchar_t *);

```

```

187 extern int wscasecmp_l(const wchar_t *, const wchar_t *, locale_t);
188 extern int wcsncasecmp(const wchar_t *, const wchar_t *, size_t);
189 extern int wcsncasecmp_l(const wchar_t *, const wchar_t *, size_t, locale_t);
190 extern size_t mbsnrtowcs(wchar_t *_RESTRIC_KYWD, const char **_RESTRIC_KYWD,
191     size_t, size_t, mbstate_t *_RESTRIC_KYWD);
192 #endif

193 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */

195 #else /* __STDC__ */

197 #if !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE) || defined(__EXTENSIONS__)
198 extern int iswalpha();
199 extern int iswupper();
200 extern int iswlower();
201 extern int iswdigit();
202 extern int iswxdigit();
203 extern int iswalnum();
204 extern int iswspace();
205 extern int iswpunct();
206 extern int iswprint();
207 extern int iswgraph();
208 extern int iswcntrl();
209 extern int iswctype();
210 extern wint_t tolower();
211 extern wint_t toupper();
212 extern wchar_t *wcswcs();
213 extern int wcswidth();
214 extern int wcwidth();
215 extern wctype_t wctype();
216 #endif /* !defined(_STRICT_STDC) || defined(_XOPEN_SOURCE)... */

218 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
219 #if defined(__EXTENSIONS__) || \
220     (!defined(_STRICT_STDC) && !defined(_XOPEN_OR_POSIX))
221     /* || defined(_XPG7) */
222 extern wchar_t *wcsdup();
223 extern size_t wcsnlen();
224 extern wchar_t *wcpncpy();
225 extern wchar_t *wcpncpy();
226 extern size_t wcsxfrm_l();
227 extern int wscoll_l();
228 extern wchar_t *wcsdup();
229 extern int wscasecmp();
230 extern int wcsncasecmp();
231 extern int wcsncasecmp_l();
232 extern size_t mbsnrtowcs();
233 #endif

234 #endif /* __STDC__ */

235 #ifdef __cplusplus
236     unchanged_portion_omitted

```

new/usr/src/head/wctype.h

1

```
*****
5283 Fri Jul 11 11:42:54 2014
new/usr/src/head/wctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, Version 1.0 only
6  * (the "License"). You may not use this file except in compliance
7  * with the License.
8  *
9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
10 * or http://www.opensolaris.org/os/licensing.
11 * See the License for the specific language governing permissions
12 * and limitations under the License.
13 *
14 * When distributing Covered Code, include this CDDL HEADER in each
15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
16 * If applicable, add the following below this CDDL HEADER, with the
17 * fields enclosed by brackets "[]" replaced with your own identifying
18 * information: Portions Copyright [yyyy] [name of copyright owner]
19 *
20 * CDDL HEADER END
21 */
22 /*      wctype.h      1.13 89/11/02 SMI; JLE  */
23 /*      from AT&T JAE 2.1      */
24 /*      definitions for international functions */

26 /*
27 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
28 * Copyright 2003 Sun Microsystems, Inc. All rights reserved.
29 * Use is subject to license terms.
30 */

32 #ifndef _WCTYPE_H
33 #define _WCTYPE_H

35 #include <sys/feature_tests.h>
34 #pragma ident  "%Z%M% %I%      %E% SMI"

36 #include <iso/wctype_iso.h>
37 #ifndef _STRICT_SYMBOLS
38 #if !defined(__XOPEN_OR_POSIX) && !defined(_STRICT_STDC) || \
39     defined(__EXTENSIONS__)
38 #include <ctype.h>
39 #include <wchar.h>
40 #endif

42 /*
43 * Allow global visibility for symbols defined in
44 * C++ "std" namespace in <iso/wctype_iso.h>.
45 */
46 #if __cplusplus >= 199711L
47 using std::wint_t;
48 using std::wctrans_t;
49 using std::iswalnum;
50 using std::iswalpha;
51 using std::iswcntrl;
52 using std::iswdigit;
53 using std::iswgraph;
54 using std::iswlower;
```

new/usr/src/head/wctype.h

2

```
55 using std::iswprint;
56 using std::iswpunct;
57 using std::iswspace;
58 using std::iswupper;
59 using std::iswxdigit;
60 using std::tolower;
61 using std::toupper;
62 using std::wctrans;
63 using std::wctrans_t;
64 using std::iswctype;
65 using std::wctype;
66 #if __cplusplus >= 201103L
67 using std::iswblank;
68 #endif
69 #endif

69 /*
70 * This header needs to be included here because it relies on the global
71 * visibility of wint_t in the C++ environment.
72 */

74 #include <iso/wctype_c99.h>

71 #ifdef __cplusplus
72 extern "C" {
73 #endif

75 /* do not allow any of the following in a strictly conforming application */
76 #ifndef _STRICT_SYMBOLS
81 #if !defined(__XOPEN_OR_POSIX) && !defined(_STRICT_STDC) || \
82     defined(__EXTENSIONS__)

78 /*
79 * data structure for supplementary code set
80 * for character class and conversion
81 */
82 struct _wctype {
83     wchar_t tmin;           /* minimum code for wctype */
84     wchar_t tmax;           /* maximum code for wctype */
85     unsigned char *index;   /* class index */
86     unsigned int *type;     /* class type */
87     wchar_t cmin;           /* minimum code for conversion */
88     wchar_t cmax;           /* maximum code for conversion */
89     wchar_t *code;         /* conversion code */
90 };

93 #ifdef _ILLUMOS_PRIVATE
94 extern int __iswrune(wint_t);
95 extern wint_t __nextwctype(wint_t, wctype_t);
96 #define iswrune(c)          __iswrune(c)
97 #define nextwctype(c, t)    __nextwctype(c, t)
98 #endif

100 /* character classification functions */

102 /* iswascii is still a macro */
103 #define iswascii(c)         isascii(c)

105 /* isw*, except iswascii(), are not macros any more. They become functions */
106 #ifdef _STDC__

114 extern unsigned iswctype(wchar_t, int);
115 extern wchar_t trwctype(wchar_t, int);
108 /* is* also become functions */
109 extern int isphonogram(wint_t);
```

```

110 extern int isideogram(wint_t);
111 extern int isenglish(wint_t);
112 extern int isnumber(wint_t);
113 extern int isspecial(wint_t);
114 /* From BSD/MacOS */
115 extern int iswideogram(wint_t);
116 extern int iswphonogram(wint_t);
117 extern int iswnumber(wint_t);
118 extern int iswhexnumber(wint_t);
119 extern int iswspecial(wint_t);
122 #else
121 #else /* __STDC__ */
124 extern unsigned _iswctype();
125 extern wchar_t _trwctype();
123 /* is* also become functions */
124 extern int isphonogram();
125 extern int isideogram();
126 extern int isenglish();
127 extern int isnumber();
128 extern int isspecial();
129 /* From BSD/MacOS */
130 extern int iswideogram();
131 extern int iswphonogram();
132 extern int iswnumber();
133 extern int iswspecial();
134 #endif
136 #define iscodeset0(c) isascii(c)
137 #define iscodeset1(c) (((c) & WCHAR_CSMASK) == WCHAR_CS1)
138 #define iscodeset2(c) (((c) & WCHAR_CSMASK) == WCHAR_CS2)
139 #define iscodeset3(c) (((c) & WCHAR_CSMASK) == WCHAR_CS3)
141 #endif /* !defined(_STRICT_SYMBOLS)... */
139 #endif /* !defined(__XOPEN_OR_POSIX)... */
144 /* XPG7 extended locale support */
145 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
147 #ifndef _LOCALE_T
148 #define _LOCALE_T
149 typedef struct locale *locale_t;
150 #endif
152 #if defined(__STDC__)
153 extern wint_t tolower_l(wint_t, locale_t);
154 extern wint_t toupper_l(wint_t, locale_t);
155 extern wint_t towctrans_l(wint_t, wctrans_t, locale_t);
156 extern int iswctype_l(wint_t, wctype_t, locale_t);
157 extern int iswalnum_l(wint_t, locale_t);
158 extern int iswalphal_l(wint_t, locale_t);
159 extern int iswcntrl_l(wint_t, locale_t);
160 extern int iswdigit_l(wint_t, locale_t);
161 extern int iswgraph_l(wint_t, locale_t);
162 extern int iswlower_l(wint_t, locale_t);
163 extern int iswprint_l(wint_t, locale_t);
164 extern int iswpunct_l(wint_t, locale_t);
165 extern int iswspace_l(wint_t, locale_t);
166 extern int iswupper_l(wint_t, locale_t);
167 extern int iswxdigit_l(wint_t, locale_t);
168 extern wctrans_t wctrans_l(const char *, locale_t);
169 extern wctype_t wctype_l(const char *, locale_t);
170 #else /* __STDC__ */
171 extern wint_t tolower_l();

```

```

172 extern wint_t toupper_l();
173 extern wint_t towctrans_l();
174 extern int iswctype_l();
175 extern int iswalnum_l();
176 extern int iswalphal_l();
177 extern int iswcntrl_l();
178 extern int iswdigit_l();
179 extern int iswgraph_l();
180 extern int iswlower_l();
181 extern int iswprint_l();
182 extern int iswpunct_l();
183 extern int iswspace_l();
184 extern int iswupper_l();
185 extern int iswxdigit_l();
186 extern wctrans_t wctrans_l();
187 extern wctype_t wctype_l();
188 #endif /* __STDC__ */
189 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */
191 #ifdef __cplusplus
192 }
_____unchanged_portion_omitted_

```

```

*****
3743 Fri Jul 11 11:42:54 2014
new/usr/src/head/xlocale.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 2014 Garrett D'Amore <garrett@damore.org>
14  */

16 #ifndef _XLOCALE_H
17 #define _XLOCALE_H

19 /*
20  * This file supplies declarations for extended locale routines, as
21  * originally delivered by MacOS X. Many of these things are now
22  * officially part of XPG7. (Note that while the interfaces are the
23  * same as MacOS X, there is no shared implementation.)
24  *
25  * Those declarations that are part of XPG7 are provided for the in the
26  * XPG7-specified location. This file lists just the declarations that
27  * were not part of the standard. These will be useful in their own right,
28  * and will aid porting programs that don't strictly follow the standard.
29  *
30  * Note that it is an error to include this file in a program with strict
31  * symbol visibility rules (under strict ANSI or POSIX_C_SOURCE rules.)
32  * If this is done, the symbols defined here will indeed be exposed to your
33  * program, but those symbols that are part of the related standards might
34  * not be.
35  */

37 #include <sys/feature_tests.h>
38 #include <wchar.h>
39 #include <locale.h>
40 #include <stdio.h>

42 #ifdef __cplusplus
43 extern "C" {
44 #endif

46 #ifndef _LOCALE_T
47 #define _LOCALE_T
48 typedef struct locale *locale_t;
49 #endif

51 extern int mbsinit_l(const mbstate_t *, locale_t);

53 extern size_t mbsrtowcs_l(wchar_t * _RESTRICT_KYWD, const char **_RESTRICT_KYWD,
54   size_t, mbstate_t * _RESTRICT_KYWD, locale_t);

56 extern size_t mbsnrtowcs_l(wchar_t * _RESTRICT_KYWD, const char **_RESTRICT_KYWD,
57   size_t, size_t, mbstate_t * _RESTRICT_KYWD, locale_t);

```

```

59 extern char *strptime_l(const char * _RESTRICT_KYWD, const char * _RESTRICT_KYWD,
60   struct tm * _RESTRICT_KYWD, locale_t);

62 extern int wwidth_l(wchar_t, locale_t);

64 extern int wcswidth_l(const wchar_t *, size_t, locale_t);

66 extern int iswspecial_l(wint_t, locale_t);
67 extern int iswnumber_l(wint_t, locale_t);
68 extern int iswhexnumber_l(wint_t, locale_t);
69 extern int iswideogram_l(wint_t, locale_t);
70 extern int iswphonogram_l(wint_t, locale_t);

72 extern wint_t btowc_l(int, locale_t);
73 extern int wctob_l(wint_t, locale_t);
74 extern size_t mbrtowc_l(wchar_t * _RESTRICT_KYWD, const char * _RESTRICT_KYWD,
75   size_t, mbstate_t * _RESTRICT_KYWD, locale_t);
76 extern size_t mbstowcs_l(wchar_t * _RESTRICT_KYWD, const char * _RESTRICT_KYWD,
77   size_t, locale_t);
78 extern int mblen_l(const char *, size_t, locale_t);
79 extern size_t mbrlen_l(const char * _RESTRICT_KYWD, size_t,
80   mbstate_t * _RESTRICT_KYWD, locale_t);
81 extern int mbtowc_l(wchar_t * _RESTRICT_KYWD, const char * _RESTRICT_KYWD, size_t,
82   locale_t);
83 extern size_t wcsrtombs_l(char * _RESTRICT_KYWD, const wchar_t **_RESTRICT_KYWD,
84   size_t len, mbstate_t * _RESTRICT_KYWD, locale_t);
85 extern size_t wcrctomb_l(char * _RESTRICT_KYWD, wchar_t,
86   mbstate_t * _RESTRICT_KYWD, locale_t);
87 extern size_t wcstombs_l(char * _RESTRICT_KYWD, const wchar_t * _RESTRICT_KYWD,
88   size_t, locale_t);
89 extern int wctomb_l(char *, wchar_t, locale_t);

91 extern unsigned char __mb_cur_max_l(locale_t);
92 #ifndef MB_CUR_MAX_L
93 #define MB_CUR_MAX_L(1) (__mb_cur_max_l(1))
94 #endif

97 #if defined(_XPG4) && !defined(_FILEDEFED) || __cplusplus >= 199711L
98 #define _FILEDEFED
99 typedef __FILE FILE;
100 #endif

102 extern wint_t fgetwc_l(FILE *, locale_t);
103 extern wint_t getwc_l(FILE *, locale_t);

105 #ifndef getwchar_l
106 #define getwchar_l(1) fgetwc_l(stdin, (1))
107 #endif

109 #ifdef __cplusplus
110 }
111 #endif

113 #endif /* _XLOCALE_H */

```

```

*****
22544 Fri Jul 11 11:42:54 2014
new/usr/src/lib/libc/amd64/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 2004, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2012, Joyent, Inc. All rights reserved.
24 #
25 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
26 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
27 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
28 # Use is subject to license terms.
29 #

31 LIBCBASE= .
32 LIBCDIR= $(SRC)/lib/libc
33 LIBRARY= libc.a
34 LIB_PIC= libc_pic.a
35 VERS= .1
36 CPP= /usr/lib/cpp
37 TARGET_ARCH= amd64

39 # objects are grouped by source directory
41 # local objects
42 STRETS=

44 CRTOBS= \
45     cerror.o

47 DYNOBJS=

49 FPOBJS= \
50     _base_il.o \
51     fpgetmask.o \
52     fpgetround.o \
53     fpsetmask.o \
54     fpsetround.o \
55     fpstart.o

57 I386FPOBJS= \
58     _D_cplx_div.o \

```

```

59     _D_cplx_div_ix.o \
60     _D_cplx_div_rx.o \
61     _D_cplx_lr_div.o \
62     _D_cplx_lr_div_ix.o \
63     _D_cplx_lr_div_rx.o \
64     _D_cplx_mul.o \
65     _F_cplx_div.o \
66     _F_cplx_div_ix.o \
67     _F_cplx_div_rx.o \
68     _F_cplx_lr_div.o \
69     _F_cplx_lr_div_ix.o \
70     _F_cplx_lr_div_rx.o \
71     _F_cplx_mul.o \
72     _X_cplx_div.o \
73     _X_cplx_div_ix.o \
74     _X_cplx_div_rx.o \
75     _X_cplx_lr_div.o \
76     _X_cplx_lr_div_ix.o \
77     _X_cplx_lr_div_rx.o \
78     _X_cplx_mul.o

80 FPASMOBJS= \
81     __xgetRD.o \
82     _xtoll.o \
83     _xtoull.o \
84     fpcw.o \
85     fpgetsticky.o \
86     fpsetsticky.o

88 ATOMICOBJS= \
89     atomic.o

91 XATROBJS= \
92     xattr_common.o

93 COMOBJS= \
94     bcmp.o \
95     bcopy.o \
96     bsearch.o \
97     bzero.o \
98     qsort.o \
99     strtol.o \
100    strtoul.o \
101    strtoll.o \
102    strtoull.o

104 GENOBJS= \
105    _getsp.o \
106    abs.o \
107    alloca.o \
108    attrat.o \
109    byteorder.o \
110    cuexit.o \
111    ecvt.o \
112    errlst.o \
113    amd64_data.o \
114    ldivide.o \
115    lock.o \
116    makecxtxt.o \
117    memccpy.o \
118    memchr.o \
119    memcmp.o \
120    memcpy.o \
121    memset.o \
122    new_list.o \
123    proc64_id.o \
124    proc64_support.o \

```

new/usr/src/lib/libc/amd64/Makefile

```
125     setjmp.o           \  
126     siginfo.o         \  
127     siglongjmp.o     \  
128     strcmp.o         \  
129     strcpy.o         \  
130     strlen.o         \  
131     strncmp.o        \  
132     strncpy.o        \  
133     strnlen.o        \  
134     sync_instruction_memory.o
```

```
136 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds  
137 # This macro should ALWAYS be empty; native APIs are already 'large file'.  
138 COMSYSOBSJS64=
```

```
140 SYSOBSJS64=
```

```
142 COMSYSOBSJS= \  
143     __clock_timer.o   \  
144     __getloadavg.o   \  
145     __rusagesys.o    \  
146     __signotify.o    \  
147     __sigrt.o        \  
148     __time.o         \  
149     __lgrp_home_fast.o \  
150     __lgrpsys.o      \  
151     __nfssys.o       \  
152     __portfs.o       \  
153     __pset.o         \  
154     __rpcsys.o       \  
155     __sigaction.o    \  
156     __so_accept.o    \  
157     __so_bind.o      \  
158     __so_connect.o   \  
159     __so_getpeername.o \  
160     __so_getsockname.o \  
161     __so_getsockopt.o \  
162     __so_listen.o    \  
163     __so_recv.o      \  
164     __so_recvfrom.o  \  
165     __so_recvmsg.o   \  
166     __so_send.o      \  
167     __so_sendmsg.o   \  
168     __so_sendto.o    \  
169     __so_setsockopt.o \  
170     __so_shutdown.o  \  
171     __so_socket.o    \  
172     __so_socketpair.o \  
173     __sockconfig.o   \  
174     acct.o           \  
175     acl.o            \  
176     adjtime.o        \  
177     alarm.o          \  
178     brk.o            \  
179     chdir.o          \  
180     chroot.o         \  
181     cladm.o          \  
182     close.o          \  
183     execve.o         \  
184     exit.o           \  
185     facl.o           \  
186     fchdir.o         \  
187     fchroot.o        \  
188     fdsync.o         \  
189     fpathconf.o     \  
190     fstatfs.o
```

3

new/usr/src/lib/libc/amd64/Makefile

```
191     fstatvfs.o       \  
192     getcpu.o         \  
193     getdents.o      \  
194     getegid.o        \  
195     geteuid.o        \  
196     getgid.o         \  
197     getgroups.o      \  
198     gethrtime.o     \  
199     getitimer.o     \  
200     getmsg.o         \  
201     getpid.o         \  
202     getpmsg.o        \  
203     getppid.o        \  
204     getrlimit.o     \  
205     getuid.o         \  
206     gttty.o          \  
207     install_utrap.o \  
208     ioctl.o          \  
209     kaio.o           \  
210     kill.o           \  
211     llseek.o         \  
212     lseek.o          \  
213     mmapobjsys.o    \  
214     memcntl.o        \  
215     mincore.o        \  
216     mmap.o           \  
217     modctl.o         \  
218     mount.o          \  
219     mprotect.o       \  
220     munmap.o         \  
221     nice.o           \  
222     ntp_adjtime.o   \  
223     ntp_gettime.o   \  
224     p_online.o       \  
225     pathconf.o      \  
226     pause.o          \  
227     pcsample.o       \  
228     pipe2.o          \  
229     pollsys.o        \  
230     pread.o          \  
231     priocntlset.o   \  
232     processor_bind.o \  
233     processor_info.o \  
234     profil.o         \  
235     putmsg.o         \  
236     putpmsg.o        \  
237     pwrite.o         \  
238     read.o           \  
239     readv.o          \  
240     resolvepath.o   \  
241     seteguid.o       \  
242     setgid.o         \  
243     setgroups.o      \  
244     setitimer.o     \  
245     setreid.o        \  
246     setrlimit.o     \  
247     setuid.o         \  
248     sigaltstk.o     \  
249     sigprocmsk.o    \  
250     sigsendset.o    \  
251     sigsuspend.o     \  
252     statfs.o         \  
253     statvfs.o        \  
254     stty.o           \  
255     sync.o           \  
256     sysconfig.o
```

4

## new/usr/src/lib/libc/amd64/Makefile

```

257 sysfs.o \
258 sysinfo.o \
259 syslwp.o \
260 times.o \
261 ulimit.o \
262 umask.o \
263 umount2.o \
264 utssys.o \
265 uucopy.o \
266 vhangup.o \
267 waitid.o \
268 write.o \
269 writev.o \
270 yield.o \

272 SYSOBJS= \
273 __clock_gettime.o \
274 __getcontext.o \
275 __uadmin.o \
276 __lwp_mutex_unlock.o \
277 __stack_grow.o \
278 door.o \
279 forkx.o \
280 forkallx.o \
281 getcontext.o \
282 gettimeofday.o \
283 lwp_private.o \
284 nuname.o \
285 syscall.o \
286 sysi86.o \
287 tls_get_addr.o \
288 uadmin.o \
289 umount.o \
290 uname.o \
291 vforkx.o \

293 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
294 # This macro should ALWAYS be empty; native APIs are already 'large file'.
295 PORTGEN64=

297 # objects from source under $(LIBCDIR)/port
298 PORTFP= \
299 __flt_decim.o \
300 __flt_rounds.o \
301 __tbl_10_b.o \
302 __tbl_10_h.o \
303 __tbl_10_s.o \
304 __tbl_2_b.o \
305 __tbl_2_h.o \
306 __tbl_2_s.o \
307 __tbl_fdq.o \
308 __tbl_tens.o \
309 __x_power.o \
310 __base_sup.o \
311 aconvert.o \
312 decimal_bin.o \
313 double_decim.o \
314 econvert.o \
315 fconvert.o \
316 file_decim.o \
317 finite.o \
318 fp_data.o \
319 func_decim.o \
320 gconvert.o \
321 hex_bin.o \
322 ieee_globals.o \

```

5

## new/usr/src/lib/libc/amd64/Makefile

```

323 pack_float.o \
324 sigfpe.o \
325 string_decim.o \

327 PORTGEN= \
328 __env_data.o \
329 __xftw.o \
330 a64l.o \
331 abort.o \
332 addsev.o \
333 ascii_strcasecmp.o \
334 ascii_strncasecmp.o \
335 assert.o \
336 atof.o \
337 atoi.o \
338 atol.o \
339 atoll.o \
340 attropen.o \
341 atexit.o \
342 atfork.o \
343 basename.o \
344 calloc.o \
345 catgets.o \
346 catopen.o \
347 cfgetispeed.o \
348 cfgetospeed.o \
349 cfree.o \
350 cfsetispeed.o \
351 cfsetospeed.o \
352 cftime.o \
353 clock.o \
354 closedir.o \
355 closefrom.o \
356 confstr.o \
357 crypt.o \
358 csetlen.o \
359 ctime.o \
360 ctime_r.o \
361 daemon.o \
362 deflt.o \
363 directio.o \
364 dirname.o \
365 div.o \
366 drand48.o \
367 dup.o \
368 env_data.o \
369 err.o \
370 errno.o \
371 euclen.o \
372 event_port.o \
373 execvp.o \
374 fattach.o \
375 fdetach.o \
376 fdopendir.o \
377 ffs.o \
378 fls.o \
379 fmtmsg.o \
380 ftime.o \
381 ftok.o \
382 ftw.o \
383 gcvt.o \
384 getauxv.o \
385 getcwd.o \
386 getdate_err.o \
387 getdtblsize.o \
388 getenv.o \

```

6



```

389     getexecname.o      \|
390     getgrnam.o         \|
391     getgrnam_r.o       \|
392     gethostid.o        \|
393     gethostname.o      \|
394     gethz.o            \|
395     getisax.o          \|
396     getloadavg.o       \|
397     getlogin.o         \|
398     getmntent.o        \|
399     getnetgrent.o      \|
400     get_nprocs.o       \|
401     getopt.o           \|
402     getopt_long.o      \|
403     getpagesize.o      \|
404     getpw.o            \|
405     getpwnam.o         \|
406     getpwnam_r.o       \|
407     getrusage.o        \|
408     getspent.o         \|
409     getspent_r.o       \|
410     getsubopt.o        \|
411     gettxt.o           \|
412     getusershell.o    \|
413     getut.o            \|
414     getutx.o           \|
415     getvfsent.o        \|
416     getwd.o            \|
417     getwidth.o         \|
418     getxby_door.o     \|
419     gtxt.o             \|
420     hsearch.o          \|
421     iconv.o            \|
422     imaxabs.o          \|
423     imaxdiv.o          \|
424     index.o            \|
425     initgroups.o       \|
426     insque.o           \|
427     isaexec.o          \|
428     isastream.o        \|
429     isatty.o           \|
430     killpg.o           \|
431     klpdlib.o          \|
432     l64a.o             \|
433     lckpwwd.o          \|
434     lconstants.o      \|
435     lexp10.o           \|
436     lfind.o            \|
437     lfmt.o             \|
438     lfmt_log.o         \|
439     lldiv.o            \|
440     llog10.o           \|
441     lltostr.o          \|
442     lmath.o            \|
443     localtime.o        \|
444     lsearch.o          \|
445     madvise.o          \|
446     malloc.o           \|
447     memalign.o         \|
448     memmem.o           \|
449     mkdev.o            \|
450     mkdtemp.o          \|
451     mkfifo.o           \|
452     mkstemp.o          \|
453     mktemp.o           \|
454     mlock.o            \|

```

```

455     mlockall.o        \|
456     mon.o             \|
457     msync.o           \|
458     munlock.o         \|
459     munlockall.o     \|
460     ndbm.o            \|
461     nftw.o            \|
462     nlspath_checks.o  \|
463     nsparse.o         \|
464     nss_common.o      \|
465     nss_dbdefs.o     \|
466     nss_deffinder.o  \|
467     opendir.o         \|
468     opt_data.o        \|
469     perror.o          \|
470     pfmt.o            \|
471     pfmt_data.o       \|
472     pfmt_print.o      \|
473     pipe.o            \|
474     plock.o           \|
475     poll.o            \|
476     posix_fadvise.o   \|
477     posix_fallocate.o \|
478     posix_madvise.o   \|
479     posix_memalign.o  \|
480     priocntl.o        \|
481     privlib.o         \|
482     priv_str_xlate.o  \|
483     psiginfo.o        \|
484     psignal.o         \|
485     pt.o              \|
486     putpwent.o        \|
487     putspent.o        \|
488     raise.o           \|
489     rand.o            \|
490     random.o          \|
491     rctlops.o         \|
492     readdir.o         \|
493     readdir_r.o       \|
494     realpath.o        \|
495     reboot.o          \|
496     regexpr.o         \|
497     remove.o          \|
498     rewinddir.o      \|
499     rindex.o          \|
500     scandir.o         \|
501     seekdir.o         \|
502     select.o          \|
503     setlabel.o        \|
504     setpriority.o     \|
505     settimeofday.o   \|
506     sh_locks.o        \|
507     sigflag.o         \|
508     siglist.o         \|
509     sigsend.o         \|
510     sigsetops.o       \|
511     ssignal.o         \|
512     stack.o           \|
513     stpcpy.o          \|
514     stpncpy.o         \|
515     str2sig.o         \|
516     strcase_charmap.o \|
517     strcat.o          \|
518     strchr.o          \|
519     strchrnul.o       \|
520     strcspn.o         \|

```

```

521      strdup.o          \
522      strerror.o       \
523      strlcat.o        \
524      strlcpy.o        \
525      strncat.o        \
526      strndup.o        \
527      strpbrk.o        \
528      strrchr.o        \
529      strsep.o         \
530      strsignal.o      \
531      strspn.o         \
532      strstr.o         \
533      strtod.o         \
534      strtointmax.o    \
535      strtok.o         \
536      strtok_r.o       \
537      strtoumax.o     \
538      swab.o           \
539      swapctl.o        \
540      sysconf.o        \
541      syslog.o         \
542      tcdrain.o        \
543      tcflow.o         \
544      tcflush.o        \
545      tcgetattr.o      \
546      tcgetpgrp.o     \
547      tcgetsid.o       \
548      tcsendbreak.o   \
549      tcsetattr.o     \
550      tcsetpgrp.o     \
551      tell.o           \
552      telldir.o        \
553      tfind.o          \
554      time_data.o      \
555      time_gdata.o    \
556      tls_data.o       \
557      truncate.o       \
558      tsdalloc.o       \
559      tsearch.o        \
560      ttyname.o        \
561      ttyslot.o       \
562      ualarm.o         \
563      ucred.o          \
564      valloc.o         \
565      vlfmt.o          \
566      vpfmt.o          \
567      waitpid.o        \
568      walkstack.o     \
569      wdata.o          \
570      xgetwidth.o     \
571      xpg4.o           \
572      xpg6.o           \
574 PORTPRINT_W=       \
575      doprnt_w.o      \
577 PORTPRINT=         \
578      asprintf.o       \
579      doprnt.o         \
580      fprintf.o        \
581      printf.o         \
582      snprintf.o       \
583      sprintf.o        \
584      vfprintf.o       \
585      vprintf.o        \
586      vsnprintf.o     \

```

```

587      vsprintf.o       \
588      vwprintf.o       \
589      wprintf.o        \
591 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
592 # This macro should ALWAYS be empty; native APIs are already 'large file'.
593 PORTSTDIO64=
595 PORTSTDIO_W=        \
596      doscan_w.o      \
598 PORTSTDIO=          \
599      __extensions.o  \
600      __endopen.o     \
601      _filbuf.o        \
602      _findbuf.o      \
603      _flsbuf.o        \
604      _wrtchk.o        \
605      clearerr.o       \
606      ctermid.o        \
607      ctermid_r.o     \
608      cuserid.o        \
609      data.o           \
610      doscan.o         \
611      fdopen.o         \
612      feof.o           \
613      ferrord.o        \
614      fgetc.o          \
615      fgets.o          \
616      fileno.o         \
617      flockf.o         \
618      flush.o          \
619      fopen.o          \
620      fpos.o           \
621      fputc.o          \
622      fputs.o          \
623      fread.o          \
624      fseek.o          \
625      fseeko.o         \
626      ftell.o          \
627      ftello.o        \
628      fwrite.o         \
629      getc.o           \
630      getchar.o        \
631      getline.o        \
632      getpass.o        \
633      gets.o           \
634      getw.o           \
635      mse.o            \
636      popen.o          \
637      putc.o           \
638      putchar.o        \
639      puts.o           \
640      putw.o           \
641      rewind.o         \
642      scanf.o          \
643      setbuf.o         \
644      setbuffer.o     \
645      setvbuf.o        \
646      system.o         \
647      tempnam.o        \
648      tmpfile.o        \
649      tmpnam_r.o       \
650      ungetc.o         \
651      vscanf.o         \
652      vwscanf.o        \

```

```

653      wscanf.o
655 PORTI18N=
656      getwchar.o
657      putwchar.o
658      putws.o
658      strcasecmp.o
659      strcasestr.o
660      strncasecmp.o
659      strtows.o
660      wcsnlen.o
661      wcsstr.o
662      wcstoimax.o
663      wcstol.o
664      wcstoul.o
665      wscwcs.o
666      wmemchr.o
667      wmemcmp.o
668      wmemcpy.o
669      wmemmove.o
670      wmemset.o
673      wscasecmp.o
671      wscat.o
672      wscr.o
673      wscmp.o
674      wscpy.o
675      wscspn.o
676      wsdup.o
677      wslen.o
681      wncasecmp.o
678      wncat.o
679      wncmp.o
680      wncpy.o
681      wspbrk.o
682      wprintf.o
683      wsrchr.o
684      wsscanf.o
685      wssp.o
686      wstod.o
687      wstok.o
688      wstol.o
689      wstoll.o
690      wsxfrm.o
691      gettext.o
692      gettext_gnu.o
693      gettext_real.o
694      gettext_util.o
699      isdigit.o
695      plural_parser.o
696      wdresolve.o
697      _ctype.o
698      isascii.o
699      toascii.o

701 PORTI18N_COND=
702      wcstol_longlong.o
703      wcstoul_longlong.o

705 PORTLOCALE=
706      big5.o
707      btowc.o
708      collate.o
709      collcmp.o
710      euc.o
711      fnmatch.o
712      fgetwc.o

```

```

713      fgetws.o
714      fix_grouping.o
715      fputwc.o
716      fputws.o
717      fwide.o
718      gb18030.o
719      gb2312.o
720      gbk.o
721      getdate.o
722      isdigit.o
723      iswctype.o
724      ldap.o
725      lmessages.o
726      lnumeric.o
727      lmonetary.o
728      localeconv.o
729      localeimpl.o
730      mbftowc.o
731      mblen.o
732      mbrlen.o
733      mbrtowc.o
734      mbsinit.o
735      mbsnrtowcs.o
736      mbsrtowcs.o
737      mbstowcs.o
738      mbtowc.o
739      mskanji.o
740      nextwctype.o
741      nl_langinfo.o
742      none.o
743      regcomp.o
744      regfree.o
745      regerror.o
746      regex.o
747      rune.o
748      runetype.o
749      setlocale.o
750      setrunelocale.o
751      strcasecmp.o
752      strcasestr.o
753      strcoll.o
754      strfmon.o
755      strftime.o
756      strncasecmp.o
757      strtptime.o
758      strxfrm.o
759      table.o
760      timelocal.o
761      tolower.o
762      tolower.o
763      ungetwc.o
764      utf8.o
765      wcrtoomb.o
766      wscasecmp.o
767      wscoll.o
768      wcsftime.o
769      wcsnrtombs.o
770      wcsrtoombs.o
771      wcswidth.o
772      wcstombs.o
773      wcsxfrm.o
774      wctob.o
775      wctomb.o
776      wctrans.o
777      wctype.o
778      wcwidth.o

```

```

779      wscoll.o

781 AIOOBJS= \
782     aio.o \
783     aio_alloc.o \
784     posix_aio.o

786 RTOBJS= \
787     clock_timer.o \
788     mqueue.o \
789     posixobj.o \
790     sched.o \
791     sem.o \
792     shm.o \
793     sigev_thread.o

795 TPOOLBJS= \
796     thread_pool.o

798 THREADSOBJS= \
799     alloc.o \
800     assfail.o \
801     cancel.o \
802     door_calls.o \
803     tmem.o \
804     pthr_attr.o \
805     pthr_barrier.o \
806     pthr_cond.o \
807     pthr_mutex.o \
808     pthr_rwlock.o \
809     pthread.o \
810     rwlock.o \
811     scalls.o \
812     sema.o \
813     sigaction.o \
814     spawn.o \
815     synch.o \
816     tdb_agent.o \
817     thr.o \
818     thread_interface.o \
819     tls.o \
820     tsd.o

822 THREADSMACHOBJS= \
823     machdep.o

825 THREADSASMOBJS= \
826     asm_subr.o

828 UNICODEOBJS= \
829     u8_textprep.o \
830     uconv.o

832 UNWINDMACHOBJS= \
833     call_frame_inst.o \
834     eh_frame.o \
835     thrp_unwind.o \
836     unwind.o

838 pics/unwind.o:= COPTFLAG64 =

840 UNWINDASMOBJS= \
841     unwind_frame.o

843 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
844 # This macro should ALWAYS be empty; native APIs are already 'large file'.

```

```

845 PORTSYS64=

847 PORTSYS= \
848     _autofssys.o \
849     access.o \
850     acctctl.o \
851     bsd_signal.o \
852     chmod.o \
853     chown.o \
854     corectl.o \
855     exaccts.o \
856     execl.o \
857     execl.o \
858     execv.o \
859     fcntl.o \
860     getpagesizes.o \
861     getpeercred.o \
862     inst_sync.o \
863     issetugid.o \
864     label.o \
865     link.o \
866     lockf.o \
867     lwp.o \
868     lwp_cond.o \
869     lwp_rwlock.o \
870     lwp_sigmask.o \
871     meminfosys.o \
872     mkdir.o \
873     mknod.o \
874     msgsys.o \
875     nfssys.o \
876     open.o \
877     pgrpsys.o \
878     posix_sigwait.o \
879     ppriv.o \
880     psetsys.o \
881     rctlsys.o \
882     readlink.o \
883     rename.o \
884     sbrk.o \
885     semsys.o \
886     set_errno.o \
887     sharefs.o \
888     shmsys.o \
889     sidsys.o \
890     siginterrupt.o \
891     signal.o \
892     sigpending.o \
893     sigstack.o \
894     stat.o \
895     symlink.o \
896     tasksys.o \
897     time.o \
898     time_util.o \
899     ucontext.o \
900     unlink.o \
901     ustat.o \
902     utimesys.o \
903     zone.o

905 PORTREGEX= \
906     glob.o \
907     regcmp.o \
908     regex.o \
909     wordexp.o

```

```

911 VALUES= \
912     values-Xa.o

914 MOSTOBSJ= \
915     $(STRETS) \
916     $(CRTOBSJ) \
917     $(DYNOBJS) \
918     $(FPOBJS) \
919     $(I386FPOBJS) \
920     $(FPASMOBJS) \
921     $(ATOMICOBJS) \
922     $(XATTROBJS) \
923     $(COMOBJS) \
924     $(GENOBJS) \
925     $(PORTFP) \
926     $(PORTGEN) \
927     $(PORTGEN64) \
928     $(PORTI18N) \
929     $(PORTI18N_COND) \
930     $(PORTLOCALE) \
931     $(PORTPRINT) \
932     $(PORTPRINT_W) \
933     $(PORTREGEX) \
934     $(PORTSTDIO) \
935     $(PORTSTDIO64) \
936     $(PORTSTDIO_W) \
937     $(PORTSYS) \
938     $(PORTSYS64) \
939     $(AIOOBJS) \
940     $(RTOBJS) \
941     $(TPOOLBJS) \
942     $(THREADSOBJS) \
943     $(THREADSMACHOBJS) \
944     $(THREADSASMOBJS) \
945     $(UNICODEOBJS) \
946     $(UNWINDMACHOBJS) \
947     $(UNWINDASMOBJS) \
948     $(COMSYSOBJS) \
949     $(SYSOBJS) \
950     $(COMSYSOBJS64) \
951     $(SYSOBJS64) \
952     $(VALUES)

954 TRACEOBSJ= \
955     plockstat.o

957 # NOTE: libc.so.1 must be linked with the minimal crt1.o and crtn.o
958 # modules whose source is provided in the $(SRC)/lib/common directory.
959 # This must be done because otherwise the Sun C compiler would insert
960 # its own versions of these modules and those versions contain code
961 # to call out to C++ initialization functions. Such C++ initialization
962 # functions can call back into libc before thread initialization is
963 # complete and this leads to segmentation violations and other problems.
964 # Since libc contains no C++ code, linking with the minimal crt1.o and
965 # crtn.o modules is safe and avoids the problems described above.
966 OBJECTS= $(CRTI) $(MOSTOBSJ) $(CRTN)
967 CRTSRCS= ../../common/amd64

969 # include common library definitions
970 include ../../Makefile.lib
971 include ../../Makefile.lib.64

973 CFLAGS64 += $(CTF_FLAGS)

975 # This is necessary to avoid problems with calling _ex_unwind().
976 # We probably don't want any inlining anyway.

```

```

977 CFLAGS64 += -xinline=

979 CERRWARN += -_gcc=-Wno-parentheses
980 CERRWARN += -_gcc=-Wno-switch
981 CERRWARN += -_gcc=-Wno-uninitialized
982 CERRWARN += -_gcc=-Wno-unused-value
983 CERRWARN += -_gcc=-Wno-unused-label
984 CERRWARN += -_gcc=-Wno-unused-variable
985 CERRWARN += -_gcc=-Wno-type-limits
986 CERRWARN += -_gcc=-Wno-char-subscripts
987 CERRWARN += -_gcc=-Wno-clobbered
988 CERRWARN += -_gcc=-Wno-unused-function
989 CERRWARN += -_gcc=-Wno-address

991 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
992 # enables ASSERT() checking in the threads portion of the library.
993 # This is automatically enabled for DEBUG builds, not for non-debug builds.
994 THREAD_DEBUG =
995 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

997 # Make string literals read-only to save memory
998 CFLAGS64 += $(XSTRCONST)

1000 ALTPICS= $(TRACEOBSJ:%=pics%)

1002 $(DYNLIB) := BUILD.SO = $(LD) -o $@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) $(EXTPICS)

1004 MAPFILES = $(LIBCDIR)/port/mapfile-vers

1006 CPPFLAGS= -D_REENTRANT -D$(MACH64) -D__$(MACH64) $(THREAD_DEBUG) \
1007     -I. -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1008 ASFLAGS= $(AS_PICFLAGS) -P -D__STDC__ -D_ASM $(CPPFLAGS) \
1009     $(amd64_AS_XARCH)

1011 # As a favor to the trace syscall provider, libc still calls the
1012 # old syscall traps that have been obsoleted by the *at() interfaces.
1013 # Delete this to compile libc using only the new *at() system call traps
1014 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1016 # proc64_id.o is built with defines in $(SRC)/uts/intel/sys/x86_archext.h
1017 pics/proc64_id.o := CFLAGS64 += -I$(SRC)/uts/intel

1019 # Inform the run-time linker about libc specialized initialization
1020 RTLDINFO = -z rtldinfo=tls_rtldinfo
1021 DYNFLAGS += $(RTLDINFO)

1023 # Force libc's internal references to be resolved immediately upon loading
1024 # in order to avoid critical region problems. Since almost all libc symbols
1025 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1026 DYNFLAGS += -znow

1028 BUILD.s= $(AS) $(ASFLAGS) $< -o $@

1030 # Override this top level flag so the compiler builds in its native
1031 # C99 mode. This has been enabled to support the complex arithmetic
1032 # added to libc.
1033 C99MODE= $(C99_ENABLE)

1035 # libc method of building an archive
1036 # The "$(GREP) -v ' L '" part is necessary only until
1037 # lorder is fixed to ignore thread-local variables.
1038 BUILD.AR= $(RM) $@ ; \
1039     $(AR) q $@ '$(LORDER) $(MOSTOBSJ:%=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR

1041 # extra files for the clean target
1042 CLEANFILES= \

```

```

1043 $(LIBCDIR)/port/gen/errlst.c \
1044 $(LIBCDIR)/port/gen/new_list.c \
1045 assym.h \
1046 genassym \
1047 crt/_rtld.s \
1048 pics/crti.o \
1049 pics/crtn.o \
1050 $(ALTPICS)

1052 CLOBBERFILES += $(LIB_PIC)

1054 # list of C source for lint
1055 SRCS=
1056 $(ATOMICOBJS:%.o=$(SRC)/common/atomic/%.c) \
1057 $(XATTROBJS:%.o=$(SRC)/common/xattr/%.c) \
1058 $(COMOBJS:%.o=$(SRC)/common/util/%.c) \
1059 $(PORTFP:%.o=$(LIBCDIR)/port/fp/%.c) \
1060 $(PORTGEN:%.o=$(LIBCDIR)/port/gen/%.c) \
1061 $(PORTI18N:%.o=$(LIBCDIR)/port/il8n/%.c) \
1062 $(PORTLOCALE:%.o=$(LIBCDIR)/port/locale/%.c) \
1063 $(PORTPRINT:%.o=$(LIBCDIR)/port/print/%.c) \
1064 $(PORTREGEX:%.o=$(LIBCDIR)/port/regex/%.c) \
1065 $(PORTSTDIO:%.o=$(LIBCDIR)/port/stdio/%.c) \
1066 $(PORTSYS:%.o=$(LIBCDIR)/port/sys/%.c) \
1067 $(AIOOBJS:%.o=$(LIBCDIR)/port/aio/%.c) \
1068 $(RTOBJS:%.o=$(LIBCDIR)/port/rt/%.c) \
1069 $(TPOOLBJS:%.o=$(LIBCDIR)/port/tpool/%.c) \
1070 $(THREADSOBJS:%.o=$(LIBCDIR)/port/threads/%.c) \
1071 $(THREADSMACHOBJS:%.o=threads/%.c) \
1072 $(UNICODEOBJS:%.o=$(SRC)/common/unicode/%.c) \
1073 $(UNWINDMACHOBJS:%.o=unwind/%.c) \
1074 $(FPOBJS:%.o=fp/%.c) \
1075 $(I386FPOBJS:%.o=$(LIBCDIR)/i386/fp/%.c) \
1076 $(LIBCBASE)/gen/ecvt.c \
1077 $(LIBCBASE)/gen/makectxt.c \
1078 $(LIBCBASE)/gen/siginfofst.c \
1079 $(LIBCBASE)/gen/siglongjmp.c \
1080 $(LIBCBASE)/gen/sync_instruction_memory.c \
1081 $(LIBCBASE)/sys/uadmin.c

1083 # conditional assignments
1084 # $(DYNLIB) $(LIB_PIC) := DYNOBJS = _rtbootld.o
1085 $(DYNLIB) := CRTI = crt.i.o
1086 $(DYNLIB) := CRTN = crtn.o

1088 # Files which need the threads .il inline template
1089 TIL=
1090 aio.o \
1091 alloc.o \
1092 assfail.o \
1093 atexit.o \
1094 atfork.o \
1095 cancel.o \
1096 door_calls.o \
1097 err.o \
1098 errno.o \
1099 lwp.o \
1100 ma.o \
1101 machdep.o \
1102 posix_aio.o \
1103 pthr_attr.o \
1104 pthr_barrier.o \
1105 pthr_cond.o \
1106 pthr_mutex.o \
1107 pthr_rwlock.o \
1108 pthread.o \

```

```

1109 rand.o \
1110 rwlock.o \
1111 scalls.o \
1112 sched.o \
1113 sema.o \
1114 sigaction.o \
1115 sigev_thread.o \
1116 spawn.o \
1117 stack.o \
1118 synch.o \
1119 tdb_agent.o \
1120 thr.o \
1121 thread_interface.o \
1122 thread_pool.o \
1123 thrp_unwind.o \
1124 tls.o \
1125 tmem.o \
1126 tsd.o

1128 $(TIL:%=pics/%) := CFLAGS64 += $(LIBCBASE)/threads/amd64.il

1130 # pics/mul64.o := CFLAGS64 += crt/mul64.il

1132 # large-file-aware components that should be built large

1134 #$(COMSYSOBS64:%=pics/%) := \
1135 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1137 #$(SYSOBS64:%=pics/%) := \
1138 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1140 #$(PORTGEN64:%=pics/%) := \
1141 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1143 #$(PORTSTDIO64:%=pics/%) := \
1144 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1146 #$(PORTSYS64:%=pics/%) := \
1147 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1149 $(PORTSTDIO_W:%=pics/%) := \
1150 CPPFLAGS += -D_WIDE

1152 $(PORTPRINT_W:%=pics/%) := \
1153 CPPFLAGS += -D_WIDE

1155 $(PORTPRINT_C89:%=pics/%) := \
1156 CPPFLAGS += -D_C89_INTMAX32

1158 $(PORTSTDIO_C89:%=pics/%) := \
1159 CPPFLAGS += -D_C89_INTMAX32

1161 $(PORTI18N_COND:%=pics/%) := \
1162 CPPFLAGS += -D_WCS_LOGLONG

1164 .KEEP_STATE:

1166 all: $(LIBS) $(LIB_PIC)

1168 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1169 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1170 lint := LINTFLAGS64 += -mn -erroff=E_SUPPRESSION_DIRECTIVE_UNUSED

1172 lint:
1173 @echo $(LINT.c) ... $(LDLIBS)
1174 @$(LINT.c) $(SRCS) $(LDLIBS)

```

```
1176 $(LINTLIB) := SRCS=$(LIBCDIR)/port/l1ib-1c
1177 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1178 $(LINTLIB) := LINTFLAGS64=-nvx -m64

1180 # object files that depend on inline template
1181 $(TIL:%=pics/%): $(LIBCBASE)/threads/amd64.il
1182 # pics/mul64.o: crt/mul64.il

1184 # include common libc targets
1185 include ../Makefile.targ

1187 # We need to strip out all CTF data from the static library
1188 $(LIB_PIC) := DIR = pics
1189 $(LIB_PIC): pics $$ (PICS)
1190     $(BUILD.AR)
1191     $(MCS) -d -n .SUNW_ctf $@ > /dev/null 2>&1
1192     $(AR) -ts $@ > /dev/null
1193     $(POST_PROCESS_A)

1195 ASSYMDEP_OBJS= \
1196     _lwp_mutex_unlock.o \
1197     _stack_grow.o \
1198     asm_subr.o \
1199     getcontext.o \
1200     setjmp.o \
1201     tls_get_addr.o \
1202     vforkx.o

1204 $(ASSYMDEP_OBJS:%=pics/%): assym.h

1206 # assym.h build rules

1208 GENASSYM_C = genassym.c

1210 genassym: $(GENASSYM_C)
1211     $(NATIVECC) -iinc -I$(LIBCDIR)/inc $(CPPFLAGS.native) \
1212     -o $@ $(GENASSYM_C)

1214 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in

1216 assym.h: $(OFFSETS) genassym
1217     $(OFFSETS_CREATE) <$(OFFSETS) >$@
1218     ./genassym >>$@

1220 # derived C source and related explicit dependencies
1221 $(LIBCDIR)/port/gen/errlst.c + \
1222 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/
1223     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist

1225 pics/errlst.o: $(LIBCDIR)/port/gen/errlst.c

1227 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c
```

```

*****
24030 Fri Jul 11 11:42:54 2014
new/usr/src/lib/libc/i386/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 2004, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2013, Joyent, Inc. All rights reserved.
24 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
25 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
26 #
27 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
28 # Use is subject to license terms.
29 #

31 LIBCDIR=      $(SRC)/lib/libc
32 LIB_PIC=      libc_pic.a
33 VERS=         .1
34 CPP=          /usr/lib/cpp
35 TARGET_ARCH= i386

37 VALUES=     values-Xa.o

39 # objects are grouped by source directory
41 # local objects
42 STRETS=

44 CRTOBS=      \
45      cerror.o \
46      cerror64.o

48 DYNOBJS=     \
49      _rtbootld.o

51 FPOBJS=      \
52      _D_cplx_div.o \
53      _D_cplx_div_ix.o \
54      _D_cplx_div_rx.o \
55      _D_cplx_lr_div.o \
56      _D_cplx_lr_div_ix.o \
57      _D_cplx_lr_div_rx.o \
58      _D_cplx_mul.o \

```

```

59      _F_cplx_div.o \
60      _F_cplx_div_ix.o \
61      _F_cplx_div_rx.o \
62      _F_cplx_lr_div.o \
63      _F_cplx_lr_div_ix.o \
64      _F_cplx_lr_div_rx.o \
65      _F_cplx_mul.o \
66      _X_cplx_div.o \
67      _X_cplx_div_ix.o \
68      _X_cplx_div_rx.o \
69      _X_cplx_lr_div.o \
70      _X_cplx_lr_div_ix.o \
71      _X_cplx_lr_div_rx.o \
72      _X_cplx_mul.o \
73      _base_il.o \
74      fpgetmask.o \
75      fpgetround.o \
76      fpgetsticky.o \
77      fpsetmask.o \
78      fpsetround.o \
79      fpsetsticky.o \
80      fpstart.o

82 FPASMOBJS=  \
83      __xgetRD.o \
84      _xtoll.o \
85      _xtoull.o \
86      fpcw.o

88 ATOMICOBJS= \
89      atomic.o

91 XATTOBJS=   \
92      xattr_common.o

94 COMOBJS=    \
95      bcmp.o \
96      bcopy.o \
97      bsearch.o \
98      bzero.o \
99      qsort.o \
100     strtol.o \
101     strtoul.o \
102     strtoll.o \
103     strtoull.o

105 DTRACEOBJS= \
106     dtrace_data.o

108 GENOBJS=    \
109     _div64.o \
110     _divdi3.o \
111     _getsp.o \
112     _mul64.o \
113     abs.o \
114     alloca.o \
115     byteorder.o \
116     byteorder64.o \
117     cuexit.o \
118     ecvt.o \
119     errlst.o \
120     i386_data.o \
121     ladd.o \
122     ldivide.o \
123     lmul.o \
124     lock.o \

```



```

125     lshiftl.o           \|
126     lsign.o            \|
127     lsub.o              \|
128     makectxt.o         \|
129     memccpy.o           \|
130     memchr.o            \|
131     memcmp.o            \|
132     memcpy.o            \|
133     memset.o            \|
134     new_list.o          \|
135     setjmp.o            \|
136     siginfolst.o        \|
137     siglongjmp.o        \|
138     strcat.o            \|
139     strchr.o            \|
140     strcmp.o            \|
141     strcpy.o            \|
142     strlen.o            \|
143     strncat.o           \|
144     strncmp.o           \|
145     strncpy.o           \|
146     strnlen.o           \|
147     strrchr.o           \|
148     sync_instruction_memory.o \|

150 # sysobjs that contain large-file interfaces
151 COMSYSOBS64=           \|
152     fstatvfs64.o        \|
153     getdents64.o        \|
154     getrlimit64.o       \|
155     lseek64.o           \|
156     mmap64.o            \|
157     pread64.o           \|
158     pwrite64.o          \|
159     setrlimit64.o       \|
160     statvfs64.o         \|

162 SYSOBS64=

164 COMSYSOBS=            \|
165     __clock_timer.o     \|
166     __getloadavg.o      \|
167     __rusagesys.o       \|
168     __signotify.o       \|
169     __sigrt.o           \|
170     __time.o            \|
171     _lgrp_home_fast.o   \|
172     _lgrpsys.o          \|
173     _nfssys.o           \|
174     _portfs.o           \|
175     _pset.o             \|
176     _rpcsys.o           \|
177     _sigaction.o        \|
178     _so_accept.o        \|
179     _so_bind.o          \|
180     _so_connect.o       \|
181     _so_getpeername.o   \|
182     _so_getsockname.o   \|
183     _so_getsockopt.o    \|
184     _so_listen.o        \|
185     _so_recv.o          \|
186     _so_recvfrom.o      \|
187     _so_recvmsg.o       \|
188     _so_send.o          \|
189     _so_sendmsg.o       \|
190     _so_sendto.o        \|

```

```

191     _so_setsockopt.o    \|
192     _so_shutdown.o     \|
193     _so_socket.o        \|
194     _so_socketpair.o    \|
195     _sockconfig.o       \|
196     acct.o              \|
197     acl.o                \|
198     adjtime.o           \|
199     alarm.o              \|
200     brk.o                \|
201     chdir.o              \|
202     chroot.o             \|
203     cladm.o              \|
204     close.o              \|
205     execve.o            \|
206     exit.o               \|
207     facl.o               \|
208     fchdir.o             \|
209     fchroot.o            \|
210     fdsync.o             \|
211     fpathconf.o          \|
212     fstatfs.o           \|
213     fstatvfs.o          \|
214     getcpuid.o           \|
215     getdents.o           \|
216     getegid.o            \|
217     geteuid.o            \|
218     getgid.o             \|
219     getgroups.o          \|
220     gethrtime.o          \|
221     getitimer.o          \|
222     getmsg.o             \|
223     getpid.o             \|
224     getpmsg.o            \|
225     getppid.o            \|
226     getrlimit.o         \|
227     getuid.o             \|
228     gtty.o               \|
229     install_ustrap.o    \|
230     ioctl.o              \|
231     kaio.o               \|
232     kill.o               \|
233     llseek.o             \|
234     lseek.o              \|
235     mmapobjsys.o        \|
236     memcntl.o           \|
237     mincore.o            \|
238     mmap.o               \|
239     modctl.o             \|
240     mount.o              \|
241     mprotect.o           \|
242     munmap.o             \|
243     nice.o               \|
244     ntp_adjtime.o        \|
245     ntp_gettime.o        \|
246     p_online.o           \|
247     pathconf.o           \|
248     pause.o              \|
249     pcsample.o           \|
250     pipe2.o              \|
251     pollsys.o            \|
252     pread.o              \|
253     priocntlset.o        \|
254     processor_bind.o     \|
255     processor_info.o     \|
256     profil.o             \|

```

```

257      putmsg.o          \
258      putpmsg.o        \
259      pwrite.o         \
260      read.o           \
261      readv.o          \
262      resolvepath.o    \
263      seteguid.o       \
264      setgid.o         \
265      setgroups.o      \
266      setitimer.o     \
267      setreid.o       \
268      setrlimit.o     \
269      setuid.o         \
270      sigaltstk.o      \
271      sigprocmsk.o    \
272      sigsendset.o    \
273      sigsuspend.o    \
274      statfs.o        \
275      statvfs.o       \
276      stty.o          \
277      sync.o          \
278      sysconfig.o     \
279      sysfs.o         \
280      sysinfo.o       \
281      syslwp.o        \
282      times.o         \
283      ulimit.o        \
284      umask.o         \
285      umount2.o       \
286      utssys.o        \
287      uucopy.o        \
288      vhangup.o       \
289      waitid.o        \
290      write.o         \
291      writev.o        \
292      yield.o         \
\
294 SYSOBSJS= \
295     __clock_gettime.o \
296     __getcontext.o   \
297     __uadmin.o       \
298     _lwp_mutex_unlock.o \
299     _stack_grow.o    \
300     door.o           \
301     forkx.o          \
302     forkallx.o       \
303     getcontext.o     \
304     gettimeofday.o  \
305     lwp_private.o    \
306     nuname.o         \
307     ptrace.o         \
308     syscall.o        \
309     sysi86.o         \
310     tls_get_addr.o   \
311     uadmin.o         \
312     umount.o         \
313     uname.o          \
314     vforkx.o         \
315     xstat.o          \
\
317 # objects under $(LIBCDIR)/port which contain transitional large file interfaces
318 PORTGEN64= \
319     _xftw64.o        \
320     attropen64.o     \
321     ftw64.o          \
322     mkstemp64.o      \

```

```

323     nftw64.o         \
324     tell64.o        \
325     truncate64.o    \
\
327 # objects from source under $(LIBCDIR)/port
328 PORTFP= \
329     __flt_decim.o   \
330     __flt_rounds.o  \
331     __tbl_10_b.o    \
332     __tbl_10_h.o    \
333     __tbl_10_s.o    \
334     __tbl_2_b.o     \
335     __tbl_2_h.o     \
336     __tbl_2_s.o     \
337     __tbl_fdq.o     \
338     __tbl_tens.o    \
339     __x_power.o     \
340     _base_sup.o     \
341     aconvert.o      \
342     decimal_bin.o   \
343     double_decim.o  \
344     econvert.o      \
345     fconvert.o      \
346     file_decim.o    \
347     finite.o        \
348     fp_data.o       \
349     func_decim.o    \
350     gconvert.o      \
351     hex_bin.o       \
352     ieee_globals.o  \
353     pack_float.o    \
354     sigfpe.o        \
355     string_decim.o  \
\
357 PORTGEN= \
358     _env_data.o     \
359     _xftw.o         \
360     a64l.o          \
361     abort.o         \
362     addsev.o        \
363     ascii_strcasecmp.o \
364     ascii_strncasecmp.o \
365     assert.o        \
366     atof.o          \
367     atoi.o          \
368     atol.o          \
369     atoll.o         \
370     attrat.o        \
371     attropen.o      \
372     atexit.o        \
373     atfork.o        \
374     basename.o      \
375     calloc.o        \
376     catgets.o       \
377     catopen.o       \
378     cfgetispeed.o   \
379     cfgetospeed.o  \
380     cfree.o         \
381     cfsetispeed.o   \
382     cfsetospeed.o  \
383     cftime.o        \
384     clock.o         \
385     closedir.o      \
386     closefrom.o     \
387     confstr.o       \
388     crypt.o         \

```

```

389      csetlen.o      \
390      ctime.o        \
391      ctime_r.o     \
392      daemon.o      \
393      deflt.o        \
394      directio.o    \
395      dirname.o     \
396      div.o          \
397      drand48.o     \
398      dup.o          \
399      env_data.o    \
400      err.o          \
401      errno.o       \
402      euclen.o      \
403      event_port.o  \
404      execvp.o      \
405      fattach.o     \
406      fdetach.o     \
407      fdopendir.o   \
408      ffs.o          \
409      fls.o          \
410      ffmtmsg.o     \
411      ftime.o       \
412      ftok.o        \
413      ftw.o         \
414      gcvt.o        \
415      getauxv.o     \
416      getcwd.o      \
417      getdate_err.o \
418      getdtblsize.o \
419      getenv.o      \
420      getexecname.o \
421      getgrnam.o    \
422      getgrnam_r.o  \
423      gethostid.o   \
424      gethostname.o \
425      gethz.o       \
426      getisax.o     \
427      getloadavg.o  \
428      getlogin.o    \
429      getmntent.o   \
430      getnetgrent.o \
431      get_nprocs.o  \
432      getopt.o      \
433      getopt_long.o \
434      getpagesize.o \
435      getpw.o        \
436      getpwnam.o    \
437      getpwnam_r.o  \
438      getrusage.o   \
439      getspent.o    \
440      getspent_r.o  \
441      getsubopt.o   \
442      gettxt.o      \
443      getusershell.o \
444      getut.o       \
445      getutx.o      \
446      getvfsent.o   \
447      getwd.o       \
448      getwidth.o    \
449      getxby_door.o \
450      gtxt.o        \
451      hsearch.o     \
452      iconv.o       \
453      imaxabs.o     \
454      imaxdiv.o     \

```

```

455      index.o       \
456      initgroups.o \
457      insque.o      \
458      isaexec.o     \
459      isastream.o   \
460      isatty.o      \
461      killpg.o      \
462      klpdlib.o     \
463      l64a.o        \
464      lckpwdf.o     \
465      lconstants.o \
466      lexp10.o     \
467      lfind.o       \
468      lfmt.o        \
469      lfmt_log.o    \
470      llabs.o       \
471      lldiv.o       \
472      llog10.o     \
473      lltostr.o     \
474      localtime.o  \
475      lsearch.o     \
476      madvise.o    \
477      malloc.o      \
478      memalign.o   \
479      memmem.o      \
480      mkdev.o       \
481      mkdtemp.o     \
482      mkfifo.o      \
483      mkstemp.o     \
484      mktemp.o      \
485      mlock.o       \
486      mlockall.o   \
487      mon.o         \
488      msync.o       \
489      munlock.o     \
490      munlockall.o \
491      ndbm.o        \
492      nftw.o        \
493      nlspath_checks.o \
494      nsparse.o     \
495      nss_common.o  \
496      nss_dbdefs.o  \
497      nss_deffinder.o \
498      opendir.o     \
499      opt_data.o    \
500      perror.o      \
501      pfmt.o        \
502      pfmt_data.o   \
503      pfmt_print.o  \
504      pipe.o        \
505      plock.o       \
506      poll.o        \
507      posix_fadvise.o \
508      posix_fallocate.o \
509      posix_madvise.o \
510      posix_memalign.o \
511      priocntl.o    \
512      privlib.o     \
513      priv_str_xlate.o \
514      psiginfo.o    \
515      psignal.o     \
516      pt.o          \
517      putpwent.o    \
518      putspent.o    \
519      raise.o       \
520      rand.o        \

```

```

521 random.o \
522 rctlops.o \
523 readdir.o \
524 readdir_r.o \
525 realpath.o \
526 reboot.o \
527 regexpr.o \
528 remove.o \
529 rewinddir.o \
530 rindex.o \
531 scandir.o \
532 seekdir.o \
533 select.o \
534 select_large_fdset.o \
535 setlabel.o \
536 setpriority.o \
537 settimeofday.o \
538 sh_locks.o \
539 sigflag.o \
540 siglist.o \
541 sigsend.o \
542 sigsetops.o \
543 signal.o \
544 stack.o \
545 stpcpy.o \
546 stpncpy.o \
547 str2sig.o \
548 strcase_charmap.o \
549 strchrnul.o \
550 strcspn.o \
551 strdup.o \
552 strerror.o \
553 strlcat.o \
554 strlcpy.o \
555 strndup.o \
556 strpbrk.o \
557 strsep.o \
558 strsignal.o \
559 strspn.o \
560 strstr.o \
561 strtod.o \
562 strtolmax.o \
563 strtok.o \
564 strtok_r.o \
565 strtoumax.o \
566 swab.o \
567 swapctl.o \
568 sysconf.o \
569 syslog.o \
570 tcdrain.o \
571 tcfLOW.o \
572 tcfLush.o \
573 tcgetattr.o \
574 tcgetpgrp.o \
575 tcgetsid.o \
576 tcsendbreak.o \
577 tcsetattr.o \
578 tcsetpgrp.o \
579 tell.o \
580 telldir.o \
581 tfind.o \
582 time_data.o \
583 time_gdata.o \
584 tls_data.o \
585 truncate.o \
586 tsdalloc.o \

```

```

587 tsearch.o \
588 ttyname.o \
589 ttyslot.o \
590 ualarm.o \
591 ucred.o \
592 valloc.o \
593 vlfmt.o \
594 vpfmt.o \
595 waitpid.o \
596 walkstack.o \
597 wdata.o \
598 xgetwidth.o \
599 xpg4.o \
600 xpg6.o \
602 PORTPRINT_W= \
603 doprnt_w.o \
605 PORTPRINT= \
606 asprintf.o \
607 doprnt.o \
608 fprintf.o \
609 printf.o \
610 snprintf.o \
611 sprintf.o \
612 vfprintf.o \
613 vprintf.o \
614 vsnprintf.o \
615 vsprintf.o \
616 vwprintf.o \
617 wprintf.o \
619 # c89 variants to support 32-bit size of c89 u/intmax_t (32-bit libc only)
620 PORTPRINT_C89= \
621 vfprintf_c89.o \
622 vprintf_c89.o \
623 vsnprintf_c89.o \
624 vsprintf_c89.o \
625 vwprintf_c89.o \
627 PORTSTDIO_C89= \
628 vscanf_c89.o \
629 vwscanf_c89.o \
631 # portable stdio objects that contain large file interfaces.
632 # Note: fopen64 is a special case, as we build it small.
633 PORTSTDIO64= \
634 fopen64.o \
635 fpos64.o \
637 PORTSTDIO_W= \
638 doscan_w.o \
640 PORTSTDIO= \
641 __extensions.o \
642 __endopen.o \
643 __filbuf.o \
644 __findbuf.o \
645 __flsbuf.o \
646 __wrtchk.o \
647 clearerr.o \
648 ctermid.o \
649 ctermid_r.o \
650 cuserid.o \
651 data.o \
652 doscan.o \

```

```

653      fdopen.o      \|
654      feof.o        \|
655      ferrord.o     \|
656      fgetc.o       \|
657      fgets.o       \|
658      fileno.o      \|
659      flockf.o      \|
660      flush.o       \|
661      fopen.o       \|
662      fpos.o        \|
663      fputc.o       \|
664      fputs.o       \|
665      fread.o       \|
666      fseek.o       \|
667      fseeko.o      \|
668      ftell.o       \|
669      ftello.o      \|
670      fwrite.o      \|
671      getc.o        \|
672      getchar.o     \|
673      getline.o    \|
674      getpass.o     \|
675      gets.o        \|
676      getw.o        \|
677      mse.o         \|
678      popen.o       \|
679      putc.o        \|
680      putchar.o    \|
681      puts.o        \|
682      putw.o        \|
683      rewind.o      \|
684      scanf.o       \|
685      setbuf.o      \|
686      setbuffer.o  \|
687      setvbuf.o    \|
688      system.o     \|
689      tempnam.o    \|
690      tmpfile.o    \|
691      tmpnam_r.o   \|
692      ungetc.o     \|
693      vscanf.o     \|
694      vwscanf.o    \|
695      wscanf.o     \|

697 PORTI18N= \|
698      getwchar.o   \|
699      putwchar.o   \|
700      putws.o      \|
700      strcasecmp.o \|
701      strcasestr.o \|
702      strncasecmp.o \|
701      strtows.o   \|
702      wcsnlen.o   \|
703      wcsstr.o    \|
704      wcstoimax.o \|
705      wcstol.o    \|
706      wcstoul.o   \|
707      wcsvsws.o   \|
708      wmemchr.o   \|
709      wmemcmp.o   \|
710      wmemcpy.o   \|
711      wmemmove.o  \|
712      wmemset.o   \|
715      wscasecmp.o \|
713      wscat.o     \|
714      wschr.o     \|

```

```

715      wscmp.o      \|
716      wscpy.o     \|
717      wscspn.o    \|
718      wsdup.o     \|
719      wslen.o     \|
723      wscasecmp.o \|
720      wscat.o     \|
721      wscmp.o     \|
722      wscpy.o     \|
723      wspbkr.o    \|
724      wsprintf.o  \|
725      wsrchr.o   \|
726      wsscanf.o  \|
727      wsspn.o     \|
728      wstod.o     \|
729      wstok.o     \|
730      wstol.o     \|
731      wstoll.o   \|
732      wsxfrm.o   \|
733      gettext.o   \|
734      gettext_gnu.o \|
735      gettext_real.o \|
736      gettext_util.o \|
741      isdigit.o   \|
737      plural_parser.o \|
738      wdresolve.o \|
739      _ctype.o    \|
740      isascii.o   \|
741      toascii.o   \|

743 PORTI18N_COND= \|
744      wcstol_longlong.o \|
745      wcstoul_longlong.o \|

747 PORTLOCALE= \|
748      big5.o      \|
749      btowc.o     \|
750      collate.o   \|
751      collcmp.o   \|
752      euc.o       \|
753      fnmatch.o   \|
754      fgetwc.o    \|
755      fgetws.o    \|
756      fix_grouping.o \|
757      fputwc.o    \|
758      fputws.o    \|
759      fwide.o     \|
760      gb18030.o   \|
761      gb2312.o   \|
762      gbk.o       \|
763      getdate.o   \|
764      isdigit.o   \|
765      iswctype.o  \|
766      ldpart.o    \|
767      lmessages.o \|
768      lnumeric.o  \|
769      lmonetary.o \|
770      localeconv.o \|
771      localeimpl.o \|
772      mbftowc.o   \|
773      mblen.o     \|
774      mbrlen.o    \|
775      mbrtowc.o   \|
776      mbsinit.o   \|
777      mbsnrtowcs.o \|
778      mbsrtowcs.o \|

```

```

779      mbstowcs.o      \
780      mbtowc.o        \
781      mskanji.o       \
782      nextwctype.o   \
783      nl_langinfo.o  \
784      none.o          \
785      regcomp.o       \
786      regfree.o       \
787      regerror.o      \
788      regexec.o       \
789      rune.o          \
790      runetype.o      \
791      setlocale.o     \
792      setrunelocale.o \
793      strcasecmp.o   \
794      strcasestr.o  \
795      strcoll.o       \
796      strfmon.o       \
797      strftime.o      \
798      strncasecmp.o \
799      strptime.o      \
800      strxfrm.o       \
801      table.o         \
802      timelocal.o    \
803      tolower.o       \
804      tolower.o       \
805      ungetwc.o       \
806      utf8.o          \
807      wcrtoomb.o     \
808      wcscasecmp.o  \
809      wcscoll.o       \
810      wcsftime.o      \
811      wcsnrtombs.o   \
812      wcsrtombs.o    \
813      wcswidth.o     \
814      wcstombs.o     \
815      wcsxfrm.o       \
816      wctob.o         \
817      wctomb.o        \
818      wctrans.o       \
819      wctype.o        \
820      wcwidth.o       \
821      wscoll.o        \
\
823 AIOBJS= \
824      aio.o \
825      aio_alloc.o \
826      posix_aio.o \
\
828 RTOBJS= \
829      clock_timer.o \
830      mqueue.o \
831      pos4obj.o \
832      sched.o \
833      sem.o \
834      shm.o \
835      sigev_thread.o \
\
837 TPOOLBJS= \
838      thread_pool.o \
\
840 THREADSOBJS= \
841      alloc.o \
842      assfail.o \
843      cancel.o \
844      door_calls.o \

```

```

845      tmem.o \
846      pthr_attr.o \
847      pthr_barrier.o \
848      pthr_cond.o \
849      pthr_mutex.o \
850      pthr_rwlock.o \
851      pthread.o \
852      rwlock.o \
853      scalls.o \
854      sema.o \
855      sigaction.o \
856      spawn.o \
857      synch.o \
858      tdb_agent.o \
859      thr.o \
860      thread_interface.o \
861      tls.o \
862      tsd.o \
\
864 THREADSMACHOBJS= \
865      machdep.o \
\
867 THREADSASMOBJS= \
868      asm_subr.o \
\
870 UNICODEOBJS= \
871      u8_textprep.o \
872      uconv.o \
\
874 UNWINDMACHOBJS= \
875      unwind.o \
\
877 UNWINDASMOBJS= \
878      unwind_frame.o \
\
880 # objects that implement the transitional large file API
881 PORTSYS64= \
882      lockf64.o \
883      stat64.o \
\
885 PORTSYS= \
886      _autofssys.o \
887      access.o \
888      acctctl.o \
889     bsd_signal.o \
890      chmod.o \
891      chown.o \
892      corectl.o \
893      exacctsyes.o \
894      execl.o \
895      execl.o \
896      execv.o \
897      fcntl.o \
898      getpagesizes.o \
899      getpeerucred.o \
900      inst_sync.o \
901      issetugid.o \
902      label.o \
903      link.o \
904      lockf.o \
905      lwp.o \
906      lwp_cond.o \
907      lwp_rwlock.o \
908      lwp_sigmask.o \
909      meminfosys.o \
910      mkdir.o \

```

```

911      mknod.o           \
912      msgsys.o          \
913      nfssys.o          \
914      open.o            \
915      pgrpsys.o         \
916      posix_sigwait.o   \
917      ppriv.o           \
918      psetsys.o         \
919      rctlsys.o         \
920      readlink.o        \
921      rename.o          \
922      sbrk.o            \
923      semsys.o          \
924      set_errno.o       \
925      sharefs.o         \
926      shmsys.o          \
927      sidsys.o          \
928      siginterrupt.o    \
929      signal.o           \
930      sigpending.o      \
931      sigstack.o        \
932      stat.o            \
933      symlink.o         \
934      tasksys.o         \
935      time.o            \
936      time_util.o       \
937      ucontext.o        \
938      unlink.o          \
939      ustat.o           \
940      utimesys.o        \
941      zone.o            \

943 PORTREGEX=          \
944      glob.o           \
945      regcmp.o         \
946      regex.o          \
947      wordexp.o        \

949 MOSTOBSJS=          \
950      $(STRETS)         \
951      $(CRTOBSJS)       \
952      $(DYNOBJS)        \
953      $(FPOBJS)         \
954      $(FPASMOBJS)      \
955      $(ATOMICOBJS)     \
956      $(XATTROBJS)      \
957      $(COMOBSJS)       \
958      $(DTRACEOBSJS)    \
959      $(GENOBJS)        \
960      $(PORTFTP)        \
961      $(PORTGEN)        \
962      $(PORTGEN64)      \
963      $(PORTI18N)       \
964      $(PORTI18N_COND)  \
965      $(PORTLOCALE)     \
966      $(PORTPRINT)      \
967      $(PORTPRINT_C89)  \
968      $(PORTPRINT_W)    \
969      $(PORTREGEX)      \
970      $(PORTSTDIO)       \
971      $(PORTSTDIO64)    \
972      $(PORTSTDIO_C89)  \
973      $(PORTSTDIO_W)    \
974      $(PORTSYS)        \
975      $(PORTSYS64)      \
976      $(AIOOBSJS)       \

```

```

977      $(RTOBSJS)        \
978      $(TPOOLOBSJS)     \
979      $(THREASOBSJS)    \
980      $(THREADSMACHOBSJS) \
981      $(THREADSASMOBJS) \
982      $(UNICODEOBSJS)   \
983      $(UNWINDMACHOBSJS) \
984      $(UNWINDASMOBJS)  \
985      $(COMSYSOBSJS)    \
986      $(SYSOBSJS)       \
987      $(COMSYSOBSJS64)  \
988      $(SYSOBSJS64)    \
989      $(VALUES)         \

991 TRACEOBSJS=         \
992      plockstat.o      \

994 # NOTE: libc.so.1 must be linked with the minimal crti.o and crtn.o
995 # modules whose source is provided in the $(SRC)/lib/common directory.
996 # This must be done because otherwise the Sun C compiler would insert
997 # its own versions of these modules and those versions contain code
998 # to call out to C++ initialization functions. Such C++ initialization
999 # functions can call back into libc before thread initialization is
1000 # complete and this leads to segmentation violations and other problems.
1001 # Since libc contains no C++ code, linking with the minimal crti.o and
1002 # crtn.o modules is safe and avoids the problems described above.
1003 OBJECTS= $(CRTI) $(MOSTOBSJS) $(CRTN)
1004 CRTSRCS= ../../common/i386

1006 LDPASS_OFF= $(POUND_SIGN)

1008 # include common library definitions
1009 include ../../Makefile.lib

1011 # we need to override the default SONAME here because we might
1012 # be building a variant object (still libc.so.1, but different filename)
1013 SONAME = libc.so.1

1015 CFLAGS += $(CCVERBOSE) $(CTF_FLAGS)

1017 # This is necessary to avoid problems with calling _ex_unwind().
1018 # We probably don't want any inlining anyway.
1019 XINLINE = -xinline=
1020 CFLAGS += $(XINLINE)

1022 CERRWARN += -_gcc=-Wno-parentheses
1023 CERRWARN += -_gcc=-Wno-switch
1024 CERRWARN += -_gcc=-Wno-uninitialized
1025 CERRWARN += -_gcc=-Wno-unused-value
1026 CERRWARN += -_gcc=-Wno-unused-label
1027 CERRWARN += -_gcc=-Wno-unused-variable
1028 CERRWARN += -_gcc=-Wno-type-limits
1029 CERRWARN += -_gcc=-Wno-char-subscripts
1030 CERRWARN += -_gcc=-Wno-clobbered
1031 CERRWARN += -_gcc=-Wno-unused-function
1032 CERRWARN += -_gcc=-Wno-address

1034 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
1035 # enables ASSERT() checking in the threads portion of the library.
1036 # This is automatically enabled for DEBUG builds, not for non-debug builds.
1037 THREAD_DEBUG =
1038 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

1040 # Make string literals read-only to save memory.
1041 CFLAGS += $(XSTRCONST)

```

```

1043 ALTPICS= $(TRACEOBSJ:%=pics/%)
1045 $(DYNLIB) := BUILD.SO = $(LD) -o $$@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) \
1046             $(EXTPICS) $(LDLIBS)
1048 MAPFILES = $(LIBCDIR)/port/mapfile-vers
1050 #
1051 # EXTN_CPPFLAGS and EXTN_CFLAGS set in enclosing Makefile
1052 #
1053 CFLAGS += $(EXTN_CFLAGS)
1054 CPPFLAGS= -D_REENTRANT -Di386 $(EXTN_CPPFLAGS) $(THREAD_DEBUG) \
1055           -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1056 ASFLAGS= $(AS_PICFLAGS) -P -D__STDC__ -D_ASM $(CPPFLAGS) $(i386_AS_XARCH)

1058 # As a favor to the dtrace syscall provider, libc still calls the
1059 # old syscall traps that have been obsoleted by the *at() interfaces.
1060 # Delete this to compile libc using only the new *at() system call traps
1061 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1063 # Inform the run-time linker about libc specialized initialization
1064 RTLDINFO = -z rtldinfo=tls_rtldinfo
1065 DYNFLAGS += $(RTLDINFO)

1067 # Force libc's internal references to be resolved immediately upon loading
1068 # in order to avoid critical region problems. Since almost all libc symbols
1069 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1070 DYNFLAGS += -znw

1072 DYNFLAGS += -e __rtboot
1073 DYNFLAGS += $(EXTN_DYNFLAGS)

1075 # Inform the kernel about the initial DTrace area (in case
1076 # libc is being used as the interpreter / runtime linker).
1077 DTRACE_DATA = -zdtrace=dtrace_data
1078 DYNFLAGS += $(DTRACE_DATA)

1080 # DTrace needs an executable data segment.
1081 MAPFILE.NED=

1083 BUILD.s= $(AS) $(ASFLAGS) $< -o $$@

1085 # Override this top level flag so the compiler builds in its native
1086 # C99 mode. This has been enabled to support the complex arithmetic
1087 # added to libc.
1088 C99MODE= $(C99_ENABLE)

1090 # libc method of building an archive
1091 # The "$(GREP) -v ' L '" part is necessary only until
1092 # lorder is fixed to ignore thread-local variables.
1093 BUILD.AR= $(RM) $$@ ; \
1094           $(AR) q $$@ '$(LORDER) $(MOSTOBSJ:%=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR

1096 # extra files for the clean target
1097 CLEANFILES= \
1098           $(LIBCDIR)/port/gen/errlst.c \
1099           $(LIBCDIR)/port/gen/new_list.c \
1100           assym.h \
1101           genassym \
1102           crt_rtld.s \
1103           crt_rtbootld.s \
1104           pics_rtbootld.o \
1105           pics/crti.o \
1106           pics/crtn.o \
1107           $(ALTPICS)

```

```

1109 CLOBBERFILES += $(LIB_PIC)

1111 # list of C source for lint
1112 SRCS=
1113     $(ATOMICOBJS:%.o=$(SRC)/common/atomic/%.c) \
1114     $(XATTROBJS:%.o=$(SRC)/common/xattr/%.c) \
1115     $(COMOBSJ:%.o=$(SRC)/common/util/%.c) \
1116     $(DTRACEOBSJ:%.o=$(SRC)/common/dtrace/%.c) \
1117     $(PORTFP:%.o=$(LIBCDIR)/port/fp/%.c) \
1118     $(PORTGEN:%.o=$(LIBCDIR)/port/gen/%.c) \
1119     $(PORTI18N:%.o=$(LIBCDIR)/port/i18n/%.c) \
1120     $(PORTLOCALE:%.o=$(LIBCDIR)/port/locale/%.c) \
1121     $(PORTPRINT:%.o=$(LIBCDIR)/port/print/%.c) \
1122     $(PORTREGEX:%.o=$(LIBCDIR)/port/regex/%.c) \
1123     $(PORTSTDIO:%.o=$(LIBCDIR)/port/stdio/%.c) \
1124     $(PORTSYS:%.o=$(LIBCDIR)/port/sys/%.c) \
1125     $(AIOOBSJ:%.o=$(LIBCDIR)/port/aio/%.c) \
1126     $(RTOBSJ:%.o=$(LIBCDIR)/port/rt/%.c) \
1127     $(TPOOLOBSJ:%.o=$(LIBCDIR)/port/tpool/%.c) \
1128     $(THREADSOBSJ:%.o=$(LIBCDIR)/port/threads/%.c) \
1129     $(THREADSMACHOBSJ:%.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1130     $(UNICODEOBSJ:%.o=$(SRC)/common/unicode/%.c) \
1131     $(UNWINDMACHOBSJ:%.o=$(LIBCDIR)/port/unwind/%.c) \
1132     $(FPOBSJ:%.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1133     $(LIBCBASE)/gen/ecvt.c \
1134     $(LIBCBASE)/gen/makeectxt.c \
1135     $(LIBCBASE)/gen/signfolst.c \
1136     $(LIBCBASE)/gen/siglongjmp.c \
1137     $(LIBCBASE)/gen/strcmp.c \
1138     $(LIBCBASE)/gen/sync_instruction_memory.c \
1139     $(LIBCBASE)/sys/ptrace.c \
1140     $(LIBCBASE)/sys/uadmin.c

1142 # conditional assignments
1143 $(DYNLIB) := CRTI = crti.o
1144 $(DYNLIB) := CRTN = crtn.o

1146 # Files which need the threads .il inline template
1147 TIL= \
1148     aio.o \
1149     alloc.o \
1150     assfail.o \
1151     atexit.o \
1152     atfork.o \
1153     cancel.o \
1154     door_calls.o \
1155     err.o \
1156     errno.o \
1157     lwp.o \
1158     ma.o \
1159     machdep.o \
1160     posix_aio.o \
1161     pthr_attr.o \
1162     pthr_barrier.o \
1163     pthr_cond.o \
1164     pthr_mutex.o \
1165     pthr_rwlock.o \
1166     pthread.o \
1167     rand.o \
1168     rwlock.o \
1169     scalls.o \
1170     sched.o \
1171     sema.o \
1172     sigaction.o \
1173     sigev_thread.o \
1174     spawn.o

```



```

1175     stack.o           \
1176     synch.o           \
1177     tdb_agent.o       \
1178     thr.o             \
1179     thread_interface.o \
1180     thread_pool.o     \
1181     tls.o             \
1182     tsd.o             \
1183     tmem.o            \
1184     unwind.o          \

1186 THREADS_INLINES = $(LIBCBASE)/threads/i386.il
1187 $(TIL:%=pics/%) := CFLAGS += $(THREADS_INLINES)

1189 # pics/mul64.o := CFLAGS += $(LIBCBASE)/crt/mul64.il

1191 # large-file-aware components that should be built large

1193 $(COMSYSOBS64:%=pics/%) := \
1194     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1196 $(SYSOBS64:%=pics/%) := \
1197     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1199 $(PORTGEN64:%=pics/%) := \
1200     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1202 $(PORTSTDIO64:%=pics/%) := \
1203     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1205 $(PORTSYS64:%=pics/%) := \
1206     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1208 $(PORTSTDIO_W:%=pics/%) := \
1209     CPPFLAGS += -D_WIDE

1211 $(PORTPRINT_W:%=pics/%) := \
1212     CPPFLAGS += -D_WIDE

1214 $(PORTPRINT_C89:%=pics/%) := \
1215     CPPFLAGS += -D_C89_INTMAX32

1217 $(PORTSTDIO_C89:%=pics/%) := \
1218     CPPFLAGS += -D_C89_INTMAX32

1220 $(PORTI18N_COND:%=pics/%) := \
1221     CPPFLAGS += -D_WCS_LOGLONG

1223 .KEEP_STATE:

1225 all: $(LIBS) $(LIB_PIC)

1227 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1228 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1229 lint := LINTFLAGS += -mn -eroff=E_SUPPRESSION_DIRECTIVE_UNUSED

1231 lint:
1232     @echo $(LINT.c) ...
1233     @$$(LINT.c) $(SRCS) $(LDLIBS)

1235 $(LINTLIB) := SRCS=$(LIBCDIR)/port/l1ib-1c
1236 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1237 $(LINTLIB) := LINTFLAGS=-nvx

1239 # object files that depend on inline template
1240 $(TIL:%=pics/%): $(LIBCBASE)/threads/i386.il

```

```

1241 # pics/mul64.o: $(LIBCBASE)/crt/mul64.il

1243 # include common libc targets
1244 include $(LIBCDIR)/Makefile.targ

1246 # We need to strip out all CTF and DOF data from the static library
1247 $(LIB_PIC) := DIR = pics
1248 $(LIB_PIC): pics $$ (PICS)
1249     $(BUILD.AR)
1250     $(MCS) -d -n .SUNW_ctf $$@ > /dev/null 2>&1
1251     $(MCS) -d -n .SUNW_dof $$@ > /dev/null 2>&1
1252     $(AR) -ts $$@ > /dev/null
1253     $(POST_PROCESS_A)

1255 $(LIBCBASE)/crt/_rtbootld.s: $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.c
1256     $(CC) $(CPPFLAGS) $(CTF_FLAGS) -O -S $(C_PICFLAGS) \
1257     $(LIBCBASE)/crt/_rtld.c -o $(LIBCBASE)/crt/_rtld.s
1258     $(CAT) $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.s > $$@
1259     $(RM) $(LIBCBASE)/crt/_rtld.s

1261 # partially built from C source
1262 pics/_rtbootld.o: $(LIBCBASE)/crt/_rtbootld.s
1263     $(AS) $(ASFLAGS) $(LIBCBASE)/crt/_rtbootld.s -o $$@
1264     $(CTFCONVERT_O)

1266 ASSYMDEP_OBJS= \
1267     _lwp_mutex_unlock.o \
1268     _stack_grow.o \
1269     getcontext.o \
1270     setjmp.o \
1271     tls_get_addr.o \
1272     vforkx.o

1274 $(ASSYMDEP_OBJS:%=pics/%) := CPPFLAGS += -I.

1276 $(ASSYMDEP_OBJS:%=pics/%): assym.h

1278 # assym.h build rules

1280 GENASSYM_C = $(LIBCDIR)/$(MACH)/genassym.c

1282 genassym: $(GENASSYM_C)
1283     $(NATIVECC) -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc \
1284     -D__EXTENSIONS__ $(CPPFLAGS.native) -o $$@ $(GENASSYM_C)

1286 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in

1288 assym.h: $(OFFSETS) genassym
1289     $(OFFSETS_CREATE) <$(OFFSETS) >$$@
1290     ./genassym >>$$@

1292 # derived C source and related explicit dependencies
1293 $(LIBCDIR)/port/gen/errlst.c + \
1294 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/
1295     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist

1297 pics/errlst.o: $(LIBCDIR)/port/gen/errlst.c

1299 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c

```

```

*****
3927 Fri Jul 11 11:42:55 2014
new/usr/src/lib/libc/port/locale/_ctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3 * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4 * Copyright (c) 1989, 1993
5 * The Regents of the University of California. All rights reserved.
6 * (c) UNIX System Laboratories, Inc.
7 * All or some portions of this file are derived from material licensed
8 * to the University of California by American Telephone and Telegraph
9 * Co. or Unix System Laboratories, Inc. and are reproduced herein with
10 * the permission of UNIX System Laboratories, Inc.
11 *
12 * This code is derived from software contributed to Berkeley by
13 * Paul Borman at Krystal Technologies.
14 *
15 * Redistribution and use in source and binary forms, with or without
16 * modification, are permitted provided that the following conditions
17 * are met:
18 * 1. Redistributions of source code must retain the above copyright
19 * notice, this list of conditions and the following disclaimer.
20 * 2. Redistributions in binary form must reproduce the above copyright
21 * notice, this list of conditions and the following disclaimer in the
22 * documentation and/or other materials provided with the distribution.
23 * 3. Neither the name of the University nor the names of its contributors
24 * may be used to endorse or promote products derived from this software
25 * without specific prior written permission.
26 *
27 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
28 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
29 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
30 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
31 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
32 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
33 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
34 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
35 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
36 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
37 * SUCH DAMAGE.
38 */

40 #ifndef __CTYPE_H_
41 #define __CTYPE_H_

43 /*
44 * Please take careful note. It turns out that the _ISxxx macros
45 * occupy the lower order byte, except for _ISGRAPH, _ISALPHA, and _ISPRINT.
46 * Those occupy 0x2000, 0x4000, and 0x8000. Now, noting *very* carefully,
47 * it turns out that this leaves some gaps in the extended bits, which
48 * are occupied by _E1 = phonogram, _E2 = ideogram, and _E3 = English.
49 * The *other* _Ex bits are reserved. We don't think these higher order
50 * bits are baked into applications (because they haven't been used before),
51 * so we believe it is safe to reuse them how we see fit.
52 *
53 * This makes it possible to define a single space which overlaps both the
54 * wctype types, and the ctype types.
55 */

57 #define _CTYPE_A _ISALPHA /* Alpha */
58 #define _CTYPE_C _ISCNTRL /* Control */

```

```

59 #define _CTYPE_D _ISDIGIT /* Digit */
60 #define _CTYPE_G _ISGRAPH /* Graph */
61 #define _CTYPE_L _ISLOWER /* Lower */
62 #define _CTYPE_P _ISPUNCT /* Punct */
63 #define _CTYPE_S _ISSPACE /* Space */
64 #define _CTYPE_U _ISUPPER /* Upper */
65 #define _CTYPE_X _ISXDIGIT /* X digit */
66 #define _CTYPE_B _ISBLANK /* Blank */
67 #define _CTYPE_R _ISPRINT /* Print */

69 #define _CTYPE_Q _E1 /* Phonogram */
70 #define _CTYPE_I _E2 /* Ideogram */
71 #define _CTYPE_E _E3 /* English (Solaris) */
72 #define _CTYPE_N _E4 /* Number */
73 #define _CTYPE_T _E5 /* Special */

75 /* These high order bits were never used for anything at all. */
76 #define _CTYPE_SW0 0x20000000 /* 0 width character */
77 #define _CTYPE_SW1 0x40000000 /* 1 width character */
78 #define _CTYPE_SW2 0x80000000 /* 2 width character */
79 #define _CTYPE_SW3 0xc0000000 /* 3 width character */
80 #define _CTYPE_SWM 0xe0000000 /* Mask for screen width data */
81 #define _CTYPE_SWS 30 /* Bits to shift to get width */

82 unsigned int __runetype(int);

83 #endif /* !_CTYPE_H_ */

```

```

*****
4970 Fri Jul 11 11:42:55 2014
new/usr/src/lib/libc/port/locale/big5.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins. All rights reserved.
5  * Copyright (c) 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * This code is derived from software contributed to Berkeley by
9  * Paul Borman at Krystal Technologies.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 * notice, this list of conditions and the following disclaimer in the
18 * documentation and/or other materials provided with the distribution.
19 * 4. Neither the name of the University nor the names of its contributors
20 * may be used to endorse or promote products derived from this software
21 * without specific prior written permission.
22 *
23 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
24 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
25 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
26 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
27 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
28 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
29 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
30 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
31 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
32 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
33 * SUCH DAMAGE.
34 */
35
36 #include "lint.h"
37 #include <sys/types.h>
38 #include <errno.h>
39 #include "runetype.h"
40 #include <stdlib.h>
41 #include <string.h>
42 #include <wchar.h>
43 #include "mblocal.h"
44
45 static size_t _BIG5_mbrtowc(wchar_t *_RESTRICT_KYWD,
46                             const char *_RESTRICT_KYWD,
47                             size_t, mbstate_t *_RESTRICT_KYWD);
48 static int _BIG5_mbsinit(const mbstate_t *);
49 static size_t _BIG5_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
50                             mbstate_t *_RESTRICT_KYWD);
51 static size_t _BIG5_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
52                                const char **_RESTRICT_KYWD, size_t, size_t,
53                                mbstate_t *_RESTRICT_KYWD);
54 static size_t _BIG5_wcsnrtombs(char *_RESTRICT_KYWD,
55                                const wchar_t **_RESTRICT_KYWD, size_t, size_t,
56                                mbstate_t *_RESTRICT_KYWD);

```

```

58 typedef struct {
59     wchar_t ch;
60 } _BIG5State;
61
62 void
63 _BIG5_init(struct lc_ctype *lct)
64 {
65     lct->lc_mbrtowc = _BIG5_mbrtowc;
66     lct->lc_wcrtomb = _BIG5_wcrtomb;
67     lct->lc_mbsnrtowcs = _BIG5_mbsnrtowcs;
68     lct->lc_wcsnrtombs = _BIG5_wcsnrtombs;
69     lct->lc_mbsinit = _BIG5_mbsinit;
70     lct->lc_max_mblen = 2;
71     lct->lc_is_ascii = 0;
72
73     __mbrtowc = _BIG5_mbrtowc;
74     __wcrtomb = _BIG5_wcrtomb;
75     __mbsinit = _BIG5_mbsinit;
76     __CurrentRuneLocale = rl;
77     __ctype[520] = 2;
78     charset_is_ascii = 0;
79     return (0);
80 }
81
82 unchanged_portion_omitted
83
84 static size_t
85 _BIG5_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc, mbstate_t *_RESTRICT_KYWD ps)
86 {
87     _BIG5State *bs;
88
89     bs = (_BIG5State *)ps;
90
91     if (bs->ch != 0) {
92         errno = EINVAL;
93         return ((size_t)-1);
94     }
95
96     if (s == NULL)
97         /* Reset to initial shift state (no-op) */
98         return (1);
99     if (wc & 0x8000) {
100         *s++ = (wc >> 8) & 0xff;
101         *s = wc & 0xff;
102         return (2);
103     }
104     *s = wc & 0xff;
105     return (1);
106 }
107
108 static size_t
109 _BIG5_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
110                  size_t nms, size_t len, mbstate_t *_RESTRICT_KYWD ps)
111 {
112     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _BIG5_mbrtowc));
113 }
114
115 static size_t
116 _BIG5_wcsnrtombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
117                  size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps)
118 {
119     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _BIG5_wcrtomb));
120 }
121
122 unchanged_portion_omitted

```

new/usr/src/lib/libc/port/locale/btowc.c

1

```
*****
2072 Fri Jul 11 11:42:55 2014
new/usr/src/lib/libc/port/locale/btowc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002, 2003 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <stdio.h>
31 #include <wchar.h>
32 #include <locale.h>
33 #include <xlocale.h>
34 #include "mblocal.h"

36 wint_t
37 btowc_l(int c, locale_t loc)
38 {
39     static const mbstate_t initial = { 0 };
40     mbstate_t mbs = initial;
41     char cc;
42     wchar_t wc;

44     if (c == EOF)
45         return (WEOF);
46     /*
47      * We expect mbrtowc() to return 0 or 1, hence the check for n > 1
48      * which detects error return values as well as "impossible" byte
49      * counts.
50      */
51     cc = (char)c;
52     if (mbrtowc_l(&wc, &cc, 1, &mbs, loc) > 1)
49     if (__mbrtowc(&wc, &cc, 1, &mbs) > 1)
53         return (WEOF);
54     return (wc);
55 }
```

new/usr/src/lib/libc/port/locale/btowc.c

2

```
57 wint_t
58 btowc(int c)
59 {
60     return (btowc_l(c, uselocale(NULL)));
61 }
_____unchanged_portion_omitted_____
```

```

*****
14317 Fri Jul 11 11:42:55 2014
new/usr/src/lib/libc/port/locale/collate.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copright 2010 Nexenta Systems, Inc. All rights reserved.
5  * Copyright (c) 1995 Alex Tatmanjants <alex@elvisti.kiev.ua>
6  *         at Electronni Visti IA, Kiev, Ukraine.
7  *         All rights reserved.
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include "file64.h"
32 #include <stdio.h>
33 #include <stdlib.h>
34 #include <stddef.h>
35 #include <string.h>
36 #include <wchar.h>
37 #include <errno.h>
38 #include <unistd.h>
39 #include <ctype.h>
40 #include <unistd.h>
41 #include <fcntl.h>
42 #include <assert.h>
43 #include <sys/stat.h>
44 #include <sys/mman.h>

46 #include "collate.h"
47 #include "setlocale.h"
48 #include "ldpart.h"

49 /*
50 * See the comments in usr/src/cmd/localedef/collate.c for further
51 * information. It would also be very helpful to have a copy of the
52 * POSIX standard for collation (in the locale format manual page)
53 * handy (www.opengroup.org).
54 */

56 /*

```

```

57 * POSIX uses empty tables and falls down to strcmp.
58 */
59 struct lc_collate lc_collate_posix = {
60     .lc_is_posix = 1,
61 };
62 static collate_subst_t      *subst_table[COLL_WEIGHTS_MAX];
63 static collate_char_t      *char_pri_table;
64 static collate_large_t     *large_pri_table;
65 static collate_chain_t     *chain_pri_table;
66 static char                 *cache = NULL;
67 static size_t               cachesz;
68 static char                 collate_encoding[ENCODING_LEN + 1];

69 struct locdata __posix_collate_locdata = {
70     .l_name = "C",
71     .l_data = { &lc_collate_posix }
72 };
73 /* Exposed externally to other parts of libc. */
74 collate_info_t             *_collate_info;
75 int _collate_load_error = 1;

76 struct locdata *
77 __lc_collate_load(const char *locname)
78 int
79 __collate_load_tables(const char *encoding)
80 {
81     int i, chains, z;
82     char buf[PATH_MAX];
83     char *TMP;
84     char *map;
85     collate_info_t *info;
86     struct stat sbuf;
87     int fd;
88     struct locdata *ldata;
89     struct lc_collate *lcc;

90     /* 'encoding' must be already checked. */
91     if (strcmp(encoding, "C") == 0 || strcmp(encoding, "POSIX") == 0) {
92         _collate_load_error = 1;
93         return (_LDP_CACHE);
94     }

95     /*
96      * If the locale name is the same as our cache, use the cache.
97      */
98     if (cache && (strncmp(encoding, collate_encoding, ENCODING_LEN) == 0)) {
99         _collate_load_error = 0;
100        return (_LDP_CACHE);
101    }

102    /*
103     * Slurp the locale file into the cache.
104     */

105    (void) snprintf(buf, sizeof (buf), "%s/%s/LC_COLLATE/LCL_DATA",
106        _PathLocale, locname);
107    _PathLocale, encoding);

108    if ((fd = open(buf, O_RDONLY)) < 0) {
109        errno = EINVAL;
110        return (NULL);
111    }

112    if ((fd = open(buf, O_RDONLY)) < 0)
113        return (_LDP_ERROR);
114    if (fstat(fd, &sbuf) < 0) {

```

```

94     (void) close(fd);
95     errno = EINVAL;
96     return (NULL);
105    return (_LDP_ERROR);
97 }
98 if (sbuf.st_size < (COLLATE_STR_LEN + sizeof (info))) {
99     (void) close(fd);
100    errno = EINVAL;
101    return (NULL);
110    return (_LDP_ERROR);
102 }
103 map = mmap(NULL, sbuf.st_size, PROT_READ, MAP_PRIVATE, fd, 0);
104 (void) close(fd);
105 if ((TMP = map) == NULL) {
106     errno = EINVAL;
107     return (NULL);
115    return (_LDP_ERROR);
108 }
110 if (strcmp(TMP, COLLATE_VERSION, COLLATE_STR_LEN) != 0) {
111     (void) munmap(map, sbuf.st_size);
112     errno = EINVAL;
113     return (NULL);
121    return (_LDP_ERROR);
114 }
115 TMP += COLLATE_STR_LEN;
117 info = (void *)TMP;
118 TMP += sizeof (*info);
120 if ((info->directive_count < 1) ||
121     (info->directive_count >= COLL_WEIGHTS_MAX) ||
122     ((chains = info->chain_count) < 0)) {
123     (void) munmap(map, sbuf.st_size);
124     errno = EINVAL;
125     return (NULL);
133    return (_LDP_ERROR);
126 }
128 i = (sizeof (collate_char_t) * (UCHAR_MAX + 1)) +
129     (sizeof (collate_chain_t) * chains) +
130     (sizeof (collate_large_t) * info->large_count);
131 for (z = 0; z < info->directive_count; z++) {
132     for (z = 0; z < (info->directive_count); z++) {
133         i += sizeof (collate_subst_t) * info->subst_count[z];
134     }
135     if (i != (sbuf.st_size - (TMP - map))) {
136         (void) munmap(map, sbuf.st_size);
137         errno = EINVAL;
138         return (NULL);
145         return (_LDP_ERROR);
138 }
141 if ((ldata = __locdata_alloc(locname, sizeof (*lcc))) == NULL) {
142     (void) munmap(map, sbuf.st_size);
143     return (NULL);
144 }
145 lcc = ldata->l_data[0];
146 ldata->l_map = map;
147 ldata->l_map_len = sbuf.st_size;
149 lcc->lc_info = info;
150 lcc->lc_directive_count = info->directive_count;
151 lcc->lc_large_count = info->large_count;

```

```

153     for (z = 0; z < COLL_WEIGHTS_MAX; z++) {
154         lcc->lc_directive[z] = info->directive[z];
155         lcc->lc_subst_count[z] = info->subst_count[z];
156         lcc->lc_pri_count[z] = info->pri_count[z];
157         lcc->lc_undef_pri[z] = info->undef_pri[z];
158     }
160 lcc->lc_char_table = (void *)TMP;
148 char_pri_table = (void *)TMP;
161 TMP += sizeof (collate_char_t) * (UCHAR_MAX + 1);
163 for (z = 0; z < lcc->lc_directive_count; z++) {
164     int count;
165     if ((count = lcc->lc_subst_count[z]) > 0) {
166         lcc->lc_subst_table[z] = (void *)TMP;
167         TMP += count * sizeof (collate_subst_t);
151     for (z = 0; z < info->directive_count; z++) {
152         if (info->subst_count[z] > 0) {
153             subst_table[z] = (void *)TMP;
154             TMP += info->subst_count[z] * sizeof (collate_subst_t);
168         } else {
169             lcc->lc_subst_table[z] = NULL;
156             subst_table[z] = NULL;
170         }
171     }
173     if (chains > 0) {
174         lcc->lc_chain_table = (void *)TMP;
161         chain_pri_table = (void *)TMP;
175         TMP += chains * sizeof (collate_chain_t);
176     } else
177         lcc->lc_chain_table = NULL;
178     lcc->lc_chain_count = chains;
179     if (lcc->lc_large_count > 0)
180         lcc->lc_large_table = (void *)TMP;
164         chain_pri_table = NULL;
165     if (info->large_count > 0)
166         large_pri_table = (void *)TMP;
181     else
182         lcc->lc_large_table = NULL;
168         large_pri_table = NULL;
184     return (ldata);
170     (void) strncpy(collate_encoding, encoding, ENCODING_LEN);
171     _collate_info = info;
173     if (cache)
174         (void) munmap(cache, cachesz);
176     cache = map;
177     cachesz = sbuf.st_size;
178     _collate_load_error = 0;
180     return (_LDP_LOADED);
185 }
187 static const int32_t *
188 substsearch(const struct lc_collate *lcc, const wchar_t key, int pass)
183 static int32_t *
184 substsearch(const wchar_t key, int pass)
189 {
190     const collate_subst_t *p;
191     int n = lcc->lc_subst_count[pass];
186     collate_subst_t *p;
187     int n = _collate_info->subst_count[pass];

```

```

193     if (n == 0)
194         return (NULL);

196     if (pass >= lcc->lc_directive_count)
197         if (pass >= _collate_info->directive_count)
198             return (NULL);

199     if (!(key & COLLATE_SUBST_PRIORITY))
200         return (NULL);

202     p = lcc->lc_subst_table[pass] + (key & ~COLLATE_SUBST_PRIORITY);
198     p = subst_table[pass] + (key & ~COLLATE_SUBST_PRIORITY);
203     assert(p->key == key);
204     return (p->pri);
205 }

207 /*
208  * Note: for performance reasons, we have expanded bsearch here. This avoids
209  * function call overhead with each comparison.
210  */

212 static collate_chain_t *
213 chainsearch(const struct lc_collate *lcc, const wchar_t *key, int *len)
209 chainsearch(const wchar_t *key, int *len)
214 {
215     int low;
216     int high;
217     int next, compar, l;
218     collate_chain_t *p;
219     collate_chain_t *tab;

221     if (lcc->lc_info->chain_count == 0)
217     if (_collate_info->chain_count == 0)
222         return (NULL);

224     low = 0;
225     high = lcc->lc_info->chain_count - 1;
226     tab = lcc->lc_chain_table;
221     high = _collate_info->chain_count - 1;
222     tab = chain_pri_table;

228     while (low <= high) {
229         next = (low + high) / 2;
230         p = tab + next;
231         compar = *key - *p->str;
232         if (compar == 0) {
233             l = wcsnlen(p->str, COLLATE_STR_LEN);
234             compar = wcsncmp(key, p->str, l);
235             if (compar == 0) {
236                 *len = l;
237                 return (p);
238             }
239         }
240         if (compar > 0)
241             low = next + 1;
242         else
243             high = next - 1;
244     }
245     return (NULL);
246 }

248 static collate_large_t *
249 largesearch(const struct lc_collate *lcc, const wchar_t key)
245 largesearch(const wchar_t key)
250 {
251     int low = 0;

```

```

252     int high = lcc->lc_info->large_count - 1;
248     int high = _collate_info->large_count - 1;
253     int next, compar;
254     collate_large_t *p;
255     collate_large_t *tab = lcc->lc_large_table;
251     collate_large_t *tab = large_pri_table;

257     if (lcc->lc_info->large_count == 0)
253     if (_collate_info->large_count == 0)
258         return (NULL);

260     while (low <= high) {
261         next = (low + high) / 2;
262         p = tab + next;
263         compar = key - p->val;
264         if (compar == 0)
265             return (p);
266         if (compar > 0)
267             low = next + 1;
268         else
269             high = next - 1;
270     }
271     return (NULL);
272 }

274 void
275 _collate_lookup(const struct lc_collate *lcc, const wchar_t *t,
276                int *len, int *pri, int which, const int **state)
271 _collate_lookup(const wchar_t *t, int *len, int *pri, int which, int **state)
277 {
278     collate_chain_t *p2;
279     collate_large_t *match;
275     collate_info_t *info = _collate_info;
280     int p, l;
281     const int *sptr;
277     int *sptr;

283     /*
284      * If this is the "last" pass for the UNDEFINED, then
285      * we just return the priority itself.
286      */
287     if (which >= lcc->lc_directive_count) {
283     if (which >= info->directive_count) {
288         *pri = *t;
289         *len = l;
290         *state = NULL;
291         return;
292     }

294     /*
295      * If we have remaining substitution data from a previous
296      * call, consume it first.
297      */
298     if ((sptr = *state) != NULL) {
299         *pri = *sptr;
300         sptr++;
301         *state = *sptr ? sptr : NULL;
302         *len = 0;
303         return;
304     }

306     /* No active substitutions */
307     *len = 1;

309     /*
310      * Check for composites such as diphthongs that collate as a

```

```

311     * single element (aka chains or collating-elements).
312     */
313     if (((p2 = chainsearch(lcc, t, &l1)) != NULL) &&
309     if (((p2 = chainsearch(t, &l1)) != NULL) &&
314     ((p = p2->pri[which]) >= 0)) {
316         *len = 1;
317         *pri = p;
319     } else if (*t <= UCHAR_MAX) {
321         /*
322         * Character is a small (8-bit) character.
323         * We just look these up directly for speed.
324         */
325         *pri = lcc->lc_char_table[*t].pri[which];
321         *pri = char_pri_table[*t].pri[which];
327     } else if ((lcc->lc_info->large_count > 0) &&
328     ((match = largesearch(lcc, *t)) != NULL)) {
323     } else if ((info->large_count > 0) &&
324     ((match = largesearch(*t)) != NULL)) {
330         /*
331         * Character was found in the extended table.
332         */
333         *pri = match->pri.pri[which];
335     } else {
336         /*
337         * Character lacks a specific definition.
338         */
339         if (lcc->lc_directive[which] & DIRECTIVE_UNDEFINED) {
335         if (info->directive[which] & DIRECTIVE_UNDEFINED) {
340             /* Mask off sign bit to prevent ordering confusion. */
341             *pri = (*t & COLLATE_MAX_PRIORITY);
342         } else {
343             *pri = lcc->lc_undef_pri[which];
339             *pri = info->undef_pri[which];
344         }
345         /* No substitutions for undefined characters! */
346         return;
347     }
349     /*
350     * Try substituting (expanding) the character. We are
351     * currently doing this *after* the chain compression. I
352     * think it should not matter, but this way might be slightly
353     * faster.
354     *
355     * We do this after the priority search, as this will help us
356     * to identify a single key value. In order for this to work,
357     * its important that the priority assigned to a given element
358     * to be substituted be unique for that level. The localedef
359     * code ensures this for us.
360     */
361     if ((sptr = substsearch(lcc, *pri, which)) != NULL) {
357     if ((sptr = substsearch(*pri, which)) != NULL) {
362         if ((*pri = *sptr) != 0) {
363             sptr++;
364             *state = *sptr ? sptr : NULL;
365         }
366     }
368 }

```

```

370 /*
371 * This is the meaty part of wcsxfrm & strxfrm. Note that it does
372 * NOT NULL terminate. That is left to the caller.
373 */
374 size_t
375 _collate_wxfrm(const struct lc_collate *lcc, const wchar_t *src, wchar_t *xf,
376               size_t room)
377 _collate_wxfrm(const wchar_t *src, wchar_t *xf, size_t room)
378 {
379     int          pri;
380     int          len;
381     const wchar_t *t;
382     wchar_t      *tr = NULL;
383     int          direc;
384     int          pass;
385     const int32_t *state;
379     int32_t      *state;
385     size_t      want = 0;
386     size_t      need = 0;
387     int          ndir = lcc->lc_directive_count;
389     assert(src);
391     for (pass = 0; pass <= ndir; pass++) {
385     for (pass = 0; pass <= _collate_info->directive_count; pass++) {
393         state = NULL;
395         if (pass != 0) {
396             /* insert level separator from the previous pass */
397             if (room) {
398                 *xf++ = 1;
399                 room--;
400             }
401             want++;
402         }
404         /* special pass for undefined */
405         if (pass == ndir) {
399         if (pass == _collate_info->directive_count) {
406             direc = DIRECTIVE_FORWARD | DIRECTIVE_UNDEFINED;
407         } else {
408             direc = lcc->lc_directive[pass];
402             direc = _collate_info->directive[pass];
409         }
411         t = src;
413         if (direc & DIRECTIVE_BACKWARD) {
414             wchar_t *bp, *fp, c;
415             if (tr)
416                 free(tr);
417             if ((tr = wcsdup(t)) == NULL) {
418                 errno = ENOMEM;
419                 goto fail;
420             }
421             bp = tr;
422             fp = tr + wcslen(tr) - 1;
423             while (bp < fp) {
424                 c = *bp;
425                 *bp++ = *fp;
426                 *fp-- = c;
427             }
428             t = (const wchar_t *)tr;
429         }

```



```

431     if (direc & DIRECTIVE_POSITION) {
432         while (*t || state) {
433             _collate_lookup(lcc, t, &len, &pri, pass,
434                             &state);
435             _collate_lookup(t, &len, &pri, pass, &state);
436             t += len;
437             if (pri <= 0) {
438                 if (pri < 0) {
439                     errno = EINVAL;
440                     goto fail;
441                 }
442                 pri = COLLATE_MAX_PRIORITY;
443             }
444             if (room) {
445                 *xf++ = pri;
446                 room--;
447             }
448             want++;
449             need = want;
450         }
451     } else {
452         while (*t || state) {
453             _collate_lookup(lcc, t, &len, &pri, pass,
454                             &state);
455             _collate_lookup(t, &len, &pri, pass, &state);
456             t += len;
457             if (pri <= 0) {
458                 if (pri < 0) {
459                     errno = EINVAL;
460                     goto fail;
461                 }
462                 continue;
463             }
464             if (room) {
465                 *xf++ = pri;
466                 room--;
467             }
468             want++;
469             need = want;
470         }
471     }
472 end:
473     if (tr)
474         free(tr);
475     return (need);
476
477 fail:
478     if (tr)
479         free(tr);
480     return ((size_t)(-1));
481 }
482
483 /*
484  * In the non-POSIX case, we transform each character into a string of
485  * characters representing the character's priority. Since char is usually
486  * signed, we are limited by 7 bits per byte. To avoid zero, we need to add
487  * XFRM_OFFSET, so we can't use a full 7 bits. For simplicity, we choose 6
488  * bits per byte.
489  *
490  * It turns out that we sometimes have real priorities that are
491  * 31-bits wide. (But: be careful using priorities where the high
492  * order bit is set -- i.e. the priority is negative. The sort order
493  * may be surprising!)
494  */

```

```

495  * TODO: This would be a good area to optimize somewhat. It turns out
496  * that real priorities *except for the last UNDEFINED pass* are generally
497  * very small. We need the localedef code to precalculate the max
498  * priority for us, and ideally also give us a mask, and then we could
499  * severely limit what we expand to.
500  */
501 #define XFRM_BYTES      6
502 #define XFRM_OFFSET    ('0') /* make all printable characters */
503 #define XFRM_SHIFT     6
504 #define XFRM_MASK      ((1 << XFRM_SHIFT) - 1)
505 #define XFRM_SEP       ('.') /* chosen to be less than XFRM_OFFSET */
506
507 static int
508 xfrm(locale_t loc, unsigned char *p, int pri, int pass)
509 xfrm(unsigned char *p, int pri, int pass)
510 {
511     /* we use unsigned to ensure zero fill on right shift */
512     uint32_t val = (uint32_t)loc->collate->lc_pri_count[pass];
513     uint32_t val = (uint32_t)_collate_info->pri_count[pass];
514     int nc = 0;
515
516     while (val) {
517         *p = (pri & XFRM_MASK) + XFRM_OFFSET;
518         pri >>= XFRM_SHIFT;
519         val >>= XFRM_SHIFT;
520         p++;
521         nc++;
522     }
523     return (nc);
524 }
525
526 size_t
527 _collate_sxfrm(const wchar_t *src, char *xf, size_t room, locale_t loc)
528 _collate_sxfrm(const wchar_t *src, char *xf, size_t room)
529 {
530     int pri;
531     int len;
532     const wchar_t *t;
533     wchar_t *tr = NULL;
534     int direc;
535     int pass;
536     const int32_t *state;
537     int32_t *state;
538     size_t want = 0;
539     size_t need = 0;
540     int b;
541     uint8_t buf[XFRM_BYTES];
542     const struct lc_collate *lcc = loc->collate;
543     int ndir = lcc->lc_directive_count;
544
545     assert(src);
546
547     for (pass = 0; pass <= ndir; pass++) {
548         for (pass = 0; pass <= _collate_info->directive_count; pass++) {
549
550             state = NULL;
551
552             if (pass != 0) {
553                 /* insert level separator from the previous pass */
554                 if (room) {
555                     *xf++ = XFRM_SEP;
556                     room--;
557                 }
558                 want++;
559             }
560         }
561     }

```

```

556 /* special pass for undefined */
557 if (pass == ndir) {
558     if (pass == _collate_info->directive_count) {
559         direc = DIRECTIVE_FORWARD | DIRECTIVE_UNDEFINED;
560     } else {
561         direc = lcc->lc_directive[pass];
562     }
563     t = src;

565     if (direc & DIRECTIVE_BACKWARD) {
566         wchar_t *bp, *fp, c;
567         if (tr)
568             free(tr);
569         if ((tr = wcsdup(t)) == NULL) {
570             errno = ENOMEM;
571             goto fail;
572         }
573         bp = tr;
574         fp = tr + wcslen(tr) - 1;
575         while (bp < fp) {
576             c = *bp;
577             *bp++ = *fp;
578             *fp-- = c;
579         }
580         t = (const wchar_t *)tr;
581     }

583     if (direc & DIRECTIVE_POSITION) {
584         while (*t || state) {
586             _collate_lookup(lcc, t, &len, &pri, pass,
587                             &state);
588             _collate_lookup(t, &len, &pri, pass, &state);
589             t += len;
590             if (pri <= 0) {
591                 if (pri < 0) {
592                     errno = EINVAL;
593                     goto fail;
594                 }
595                 pri = COLLATE_MAX_PRIORITY;
596             }

597             b = xfrm(loc, buf, pri, pass);
598             b = xfrm(buf, pri, pass);
599             want += b;
600             if (room) {
601                 while (b) {
602                     b--;
603                     if (room) {
604                         *xf++ = buf[b];
605                         room--;
606                     }
607                 }
608                 need = want;
609             }
610         } else {
611             while (*t || state) {
612                 _collate_lookup(lcc, t, &len, &pri, pass,
613                                 &state);
614                 _collate_lookup(t, &len, &pri, pass, &state);
615                 t += len;
616                 if (pri <= 0) {
617                     if (pri < 0) {

```

```

617         errno = EINVAL;
618         goto fail;
619     }
620     continue;
621 }

623 b = xfrm(loc, buf, pri, pass);
624 b = xfrm(buf, pri, pass);
625 want += b;
626 if (room) {
627     while (b) {
628         b--;
629         if (room) {
630             *xf++ = buf[b];
631             room--;
632         }
633     }
634 }
635 need = want;
636 }
637 }
638 }

640 end:
641 if (tr)
642     free(tr);
643 return (need);

645 fail:
646 if (tr)
647     free(tr);
648 return ((size_t)(-1));
649 }

```

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/collate.h

1

```
*****
4322 Fri Jul 11 11:42:55 2014
new/usr/src/lib/libc/port/locale/collate.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systemes, Inc. All rights reserved.
4  * Copyright (c) 1995 Alex Tatmanjants <alex@elvisti.kiev.ua>
5  *       at Electronni Visti IA, Kiev, Ukraine.
6  *       All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #ifndef _COLLATE_H
31 #define _COLLATE_H

33 #include <sys/types.h>
34 #include <limits.h>
35 #include <locale.h>
36 #include "localeimpl.h"

38 #define COLLATE_STR_LEN      24          /* should be 64-bit multiple */
39 #define COLLATE_VERSION     "illumosCollate2\n"

41 #define COLLATE_MAX_PRIORITY (0x7fffffff) /* max signed value */
42 #define COLLATE_SUBST_PRIORITY (0x40000000) /* bit indicates subst table */

44 #define DIRECTIVE_UNDEF     0x00
45 #define DIRECTIVE_FORWARD  0x01
46 #define DIRECTIVE_BACKWARD 0x02
47 #define DIRECTIVE_POSITION 0x04
48 #define DIRECTIVE_UNDEFINED 0x08      /* special last weight for UNDEFINED */

50 #define DIRECTIVE_DIRECTION_MASK (DIRECTIVE_FORWARD | DIRECTIVE_BACKWARD)

52 /*
53  * The collate file format is as follows:
54  *
55  * char      version[COLLATE_STR_LEN];      /* must be COLLATE_VERSION
56  * collate_info_t  info;                  /* see below, includes padding
57  * collate_char_pri_t  char_data[256];    /* 8 bit char values
58  * collate_subst_t   subst[*];           /* 0 or more substitutions
```

new/usr/src/lib/libc/port/locale/collate.h

2

```
59 * collate_chain_pri_t  chains[*];        // 0 or more chains
60 * collate_large_pri_t  large[*];        // extended char priorities
61 *
62 * Note that all structures must be 32-bit aligned, as each structure
63 * contains 32-bit member fields. The entire file is mmap'd, so its
64 * critical that alignment be observed. It is not generally safe to
65 * use any 64-bit values in the structures.
66 */

68 typedef struct collate_info {
69     uint8_t directive_count;
70     uint8_t directive[COLL_WEIGHTS_MAX];
71     int32_t pri_count[COLL_WEIGHTS_MAX];
72     int32_t flags;
73     int32_t chain_count;
74     int32_t large_count;
75     int32_t subst_count[COLL_WEIGHTS_MAX];
76     int32_t undef_pri[COLL_WEIGHTS_MAX];
77 } collate_info_t;
    unchanged portion omitted

98 struct lc_collate {
99     int      lc_is_posix;
100     int      _collate_load_tables(const char *);
101     void      _collate_lookup(const wchar_t *, int *, int *, int *, int **);
102     size_t    _collate_wxfirm(const wchar_t *, wchar_t *, size_t);
103     size_t    _collate_sxfirm(const wchar_t *, char *, size_t);
104     int      _collate_range_cmp(wchar_t, wchar_t);

101     uint8_t      lc_directive_count;
102     uint8_t      lc_directive[COLL_WEIGHTS_MAX];
103     int32_t      lc_pri_count[COLL_WEIGHTS_MAX];
104     int32_t      lc_flags;
105     int32_t      lc_chain_count;
106     int32_t      lc_large_count;
107     int32_t      lc_subst_count[COLL_WEIGHTS_MAX];
108     int32_t      lc_undef_pri[COLL_WEIGHTS_MAX];
109     extern int    _collate_load_error;
110     extern int    _collate_substitute_nontrivial;
111     extern collate_info_t *_collate_info;

110     collate_info_t *lc_info;
111     collate_char_t *lc_char_table;
112     collate_large_t *lc_large_table;
113     collate_chain_t *lc_chain_table;
114     collate_subst_t *lc_subst_table[COLL_WEIGHTS_MAX];
115 };

117 void      _collate_lookup(const struct lc_collate *, const wchar_t *,
118     int *, int *, int, const int **);
119 size_t    _collate_wxfirm(const struct lc_collate *, const wchar_t *,
120     wchar_t *, size_t);
121 size_t    _collate_sxfirm(const wchar_t *, char *, size_t, locale_t);
122 int      _collate_range_cmp(wchar_t, wchar_t, locale_t);

124 #endif /* !_COLLATE_H */
```

new/usr/src/lib/libc/port/locale/collcmp.c

1

\*\*\*\*\*

1733 Fri Jul 11 11:42:56 2014

new/usr/src/lib/libc/port/locale/collcmp.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (C) 1996 by Andrey A. Chernov, Moscow, Russia.
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #include "lint.h"
29 #include <string.h>
30 #include <wchar.h>
31 #include "collate.h"

33 /*
34  * Compare two characters using collate - thread safe.
35  * Compare two characters using collate
36  */

37 int
38 _collate_range_cmp(wchar_t c1, wchar_t c2, locale_t loc)
39 _collate_range_cmp(wchar_t c1, wchar_t c2)
40 {
41     wchar_t s1[2], s2[2];
42     static wchar_t s1[2], s2[2];

43     s1[0] = c1;
44     s1[1] = 0;
45     s2[0] = c2;
46     s2[1] = 0;
47     return (wcsoll_l(s1, s2, loc));
48     return (wcsoll(s1, s2));
49 }
50
51 unchanged_portion_omitted
```

```

*****
11749 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/euc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins. All rights reserved.
5  * Copyright (c) 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * This code is derived from software contributed to Berkeley by
9  * Paul Borman at Krystal Technologies.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 *   notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 *   notice, this list of conditions and the following disclaimer in the
18 *   documentation and/or other materials provided with the distribution.
19 * 4. Neither the name of the University nor the names of its contributors
20 *   may be used to endorse or promote products derived from this software
21 *   without specific prior written permission.
22 *
23 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
24 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
25 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
26 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
27 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
28 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
29 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
30 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
31 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
32 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
33 * SUCH DAMAGE.
34 */
35
36 #include "lint.h"
37 #include <errno.h>
38 #include <limits.h>
39 #include <stdlib.h>
40 #include <string.h>
41 #include <wchar.h>
42 #include <sys/types.h>
43 #include <sys/euc.h>
44 #include "runetype.h"
45 #include "mblocal.h"
46 #include "lctype.h"
47
48 static size_t _EUC_mbrtowc_impl(wchar_t *_RESTRICT_KYWD,
49                               const char *_RESTRICT_KYWD,
50                               size_t, mbstate_t *_RESTRICT_KYWD, uint8_t, uint8_t, uint8_t, uint8_t);
51 static size_t _EUC_wrtomb_impl(char *_RESTRICT_KYWD, wchar_t,
52                               mbstate_t *_RESTRICT_KYWD, uint8_t, uint8_t, uint8_t, uint8_t);
53
54 static size_t _EUC_CN_mbrtowc(wchar_t *_RESTRICT_KYWD,
55                               const char *_RESTRICT_KYWD,
56                               size_t, mbstate_t *_RESTRICT_KYWD);
57 static size_t _EUC_JP_mbrtowc(wchar_t *_RESTRICT_KYWD,
58                               const char *_RESTRICT_KYWD,

```

```

58                               size_t, mbstate_t *_RESTRICT_KYWD);
59 static size_t _EUC_KR_mbrtowc(wchar_t *_RESTRICT_KYWD,
60                               const char *_RESTRICT_KYWD,
61                               size_t, mbstate_t *_RESTRICT_KYWD);
62 static size_t _EUC_TW_mbrtowc(wchar_t *_RESTRICT_KYWD,
63                               const char *_RESTRICT_KYWD,
64                               size_t, mbstate_t *_RESTRICT_KYWD);
65
66 static size_t _EUC_CN_wrtomb(char *_RESTRICT_KYWD, wchar_t,
67                               mbstate_t *_RESTRICT_KYWD);
68 static size_t _EUC_JP_wrtomb(char *_RESTRICT_KYWD, wchar_t,
69                               mbstate_t *_RESTRICT_KYWD);
70 static size_t _EUC_KR_wrtomb(char *_RESTRICT_KYWD, wchar_t,
71                               mbstate_t *_RESTRICT_KYWD);
72 static size_t _EUC_TW_wrtomb(char *_RESTRICT_KYWD, wchar_t,
73                               mbstate_t *_RESTRICT_KYWD);
74
75 static size_t _EUC_CN_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
76                               const char **_RESTRICT_KYWD, size_t, size_t,
77                               mbstate_t *_RESTRICT_KYWD);
78 static size_t _EUC_JP_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
79                               const char **_RESTRICT_KYWD, size_t, size_t,
80                               mbstate_t *_RESTRICT_KYWD);
81 static size_t _EUC_KR_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
82                               const char **_RESTRICT_KYWD, size_t, size_t,
83                               mbstate_t *_RESTRICT_KYWD);
84 static size_t _EUC_TW_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
85                               const char **_RESTRICT_KYWD, size_t, size_t,
86                               mbstate_t *_RESTRICT_KYWD);
87
88 static size_t _EUC_CN_wcsnrtombs(char *_RESTRICT_KYWD,
89                               const wchar_t **_RESTRICT_KYWD, size_t, size_t,
90                               mbstate_t *_RESTRICT_KYWD);
91 static size_t _EUC_JP_wcsnrtombs(char *_RESTRICT_KYWD,
92                               const wchar_t **_RESTRICT_KYWD, size_t, size_t,
93                               mbstate_t *_RESTRICT_KYWD);
94 static size_t _EUC_KR_wcsnrtombs(char *_RESTRICT_KYWD,
95                               const wchar_t **_RESTRICT_KYWD, size_t, size_t,
96                               mbstate_t *_RESTRICT_KYWD);
97 static size_t _EUC_TW_wcsnrtombs(char *_RESTRICT_KYWD,
98                               const wchar_t **_RESTRICT_KYWD, size_t, size_t,
99                               mbstate_t *_RESTRICT_KYWD);
100
101 static int _EUC_mbsinit(const mbstate_t *);
102
103 typedef struct {
104     wchar_t ch;
105     int set;
106     int want;
107 } _EucState;
108
109 int
110 static int
111 _EUC_mbsinit(const mbstate_t *ps)
112 {
113     return (ps == NULL || ((const _EucState *)ps)->want == 0);
114 }
115
116 /*
117  * EUC-CN uses CS0, CS1 and CS2 (4 bytes).
118  */
119 void
120 _EUC_CN_init(struct lc_ctype *lct)
121 {
122     int
123     _EUC_CN_init(_RuneLocale *r1)

```

```

121 {
122     lct->lc_mbrtowc = _EUC_CN_mbrtowc;
123     lct->lc_wrtomb = _EUC_CN_wrtomb;
124     lct->lc_mbsnrtowcs = _EUC_CN_mbsnrtowcs;
125     lct->lc_wcsnrtombs = _EUC_CN_wcsnrtombs;
126     lct->lc_mbsinit = _EUC_mbsinit;
93     __mbrtowc = _EUC_CN_mbrtowc;
94     __wrtomb = _EUC_CN_wrtomb;
95     __mbsinit = _EUC_mbsinit;

128     lct->lc_max_mblen = 4;
129     lct->lc_is_ascii = 0;
97     _CurrentRuneLocale = rl;

99     __ctype[520] = 4;
100     charset_is_ascii = 0;
101     return (0);
130 }
    unchanged_portion_omitted_

139 static size_t
140 _EUC_CN_mbsnrtowcs(wchar_t *_RESTRIC KYWD dst,
141     const char **_RESTRIC KYWD src,
142     size_t nms, size_t len, mbstate_t *_RESTRIC KYWD ps)
143 {
144     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_CN_mbrtowc));
145 }

147 static size_t
148 _EUC_CN_wrtomb(char *_RESTRIC KYWD s, wchar_t wc,
149     mbstate_t *_RESTRIC KYWD ps)
150 {
151     return (_EUC_wrtomb_impl(s, wc, ps, SS2, 4, 0, 0));
152 }

154 static size_t
155 _EUC_CN_wcsnrtombs(char *_RESTRIC KYWD dst, const wchar_t **_RESTRIC KYWD src,
156     size_t nwc, size_t len, mbstate_t *_RESTRIC KYWD ps)
157 {
158     return (__wcsnrtombs_std(dst, src, nwc, len, ps, _EUC_CN_wrtomb));
159 }

161 /*
162  * EUC-KR uses only CS0 and CS1.
163  */
164 void
165 _EUC_KR_init(struct lc_ctype *lct)
121 int
122 _EUC_KR_init(_RuneLocale *rl)
166 {
167     lct->lc_mbrtowc = _EUC_KR_mbrtowc;
168     lct->lc_wrtomb = _EUC_KR_wrtomb;
169     lct->lc_mbsnrtowcs = _EUC_KR_mbsnrtowcs;
170     lct->lc_wcsnrtombs = _EUC_KR_wcsnrtombs;
171     lct->lc_mbsinit = _EUC_mbsinit;
124     __mbrtowc = _EUC_KR_mbrtowc;
125     __wrtomb = _EUC_KR_wrtomb;
126     __mbsinit = _EUC_mbsinit;

173     lct->lc_max_mblen = 2;
174     lct->lc_is_ascii = 0;
128     _CurrentRuneLocale = rl;

130     __ctype[520] = 2;
131     charset_is_ascii = 0;
132     return (0);

```

```

175 }
    unchanged_portion_omitted_

184 static size_t
185 _EUC_KR_mbsnrtowcs(wchar_t *_RESTRIC KYWD dst,
186     const char **_RESTRIC KYWD src,
187     size_t nms, size_t len, mbstate_t *_RESTRIC KYWD ps)
188 {
189     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_KR_mbrtowc));
190 }

192 static size_t
193 _EUC_KR_wrtomb(char *_RESTRIC KYWD s, wchar_t wc,
194     mbstate_t *_RESTRIC KYWD ps)
195 {
196     return (_EUC_wrtomb_impl(s, wc, ps, 0, 0, 0, 0));
197 }

199 static size_t
200 _EUC_KR_wcsnrtombs(char *_RESTRIC KYWD dst, const wchar_t **_RESTRIC KYWD src,
201     size_t nwc, size_t len, mbstate_t *_RESTRIC KYWD ps)
202 {
203     return (__wcsnrtombs_std(dst, src, nwc, len, ps, _EUC_KR_wrtomb));
204 }

206 /*
207  * EUC-JP uses CS0, CS1, CS2, and CS3.
208  */
209 void
210 _EUC_JP_init(struct lc_ctype *lct)
152 int
153 _EUC_JP_init(_RuneLocale *rl)
211 {
212     lct->lc_mbrtowc = _EUC_JP_mbrtowc;
213     lct->lc_wrtomb = _EUC_JP_wrtomb;
214     lct->lc_mbsnrtowcs = _EUC_JP_mbsnrtowcs;
215     lct->lc_wcsnrtombs = _EUC_JP_wcsnrtombs;
216     lct->lc_mbsinit = _EUC_mbsinit;
155     __mbrtowc = _EUC_JP_mbrtowc;
156     __wrtomb = _EUC_JP_wrtomb;
157     __mbsinit = _EUC_mbsinit;

218     lct->lc_max_mblen = 3;
219     lct->lc_is_ascii = 0;
159     _CurrentRuneLocale = rl;

161     __ctype[520] = 3;
162     charset_is_ascii = 0;
163     return (0);
220 }
    unchanged_portion_omitted_

229 static size_t
230 _EUC_JP_mbsnrtowcs(wchar_t *_RESTRIC KYWD dst,
231     const char **_RESTRIC KYWD src,
232     size_t nms, size_t len, mbstate_t *_RESTRIC KYWD ps)
233 {
234     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_JP_mbrtowc));
235 }

237 static size_t
238 _EUC_JP_wrtomb(char *_RESTRIC KYWD s, wchar_t wc,
239     mbstate_t *_RESTRIC KYWD ps)
240 {
241     return (_EUC_wrtomb_impl(s, wc, ps, SS2, 2, SS3, 3));
242 }

```

```

244 static size_t
245 _EUC_JP_wcsnrtoombs(char * _RESTRICT_KYWD dst, const wchar_t ** _RESTRICT_KYWD src,
246                    size_t nwc, size_t len, mbstate_t * _RESTRICT_KYWD ps)
247 {
248     return (_wcsnrtoombs_std(dst, src, nwc, len, ps, _EUC_JP_wcrtomb));
249 }

251 /*
252  * EUC-TW uses CS0, CS1, and CS2.
253  */
254 void
255 _EUC_TW_init(struct lc_ctype *lct)
256 {
257     lct->lc_mbrtowc = _EUC_TW_mbrtowc;
258     lct->lc_wcrtomb = _EUC_TW_wcrtomb;
259     lct->lc_mbsnrtoombs = _EUC_TW_mbsnrtoombs;
260     lct->lc_wcsnrtoombs = _EUC_TW_wcsnrtoombs;
261     lct->lc_mbsinit = _EUC_mbsinit;
262     __mbrtowc = _EUC_TW_mbrtowc;
263     __wcrtomb = _EUC_TW_wcrtomb;
264     __mbsinit = _EUC_mbsinit;

263     lct->lc_max_mblen = 4;
264     lct->lc_is_ascii = 0;
265     _CurrentRuneLocale = rl;

266     __ctype[520] = 4;
267     charset_is_ascii = 0;
268     return (0);
269 }

    unchanged portion omitted

274 static size_t
275 _EUC_TW_mbsnrtoombs(wchar_t * _RESTRICT_KYWD dst,
276                    const char ** _RESTRICT_KYWD src,
277                    size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
278 {
279     return (_mbsnrtoombs_std(dst, src, nms, len, ps, _EUC_TW_mbrtowc));
280 }

282 static size_t
283 _EUC_TW_wcrtomb(char * _RESTRICT_KYWD s, wchar_t wc,
284                mbstate_t * _RESTRICT_KYWD ps)
285 {
286     return (_EUC_wcrtomb_impl(s, wc, ps, SS2, 4, 0, 0));
287 }

289 static size_t
290 _EUC_TW_wcsnrtoombs(char * _RESTRICT_KYWD dst, const wchar_t ** _RESTRICT_KYWD src,
291                    size_t nwc, size_t len, mbstate_t * _RESTRICT_KYWD ps)
292 {
293     return (_wcsnrtoombs_std(dst, src, nwc, len, ps, _EUC_TW_wcrtomb));
294 }

296 /*
297  * Common EUC code.
298  */

300 static size_t
301 _EUC_mbrtowc_impl(wchar_t * _RESTRICT_KYWD pwc, const char * _RESTRICT_KYWD s,
302                 size_t n, mbstate_t * _RESTRICT_KYWD ps,
303                 uint8_t cs2, uint8_t cs2width, uint8_t cs3, uint8_t cs3width)
304 {

```

```

305     _EucState *es;
306     int i, want;
307     wchar_t wc;
308     unsigned char ch;

310     es = (_EucState *)ps;

312     if (es->want < 0 || es->want > MB_CUR_MAX) {
313         errno = EINVAL;
314         return ((size_t)-1);
315     }

317     if (s == NULL) {
318         s = "";
319         n = 1;
320         pwc = NULL;
321     }

323     if (n == 0)
324         /* Incomplete multibyte sequence */
325         return ((size_t)-2);

327     if (es->want == 0) {
328         /* Fast path for plain ASCII (CS0) */
329         if (((ch = (unsigned char)*s) & 0x80) == 0) {
330             if (pwc != NULL)
331                 *pwc = ch;
332             return (ch != '\0' ? 1 : 0);
333         }

335         if (ch >= 0xa1) {
336             /* CS1 */
337             want = 2;
338         } else if (ch == cs2) {
339             want = cs2width;
340         } else if (ch == cs3) {
341             want = cs3width;
342         } else {
343             errno = EILSEQ;
344             return ((size_t)-1);
345         }

348         es->want = want;
349         es->ch = 0;
350     } else {
351         want = es->want;
352         wc = es->ch;
353     }

355     for (i = 0; i < MIN(want, n); i++) {
356         wc <<= 8;
357         wc |= *s;
358         s++;
359     }
360     if (i < want) {
361         /* Incomplete multibyte sequence */
362         es->want = want - i;
363         es->ch = wc;
364         return ((size_t)-2);
365     }
366     if (pwc != NULL)
367         *pwc = wc;
368     es->want = 0;
369     return (wc == L'\0' ? 0 : want);
370 }

    unchanged portion omitted

```

```

*****
3763 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/fgetwc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include "mse_int.h"
31 #include "file64.h"
32 #include "mtlib.h"
33 #include <errno.h>
34 #include <stdio.h>
35 #include <stdlib.h>
36 #include <wchar.h>
37 #include "mblocal.h"
38 #include "stdiom.h"
39 #include "localeimpl.h"
40 #include "lctype.h"

42 /*
43  * Non-MT-safe version.
44  */
45 wint_t
46 _fgetwc_unlocked_l(FILE *fp, locale_t loc)
47 _fgetwc_unlocked(FILE *fp)
48 {
49     wchar_t wc;
50     size_t nconv;
51     int c;
52     mbstate_t *statep;
53     const struct lc_ctype *lct;

54     if ((c = GETC(fp)) == EOF)
55         return (WEOF);

57     lct = loc->ctype;

```

```

58     if (lct->lc_max_mblen == 1) {
59         if (MB_CUR_MAX == 1) {
60             /* Fast path for single-byte encodings. */
61             return ((wint_t)c);
62         }
63         if ((statep = _getmbstate(fp)) == NULL) {
64             fp->_flag = _IOERR;
65             errno = EBADF;
66             return (WEOF);
67         }
68         do {
69             char x = (char)c;
70             nconv = lct->lc_mbrtowc(&wc, &x, 1, statep);
71             nconv = __mbrtowc(&wc, &x, 1, statep);
72             if (nconv == (size_t)-1) {
73                 break;
74             } else if (nconv == (size_t)-2) {
75                 /* Incompletely decoded, consume another char */
76                 continue;
77             } else if (nconv == 0) {
78                 /*
79                  * Assume that the only valid representation of
80                  * the null wide character is a single null byte.
81                  */
82                 return (L'\0');
83             } else {
84                 return (wc);
85             }
86         } while ((c = GETC(fp)) != EOF);

87     /*
88      * If we got here it means we got truncated in a character, or
89      * the character did not decode properly. Note that in the case
90      * of a botched decoding, we don't UNGETC the bad bytes. Should
91      * we?
92      */
93     fp->_flag |= _IOERR;
94     errno = EILSEQ;
95     return (WEOF);

97 wint_t
98 _fgetwc_unlocked(FILE *fp)
99 {
100     return (_fgetwc_unlocked_l(fp, uselocale(NULL)));
101 }

104 /*
105  * MT safe version
106  */
107 #undef getwc
108 #pragma weak getwc = fgetwc
109 wint_t
110 fgetwc(FILE *fp)
111 {
112     wint_t r;
113     rmutex_t *l;
114     locale_t loc = uselocale(NULL);

116     FLOCKFILE(l, fp);
117     r = _fgetwc_unlocked_l(fp, loc);
118     r = _fgetwc_unlocked(fp);
119     FUNLOCKFILE(l);

120     return (r);

```



```
121 }

109 #undef getwc
110 wint_t
111 getwc(FILE *fp)
112 {
113     return (getwc(fp));
114 }

123 /*
124  * XPG5 version.
125  */
126 #undef __getwc_xpg5
127 #pragma weak __getwc_xpg5 = __fgetwc_xpg5
128 wint_t
129 __fgetwc_xpg5(FILE *fp)
130 {
131     wint_t      r;
132     rmutex_t    *l;
133     locale_t     loc = uselocale(NULL);

135     FLOCKFILE(l, fp);
136     if (GET_NO_MODE(fp))
137         _setorientation(fp, _WC_MODE);
138     r = _fgetwc_unlocked_l(fp, loc);
139     r = _fgetwc_unlocked(fp);
140     FUNLOCKFILE(l);

141     return (r);
142 }

144 #pragma weak getwc_l = fgetwc_l
145 #undef __getwc_xpg5
146 wint_t
147 fgetwc_l(FILE *fp, locale_t loc)
148 __getwc_xpg5(FILE *fp)
149 {
150     wint_t      r;
151     rmutex_t    *l;
152     FLOCKFILE(l, fp);
153     if (GET_NO_MODE(fp))
154         _setorientation(fp, _WC_MODE);
155     r = _fgetwc_unlocked_l(fp, loc);
156     FUNLOCKFILE(l);

157     return (r);
158     return (__fgetwc_xpg5(fp));
159 }

unchanged_portion_omitted_
```

new/usr/src/lib/libc/port/locale/fnmatch.c

1

```
*****
8483 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/fnmatch.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright (c) 1989, 1993, 1994
3  *   The Regents of the University of California. All rights reserved.
4  *
5  * This code is derived from software contributed to Berkeley by
6  * Guido van Rossum.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 *   notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 *   notice, this list of conditions and the following disclaimer in the
15 *   documentation and/or other materials provided with the distribution.
16 * 4. Neither the name of the University nor the names of its contributors
17 *   may be used to endorse or promote products derived from this software
18 *   without specific prior written permission.
19 *
20 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
21 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
22 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
23 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
24 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
25 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
26 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
27 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
28 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
29 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
30 * SUCH DAMAGE.
31 */
32
33 /*
34  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
35  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
36  * Use is subject to license terms.
37 */
38
39 /*
40  * Function fnmatch() as specified in POSIX 1003.2-1992, section B.6.
41  * Compares a filename or pathname to a pattern.
42 */
43
44 /*
45  * Some notes on multibyte character support:
46  * 1. Patterns with illegal byte sequences match nothing.
47  * 2. Illegal byte sequences in the "string" argument are handled by treating
48  *   them as single-byte characters with a value of the first byte of the
49  *   sequence cast to wchar_t.
50  * 3. Multibyte conversion state objects (mbstate_t) are passed around and
51  *   used for most, but not all, conversions. Further work will be required
52  *   to support state-dependent encodings.
53 */
54
55 #include "lint.h"
56 #include <fnmatch.h>
57 #include <limits.h>
58 #include <string.h>
```

new/usr/src/lib/libc/port/locale/fnmatch.c

2

```
59 #include <wchar.h>
60 #include <xlocale.h>
61 #include <wctype.h>
62 #include "localeimpl.h"
63
64 #include "collate.h"
65
66 #define EOS '\0'
67
68 #define RANGE_MATCH 1
69 #define RANGE_NOMATCH 0
70 #define RANGE_ERROR (-1)
71
72 static int rangematch(const char *, wchar_t, int, char **, mbstate_t *,
73 locale_t);
74 static int rangematch(const char *, wchar_t, int, char **, mbstate_t *);
75 static int fnmatchl(const char *, const char *, const char *, int, mbstate_t,
76 mbstate_t, locale_t);
77 mbstate_t);
78
79 int
80 fnmatch(pattern, string, flags)
81 const char *pattern, *string;
82 int flags;
83 {
84 locale_t loc = uselocale(NULL);
85 static const mbstate_t initial = { 0 };
86
87 return (fnmatchl(pattern, string, string, flags, initial, initial,
88 loc));
89 return (fnmatchl(pattern, string, string, flags, initial, initial));
90 }
91
92 static int
93 fnmatchl(const char *pattern, const char *string, const char *stringstart,
94 int flags, mbstate_t patmbs, mbstate_t strmbms, locale_t loc)
95 int flags, mbstate_t patmbs, mbstate_t strmbms)
96 {
97 char *newp;
98 char c;
99 wchar_t pc, sc;
100 size_t pflen, sclen;
101
102 for (;;) {
103 pflen = mbrtowc_l(&pc, pattern, MB_LEN_MAX, &patmbs, loc);
104 pflen = mbrtowc(&pc, pattern, MB_LEN_MAX, &patmbs);
105 if (pflen == (size_t)-1 || pflen == (size_t)-2)
106 return (FNM_NOMATCH);
107 pattern += pflen;
108 sclen = mbrtowc_l(&sc, string, MB_LEN_MAX, &strmbms, loc);
109 sclen = mbrtowc(&sc, string, MB_LEN_MAX, &strmbms);
110 if (sclen == (size_t)-1 || sclen == (size_t)-2) {
111 sc = (unsigned char)*string;
112 sclen = 1;
113 (void) memset(&strmbms, 0, sizeof (strmbms));
114 }
115 switch (pc) {
116 case EOS:
117 /*
118  * Removed FNM_LEADING_DIR, as it is not present
119  * on Solaris.
120  */
121 return (sc == EOS ? 0 : FNM_NOMATCH);
122 case '?':
123 if (sc == EOS)
124 return (FNM_NOMATCH);
```

```

118     if (sc == '/' && (flags & FNM_PATHNAME))
119         return (FNM_NOMATCH);
120     if (sc == '.' && (flags & FNM_PERIOD) &&
121         (string == stringstart ||
122          ((flags & FNM_PATHNAME) && *(string - 1) == '/'))
123         return (FNM_NOMATCH);
124     string += sclen;
125     break;
126 case '*':
127     c = *pattern;
128     /* Collapse multiple stars. */
129     while (c == '*')
130         c = *++pattern;
131
132     if (sc == '.' && (flags & FNM_PERIOD) &&
133         (string == stringstart ||
134          ((flags & FNM_PATHNAME) && *(string - 1) == '/'))
135         return (FNM_NOMATCH);
136
137     /* Optimize for pattern with * at end or before /. */
138     if (c == EOS)
139         if (flags & FNM_PATHNAME)
140             return (strchr(string, '/') == NULL ?
141                 0 : FNM_NOMATCH);
142         else
143             return (0);
144     else if (c == '/' && flags & FNM_PATHNAME) {
145         if ((string = strchr(string, '/')) == NULL)
146             return (FNM_NOMATCH);
147         break;
148     }
149
150     /* General case, use recursion. */
151     while (sc != EOS) {
152         if (!fnmatch1(pattern, string, stringstart,
153                     flags, patmbs, strmbs, loc))
154             return (0);
155         sclen = mbrtowc_l(&sc, string, MB_LEN_MAX,
156                        &strmbs, loc);
157         sclen = mbrtowc(&sc, string, MB_LEN_MAX,
158                       &strmbs);
159         if (sclen == (size_t)-1 ||
160             sclen == (size_t)-2) {
161             sc = (unsigned char)*string;
162             sclen = 1;
163             (void) memset(&strmbs, 0,
164                          sizeof (strmbs));
165         }
166         if (sc == '/' && flags & FNM_PATHNAME)
167             break;
168         string += sclen;
169     }
170     return (FNM_NOMATCH);
171 case '[':
172     if (sc == EOS)
173         return (FNM_NOMATCH);
174     if (sc == '/' && (flags & FNM_PATHNAME))
175         return (FNM_NOMATCH);
176     if (sc == '.' && (flags & FNM_PERIOD) &&
177         (string == stringstart ||
178          ((flags & FNM_PATHNAME) && *(string - 1) == '/'))
179         return (FNM_NOMATCH);
180
181     switch (rangematch(pattern, sc, flags, &newp,
182                      &patmbs, loc)) {

```

```

175         &patmbs)) {
181     case RANGE_ERROR:
182         goto norm;
183     case RANGE_MATCH:
184         pattern = newp;
185         break;
186     case RANGE_NOMATCH:
187         return (FNM_NOMATCH);
188     }
189     string += sclen;
190     break;
191 case '\\':
192     if (!(flags & FNM_NOESCAPE)) {
193         pclen = mbrtowc_l(&pc, pattern, MB_LEN_MAX,
194                        &patmbs, loc);
195         pclen = mbrtowc(&pc, pattern, MB_LEN_MAX,
196                       &patmbs);
197         if (pclen == (size_t)-1 || pclen == (size_t)-2)
198             return (FNM_NOMATCH);
199         if (pclen == 0)
200             pc = '\\';
201         pattern += pclen;
202     }
203     /* FALLTHROUGH */
204 default:
205     norm:
206     if (pc == sc)
207         string += sclen;
208     else if ((flags & FNM_IGNORECASE) &&
209             (tolower_l(pc, loc) == tolower_l(sc, loc)))
210         string += sclen;
211     else
212         return (FNM_NOMATCH);
213     break;
214 }
215 }
216 /* NOTREACHED */
217 }
218
219 static int
220 rangematch(const char *pattern, wchar_t test, int flags, char **newp,
221            mbstate_t *patmbs, locale_t loc)
222 rangematch(pattern, test, flags, newp, patmbs)
223     const char *pattern;
224     wchar_t test;
225     int flags;
226     char **newp;
227     mbstate_t *patmbs;
228 {
229     int negate, ok;
230     wchar_t c, c2;
231     size_t polen;
232     const char *origpat;
233
234     /*
235      * A bracket expression starting with an unquoted circumflex
236      * character produces unspecified results (IEEE 1003.2-1992,
237      * 3.13.2). This implementation treats it like '!', for
238      * consistency with the regular expression syntax.
239      * J.T. Conklin (conklin@gai.kaleida.com)
240      */
241     if ((negate = (*pattern == '^' || *pattern == '^')) != 0)
242         ++pattern;

```

```

238     if (flags & FNM_IGNORECASE)
239         test = towlower_l(test, loc);
238         test = towlower(test);

241     /*
242     * A right bracket shall lose its special meaning and represent
243     * itself in a bracket expression if it occurs first in the list.
244     * -- POSIX.2 2.8.3.2
245     */
246     ok = 0;
247     origpat = pattern;
248     for (;;) {
249         if (*pattern == '[' && pattern > origpat) {
250             pattern++;
251             break;
252         } else if (*pattern == '\0') {
253             return (RANGE_ERROR);
254         } else if (*pattern == '/' && (flags & FNM_PATHNAME)) {
255             return (RANGE_NOMATCH);
256         } else if (*pattern == '\\') && !(flags & FNM_NOESCAPE))
257             pattern++;
258         pflen = mbrtowc_l(&c, pattern, MB_LEN_MAX, patmbs, loc);
257         pflen = mbrtowc(&c, pattern, MB_LEN_MAX, patmbs);
259         if (pflen == (size_t)-1 || pflen == (size_t)-2)
260             return (RANGE_NOMATCH);
261         pattern += pflen;

263         if (flags & FNM_IGNORECASE)
264             c = towlower_l(c, loc);
263             c = towlower(c);

266         if (*pattern == '-' && *(pattern + 1) != EOS &&
267             *(pattern + 1) != ']') {
268             if (*++pattern == '\\') && !(flags & FNM_NOESCAPE))
269                 if (*pattern != EOS)
270                     pattern++;
271             pflen = mbrtowc_l(&c2, pattern, MB_LEN_MAX, patmbs,
272                             loc);
270             pflen = mbrtowc(&c2, pattern, MB_LEN_MAX, patmbs);
273             if (pflen == (size_t)-1 || pflen == (size_t)-2)
274                 return (RANGE_NOMATCH);
275             pattern += pflen;
276             if (c2 == EOS)
277                 return (RANGE_ERROR);

279             if (flags & FNM_IGNORECASE)
280                 c2 = towlower_l(c2, loc);
278                 c2 = towlower(c2);

282             if (loc->collate->lc_is_posix ?
280             if (_collate_load_error ?
283                 c <= test && test <= c2 :
284                 _collate_range_cmp(c, test, loc) <= 0 &&
285                 _collate_range_cmp(test, c2, loc) <= 0)
282                 _collate_range_cmp(c, test) <= 0 &&
283                 _collate_range_cmp(test, c2) <= 0)
286                 ok = 1;
287             } else if (c == test)
288                 ok = 1;
289         }

291     *newp = (char *)pattern;
292     return (ok == negate ? RANGE_NOMATCH : RANGE_MATCH);
293 }

```

unchanged\_portion\_omitted

```

*****
3811 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/fputwc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 2002-2004 Tim J. Robbins.
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 /*
29  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
30  * Use is subject to license terms.
31 */

33 #include "lint.h"
34 #include "file64.h"
35 #include "mtlib.h"
36 #include "mse_int.h"
37 #include <errno.h>
38 #include <limits.h>
39 #include <stdio.h>
40 #include <stdlib.h>
41 #include <wchar.h>
42 #include <synch.h>
43 #include "mblocal.h"
44 #include "stdiom.h"
45 #include "mse.h"

47 #pragma weak _putwc = putwc

49 /*
50  * FreeBSD had both a MT safe and non-MT safe version. For whatever reason,
51  * we don't need the non-MT safe version. We do this because its faster,
52  * since we don't have to lock the file while doing the potentially expensive
53  * conversion from wide to mb.
54  *
55  * Solaris also has XPG5 and legacy semantics. The new standard requires
56  * that the stream orientation change, but legacy calls don't do that.
57  *
58  * Note that we had the source for the XPG5 version of this, but it relied

```

```

59  * on closed implementation bits that we lack, so we supply replacements
60  * here.
61  */
62 static wint_t
63 __fputwc_impl(wchar_t wc, FILE *fp, int orient)
64 {
65     char buf[MB_LEN_MAX];
66     size_t i, len;
67     rmutex_t *mx;

69     /* If we are given WEOF, then we have to stop */
70     if (wc == WEOF)
71         return (WEOF);

73     if (MB_CUR_MAX == 1 && wc > 0 && wc <= UCHAR_MAX) {
74         /*
75          * Assume single-byte locale with no special encoding.
76          * A more careful test would be to check
77          * _CurrentRuneLocale->encoding.
78          */
79         *buf = (unsigned char)wc;
80         len = 1;
81     } else {
82         /*
83          * FreeBSD used restartable wctomb. I think we can use
84          * the simpler wctomb form here. We should have a complete
85          * decode.
86          */
87         if ((len = wctomb(buf, wc)) == (size_t)-1) {
88             fp->_flag |= _IOERR;
89             errno = EILSEQ;
90             return (WEOF);
91         }

92         FLOCKFILE(mx, fp);
93         /*
94          * This is used for XPG 5 semantics, which requires the stream
95          * orientation to be changed when the function is called.
96          */
97         if (orient && GET_NO_MODE(fp)) {
98             _setorientation(fp, _WC_MODE);
99         }
100        for (i = 0; i < len; i++) {
101            if (PUTC((unsigned char)buf[i], fp) == EOF) {
102                FUNLOCKFILE(mx);
103                return (WEOF);
104            }
105        }
106        FUNLOCKFILE(mx);
107        return ((wint_t)wc);
108    }
}

unchanged_portion_omitted_

```

```

*****
6217 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/gb18030.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 /*
30  * PRC National Standard GB 18030-2000 encoding of Chinese text.
31  *
32  * See gb18030(5) for details.
33  */

35 #include "lint.h"
36 #include <sys/types.h>
37 #include <errno.h>
38 #include "runetype.h"
39 #include <stdlib.h>
40 #include <string.h>
41 #include <wchar.h>
42 #include "mblocal.h"

45 static size_t _GB18030_mbrtowc(wchar_t *_RESTRICT_KYWD,
46                               const char *_RESTRICT_KYWD,
47                               size_t, mbstate_t *_RESTRICT_KYWD);
48 static int _GB18030_mbsinit(const mbstate_t *);
49 static size_t _GB18030_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
50                               mbstate_t *_RESTRICT_KYWD);
51 static size_t _GB18030_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
52                                   const char **_RESTRICT_KYWD, size_t, size_t,
53                                   mbstate_t *_RESTRICT_KYWD);
54 static size_t _GB18030_wcsnrtombs(char *_RESTRICT_KYWD,
55                                   const wchar_t **_RESTRICT_KYWD, size_t, size_t,
56                                   mbstate_t *_RESTRICT_KYWD);

```

```

59 typedef struct {
60     int count;
61     uchar_t bytes[4];
62 } _GB18030State;

64 void
65 _GB18030_init(struct lc_ctype *lct)
66 {
67     lct->lc_mbrtowc = _GB18030_mbrtowc;
68     lct->lc_wcrtomb = _GB18030_wcrtomb;
69     lct->lc_mbsinit = _GB18030_mbsinit;
70     lct->lc_mbsnrtowcs = _GB18030_mbsnrtowcs;
71     lct->lc_wcsnrtombs = _GB18030_wcsnrtombs;
72     lct->lc_max_mblen = 4;
73     lct->lc_is_ascii = 0;
74     __mbrtowc = _GB18030_mbrtowc;
75     __wcrtomb = _GB18030_wcrtomb;
76     __mbsinit = _GB18030_mbsinit;
77     __CurrentRuneLocale = rl;
78     __ctype[520] = 4;
79     charset_is_ascii = 0;

80     return (0);
81 }

82 #ifndef _unchanged_portion_omitted_
83 static size_t
84 _GB18030_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc,
85                 mbstate_t *_RESTRICT_KYWD ps)
86 {
87     _GB18030State *gs;
88     size_t len;
89     int c;

90     gs = (_GB18030State *)ps;

91     if (gs->count != 0) {
92         errno = EINVAL;
93         return ((size_t)-1);
94     }

95     if (s == NULL)
96         /* Reset to initial shift state (no-op) */
97         return (1);
98     if ((wc & ~0x7fffffff) != 0)
99         goto ilseq;
100     if (wc & 0x7f000000) {
101         /* Replace high bit that mbrtowc() removed. */
102         wc |= 0x80000000;
103         c = (wc >> 24) & 0xff;
104         if (c < 0x81 || c > 0xfe)
105             goto ilseq;
106         *s++ = c;
107         c = (wc >> 16) & 0xff;
108         if (c < 0x30 || c > 0x39)
109             goto ilseq;
110         *s++ = c;
111         c = (wc >> 8) & 0xff;
112         if (c < 0x81 || c > 0xfe)
113             goto ilseq;
114         *s++ = c;
115         c = wc & 0xff;

```

```
204         if (c < 0x30 || c > 0x39)
205             goto ilseq;
206         *s++ = c;
207         len = 4;
208     } else if (wc & 0x00ff0000)
209         goto ilseq;
210     else if (wc & 0x0000ff00) {
211         c = (wc >> 8) & 0xff;
212         if (c < 0x81 || c > 0xfe)
213             goto ilseq;
214         *s++ = c;
215         c = wc & 0xff;
216         if (c < 0x40 || c == 0x7f || c == 0xff)
217             goto ilseq;
218         *s++ = c;
219         len = 2;
220     } else if (wc <= 0x7f) {
221         *s++ = wc;
222         len = 1;
223     } else
224         goto ilseq;
226     return (len);
227 ilseq:
228     errno = EILSEQ;
229     return ((size_t)-1);
230 }
232 static size_t
233 _GB18030_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst,
234                    const char **_RESTRICT_KYWD src, size_t nms, size_t len,
235                    mbstate_t *_RESTRICT_KYWD ps)
236 {
237     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _GB18030_mbrtowc));
238 }
240 static size_t
241 _GB18030_wcsnrtombs(char *_RESTRICT_KYWD dst,
242                    const wchar_t **_RESTRICT_KYWD src, size_t nwc, size_t len,
243                    mbstate_t *_RESTRICT_KYWD ps)
244 {
245     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _GB18030_wcrtomb));
246 }
247 _____
248 unchanged_portion_omitted
```

```

*****
4907 Fri Jul 11 11:42:56 2014
new/usr/src/lib/libc/port/locale/gb2312.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2004 Tim J. Robbins. All rights reserved.
5  * Copyright (c) 2003 David Xu <davidxu@freebsd.org>
6  * All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include <sys/types.h>
32 #include <errno.h>
33 #include "runetype.h"
34 #include <stdlib.h>
35 #include <string.h>
36 #include <wchar.h>
37 #include "mblocal.h"
38 #include "lctype.h"

40 static size_t  _GB2312_mbrtowc(wchar_t *_RESTRICT_KYWD,
41                             const char *_RESTRICT_KYWD,
42                             size_t, mbstate_t *_RESTRICT_KYWD);
43 static int     _GB2312_mbsinit(const mbstate_t *);
44 static size_t  _GB2312_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
45                             mbstate_t *_RESTRICT_KYWD);
46 static size_t  _GB2312_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
47                                 const char **_RESTRICT_KYWD, size_t, size_t,
48                                 mbstate_t *_RESTRICT_KYWD);
49 static size_t  _GB2312_wcsnrtombs(char *_RESTRICT_KYWD,
50                                 const wchar_t **_RESTRICT_KYWD, size_t, size_t,
51                                 mbstate_t *_RESTRICT_KYWD);

54 typedef struct {
55     int     count;
56     uchar_t bytes[2];
57 } _GB2312State;

```

```

59 void
60 _GB2312_init(struct lc_ctype *lct)
61 int
62 _GB2312_init(_RuneLocale *rl)
63 {
64     lct->lc_mbrtowc = _GB2312_mbrtowc;
65     lct->lc_wcrtomb = _GB2312_wcrtomb;
66     lct->lc_mbsinit = _GB2312_mbsinit;
67     lct->lc_mbsnrtowcs = _GB2312_mbsnrtowcs;
68     lct->lc_wcsnrtombs = _GB2312_wcsnrtombs;
69     lct->lc_max_mblen = 2;
70     lct->lc_is_ascii = 0;
71     _CurrentRuneLocale = rl;
72     _mbrtowc = _GB2312_mbrtowc;
73     _wcrtomb = _GB2312_wcrtomb;
74     _mbsinit = _GB2312_mbsinit;
75     _ctype[520] = 2;
76     charset_is_ascii = 0;
77     return (0);
78 }
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182

```

unchanged portion omitted

```

143 static size_t
144 _GB2312_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc,
145                mbstate_t *_RESTRICT_KYWD ps)
146 {
147     _GB2312State *gs;
148
149     gs = (_GB2312State *)ps;
150
151     if (gs->count != 0) {
152         errno = EINVAL;
153         return ((size_t)-1);
154     }
155
156     if (s == NULL)
157         /* Reset to initial shift state (no-op) */
158         return (1);
159     if (wc & 0x8000) {
160         *s++ = (wc >> 8) & 0xff;
161         *s = wc & 0xff;
162         return (2);
163     }
164     *s = wc & 0xff;
165     return (1);
166 }

168 static size_t
169 _GB2312_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst,
170                   const char **_RESTRICT_KYWD src, size_t nms, size_t len,
171                   mbstate_t *_RESTRICT_KYWD ps)
172 {
173     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _GB2312_mbrtowc));
174 }

176 static size_t
177 _GB2312_wcsnrtombs(char *_RESTRICT_KYWD dst,
178                   const wchar_t **_RESTRICT_KYWD src, size_t nwc, size_t len,
179                   mbstate_t *_RESTRICT_KYWD ps)
180 {
181     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _GB2312_wcrtomb));
182 }

```

unchanged portion omitted



```

*****
4945 Fri Jul 11 11:42:57 2014
new/usr/src/lib/libc/port/locale/gbk.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins. All rights reserved.
5  * Copyright (c) 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * This code is derived from software contributed to Berkeley by
9  * Paul Borman at Krystal Technologies.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 * notice, this list of conditions and the following disclaimer in the
18 * documentation and/or other materials provided with the distribution.
19 * 4. Neither the name of the University nor the names of its contributors
20 * may be used to endorse or promote products derived from this software
21 * without specific prior written permission.
22 *
23 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
24 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
25 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
26 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
27 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
28 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
29 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
30 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
31 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
32 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
33 * SUCH DAMAGE.
34 */
35
36 #include "lint.h"
37 #include <sys/types.h>
38 #include <errno.h>
39 #include "runetype.h"
40 #include <stdlib.h>
41 #include <string.h>
42 #include <wchar.h>
43 #include "mblocal.h"
44
45 static size_t  _GBK_mbrtowc(wchar_t *_RESTRICT_KYWD,
46                          const char *_RESTRICT_KYWD,
47                          size_t, mbstate_t *_RESTRICT_KYWD);
48 static int     _GBK_mbsinit(const mbstate_t *);
49 static size_t  _GBK_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
50                          mbstate_t *_RESTRICT_KYWD);
51 static size_t  _GBK_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
52                              const char **_RESTRICT_KYWD, size_t, size_t,
53                              mbstate_t *_RESTRICT_KYWD);
54 static size_t  _GBK_wcsnrtombs(char *_RESTRICT_KYWD,
55                              const wchar_t **_RESTRICT_KYWD, size_t, size_t,
56                              mbstate_t *_RESTRICT_KYWD);

```

```

58 typedef struct {
59     wchar_t ch;
60 } _GBKState;
61
62 void
63 _GBK_init(struct lc_ctype *lct)
64 {
65     _GBK_init(_RuneLocale *rl)
66     {
67         lct->lc_mbrtowc = _GBK_mbrtowc;
68         lct->lc_wcrtomb = _GBK_wcrtomb;
69         lct->lc_mbsinit = _GBK_mbsinit;
70         lct->lc_mbsnrtowcs = _GBK_mbsnrtowcs;
71         lct->lc_wcsnrtombs = _GBK_wcsnrtombs;
72         lct->lc_max_mblen = 2;
73         lct->lc_is_ascii = 0;
74
75         __mbrtowc = _GBK_mbrtowc;
76         __wcrtomb = _GBK_wcrtomb;
77         __mbsinit = _GBK_mbsinit;
78         __CurrentRuneLocale = rl;
79         __ctype[520] = 2;
80         charset_is_ascii = 0;
81         return (0);
82     }
83 }
84
85 unchanged_portion_omitted
86
150 static size_t
151 _GBK_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc, mbstate_t *_RESTRICT_KYWD ps)
152 {
153     _GBKState *gs;
154
155     gs = (_GBKState *)ps;
156
157     if (gs->ch != 0) {
158         errno = EINVAL;
159         return ((size_t)-1);
160     }
161
162     if (s == NULL)
163         /* Reset to initial shift state (no-op) */
164         return (1);
165     if (wc & 0x8000) {
166         *s++ = (wc >> 8) & 0xff;
167         *s = wc & 0xff;
168         return (2);
169     }
170     *s = wc & 0xff;
171     return (1);
172 }
173
174 static size_t
175 _GBK_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
176               size_t nms, size_t len, mbstate_t *_RESTRICT_KYWD ps)
177 {
178     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _GBK_mbrtowc));
179 }
180
181 static size_t
182 _GBK_wcsnrtombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
183               size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps)
184 {
185     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _GBK_wcrtomb));
186 }
187
188 unchanged_portion_omitted

```

new/usr/src/lib/libc/port/locale/isdigit.c

1

```
*****
2351 Fri Jul 11 11:42:57 2014
new/usr/src/lib/libc/port/locale/isdigit.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */

17 /*
18  * This file contains the implementation of various functional forms
19  * of the ctype tests, specifically the required by ISO C. These are defined
20  * in the "C" (POSIX) locale.
21 */

23 #include "lint.h"
24 #include <ctype.h>
25 #include <locale.h>
26 #include "localeimpl.h"
27 #include "_ctype.h"
28 #include "lctype.h"

30 /*
31  * We are supplying functional forms, so make sure to suppress any macros
32  * we might have imported.
33 */

35 /*
36  * Performance note: ASCII test is *much* faster, as we can avoid expensive
37  * function call overhead. This is the hot case, so we try to do that
38  * whenever possible. As far as we know, *every* encoding we support
39  * is a strict superset of ASCII. So we can make things faster by trying
40  * ASCII first, and only then falling to locale specific checks.
41 */

43 static int
44 isctype_l(int c, int mask, locale_t loc)
45 {
46     return ((unsigned)c > 255 ? 0 : (loc->ctype->lc_ctype_mask[c] & mask));
47 }

49 #define ISTYPE_L(c, mask, loc) \
50     (isascii(c) ? (__ctype_mask[c] & (mask)) : isctype_l(c, mask, loc))

52 #define ISTYPE(c, mask) ISTYPE_L(c, mask, uselocale(NULL))

54 #define DEFN_ISTYPE(type, mask) \
55 int \
56 is##type##_l(int c, locale_t l) \
57 { \
58     return (ISTYPE_L(c, mask, l)); \
```

new/usr/src/lib/libc/port/locale/isdigit.c

2

```
59 } \
60 \
61 int \
62 is##type(int c) \
63 { \
64     return (ISTYPE(c, mask)); \
65 }

67 #undef isblank
68 #undef isupper
69 #undef islower
70 #undef isdigit
71 #undef isxdigit
72 #undef isalpha
73 #undef isalnum
74 #undef isspace
75 #undef iscntrl
76 #undef isgraph
77 #undef ispunct
78 #undef isprint

80 DEFN_ISTYPE(blank, _ISBLANK)
81 DEFN_ISTYPE(upper, _ISUPPER)
82 DEFN_ISTYPE(lower, _ISLOWER)
83 DEFN_ISTYPE(digit, _ISDIGIT)
84 DEFN_ISTYPE(xdigit, _ISXDIGIT)
85 DEFN_ISTYPE(alpha, _ISALPHA)
86 DEFN_ISTYPE(alnum, _ISALNUM)
87 DEFN_ISTYPE(space, _ISSPACE)
88 DEFN_ISTYPE(cntrl, _ISCNTRL)
89 DEFN_ISTYPE(graph, _ISGRAPH)
90 DEFN_ISTYPE(punct, _ISPUNCT)
91 DEFN_ISTYPE(print, _ISPRINT)
```

```

*****
6314 Fri Jul 11 11:42:57 2014
new/usr/src/lib/libc/port/locale/iswctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1989, 1993
5  * The Regents of the University of California. All rights reserved.
6  * (c) UNIX System Laboratories, Inc.
7  * All or some portions of this file are derived from material licensed
8  * to the University of California by American Telephone and Telegraph
9  * Co. or Unix System Laboratories, Inc. and are reproduced herein with
10 * the permission of UNIX System Laboratories, Inc.
11 *
12 * This code is derived from software contributed to Berkeley by
13 * Paul Borman at Krystal Technologies.
14 *
15 * Redistribution and use in source and binary forms, with or without
16 * modification, are permitted provided that the following conditions
17 * are met:
18 * 1. Redistributions of source code must retain the above copyright
19 * notice, this list of conditions and the following disclaimer.
20 * 2. Redistributions in binary form must reproduce the above copyright
21 * notice, this list of conditions and the following disclaimer in the
22 * documentation and/or other materials provided with the distribution.
23 * 4. Neither the name of the University nor the names of its contributors
24 * may be used to endorse or promote products derived from this software
25 * without specific prior written permission.
26 *
27 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
28 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
29 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
30 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
31 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
32 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
33 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
34 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
35 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
36 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
37 * SUCH DAMAGE.
38 */

40 #include "lint.h"
41 #include <wctype.h>
42 #include <locale.h>
43 #include "runefile.h"
44 #include "runetype.h"
45 #include "localeimpl.h"
46 #include "_ctype.h"

48 /*
49  * Note that the standard requires iswascii to be a macro, so it is defined
50  * in our headers.
51  * We removed: iswascii, iswhexnumber, and iswnumber, as
52  * these are not present on Solaris. Note that the standard requires
53  * iswascii to be a macro, so it is defined in our headers.
54  * We aliased (per Solaris) iswideogram, iswspecial, iswspecial to the
55  * equivalent values without "w". The Solaris specific function isenglish()
56  * is here, but does not get an isw* equivalent.
57  */

```

```

56  * Note that various code assumes that "numbers" (iswdigit, iswxdigit)
57  * only return true for characters in the portable set. While the assumption
58  * is not technically correct, it turns out that for all of our locales this
59  * is true. iswhexnumber is aliased to iswxdigit.
60  * We renamed (per Solaris) iswideogram, iswspecial, iswspecial to the
61  * equivalent values without "w". We added a new isnumber, that looks
62  * for non-ASCII numbers.
63  */

62 static int
63 __istype_l(locale_t loc, wint_t c, unsigned int f)
64 {
65     unsigned int rt;

66     /* Fast path for single byte locales */
67     if (c < 0 || c >= _CACHED_RUNES)
68         rt = __runetype(loc->runelocale, c);
69     else
70         rt = loc->runelocale->__runetype[c];
71     return (rt & f);
72 }

74 static int
75 __istype(wint_t c, unsigned int f)
76 {
77     return (__istype_l(uselocale(NULL), c, f));
78 }

79 unsigned int rt;

80 int
81 iswctype_l(wint_t wc, wctype_t class, locale_t loc)
82 {
83     if (iswascii(wc))
84         return (__ctype_mask[wc] & class);
85     return (__istype_l(loc, wc, class));
86     /* Fast path for single byte locales */
87     if (c < 0 || c >= _CACHED_RUNES)
88         return (0);
89     else
90         rt = _CurrentRuneLocale->__runetype[c];
91     return (rt & f);
92 }

88 #undef iswctype
89 int
90 iswctype(wint_t wc, wctype_t class)
91 {
92     /*
93      * Note that we don't just call iswctype_l because we optimize for
94      * the iswascii() case, so that most of the time we have no need to
95      * call uselocale().
96      */
97     if (iswascii(wc))
98         return (__ctype_mask[wc] & class);
99     return (__istype(wc, class));
100 }

102 /*
103  * This is a legacy version, baked into binaries.
104  */
105 #undef _iswctype
106 unsigned

```

```

107 iswctype(wchar_t wc, int class)
108 {
109     if (iswascii(wc))
110         return (__ctype_mask[wc] & class);
111     return (__istype((wint_t)wc, (unsigned int)class));
112 }

114 #define DEFN_ISWTYPE(type, mask) \
115 int \
116 isw##type##_l(wint_t wc, locale_t loc) \
117 { \
118     return (iswascii(wc) ? \
119         (__ctype_mask[wc] & (mask)) : \
120         __istype_l(loc, wc, mask)); \
121 } \
122 \
123 int \
124 isw##type(wint_t wc) \
125 { \
126     return (iswascii(wc) ? \
127         (__ctype_mask[wc] & (mask)) : \
128         __istype(wc, mask)); \
129 }

95 #undef iswalnum
96 int
97 iswalnum(wint_t wc)
98 {
99     return (__istype(wc, _CTYPE_A|_CTYPE_D));
129 }

131 /* kill off any macros */
132 #undef iswalnum
133 #undef iswalpha
103 int
104 iswalpha(wint_t wc)
105 {
106     return (__istype(wc, _CTYPE_A));
107 }

134 #undef iswblank
110 int
111 iswblank(wint_t wc)
112 {
113     return (__istype(wc, _CTYPE_B));
114 }

136 DEFN_ISWTYPE(alnum, _CTYPE_A|_CTYPE_D)
137 DEFN_ISWTYPE(alpha, _CTYPE_A)
138 DEFN_ISWTYPE(blank, _CTYPE_B)
139 DEFN_ISWTYPE(cntrl, _CTYPE_C)
140 DEFN_ISWTYPE(digit, _CTYPE_D)
141 DEFN_ISWTYPE(graph, _CTYPE_D)
142 DEFN_ISWTYPE(lower, _CTYPE_L)
143 DEFN_ISWTYPE(upper, _CTYPE_U)
144 DEFN_ISWTYPE(print, _CTYPE_R)
145 DEFN_ISWTYPE(punct, _CTYPE_P)
146 DEFN_ISWTYPE(space, _CTYPE_S)
147 DEFN_ISWTYPE(xdigit, _CTYPE_X)
148 DEFN_ISWTYPE(ideogram, _CTYPE_I)
149 DEFN_ISWTYPE(phonogram, _CTYPE_Q)
150 DEFN_ISWTYPE(special, _CTYPE_T)
151 DEFN_ISWTYPE(number, _CTYPE_N)
116 #undef iswcntrl
117 int
118 iswcntrl(wint_t wc)
119 {
120     return (__istype(wc, _CTYPE_C));

```

```

121 }

123 #undef iswdigit
124 int
125 iswdigit(wint_t wc)
126 {
127     return (__istype(wc, _CTYPE_D));
128 }

154 #undef iswhexnumber
155 #pragma weak iswhexnumber = iswxdigit
156 #pragma weak iswhexnumber_l = iswxdigit_l
130 #undef iswgraph
131 int
132 iswgraph(wint_t wc)
133 {
134     return (__istype(wc, _CTYPE_G));
135 }

158 #undef iswideogram
159 #pragma weak iswideogram = iswideogram
138 int
139 iswideogram(wint_t wc)
140 {
141     return (__istype(wc, _CTYPE_I));
142 }

144 #undef iswlower
145 int
146 iswlower(wint_t wc)
147 {
148     return (__istype(wc, _CTYPE_L));
149 }

161 #undef isphonogram
162 #pragma weak isphonogram = iswphonogram
152 int
153 isphonogram(wint_t wc)
154 {
155     return (__istype(wc, _CTYPE_Q));
156 }

158 #undef iswprint
159 int
160 iswprint(wint_t wc)
161 {
162     return (__istype(wc, _CTYPE_R));
163 }

165 #undef iswpunct
166 int
167 iswpunct(wint_t wc)
168 {
169     return (__istype(wc, _CTYPE_P));
170 }

172 #undef iswspace
173 int
174 iswspace(wint_t wc)
175 {
176     return (__istype(wc, _CTYPE_S));
177 }

179 #undef iswupper
180 int
181 iswupper(wint_t wc)

```

```

182 {
183     return (__istype(wc, _CTYPE_U));
184 }

186 #undef iswxdigit
187 int
188 iswxdigit(wint_t wc)
189 {
190     return (__isctype(wc, _CTYPE_X));
191 }

193 #undef isenglish
194 int
195 isenglish(wint_t wc)
196 {
197     return (__istype(wc, _CTYPE_E));
198 }

164 #undef isspecial
165 #pragma weak isspecial = iswspecial
201 int
202 isspecial(wint_t wc)
203 {
204     return (__istype(wc, _CTYPE_T));
205 }

167 #undef isnumber
168 #pragma weak isnumber = iswnumber
208 int
209 isnumber(wint_t wc)
210 {
211     return (__istype(wc, _CTYPE_N));
212 }

170 /*
171  * FreeBSD has iswrunes() for use by external programs, and this is used by
172  * the "tr" program. As that program is part of our consolidation, we
173  * provide an _ILLUMOS_PRIVATE version of this function that we can use.
174  *
175  * No programs that are not part of the illumos stack itself should use
176  * this function -- programs that do reference will not be portable to
177  * other versions of SunOS or Solaris.
178  */
179 int
180 __iswrunes(wint_t wc)
181 {
182     /*
183      * Note, FreeBSD ignored the low order byte, as they encode their
184      * ctype values differently. We can't do that (ctype is baked into
185      * applications), but instead can just check if *any* bit is set in
186      * the ctype. Any bit being set indicates its a valid rune.
187      *
188      * NB: For ASCII all positions except NULL are runes.
189      */
190     return (wc == 0 ? 0 : iswascii(wc) ? 1 : __istype(wc, 0xffffffffU));
191 }

193 /*
194  * isenglish is a Solaris legacy. No isw* equivalent. Note that this most
195  * likely doesn't work, as the locale data we have doesn't include it. It
196  * specifically is only valid for non-ASCII characters. We're not sure this
197  * is in actual use in the wild.
198  */
199 #undef isenglish
200 int
201 isenglish(wint_t wc)

```

```

202 {
203     return (__istype(wc, _CTYPE_E));
204     return (__istype(wc, 0xffffffffU));
205 }

```

---

unchanged\_portion\_omitted\_

new/usr/src/lib/libc/port/locale/lctype.h

1

```
*****
1891 Fri Jul 11 11:42:57 2014
new/usr/src/lib/libc/port/locale/lctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 2013 Garrett D'Amore <garrett@damore.org>
14 */
16 #ifndef _LCTYPE_H_
17 #define _LCTYPE_H_
19 #include <wchar.h>
21 /* private LC_CTYPE related structures */
23 /* encoding callbacks */
24 struct lc_ctype {
26     size_t (*lc_mbrtowc)(wchar_t *_RESTRICT_KYWD,
27                         const char *_RESTRICT_KYWD, size_t, mbstate_t *_RESTRICT_KYWD);
29     int (*lc_mbsinit)(const mbstate_t *);
31     size_t (*lc_mbsnrtowcs)(wchar_t *_RESTRICT_KYWD,
32                            const char **_RESTRICT_KYWD, size_t, size_t,
33                            mbstate_t *_RESTRICT_KYWD);
35     size_t (*lc_wrtomb)(char *_RESTRICT_KYWD, wchar_t,
36                       mbstate_t *_RESTRICT_KYWD);
38     size_t (*lc_wcsnrtombs)(char *_RESTRICT_KYWD,
39                            const wchar_t **_RESTRICT_KYWD, size_t, size_t,
40                            mbstate_t *_RESTRICT_KYWD);
42     unsigned char lc_is_ascii;
43     unsigned char lc_max_mblen;
45     const int *lc_trans_upper;
46     const int *lc_trans_lower;
47     const unsigned *lc_ctype_mask;
48 };
50 /*
51  * Default implementation (C locale, i.e. ASCII).
52 */
53 size_t __mbrtowc_ascii(wchar_t *_RESTRICT_KYWD,
54                       const char *_RESTRICT_KYWD, size_t, mbstate_t *_RESTRICT_KYWD);
55 int __mbsinit_ascii(const mbstate_t *);
56 size_t __mbsnrtowcs_ascii(wchar_t *_RESTRICT_KYWD dst,
57                           const char **_RESTRICT_KYWD src, size_t nms, size_t len,
58                           mbstate_t *_RESTRICT_KYWD);
```

new/usr/src/lib/libc/port/locale/lctype.h

2

```
59 size_t __wrtomb_ascii(char *_RESTRICT_KYWD, wchar_t,
60                      mbstate_t *_RESTRICT_KYWD);
61 size_t __wcsnrtombs_ascii(char *_RESTRICT_KYWD,
62                           const wchar_t **_RESTRICT_KYWD,
63                           size_t, size_t, mbstate_t *_RESTRICT_KYWD);
66 #endif /* !_LCTYPE_H_ */
```

new/usr/src/lib/libc/port/locale/ldpart.c

1

\*\*\*\*\*

3785 Fri Jul 11 11:42:57 2014

new/usr/src/lib/libc/port/locale/ldpart.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */
29 #include "lint.h"
30 #include "file64.h"
31 #include <sys/types.h>
32 #include <sys/stat.h>
33 #include <errno.h>
34 #include <fcntl.h>
35 #include <limits.h>
36 #include <stdlib.h>
37 #include <string.h>
38 #include <unistd.h>
39 #include <stdio.h>
41 #include "libc.h"
42 #include "ldpart.h"
43 #include "setlocale.h"
45 static int split_lines(char *, const char *);
47 int
48 _part_load_locale(const char *name,
49 _part_load_locale(const char *name, int *using_locale,
50 char **locale_buf, const char *category_filename,
51 int locale_buf_size_max, int locale_buf_size_min,
52 const char **dst_localebuf)
53 {
54     int saverr, fd, i, num_lines;
55     char *lbuf, *p;
56     const char *plim;
57     char filename[PATH_MAX];
58     struct stat st;
```

new/usr/src/lib/libc/port/locale/ldpart.c

2

```
58     size_t namesize, bufsize;
59
60     /* 'name' must be already checked. */
61     if (strcmp(name, "C") == 0 || strcmp(name, "POSIX") == 0) {
62         *using_locale = 0;
63         return (_LDP_CACHE);
64     }
65
66     /*
67      * If the locale name is the same as our cache, use the cache.
68      */
69     if (*locale_buf != NULL && strcmp(name, *locale_buf) == 0) {
70         *using_locale = 1;
71         return (_LDP_CACHE);
72     }
73
74     /* Slurp the locale file into the cache.
75      */
76     namesize = strlen(name) + 1;
77
78     /* 'PathLocale' must be already set & checked. */
79
80     (void) snprintf(filename, sizeof (filename), "%s/%s/%s/LCL_DATA",
81                     _PathLocale, name, category_filename);
82
83     if ((fd = open(filename, O_RDONLY)) < 0)
84         return (_LDP_ERROR);
85     if (fstat(fd, &st) != 0)
86         goto bad_locale;
87     if (st.st_size <= 0) {
88         errno = EINVAL;
89         goto bad_locale;
90     }
91     bufsize = namesize + st.st_size;
92     if ((lbuf = libc_malloc(bufsize)) == NULL) {
93         if ((lbuf = malloc(bufsize)) == NULL) {
94             errno = ENOMEM;
95             goto bad_locale;
96         }
97     }
98     (void) strcpy(lbuf, name);
99     p = lbuf + namesize;
100     plim = p + st.st_size;
101     if (read(fd, p, (size_t)st.st_size) != st.st_size)
102         goto bad_lbuf;
103
104     /*
105      * Parse the locale file into localebuf.
106      */
107     if (plim[-1] != '\n') {
108         errno = EINVAL;
109         goto bad_lbuf;
110     }
111     num_lines = split_lines(p, plim);
112     if (num_lines >= locale_buf_size_max)
113         num_lines = locale_buf_size_max;
114     else if (num_lines >= locale_buf_size_min)
115         num_lines = locale_buf_size_min;
116     else {
117         errno = EINVAL;
118         goto bad_lbuf;
119     }
120     (void) close(fd);
121     /*
122      * Record the successful parse in the cache.
123      */
124     if (*locale_buf != NULL)
```

```
121         free(*locale_buf);
108     *locale_buf = lbuf;
109     for (p = *locale_buf, i = 0; i < num_lines; i++)
110         dst_localebuf[i] = (p += strlen(p) + 1);
111     for (i = num_lines; i < locale_buf_size_max; i++)
112         dst_localebuf[i] = NULL;
127     *using_locale = 1;
```

```
114     return (_LDP_LOADED);
```

```
116 bad_lbuf:
117     saverr = errno;
118     free(lbuf);
119     errno = saverr;
120 bad_locale:
121     saverr = errno;
122     (void) close(fd);
123     errno = saverr;
```

```
125     return (_LDP_ERROR);
126 }
```

\_\_\_\_\_unchanged\_portion\_omitted\_\_\_\_\_



new/usr/src/lib/libc/port/locale/ldpart.h

1

\*\*\*\*\*

1665 Fri Jul 11 11:42:57 2014

new/usr/src/lib/libc/port/locale/ldpart.h

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3 * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
4 * All rights reserved.
5 *
6 * Redistribution and use in source and binary forms, with or without
7 * modification, are permitted provided that the following conditions
8 * are met:
9 * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #ifndef _LDPART_H_
29 #define _LDPART_H_

31 #define _LDP_LOADED 0
32 #define _LDP_ERROR (-1)
33 #define _LDP_CACHE 1

35 int __part_load_locale(const char *, char **, const char *,
34 int __part_load_locale(const char *, int *, char **, const char *,
36 int, int, const char **);

38 #endif /* !_LDPART_H_ */
```

```

*****
2638 Fri Jul 11 11:42:57 2014
new/usr/src/lib/libc/port/locale/lmessages.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <stddef.h>
31 #include <errno.h>
32 #include "ldpart.h"
33 #include "lmessages.h"
34 #include "localeimpl.h"

36 #define LCMESSAGES_SIZE_FULL (sizeof (struct lc_messages) / sizeof (char *))
33 #define LCMESSAGES_SIZE_FULL (sizeof (struct lc_messages_T) / sizeof (char *))
37 #define LCMESSAGES_SIZE_MIN \
38 (offsetof(struct lc_messages, yesstr) / sizeof (char *))
35 (offsetof(struct lc_messages_T, yesstr) / sizeof (char *))

40 static char empty[] = "";

42 struct lc_messages lc_messages_posix = {
39 static const struct lc_messages_T _C_messages_locale = {
43     "^[yY]", /* yesexpr */
44     "^[nN]", /* noexpr */
45     "yes", /* yesstr */
46     "no" /* nostr */
47 };

49 struct locdata __posix_messages_locdata = {
50     .l_lname = "C",
51     .l_data = { &lc_messages_posix }
52 };
46 static struct lc_messages_T _messages_locale;
47 static int _messages_using_locale;
48 static char *_messages_locale_buf;

```

```

54 struct locdata *
55 __lc_messages_load(const char *name)
56 {
57     struct locdata *ldata;
58     struct lc_messages *lmsgs;
59     int ret;

61     if ((ldata = __locdata_alloc(name, sizeof (*lmsgs))) == NULL)
62         return (NULL);
63     lmsgs = ldata->l_data[0];

65     ret = __part_load_locale(name, (char **)&ldata->l_data[1],
66     "LC_MESSAGES", LCMESSAGES_SIZE_FULL, LCMESSAGES_SIZE_MIN,
67     (const char **)lmsgs);

69     if (ret != _LDP_LOADED) {
70         __locdata_free(ldata);
71         errno = EINVAL;
72         return (NULL);
73     }
55     ret = __part_load_locale(name, &_messages_using_locale,
56     &_messages_locale_buf, "LC_MESSAGES",
57     LCMESSAGES_SIZE_FULL, LCMESSAGES_SIZE_MIN,
58     (const char **)&_messages_locale);
59     if (ret == _LDP_LOADED) {
60         if (_messages_locale.yesstr == NULL)
61             _messages_locale.yesstr = empty;
62         if (_messages_locale.nostr == NULL)
63             _messages_locale.nostr = empty;
64     }
65     return (ret);
66 }

75     if (lmsgs->yesstr == NULL)
76         lmsgs->yesstr = empty;
77     if (lmsgs->nostr == NULL)
78         lmsgs->nostr = empty;

80     return (ldata);
68 struct lc_messages_T *
69 __get_current_messages_locale(void)
70 {
71     return (_messages_using_locale ? &_messages_locale :
72     (struct lc_messages_T *)&_C_messages_locale);
81 }

```

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/lmessages.h

1

\*\*\*\*\*

1620 Fri Jul 11 11:42:57 2014

new/usr/src/lib/libc/port/locale/lmessages.h

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #ifndef _LMESSAGES_H_
29 #define _LMESSAGES_H_

31 struct lc_messages {
30 struct lc_messages_T {
32     const char *yesexpr;
33     const char *noexpr;
34     const char *yesstr;
35     const char *nostr;
36 };

37 struct lc_messages_T *__get_current_messages_locale(void);
38 int __messages_load_locale(const char *);

38 #endif /* !_LMESSAGES_H_ */
```

```

*****
5228 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/lmonetary.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stddef.h>
32 #include <stdlib.h>
33 #include <stdio.h>
34 #include <string.h>
35 #include <errno.h>
36 #include "libc.h"
37 #include "ldpart.h"
38 #include "lmonetary.h"
39 #include "localeimpl.h"

35 extern int __mlocale_changed;
41 extern const char *__fix_locale_grouping_str(const char *);

43 #define LCMONETARY_SIZE_FULL (sizeof (struct lc_monetary) / sizeof (char *))
38 #define LCMONETARY_SIZE_FULL (sizeof (struct lc_monetary_T) / sizeof (char *))
44 #define LCMONETARY_SIZE_MIN \
45     (offsetof(struct lc_monetary, int_p_cs_precedes) / sizeof (char *))
40     (offsetof(struct lc_monetary_T, int_p_cs_precedes) / sizeof (char *))

47 static char empty[] = "";
48 static char numempty[] = { CHAR_MAX, '\0' };

50 struct lc_monetary lc_monetary_posix = {
45 static const struct lc_monetary_T _C_monetary_locale = {
51     empty, /* int_curr_symbol */
52     empty, /* currency_symbol */
53     empty, /* mon_decimal_point */
54     empty, /* mon_thousands_sep */

```

```

55     numempty, /* mon_grouping */
56     empty, /* positive_sign */
57     empty, /* negative_sign */
58     numempty, /* int_frac_digits */
59     numempty, /* frac_digits */
60     numempty, /* p_cs_precedes */
61     numempty, /* p_sep_by_space */
62     numempty, /* n_cs_precedes */
63     numempty, /* n_sep_by_space */
64     numempty, /* p_sign_posn */
65     numempty, /* n_sign_posn */
66     numempty, /* int_p_cs_precedes */
67     numempty, /* int_n_cs_precedes */
68     numempty, /* int_p_sep_by_space */
69     numempty, /* int_n_sep_by_space */
70     numempty, /* int_p_sign_posn */
71     numempty, /* int_n_sign_posn */
72     empty, /* crncystr */
66     numempty, /* int_n_sign_posn */
73 };

75 struct locdata __posix_monetary_locdata = {
76     .l_lname = "C",
77     .l_data = { &lc_monetary_posix }
78 };
69 static struct lc_monetary_T _monetary_locale;
70 static int _monetary_using_locale;
71 static char *_monetary_locale_buf;

80 static char
81 cnv(const char *str)
82 {
83     int i = strtol(str, NULL, 10);

85     if (i == -1)
86         i = CHAR_MAX;
87     return ((char)i);
88 }

90 struct locdata *
91 __lc_monetary_load(const char *name)
83 int
84 __monetary_load_locale(const char *name)
92 {
93     int ret;
94     int clen;
95     struct lc_monetary *lmon;
96     struct locdata *ldata;

98     if ((ldata = __locdata_alloc(name, sizeof (*lmon))) == NULL) {
99         return (NULL);
100     }
101     lmon = ldata->l_data[0];
88     ret = __part_load_locale(name, &_monetary_using_locale,
89         &_monetary_locale_buf, "LC_MONETARY",
90         LCMONETARY_SIZE_FULL, LCMONETARY_SIZE_MIN,
91         (const char **)&_monetary_locale);
92     if (ret != _LDP_ERROR)
93         __mlocale_changed = 1;
94     if (ret == _LDP_LOADED) {
95         _monetary_locale.mon_grouping =
96             __fix_locale_grouping_str(_monetary_locale.mon_grouping);

103     ret = __part_load_locale(name, (char **)&ldata->l_data[1],
104         "LC_MONETARY", LCMONETARY_SIZE_FULL, LCMONETARY_SIZE_MIN,
105         (const char **)&lmon);

```

```

107     if (ret != _LDP_LOADED) {
108         __locdata_free(ldata);
109         errno = EINVAL;
110         return (NULL);
111     }
112
113     /* special storage for currency string */
114     clen = strlen(lmon->currency_symbol) + 2;
115     ldata->l_data[2] = libc_malloc(clen);
116     lmon->crncystr = ldata->l_data[2];
117
118     lmon->mon_grouping = __fix_locale_grouping_str(lmon->mon_grouping);
119
120 #define M_ASSIGN_CHAR(NAME) \
121     (((char *)lmon->NAME)[0] = cnv(lmon->NAME))
122     (((char *)_monetary_locale.NAME)[0] = \
123         cnv(_monetary_locale.NAME))
124
125 M_ASSIGN_CHAR(int_frac_digits);
126 M_ASSIGN_CHAR(frac_digits);
127 M_ASSIGN_CHAR(p_cs_precedes);
128 M_ASSIGN_CHAR(p_sep_by_space);
129 M_ASSIGN_CHAR(n_cs_precedes);
130 M_ASSIGN_CHAR(n_sep_by_space);
131 M_ASSIGN_CHAR(p_sign_posn);
132 M_ASSIGN_CHAR(n_sign_posn);
133
134 /*
135  * The six additional C99 international monetary formatting
136  * parameters default to the national parameters when
137  * reading FreeBSD LC_MONETARY data files.
138  */
139 #define M_ASSIGN_ICHAR(NAME) \
140     if (lmon->int_##NAME == NULL) \
141         lmon->int_##NAME = lmon->NAME; \
142     if (_monetary_locale.int_##NAME == NULL) \
143         _monetary_locale.int_##NAME = \
144         _monetary_locale.NAME; \
145     else \
146         M_ASSIGN_CHAR(int_##NAME);
147
148 M_ASSIGN_ICHAR(p_cs_precedes);
149 M_ASSIGN_ICHAR(n_cs_precedes);
150 M_ASSIGN_ICHAR(p_sep_by_space);
151 M_ASSIGN_ICHAR(n_sep_by_space);
152 M_ASSIGN_ICHAR(p_sign_posn);
153 M_ASSIGN_ICHAR(n_sign_posn);
154
155 /*
156  * Now calculate the currency string (CRNCYSTR) for nl_langinfo.
157  * This is a legacy SUSv2 interface.
158  */
159 if ((lmon->p_cs_precedes[0] == lmon->n_cs_precedes[0]) &&
160     (lmon->currency_symbol[0] != '\0')) {
161     char sign = '\0';
162     switch (lmon->p_cs_precedes[0]) {
163     case 0:
164         sign = '-';
165         break;
166     case 1:
167         sign = '+';
168         break;
169     case CHAR_MAX:
170         /*
171          * Substitute currency string for radix character.

```

```

167         * To the best of my knowledge, no locale uses this.
168         */
169         if (strcmp(lmon->mon_decimal_point,
170                 lmon->currency_symbol) == 0)
171             sign = '.';
172         break;
173     }
174     (void) snprintf(lmon->crncystr, clen, "%c%s", sign,
175                   lmon->currency_symbol);
176 }
177     return (ret);
178 }
179
180 return (ldata);
181 struct lc_monetary_T *
182 __get_current_monetary_locale(void)
183 {
184     return (_monetary_using_locale ? &_monetary_locale :
185           (struct lc_monetary_T *)&_C_monetary_locale);
186 }
187
188 _____unchanged_portion_omitted_____

```

new/usr/src/lib/libc/port/locale/lmonetary.h

1

```
*****
2235 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/lmonetary.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #ifndef _LMONETARY_H_
29 #define _LMONETARY_H_

31 struct lc_monetary {
30 struct lc_monetary_T {
32     const char    *int_curr_symbol;
33     const char    *currency_symbol;
34     const char    *mon_decimal_point;
35     const char    *mon_thousands_sep;
36     const char    *mon_grouping;
37     const char    *positive_sign;
38     const char    *negative_sign;
39     const char    *int_frac_digits;
40     const char    *frac_digits;
41     const char    *p_cs_precedes;
42     const char    *p_sep_by_space;
43     const char    *n_cs_precedes;
44     const char    *n_sep_by_space;
45     const char    *p_sign_posn;
46     const char    *n_sign_posn;
47     const char    *int_p_cs_precedes;
48     const char    *int_n_cs_precedes;
49     const char    *int_p_sep_by_space;
50     const char    *int_n_sep_by_space;
51     const char    *int_p_sign_posn;
52     const char    *int_n_sign_posn;
53     char          *crNCYstr;          /* nl_langinfo */
54 };

54 struct lc_monetary_T *__get_current_monetary_locale(void);
56 int    __monetary_load_locale(const char *);
```

new/usr/src/lib/libc/port/locale/lmonetary.h

2

```
58 #endif /* !_LMONETARY_H_ */
```

```

*****
2721 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/lnumeric.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore.
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <errno.h>
32 #include "ldpart.h"
33 #include "lnumeric.h"
34 #include "localeimpl.h"
35 #include "../i18n/_locale.h"

34 extern int __nlocale_changed;
36 extern const char *__fix_locale_grouping_str(const char *);

38 #define LCNUMERIC_SIZE (sizeof (struct lc_numeric) / sizeof (char *))
37 #define LCNUMERIC_C_SIZE (sizeof (struct lc_numeric_T) / sizeof (char *))

40 static char numempty[] = { CHAR_MAX, '\0' };

42 struct lc_numeric lc_numeric_posix = {
41 static const struct lc_numeric_T _C_numeric_locale = {
43     ".", /* decimal_point */
44     ",", /* thousands_sep */
45     numempty /* grouping */
46 };

48 struct locdata __posix_numeric_locdata = {
49     .l_lname = "C",
50     .l_data = { &lc_numeric_posix }
51 };
47 static struct lc_numeric_T _numeric_locale;
48 static int _numeric_using_locale;
49 static char *_numeric_locale_buf;

```

```

54 /*
55  * Return the locale's numeric locdata structure.
56  */
57 struct locdata *
58 __lc_numeric_load(const char *name)
59 {
60     struct locdata *ldata;
61     struct lc_numeric *lnum;
62     const struct lc_numeric_T *leg = &_C_numeric_locale;

62     int ret;

64     if ((ldata = __locdata_alloc(name, sizeof (*lnum))) == NULL) {
65         errno = EINVAL;
66         return (NULL);
67     }
68     lnum = ldata->l_data[0];
69     ret = __part_load_locale(name, &numeric_using_locale,
70                             &numeric_locale_buf, "LC_NUMERIC", LCNUMERIC_SIZE, LCNUMERIC_C_SIZE,
71                             (const char **)&numeric_locale);
72     if (ret == _LDP_ERROR)
73         return (_LDP_ERROR);

70     ret = __part_load_locale(name, (char **)&ldata->l_data[1],
71                             "LC_NUMERIC", LCNUMERIC_SIZE, LCNUMERIC_C_SIZE, (const char **)&lnum);

73     if (ret != _LDP_LOADED) {
74         __locdata_free(ldata);
75         return (NULL);
76     }
77     __nlocale_changed = 1;
78     if (ret == _LDP_LOADED) {
79         /* Can't be empty according to C99 */
80         if (*_numeric_locale.decimal_point == '\0')
81             __numeric_locale.decimal_point =
82                 _C_numeric_locale.decimal_point;
83         __numeric_locale.grouping =
84             __fix_locale_grouping_str(_numeric_locale.grouping);
85         leg = (const struct lc_numeric_T *)&numeric_locale;

74     /* This is Solaris legacy, required for ABI compatibility */
75     __numeric[0] = *leg->decimal_point;
76     __numeric[1] = *leg->grouping;
77     return (ret);
78 }

78     /* Can't be empty according to C99 */
79     if (*lnum->decimal_point == '\0')
80         lnum->decimal_point = lc_numeric_posix.decimal_point;
81     lnum->grouping = __fix_locale_grouping_str(lnum->grouping);

83     return (ldata);
80 struct lc_numeric_T *
81 __get_current_numeric_locale(void)
82 {
83     return (_numeric_using_locale ? &numeric_locale :
84         (struct lc_numeric_T *)&_C_numeric_locale);
84 }

__unchanged_portion_omitted_

```

new/usr/src/lib/libc/port/locale/lnumeric.h

1

\*\*\*\*\*

1611 Fri Jul 11 11:42:58 2014

new/usr/src/lib/libc/port/locale/lnumeric.h

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 2000, 2001 Alexey Zelkin <phantom@FreeBSD.org>
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #ifndef _LNUMERIC_H_
29 #define _LNUMERIC_H_

31 struct lc_numeric {
30 struct lc_numeric_T {
32     const char    *decimal_point;
33     const char    *thousands_sep;
34     const char    *grouping;
35 };

36 struct lc_numeric_T *__get_current_numeric_locale(void);
37 int    __numeric_load_locale(const char *);

37 #endif /* !_LNUMERIC_H_ */
```



```

*****
3818 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/localeconv.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * Copyright (c) 1991, 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 * 4. Neither the name of the University nor the names of its contributors
17 * may be used to endorse or promote products derived from this software
18 * without specific prior written permission.
19 *
20 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
21 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
22 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
23 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
24 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
25 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
26 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
27 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
28 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
29 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
30 * SUCH DAMAGE.
31 */

33 #ifndef _LCONV_C99
34 #define _LCONV_C99 /* so we get all the extensions */
35 #endif

37 #include "lint.h"
38 #include <locale.h>
39 #include "lmonetary.h"
40 #include "lnumeric.h"
41 #include "localeimpl.h"

43 /*
44  * Return the current locale conversion.
45  * The localeconv() function constructs a struct lconv from the current
46  * monetary and numeric locales.
47  *
48  * Note that XPG7 specifically states that localeconv's return value may
49  * be invalidated if the application calls setlocale() or uselocale() within
50  * the same thread.
51  *
52  * Because localeconv() may be called many times (especially by library
53  * routines like printf() & strtod()), the appropriate members of the
54  * lconv structure are computed only when the monetary or numeric
55  * locale has been changed.
56  */
50 int __mlocale_changed = 1;
51 int __nlocale_changed = 1;

```

```

53 /*
54  * Return the current locale conversion.
55  */
55 struct lconv *
56 localeconv(void)
57 {
58     struct lconv *lconv;
59     locale_t loc;
60     struct lc_monetary *mptr;
61     struct lc_numeric *nptr;
62     static struct lconv ret;

63     loc = uselocale(NULL);
64     lconv = &loc->lconv;
65     if (__mlocale_changed) {
66         /* LC_MONETARY part */
67         struct lc_monetary_T *mptr;

68         if (loc->loaded[LC_MONETARY] == 0) {
69             mptr = loc->locdata[LC_MONETARY]->l_data[0];
70             #define M_ASSIGN_STR(NAME) (ret.NAME = (char *)mptr->NAME)
71             #define M_ASSIGN_CHAR(NAME) (ret.NAME = mptr->NAME[0])

72             #define M_ASSIGN_STR(NAME) (lconv->NAME = (char *)mptr->NAME)
73             #define M_ASSIGN_CHAR(NAME) (lconv->NAME = mptr->NAME[0])

68             mptr = __get_current_monetary_locale();
72             M_ASSIGN_STR(int_curr_symbol);
73             M_ASSIGN_STR(currency_symbol);
74             M_ASSIGN_STR(mon_decimal_point);
75             M_ASSIGN_STR(mon_thousands_sep);
76             M_ASSIGN_STR(mon_grouping);
77             M_ASSIGN_STR(positive_sign);
78             M_ASSIGN_STR(negative_sign);
79             M_ASSIGN_CHAR(int_frac_digits);
80             M_ASSIGN_CHAR(frac_digits);
81             M_ASSIGN_CHAR(p_cs_precedes);
82             M_ASSIGN_CHAR(p_sep_by_space);
83             M_ASSIGN_CHAR(n_cs_precedes);
84             M_ASSIGN_CHAR(n_sep_by_space);
85             M_ASSIGN_CHAR(p_sign_posn);
86             M_ASSIGN_CHAR(n_sign_posn);
87             M_ASSIGN_CHAR(int_p_cs_precedes);
88             M_ASSIGN_CHAR(int_n_cs_precedes);
89             M_ASSIGN_CHAR(int_p_sep_by_space);
90             M_ASSIGN_CHAR(int_n_sep_by_space);
91             M_ASSIGN_CHAR(int_p_sign_posn);
92             M_ASSIGN_CHAR(int_n_sign_posn);
93             loc->loaded[LC_MONETARY] = 1;
94             __mlocale_changed = 0;
95         }

96         if (loc->loaded[LC_NUMERIC] == 0) {
97             nptr = loc->locdata[LC_NUMERIC]->l_data[0];
98             if (__nlocale_changed) {
99                 /* LC_NUMERIC part */
100                struct lc_numeric_T *nptr;

99                #define N_ASSIGN_STR(NAME) (lconv->NAME = (char *)nptr->NAME)
100                #define N_ASSIGN_STR(NAME) (ret.NAME = (char *)nptr->NAME)

99                nptr = __get_current_numeric_locale();
101                N_ASSIGN_STR(decimal_point);
102                N_ASSIGN_STR(thousands_sep);
103                N_ASSIGN_STR(grouping);

```

new/usr/src/lib/libc/port/locale/localeconv.c

3

```
104         loc->loaded[LC_NUMERIC] = 1;
103         __nlocale_changed = 0;
105     }
```

```
107     return (lconv);
106     return (&ret);
108 }
```

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/localeimpl.c

1

```
*****
13954 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/localeimpl.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 2014 Garrett D'Amore <garrett@damore.org>
14 */
16 /*
17  * This file implements the 2008 newlocale and friends handling.
18 */
20 #ifndef _LCONV_C99
21 #define _LCONV_C99
22 #endif
24 #include "lint.h"
25 #include <atomic.h>
26 #include <locale.h>
27 #include <sys/types.h>
28 #include <sys/mman.h>
29 #include <errno.h>
30 #include <string.h>
31 #include "libc.h"
32 #include "mtlib.h"
33 #include "tsd.h"
34 #include "localeimpl.h"
35 #include "lctype.h"
37 /*
38  * Big Theory of Locales:
39  *
40  * (It is recommended that readers familiarize themselves with the POSIX
41  * 2008 (XPG Issue 7) specifications for locales, first.)
42  *
43  * Historically, we had a bunch of global variables that stored locale
44  * data. While this worked well, it limited applications to a single locale
45  * at a time. This doesn't work well in certain server applications.
46  *
47  * Issue 7, X/Open introduced the concept of a locale_t object, along with
48  * versions of functions that can take this object as a parameter, along
49  * with functions to clone and manipulate these locale objects. The new
50  * functions are named with a _l() suffix.
51  *
52  * Additionally uselocale() is introduced which can change the locale of
53  * of a single thread. However, setlocale() can still be used to change
54  * the global locale.
55  *
56  * In our implementation, we use libc's TSD to store the locale data that
57  * was previously global. We still have global data because some applications
58  * have had those global objects compiled into them. (Such applications will
```

new/usr/src/lib/libc/port/locale/localeimpl.c

2

```
59  * be unable to benefit from uselocale(), btw.) The legacy routines are
60  * reimplemented as wrappers that use the appropriate locale object by
61  * calling uselocale(). uselocale() when passed a NULL pointer returns the
62  * thread-specific locale object if one is present, or the global locale
63  * object otherwise. Note that once the TSD data is set, the only way
64  * to revert to the global locale is to pass the global locale LC_GLOBAL_LOCALE
65  * to uselocale().
66  *
67  * We are careful to minimize performance impact of multiple calls to
68  * uselocale() or setlocale() by using a cache of locale data whenever possible.
69  * As a consequence of this, applications that iterate over all possible
70  * locales will burn through a lot of virtual memory, but we find such
71  * applications rare. (locale -a might be an exception, but it is short lived.)
72  *
73  * Category data is never released (although enclosing locale objects might be),
74  * in order to guarantee thread-safety. Calling freelocale() on an object
75  * while it is in use by another thread is a programmer error (use-after-free)
76  * and we don't bother to note it further.
77  *
78  * Locale objects (global locales) established by setlocale() are also
79  * never freed (for MT safety), but we will save previous locale objects
80  * and reuse them when we can.
81 */
83 typedef struct locdata>(*loadfn_t)(const char *);
85 static const loadfn_t loaders[LC_ALL] = {
86     __lc_ctype_load,
87     __lc_numeric_load,
88     __lc_time_load,
89     __lc_collate_load,
90     __lc_monetary_load,
91     __lc_messages_load,
92 };
94 extern struct lc_monetary lc_monetary_posix;
95 extern struct lc_numeric lc_numeric_posix;
96 extern struct lc_messages lc_messages_posix;
97 extern struct lc_time lc_time_posix;
98 extern struct lc_ctype lc_ctype_posix;
99 extern struct lc_collate lc_collate_posix;
101 static struct locale posix_locale = {
102     /* locdata */
103     .locdata = {
104         &__posix_ctype_locdata,
105         &__posix_numeric_locdata,
106         &__posix_time_locdata,
107         &__posix_collate_locdata,
108         &__posix_monetary_locdata,
109         &__posix_messages_locdata,
110     },
111     .locname = "C",
112     .ctype = &lc_ctype_posix,
113     .numeric = &lc_numeric_posix,
114     .collate = &lc_collate_posix,
115     .monetary = &lc_monetary_posix,
116     .messages = &lc_messages_posix,
117     .time = &lc_time_posix,
118     .runelocale = &DefaultRuneLocale,
119 };
121 locale_t __global_locale = &posix_locale;
123 locale_t
124 __global_locale(void)
```

```

125 {
126     return (___global_locale);
127 }

129 /*
130  * Category names for getenv() Note that this was modified
131  * for Solaris. See <iso/locale_iso.h>.
132  */
133 #define NUM_CATS        7
134 static char *categories[7] = {
135     "LC_CTYPE",
136     "LC_NUMERIC",
137     "LC_TIME",
138     "LC_COLLATE",
139     "LC_MONETARY",
140     "LC_MESSAGES",
141     "LC_ALL",
142 };

144 /*
145  * Prototypes.
146  */
147 static const char *get_locale_env(int);
148 static struct locdata *locdata_get(int, const const char *);
149 static struct locdata *locdata_get_cache(int, const char *);
150 static locale_t mklocname(locale_t);

152 /*
153  * Some utility routines.
154  */

156 struct locdata *
157 __locdata_alloc(const char *name, size_t memsz)
158 {
159     struct locdata *ldata;

161     if ((ldata = lmalloc(sizeof(*ldata))) == NULL) {
162         return (NULL);
163     }
164     if ((ldata->l_data[0] = libc_malloc(memsz)) == NULL) {
165         lfree(ldata, sizeof(*ldata));
166         errno = ENOMEM;
167         return (NULL);
168     }
169     (void) strncpy(ldata->l_name, name, sizeof(ldata->l_name));

171     return (ldata);
172 }

174 /*
175  * Normally we never free locale data truly, but if we failed to load it
176  * for some reason, this routine is used to cleanup the partial mess.
177  */
178 void
179 __locdata_free(struct locdata *ldata)
180 {
181     for (int i = 0; i < NLOCDATA; i++)
182         libc_free(ldata->l_data[i]);
183     if (ldata->l_map != NULL && ldata->l_map_len)
184         (void) munmap(ldata->l_map, ldata->l_map_len);
185     lfree(ldata, sizeof(*ldata));
186 }

188 /*
189  * It turns out that for performance reasons we would really like to
190  * cache the most recently referenced locale data to avoid wasteful

```

```

191  * loading from files.
192  */

194 static struct locdata *cache_data[LC_ALL];
195 static struct locdata *cat_data[LC_ALL];
196 static mutex_t cache_lock = DEFAULTMUTEX;

198 /*
199  * Returns the cached data if the locale name is the same. If not,
200  * returns NULL (cache miss). The locdata is returned with a hold on
201  * it, taken on behalf of the caller. The caller should drop the hold
202  * when it is finished.
203  */
204 static struct locdata *
205 locdata_get_cache(int category, const char *locname)
206 {
207     struct locdata *loc;

209     if (category < 0 || category >= LC_ALL)
210         return (NULL);

212     /* Try cache first. */
213     lmutex_lock(&cache_lock);
214     loc = cache_data[category];

216     if ((loc != NULL) && (strcmp(loc->l_name, locname) == 0)) {
217         lmutex_unlock(&cache_lock);
218         return (loc);
219     }

221     /*
222      * Failing that try previously loaded locales (linear search) --
223      * this could be optimized to a hash, but its unlikely that a single
224      * application will ever need to work with more than a few locales.
225      */
226     for (loc = cat_data[category]; loc != NULL; loc = loc->l_next) {
227         if (strcmp(locname, loc->l_name) == 0) {
228             break;
229         }
230     }

232     /*
233      * Finally, if we still don't have one, try loading the locale
234      * data from the actual on-disk data.
235      *
236      * We drop the lock (libc wants to ensure no internal locks
237      * are held when we call other routines required to read from
238      * files, allocate memory, etc.) There is a small race here,
239      * but the consequences of the race are benign -- if multiple
240      * threads hit this at precisely the same point, we could
241      * wind up with duplicates of the locale data in the cache.
242      *
243      * This wastes the memory for an extra copy of the locale
244      * data, but there is no further harm beyond that. Its not
245      * worth the effort to recode this to something "safe"
246      * (which would require rescanning the list, etc.), given
247      * that this race will probably never actually occur.
248      */
249     if (loc == NULL) {
250         lmutex_unlock(&cache_lock);
251         loc = (*loaders[category])(locname);
252         lmutex_lock(&cache_lock);
253         if (loc != NULL)
254             (void) strncpy(loc->l_name, locname,
255                 sizeof(loc->l_name));
256     }

```

```

258     /*
259     * Assuming we got one, update the cache, and stick us on the list
260     * of loaded locale data. We insert into the head (more recent
261     * use is likely to win.)
262     */
263     if (loc != NULL) {
264         cache_data[category] = loc;
265         if (!loc->l_cached) {
266             loc->l_cached = 1;
267             loc->l_next = cat_data[category];
268             cat_data[category] = loc;
269         }
270     }

272     lmutex_unlock(&cache_lock);
273     return (loc);
274 }

276 /*
277 * Routine to get the locdata for a given category and locale.
278 * This includes retrieving it from cache, retrieving it from
279 * a file, etc.
280 */
281 static struct locdata *
282 locdata_get(int category, const char *locname)
283 {
284     char scratch[ENCODING_LEN + 1];
285     char *slash;
286     int cnt;
287     int len;

289     if (locname == NULL || *locname == 0) {
290         locname = get_locale_env(category);
291     }

293     /*
294     * Extract the locale name for the category if it is a composite
295     * locale.
296     */
297     if ((slash = strchr(locname, '/')) != NULL) {
298         for (cnt = category; cnt && slash != NULL; cnt--) {
299             locname = slash + 1;
300             slash = strchr(locname, '/');
301         }
302         if (slash) {
303             len = slash - locname + 1;
304             if (len >= sizeof (scratch)) {
305                 len = sizeof (scratch);
306             }
307         } else {
308             len = sizeof (scratch);
309         }
310         (void) strncpy(scratch, locname, len);
311         locname = scratch;
312     }

314     if ((strcmp(locname, "C") == 0) || (strcmp(locname, "POSIX") == 0))
315         return (posix_locale.locdata[category]);

317     return (locdata_get_cache(category, locname));
318 }

320 /* tsd destructor */
321 static void
322 freelocptr(void *arg)

```

```

323 {
324     locale_t *locptr = arg;
325     if (*locptr != NULL)
326         freelocale(*locptr);
327 }

329 static const char *
330 get_locale_env(int category)
331 {
332     const char *env;

334     /* 1. check LC_ALL. */
335     env = getenv(categories[LC_ALL]);

337     /* 2. check LC_* */
338     if (env == NULL || *env == '\0')
339         env = getenv(categories[category]);

341     /* 3. check LANG */
342     if (env == NULL || *env == '\0')
343         env = getenv("LANG");

345     /* 4. if none is set, fall to "C" */
346     if (env == NULL || *env == '\0')
347         env = "C";

349     return (env);
350 }

353 /*
354 * This routine is exposed via the MB_CUR_MAX macro. Note that legacy
355 * code will continue to use _ctype[520], but we prefer this function as
356 * it is the only way to get thread-specific information.
357 */
358 unsigned char
359 __mb_cur_max_l(locale_t loc)
360 {
361     return (loc->ctype->lc_max_mblen);
362 }

364 unsigned char
365 __mb_cur_max(void)
366 {
367     return (__mb_cur_max_l(uselocale(NULL)));
368 }

370 /*
371 * Public interfaces.
372 */

374 locale_t
375 duplocale(locale_t src)
376 {
377     locale_t loc;
378     int i;

380     loc = lmalloc(sizeof (*loc));
381     if (loc == NULL) {
382         return (NULL);
383     }
384     if (src == NULL) {
385         /* illumos extension: POSIX says LC_GLOBAL_LOCALE here */
386         src = __global_locale;
387     }
388     for (i = 0; i < LC_ALL; i++) {

```

```

389     loc->locdata[i] = src->locdata[i];
390     loc->loaded[i] = 0;
391 }
392 loc->collate = loc->locdata[LC_COLLATE]->l_data[0];
393 loc->ctype = loc->locdata[LC_CTYPE]->l_data[0];
394 loc->runelocale = loc->locdata[LC_CTYPE]->l_data[1];
395 loc->messages = loc->locdata[LC_MESSAGES]->l_data[0];
396 loc->monetary = loc->locdata[LC_MONETARY]->l_data[0];
397 loc->numeric = loc->locdata[LC_NUMERIC]->l_data[0];
398 loc->time = loc->locdata[LC_TIME]->l_data[0];
399 return (loc);
400 }

402 void
403 freelocale(locale_t loc)
404 {
405     /*
406     * We take extra care never to free a saved locale created by
407     * setlocale(). This shouldn't be strictly necessary, but a little
408     * extra safety doesn't hurt here.
409     */
410     if ((loc != NULL) && (loc != &posix_locale) && (!loc->on_list))
411         lfree(loc, sizeof (*loc));
412 }

414 locale_t
415 newlocale(int catmask, const char *locname, locale_t base)
416 {
417     locale_t loc;
418     int i, e;

420     if (catmask & ~(LC_ALL_MASK)) {
421         errno = EINVAL;
422         return (NULL);
423     }

425     /*
426     * Technically passing LC_GLOBAL_LOCALE here is illegal,
427     * but we allow it.
428     */
429     if (base == NULL || base == __global_locale) {
430         loc = duplocale(__global_locale);
431     } else {
432         loc = duplocale(base);
433     }
434     if (loc == NULL) {
435         return (NULL);
436     }

438     for (i = 0; i < LC_ALL; i++) {
439         struct locdata *ldata;
440         loc->loaded[i] = 0;
441         if (((1 << i) & catmask) == 0) {
442             /* Default to base locale if not overriding */
443             continue;
444         }
445         ldata = locdata_get(i, locname);
446         if (ldata == NULL) {
447             e = errno;
448             freelocale(loc);
449             errno = e;
450             return (NULL);
451         }
452         loc->locdata[i] = ldata;
453     }
454     loc->collate = loc->locdata[LC_COLLATE]->l_data[0];

```

```

455     loc->ctype = loc->locdata[LC_CTYPE]->l_data[0];
456     loc->runelocale = loc->locdata[LC_CTYPE]->l_data[1];
457     loc->messages = loc->locdata[LC_MESSAGES]->l_data[0];
458     loc->monetary = loc->locdata[LC_MONETARY]->l_data[0];
459     loc->numeric = loc->locdata[LC_NUMERIC]->l_data[0];
460     loc->time = loc->locdata[LC_TIME]->l_data[0];
461     freelocale(base);

463     return (mklocname(loc));
464 }

466 locale_t
467 uselocale(locale_t loc)
468 {
469     locale_t lastloc = __global_locale;
470     locale_t *locptr;

472     locptr = tsdalloc(_T_SETLOCALE, sizeof (locale_t), freelocptr);
473     /* Should never occur */
474     if (locptr == NULL) {
475         errno = EINVAL;
476         return (NULL);
477     }

479     if (*locptr != NULL)
480         lastloc = *locptr;

482     /* Argument loc is NULL if we are just querying. */
483     if (loc != NULL) {
484         /*
485         * Set it to LC_GLOBAL_LOCALE to return to using
486         * the global locale (setlocale).
487         */
488         if (loc == __global_locale) {
489             *locptr = NULL;
490         } else {
491             /* No validation of the provided locale at present */
492             *locptr = loc;
493         }
494     }

496     /*
497     * The caller is responsible for freeing, of course it would be
498     * gross error to call freelocale() on a locale object that is still
499     * in use.
500     */
501     return (lastloc);
502 }

504 static locale_t
505 mklocname(locale_t loc)
506 {
507     int composite = 0;

509     /* Look to see if any category is different */
510     for (int i = 1; i < LC_ALL; ++i) {
511         if (strcmp(loc->locdata[0]->l_lname,
512             loc->locdata[i]->l_lname) != 0) {
513             composite = 1;
514             break;
515         }
516     }

518     if (composite) {
519         /*
520         * Note ordering of these follows the numeric order,

```

```
521     * if the order is changed, then setlocale() will need
522     * to be changed as well.
523     */
524     (void) snprintf(loc->locname, sizeof (loc->locname),
525                  "%s/%s/%s/%s/%s",
526                  loc->locdata[LC_CTYPE]->l_name,
527                  loc->locdata[LC_NUMERIC]->l_name,
528                  loc->locdata[LC_TIME]->l_name,
529                  loc->locdata[LC_COLLATE]->l_name,
530                  loc->locdata[LC_MONETARY]->l_name,
531                  loc->locdata[LC_MESSAGES]->l_name);
532 } else {
533     (void) strcpy(loc->locname, loc->locdata[LC_CTYPE]->l_name,
534                 sizeof (loc->locname));
535 }
536 return (loc);
537 }
```

```

*****
3482 Fri Jul 11 11:42:58 2014
new/usr/src/lib/libc/port/locale/localeimpl.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 2014 Garrett D'Amore <garrett@damore.org>
14 */
16 /*
17  * This file implements the 2008 newlocale and friends handling. It is
18  * private to libc.
19 */
20 #ifndef _LOCALEIMPL_H
21 #define _LOCALEIMPL_H
23 #ifndef _LCONV_C99
24 #define _LCONV_C99 /* so we get all the extensions */
25 #endif
27 #include <sys/types.h>
28 #include <locale.h>
29 #include <xlocale.h>
30 #include "setlocale.h"
31 #include "runetype.h"
33 /* private locale structures */
35 /*
36  * Because some locale data is rather ahem.. large, we would like to keep
37  * reference counts on it. We create an abstract header (locdata) structure
38  * which keeps a point to the opaque per-category data, along with a reference
39  * count to it. To be threadsafe, we will use atomics when holding it or
40  * freeing it. (This only occurs when locale objects are created or destroyed,
41  * so there should be no performance impact on hot code paths. If your code
42  * uses locale_t creation/destruction on a hot code path, its broken. But
43  * even so, the atomic and reference counting will probably *greatly* improve
44  * your life as bootstrapping locale data from files is quite expensive.
45 */
47 #define NLOCDATA 4
48 struct locdata {
49     char          l_name[ENCODING_LEN+1]; /* locale name */
50     void          *l_data[NLOCDATA];    /* storage area */
51     void          *l_map;               /* mapped file */
52     size_t       l_map_len;
53     struct locdata *l_next;             /* link cached list */
54     int          l_cached;              /* nonzero if cached */
55 };
58 struct locale {

```

```

59     struct locdata *locdata[LC_ALL];
60     struct locale *next;
61     int          on_list; /* on linked list */
62     char         locname[(ENCODING_LEN+1)*NLOCDATA + 1];
64     /*
65     * Convenience pointers.
66     */
67     const struct lc_ctype *ctype;
68     const struct lc_collate *collate;
69     const struct lc_messages *messages;
70     const struct lc_monetary *monetary;
71     const struct lc_numeric *numeric;
72     const struct lc_time *time;
73     const _RuneLocale *runelocale;
75     /*
76     * The loaded value is used for localeconv. In particular, when
77     * when we change the value of one of the above categories, we will
78     * also need to update the lconv structure. The loaded bit indicates
79     * that the lconv structure is "current" for that category. It's
80     * sort of an "inverse dirty" bit.
81     */
82     int          loaded[LC_ALL];
83     struct lconv lconv;
84 };
87 struct locdata *__locdata_alloc(const char *, size_t);
88 void __locdata_free(struct locdata *);
89 struct locdata *__locdata_get_cache(int, const char *);
90 void __locdata_set_cache(int, struct locdata *);
92 struct locdata *__lc_numeric_load(const char *name);
93 struct locdata *__lc_monetary_load(const char *name);
94 struct locdata *__lc_messages_load(const char *name);
95 struct locdata *__lc_time_load(const char *name);
96 struct locdata *__lc_ctype_load(const char *name);
97 struct locdata *__lc_collate_load(const char *name);
99 extern struct locdata __posix_numeric_locdata;
100 extern struct locdata __posix_monetary_locdata;
101 extern struct locdata __posix_messages_locdata;
102 extern struct locdata __posix_time_locdata;
103 extern struct locdata __posix_ctype_locdata;
104 extern struct locdata __posix_collate_locdata;
105 extern locale_t __global_locale;
107 #endif /* _LOCALEIMPL_H */

```



new/usr/src/lib/libc/port/locale/mblen.c

1

\*\*\*\*\*

2009 Fri Jul 11 11:42:59 2014

new/usr/src/lib/libc/port/locale/mblen.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */
29 #include "lint.h"
30 #include <stdlib.h>
31 #include <wchar.h>
32 #include <locale.h>
33 #include <xlocale.h>
34 #include "mblocal.h"
35 #include "localeimpl.h"
37 int
38 mblen_l(const char *s, size_t n, locale_t loc)
39 mblen(const char *s, size_t n)
40 {
41     mbstate_t mbs = { 0 };
42     static const mbstate_t initial = { 0 };
43     static mbstate_t mbs;
44     size_t rval;
45
46     if (s == NULL) {
47         /* No support for state dependent encodings. */
48         mbs = initial;
49         return (0);
50     }
51     rval = mbrtowc_l(NULL, s, n, &mbs, loc);
52     rval = __mbrtowc(NULL, s, n, &mbs);
53     if (rval == (size_t)-1 || rval == (size_t)-2)
54         return (-1);
55     return ((int)rval);
56 }
57 int
```

new/usr/src/lib/libc/port/locale/mblen.c

2

```
54 mblen(const char *s, size_t n)
55 {
56     return (mblen_l(s, n, uselocale(NULL)));
57 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libc/port/locale/mblocal.h

1

```
*****
2592 Fri Jul 11 11:42:59 2014
new/usr/src/lib/libc/port/locale/mblocal.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #ifndef _MBLOCAL_H_
30 #define _MBLOCAL_H_

32 #include "runetype.h"
33 #include "lctype.h"

35 /*
36  * Rune initialization function prototypes.
37  */
38 void _none_init(struct lc_ctype *);
39 void _UTF8_init(struct lc_ctype *);
40 void _EUC_CN_init(struct lc_ctype *);
41 void _EUC_JP_init(struct lc_ctype *);
42 void _EUC_KR_init(struct lc_ctype *);
43 void _EUC_TW_init(struct lc_ctype *);
44 void _GB18030_init(struct lc_ctype *);
45 void _GB2312_init(struct lc_ctype *);
46 void _GBK_init(struct lc_ctype *);
47 void _BIG5_init(struct lc_ctype *);
48 void _MSKanji_init(struct lc_ctype *);
36 int _none_init(_RuneLocale *);
37 int _UTF8_init(_RuneLocale *);
38 int _EUC_CN_init(_RuneLocale *);
39 int _EUC_JP_init(_RuneLocale *);
40 int _EUC_KR_init(_RuneLocale *);
41 int _EUC_TW_init(_RuneLocale *);
42 int _GB18030_init(_RuneLocale *);
43 int _GB2312_init(_RuneLocale *);
44 int _GBK_init(_RuneLocale *);
45 int _BIG5_init(_RuneLocale *);
```

new/usr/src/lib/libc/port/locale/mblocal.h

2

```
46 int _MSKanji_init(_RuneLocale *);

50 typedef size_t (*mbrtowc_pfn_t)(wchar_t *_RESTRICT_KYWD,
48 /*
49  * Conversion function pointers for current encoding.
50  */
51 extern size_t (*__mbrtowc)(wchar_t *_RESTRICT_KYWD,
51 const char *_RESTRICT_KYWD, size_t, mbstate_t *_RESTRICT_KYWD);
52 typedef size_t (*wcrctomb_pfn_t)(char *_RESTRICT_KYWD, wchar_t,
53 extern int (*__mbsinit)(const mbstate_t *);
54 extern size_t (*__mbsnrtowcs)(wchar_t *_RESTRICT_KYWD,
55 const char **_RESTRICT_KYWD, size_t, size_t, mbstate_t *_RESTRICT_KYWD);

57 extern size_t (*__wcrctomb)(char *_RESTRICT_KYWD, wchar_t,
53 mbstate_t *_RESTRICT_KYWD);

60 extern size_t (*__wcsnrtombs)(char *_RESTRICT_KYWD,
61 const wchar_t **_RESTRICT_KYWD, size_t, size_t, mbstate_t *_RESTRICT_KYWD);

63 extern int charset_is_ascii;

54 size_t __mbsnrtowcs_std(wchar_t *_RESTRICT_KYWD, const char **_RESTRICT_KYWD,
55 size_t, size_t, mbstate_t *_RESTRICT_KYWD, mbrtowc_pfn_t);
66 size_t, size_t, mbstate_t *_RESTRICT_KYWD);
56 size_t __wcsnrtombs_std(char *_RESTRICT_KYWD, const wchar_t **_RESTRICT_KYWD,
57 size_t, size_t, mbstate_t *_RESTRICT_KYWD, wcrctomb_pfn_t);
68 size_t, size_t, mbstate_t *_RESTRICT_KYWD);

59 #define MIN(a, b) ((a) < (b) ? (a) : (b))

61 #endif /* _MBLOCAL_H_ */
```

new/usr/src/lib/libc/port/locale/mbrlen.c

1

\*\*\*\*\*

1919 Fri Jul 11 11:42:59 2014

new/usr/src/lib/libc/port/locale/mbrlen.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <wchar.h>
31 #include <locale.h>
32 #include <xlocale.h>
33 #include "mblocal.h"

35 size_t
36 mbrlen_l(const char *_RESTRICT_KYWD s, size_t n, mbstate_t *_RESTRICT_KYWD ps,
37         locale_t loc)
38 {
39     static mbstate_t mbs;

41     if (ps == NULL)
42         ps = &mbs;
43     return (mbrtowc_l(NULL, s, n, ps, loc));
44 }

46 size_t
47 mbrlen(const char *_RESTRICT_KYWD s, size_t n, mbstate_t *_RESTRICT_KYWD ps)
48 {
49     return (mbrlen_l(s, n, ps, uselocale(NULL)));
50     return (__mbrtowc(NULL, s, n, ps));
51 }

unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/mbrtowc.c

1

\*\*\*\*\*

1999 Fri Jul 11 11:42:59 2014

new/usr/src/lib/libc/port/locale/mbrtowc.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <wchar.h>
31 #include "mblocal.h"
32 #include "localeimpl.h"
33 #include "lctype.h"

35 size_t
36 mbrtowc_l(wchar_t * _RESTRICT_KYWD pwc, const char * _RESTRICT_KYWD s,
37           size_t n, mbstate_t * _RESTRICT_KYWD ps, locale_t loc)
38 mbrtowc(wchar_t * _RESTRICT_KYWD pwc, const char * _RESTRICT_KYWD s,
39         size_t n, mbstate_t * _RESTRICT_KYWD ps)
40 {
41     static mbstate_t mbs;
42
43     if (ps == NULL)
44         ps = &mbs;
45     return (loc->ctype->lc_mbrtowc(pwc, s, n, ps));
46 }

47 size_t
48 mbrtowc(wchar_t * _RESTRICT_KYWD pwc, const char * _RESTRICT_KYWD s,
49         size_t n, mbstate_t * _RESTRICT_KYWD ps)
50 {
51     return (mbrtowc_l(pwc, s, n, ps, uselocale(NULL)));
52     return (__mbrtowc(pwc, s, n, ps));
53 }
54
55 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libc/port/locale/mbsinit.c

1

\*\*\*\*\*

727 Fri Jul 11 11:42:59 2014

new/usr/src/lib/libc/port/locale/mbsinit.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
7  * Copyright (c) 2002-2004 Tim J. Robbins.
8  * All rights reserved.
9  *
10 * A full copy of the text of the CDDL should have accompanied this
11 * source. A copy of the CDDL is also available via the Internet at
12 * http://www.illumos.org/license/CDDL.
13 * Redistribution and use in source and binary forms, with or without
14 * modification, are permitted provided that the following conditions
15 * are met:
16 * 1. Redistributions of source code must retain the above copyright
17 * notice, this list of conditions and the following disclaimer.
18 * 2. Redistributions in binary form must reproduce the above copyright
19 * notice, this list of conditions and the following disclaimer in the
20 * documentation and/or other materials provided with the distribution.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */
34
35 #include "lint.h"
36 #include <locale.h>
37 #include "localeimpl.h"
38 #include "lctype.h"
39 #include <wchar.h>
40 #include "mblocal.h"
41
42 int
43 mbsinit_l(const mbstate_t *s, locale_t loc)
44 {
45     return (loc->ctype->lc_mbsinit(s));
46 }
47
48 int
49 mbsinit(const mbstate_t *s)
50 {
51     return (mbsinit_l(s, uselocale(NULL)));
52 }
53
54 return (__mbsinit(ps));
55 }
```

```

*****
2983 Fri Jul 11 11:42:59 2014
new/usr/src/lib/libc/port/locale/mbsnrtowcs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <errno.h>
31 #include <limits.h>
32 #include <stdlib.h>
33 #include <wchar.h>
34 #include "mblocal.h"
35 #include "localeimpl.h"
36 #include "lctype.h"

38 size_t
39 mbsnrtowcs_l(wchar_t * _RESTRICT_KYWD dst, const char ** _RESTRICT_KYWD src,
40             size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps, locale_t loc)
41 mbsnrtowcs(wchar_t * _RESTRICT_KYWD dst, const char ** _RESTRICT_KYWD src,
42            size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
43 {
44     static mbstate_t mbs;
45
46     if (ps == NULL)
47         ps = &mbs;
48     return (loc->ctype->lc_mbsnrtowcs(dst, src, nms, len, ps));
49     return (__mbsnrtowcs(dst, src, nms, len, ps));
43 }

49 size_t
50 mbsnrtowcs(wchar_t * _RESTRICT_KYWD dst, const char ** _RESTRICT_KYWD src,
51            __mbsnrtowcs_std(wchar_t * _RESTRICT_KYWD dst, const char ** _RESTRICT_KYWD src,
52                           size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
53 {
54     return (mbsnrtowcs_l(dst, src, nms, len, ps, uselocale(NULL)));
54 }

```

```

56 size_t
57 __mbsnrtowcs_std(wchar_t * _RESTRICT_KYWD dst, const char ** _RESTRICT_KYWD src,
58                size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps,
59                mbrtowc_pfn_t pmbrtowc)
60 {
61     const char *s;
62     size_t nchr;
63     wchar_t wc;
64     size_t nb;
65
66     s = *src;
67     nchr = 0;
68
69     if (dst == NULL) {
70         for (;;) {
71             if ((nb = pmbrtowc(&wc, s, nms, ps)) == (size_t)-1)
72                 if ((nb = __mbrtowc(&wc, s, nms, ps)) == (size_t)-1)
73                     /* Invalid sequence - mbrtowc() sets errno. */
74                     return ((size_t)-1);
75             else if (nb == 0 || nb == (size_t)-2)
76                 return (nchr);
77             s += nb;
78             nms -= nb;
79             nchr++;
80         } /*NOTREACHED*/
81     }
82
83     while (len-- > 0) {
84         if ((nb = pmbrtowc(dst, s, nms, ps)) == (size_t)-1) {
85             if ((nb = __mbrtowc(dst, s, nms, ps)) == (size_t)-1) {
86                 *src = s;
87                 return ((size_t)-1);
88             } else if (nb == (size_t)-2) {
89                 *src = s + nms;
90                 return (nchr);
91             } else if (nb == 0) {
92                 *src = NULL;
93                 return (nchr);
94             }
95             s += nb;
96             nms -= nb;
97             nchr++;
98             dst++;
99         }
100         *src = s;
101         return (nchr);
101 }

```

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/mbsrtowcs.c

1

\*\*\*\*\*

2043 Fri Jul 11 11:42:59 2014

new/usr/src/lib/libc/port/locale/mbsrtowcs.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */
28
29 #include "lint.h"
30 #include <locale.h>
29 #include <errno.h>
31 #include <limits.h>
32 #include <stdlib.h>
33 #include <wchar.h>
34 #include "mblocal.h"
35 #include "localeimpl.h"
36 #include "lctype.h"
37
38 size_t
39 mbsrtowcs_l(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
40             size_t len, mbstate_t *_RESTRICT_KYWD ps, locale_t loc)
41 {
42     return (loc->ctype->lc_mbsnrtowcs(dst, src, ULONG_MAX, len, ps));
43 }
44
45 size_t
46 mbsrtowcs(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
47           size_t len, mbstate_t *_RESTRICT_KYWD ps)
48 {
49     return (mbsrtowcs_l(dst, src, len, ps, uselocale(NULL)));
39     static mbstate_t mbs;
40
41     if (ps == NULL)
42         ps = &mbs;
43     return (__mbsnrtowcs(dst, src, ULONG_MAX, len, ps));
44 }
45
46 _____
47 unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/mbstowcs.c

1

\*\*\*\*\*

2062 Fri Jul 11 11:43:00 2014

new/usr/src/lib/libc/port/locale/mbstowcs.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stdlib.h>
32 #include <wchar.h>
33 #include <locale.h>
34 #include "mblocal.h"
35 #include "localeimpl.h"
36 #include "lctype.h"

38 size_t
39 mbstowcs_l(wchar_t *_RESTRICT_KYWD pwcs, const char *_RESTRICT_KYWD s,
40            size_t n, locale_t loc)
41 {
42     static const mbstate_t initial = { 0 };
43     mbstate_t mbs;
44     const char *sp;

46     mbs = initial;
47     sp = s;
48     return (loc->ctype->lc_mbsnrtowcs(pwcs, &sp, ULONG_MAX, n, &mbs));
49 }

51 size_t
52 mbstowcs(wchar_t *_RESTRICT_KYWD pwcs, const char *_RESTRICT_KYWD s, size_t n)
53 {
54     return (mbstowcs_l(pwcs, s, n, uselocale(NULL)));
55     return (__mbsnrtowcs(pwcs, &sp, ULONG_MAX, n, &mbs));
56 }

    unchanged_portion_omitted
```



new/usr/src/lib/libc/port/locale/mbtowc.c

1

\*\*\*\*\*

2084 Fri Jul 11 11:43:00 2014

new/usr/src/lib/libc/port/locale/mbtowc.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <stdlib.h>
31 #include <wchar.h>
32 #include <locale.h>
33 #include <xlocale.h>
34 #include "mblocal.h"

36 int
37 mbtowc_l(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s, size_t n,
38          locale_t loc)
39 mbtowc(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s, size_t n)
40 {
41     mbstate_t mbs = { 0 };
42     static const mbstate_t initial = { 0 };
43     static mbstate_t mbs;
44     size_t rval;

45     if (s == NULL) {
46         /* No support for state dependent encodings. */
47         mbs = initial;
48         return (0);
49     }
50     rval = mbrtowc_l(pwc, s, n, &mbs, loc);
51     rval = __mbrtowc(pwc, s, n, &mbs);
52     if (rval == (size_t)-1 || rval == (size_t)-2)
53         return (-1);
54     return ((int)rval);
55 }
```

new/usr/src/lib/libc/port/locale/mbtowc.c

2

```
54 mbtowc(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s, size_t n)
55 {
56     return (mbtowc_l(pwc, s, n, uselocale(NULL)));
57 }
_____unchanged_portion_omitted_____
```

```

*****
5036 Fri Jul 11 11:43:00 2014
new/usr/src/lib/libc/port/locale/mskanji.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins. All rights reserved.
5  *
6  * ja_JP.SJIS locale table for BSD4.4/rune
7  * version 1.0
8  * (C) Sin'ichiro MIYATANI / Phase One, Inc
9  * May 12, 1995
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 * notice, this list of conditions and the following disclaimer in the
18 * documentation and/or other materials provided with the distribution.
19 * 3. All advertising materials mentioning features or use of this software
20 * must display the following acknowledgement:
21 * This product includes software developed by Phase One, Inc.
22 * 4. The name of Phase One, Inc. may be used to endorse or promote products
23 * derived from this software without specific prior written permission.
24 *
25 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
26 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
27 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
28 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
29 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
30 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
31 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
32 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
33 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
34 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
35 * SUCH DAMAGE.
36 */
37
38 #include "lint.h"
39 #include <sys/types.h>
40 #include <errno.h>
41 #include "runetype.h"
42 #include <stdlib.h>
43 #include <string.h>
44 #include <wchar.h>
45 #include "mblocal.h"
46
47 static size_t _MSKanji_mbrtowc(wchar_t *_RESTRICT_KYWD,
48 const char *_RESTRICT_KYWD,
49 size_t, mbstate_t *_RESTRICT_KYWD);
50 static int _MSKanji_mbsinit(const mbstate_t *);
51 static size_t _MSKanji_wrtomb(char *_RESTRICT_KYWD, wchar_t,
52 mbstate_t *_RESTRICT_KYWD);
53 static size_t _MSKanji_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
54 const char **_RESTRICT_KYWD, size_t, size_t,
55 mbstate_t *_RESTRICT_KYWD);
56 static size_t _MSKanji_wcsnrtombs(char *_RESTRICT_KYWD,
57 const wchar_t **_RESTRICT_KYWD, size_t, size_t,

```

```

58 mbstate_t *_RESTRICT_KYWD);
59
60 typedef struct {
61     wchar_t ch;
62 } _MSKanjiState;
63
64 void
65 _MSKanji_init(struct lc_ctype *lct)
66 {
67     lct->lc_mbrtowc = _MSKanji_mbrtowc;
68     lct->lc_wrtomb = _MSKanji_wrtomb;
69     lct->lc_mbsnrtowcs = _MSKanji_mbsnrtowcs;
70     lct->lc_wcsnrtombs = _MSKanji_wcsnrtombs;
71     lct->lc_mbsinit = _MSKanji_mbsinit;
72     lct->lc_max_mblen = 2;
73     lct->lc_is_ascii = 0;
74     _mbrtowc = _MSKanji_mbrtowc;
75     _wrtomb = _MSKanji_wrtomb;
76     _mbsinit = _MSKanji_mbsinit;
77     _CurrentRuneLocale = rl;
78     _ctype[520] = 2;
79     charset_is_ascii = 0;
80     return (0);
81 }
82
83 #ifndef _unchanged_portion_omitted
84
85 static size_t
86 _MSKanji_wrtomb(char *_RESTRICT_KYWD s, wchar_t wc,
87 mbstate_t *_RESTRICT_KYWD ps)
88 {
89     _MSKanjiState *ms;
90     int len, i;
91
92     ms = (_MSKanjiState *)ps;
93
94     if (ms->ch != 0) {
95         errno = EINVAL;
96         return ((size_t)-1);
97     }
98
99     if (s == NULL)
100         /* Reset to initial shift state (no-op) */
101         return (1);
102     len = (wc > 0x100) ? 2 : 1;
103     for (i = len; i-- > 0; )
104         *s++ = wc >> (i << 3);
105     return (len);
106 }
107
108 static size_t
109 _MSKanji_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst,
110 const char **_RESTRICT_KYWD src, size_t nms,
111 size_t len, mbstate_t *_RESTRICT_KYWD ps)
112 {
113     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _MSKanji_mbrtowc));
114 }
115
116 static size_t
117 _MSKanji_wcsnrtombs(char *_RESTRICT_KYWD dst,
118 const wchar_t **_RESTRICT_KYWD src, size_t nwc,
119 size_t len, mbstate_t *_RESTRICT_KYWD ps)
120 {
121     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _MSKanji_wrtomb));
122 }

```

new/usr/src/lib/libc/port/locale/mskanji.c

3

179 }

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/nextwctype.c

1

```
*****
3268 Fri Jul 11 11:43:00 2014
new/usr/src/lib/libc/port/locale/nextwctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 2004 Tim J. Robbins.
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #include "lint.h"
29 #include "runetype.h"
30 #include <wchar.h>
31 #include <wctype.h>
32 #include "localeimpl.h"

34 /*
35  * nextwctype, while exposed on *BSD/MacOS X, is considered "consolidation
36  * private" for illumos. Hence, we keep the _l version static for now.
37  * If we decide to make this public, just remove the static keyword and
38  * put it in the headers and mapfile. (Should fix up the underscore prefix
39  * to __nextwctype() as well.)
40  */
41 static wint_t
42 nextwctype_l(wint_t wc, wctype_t wct, locale_t loc)
43 {
44     size_t lim;
45     const _RuneLocale *rl;
46     const _RuneRange *rr;
47     const _RuneEntry *base, *re;
48     _RuneRange *rr_ext = &_CurrentRuneLocale->__runetype_ext;
49     _RuneEntry *base_ext, *re_ext;
50     int noinc;

51     rl = loc->runelocale;
52     rr = &rl->__runetype_ext;

53     noinc = 0;
54     if (wc < _CACHED_RUNES) {
```

new/usr/src/lib/libc/port/locale/nextwctype.c

2

```
55         wc++;
56         while (wc < _CACHED_RUNES) {
57             if (rl->__runetype[wc] & wct)
58                 if (_CurrentRuneLocale->__runetype[wc] & wct)
59                     return (wc);
60         }
61         wc--;
62     }
63     if (rr->__ranges != NULL && wc < rr->__ranges[0].__min) {
64         wc = rr->__ranges[0].__min;
65         noinc = 1;
66     }

68     /* Binary search -- see bsearch.c for explanation. */
69     base = rr->__ranges;
70     for (lim = rr->__nranges; lim != 0; lim >>= 1) {
71         re = base + (lim >> 1);
72         if (re->__min <= wc && wc <= re->__max)
73             goto found;
74         else if (wc > re->__max) {
75             base = re + 1;
76             lim--;
77         }
78     }
79     return (-1);
80 found:
81     if (!noinc)
82         wc++;
83     if (re->__min <= wc && wc <= re->__max) {
84         if (re->__types != NULL) {
85             for (; wc <= re->__max; wc++)
86                 if (re->__types[wc - re->__min] & wct)
87                     return (wc);
88             } else if (re->__map & wct)
89                 return (wc);
90     }
91     while (++re < rr->__ranges + rr->__nranges) {
92         wc = re->__min;
93         if (re->__types != NULL) {
94             for (; wc <= re->__max; wc++)
95                 if (re->__types[wc - re->__min] & wct)
96                     return (wc);
97             } else if (re->__map & wct)
98                 return (wc);
99     }
100     return (-1);
101 }

103 /*
104  * External, but consolidation private routine.
105  */
106 wint_t
107 __nextwctype(wint_t wc, wctype_t wct)
108 {
109     return (nextwctype_l(wc, wct, uselocale(NULL)));
110 }
111 }
112 }
113 }
114 }
115 }
116 }
117 }
118 }
119 }
120 }
121 }
122 }
123 }
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133 }
134 }
135 }
136 }
137 }
138 }
139 }
140 }
141 }
142 }
143 }
144 }
145 }
146 }
147 }
148 }
149 }
150 }
151 }
152 }
153 }
154 }
155 }
156 }
157 }
158 }
159 }
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
201 }
202 }
203 }
204 }
205 }
206 }
207 }
208 }
209 }
210 }
211 }
212 }
213 }
214 }
215 }
216 }
217 }
218 }
219 }
220 }
221 }
222 }
223 }
224 }
225 }
226 }
227 }
228 }
229 }
230 }
231 }
232 }
233 }
234 }
235 }
236 }
237 }
238 }
239 }
240 }
241 }
242 }
243 }
244 }
245 }
246 }
247 }
248 }
249 }
250 }
251 }
252 }
253 }
254 }
255 }
256 }
257 }
258 }
259 }
260 }
261 }
262 }
263 }
264 }
265 }
266 }
267 }
268 }
269 }
270 }
271 }
272 }
273 }
274 }
275 }
276 }
277 }
278 }
279 }
280 }
281 }
282 }
283 }
284 }
285 }
286 }
287 }
288 }
289 }
290 }
291 }
292 }
293 }
294 }
295 }
296 }
297 }
298 }
299 }
300 }
301 }
302 }
303 }
304 }
305 }
306 }
307 }
308 }
309 }
310 }
311 }
312 }
313 }
314 }
315 }
316 }
317 }
318 }
319 }
320 }
321 }
322 }
323 }
324 }
325 }
326 }
327 }
328 }
329 }
330 }
331 }
332 }
333 }
334 }
335 }
336 }
337 }
338 }
339 }
340 }
341 }
342 }
343 }
344 }
345 }
346 }
347 }
348 }
349 }
350 }
351 }
352 }
353 }
354 }
355 }
356 }
357 }
358 }
359 }
360 }
361 }
362 }
363 }
364 }
365 }
366 }
367 }
368 }
369 }
370 }
371 }
372 }
373 }
374 }
375 }
376 }
377 }
378 }
379 }
380 }
381 }
382 }
383 }
384 }
385 }
386 }
387 }
388 }
389 }
390 }
391 }
392 }
393 }
394 }
395 }
396 }
397 }
398 }
399 }
400 }
401 }
402 }
403 }
404 }
405 }
406 }
407 }
408 }
409 }
410 }
411 }
412 }
413 }
414 }
415 }
416 }
417 }
418 }
419 }
420 }
421 }
422 }
423 }
424 }
425 }
426 }
427 }
428 }
429 }
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }
454 }
455 }
456 }
457 }
458 }
459 }
460 }
461 }
462 }
463 }
464 }
465 }
466 }
467 }
468 }
469 }
470 }
471 }
472 }
473 }
474 }
475 }
476 }
477 }
478 }
479 }
480 }
481 }
482 }
483 }
484 }
485 }
486 }
487 }
488 }
489 }
490 }
491 }
492 }
493 }
494 }
495 }
496 }
497 }
498 }
499 }
500 }
501 }
502 }
503 }
504 }
505 }
506 }
507 }
508 }
509 }
510 }
511 }
512 }
513 }
514 }
515 }
516 }
517 }
518 }
519 }
520 }
521 }
522 }
523 }
524 }
525 }
526 }
527 }
528 }
529 }
530 }
531 }
532 }
533 }
534 }
535 }
536 }
537 }
538 }
539 }
540 }
541 }
542 }
543 }
544 }
545 }
546 }
547 }
548 }
549 }
550 }
551 }
552 }
553 }
554 }
555 }
556 }
557 }
558 }
559 }
560 }
561 }
562 }
563 }
564 }
565 }
566 }
567 }
568 }
569 }
570 }
571 }
572 }
573 }
574 }
575 }
576 }
577 }
578 }
579 }
580 }
581 }
582 }
583 }
584 }
585 }
586 }
587 }
588 }
589 }
590 }
591 }
592 }
593 }
594 }
595 }
596 }
597 }
598 }
599 }
600 }
601 }
602 }
603 }
604 }
605 }
606 }
607 }
608 }
609 }
610 }
611 }
612 }
613 }
614 }
615 }
616 }
617 }
618 }
619 }
620 }
621 }
622 }
623 }
624 }
625 }
626 }
627 }
628 }
629 }
630 }
631 }
632 }
633 }
634 }
635 }
636 }
637 }
638 }
639 }
640 }
641 }
642 }
643 }
644 }
645 }
646 }
647 }
648 }
649 }
650 }
651 }
652 }
653 }
654 }
655 }
656 }
657 }
658 }
659 }
660 }
661 }
662 }
663 }
664 }
665 }
666 }
667 }
668 }
669 }
670 }
671 }
672 }
673 }
674 }
675 }
676 }
677 }
678 }
679 }
680 }
681 }
682 }
683 }
684 }
685 }
686 }
687 }
688 }
689 }
690 }
691 }
692 }
693 }
694 }
695 }
696 }
697 }
698 }
699 }
700 }
701 }
702 }
703 }
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }
721 }
722 }
723 }
724 }
725 }
726 }
727 }
728 }
729 }
730 }
731 }
732 }
733 }
734 }
735 }
736 }
737 }
738 }
739 }
740 }
741 }
742 }
743 }
744 }
745 }
746 }
747 }
748 }
749 }
750 }
751 }
752 }
753 }
754 }
755 }
756 }
757 }
758 }
759 }
760 }
761 }
762 }
763 }
764 }
765 }
766 }
767 }
768 }
769 }
770 }
771 }
772 }
773 }
774 }
775 }
776 }
777 }
778 }
779 }
780 }
781 }
782 }
783 }
784 }
785 }
786 }
787 }
788 }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799 }
800 }
801 }
802 }
803 }
804 }
805 }
806 }
807 }
808 }
809 }
810 }
811 }
812 }
813 }
814 }
815 }
816 }
817 }
818 }
819 }
820 }
821 }
822 }
823 }
824 }
825 }
826 }
827 }
828 }
829 }
830 }
831 }
832 }
833 }
834 }
835 }
836 }
837 }
838 }
839 }
840 }
841 }
842 }
843 }
844 }
845 }
846 }
847 }
848 }
849 }
850 }
851 }
852 }
853 }
854 }
855 }
856 }
857 }
858 }
859 }
860 }
861 }
862 }
863 }
864 }
865 }
866 }
867 }
868 }
869 }
870 }
871 }
872 }
873 }
874 }
875 }
876 }
877 }
878 }
879 }
880 }
881 }
882 }
883 }
884 }
885 }
886 }
887 }
888 }
889 }
890 }
891 }
892 }
893 }
894 }
895 }
896 }
897 }
898 }
899 }
900 }
901 }
902 }
903 }
904 }
905 }
906 }
907 }
908 }
909 }
910 }
911 }
912 }
913 }
914 }
915 }
916 }
917 }
918 }
919 }
920 }
921 }
922 }
923 }
924 }
925 }
926 }
927 }
928 }
929 }
930 }
931 }
932 }
933 }
934 }
935 }
936 }
937 }
938 }
939 }
940 }
941 }
942 }
943 }
944 }
945 }
946 }
947 }
948 }
949 }
950 }
951 }
952 }
953 }
954 }
955 }
956 }
957 }
958 }
959 }
960 }
961 }
962 }
963 }
964 }
965 }
966 }
967 }
968 }
969 }
970 }
971 }
972 }
973 }
974 }
975 }
976 }
977 }
978 }
979 }
980 }
981 }
982 }
983 }
984 }
985 }
986 }
987 }
988 }
989 }
990 }
991 }
992 }
993 }
994 }
995 }
996 }
997 }
998 }
999 }
1000 }
```

```

*****
5181 Fri Jul 11 11:43:00 2014
new/usr/src/lib/libc/port/locale/nl_langinfo.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2001, 2003 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <langinfo.h>
31 #include <limits.h>
32 #include <locale.h>
33 #include <stdlib.h>
34 #include <string.h>

36 #include "lnumeric.h"
37 #include "lmessages.h"
38 #include "lmonetary.h"
39 #include "timelocal.h"
40 #include "localeimpl.h"

42 #define _REL(BASE) ((int)item-BASE)

42 #define MONETARY      (__get_current_monetary_locale())
43 #define TIME          (__get_current_time_locale())
44 #define MESSAGES     (__get_current_messages_locale())
45 #define NUMERIC      (__get_current_numeric_locale())

44 #pragma weak _nl_langinfo = _nl_langinfo

46 char *
47 _nl_langinfo_l(nl_item item, locale_t loc)
48 {
49     char *ret, *s, *cs;
50     struct locdata *ldata;
51     const struct lc_monetary *lmon = loc->monetary;
52     const struct lc_numeric *lnum = loc->numeric;

```

```

53     const struct lc_messages *lmsgs = loc->messages;
54     const struct lc_time *ltime = loc->time;
55     static char *csym = NULL;

56     switch (item) {
57     case CODESET:
58         ret = "";
59         /*
60          * The codeset is the suffix of a locale, for most it will
61          * will be UTF-8, as in "en_US.UTF-8". Short form locales are
62          * will be UTF-8, as in "en.UTF-8". Short form locales are
63          * not supported. Note also that although FreeBSD uses
64          * US-ASCII, Solaris historically has reported "646" for the
65          * C locale.
66          *
67          * Note that this code will need to change if we ever support
68          * POSIX defined locale variants (suffixes with an @ sign)
69          */
70         ldata = loc->locdata[LC_CTYPE];
71         s = ldata ? ldata->l_lname : NULL;
72         if (s != NULL) {
73             if ((s = setlocale(LC_CTYPE, NULL)) != NULL) {
74                 if ((cs = strchr(s, '.')) != NULL)
75                     ret = cs + 1;
76                 else if (strcmp(s, "C") == 0 || strcmp(s, "POSIX") == 0)
77                     ret = "646";
78             }
79             break;
80         case D_T_FMT:
81             ret = (char *)ltime->c_fmt;
82             ret = (char *)TIME->c_fmt;
83             break;
84         case D_FMT:
85             ret = (char *)ltime->x_fmt;
86             ret = (char *)TIME->x_fmt;
87             break;
88         case T_FMT:
89             ret = (char *)ltime->X_fmt;
90             ret = (char *)TIME->X_fmt;
91             break;
92         case T_FMT_AMPM:
93             ret = (char *)ltime->ampm_fmt;
94             ret = (char *)TIME->ampm_fmt;
95             break;
96         case AM_STR:
97             ret = (char *)ltime->am;
98             ret = (char *)TIME->am;
99             break;
100        case PM_STR:
101            ret = (char *)ltime->pm;
102            ret = (char *)TIME->pm;
103            break;
104        case DAY_1: case DAY_2: case DAY_3:
105        case DAY_4: case DAY_5: case DAY_6: case DAY_7:
106            ret = (char *)ltime->weekday[_REL(DAY_1)];
107            ret = (char *)TIME->weekday[_REL(DAY_1)];
108            break;
109        case ABDAY_1: case ABDAY_2: case ABDAY_3:
110        case ABDAY_4: case ABDAY_5: case ABDAY_6: case ABDAY_7:
111            ret = (char *)ltime->wday[_REL(ABDAY_1)];
112            ret = (char *)TIME->wday[_REL(ABDAY_1)];
113            break;
114        case MON_1: case MON_2: case MON_3: case MON_4:
115        case MON_5: case MON_6: case MON_7: case MON_8:
116        case MON_9: case MON_10: case MON_11: case MON_12:
117            ret = (char *)ltime->month[_REL(MON_1)];

```

```

101         ret = (char *)TIME->month[_REL(MON_1)];
102         break;
103     case ABMON_1: case ABMON_2: case ABMON_3: case ABMON_4:
104     case ABMON_5: case ABMON_6: case ABMON_7: case ABMON_8:
105     case ABMON_9: case ABMON_10: case ABMON_11: case ABMON_12:
106         ret = (char *)ltime->mon[_REL(ABMON_1)];
107         ret = (char *)TIME->mon[_REL(ABMON_1)];
108         break;
109     case ERA:
110         /* XXX: need to be implemented */
111         ret = "";
112         break;
113     case ERA_D_FMT:
114         /* XXX: need to be implemented */
115         ret = "";
116         break;
117     case ERA_D_T_FMT:
118         /* XXX: need to be implemented */
119         ret = "";
120         break;
121     case ERA_T_FMT:
122         /* XXX: need to be implemented */
123         ret = "";
124         break;
125     case ALT_DIGITS:
126         /* XXX: need to be implemented */
127         ret = "";
128         break;
129     case RADIXCHAR:
130         ret = (char *)lnum->decimal_point;
131         ret = (char *)NUMERIC->decimal_point;
132         break;
133     case THOUSEP:
134         ret = (char *)lnum->thousands_sep;
135         ret = (char *)NUMERIC->thousands_sep;
136         break;
137     case YESEXPR:
138         ret = (char *)lmsgs->yesexpr;
139         ret = (char *)MESSAGES->yesexpr;
140         break;
141     case NOEXPR:
142         ret = (char *)lmsgs->noexpr;
143         ret = (char *)MESSAGES->noexpr;
144         break;
145     /*
146     * YESSTR and NOSTR items were removed from Issue 7. But
147     * older applications might still need them. Their use is
148     * discouraged.
149     * YESSTR and NOSTR items marked with LEGACY are available, but not
150     * recommended by SUSv2 to be used in portable applications since
151     * they're subject to remove in future specification editions.
152     */
153     case YESSTR: /* LEGACY */
154         ret = (char *)lmsgs->yesstr;
155         ret = (char *)MESSAGES->yesstr;
156         break;
157     case NOSTR: /* LEGACY */
158         ret = (char *)lmsgs->nostr;
159         ret = (char *)MESSAGES->nostr;
160         break;
161     /*
162     * SUSv2 special formatted currency string
163     */
164     case CRNCYSTR:
165         ret = lmon->crncystr;
166         break;

```

```

155         ret = "";
156         cs = (char *)MONETARY->currency_symbol;
157         if (*cs != '\0') {
158             char pos = localeconv()->p_cs_precedes;
159
160             if (pos == localeconv()->n_cs_precedes) {
161                 char psn = '\0';
162
163                 if (pos == CHAR_MAX) {
164                     if (strcmp(cs,
165                             MONETARY->mon_decimal_point) == 0)
166                         psn = '.';
167                 } else
168                     psn = pos ? '-' : '+';
169                 if (psn != '\0') {
170                     int clen = strlen(cs);
171                     char *newc;
172
173                     newc = realloc(csym, clen + 2);
174                     if (newc != NULL) {
175                         free(csym);
176                         csym = newc;
177                         *csym = psn;
178                         (void) strcpy(csym + 1, cs);
179                         ret = csym;
180                     }
181                 }
182             }
183         }
184         break;
185     case _DATE_FMT: /* Solaris specific extension */
186         ret = (char *)ltime->date_fmt;
187         ret = (char *)TIME->date_fmt;
188         break;
189     /*
190     * Note that FreeBSD also had a private D_MD_ORDER, but that appears
191     * to have been specific to FreeBSD, so we have not included it here.
192     */
193     default:
194         ret = "";
195     }
196     return (ret);
197 }
198
199 char *
200 nl_langinfo(nl_item item)
201 {
202     return (nl_langinfo_l(item, uselocale(NULL)));
203 }
204
205 _____unchanged_portion_omitted_____

```

new/usr/src/lib/libc/port/locale/none.c

1

```
*****
4427 Fri Jul 11 11:43:00 2014
new/usr/src/lib/libc/port/locale/none.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins. All rights reserved.
5  * Copyright (c) 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * This code is derived from software contributed to Berkeley by
9  * Paul Borman at Krystal Technologies.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 * notice, this list of conditions and the following disclaimer in the
18 * documentation and/or other materials provided with the distribution.
19 * 4. Neither the name of the University nor the names of its contributors
20 * may be used to endorse or promote products derived from this software
21 * without specific prior written permission.
22 *
23 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
24 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
25 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
26 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
27 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
28 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
29 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
30 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
31 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
32 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
33 * SUCH DAMAGE.
34 */
36 #include "lint.h"
37 #include <errno.h>
38 #include <limits.h>
39 #include <stddef.h>
40 #include <stdio.h>
41 #include <stdlib.h>
42 #include <string.h>
43 #include <wchar.h>
44 #include <note.h>
44 #include "runetype.h"
45 #include "mblocal.h"
46 #include "lctype.h"
47 static size_t _none_mbrtowc(wchar_t *_RESTRICT_KYWD,
48     const char *_RESTRICT_KYWD, size_t, mbstate_t *_RESTRICT_KYWD);
50 static int _none_mbsinit(const mbstate_t *);
51 static size_t _none_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst,
52     const char **_RESTRICT_KYWD src, size_t nms, size_t len,
53     mbstate_t *_RESTRICT_KYWD);
54 static size_t _none_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
55     mbstate_t *_RESTRICT_KYWD);
56 static size_t _none_wcsnrtombs(char *_RESTRICT_KYWD,
```

new/usr/src/lib/libc/port/locale/none.c

2

```
57     const wchar_t **_RESTRICT_KYWD,
58     size_t, size_t, mbstate_t *_RESTRICT_KYWD);
48 /* setup defaults */
50 void
51 _none_init(struct lc_ctype *lct)
52 int
53 _none_init(_RuneLocale *rl)
52 {
53     lct->lc_is_ascii = 1;
54     lct->lc_mbrtowc = __mbrtowc_ascii;
55     lct->lc_mbsinit = __mbsinit_ascii;
56     lct->lc_mbsnrtowcs = __mbsnrtowcs_ascii;
57     lct->lc_wcrtomb = __wcrtomb_ascii;
58     lct->lc_wcsnrtombs = __wcsnrtombs_ascii;
59     lct->lc_max_mblen = 1;
60     charset_is_ascii = 1;
62     __mbrtowc = _none_mbrtowc;
63     __mbsinit = _none_mbsinit;
64     __mbsnrtowcs = _none_mbsnrtowcs;
65     __wcrtomb = _none_wcrtomb;
66     __wcsnrtombs = _none_wcsnrtombs;
67     _CurrentRuneLocale = rl;
68     return (0);
69 }
70 int
71 _mbsinit_ascii(const mbstate_t *_unused)
72 static int
73 _none_mbsinit(const mbstate_t *_unused)
74 {
75     _NOTE(ARGUNUSED(unused));
76 }
77 /*
78  * Encoding is not state dependent - we are always in the
79  * initial state.
80  */
81 return (1);
82 }
83 size_t
84 __mbrtowc_ascii(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s,
85     static size_t
86     _none_mbrtowc(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s,
87     size_t n, mbstate_t *_RESTRICT_KYWD unused)
88 {
89     _NOTE(ARGUNUSED(unused));
90 }
91 if (s == NULL)
92     /* Reset to initial shift state (no-op) */
93     return (0);
94 if (n == 0)
95     /* Incomplete multibyte sequence */
96     return ((size_t)-2);
97 if (pwc != NULL)
98     *pwc = (unsigned char)*s;
99 return (*s == '\0' ? 0 : 1);
100 }
101 size_t
102 __wcrtomb_ascii(char *_RESTRICT_KYWD s, wchar_t wc,
103     static size_t
104     _none_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc,
105     mbstate_t *_RESTRICT_KYWD unused)
```

```

94 {
95     _NOTE(ARGUNUSED(unused));

97     if (s == NULL)
98         /* Reset to initial shift state (no-op) */
99         return (1);
100     if (wc < 0 || wc > UCHAR_MAX) {
101         errno = EILSEQ;
102         return ((size_t)-1);
103     }
104     *s = (unsigned char)wc;
105     return (1);
106 }

108 size_t
109 __mbsnrtowcs_ascii(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
122 static size_t
123 _none_mbsnrtowcs(wchar_t *_RESTRICT_KYWD dst, const char **_RESTRICT_KYWD src,
110     size_t nms, size_t len, mbstate_t *_RESTRICT_KYWD unused)
111 {
112     const char *s;
113     size_t nchr;

115     _NOTE(ARGUNUSED(unused));

117     if (dst == NULL) {
118         s = memchr(*src, '\\0', nms);
119         return (s != NULL ? s - *src : nms);
120     }

122     s = *src;
123     nchr = 0;
124     while (len-- > 0 && nms-- > 0) {
125         if ((*dst++ = (unsigned char)*s++) == L'\\0') {
126             *src = NULL;
127             return (nchr);
128         }
129         nchr++;
130     }
131     *src = s;
132     return (nchr);
133 }

135 size_t
136 __wcsnrtombs_ascii(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
149 static size_t
150 _none_wcsnrtombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
137     size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD unused)
138 {
139     const wchar_t *s;
140     size_t nchr;

142     _NOTE(ARGUNUSED(unused));

144     if (dst == NULL) {
145         for (s = *src; nwc > 0 && *s != L'\\0'; s++, nwc--) {
146             if (*s < 0 || *s > UCHAR_MAX) {
147                 errno = EILSEQ;
148                 return ((size_t)-1);
149             }
150         }
151         return (s - *src);
152     }

154     s = *src;
155     nchr = 0;

```

```

156     while (len-- > 0 && nwc-- > 0) {
157         if (*s < 0 || *s > UCHAR_MAX) {
158             errno = EILSEQ;
159             return ((size_t)-1);
160         }
161         if ((*dst++ = *s++) == '\\0') {
162             *src = NULL;
163             return (nchr);
164         }
165         nchr++;
166     }
167     *src = s;
168     return (nchr);
169 }

185 /* setup defaults */

187 size_t (*__mbrtowc)(wchar_t *_RESTRICT_KYWD, const char *_RESTRICT_KYWD,
188     size_t, mbstate_t *_RESTRICT_KYWD) = _none_mbrtowc;

190 int (*__mbsinit)(const mbstate_t *) = _none_mbsinit;

192 size_t (*__mbsnrtowcs)(wchar_t *_RESTRICT_KYWD, const char **_RESTRICT_KYWD,
193     size_t, size_t, mbstate_t *_RESTRICT_KYWD) = _none_mbsnrtowcs;

195 size_t (*__wctomb)(char *_RESTRICT_KYWD, wchar_t, mbstate_t *_RESTRICT_KYWD) =
196     _none_wctomb;

198 size_t (*__wcsnrtombs)(char *_RESTRICT_KYWD, const wchar_t **_RESTRICT_KYWD,
199     size_t, size_t, mbstate_t *_RESTRICT_KYWD) = _none_wcsnrtombs;

```



```

*****
41502 Fri Jul 11 11:43:01 2014
new/usr/src/lib/libc/port/locale/regcomp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright 2012 Milan Jurik. All rights reserved.
5  * Copyright (c) 1992, 1993, 1994 Henry Spencer.
6  * Copyright (c) 1992, 1993, 1994
7  *   The Regents of the University of California. All rights reserved.
8  *
9  * This code is derived from software contributed to Berkeley by
10 * Henry Spencer.
11 *
12 * Redistribution and use in source and binary forms, with or without
13 * modification, are permitted provided that the following conditions
14 * are met:
15 * 1. Redistributions of source code must retain the above copyright
16 * notice, this list of conditions and the following disclaimer.
17 * 2. Redistributions in binary form must reproduce the above copyright
18 * notice, this list of conditions and the following disclaimer in the
19 * documentation and/or other materials provided with the distribution.
20 * 4. Neither the name of the University nor the names of its contributors
21 * may be used to endorse or promote products derived from this software
22 * without specific prior written permission.
23 *
24 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
25 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
26 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
27 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
28 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
29 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
30 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
31 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
32 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
33 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
34 * SUCH DAMAGE.
35 */
37 #include "lint.h"
38 #include "file64.h"
39 #include <sys/types.h>
40 #include <stdio.h>
41 #include <string.h>
42 #include <ctype.h>
43 #include <limits.h>
44 #include <stdlib.h>
45 #include <regex.h>
46 #include <wchar.h>
47 #include <wctype.h>
49 #include "runetype.h"
50 #include "collate.h"
52 #include "utils.h"
53 #include "regex2.h"
55 #include "cname.h"
56 #include "mblocal.h"
58 /*

```

```

59 * parse structure, passed up and down to avoid global variables and
60 * other clumsinesses
61 */
62 struct parse {
63     char *next;           /* next character in RE */
64     char *end;           /* end of string (-> NUL normally) */
65     int error;           /* has an error been seen? */
66     sop *strip;          /* malloced strip */
67     sopno ssize;         /* malloced strip size (allocated) */
68     sopno slen;          /* malloced strip length (used) */
69     int ncsalloc;        /* number of csets allocated */
70     struct re_guts *g;
71 #define NPAREN 10        /* we need to remember () 1-9 for back refs */
72     sopno pbegin[NPAREN]; /* -> ( ([0] unused) */
73     sopno pend[NPAREN];  /* -> ) ([0] unused) */
74 };
75
76 unchanged_portion_omitted
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120 /*
121 * p_b_term - parse one term of a bracketed character list
122 */
123 static void
124 p_b_term(struct parse *p, cset *cs)
125 {
126     char c;
127     wint_t start, finish;
128     wint_t i;
129     locale_t loc = uselocale(NULL);
130
131     /* classify what we've got */
132     switch ((MORE()) ? PEEK() : '\0') {
133     case '[':
134         c = (MORE2()) ? PEEK2() : '\0';
135         break;
136     case '-':
137         SETERROR(REG_ERANGE);
138         return; /* NOTE RETURN */
139     default:
140         c = '\0';
141         break;
142     }
143
144     switch (c) {
145     case ':': /* character class */
146         NEXT2();
147         (void) REQUIRE(MORE(), REG_EBRACK);
148         c = PEEK();
149         (void) REQUIRE(c != '-' && c != ']', REG_ECTYPE);
150         p_b_cclass(p, cs);
151         (void) REQUIRE(MORE(), REG_EBRACK);
152         (void) REQUIRE(EATTWO(':', ']'), REG_ECTYPE);
153         break;
154     case '=': /* equivalence class */
155         NEXT2();
156         (void) REQUIRE(MORE(), REG_EBRACK);
157         c = PEEK();
158         (void) REQUIRE(c != '-' && c != ']', REG_ECOLLATE);
159         p_b_ecclass(p, cs);
160         (void) REQUIRE(MORE(), REG_EBRACK);
161         (void) REQUIRE(EATTWO('=', ']'), REG_ECOLLATE);
162         break;
163     default: /* symbol, ordinary character, or range */
164         start = p_b_symbol(p);
165         if (SEE('-') && MORE2() && PEEK2() != ']') {
166             /* range */
167             NEXT();

```

```

768         if (EAT('-'))
769             finish = '-';
770     else
771         finish = p_b_symbol(p);
772 } else
773     finish = start;
774 if (start == finish)
775     CHadd(p, cs, start);
776 else {
777     if (loc->collate->lc_is_posix) {
778         if (!_collate_load_error) {
779             (void) REQUIRE((uch)start <= (uch)finish,
780                 REG_ERANGE);
781             CHaddrange(p, cs, start, finish);
782         } else {
783             (void) REQUIRE(_collate_range_cmp(start,
784                 finish, loc) <= 0, REG_ERANGE);
785             (void) REQUIRE(_collate_range_cmp(i, finish, loc)
786                 <= 0 &&
787                 _collate_range_cmp(i, finish, loc)
788                 <= 0);
789             if (_collate_range_cmp(start, i) <= 0 &&
790                 _collate_range_cmp(i, finish) <= 0)
791                 CHadd(p, cs, i);
792         }
793     }
794 }
795 }

```

unchanged\_portion\_omitted

```

1352 /*
1353 * findmust - fill in must and mlen with longest mandatory literal string
1354 *
1355 * This algorithm could do fancy things like analyzing the operands of |
1356 * for common subsequences. Someday. This code is simple and finds most
1357 * of the interesting cases.
1358 *
1359 * Note that must and mlen got initialized during setup.
1360 */
1361 static void
1362 findmust(struct parse *p, struct re_guts *g)
1363 {
1364     sop *scan;
1365     sop *start;
1366     sop *newstart;
1367     sopno newlen;
1368     sop s;
1369     char *cp;
1370     int offset;
1371     char buf[MB_LEN_MAX];
1372     size_t clen;
1373     mbstate_t mbs;
1374     locale_t loc = uselocale(NULL);

```

/\* avoid making error situations worse \*/

```

1376     if (p->error != 0)
1377         return;

```

/\*

```

1380     * It's not generally safe to do a 'char' substring search on
1381     * multibyte character strings, but it's safe for at least
1382     * UTF-8 (see RFC 3629).
1383

```

```

1384     /*
1385     if (MB_CUR_MAX > 1 &&
1386         strcmp(loc->runelocale->__encoding, "UTF-8") != 0)
1387         strcmp(_CurrentRuneLocale->__encoding, "UTF-8") != 0)
1388             return;

```

/\* find the longest OCHAR sequence in strip \*/

```

1389     newlen = 0;
1390     offset = 0;
1391     g->moffset = 0;
1392     scan = g->strip + 1;
1393     do {
1394         s = *scan++;
1395         switch (OP(s)) {
1396             case OCHAR: /* sequence member */
1397                 if (newlen == 0) { /* new sequence */
1398                     (void) memset(&mbs, 0, sizeof(mbs));
1399                     newstart = scan - 1;
1400                 }
1401                 clen = wctomb(buf, OP(s), &mbs);
1402                 if (clen == (size_t)-1)
1403                     goto toohard;
1404                 newlen += clen;
1405                 break;
1406             case OPLUS: /* things that don't break one */
1407             case OLPAREN:
1408             case ORPAREN:
1409                 break;
1410             case OQUEST: /* things that must be skipped */
1411             case OCH:
1412                 offset = altoffset(scan, offset);
1413                 scan--;
1414                 do {
1415                     scan += OPND(s);
1416                     s = *scan;
1417                     /* assert() interferes w debug printouts */
1418                     if (OP(s) != O_QUEST && OP(s) != O_CH &&
1419                         OP(s) != OOR2) {
1420                         g->iflags |= BAD;
1421                         return;
1422                     }
1423                 } while (OP(s) != O_QUEST && OP(s) != O_CH);
1424                 /* FALLTHROUGH */
1425             case OBOW: /* things that break a sequence */
1426             case OEOW:
1427             case OBOL:
1428             case OEOL:
1429             case O_QUEST:
1430             case O_CH:
1431             case OEND:
1432                 if (newlen > g->mlen) { /* ends one */
1433                     start = newstart;
1434                     g->mlen = newlen;
1435                     if (offset > -1) {
1436                         g->moffset += offset;
1437                         offset = newlen;
1438                     } else
1439                         g->moffset = offset;
1440                 } else {
1441                     if (offset > -1)
1442                         offset += newlen;
1443                 }
1444                 newlen = 0;
1445                 break;
1446             case OANY:
1447                 if (newlen > g->mlen) { /* ends one */

```

```

1449         start = newstart;
1450         g->mrlen = newlen;
1451         if (offset > -1) {
1452             g->moffset += offset;
1453             offset = newlen;
1454         } else
1455             g->moffset = offset;
1456     } else {
1457         if (offset > -1)
1458             offset += newlen;
1459     }
1460     if (offset > -1)
1461         offset++;
1462     newlen = 0;
1463     break;
1464 case OANYOF: /* may or may not invalidate offset */
1465     /* First, everything as OANY */
1466     if (newlen > g->mrlen) { /* ends one */
1467         start = newstart;
1468         g->mrlen = newlen;
1469         if (offset > -1) {
1470             g->moffset += offset;
1471             offset = newlen;
1472         } else
1473             g->moffset = offset;
1474     } else {
1475         if (offset > -1)
1476             offset += newlen;
1477     }
1478     if (offset > -1)
1479         offset++;
1480     newlen = 0;
1481     break;
1482 toohard:
1483 default:
1484     /*
1485     * Anything here makes it impossible or too hard
1486     * to calculate the offset -- so we give up;
1487     * save the last known good offset, in case the
1488     * must sequence doesn't occur later.
1489     */
1490     if (newlen > g->mrlen) { /* ends one */
1491         start = newstart;
1492         g->mrlen = newlen;
1493         if (offset > -1)
1494             g->moffset += offset;
1495         else
1496             g->moffset = offset;
1497     }
1498     offset = -1;
1499     newlen = 0;
1500     break;
1501 }
1502 } while (OP(s) != OEND);

1504 if (g->mrlen == 0) { /* there isn't one */
1505     g->moffset = -1;
1506     return;
1507 }

1509 /* turn it into a character string */
1510 g->must = malloc((size_t)g->mrlen + 1);
1511 if (g->must == NULL) { /* argh; just forget it */
1512     g->mrlen = 0;
1513     g->moffset = -1;
1514     return;

```

```

1515     }
1516     cp = g->must;
1517     scan = start;
1518     (void) memset(&mbs, 0, sizeof (mbs));
1519     while (cp < g->must + g->mrlen) {
1520         while (OP(s = *scan++) != OCHAR)
1521             continue;
1522         clen = wcrctomb(cp, OPND(s), &mbs);
1523         assert(clen != (size_t)-1);
1524         cp += clen;
1525     }
1526     assert(cp == g->must + g->mrlen);
1527     *cp++ = '\\0'; /* just on general principles */
1528 }
_____ unchanged_portion_omitted_

```

new/usr/src/lib/libc/port/locale/rune.c

1

```
*****
6853 Fri Jul 11 11:43:01 2014
new/usr/src/lib/libc/port/locale/rune.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1993
5  *   The Regents of the University of California. All rights reserved.
6  *
7  * This code is derived from software contributed to Berkeley by
8  * Paul Borman at Krystal Technologies.
9  *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 * notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 * notice, this list of conditions and the following disclaimer in the
17 * documentation and/or other materials provided with the distribution.
18 * 4. Neither the name of the University nor the names of its contributors
19 * may be used to endorse or promote products derived from this software
20 * without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */

35 #include "lint.h"
36 #include "file64.h"
37 #include <errno.h>
38 #include <stdio.h>
39 #include <string.h>
40 #include <stdlib.h>
41 #include <sys/types.h>
42 #include <sys/stat.h>
43 #include <sys/mman.h>
44 #include <fcntl.h>
45 #include <unistd.h>

47 #include "libc.h"
48 #include "runetype.h"
49 #include "runefile.h"

46 _RuneLocale *_Read_RuneMagi(FILE *);

51 _RuneLocale *
52 _Read_RuneMagi(const char *fname)
53 _Read_RuneMagi(FILE *fp)
54 {
55     char *fdata, *data;
56     void *lastp;
```

new/usr/src/lib/libc/port/locale/rune.c

2

```
56     _FileRuneLocale *frl;
57     _RuneLocale *rl;
58     _FileRuneEntry *frr;
59     _RuneEntry *rr;
60     struct stat sb;
61     int x, saverr;
62     void *variable;
63     _FileRuneEntry *runetype_ext_ranges;
64     _FileRuneEntry *maplower_ext_ranges;
65     _FileRuneEntry *mapupper_ext_ranges;
66     int runetype_ext_len = 0;
67     int fd;

69     if ((fd = open(fname, O_RDONLY)) < 0) {
70         errno = EINVAL;
65     if (fstat(fileno(fp), &sb) < 0)
71         return (NULL);
72     }

74     if (fstat(fd, &sb) < 0) {
75         (void) close(fd);
68     if ((size_t)sb.st_size < sizeof (_FileRuneLocale)) {
76         errno = EINVAL;
77         return (NULL);
78     }

80     if ((size_t)sb.st_size < sizeof (_FileRuneLocale)) {
81         (void) close(fd);
82         errno = EINVAL;
73     if ((fdata = malloc(sb.st_size)) == NULL)
83         return (NULL);

76     errno = 0;
77     rewind(fp); /* Someone might have read the magic number once already */
78     if (errno) {
79         saverr = errno;
80         free(fdata);
81         errno = saverr;
82         return (NULL);
84     }

87     fdata = mmap(NULL, sb.st_size, PROT_READ, MAP_PRIVATE, fd, 0);
88     (void) close(fd);
89     if (fdata == NULL) {
90         errno = EINVAL;
85     if (fread(fdata, sb.st_size, 1, fp) != 1) {
86         saverr = errno;
87         free(fdata);
88         errno = saverr;
91         return (NULL);
92     }

94     frl = (_FileRuneLocale *) (void *) fdata;
95     lastp = fdata + sb.st_size;

97     variable = frl + 1;

99     if (memcmp(frl->magic, _FILE_RUNE_MAGIC_1, sizeof (frl->magic))) {
100         goto invalid;
98     free(fdata);
99     errno = EINVAL;
100     return (NULL);
101     }

103     runetype_ext_ranges = (_FileRuneEntry *) variable;
```

```

104     variable = runetype_ext_ranges + frl->runetype_ext_nranges;
105     if (variable > lastp) {
106         goto invalid;
106         free(fdata);
107         errno = EINVAL;
108         return (NULL);
107     }

109     maplower_ext_ranges = (_FileRuneEntry *)variable;
110     variable = maplower_ext_ranges + frl->maplower_ext_nranges;
111     if (variable > lastp) {
112         goto invalid;
114         free(fdata);
115         errno = EINVAL;
116         return (NULL);
113     }

115     mapupper_ext_ranges = (_FileRuneEntry *)variable;
116     variable = mapupper_ext_ranges + frl->mapupper_ext_nranges;
117     if (variable > lastp) {
118         goto invalid;
122         free(fdata);
123         errno = EINVAL;
124         return (NULL);
119     }

121     frr = runetype_ext_ranges;
122     for (x = 0; x < frl->runetype_ext_nranges; ++x) {
123         uint32_t *types;

125         if (frr[x].map == 0) {
126             int len = frr[x].max - frr[x].min + 1;
127             types = variable;
128             variable = types + len;
129             runetype_ext_len += len;
130             if (variable > lastp) {
131                 goto invalid;
137                 free(fdata);
138                 errno = EINVAL;
139                 return (NULL);
132             }
133         }
134     }

136     if ((char *)variable + frl->variable_len > (char *)lastp) {
137         goto invalid;
145         free(fdata);
146         errno = EINVAL;
147         return (NULL);
138     }

140     /*
141     * Convert from disk format to host format.
142     */
143     data = libc_malloc(sizeof (_RuneLocale) +
153     data = malloc(sizeof (_RuneLocale) +
144     (frl->runetype_ext_nranges + frl->maplower_ext_nranges +
145     frl->mapupper_ext_nranges) * sizeof (_RuneEntry) +
146     runetype_ext_len * sizeof (*rr->__types) +
147     frl->variable_len);
148     if (data == NULL) {
149         saverr = errno;
150         (void) munmap(fdata, sb.st_size);
160         free(fdata);
151         errno = saverr;
152         return (NULL);

```

```

153     }

155     rl = (_RuneLocale *) (void *)data;
156     rl->__variable = rl + 1;

158     (void) memcpy(rl->__magic, _RUNE_MAGIC_1, sizeof (rl->__magic));
159     (void) memcpy(rl->__encoding, frl->encoding, sizeof (rl->__encoding));

161     rl->__variable_len = frl->variable_len;
162     rl->__runetype_ext.__nranges = frl->runetype_ext_nranges;
163     rl->__maplower_ext.__nranges = frl->maplower_ext_nranges;
164     rl->__mapupper_ext.__nranges = frl->mapupper_ext_nranges;

166     for (x = 0; x < _CACHED_RUNES; ++x) {
167         rl->__runetype[x] = frl->runetype[x];
168         rl->__maplower[x] = frl->maplower[x];
169         rl->__mapupper[x] = frl->mapupper[x];
170     }

172     rl->__runetype_ext.__ranges = (_RuneEntry *)rl->__variable;
173     rl->__variable = rl->__runetype_ext.__ranges +
174     rl->__runetype_ext.__nranges;

176     rl->__maplower_ext.__ranges = (_RuneEntry *)rl->__variable;
177     rl->__variable = rl->__maplower_ext.__ranges +
178     rl->__maplower_ext.__nranges;

180     rl->__mapupper_ext.__ranges = (_RuneEntry *)rl->__variable;
181     rl->__variable = rl->__mapupper_ext.__ranges +
182     rl->__mapupper_ext.__nranges;

184     variable = mapupper_ext_ranges + frl->mapupper_ext_nranges;
185     frr = runetype_ext_ranges;
186     rr = rl->__runetype_ext.__ranges;
187     for (x = 0; x < rl->__runetype_ext.__nranges; ++x) {
188         uint32_t *types;

190         rr[x].__min = frr[x].min;
191         rr[x].__max = frr[x].max;
192         rr[x].__map = frr[x].map;
193         if (rr[x].__map == 0) {
194             int len = rr[x].__max - rr[x].__min + 1;
195             types = variable;
196             variable = types + len;
197             rr[x].__types = rl->__variable;
198             rl->__variable = rr[x].__types + len;
199             while (len-- > 0)
200                 rr[x].__types[len] = types[len];
201         } else
202             rr[x].__types = NULL;
203     }

205     frr = maplower_ext_ranges;
206     rr = rl->__maplower_ext.__ranges;
207     for (x = 0; x < rl->__maplower_ext.__nranges; ++x) {
208         rr[x].__min = frr[x].min;
209         rr[x].__max = frr[x].max;
210         rr[x].__map = frr[x].map;
211     }

213     frr = mapupper_ext_ranges;
214     rr = rl->__mapupper_ext.__ranges;
215     for (x = 0; x < rl->__mapupper_ext.__nranges; ++x) {
216         rr[x].__min = frr[x].min;
217         rr[x].__max = frr[x].max;
218         rr[x].__map = frr[x].map;

```

```
219     }
221     (void) memcpy(rl->__variable, variable, rl->__variable_len);
222     (void) munmap(fdata, sb.st_size);
223     free(fdata);
224     /*
225     * Go out and zero pointers that should be zero.
226     */
227     if (!rl->__variable_len)
228         rl->__variable = NULL;
229
230     if (!rl->__runetype_ext.__nranges)
231         rl->__runetype_ext.__ranges = NULL;
232
233     if (!rl->__maplower_ext.__nranges)
234         rl->__maplower_ext.__ranges = NULL;
235
236     if (!rl->__mapupper_ext.__nranges)
237         rl->__mapupper_ext.__ranges = NULL;
238
239     return (rl);
240
241 invalid:
242     (void) munmap(fdata, sb.st_size);
243     errno = EINVAL;
244     return (NULL);
245 }
246
247 unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/runetype.c

1

```
*****
2456 Fri Jul 11 11:43:01 2014
new/usr/src/lib/libc/port/locale/runetype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1993
5  *   The Regents of the University of California. All rights reserved.
6  *
7  * This code is derived from software contributed to Berkeley by
8  * Paul Borman at Krystal Technologies.
9  *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 * notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 * notice, this list of conditions and the following disclaimer in the
17 * documentation and/or other materials provided with the distribution.
18 * 4. Neither the name of the University nor the names of its contributors
19 * may be used to endorse or promote products derived from this software
20 * without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */

35 #include "lint.h"
36 #include <ctype.h>
37 #include <stdio.h>
38 #include "runetype.h"

40 unsigned int
41 __runetype(const _RuneLocale *rl, __ct_rune_t c)
42 {
43     size_t lim;
44     const _RuneRange *rr = &rl->__runetype_ext;
45     const _RuneEntry *base, *re;
46     _RuneRange *rrr = &_CurrentRuneLocale->__runetype_ext;
47     _RuneEntry *base, *re;

47     if (c < 0 || c == EOF)
48         return (0L);

50     /* Binary search -- see bsearch.c for explanation. */
51     base = rr->__ranges;
52     for (lim = rr->__nranges; lim != 0; lim >>= 1) {
53         re = base + (lim >> 1);
54         if (re->__min <= c && c <= re->__max) {
55             if (re->__types)
```

new/usr/src/lib/libc/port/locale/runetype.c

2

```
56         return (re->__types[c - re->__min]);
57     else
58         return (re->__map);
59     } else if (c > re->__max) {
60         base = re + 1;
61         lim--;
62     }
63 }

65     return (0L);
66 }
_____unchanged_portion_omitted_____
```

```

*****
5940 Fri Jul 11 11:43:01 2014
new/usr/src/lib/libc/port/locale/runetype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3 * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4 * Copyright (c) 1993
5 * The Regents of the University of California. All rights reserved.
6 *
7 * This code is derived from software contributed to Berkeley by
8 * Paul Borman at Krystal Technologies.
9 *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 * notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 * notice, this list of conditions and the following disclaimer in the
17 * documentation and/or other materials provided with the distribution.
18 * 3. Neither the name of the University nor the names of its contributors
19 * may be used to endorse or promote products derived from this software
20 * without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */
34
35 * Because we borrowed the __rune_t definitions from _types.h, we need
36 * this copyright notice as well:
37 *
38 * Copyright (c) 2002 Mike Barcroft <mike@FreeBSD.org>
39 * All rights reserved.
40 *
41 * Redistribution and use in source and binary forms, with or without
42 * modification, are permitted provided that the following conditions
43 * are met:
44 * 1. Redistributions of source code must retain the above copyright
45 * notice, this list of conditions and the following disclaimer.
46 * 2. Redistributions in binary form must reproduce the above copyright
47 * notice, this list of conditions and the following disclaimer in the
48 * documentation and/or other materials provided with the distribution.
49 *
50 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
51 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
52 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
53 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
54 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
55 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
56 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
57 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
58 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

```

```

59 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
60 * SUCH DAMAGE.
61 */

62
63 #ifndef _RUNETYPE_H_
64 #define _RUNETYPE_H_

65
66 #define _CACHED_RUNES (1 << 8) /* Must be a power of 2 */
67 #define _CRMASK (~(_CACHED_RUNES - 1))

68
69 /*
70 * rune_t is declared to be an ``int'' instead of the more natural
71 * ``unsigned long'' or ``long''. Two things are happening here. It is not
72 * unsigned so that EOF (-1) can be naturally assigned to it and used. Also,
73 * it looks like 10646 will be a 31 bit standard. This means that if your
74 * ints cannot hold 32 bits, you will be in trouble. The reason an int was
75 * chosen over a long is that the is*() and to*() routines take ints (says
76 * ANSI C), but they use __ct_rune_t instead of int.
77 *
78 * NOTE: rune_t is not covered by ANSI nor other standards, and should not
79 * be instantiated outside of lib/libc/locale. Use wchar_t, wchar_t and
80 * rune_t must be the same type. Also, wint_t must be no narrower than
81 * wchar_t, and should be able to hold all members of the largest
82 * character set plus one extra value (WEOF), and must be at least 16 bits.
83 *
84 * For compatibility with Solaris, we want to use long in ILP32, and int in
85 * LP64. This is due to historical Solaris legacy. (See <wchar.h> for
86 * the definition.)
87 */

88
89 #if defined(_LP64)
90 typedef int __ct_rune_t; /* arg type for ctype funcs */
91 #else
92 typedef int __ct_rune_t;
93 #endif

94
95 typedef int __rune_t; /* rune_t (see above) */

96
97 /*
98 * The lower 8 bits of runetype[] contain the digit value of the rune.
99 */
100 typedef struct {
101 __rune_t __min; /* First rune of the range */
102 __rune_t __max; /* Last rune (inclusive) of the range */
103 __rune_t __map; /* What first maps to in maps */
104 unsigned *_types; /* Array of types in range */
105 } _RuneEntry;
106
107 unchanged_portion_omitted
108
109
110 typedef struct {
111 char __magic[8]; /* Magic saying what version we are */
112 char __encoding[32]; /* ASCII name of encoding */
113 char __encoding[32]; /* ASCII name of this encoding */
114
115
116 unsigned int __runetype[_CACHED_RUNES];
117 __rune_t __maplower[_CACHED_RUNES];
118 __rune_t __mapupper[_CACHED_RUNES];
119
120 /*
121 * The following are to deal with Runes larger than _CACHED_RUNES - 1.
122 * Their data is actually contiguous with this structure so as to make
123 * it easier to read/write from/to disk.
124 */
125 _RuneRange __runetype_ext;
126 _RuneRange __maplower_ext;
127 _RuneRange __mapupper_ext;

```



```
129     void          *__variable; /* Data which depends on the encoding */
130     int           __variable_len; /* how long that data is */
131 } _RuneLocale;

133 #define _RUNE_MAGIC_1 "RuneMagi" /* Indicates version 0 of RuneLocale */

135 extern _RuneLocale _DefaultRuneLocale;
135 extern _RuneLocale *_CurrentRuneLocale;

137 unsigned int __runetype(const _RuneLocale *, int);

139 #endif /* !_RUNETYPE_H_ */
```

```

*****
5579 Fri Jul 11 11:43:01 2014
new/usr/src/lib/libc/port/locale/setlocale.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1996 - 2002 FreeBSD Project
5  * Copyright (c) 1991, 1993
6  *   The Regents of the University of California. All rights reserved.
7  *
8  * This code is derived from software contributed to Berkeley by
9  * Paul Borman at Krystal Technologies.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer.
16 * 2. Redistributions in binary form must reproduce the above copyright
17 * notice, this list of conditions and the following disclaimer in the
18 * documentation and/or other materials provided with the distribution.
19 * 4. Neither the name of the University nor the names of its contributors
20 * may be used to endorse or promote products derived from this software
21 * without specific prior written permission.
22 *
23 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
24 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
25 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
26 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
27 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
28 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
29 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
30 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
31 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
32 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
33 * SUCH DAMAGE.
34 */
36 #include "lint.h"
37 #include <sys/types.h>
38 #include <sys/stat.h>
39 #include <errno.h>
40 #include <limits.h>
41 #include <locale.h>
42 #include <stdlib.h>
43 #include <string.h>
44 #include <unistd.h>
44 #include <alloca.h>
45 #include <stdio.h>
46 #include "mtlib.h"
47 #include "collate.h"
48 #include "lnumeric.h" /* for struct lc_numeric */
49 #include "lctype.h" /* for struct lc_ctype */
47 #include "lmonetary.h" /* for __monetary_load_locale() */
48 #include "lnumeric.h" /* for __numeric_load_locale() */
49 #include "lmessages.h" /* for __messages_load_locale() */
50 #include "setlocale.h"
51 #include "ldpart.h"
52 #include "timelocal.h" /* for __time_load_locale() */
51 #include "../i18n/_loc_path.h"
52 #include "localeimpl.h"

```

```

53 #include "../i18n/_locale.h"
55 /*
56  * Category names for getenv() Note that this was modified
57  * for Solaris. See <iso/locale_iso.h>.
58  */
59 #define NUM_CATS 7
60 static char *categories[7] = {
61     "LC_CTYPE",
62     "LC_NUMERIC",
63     "LC_TIME",
64     "LC_COLLATE",
65     "LC_MONETARY",
66     "LC_MESSAGES",
67     "LC_ALL",
68 };
70 /*
71  * Current locales for each category
72  */
73 static char current_categories[NUM_CATS][ENCODING_LEN + 1] = {
74     "C",
75     "C",
76     "C",
77     "C",
78     "C",
79     "C",
80     "C",
81 };
83 /*
84  * Path to locale storage directory. See ../i18n/_loc_path.h
85  */
86 char *_PathLocale = _DFLT_LOCALE_PATH;
88 /*
89  * The locales we are going to try and load
90  */
91 static char new_categories[NUM_CATS][ENCODING_LEN + 1];
92 static char saved_categories[NUM_CATS][ENCODING_LEN + 1];
93 static char current_locale_string[NUM_CATS * (ENCODING_LEN + 1 + 1)];
95 /*
96  * Current locale
97  */
98 static mutex_t setlocale_lock = DEFAULTMUTEX;
99 static locale_t setlocale_list = NULL;
100 static char *currentlocale(void);
101 static char *loadlocale(int);
102 static const char *__get_locale_env(int);
104 char *
105 setlocale(int category, const char *locname)
106 setlocale(int category, const char *locale)
107 {
108     locale_t loc;
109     locale_t srch;
110     int mask;
111     int i, j, saverr;
112     const char *env, *r;
114     if (category < 0 || category > LC_ALL) {
115         if (category < 0 || category >= NUM_CATS) {
116             errno = EINVAL;
117             return (NULL);
118         }
119     }

```

```

78     if (locname == NULL)
79         return (current_locale(__global_locale, category));
110    if (locale == NULL)
111        return (category != LC_ALL ?
112            current_categories[category] : currentlocale());

81    mask = (category == LC_ALL ? LC_ALL_MASK : (1 << category));
114    /*
115     * Default to the current locale for everything.
116     */
117    for (i = 0; i < NUM_CATS; ++i)
118        (void) strcpy(new_categories[i], current_categories[i]);

83    loc = newlocale(mask, locname, NULL);
84    if (loc == NULL) {
120    /*
121     * Now go fill up new_categories from the locale argument
122     */
123    if (!*locale) {
124        if (category == LC_ALL) {
125            for (i = 0; i < NUM_CATS; ++i) {
126                if (i == LC_ALL)
127                    continue;
128                env = __get_locale_env(i);
129                if (strlen(env) > ENCODING_LEN) {
130                    errno = EINVAL;
131                    return (NULL);
132                }
133                (void) strcpy(new_categories[i], env);
134            }
135        } else {
136            env = __get_locale_env(category);
137            if (strlen(env) > ENCODING_LEN) {
138                errno = EINVAL;
139                return (NULL);
140            }
141            (void) strcpy(new_categories[category], env);
142        }
143    } else if (category != LC_ALL) {
144        if (strlen(locale) > ENCODING_LEN) {
145            errno = EINVAL;
146            return (NULL);
147        }
148        (void) strcpy(new_categories[category], locale);
149    } else {
150        if ((r = strchr(locale, '/')) == NULL) {
151            if (strlen(locale) > ENCODING_LEN) {
152                errno = EINVAL;
153                return (NULL);
154            }
155            for (i = 0; i < NUM_CATS; ++i)
156                (void) strcpy(new_categories[i], locale);
157        } else {
158            char *buf;
159            char *save;

161            buf = alloca(strlen(locale) + 1);
162            (void) strcpy(buf, locale);

164            save = NULL;
165            r = strtok_r(buf, "/", &save);
166            for (i = 0; i < NUM_CATS; i++) {
167                if (i == LC_ALL)
168                    continue;
169                if (r == NULL) {

```

88

/\*

```

89     * This next logic looks to see if we have ever used the same locale
90     * settings before.  If so, we reuse it.  We avoid ever calling
91     * freelocale() on a locale setting built up by setlocale, this
92     * ensures that consumers (uselocale) will always be thread safe;
93     * the actual locale data objects are never freed, and unique
94     * locale objects are also never freed.  We reuse to avoid leaking
95     * memory in applications that call setlocale repeatedly.
171     * Composite Locale is inadequately
172     * specified! (Or with empty fields.)
173     * The old code would fill fields
174     * out from the last one, but I think
175     * this is suboptimal.
96     */
97    lmutex_lock(&setlocale_lock);
98    for (srch = setlocale_list; srch != NULL; srch = srch->next) {
99        if (strcmp(srch->locname, loc->locname) == 0) {
100            break;
101        }
102    }
103    if (srch == NULL) {
104        if (r != NULL) {
105            /*
106             * Too many components - we had left over
107             * data in the LC_ALL.  It is malformed.
108             */
109            errno = EINVAL;
110            return (NULL);
111        }
112    }
113    if (srch == NULL) {
114        /* this is a new locale, save it for reuse later */
115        loc->next = setlocale_list;
116        loc->on_list = 1;
117        setlocale_list = loc;
118    } else {
119        /* we already had it, toss the new, and use what we found */
120        freelocale(loc);
121        loc = srch;
122    }
123    if (category != LC_ALL)
124        return (loadlocale(category));

125    for (i = 0; i < NUM_CATS; ++i) {
126        (void) strcpy(saved_categories[i], current_categories[i]);
127        if (i == LC_ALL)
128            continue;
129        if (loadlocale(i) == NULL) {
130            saverr = errno;
131            for (j = 0; j < i; j++) {
132                (void) strcpy(new_categories[j],
133                    saved_categories[j]);
134                if (i == LC_ALL)
135                    continue;
136                if (loadlocale(j) == NULL) {
137                    (void) strcpy(new_categories[j], "C");
138                    (void) loadlocale(j);
139                }
140            }
141        }
142        __global_locale = loc;
143    }
144    errno = saverr;
145    return (NULL);
146    }
147    }
148    }
149    }
150    }
151    }
152    }
153    }
154    }
155    }
156    }
157    }
158    }
159    }
160    }
161    }
162    }
163    }
164    }
165    }
166    }
167    }
168    }
169    }
170    }
171    }
172    }
173    }
174    }
175    }
176    }
177    }
178    }
179    }
180    }
181    }
182    }
183    }
184    }
185    }
186    }
187    }
188    }
189    }
190    }
191    }
192    }
193    }
194    }
195    }
196    }
197    }
198    }
199    }
200    }
201    }
202    }
203    }
204    }
205    }
206    }
207    }
208    }
209    }
210    }
211    }
212    }
213    }
214    }
215    }

```

```

216     }
217 }
218 return (currentlocale());
219 }

116 install_legacy(loc, mask);
117 lmutex_unlock(&setlocale_lock);
221 static char *
222 currentlocale(void)
223 {
224     int i;
225     int composite = 0;

119 return (current_locale(loc, category));
227 /* Look to see if any category is different */
228 for (i = 1; i < NUM_CATS; ++i) {
229     if (i == LC_ALL)
230         continue;
231     if (strcmp(current_categories[0], current_categories[i])) {
232         composite = 1;
233         break;
234     }
235 }

237 if (composite) {
238     /*
239     * Note ordering of these follows the numeric order,
240     * if the order is changed, then setlocale() will need
241     * to be changed as well.
242     */
243     (void) snprintf(current_locale_string,
244         sizeof (current_locale_string),
245         "%s/%s/%s/%s/%s/%s",
246         current_categories[LC_CTYPE],
247         current_categories[LC_NUMERIC],
248         current_categories[LC_TIME],
249         current_categories[LC_COLLATE],
250         current_categories[LC_MONETARY],
251         current_categories[LC_MESSAGES]);
252 } else {
253     (void) strcpy(current_locale_string, current_categories[0],
254         sizeof (current_locale_string));
255 }
256 return (current_locale_string);
120 }

122 static char *
123 current_locale(locale_t loc, int cat)
260 loadlocale(int category)
124 {
125     switch (cat) {
262     char *new = new_categories[category];
263     char *old = current_categories[category];
264     int (*func)(const char *);

266     if ((new[0] == '.' &&
267         (new[1] == '\0' || (new[1] == '.' && new[2] == '\0')) ||
268         strchr(new, '/') != NULL) {
269         errno = EINVAL;
270         return (NULL);
271     }

273     switch (category) {
126     case LC_CTYPE:
275         func = __wrap_setrunelocale;
276         break;

```

```

127     case LC_COLLATE:
128     case LC_MESSAGES:
129     case LC_MONETARY:
130     case LC_NUMERIC:
278         func = _collate_load_tables;
279         break;
131     case LC_TIME:
132         return (loc->locdata[cat]->l_lname);
133     case LC_ALL:
134         return (loc->locname);
281         func = _time_load_locale;
282         break;
283     case LC_NUMERIC:
284         func = __numeric_load_locale;
285         break;
286     case LC_MONETARY:
287         func = __monetary_load_locale;
288         break;
289     case LC_MESSAGES:
290         func = __messages_load_locale;
291         break;
135     default:
293         errno = EINVAL;
136         return (NULL);
137     }

297     if (strcmp(new, old) == 0)
298         return (old);

300     if (func(new) != _LDP_ERROR) {
301         (void) strcpy(old, new);
302         return (old);
303     }

305     return (NULL);
138 }

140 static void
141 install_legacy(locale_t loc, int mask)
308 static const char *
309 _get_locale_env(int category)
142 {
143     /*
144     * Update the legacy fixed variables that may be baked into
145     * legacy programs. This is really unfortunate, but we can't
146     * solve for them otherwise. Note that such legacy programs
147     * are only going to see the global locale settings, and cannot
148     * benefit from uselocale().
149     */
150     if (mask & LC_NUMERIC_MASK) {
151         struct lc_numeric *lnum;
152         lnum = loc->locdata[LC_NUMERIC]->l_data[0];
153         _numeric[0] = *lnum->decimal_point;
154         _numeric[1] = *lnum->thousands_sep;
155     }
311     const char *env;

157     if (mask & LC_CTYPE_MASK) {
158         struct lc_ctype *lct;
159         lct = loc->locdata[LC_CTYPE]->l_data[0];
160         for (int i = 0; i < _CACHED_RUNES; i++) {
161             /* ctype can only encode the lower 8 bits. */
162             _ctype[i+1] = lct->lc_ctype_mask[i] & 0xff;
163             _ctype_mask[i] = lct->lc_ctype_mask[i];
164         }
313     /* 1. check LC_ALL. */

```

```
314     env = getenv(categories[LC_ALL]);

166     /* The bottom half is the toupper/lower array */
167     for (int i = 0; i < _CACHED_RUNES; i++) {
168         int u, l;
169         __ctype[258 + i] = i;
170         u = lct->lc_trans_upper[i];
171         l = lct->lc_trans_lower[i];
172         if (u && u != i)
173             __ctype[258+i] = u;
174         if (l && l != i)
175             __ctype[258+i] = l;
176
177         /* Don't forget these annoyances either! */
178         __trans_upper[i] = u;
179         __trans_lower[i] = l;
180     }
181
182     /* 2. check LC_* */
183     if (env == NULL || !*env)
184         env = getenv(categories[category]);
185
186     /* 3. check LANG */
187     if (env == NULL || !*env)
188         env = getenv("LANG");
189
190     /* Maximum mblen, cswidth, weird legacy */
191     __ctype[520] = lct->lc_max_mblen;
192
193     /* 4. if none is set, fall to "C" */
194     if (env == NULL || !*env)
195         env = "C";
196
197     return (env);
198 }
199
200 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libc/port/locale/setlocale.h

1

\*\*\*\*\*

1608 Fri Jul 11 11:43:01 2014

new/usr/src/lib/libc/port/locale/setlocale.h

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (C) 1997 by Andrey A. Chernov, Moscow, Russia.
4  * All rights reserved.
5  *
6  * Redistribution and use in source and binary forms, with or without
7  * modification, are permitted provided that the following conditions
8  * are met:
9  * 1. Redistributions of source code must retain the above copyright
10 * notice, this list of conditions and the following disclaimer.
11 * 2. Redistributions in binary form must reproduce the above copyright
12 * notice, this list of conditions and the following disclaimer in the
13 * documentation and/or other materials provided with the distribution.
14 *
15 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
16 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
17 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
18 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
19 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
20 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
21 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
22 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
23 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
24 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
25 * SUCH DAMAGE.
26 */

28 #ifndef _SETLOCALE_H_
29 #define _SETLOCALE_H_

31 #define _PATH_LOCALE "/usr/share/locale"
32 #define ENCODING_LEN 31
33 #define CATEGORY_LEN 11

35 extern char *_PathLocale;

36 int __detect_path_locale(void);
37 int __wrap_setrunelocale(const char *);

37 #endif /* !_SETLOCALE_H_ */
```

```

*****
4433 Fri Jul 11 11:43:02 2014
new/usr/src/lib/libc/port/locale/setrunelocale.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1993
5  *   The Regents of the University of California. All rights reserved.
6  *
7  * This code is derived from software contributed to Berkeley by
8  * Paul Borman at Krystal Technologies.
9  *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 * notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 * notice, this list of conditions and the following disclaimer in the
17 * documentation and/or other materials provided with the distribution.
18 * 4. Neither the name of the University nor the names of its contributors
19 * may be used to endorse or promote products derived from this software
20 * without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */

35 #include "lint.h"
36 #include "file64.h"
37 #include <errno.h>
38 #include <limits.h>
39 #include <string.h>
40 #include <stdio.h>
41 #include <stdlib.h>
42 #include <unistd.h>
43 #include <wchar.h>
44 #include "runetype.h"
45 #include "ldpart.h"
46 #include "mblocal.h"
47 #include "setlocale.h"
48 #include "_ctype.h"
49 #include "lctype.h"
50 #include "localeimpl.h"
51 #include "../i18n/_locale.h"

52 extern _RuneLocale * _Read_RuneMagi(const char *);
53 extern _RuneLocale * _Read_RuneMagi(FILE *);
54 extern unsigned char _ctype_C[];

54 struct lc_ctype lc_ctype_posix = {
55     .lc_mbrtowc = _mbrtowc_ascii,

```

```

56     .lc_mbsinit = _mbsinit_ascii,
57     .lc_mbsnrtowcs = _mbsnrtowcs_ascii,
58     .lc_wcrtomb = _wcrtomb_ascii,
59     .lc_wcsnrtombs = _wcsnrtombs_ascii,
60     .lc_is_ascii = 1,
61     .lc_max_mblen = 1,
62     .lc_trans_upper = _DefaultRuneLocale._mapupper,
63     .lc_trans_lower = _DefaultRuneLocale._maplower,
64     .lc_ctype_mask = _DefaultRuneLocale._runetype,
65 };
53 static int _setrunelocale(const char *);

67 struct locdata __posix_ctype_locdata = {
68     .l_lname = "C",
69     .l_data = { &lc_ctype_posix, &_DefaultRuneLocale }
70 };
55 static int
56 _setrunelocale(const char *encoding)
57 {
58     FILE *fp;
59     char name[PATH_MAX];
60     _RuneLocale *rl;
61     int saverr, ret;
62     size_t (*old_mbrtowc)(wchar_t * _RESTRICT_KYWD,
63         const char * _RESTRICT_KYWD, size_t, mbstate_t * _RESTRICT_KYWD);
64     size_t (*old_wcrtomb)(char * _RESTRICT_KYWD, wchar_t,
65         mbstate_t * _RESTRICT_KYWD);
66     int (*old_mbsinit)(const mbstate_t *);
67     size_t (*old_mbsnrtowcs)(wchar_t * _RESTRICT_KYWD,
68         const char ** _RESTRICT_KYWD, size_t, size_t,
69         mbstate_t * _RESTRICT_KYWD);
70     size_t (*old_wcsnrtombs)(char * _RESTRICT_KYWD,
71         const wchar_t ** _RESTRICT_KYWD, size_t, size_t,
72         mbstate_t * _RESTRICT_KYWD);
73     static char ctype_encoding[ENCODING_LEN + 1];
74     static _RuneLocale *CachedRuneLocale;
75     static size_t (*Cached_mbrtowc)(wchar_t * _RESTRICT_KYWD,
76         const char * _RESTRICT_KYWD, size_t, mbstate_t * _RESTRICT_KYWD);
77     static size_t (*Cached_wcrtomb)(char * _RESTRICT_KYWD, wchar_t,
78         mbstate_t * _RESTRICT_KYWD);
79     static int (*Cached_mbsinit)(const mbstate_t *);
80     static size_t (*Cached_mbsnrtowcs)(wchar_t * _RESTRICT_KYWD,
81         const char ** _RESTRICT_KYWD, size_t, size_t,
82         mbstate_t * _RESTRICT_KYWD);
83     static size_t (*Cached_wcsnrtombs)(char * _RESTRICT_KYWD,
84         const wchar_t ** _RESTRICT_KYWD, size_t, size_t,
85         mbstate_t * _RESTRICT_KYWD);

73 /*
74  * Table of initializers for encodings. When you add a new encoding type,
75  * this table should be updated.
76  */
77 /*
78  * The "C" and "POSIX" locale are always here.
79  */
77 static struct {
78     const char *e_name;
79     void (*e_init)(struct lc_ctype *);
80 } encodings[] = {
81     { "NONE", _none_init },
82     { "UTF-8", _UTF8_init },
83     { "EUC-CN", _EUC_CN_init },
84     { "EUC-JP", _EUC_JP_init },
85     { "EUC-KR", _EUC_KR_init },
86     { "EUC-TW", _EUC_TW_init },
87     { "GB18030", _GB18030_init },

```

```

88     { "GB2312",    _GB2312_init },
89     { "GBK",      _GBK_init   },
90     { "BIG5",     _BIG5_init  },
91     { "MSKanji",  _MSKanji_init },
92     { NULL,       NULL       },
93 };
90     if (strcmp(encoding, "C") == 0 || strcmp(encoding, "POSIX") == 0) {
91         int i;
92
93         (void) memcpy(__ctype, __ctype_C, SZ_TOTAL);
94
95     struct locdata *
96     _lc_ctype_load(const char *name)
97     {
98         struct locdata *ldata;
99         struct lc_ctype *lct;
100         RuneLocale *rl;
101         int i;
102         char path[PATH_MAX];
103         for (i = 0; i < _CACHED_RUNES; i++) {
104             __ctype_mask[i] = _DefaultRuneLocale.__runetype[i];
105             __trans_upper[i] = _DefaultRuneLocale.__mapupper[i];
106             __trans_lower[i] = _DefaultRuneLocale.__maplower[i];
107         }
108
109         if ((ldata = _locdata_alloc(name, sizeof (*lct))) == NULL)
110             return (NULL);
111         lct = ldata->l_data[0];
112         (void) _none_init(&_DefaultRuneLocale);
113         return (0);
114     }
115
116     /*
117     * If the locale name is the same as our cache, use the cache.
118     */
119     if (CachedRuneLocale != NULL &&
120         strcmp(encoding, ctype_encoding) == 0) {
121         _CurrentRuneLocale = CachedRuneLocale;
122         __mbrtowc = Cached__mbrtowc;
123         __mbsinit = Cached__mbsinit;
124         __mbsnrtowcs = Cached__mbsnrtowcs;
125         __wctomb = Cached__wctomb;
126         __wcsnrtombs = Cached__wcsnrtombs;
127         return (0);
128     }
129
130     /*
131     * Slurp the locale file into the cache.
132     */
133     (void) snprintf(path, sizeof (path), "%s/%s/LC_CTYPE/LCL_DATA",
134                    _PathLocale, name);
135     (void) snprintf(name, sizeof (name), "%s/%s/LC_CTYPE/LCL_DATA",
136                    _PathLocale, encoding);
137
138     if ((rl = _Read_RuneMagi(path)) == NULL) {
139         _locdata_free(ldata);
140         errno = EINVAL;
141         return (NULL);
142     }
143     if ((fp = fopen(name, "r")) == NULL)
144         return (errno == 0 ? ENOENT : errno);
145
146     if ((rl = _Read_RuneMagi(fp)) == NULL) {
147         saverr = (errno == 0 ? EINVAL : errno);
148         (void) fclose(fp);
149         return (saverr);

```

```

119     }
120     ldata->l_data[1] = rl;
121     (void) fclose(fp);
122
123     lct->lc_mbrtowc = NULL;
124     lct->lc_mbsinit = NULL;
125     lct->lc_mbsnrtowcs = NULL;
126     lct->lc_wctomb = NULL;
127     lct->lc_wcsnrtombs = NULL;
128     lct->lc_ctype_mask = rl->__runetype;
129     lct->lc_trans_upper = rl->__mapupper;
130     lct->lc_trans_lower = rl->__maplower;
131     old__mbrtowc = __mbrtowc;
132     old__mbsinit = __mbsinit;
133     old__mbsnrtowcs = __mbsnrtowcs;
134     old__wctomb = __wctomb;
135     old__wcsnrtombs = __wcsnrtombs;
136
137     /* set up the function pointers */
138     for (i = 0; encodings[i].e_name != NULL; i++) {
139         int l = strlen(encodings[i].e_name);
140         if ((strcmp(rl->__encoding, encodings[i].e_name, l) == 0) &&
141             (rl->__encoding[l] == '0' || rl->__encoding[l] == '@')) {
142             encodings[i].e_init(lct);
143             break;
144         }
145         __mbrtowc = NULL;
146         __mbsinit = NULL;
147         __mbsnrtowcs = __mbsnrtowcs_std;
148         __wctomb = NULL;
149         __wcsnrtombs = __wcsnrtombs_std;
150
151         if (strcmp(rl->__encoding, "NONE") == 0)
152             ret = _none_init(rl);
153         else if (strcmp(rl->__encoding, "UTF-8") == 0)
154             ret = _UTF8_init(rl);
155         else if (strcmp(rl->__encoding, "EUC-CN") == 0)
156             ret = _EUC_CN_init(rl);
157         else if (strcmp(rl->__encoding, "EUC-JP") == 0)
158             ret = _EUC_JP_init(rl);
159         else if (strcmp(rl->__encoding, "EUC-KR") == 0)
160             ret = _EUC_KR_init(rl);
161         else if (strcmp(rl->__encoding, "EUC-TW") == 0)
162             ret = _EUC_TW_init(rl);
163         else if (strcmp(rl->__encoding, "GB18030") == 0)
164             ret = _GB18030_init(rl);
165         else if (strcmp(rl->__encoding, "GB2312") == 0)
166             ret = _GB2312_init(rl);
167         else if (strcmp(rl->__encoding, "GBK") == 0)
168             ret = _GBK_init(rl);
169         else if (strcmp(rl->__encoding, "BIG5") == 0)
170             ret = _BIG5_init(rl);
171         else if (strcmp(rl->__encoding, "MSKanji") == 0)
172             ret = _MSKanji_init(rl);
173         else
174             ret = EINVAL;
175
176         if (ret == 0) {
177             if (CachedRuneLocale != NULL) {
178                 free(CachedRuneLocale);
179             }
180             CachedRuneLocale = _CurrentRuneLocale;
181             Cached__mbrtowc = __mbrtowc;
182             Cached__mbsinit = __mbsinit;
183             Cached__mbsnrtowcs = __mbsnrtowcs;
184             Cached__wctomb = __wctomb;
185             Cached__wcsnrtombs = __wcsnrtombs;

```



```

183     (void) strcpy(ctype_encoding, encoding);
184
185     /*
186     * We need to overwrite the _ctype array. This requires
187     * some finagling. This is because references to it may
188     * have been baked into applications.
189     *
190     * Note that it is interesting that toupper/tolower only
191     * produce defined results when the input is representable
192     * as a byte.
193     */
194
195     /*
196     * The top half is the type mask array. Because we
197     * want to support both legacy Solaris code (which have
198     * mask valeus baked in to them), and we want to be able
199     * to import locale files from other sources (FreeBSD)
200     * which probably uses different masks, we have to perform
201     * a conversion here. Ugh. Note that the _CTYPE definitions
202     * we use from FreeBSD are richer than the Solaris legacy.
203     *
204     * We have to cope with these limitations though, because the
205     * inadequate Solaris definitions were baked into binaries.
206     */
207     for (int i = 0; i < _CACHED_RUNES; i++) {
208         /* ctype can only encode the lower 8 bits. */
209         __ctype[i+1] = rl->__runetype[i] & 0xff;
210         __ctype_mask[i] = rl->__runetype[i];
211     }
212
213     if (encodings[i].e_name == NULL) {
214         __locdata_free(ldata);
215         errno = EINVAL;
216         return (NULL);
217     }
218
219     /* The bottom half is the toupper/lower array */
220     for (int i = 0; i < _CACHED_RUNES; i++) {
221         __ctype[258 + i] = i;
222         if (rl->__mapupper[i] && rl->__mapupper[i] != i)
223             __ctype[258+i] = rl->__mapupper[i];
224         if (rl->__maplower[i] && rl->__maplower[i] != i)
225             __ctype[258+i] = rl->__maplower[i];
226
227         /* Don't forget these annoyances either! */
228         __trans_upper[i] = rl->__mapupper[i];
229         __trans_lower[i] = rl->__maplower[i];
230     }
231
232     /*
233     * Note that we expect the init code will have populated
234     * the CSWIDTH array (__ctype[514-520]) properly.
235     */
236     } else {
237         __mbrtowc = old__mbrtowc;
238         __mbsinit = old__mbsinit;
239         __mbsnrtowcs = old__mbsnrtowcs;
240         __wctomb = old__wctomb;
241         __wcsnrtombs = old__wcsnrtombs;
242         free(rl);
243     }
244
245     return (ldata);
246 }
247
248 int
249 __wrap_setrunelocale(const char *locale)

```

```

244 {
245     int ret = __setrunelocale(locale);
246
247     if (ret != 0) {
248         errno = ret;
249         return (_LDP_ERROR);
250     }
251     return (_LDP_LOADED);
252 }
253
254 _____unchanged_portion_omitted_____

```

```

*****
2119 Fri Jul 11 11:43:02 2014
new/usr/src/lib/libc/port/locale/strcasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
24  * Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
25  */

27 /*      Copyright (c) 1983, 1984, 1985, 1986, 1987, 1988, 1989 AT&T      */
28 /*      All Rights Reserved      */

30 /*
31  * Portions of this source code were derived from Berkeley 4.3 BSD
32  * under license from the Regents of the University of California.
33  */

35 #include "lint.h"
36 #include <sys/types.h>
37 #include <strings.h>
38 #include <ctype.h>
39 #include <locale.h>
40 #include "localeimpl.h"
41 #include "lctype.h"

43 int
44 strcasecmp_l(const char *s1, const char *s2, locale_t loc)
45 strcasecmp(const char *s1, const char *s2)
46 {
47     extern int charset_is_ascii;
48     extern int ascii_strcasecmp(const char *s1, const char *s2);
49     const int *cm;
50     int *cm;
51     const uchar_t *us1;
52     const uchar_t *us2;
53     const struct lc_ctype *lct = loc->ctype;

55     /*
56      * If we are in a locale that uses the ASCII character set
57      * (C or POSIX), use the fast ascii_strcasecmp() function.
58      */

```

```

56     if (lct->lc_is_ascii)
57     if (charset_is_ascii)
58         return (ascii_strcasecmp(s1, s2));

59     cm = lct->lc_trans_lower;
60     cm = __trans_lower;
61     us1 = (const uchar_t *)s1;
62     us2 = (const uchar_t *)s2;

63     while (cm[*us1] == cm[*us2++])
64         if (*us1++ == '\0')
65             return (0);
66     return (cm[*us1] - cm[*us2 - 1]);
67 }

69 int
70 strcasecmp(const char *s1, const char *s2)
71 {
72     /* would be nice to avoid uselocale()... but I don't see how */
73     return (strcasecmp_l(s1, s2, uselocale(NULL)));
74 }

_____unchanged_portion_omitted_____

```

```

*****
2170 Fri Jul 11 11:43:02 2014
new/usr/src/lib/libc/port/locale/strcasestr.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
24 * Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
25 */

27 /*      Copyright (c) 1988 AT&T */
28 /*      All Rights Reserved      */

30 #include "lint.h"
31 #include <string.h>
32 #include <ctype.h>
33 #include <sys/types.h>
34 #include <locale.h>
35 #include "lctype.h"
36 #include "localeimpl.h"

38 /*
39  * strcasestr() locates the first occurrence in the string s1 of the
40  * sequence of characters (excluding the terminating null character)
41  * in the string s2, ignoring case.  strcasestr() returns a pointer
42  * to the located string, or a null pointer if the string is not found.
43  * If s2 is empty, the function returns s1.
44  */

46 char *
47 strcasestr_l(const char *s1, const char *s2, locale_t loc)
48 strcasestr(const char *s1, const char *s2)
49 {
50     const int *cm = loc->ctype->lc_trans_lower;
51     int *cm = __trans_lower;
52     const uchar_t *us1 = (const uchar_t *)s1;
53     const uchar_t *us2 = (const uchar_t *)s2;
54     const uchar_t *tptr;
55     int c;

56     if (us2 == NULL || *us2 == '\0')
57         return ((char *)us1);

```

```

58     c = cm[*us2];
59     while (*us1 != '\0') {
60         if (c == cm[*us1++]) {
61             tptr = us1;
62             while (cm[c = *++us2] == cm[*us1++] && c != '\0')
63                 continue;
64             if (c == '\0')
65                 return ((char *)tptr - 1);
66             us1 = tptr;
67             us2 = (const uchar_t *)s2;
68             c = cm[*us2];
69         }
70     }

72     return (NULL);
73 }

75 char *
76 strcasestr(const char *s1, const char *s2)
77 {
78     return (strcasestr_l(s1, s2, uselocale(NULL)));
79 }

```

unchanged\_portion\_omitted

```

*****
3408 Fri Jul 11 11:43:02 2014
new/usr/src/lib/libc/port/locale/strcoll.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1995 Alex Tatmanjants <alex@elvisti.kiev.ua>
5  *      at Electronni Visti IA, Kiev, Ukraine.
6  *      All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include "file64.h"
32 #include <alloca.h>
33 #include <stdlib.h>
34 #include <string.h>
35 #include <errno.h>
36 #include <wchar.h>
37 #include <xlocale.h>
38 #include "localeimpl.h"
39 #include "collate.h"

41 #define ALLOCA_LIMIT 16

43 /*
44  * In order to properly handle multibyte locales, its easiet to just
45  * convert to wide characters and then use wcscoll. However if an
46  * error occurs, we gracefully fall back to simple strcmp. Caller
47  * should check errno.
48  */
49 int
50 strcoll_l(const char *s1, const char *s2, locale_t loc)
51 {
52     int ret;
53     wchar_t *t1 = NULL, *t2 = NULL;
54     wchar_t *w1 = NULL, *w2 = NULL;
55     size_t sz1, sz2;
56     const struct lc_collate *lcc = loc->collate;

```

```

58     mbstate_t mbs1 = { 0 }; /* initial states */
59     mbstate_t mbs2 = { 0 };
60     if (!_collate_load_error)
61         goto error;

61     if (lcc->lc_is_posix)
62         return (strcmp(s1, s2));

64     sz1 = strlen(s1) + 1;
65     sz2 = strlen(s2) + 1;

67     /*
68      * Simple assumption: conversion to wide format is strictly
69      * reducing, i.e. a single byte (or multibyte character)
70      * cannot result in multiple wide characters.
71      *
72      * We gain a bit of performance by giving preference to alloca
73      * for small string allocations.
74      */
75     if (sz1 > ALLOCA_LIMIT) {
76         if ((t1 = malloc(sz1 * sizeof (wchar_t))) == NULL)
77             goto error;
78         w1 = t1;
79     } else {
80         if ((w1 = alloca(sz1 * sizeof (wchar_t))) == NULL)
81             goto error;
82     }
83     if (sz2 > ALLOCA_LIMIT) {
84         if ((t2 = malloc(sz2 * sizeof (wchar_t))) == NULL)
85             goto error;
86         w2 = t2;
87     } else {
88         if ((w2 = alloca(sz2 * sizeof (wchar_t))) == NULL)
89             goto error;
90     }

92     if ((mbsrtowcs_l(w1, &s1, sz1, &mbs1, loc)) == (size_t)-1)
93         if ((mbstowcs(w1, s1, sz1)) == (size_t)-1)
94             goto error;

95     if ((mbsrtowcs_l(w2, &s2, sz2, &mbs2, loc)) == (size_t)-1)
96         if ((mbstowcs(w2, s2, sz2)) == (size_t)-1)
97             goto error;

98     ret = wcscoll_l(w1, w2, loc);
99     ret = wcscoll(w1, w2);
100     if (t1)
101         free(t1);
102     if (t2)
103         free(t2);

104     return (ret);

106 error:
107     if (t1)
108         free(t1);
109     if (t2)
110         free(t2);
111     return (strcmp(s1, s2));
112 }

114 int
115 strcoll(const char *s1, const char *s2)
116 {
117     return (strcoll_l(s1, s2, uselocale(NULL)));
118 }

```

```

*****
16934 Fri Jul 11 11:43:02 2014
new/usr/src/lib/libc/port/locale/strfmon.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 *
28 */

30 #ifndef _LCONV_C99
31 #define _LCONV_C99
32 #endif

34 #include "lint.h"
35 #include <sys/types.h>
36 #include <ctype.h>
37 #include <errno.h>
38 #include <limits.h>
39 #include <locale.h>
40 #include <monetary.h>
41 #include <stdarg.h>
42 #include <stdio.h>
43 #include <stdlib.h>
44 #include <string.h>
45 #include "localeimpl.h"
46 #include "lmonetary.h"
47 #include "lnumeric.h"

49 /* internal flags */
50 #define NEED_GROUPING 0x01 /* print digits grouped (default) */
51 #define SIGN_POSN_USED 0x02 /* '+' or '(' usage flag */
52 #define LOCALE_POSN 0x04 /* use locale defined +/- (default) */
53 #define PARENTH_POSN 0x08 /* enclose negative amount in () */
54 #define SUPPRESS_CURR_SYMBOL 0x10 /* suppress the currency from output */
55 #define LEFT_JUSTIFY 0x20 /* left justify */
56 #define USE_INTL_CURRENCY 0x40 /* use international currency symbol */
57 #define IS_NEGATIVE 0x80 /* is argument value negative ? */

```

```

59 /* internal macros */
60 #define PRINT(CH) {
61     if (dst >= s + maxsize)
62         goto e2big_error;
63     *dst++ = CH;
64 }

66 #define PRINTS(STR) {
67     const char *tmps = STR;
68     char *tmps = STR;
69     while (*tmps != '\0')
70         PRINT(*tmps++);
71 }

unchanged_portion_omitted

93 #define GRPSEP {
94     bufend -= thousands_len;
95     (void) memcpy(bufend, thousands_sep, thousands_len);
96     *--bufend = thousands_sep;
97 }

99 static void setup_vars(const struct lc_monetary *, int, char *, char *, char *,
100 const char *);
101 static int calc_left_pad(const struct lc_monetary *, int, const char *);
102 static char *format_grouped_double(const struct lc_monetary *,
103 const struct lc_numeric *, double, int *, int, int, int);
104 static void __setup_vars(int, char *, char *, char *, char *);
105 static int __calc_left_pad(int, char *);
106 static char *__format_grouped_double(double, int *, int, int, int);

105 ssize_t
106 strfmon_impl(char * RESTRICT_KYWD s, size_t maxsize, locale_t loc,
107 const char * RESTRICT_KYWD format, va_list ap)
108 strfmon(char * RESTRICT_KYWD s, size_t maxsize,
109 const char * RESTRICT_KYWD format, ...)
110 {
111     va_list ap;
112     char *dst; /* output destination pointer */
113     const char *fmt; /* current format position pointer */
114     struct lconv *lc; /* pointer to lconv structure */
115     char *asciivalue; /* formatted double pointer */

116     int flags; /* formatting options */
117     int pad_char; /* padding character */
118     int pad_size; /* pad size */
119     int width; /* field width */
120     int left_prec; /* left precision */
121     int right_prec; /* right precision */
122     double value; /* just value */
123     char space_char = ' '; /* space after currency */

124     char cs_precedes; /* values from struct lc_monetary */
125     char sep_by_space; /* values gathered from struct lconv */
126     char sign_posn;
127     const char *signstr;
128     const char *currency_symbol;
129     char *signstr;
130     char *currency_symbol;

128     char *tmpptr; /* temporary vars */
129     int sverrno;
130     const struct lc_monetary *lmon; /* monetary structure */
131     const struct lc_numeric *lnum; /* numeric structure */

```

```

133     lmon = loc->monetary;
134     lnum = loc->numeric;
135     va_start(ap, format);

128     lc = localeconv();
136     dst = s;
137     fmt = format;
138     asciivalue = NULL;
139     currency_symbol = NULL;
140     pad_size = 0;

142     while (*fmt) {
143         /* pass nonformatting characters AS IS */
144         if (*fmt != '%')
145             goto literal;

147         /* '%' found ! */

149         /* "%%" mean just '%' */
150         if (*(fmt+1) == '%') {
151             fmt++;
152             literal:
153                 PRINT(*fmt++);
154                 continue;
155         }

157         /* set up initial values */
158         flags = (NEED_GROUPING|LOCALE_POSN);
159         pad_char = ' '; /* padding character is "space" */
160         left_prec = -1; /* no left precision specified */
161         right_prec = -1; /* no right precision specified */
162         width = -1; /* no width specified */
163         value = 0; /* we have no value to print now */

165         /* Flags */
166         for (;;) {
167             switch (*++fmt) {
168                 case '=': /* fill character */
169                     pad_char = *++fmt;
170                     if (pad_char == '\0')
171                         goto format_error;
172                     continue;
173                 case '^': /* not group currency */
174                     flags &= ~(NEED_GROUPING);
175                     continue;
176                 case '+': /* use locale defined signs */
177                     if (flags & SIGN_POSN_USED)
178                         goto format_error;
179                     flags |= (SIGN_POSN_USED|LOCALE_POSN);
180                     continue;
181                 case '(': /* enclose negatives with () */
182                     if (flags & SIGN_POSN_USED)
183                         goto format_error;
184                     flags |= (SIGN_POSN_USED|PARENTH_POSN);
185                     continue;
186                 case '!': /* suppress currency symbol */
187                     flags |= SUPPRESS_CURR_SYMBOL;
188                     continue;
189                 case '-': /* alignment (left) */
190                     flags |= LEFT_JUSTIFY;
191                     continue;
192                 default:
193                     break;
194             }
195             break;
196         }

```

```

198         /* field Width */
199         if (isdigit((unsigned char)*fmt)) {
200             GET_NUMBER(width);
201             /*
202              * Do we have enough space to put number with
203              * required width ?
204              */
205             if ((unsigned int)width >= maxsize - (dst - s))
206                 goto e2big_error;
207         }

209         /* Left precision */
210         if (*fmt == '#') {
211             if (!isdigit((unsigned char)*++fmt))
212                 goto format_error;
213             GET_NUMBER(left_prec);
214             if ((unsigned int)left_prec >= maxsize - (dst - s))
215                 goto e2big_error;
216         }

218         /* Right precision */
219         if (*fmt == ',') {
220             if (!isdigit((unsigned char)*++fmt))
221                 goto format_error;
222             GET_NUMBER(right_prec);
223             if ((unsigned int)right_prec >= maxsize - (dst - s) -
224                 left_prec)
225                 goto e2big_error;
226         }

228         /* Conversion Characters */
229         switch (*fmt++) {
230             case 'i': /* use international currency format */
231                 flags |= USE_INTL_CURRENCY;
232                 break;
233             case 'n': /* use national currency format */
234                 flags &= ~(USE_INTL_CURRENCY);
235                 break;
236             default:
237                 /* required char missing or premature EOS */
238                 goto format_error;
239         }

243         if (currency_symbol != NULL)
244             free(currency_symbol);
245         if (flags & USE_INTL_CURRENCY) {
246             currency_symbol = lmon->int_curr_symbol;
247             /* by definition three letters followed by a space */
248             currency_symbol = strdup(lc->int_curr_symbol);
249             if (currency_symbol != NULL)
250                 space_char = currency_symbol[3];
251                 space_char = *(currency_symbol+3);
252         } else
253             currency_symbol = lmon->currency_symbol;
254             currency_symbol = strdup(lc->currency_symbol);

243         if (currency_symbol == NULL)
244             goto end_error; /* ENOMEM. */

249         /* value itself */
250         value = va_arg(ap, double);

252         /* detect sign */
253         if (value < 0) {
254             flags |= IS_NEGATIVE;

```

```

255         value = -value;
256     }
257
258     /* fill left_prec with amount of padding chars */
259     if (left_prec >= 0) {
260         pad_size = calc_left_pad(lmon, (flags ^ IS_NEGATIVE),
261                                 pad_size = __calc_left_pad((flags ^ IS_NEGATIVE),
262                                                             currency_symbol) -
263                                 calc_left_pad(lmon, flags, currency_symbol);
264         __calc_left_pad(flags, currency_symbol);
265         if (pad_size < 0)
266             pad_size = 0;
267     }
268
269     if (asciivalue != NULL)
270         free(asciivalue);
271     asciivalue = format_grouped_double(lmon, lnum, value, &flags,
272                                       asciivalue = __format_grouped_double(value, &flags,
273                                       left_prec, right_prec, pad_char);
274     if (asciivalue == NULL)
275         goto end_error; /* errno already set */
276     /* to ENOMEM by malloc() */
277
278     /* set some variables for later use */
279     setup_vars(lmon, flags, &cs_precedes, &sep_by_space,
280              &sign_posn, &signstr);
281     __setup_vars(flags, &cs_precedes, &sep_by_space, &sign_posn,
282                &signstr);
283
284     /*
285     * Description of some LC_MONETARY's values:
286     *
287     * * p_cs_precedes & n_cs_precedes
288     * * = 1 - $currency_symbol precedes the value
289     * *   for a monetary quantity with a non-negative value
290     * * = 0 - symbol succeeds the value
291     * *
292     * * p_sep_by_space & n_sep_by_space
293     * * = 0 - no space separates $currency_symbol
294     * *   from the value for a monetary quantity with a
295     * *   non-negative value
296     * * = 1 - space separates the symbol from the value
297     * * = 2 - space separates the symbol and the sign string,
298     * *   if adjacent.
299     * *
300     * * p_sign_posn & n_sign_posn
301     * * = 0 - parentheses enclose the quantity and the
302     * *   $currency_symbol
303     * * = 1 - the sign string precedes the quantity and the
304     * *   $currency_symbol
305     * * = 2 - the sign string succeeds the quantity and the
306     * *   $currency_symbol
307     * * = 3 - the sign string precedes the $currency_symbol
308     * * = 4 - the sign string succeeds the $currency_symbol
309     */
310
311     tmppptr = dst;
312
313     while (pad_size-- > 0)
314         PRINT(' ');
315
316     if (sign_posn == 0 && (flags & IS_NEGATIVE))

```

```

316         PRINT('(');
317
318     if (cs_precedes == 1) {
319         if (sign_posn == 1 || sign_posn == 3) {
320             PRINTS(signstr);
321             if (sep_by_space == 2)
322                 if (sep_by_space == 2) /* XXX: ? */
323                     PRINT(' ');
324         }
325     } else if (!(flags & SUPPRESS_CURR_SYMBOL)) {
326         PRINTS(currency_symbol);
327
328         if (sign_posn == 4) {
329             if (sep_by_space == 2)
330                 PRINT(space_char);
331             PRINTS(signstr);
332             if (sep_by_space == 1)
333                 PRINT(' ');
334         } else if (sep_by_space == 1)
335             PRINT(space_char);
336     }
337 } else if (sign_posn == 1)
338     PRINTS(signstr);
339
340 PRINTS(asciivalue);
341
342 if (cs_precedes == 0) {
343     if (sign_posn == 3) {
344         if (sep_by_space == 1)
345             PRINT(' ');
346         PRINTS(signstr);
347     }
348
349     if (!(flags & SUPPRESS_CURR_SYMBOL)) {
350         if ((sign_posn == 3 && sep_by_space == 2) ||
351             (sep_by_space == 1 && (sign_posn == 0 ||
352             sign_posn == 1 || sign_posn == 2 ||
353             sign_posn == 4)))
354             PRINT(space_char);
355         PRINTS(currency_symbol); /* XXX: len */
356         if (sign_posn == 4) {
357             if (sep_by_space == 2)
358                 PRINT(' ');
359             PRINTS(signstr);
360         }
361     }
362 }
363
364 if (sign_posn == 2) {
365     if (sep_by_space == 2)
366         PRINT(' ');
367     PRINTS(signstr);
368 }
369
370 if (sign_posn == 0 && (flags & IS_NEGATIVE))
371     PRINT('(');
372
373 if (dst - tmppptr < width) {
374     if (flags & LEFT_JUSTIFY) {
375         while (dst - tmppptr < width)
376             PRINT(' ');
377     } else {
378         pad_size = dst - tmppptr;
379         (void) memmove(tmppptr + width - pad_size, tmppptr,
380                       pad_size);

```

```

381             (void) memset(tmpptr, ' ', width-pad_size);
382             dst += width-pad_size;
383         }
384     }
385 }

387     PRINT('\0');
385     va_end(ap);
388     free(asciivalue);
387     free(currency_symbol);
389     return (dst - s - 1); /* size of put data except trailing '\0' */

391 e2big_error:
392     errno = E2BIG;
393     goto end_error;

395 format_error:
396     errno = EINVAL;

398 end_error:
399     serrno = errno;
400     if (asciivalue != NULL)
401         free(asciivalue);
401     if (currency_symbol != NULL)
402         free(currency_symbol);
402     errno = serrno;
404     va_end(ap);
403     return (-1);
404 }

406 ssize_t
407 strfmon(char *_RESTRICT_KYWD s, size_t maxsize,
408         const char *_RESTRICT_KYWD format, ...)
409 static void
409 __setup_vars(int flags, char *cs_precedes, char *sep_by_space,
410             char *sign_posn, char **signstr)
409 {
410     va_list ap;
411     ssize_t ret;

413     va_start(ap, format);
414     ret = strfmon_impl(s, maxsize, uselocale(NULL), format, ap);
415     va_end(ap);
416     return (ret);
417 }
413     struct lconv *lc = localeconv();

419 ssize_t
420 strfmon_l(char *_RESTRICT_KYWD s, size_t maxsize, locale_t loc,
421          const char *_RESTRICT_KYWD format, ...)
422 {
423     ssize_t ret;
424     va_list ap;
425     va_start(ap, format);
426     ret = strfmon_impl(s, maxsize, loc, format, ap);
427     va_end(ap);
428     return (ret);
429 }

431 static void
432 setup_vars(const struct lc_monetary *lmon, int flags, char *cs_precedes,
433          char *sep_by_space, char *sign_posn, const char **signstr)
434 {
435     if ((flags & IS_NEGATIVE) && (flags & USE_INTL_CURRENCY)) {
436         *cs_precedes = lmon->int_n_cs_precedes[0];
437         *sep_by_space = lmon->int_n_sep_by_space[0];

```

```

438     *sign_posn = (flags & PARENTH_POSN) ? 0 :
439         lmon->int_n_sign_posn[0];
440     *signstr = (lmon->negative_sign[0] == '\0') ? "-" :
441         lmon->negative_sign;
442     *cs_precedes = lc->int_n_cs_precedes;
443     *sep_by_space = lc->int_n_sep_by_space;
444     *sign_posn = (flags & PARENTH_POSN) ? 0 : lc->int_n_sign_posn;
445     *signstr = (lc->negative_sign[0] == '\0') ? "-"
446         : lc->negative_sign;
447     } else if (flags & USE_INTL_CURRENCY) {
448     *cs_precedes = lmon->int_p_cs_precedes[0];
449     *sep_by_space = lmon->int_p_sep_by_space[0];
450     *sign_posn = (flags & PARENTH_POSN) ? 0 :
451         lmon->int_p_sign_posn[0];
452     *signstr = lmon->positive_sign;
453     *cs_precedes = lc->int_p_cs_precedes;
454     *sep_by_space = lc->int_p_sep_by_space;
455     *sign_posn = (flags & PARENTH_POSN) ? 0 : lc->int_p_sign_posn;
456     *signstr = lc->positive_sign;
457     } else if (flags & IS_NEGATIVE) {
458     *cs_precedes = lmon->n_cs_precedes[0];
459     *sep_by_space = lmon->n_sep_by_space[0];
460     *sign_posn = (flags & PARENTH_POSN) ? 0 : lmon->n_sign_posn[0];
461     *signstr = (lmon->negative_sign[0] == '\0') ? "-" :
462         lmon->negative_sign;
463     *cs_precedes = lc->n_cs_precedes;
464     *sep_by_space = lc->n_sep_by_space;
465     *sign_posn = (flags & PARENTH_POSN) ? 0 : lc->n_sign_posn;
466     *signstr = (lc->negative_sign[0] == '\0') ? "-"
467         : lc->negative_sign;
468     } else {
469     *cs_precedes = lmon->p_cs_precedes[0];
470     *sep_by_space = lmon->p_sep_by_space[0];
471     *sign_posn = (flags & PARENTH_POSN) ? 0 : lmon->p_sign_posn[0];
472     *signstr = lmon->positive_sign;
473     *cs_precedes = lc->p_cs_precedes;
474     *sep_by_space = lc->p_sep_by_space;
475     *sign_posn = (flags & PARENTH_POSN) ? 0 : lc->p_sign_posn;
476     *signstr = lc->positive_sign;
477     }

479 /* Set default values for unspecified information. */
480 /* Set default values for unspecified information. */
481 if (*cs_precedes != 0)
482     *cs_precedes = 1;
483 if (*sep_by_space == CHAR_MAX)
484     *sep_by_space = 0;
485 if (*sign_posn == CHAR_MAX)
486     *sign_posn = 0;
487 }

470 static int
471 calc_left_pad(const struct lc_monetary *lmon, int flags, const char *cur_symb)
472 {
473     char cs_precedes, sep_by_space, sign_posn;
474     const char *signstr;

476     char cs_precedes, sep_by_space, sign_posn, *signstr;
475     int left_chars = 0;

477     setup_vars(lmon, flags, &cs_precedes, &sep_by_space, &sign_posn,
478             &signstr);
479     __setup_vars(flags, &cs_precedes, &sep_by_space, &sign_posn, &signstr);

480     if (cs_precedes != 0) {

```



```

481     left_chars += strlen(cur_symb);
482     if (sep_by_space != 0)
483         left_chars++;
484 }

486     switch (sign_posn) {
487     case 1:
488         left_chars += strlen(signstr);
489         break;
490     case 3:
491     case 4:
492         if (cs_precedes != 0)
493             left_chars += strlen(signstr);
494     }
495     return (left_chars);
496 }

498 static int
499 get_groups(int size, const char *grouping)
500 get_groups(int size, char *grouping)
501 {

502     int     chars = 0;

504     if (*grouping == CHAR_MAX || *grouping <= 0) /* no grouping ? */
505         return (0);

507     while (size > (int)*grouping) {
508         chars++;
509         size -= (int)*grouping++;
510         /* no more grouping ? */
511         if (*grouping == CHAR_MAX)
512             break;
513         /* rest grouping with same value ? */
514         if (*grouping == 0) {
515             chars += (size - 1) / *(grouping - 1);
516             break;
517         }
518     }
519     return (chars);
520 }

522 /* convert double to ASCII */
523 static char *
524 format_grouped_double(const struct lc_monetary *lmon,
525     const struct lc_numeric *lnum,
526     double value, int *flags, int left_prec, int right_prec, int pad_char)
527 __format_grouped_double(double value, int *flags,
528     int left_prec, int right_prec, int pad_char)
529 {

529     char     *rslt;
530     char     *avalue;
531     int     avalue_size;
532     char     fmt[32];

534     size_t   bufsize;
535     char     *bufend;

537     int     padded;

539     const char *grouping;
540     const char *decimal_point;
541     const char *thousands_sep;
542     int     decimal_len;
543     int     thousands_len;

```

```

515     struct lconv *lc = localeconv();
516     char *grouping;
517     char decimal_point;
518     char thousands_sep;

545     int groups = 0;

547     grouping = lmon->mon_grouping;
548     decimal_point = lmon->mon_decimal_point;
549     if (*decimal_point == '\0')
550         decimal_point = lnum->decimal_point;
551     thousands_sep = lmon->mon_thousands_sep;
552     if (*thousands_sep == '\0')
553         thousands_sep = lnum->thousands_sep;
554     grouping = lc->mon_grouping;
555     decimal_point = *lc->mon_decimal_point;
556     if (decimal_point == '\0')
557         decimal_point = *lc->decimal_point;
558     thousands_sep = *lc->mon_thousands_sep;
559     if (thousands_sep == '\0')
560         thousands_sep = *lc->thousands_sep;

562     decimal_len = strlen(decimal_point); /* usually 1 */
563     thousands_len = strlen(thousands_sep); /* 0 or 1 usually */

565     /* fill left_prec with default value */
566     if (left_prec == -1)
567         left_prec = 0;

569     /* fill right_prec with default value */
570     if (right_prec == -1) {
571         if (*flags & USE_INTL_CURRENCY)
572             right_prec = lmon->int_frac_digits[0];
573             right_prec = lc->int_frac_digits;
574         else
575             right_prec = lmon->frac_digits[0];
576             right_prec = lc->frac_digits;

578         if (right_prec == CHAR_MAX) /* POSIX locale ? */
579             right_prec = 2;
580     }

582     if (*flags & NEED_GROUPING)
583         left_prec += get_groups(left_prec, grouping);

585     /* convert to string */
586     (void) sprintf(fmt, sizeof (fmt), "%%.d.%df",
587         left_prec + right_prec + 1, right_prec);
588     avalue_size = asprintf(&avalue, fmt, value);
589     if (avalue_size < 0)
590         return (NULL);

592     /*
593     * Make sure that we've enough space for result string.
594     * This assumes that digits take up at least much space as
595     * grouping and radix characters. The worst case currently known
596     * is for Arabic, where two-byte UTF-8 sequences are used for both
597     * decimal and thousands separators, and groups can be as small as two
598     * decimal digits. This will do no worse than doubling the storage
599     * requirement.
600     */
601     /* make sure that we've enough space for result string */
602     bufsize = strlen(avalue)*2+1;
603     rslt = calloc(1, bufsize);
604     if (rslt == NULL) {
605         free(avalue);

```

```

596         return (NULL);
597     }
598     bufend = rslt + bufsize - 1; /* reserve space for trailing '\0' */

600     /* skip spaces at beginning */
601     /* skip spaces at beginning */
602     padded = 0;
603     while (avalue[padded] == ' ') {
604         padded++;
605         avalue_size--;
606     }

607     if (right_prec > 0) {
608         bufend -= right_prec;
609         (void) memcpy(bufend, avalue + avalue_size+padded-right_prec,
610             right_prec);
611         bufend -= decimal_len;
612         (void) memcpy(bufend, decimal_point, decimal_len);
613         avalue_size -= (right_prec + decimal_len);
614         *--bufend = decimal_point;
615         avalue_size -= (right_prec + 1);
616     }

617     if ((*flags & NEED_GROUPING) &&
618         thousands_len != 0 &&
619         thousands_sep != '\0' && /* XXX: need investigation */
620         *grouping != CHAR_MAX &&
621         *grouping > 0) {
622         while (avalue_size > (int)*grouping) {
623             GRPCPY(*grouping);
624             GRPSEP;
625             grouping++;
626         }
627         /* no more grouping ? */
628         if (*grouping == CHAR_MAX)
629             break;
630         /* rest grouping with same value ? */
631         if (*grouping == 0) {
632             grouping--;
633             while (avalue_size > *grouping) {
634                 GRPCPY(*grouping);
635                 GRPSEP;
636             }
637         }
638         if (avalue_size != 0)
639             GRPCPY(avalue_size);
640         padded -= groups;
641     } else {
642         bufend -= avalue_size;
643         (void) memcpy(bufend, avalue+padded, avalue_size);
644         if (right_prec == 0)
645             padded--; /* decrease assumed $decimal_point */
646     }
647 }

648 /* do padding with pad_char */
649 if (padded > 0) {
650     bufend -= padded;
651     (void) memset(bufend, pad_char, padded);
652 }

653
654
655 bufsize = bufsize - (bufend - rslt) + 1;
656 (void) memmove(rslt, bufend, bufsize);
657 free(avalue);

```

```

658         return (rslt);
659     }
660     _____unchanged_portion_omitted_____

```



```

119         pt, ptlim);
120     continue;
121     case 'B':
122         pt = _add((t->tm_mon < 0 ||
123                 t->tm_mon >= MONSPERYEAR) ?
124                 "?" : (tptr->month)[t->tm_mon],
125                 pt, ptlim);
126     continue;
127     case 'b':
128     case 'h':
129         pt = _add((t->tm_mon < 0 ||
130                 t->tm_mon >= MONSPERYEAR) ?
131                 "?" : tptr->mon[t->tm_mon],
132                 pt, ptlim);
133     continue;
134     case 'C':
135         /*
136          * %C used to do a...
137          * _fmt("%a %b %e %X %Y", t);
138          * ..whereas now POSIX 1003.2 calls for
139          * something completely different.
140          * (ado, 1993-05-24)
141          */
142         pt = _yconv(t->tm_year, TM_YEAR_BASE, 1, 0,
143                 pt, ptlim);
144     continue;
145     case 'c':
146         pt = _fmt(loc, tptr->c_fmt, t, pt, ptlim);
147         pt = _fmt(tptr->c_fmt, t, pt, ptlim);
148     continue;
149     case 'D':
150         pt = _fmt(loc, "%m/%d/%y", t, pt, ptlim);
151         pt = _fmt("%m/%d/%y", t, pt, ptlim);
152     continue;
153     case 'd':
154         pt = _conv(t->tm_mday,
155                 PADDING(PAD_FMT_DAYOFMONTH), pt, ptlim);
156     continue;
157     case 'E':
158         if (Ealternative || Oalternative)
159             break;
160         Ealternative++;
161         goto label;
162     case 'O':
163         /*
164          * C99 locale modifiers.
165          * The sequences
166          * %Ec %EC %Ex %EX %Ey %EY
167          * %Od %oe %OH %OI %Om %OM
168          * %OS %Ou %OU %OV %Ow %OW %Oy
169          * are supposed to provide alternate
170          * representations.
171          */
172         if (Ealternative || Oalternative)
173             break;
174         Oalternative++;
175         goto label;
176     case 'e':
177         pt = _conv(t->tm_mday,
178                 PADDING(PAD_FMT_SDAYOFMONTH), pt, ptlim);
179     continue;
180     case 'F':
181         pt = _fmt(loc, "%Y-%m-%d", t, pt, ptlim);
182         pt = _fmt("%Y-%m-%d", t, pt, ptlim);
183     continue;
184     case 'H':

```

```

182         pt = _conv(t->tm_hour, PADDING(PAD_FMT_HMS),
183                 pt, ptlim);
184     continue;
185     case 'I':
186         pt = _conv((t->tm_hour % 12) ?
187                 (t->tm_hour % 12) : 12,
188                 PADDING(PAD_FMT_HMS), pt, ptlim);
189     continue;
190     case 'j':
191         pt = _conv(t->tm_yday + 1,
192                 PADDING(PAD_FMT_DAYOFYEAR), pt, ptlim);
193     continue;
194     case 'k':
195         /*
196          * This used to be...
197          * _conv(t->tm_hour % 12 ?
198          *         t->tm_hour % 12 : 12, 2, ' ');
199          * ...and has been changed to the below to
200          * match SunOS 4.1.1 and Arnold Robbins'
201          * strftime version 3.0. That is, "%k" and
202          * "%l" have been swapped.
203          * (ado, 1993-05-24)
204          */
205         pt = _conv(t->tm_hour,
206                 PADDING(PAD_FMT_SHMS), pt, ptlim);
207     continue;
208     case 'l':
209         /*
210          * This used to be...
211          * _conv(t->tm_hour, 2, ' ');
212          * ...and has been changed to the below to
213          * match SunOS 4.1.1 and Arnold Robbins'
214          * strftime version 3.0. That is, "%k" and
215          * "%l" have been swapped.
216          * (ado, 1993-05-24)
217          */
218         pt = _conv((t->tm_hour % 12) ?
219                 (t->tm_hour % 12) : 12,
220                 PADDING(PAD_FMT_SHMS), pt, ptlim);
221     continue;
222     case 'M':
223         pt = _conv(t->tm_min, PADDING(PAD_FMT_HMS),
224                 pt, ptlim);
225     continue;
226     case 'm':
227         pt = _conv(t->tm_mon + 1,
228                 PADDING(PAD_FMT_MONTH),
229                 pt, ptlim);
230     continue;
231     case 'n':
232         pt = _add("\n", pt, ptlim);
233     continue;
234     case 'p':
235         pt = _add((t->tm_hour >= (HOURSPERDAY / 2)) ?
236                 tptr->pm : tptr->am, pt, ptlim);
237     continue;
238     case 'R':
239         pt = _fmt(loc, "%H:%M", t, pt, ptlim);
240         pt = _fmt("%H:%M", t, pt, ptlim);
241     continue;
242     case 'r':
243         pt = _fmt(loc, tptr->ampm_fmt, t, pt, ptlim);
244         pt = _fmt(tptr->ampm_fmt, t, pt, ptlim);
245     continue;
246     case 'S':
247         pt = _conv(t->tm_sec, PADDING(PAD_FMT_HMS),

```

```

246         pt, ptlim);
247         continue;
249     case 's':
250     {
251         struct tm tm;
252         char *buf;
254
255         tm = *t;
256         (void) asprintf(&buf, "%ld", mktime(&tm));
257         pt = _add(buf, pt, ptlim);
258         continue;
259     }
260     case 'T':
261         pt = _fmt(loc, "%H:%M:%S", t, pt, ptlim);
262         pt = _fmt("%H:%M:%S", t, pt, ptlim);
263         continue;
264     case 't':
265         pt = _add("\t", pt, ptlim);
266         continue;
267     case 'U':
268         pt = _conv((t->tm_yday + DAYSPERWEEK -
269                   t->tm_wday) / DAYSPERWEEK,
270                   PADDING(PAD_FMT_WEEKOFYEAR),
271                   pt, ptlim);
272         continue;
273     case 'u':
274         /*
275          * From Arnold Robbins' strftime version 3.0:
276          * "ISO 8601: Weekday as a decimal number
277          * [1 (Monday) - 7]"
278          * (ado, 1993-05-24)
279          */
280         pt = _conv((t->tm_wday == 0) ?
281                   DAYSPERWEEK : t->tm_wday,
282                   "%d", pt, ptlim);
283         continue;
284     case 'V': /* ISO 8601 week number */
285     case 'G': /* ISO 8601 year (four digits) */
286     case 'g': /* ISO 8601 year (two digits) */
287     /*
288      * From Arnold Robbins' strftime version 3.0: "the week number of the
289      * year (the first Monday as the first day of week 1) as a decimal number
290      * (01-53)."
291      * (ado, 1993-05-24)
292      *
293      * From "http://www.ft.uni-erlangen.de/~mskuhn/iso-time.html" by Markus Kuhn:
294      * "Week 01 of a year is per definition the first week which has the
295      * Thursday in this year, which is equivalent to the week which contains
296      * the fourth day of January. In other words, the first week of a new year
297      * is the week which has the majority of its days in the new year. Week 01
298      * might also contain days from the previous year and the week before week
299      * 01 of a year is the last week (52 or 53) of the previous year even if
300      * it contains days from the new year. A week starts with Monday (day 1)
301      * and ends with Sunday (day 7). For example, the first week of the year
302      * 1997 lasts from 1996-12-30 to 1997-01-05..."
303      * (ado, 1996-01-02)
304      */
305     {
306         int year;
307         int base;
308         int yday;
309         int wday;
310         int w;

```

```

311         year = t->tm_year;
312         base = TM_YEAR_BASE;
313         yday = t->tm_yday;
314         wday = t->tm_wday;
315         for (;;) {
316             int len;
317             int bot;
318             int top;
320
321             len = isleap_sum(year, base) ?
322                 DAYSPERLYEAR : DAYSPERNYEAR;
323             /*
324              * What yday (-3 ... 3) does
325              * the ISO year begin on?
326              */
327             bot = ((yday + 11 - wday) %
328                  DAYSPERWEEK) - 3;
329             /*
330              * What yday does the NEXT
331              * ISO year begin on?
332              */
333             top = bot - (len % DAYSPERWEEK);
334             if (top < -3)
335                 top += DAYSPERWEEK;
336             top += len;
337             if (yday >= top) {
338                 ++base;
339                 w = 1;
340                 break;
341             }
342             if (yday >= bot) {
343                 w = 1 + ((yday - bot) /
344                       DAYSPERWEEK);
345                 break;
346             }
347             --base;
348             yday += isleap_sum(year, base) ?
349                 DAYSPERLYEAR : DAYSPERNYEAR;
350         }
351     #ifdef XPG4_1994_04_09
352     if ((w == 52 && t->tm_mon == TM_JANUARY) ||
353         (w == 1 && t->tm_mon == TM_DECEMBER))
354         w = 53;
355     #endif /* defined XPG4_1994_04_09 */
356     if (*format == 'V')
357         pt = _conv(w,
358                   PADDING(PAD_FMT_WEEKOFYEAR),
359                   pt, ptlim);
360     else if (*format == 'g') {
361         pt = _yconv(year, base, 0, 1,
362                   pt, ptlim);
363     } else
364         pt = _yconv(year, base, 1, 1,
365                   pt, ptlim);
366     }
367     continue;
368     case 'v':
369     /*
370      * From Arnold Robbins' strftime version 3.0:
371      * "date as dd-bbb-YYYY"
372      * (ado, 1993-05-24)
373      */
374     pt = _fmt(loc, "%e-%b-%Y", t, pt, ptlim);
375     pt = _fmt("%e-%b-%Y", t, pt, ptlim);
376     continue;
377     case 'W':

```

```

376         pt = _conv((t->tm_yday + DAYSPERWEEK -
377                    (t->tm_wday ?
378                     (t->tm_wday - 1) :
379                     (DAYSPERWEEK - 1))) / DAYSPERWEEK,
380                    PADDING(PAD_FMT_WEEKOFYEAR),
381                    pt, ptlim);
382         continue;
383     case 'w':
384         pt = _conv(t->tm_wday, "%d", pt, ptlim);
385         continue;
386     case 'X':
387         pt = _fmt(loc, tptr->X_fmt, t, pt, ptlim);
388         pt = _fmt(tptr->X_fmt, t, pt, ptlim);
389         continue;
390     case 'x':
391         pt = _fmt(loc, tptr->x_fmt, t, pt, ptlim);
392         pt = _fmt(tptr->x_fmt, t, pt, ptlim);
393         continue;
394     case 'y':
395         pt = _yconv(t->tm_year, TM_YEAR_BASE, 0, 1,
396                   pt, ptlim);
397         continue;
398     case 'Y':
399         pt = _yconv(t->tm_year, TM_YEAR_BASE, 1, 1,
400                   pt, ptlim);
401         continue;
402     case 'Z':
403         if (t->tm_isdst >= 0)
404             pt = _add(tzname[t->tm_isdst != 0],
405                     pt, ptlim);
406         /*
407          * C99 says that %Z must be replaced by the
408          * empty string if the time zone is not
409          * determinable.
410          */
411         continue;
412     case 'z':
413         {
414             int diff;
415             char const * sign;
416
417             if (t->tm_isdst < 0)
418                 continue;
419             /*
420              * C99 says that the UTC offset must
421              * be computed by looking only at
422              * tm_isdst. This requirement is
423              * incorrect, since it means the code
424              * must rely on magic (in this case
425              * altzone and timezone), and the
426              * magic might not have the correct
427              * offset. Doing things correctly is
428              * tricky and requires disobeying C99;
429              * see GNU C strftime for details.
430              * For now, punt and conform to the
431              * standard, even though it's incorrect.
432              */
433             * C99 says that %z must be replaced by the
434             * empty string if the time zone is not
435             * determinable, so output nothing if the
436             * appropriate variables are not available.
437             */
438             if (t->tm_isdst == 0)
439                 diff = -timezone;
440             else
441                 diff = -altzone;

```

```

440         if (diff < 0) {
441             sign = "-";
442             diff = -diff;
443         } else
444             sign = "+";
445         pt = _add(sign, pt, ptlim);
446         diff /= SECSPERMIN;
447         diff = (diff / MINSPERHOUR) * 100 +
448               (diff % MINSPERHOUR);
449         pt = _conv(diff, PADDING(PAD_FMT_YEAR),
450                   pt, ptlim);
451     }
452     continue;
453     case '+':
454         pt = _fmt(loc, tptr->date_fmt, t, pt, ptlim);
455         pt = _fmt(tptr->date_fmt, t, pt, ptlim);
456         continue;
457     case '-':
458         if (PadIndex != PAD_DEFAULT)
459             break;
460         PadIndex = PAD_LESS;
461         goto label;
462     case '_':
463         if (PadIndex != PAD_DEFAULT)
464             break;
465         PadIndex = PAD_SPACE;
466         goto label;
467     case '0':
468         if (PadIndex != PAD_DEFAULT)
469             break;
470         PadIndex = PAD_ZERO;
471         goto label;
472     case '%':
473         /*
474          * X311J/88-090 (4.12.3.5): if conversion char is
475          * undefined, behavior is undefined. Print out the
476          * character itself as printf(3) also does.
477          */
478         default:
479             break;
480     }
481     if (pt == ptlim)
482         break;
483     *pt++ = *format;
484 }
485 return (pt);
486 }

```

unchanged\_portion\_omitted

new/usr/src/lib/libc/port/locale/strncasecmp.c

1

```
*****
2131 Fri Jul 11 11:43:03 2014
new/usr/src/lib/libc/port/locale/strncasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
24 * Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
25 */

27 /*      Copyright (c) 1983, 1984, 1985, 1986, 1987, 1988, 1989 AT&T      */
28 /*      All Rights Reserved      */

30 /*
31 * Portions of this source code were derived from Berkeley 4.3 BSD
32 * under license from the Regents of the University of California.
33 */

35 #include "lint.h"
36 #include <sys/types.h>
37 #include <strings.h>
38 #include <ctype.h>
39 #include <locale.h>
40 #include "lctype.h"
41 #include "localeimpl.h"

43 int
44 strncasecmp_l(const char *s1, const char *s2, size_t n, locale_t loc)
45 strncasecmp(const char *s1, const char *s2, size_t n)
46 {
47     extern int charset_is_ascii;
48     extern int ascii_strncasecmp(const char *s1, const char *s2, size_t n);
49     const int *cm;
50     int *cm;
51     int *cm;
52     const uchar_t *us1;
53     const uchar_t *us2;
54     const struct lc_ctype *lct = loc->ctype;

55     /*
56      * If we are in a locale that uses the ASCII character set
57      * (C or POSIX), use the fast ascii_strncasecmp() function.
58      */
```

new/usr/src/lib/libc/port/locale/strncasecmp.c

2

```
56     if (lct->lc_is_ascii)
57     if (charset_is_ascii)
58         return (ascii_strncasecmp(s1, s2, n));

59     cm = lct->lc_trans_lower;
60     cm = __trans_lower;
61     us1 = (const uchar_t *)s1;
62     us2 = (const uchar_t *)s2;

63     while (n != 0 && cm[*us1] == cm[*us2++]) {
64         if (*us1++ == '\0')
65             return (0);
66         n--;
67     }
68     return (n == 0 ? 0 : cm[*us1] - cm[*us2 - 1]);
69 }

71 int
72 strncasecmp(const char *s1, const char *s2, size_t n)
73 {
74     return (strncasecmp_l(s1, s2, n, uselocale(NULL)));
75 }

_____ unchanged_portion_omitted_____
```

```

*****
11596 Fri Jul 11 11:43:03 2014
new/usr/src/lib/libc/port/locale/strptime.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright (c) 2014 Gary Mills
3  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
4  * Copyright 2011, Nexenta Systems, Inc. All rights reserved.
5  * Copyright (c) 1994 Powerdog Industries. All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 *
11 * 1. Redistributions of source code must retain the above copyright
12 *    notice, this list of conditions and the following disclaimer.
13 *
14 * 2. Redistributions in binary form must reproduce the above copyright
15 *    notice, this list of conditions and the following disclaimer
16 *    in the documentation and/or other materials provided with the
17 *    distribution.
18 *
19 * THIS SOFTWARE IS PROVIDED BY POWERDOG INDUSTRIES ``AS IS'' AND ANY
20 * EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
21 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
22 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE POWERDOG INDUSTRIES BE
23 * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
24 * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
25 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
26 * BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
27 * WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
28 * OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
29 * EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
30 *
31 * The views and conclusions contained in the software and documentation
32 * are those of the authors and should not be interpreted as representing
33 * official policies, either expressed or implied, of Powerdog Industries.
34 */

36 #include "lint.h"
37 #include <time.h>
38 #include <ctype.h>
39 #include <errno.h>
40 #include <stdlib.h>
41 #include <string.h>
42 #include <pthread.h>
43 #include <alloca.h>
44 #include <locale.h>
45 #include "timelocal.h"
46 #include "localeimpl.h"

48 #define asizeof(a)      (sizeof (a) / sizeof ((a)[0]))

50 #define F_GMT           (1 << 0)
48 #define F_ZERO         (1 << 1)
51 #define F_RECURSE     (1 << 2)

53 static char *
54 __strptime(const char * _RESTRICT_KYWD buf, const char * _RESTRICT_KYWD fmt,
55            struct tm * _RESTRICT_KYWD tm, int * _RESTRICT_KYWD flagsp,
56            locale_t _RESTRICT_KYWD loc)
52 __strptime(const char *buf, const char *fmt, struct tm *tm, int *flagsp)

```

```

57 {
58     char    c;
59     const char *ptr;
60     int     i, len, recurse = 0;
61     int     Ealternative, Oalternative;
62     const struct lc_time *tptr = loc->time;
58     struct lc_time_T *tpr = __get_current_time_locale();

64     if (*flagsp & F_RECURSE)
65         recurse = 1;
66     *flagsp |= F_RECURSE;

64     if (*flagsp & F_ZERO)
65         (void) memset(tm, 0, sizeof (*tm));
66     *flagsp &= ~F_ZERO;

68     ptr = fmt;
69     while (*ptr != 0) {
70         if (*buf == 0)
71             break;

73         c = *ptr++;

75         if (c != '%') {
76             if (isspace(c))
77                 while (isspace(*buf))
78                     buf++;
79             else if (c != *buf++)
80                 return (NULL);
81             continue;
82         }

84         Ealternative = 0;
85         Oalternative = 0;
86 label:
87         c = *ptr++;
88         switch (c) {
89             case 0:
90             case '%':
91                 if (*buf++ != '%')
92                     return (NULL);
93                 break;

95             case '+':
96                 buf = __strptime(buf, tptr->date_fmt, tm, flagsp, loc);
96                 buf = __strptime(buf, tptr->date_fmt, tm, flagsp);
97                 if (buf == NULL)
98                     return (NULL);
99                 break;

101             case 'C':
102                 if (!isdigit(*buf))
103                     return (NULL);

105                 /* XXX This will break for 3-digit centuries. */
106                 len = 2;
107                 for (i = 0; len && isdigit(*buf); buf++) {
108                     i *= 10;
109                     i += *buf - '0';
110                     len--;
111                 }
112                 if (i < 19)
113                     return (NULL);

115                 tm->tm_year = i * 100 - 1900;
116                 break;

```



```

118     case 'c':
119         buf = __strptime(buf, tptr->c_fmt, tm, flagsp, loc);
119         buf = __strptime(buf, tptr->c_fmt, tm, flagsp);
120         if (buf == NULL)
121             return (NULL);
122         break;

124     case 'D':
125         buf = __strptime(buf, "%m/%d/%y", tm, flagsp, loc);
125         buf = __strptime(buf, "%m/%d/%y", tm, flagsp);
126         if (buf == NULL)
127             return (NULL);
128         break;

130     case 'E':
131         if (Ealternative || Oalternative)
132             break;
133         Ealternative++;
134         goto label;

136     case 'O':
137         if (Ealternative || Oalternative)
138             break;
139         Oalternative++;
140         goto label;

142     case 'F':
143         buf = __strptime(buf, "%Y-%m-%d", tm, flagsp, loc);
143         buf = __strptime(buf, "%Y-%m-%d", tm, flagsp);
144         if (buf == NULL)
145             return (NULL);
146         break;

148     case 'R':
149         buf = __strptime(buf, "%H:%M", tm, flagsp, loc);
149         buf = __strptime(buf, "%H:%M", tm, flagsp);
150         if (buf == NULL)
151             return (NULL);
152         break;

154     case 'r':
155         buf = __strptime(buf, tptr->ampm_fmt, tm, flagsp, loc);
155         buf = __strptime(buf, tptr->ampm_fmt, tm, flagsp);
156         if (buf == NULL)
157             return (NULL);
158         break;

160     case 'T':
161         buf = __strptime(buf, "%H:%M:%S", tm, flagsp, loc);
161         buf = __strptime(buf, "%H:%M:%S", tm, flagsp);
162         if (buf == NULL)
163             return (NULL);
164         break;

166     case 'X':
167         buf = __strptime(buf, tptr->X_fmt, tm, flagsp, loc);
167         buf = __strptime(buf, tptr->X_fmt, tm, flagsp);
168         if (buf == NULL)
169             return (NULL);
170         break;

172     case 'x':
173         buf = __strptime(buf, tptr->x_fmt, tm, flagsp, loc);
173         buf = __strptime(buf, tptr->x_fmt, tm, flagsp);
174         if (buf == NULL)

```

```

175         return (NULL);
176         break;

178     case 'j':
179         if (!isdigit(*buf))
180             return (NULL);

182         len = 3;
183         for (i = 0; len && isdigit(*buf); buf++) {
184             i *= 10;
185             i += *buf - '0';
186             len--;
187         }
188         if (i < 1 || i > 366)
189             return (NULL);

191         tm->tm_yday = i - 1;
192         break;

194     case 'M':
195     case 'S':
196         if (*buf == 0 || isspace(*buf))
197             break;

199         if (!isdigit(*buf))
200             return (NULL);

202         len = 2;
203         for (i = 0; len && isdigit(*buf); buf++) {
204             i *= 10;
205             i += *buf - '0';
206             len--;
207         }

209         if (c == 'M') {
210             if (i > 59)
211                 return (NULL);
212             tm->tm_min = i;
213         } else {
214             if (i > 60)
215                 return (NULL);
216             tm->tm_sec = i;
217         }

219         break;

221     case 'H':
222     case 'I':
223     case 'k':
224     case 'l':
225         /*
226          * Of these, %l is the only specifier explicitly
227          * documented as not being zero-padded. However,
228          * there is no harm in allowing zero-padding.
229          *
230          * XXX The %l specifier may gobble one too many
231          * digits if used incorrectly.
232          */
233         if (!isdigit(*buf))
234             return (NULL);

236         len = 2;
237         for (i = 0; len && isdigit(*buf); buf++) {
238             i *= 10;
239             i += *buf - '0';
240             len--;

```

```

241     }
242     if (c == 'H' || c == 'k') {
243         if (i > 23)
244             return (NULL);
245     } else if (i > 12)
246         return (NULL);
247
248     tm->tm_hour = i;
249
250     break;
251
252 case 'p':
253     /*
254     * XXX This is bogus if parsed before hour-related
255     * specifiers.
256     */
257     len = strlen(tptr->am);
258     if (strncasecmp(buf, tptr->am, len) == 0) {
259         if (tm->tm_hour > 12)
260             return (NULL);
261         if (tm->tm_hour == 12)
262             tm->tm_hour = 0;
263         buf += len;
264         break;
265     }
266
267     len = strlen(tptr->pm);
268     if (strncasecmp(buf, tptr->pm, len) == 0) {
269         if (tm->tm_hour > 12)
270             return (NULL);
271         if (tm->tm_hour != 12)
272             tm->tm_hour += 12;
273         buf += len;
274         break;
275     }
276
277     return (NULL);
278
279 case 'A':
280 case 'a':
281     for (i = 0; i < asizeof(tptr->weekday); i++) {
282         len = strlen(tptr->weekday[i]);
283         if (strncasecmp(buf, tptr->weekday[i], len) ==
284             0)
285             break;
286         len = strlen(tptr->wday[i]);
287         if (strncasecmp(buf, tptr->wday[i], len) == 0)
288             break;
289     }
290     if (i == asizeof(tptr->weekday))
291         return (NULL);
292
293     tm->tm_wday = i;
294     buf += len;
295     break;
296
297 case 'U':
298 case 'W':
299     /*
300     * XXX This is bogus, as we can not assume any valid
301     * information present in the tm structure at this
302     * point to calculate a real value, so just check the
303     * range for now.
304     */
305     if (!isdigit(*buf))
306         return (NULL);

```

```

308     len = 2;
309     for (i = 0; len && isdigit(*buf); buf++) {
310         i *= 10;
311         i += *buf - '0';
312         len--;
313     }
314     if (i > 53)
315         return (NULL);
316
317     break;
318
319 case 'w':
320     if (!isdigit(*buf))
321         return (NULL);
322
323     i = *buf - '0';
324     if (i > 6)
325         return (NULL);
326
327     tm->tm_wday = i;
328
329     break;
330
331 case 'd':
332 case 'e':
333     /*
334     * The %e format has a space before single digits
335     * which we need to skip.
336     */
337     if (isspace(*buf))
338         buf++;
339
340     /*
341     * The %e specifier is explicitly documented as not
342     * being zero-padded but there is no harm in allowing
343     * such padding.
344     *
345     * XXX The %e specifier may gobble one too many
346     * digits if used incorrectly.
347     */
348     if (!isdigit(*buf))
349         return (NULL);
350
351     len = 2;
352     for (i = 0; len && isdigit(*buf); buf++) {
353         i *= 10;
354         i += *buf - '0';
355         len--;
356     }
357     if (i > 31)
358         return (NULL);
359
360     tm->tm_mday = i;
361
362     break;
363
364 case 'B':
365 case 'b':
366 case 'h':
367     for (i = 0; i < asizeof(tptr->month); i++) {
368         len = strlen(tptr->month[i]);
369         if (strncasecmp(buf, tptr->month[i], len) == 0)
370             break;
371     }
372     /*
373     * Try the abbreviated month name if the full name

```

```

373     * wasn't found.
374     */
375     if (i == asizeof(tptr->month)) {
376         for (i = 0; i < asizeof(tptr->month); i++) {
377             len = strlen(tptr->mon[i]);
378             if (strncasecmp(buf, tptr->mon[i],
379                 len) == 0)
380                 break;
381         }
382     }
383     if (i == asizeof(tptr->month))
384         return (NULL);

386     tm->tm_mon = i;
387     buf += len;
388     break;

390 case 'm':
391     if (!isdigit(*buf))
392         return (NULL);

394     len = 2;
395     for (i = 0; len && isdigit(*buf); buf++) {
396         i *= 10;
397         i += *buf - '0';
398         len--;
399     }
400     if (i < 1 || i > 12)
401         return (NULL);

403     tm->tm_mon = i - 1;

405     break;

407 case 's':
408     {
409         char *cp;
410         int sverrno;
411         time_t t;

413         sverrno = errno;
414         errno = 0;
415         t = strtol(buf, &cp, 10);
416         if (errno == ERANGE) {
417             errno = sverrno;
418             return (NULL);
419         }
420         errno = sverrno;
421         buf = cp;
422         (void) gmtime_r(&t, tm);
423         *flagsp |= F_GMT;
424     }
425     break;

427 case 'Y':
428 case 'y':
429     if (*buf == NULL || isspace(*buf))
430         break;

432     if (!isdigit(*buf))
433         return (NULL);

435     len = (c == 'Y') ? 4 : 2;
436     for (i = 0; len && isdigit(*buf); buf++) {
437         i *= 10;
438         i += *buf - '0';

```

```

439         len--;
440     }
441     if (c == 'Y')
442         i -= 1900;
443     if (c == 'y' && i < 69)
444         i += 100;
445     if (i < 0)
446         return (NULL);

448     tm->tm_year = i;

450     break;

452 case 'Z':
453     {
454         const char *cp = buf;
455         char *zonestr;

457         while (isupper(*cp))
458             ++cp;
459         if (cp - buf) {
460             zonestr = alloca(cp - buf + 1);
461             (void) strncpy(zonestr, buf, cp - buf);
462             zonestr[cp - buf] = '\0';
463             tzset();
464             if (strcmp(zonestr, "GMT") == 0) {
465                 *flagsp |= F_GMT;
466             } else if (0 == strcmp(zonestr, tzname[0])) {
467                 tm->tm_isdst = 0;
468             } else if (0 == strcmp(zonestr, tzname[1])) {
469                 tm->tm_isdst = 1;
470             } else {
471                 return (NULL);
472             }
473             buf += cp - buf;
474         }
475     }
476     break;

478 case 'z':
479     {
480         int sign = 1;

482         if (*buf != '+') {
483             if (*buf == '-')
484                 sign = -1;
485             else
486                 return (NULL);
487         }
488         buf++;
489         i = 0;
490         for (len = 4; len > 0; len--) {
491             if (!isdigit(*buf))
492                 return (NULL);
493             i *= 10;
494             i += *buf - '0';
495             buf++;
496         }

498         tm->tm_hour -= sign * (i / 100);
499         tm->tm_min -= sign * (i % 100);
500         *flagsp |= F_GMT;
501     }
502     break;

503 case 'n':
504 case 't':

```

```

505         while (isspace(*buf))
506             buf++;
507         break;
508     }
509 }

511 if (!recurse) {
512     if (buf && (*flagsp & F_GMT)) {
513         time_t t = timegm(tm);
514         (void) localtime_r(&t, tm);
515     }
516 }

518 return ((char *)buf);
519 }

521 char *
522 strptime(const char *_RESTRICT_KYWD buf, const char *_RESTRICT_KYWD fmt,
523          struct tm *_RESTRICT_KYWD tm)
524 strptime(const char *buf, const char *fmt, struct tm *tm)
524 {
525     int    flags = 0;
526     int    flags = F_ZERO;

527     (void) memset(tm, 0, sizeof (*tm));

529     return (__strptime(buf, fmt, tm, &flags, uselocale(NULL)));
526     return (__strptime(buf, fmt, tm, &flags));
530 }

532 /*
533  * This is used by Solaris, and is a variant that does not clear the
534  * incoming tm. It is triggered by -D_STRPTIME_DONTZERO.
535  */
536 char *
537 __strptime_dontzero(const char *_RESTRICT_KYWD buf,
538                    const char *_RESTRICT_KYWD fmt, struct tm *_RESTRICT_KYWD tm)
539 __strptime_dontzero(const char *buf, const char *fmt, struct tm *tm)
539 {
540     int    flags = 0;

542     return (__strptime(buf, fmt, tm, &flags, uselocale(NULL)));
543 }

545 /*
546  * strptime_l is an extension that seems natural, and indeed, MacOS X
547  * includes it within their <xlocale.h> and it is part of GNU libc as well.
548  * For now we restrict it to the cases where strict namespaces are not
549  * included. We expect to see it in a future version of POSIX. locale_t is
550  * not a restrict, since the spec for it doesn't assume its a pointer. We
551  * therefore pass it analogously to the way strftime_l is specified.
552  *
553  * We are not providing a non-zeroing version at this time.
554  */
555 char *
556 strptime_l(const char *_RESTRICT_KYWD buf, const char *_RESTRICT_KYWD fmt,
557            struct tm *_RESTRICT_KYWD tm, locale_t loc)
558 {
559     int    flags = 0;

561     (void) memset(tm, 0, sizeof (*tm));

563     return (__strptime(buf, fmt, tm, &flags, loc));
538     return (__strptime(buf, fmt, tm, &flags));
564 }

```

unchanged\_portion\_omitted

```

*****
2747 Fri Jul 11 11:43:03 2014
new/usr/src/lib/libc/port/locale/strxfrm.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1995 Alex Tatmanjants <alex@elvisti.kiev.ua>
5  *      at Electronni Visti IA, Kiev, Ukraine.
6  *      All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include <stdlib.h>
32 #include <string.h>
33 #include <errno.h>
34 #include <wchar.h>
35 #include <assert.h>
36 #include <xlocale.h>
37 #include "collate.h"

39 size_t
40 strxfrm_l(char *_RESTRICT_KYWD xf, const char *_RESTRICT_KYWD src,
41           size_t dlen, locale_t loc)
38 strxfrm(char *_RESTRICT_KYWD xf, const char *_RESTRICT_KYWD src, size_t dlen)
42 {
43     size_t slen;
44     size_t xlen;
45     wchar_t *wcs = NULL;

47     if (!*src) {
48         if (dlen > 0)
49             *xf = '\0';
50         return (0);
51     }

53     /*
54     * The conversion from multibyte to wide character strings is
55     * strictly reducing (one byte of an mbs cannot expand to more
56     * than one wide character.)
57     */

```

```

58     slen = strlen(src);

60     if (loc->collate->lc_is_posix)
57     if (!_collate_load_error)
61         goto error;

63     if ((wcs = malloc((slen + 1) * sizeof (wchar_t))) == NULL)
64         goto error;

66     if (mbstowcs_l(wcs, src, slen + 1, loc) == (size_t)-1)
63     if (mbstowcs(wcs, src, slen + 1) == (size_t)-1)
67         goto error;

69     if ((xlen = _collate_sxfrm(wcs, xf, dlen, loc)) == (size_t)-1)
66     if ((xlen = _collate_sxfrm(wcs, xf, dlen)) == (size_t)-1)
70         goto error;

72     if (wcs)
73         free(wcs);

75     if (dlen > xlen) {
76         xf[xlen] = 0;
77     } else if (dlen) {
78         xf[dlen-1] = 0;
79     }

81     return (xlen);

83 error:
84     /* errno should be set to ENOMEM if malloc failed */
85     if (wcs)
86         free(wcs);
87     (void) strcpy(xf, src, dlen);

89     return (slen);
90 }

92 size_t
93 strxfrm(char *_RESTRICT_KYWD xf, const char *_RESTRICT_KYWD src, size_t dlen)
94 {
95     return (strxfrm_l(xf, src, dlen, uselocale(NULL)));
96 }

    _____
    unchanged_portion_omitted_

```

```

*****
9592 Fri Jul 11 11:43:03 2014
new/usr/src/lib/libc/port/locale/table.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2 * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3 * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4 * Copyright (c) 1993
5 * The Regents of the University of California. All rights reserved.
6 *
7 * This code is derived from software contributed to Berkeley by
8 * Paul Borman at Krystal Technologies.
9 *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 * notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 * notice, this list of conditions and the following disclaimer in the
17 * documentation and/or other materials provided with the distribution.
18 * 4. Neither the name of the University nor the names of its contributors
19 * may be used to endorse or promote products derived from this software
20 * without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */

35 #include "lint.h"
36 #include <ctype.h>
37 #include <wchar.h>
38 #include "runetype.h"
39 #include "mblocal.h"
40 #include "_ctype.h"

42 _RuneLocale _DefaultRuneLocale = {
43     _RUNE_MAGIC_1,
44     "NONE",
45     {
46         /* 00 */
47         _CTYPE_C,
48         _CTYPE_C,
49         _CTYPE_C,
50         _CTYPE_C,
51         _CTYPE_C,
52         _CTYPE_C,
53         _CTYPE_C,
54         _CTYPE_C,
55         /* 08 */
56         _CTYPE_C,
57         _CTYPE_C|_CTYPE_S|_CTYPE_B,
58         _CTYPE_C|_CTYPE_S,

```

```

59         _CTYPE_C|_CTYPE_S,
60         _CTYPE_C|_CTYPE_S,
61         _CTYPE_C|_CTYPE_S,
62         _CTYPE_C,
63         _CTYPE_C,
64         /* 10 */
65         _CTYPE_C,
66         _CTYPE_C,
67         _CTYPE_C,
68         _CTYPE_C,
69         _CTYPE_C,
70         _CTYPE_C,
71         _CTYPE_C,
72         _CTYPE_C,
73         /* 18 */
74         _CTYPE_C,
75         _CTYPE_C,
76         _CTYPE_C,
77         _CTYPE_C,
78         _CTYPE_C,
79         _CTYPE_C,
80         _CTYPE_C,
81         _CTYPE_C,
82         /* 20 */
83         _CTYPE_S|_CTYPE_B|_CTYPE_R,
84         _CTYPE_P|_CTYPE_R|_CTYPE_G,
85         _CTYPE_P|_CTYPE_R|_CTYPE_G,
86         _CTYPE_P|_CTYPE_R|_CTYPE_G,
87         _CTYPE_P|_CTYPE_R|_CTYPE_G,
88         _CTYPE_P|_CTYPE_R|_CTYPE_G,
89         _CTYPE_P|_CTYPE_R|_CTYPE_G,
90         _CTYPE_P|_CTYPE_R|_CTYPE_G,
91         /* 28 */
92         _CTYPE_P|_CTYPE_R|_CTYPE_G,
93         _CTYPE_P|_CTYPE_R|_CTYPE_G,
94         _CTYPE_P|_CTYPE_R|_CTYPE_G,
95         _CTYPE_P|_CTYPE_R|_CTYPE_G,
96         _CTYPE_P|_CTYPE_R|_CTYPE_G,
97         _CTYPE_P|_CTYPE_R|_CTYPE_G,
98         _CTYPE_P|_CTYPE_R|_CTYPE_G,
99         _CTYPE_P|_CTYPE_R|_CTYPE_G,
100        /* 30 */
101        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
102        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
103        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
104        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
105        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
106        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
107        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
108        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
109        /* 38 */
110        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
111        _CTYPE_D|_CTYPE_R|_CTYPE_G|_CTYPE_X,
112        _CTYPE_P|_CTYPE_R|_CTYPE_G,
113        _CTYPE_P|_CTYPE_R|_CTYPE_G,
114        _CTYPE_P|_CTYPE_R|_CTYPE_G,
115        _CTYPE_P|_CTYPE_R|_CTYPE_G,
116        _CTYPE_P|_CTYPE_R|_CTYPE_G,
117        _CTYPE_P|_CTYPE_R|_CTYPE_G,
118        /* 40 */
119        _CTYPE_P|_CTYPE_R|_CTYPE_G,
120        _CTYPE_U|_CTYPE_X|_CTYPE_R|_CTYPE_G|_CTYPE_A,
121        _CTYPE_U|_CTYPE_X|_CTYPE_R|_CTYPE_G|_CTYPE_A,
122        _CTYPE_U|_CTYPE_X|_CTYPE_R|_CTYPE_G|_CTYPE_A,
123        _CTYPE_U|_CTYPE_X|_CTYPE_R|_CTYPE_G|_CTYPE_A,
124        _CTYPE_U|_CTYPE_X|_CTYPE_R|_CTYPE_G|_CTYPE_A,

```

```

125 _CTYPE_U _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
126 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
127 /* 48 */
128 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
129 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
130 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
131 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
132 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
133 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
134 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
135 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
136 /* 50 */
137 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
138 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
139 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
140 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
141 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
142 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
143 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
144 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
145 /* 58 */
146 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
147 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
148 _CTYPE_U _CTYPE_R _CTYPE_G _CTYPE_A,
149 _CTYPE_P _CTYPE_R _CTYPE_G,
150 _CTYPE_P _CTYPE_R _CTYPE_G,
151 _CTYPE_P _CTYPE_R _CTYPE_G,
152 _CTYPE_P _CTYPE_R _CTYPE_G,
153 _CTYPE_P _CTYPE_R _CTYPE_G,
154 /* 60 */
155 _CTYPE_P _CTYPE_R _CTYPE_G,
156 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
157 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
158 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
159 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
160 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
161 _CTYPE_L _CTYPE_X _CTYPE_R _CTYPE_G _CTYPE_A,
162 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
163 /* 68 */
164 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
165 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
166 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
167 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
168 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
169 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
170 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
171 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
172 /* 70 */
173 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
174 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
175 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
176 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
177 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
178 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
179 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
180 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
181 /* 78 */
182 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
183 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
184 _CTYPE_L _CTYPE_R _CTYPE_G _CTYPE_A,
185 _CTYPE_P _CTYPE_R _CTYPE_G,
186 _CTYPE_P _CTYPE_R _CTYPE_G,
187 _CTYPE_P _CTYPE_R _CTYPE_G,
188 _CTYPE_P _CTYPE_R _CTYPE_G,
189 _CTYPE_C,
190 }

```

```

192 /* BEGIN CSTYLED */
193 {
194 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07,
195 0x08, 0x09, 0x0a, 0x0b, 0x0c, 0x0d, 0x0e, 0x0f,
196 0x10, 0x11, 0x12, 0x13, 0x14, 0x15, 0x16, 0x17,
197 0x18, 0x19, 0x1a, 0x1b, 0x1c, 0x1d, 0x1e, 0x1f,
198 0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26, 0x27,
199 0x28, 0x29, 0x2a, 0x2b, 0x2c, 0x2d, 0x2e, 0x2f,
200 0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37,
201 0x38, 0x39, 0x3a, 0x3b, 0x3c, 0x3d, 0x3e, 0x3f,
202 0x40, 'a', 'b', 'c', 'd', 'e', 'f', 'g',
203 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
204 'p', 'q', 'r', 's', 't', 'u', 'v', 'w',
205 'x', 'y', 'z', 0x5b, 0x5c, 0x5d, 0x5e, 0x5f,
206 0x60, 'a', 'b', 'c', 'd', 'e', 'f', 'g',
207 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
208 'p', 'q', 'r', 's', 't', 'u', 'v', 'w',
209 'x', 'y', 'z', 0x7b, 0x7c, 0x7d, 0x7e, 0x7f,
210 0x80, 0x81, 0x82, 0x83, 0x84, 0x85, 0x86, 0x87,
211 0x88, 0x89, 0x8a, 0x8b, 0x8c, 0x8d, 0x8e, 0x8f,
212 0x90, 0x91, 0x92, 0x93, 0x94, 0x95, 0x96, 0x97,
213 0x98, 0x99, 0x9a, 0x9b, 0x9c, 0x9d, 0x9e, 0x9f,
214 0xa0, 0xa1, 0xa2, 0xa3, 0xa4, 0xa5, 0xa6, 0xa7,
215 0xa8, 0xa9, 0xaa, 0xab, 0xac, 0xad, 0xae, 0xaf,
216 0xb0, 0xb1, 0xb2, 0xb3, 0xb4, 0xb5, 0xb6, 0xb7,
217 0xb8, 0xb9, 0xba, 0xbb, 0xbc, 0xbd, 0xbe, 0xbf,
218 0xc0, 0xc1, 0xc2, 0xc3, 0xc4, 0xc5, 0xc6, 0xc7,
219 0xc8, 0xc9, 0xca, 0xcb, 0xcc, 0xcd, 0xce, 0xcf,
220 0xd0, 0xd1, 0xd2, 0xd3, 0xd4, 0xd5, 0xd6, 0xd7,
221 0xd8, 0xd9, 0xda, 0xdb, 0xdc, 0xdd, 0xde, 0xdf,
222 0xe0, 0xe1, 0xe2, 0xe3, 0xe4, 0xe5, 0xe6, 0xe7,
223 0xe8, 0xe9, 0xea, 0xeb, 0xec, 0xed, 0xee, 0xef,
224 0xf0, 0xf1, 0xf2, 0xf3, 0xf4, 0xf5, 0xf6, 0xf7,
225 0xf8, 0xf9, 0xfa, 0xfb, 0xfc, 0xfd, 0xfe, 0xff,
226 },
227 {
228 0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07,
229 0x08, 0x09, 0x0a, 0x0b, 0x0c, 0x0d, 0x0e, 0x0f,
230 0x10, 0x11, 0x12, 0x13, 0x14, 0x15, 0x16, 0x17,
231 0x18, 0x19, 0x1a, 0x1b, 0x1c, 0x1d, 0x1e, 0x1f,
232 0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26, 0x27,
233 0x28, 0x29, 0x2a, 0x2b, 0x2c, 0x2d, 0x2e, 0x2f,
234 0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37,
235 0x38, 0x39, 0x3a, 0x3b, 0x3c, 0x3d, 0x3e, 0x3f,
236 0x40, 'A', 'B', 'C', 'D', 'E', 'F', 'G',
237 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',
238 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W',
239 'X', 'Y', 'Z', 0x5b, 0x5c, 0x5d, 0x5e, 0x5f,
240 0x60, 'A', 'B', 'C', 'D', 'E', 'F', 'G',
241 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O',
242 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W',
243 'X', 'Y', 'Z', 0x7b, 0x7c, 0x7d, 0x7e, 0x7f,
244 0x80, 0x81, 0x82, 0x83, 0x84, 0x85, 0x86, 0x87,
245 0x88, 0x89, 0x8a, 0x8b, 0x8c, 0x8d, 0x8e, 0x8f,
246 0x90, 0x91, 0x92, 0x93, 0x94, 0x95, 0x96, 0x97,
247 0x98, 0x99, 0x9a, 0x9b, 0x9c, 0x9d, 0x9e, 0x9f,
248 0xa0, 0xa1, 0xa2, 0xa3, 0xa4, 0xa5, 0xa6, 0xa7,
249 0xa8, 0xa9, 0xaa, 0xab, 0xac, 0xad, 0xae, 0xaf,
250 0xb0, 0xb1, 0xb2, 0xb3, 0xb4, 0xb5, 0xb6, 0xb7,
251 0xb8, 0xb9, 0xba, 0xbb, 0xbc, 0xbd, 0xbe, 0xbf,
252 0xc0, 0xc1, 0xc2, 0xc3, 0xc4, 0xc5, 0xc6, 0xc7,
253 0xc8, 0xc9, 0xca, 0xcb, 0xcc, 0xcd, 0xce, 0xcf,
254 0xd0, 0xd1, 0xd2, 0xd3, 0xd4, 0xd5, 0xd6, 0xd7,
255 0xd8, 0xd9, 0xda, 0xdb, 0xdc, 0xdd, 0xde, 0xdf,
256 0xe0, 0xe1, 0xe2, 0xe3, 0xe4, 0xe5, 0xe6, 0xe7,

```

```
257     0xe8,  0xe9,  0xea,  0xeb,  0xec,  0xed,  0xee,  0xef,  
258     0xf0,  0xf1,  0xf2,  0xf3,  0xf4,  0xf5,  0xf6,  0xf7,  
259     0xf8,  0xf9,  0xfa,  0xfb,  0xfc,  0xfd,  0xfe,  0xff,  
260     },  
261     /* END CSTYLED */  
262 };  
  
263 _RuneLocale *_CurrentRuneLocale = &_DefaultRuneLocale;  
  
264 /* Taken from former _ctype.c */  
265 unsigned int *_ctype_mask = _DefaultRuneLocale.__runetype;  
  
267 int *__trans_lower = _DefaultRuneLocale.__maplower;  
268 int *__trans_upper = _DefaultRuneLocale.__mapupper;  
  
271 /*  
272  * Used in various string routines to conditionalize versions optimized for  
273  * the ASCII case  
274  */  
275 int charset_is_ascii = 1;
```



```

*****
3167 Fri Jul 11 11:43:03 2014
new/usr/src/lib/libc/port/locale/timelocal.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2001 Alexey Zelkin <phantom@FreeBSD.org>
5  * Copyright (c) 1997 FreeBSD Inc.
6  * All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include <stddef.h>
32 #include <errno.h>
33 #include "ldpart.h"
34 #include "timelocal.h"
35 #include "localeimpl.h"

37 #define LCTIME_SIZE (sizeof (struct lc_time) / sizeof (char *))
34 static struct lc_time_T _time_locale;
35 static int _time_using_locale;
36 static char *time_locale_buf;

39 struct lc_time lc_time_posix = {
38 #define LCTIME_SIZE (sizeof (struct lc_time_T) / sizeof (char *))

40 static const struct lc_time_T _C_time_locale = {
40 {
41     "Jan", "Feb", "Mar", "Apr", "May", "Jun",
42     "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
43 }, {
44     "January", "February", "March", "April", "May", "June",
45     "July", "August", "September", "October", "November", "December"
46 }, {
47     "Sun", "Mon", "Tue", "Wed",
48     "Thu", "Fri", "Sat"
49 }, {
50     "Sunday", "Monday", "Tuesday", "Wednesday",
51     "Thursday", "Friday", "Saturday"
52 },

```

```

54 /* X_fmt */
55 "%H:%M:%S",

57 /*
58  * x_fmt
59  * Since the C language standard calls for
60  * "date, using locale's date format," anything goes.
61  * Using just numbers (as here) makes Quakers happier;
62  * it's also compatible with SVR4.
63  */
64 "%m/%d/%y",

66 /*
67  * c_fmt
68  */
69 "%a %b %e %H:%M:%S %Y",

71 /* am */
72 "AM",

74 /* pm */
75 "PM",

77 /* date_fmt */
78 "%a %b %e %H:%M:%S %Z %Y",

80 /*
81  * ampm_fmt - To determine 12-hour clock format time (empty, if N/A)
82  */
83 "%I:%M:%S %p"
84 };

86 struct locdata __posix_time_locdata = {
87     .l_lname = "C",
88     .l_data = { &lc_time_posix }
89 };

91 struct locdata *
92 __lc_time_load(const char *name)
87 struct lc_time_T *
88 __get_current_time_locale(void)
93 {
94     struct locdata *ldata;
95     struct lc_time *ltime;
96     int ret;
90 return (_time_using_locale ? &_time_locale :
91         (struct lc_time_T *)&_C_time_locale);
92 }

98 if ((ldata = __locdata_alloc(name, sizeof (*ltime))) == NULL) {
99     errno = EINVAL;
100     return (NULL);
101 }
102 ltime = ldata->l_data[0];

104 ret = __part_load_locale(name, (char **)&ldata->l_data[1],
105     "LC_TIME", LCTIME_SIZE, LCTIME_SIZE, (const char **)ltime);

107 if (ret != _LDP_LOADED) {
108     __locdata_free(ldata);
109     errno = EINVAL;
110     return (NULL);
111 }

113 return (ldata);

```

```
94 int
95 __time_load_locale(const char *name)
96 {
97     return (__part_load_locale(name, &_time_using_locale,
98         &time_locale_buf, "LC_TIME", LCTIME_SIZE, LCTIME_SIZE,
99         (const char **)&_time_locale));
114 }
unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/timelocal.h

1

\*\*\*\*\*

1978 Fri Jul 11 11:43:03 2014

new/usr/src/lib/libc/port/locale/timelocal.h

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1997-2002 FreeBSD Project.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #ifndef _TIMELOCAL_H_
30 #define _TIMELOCAL_H_

32 /*
33  * Private header file for the strftime and strptime localization
34  * stuff.
35  */
36 struct lc_time {
37 struct lc_time_T {
38     const char    *mon[12];
39     const char    *month[12];
40     const char    *wday[7];
41     const char    *weekday[7];
42     const char    *X_fmt;
43     const char    *x_fmt;
44     const char    *c_fmt;
45     const char    *am;
46     const char    *pm;
47     const char    *date_fmt;
48     const char    *ampm_fmt;
49 };

50 struct lc_time *_get_current_time_locale(void);
51 struct lc_time_T *_get_current_time_locale(void);
52 int __time_load_locale(const char *);

53 #endif /* !_TIMELOCAL_H_ */
```

new/usr/src/lib/libc/port/locale/tolower.c

1

\*\*\*\*\*

1142 Fri Jul 11 11:43:03 2014

new/usr/src/lib/libc/port/locale/tolower.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2013 Garrett D'Amore <garrett@damore.org>
14  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
15 */
16 #include "lint.h"
17 #include <ctype.h>
18 #include <locale.h>
19 #include "localeimpl.h"
20 #include "lctype.h"

22 #pragma weak _tolower = tolower
23 #pragma weak _toupper = toupper

25 int
26 tolower_l(int c, locale_t loc)
27 {
28     return (((unsigned)c > 255) ? c : loc->ctype->lc_trans_lower[c]);
29 }

31 int
32 toupper_l(int c, locale_t loc)
33 {
34     return (((unsigned)c > 255) ? c : loc->ctype->lc_trans_upper[c]);
35 }

37 #undef tolower
38 int
39 tolower(int c)
40 {
41     return (isascii(c) ? __trans_lower[c] : tolower_l(c, uselocale(NULL)));
42     return (((unsigned)c > 255) ? c : __trans_lower[c]);
43 }

44 #undef toupper
45 int
46 toupper(int c)
47 {
48     return (isascii(c) ? __trans_upper[c] : toupper_l(c, uselocale(NULL)));
49     return (((unsigned)c > 255) ? c : __trans_upper[c]);
50 }

unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/towlower.c

1

```
*****
3709 Fri Jul 11 11:43:04 2014
new/usr/src/lib/libc/port/locale/towlower.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1989, 1993
5  * The Regents of the University of California. All rights reserved.
6  * (c) UNIX System Laboratories, Inc.
7  * All or some portions of this file are derived from material licensed
8  * to the University of California by American Telephone and Telegraph
9  * Co. or Unix System Laboratories, Inc. and are reproduced herein with
10 * the permission of UNIX System Laboratories, Inc.
11 *
12 * This code is derived from software contributed to Berkeley by
13 * Paul Borman at Krystal Technologies.
14 *
15 * Redistribution and use in source and binary forms, with or without
16 * modification, are permitted provided that the following conditions
17 * are met:
18 * 1. Redistributions of source code must retain the above copyright
19 * notice, this list of conditions and the following disclaimer.
20 * 2. Redistributions in binary form must reproduce the above copyright
21 * notice, this list of conditions and the following disclaimer in the
22 * documentation and/or other materials provided with the distribution.
23 * 4. Neither the name of the University nor the names of its contributors
24 * may be used to endorse or promote products derived from this software
25 * without specific prior written permission.
26 *
27 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
28 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
29 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
30 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
31 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
32 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
33 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
34 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
35 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
36 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
37 * SUCH DAMAGE.
38 */
40 #include "lint.h"
41 #include <wctype.h>
42 #include <stdio.h>
43 #include "localeimpl.h"
44 #include "lctype.h"
45 #include "runetype.h"
47 static wint_t
48 change_case_ext(locale_t loc, wint_t c, int lower)
49 {
50     const RuneLocale *rl;
51     const RuneRange *rr;
52     const RuneEntry *base, *re;
53     size_t lim;
54     _RuneRange *rr_;
55     _RuneEntry *base_, *re_;
57     if (c < 0 || c == EOF)
```

new/usr/src/lib/libc/port/locale/towlower.c

2

```
56     return (c);
58     rl = loc->runelocale;
59     rr = lower ? &rl->__maplower_ext : &rl->__mapupper_ext;
60     rr = lower ?
61         &_CurrentRuneLocale->__maplower_ext :
62         &_CurrentRuneLocale->__mapupper_ext;
63     /* Binary search -- see bsearch.c for explanation. */
64     base = rr->__ranges;
65     for (lim = rr->__nranges; lim != 0; lim >>= 1) {
66         re = base + (lim >> 1);
67         if (re->__min <= c && c <= re->__max) {
68             return (re->__map + c - re->__min);
69         } else if (c > re->__max) {
70             base = re + 1;
71             lim--;
72         }
73     }
74     return (c);
75 wint_t
76 tolower_l(wint_t wc, locale_t loc)
77 {
78     return (iswascii(wc) ? __trans_lower[wc] :
79         (wc < 0 || wc >= _CACHED_RUNES) ?
80         change_case_ext(loc, wc, 1) :
81         loc->runelocale->__maplower[wc]);
82 }
84 #undef tolower
85 wint_t
86 tolower(wint_t wc)
87 {
88     return (iswascii(wc) ? __trans_lower[wc] :
89         (wc < 0 || wc >= _CACHED_RUNES) ?
90         change_case_ext(uselocale(NULL), wc, 1) :
91         uselocale(NULL)->runelocale->__maplower[wc]);
92     return ((wc < 0 || wc >= _CACHED_RUNES) ?
93         __change_case_ext(wc, 1) :
94         _CurrentRuneLocale->__maplower[wc]);
95 }
96 wint_t
97 toupper_l(wint_t wc, locale_t loc)
98 {
99     return (iswascii(wc) ? __trans_upper[wc] :
100         (wc < 0 || wc >= _CACHED_RUNES) ?
101         change_case_ext(loc, wc, 0) :
102         loc->runelocale->__mapupper[wc]);
103 }
104 #undef toupper
105 wint_t
106 toupper(wint_t wc)
107 {
108     return (iswascii(wc) ? __trans_upper[wc] :
109         (wc < 0 || wc >= _CACHED_RUNES) ?
110         change_case_ext(uselocale(NULL), wc, 0) :
111         uselocale(NULL)->runelocale->__mapupper[wc]);
112     return ((wc < 0 || wc >= _CACHED_RUNES) ?
113         __change_case_ext(wc, 0) :
114         _CurrentRuneLocale->__mapupper[wc]);
115 }
116 #endif
117 #endif
118 #endif
119 #endif
120 #endif
121 #endif
122 #endif
123 #endif
124 #endif
125 #endif
126 #endif
127 #endif
128 #endif
129 #endif
130 #endif
131 #endif
132 #endif
133 #endif
134 #endif
135 #endif
136 #endif
137 #endif
138 #endif
139 #endif
140 #endif
141 #endif
142 #endif
143 #endif
144 #endif
145 #endif
146 #endif
147 #endif
148 #endif
149 #endif
150 #endif
151 #endif
152 #endif
153 #endif
154 #endif
155 #endif
156 #endif
157 #endif
158 #endif
159 #endif
160 #endif
161 #endif
162 #endif
163 #endif
164 #endif
165 #endif
166 #endif
167 #endif
168 #endif
169 #endif
170 #endif
171 #endif
172 #endif
173 #endif
174 #endif
175 #endif
176 #endif
177 #endif
178 #endif
179 #endif
180 #endif
181 #endif
182 #endif
183 #endif
184 #endif
185 #endif
186 #endif
187 #endif
188 #endif
189 #endif
190 #endif
191 #endif
192 #endif
193 #endif
194 #endif
195 #endif
196 #endif
197 #endif
198 #endif
199 #endif
200 #endif
201 #endif
202 #endif
203 #endif
204 #endif
205 #endif
206 #endif
207 #endif
208 #endif
209 #endif
210 #endif
211 #endif
212 #endif
213 #endif
214 #endif
215 #endif
216 #endif
217 #endif
218 #endif
219 #endif
220 #endif
221 #endif
222 #endif
223 #endif
224 #endif
225 #endif
226 #endif
227 #endif
228 #endif
229 #endif
230 #endif
231 #endif
232 #endif
233 #endif
234 #endif
235 #endif
236 #endif
237 #endif
238 #endif
239 #endif
240 #endif
241 #endif
242 #endif
243 #endif
244 #endif
245 #endif
246 #endif
247 #endif
248 #endif
249 #endif
250 #endif
251 #endif
252 #endif
253 #endif
254 #endif
255 #endif
256 #endif
257 #endif
258 #endif
259 #endif
260 #endif
261 #endif
262 #endif
263 #endif
264 #endif
265 #endif
266 #endif
267 #endif
268 #endif
269 #endif
270 #endif
271 #endif
272 #endif
273 #endif
274 #endif
275 #endif
276 #endif
277 #endif
278 #endif
279 #endif
280 #endif
281 #endif
282 #endif
283 #endif
284 #endif
285 #endif
286 #endif
287 #endif
288 #endif
289 #endif
290 #endif
291 #endif
292 #endif
293 #endif
294 #endif
295 #endif
296 #endif
297 #endif
298 #endif
299 #endif
300 #endif
301 #endif
302 #endif
303 #endif
304 #endif
305 #endif
306 #endif
307 #endif
308 #endif
309 #endif
310 #endif
311 #endif
312 #endif
313 #endif
314 #endif
315 #endif
316 #endif
317 #endif
318 #endif
319 #endif
320 #endif
321 #endif
322 #endif
323 #endif
324 #endif
325 #endif
326 #endif
327 #endif
328 #endif
329 #endif
330 #endif
331 #endif
332 #endif
333 #endif
334 #endif
335 #endif
336 #endif
337 #endif
338 #endif
339 #endif
340 #endif
341 #endif
342 #endif
343 #endif
344 #endif
345 #endif
346 #endif
347 #endif
348 #endif
349 #endif
350 #endif
351 #endif
352 #endif
353 #endif
354 #endif
355 #endif
356 #endif
357 #endif
358 #endif
359 #endif
360 #endif
361 #endif
362 #endif
363 #endif
364 #endif
365 #endif
366 #endif
367 #endif
368 #endif
369 #endif
370 #endif
371 #endif
372 #endif
373 #endif
374 #endif
375 #endif
376 #endif
377 #endif
378 #endif
379 #endif
380 #endif
381 #endif
382 #endif
383 #endif
384 #endif
385 #endif
386 #endif
387 #endif
388 #endif
389 #endif
390 #endif
391 #endif
392 #endif
393 #endif
394 #endif
395 #endif
396 #endif
397 #endif
398 #endif
399 #endif
400 #endif
401 #endif
402 #endif
403 #endif
404 #endif
405 #endif
406 #endif
407 #endif
408 #endif
409 #endif
410 #endif
411 #endif
412 #endif
413 #endif
414 #endif
415 #endif
416 #endif
417 #endif
418 #endif
419 #endif
420 #endif
421 #endif
422 #endif
423 #endif
424 #endif
425 #endif
426 #endif
427 #endif
428 #endif
429 #endif
430 #endif
431 #endif
432 #endif
433 #endif
434 #endif
435 #endif
436 #endif
437 #endif
438 #endif
439 #endif
440 #endif
441 #endif
442 #endif
443 #endif
444 #endif
445 #endif
446 #endif
447 #endif
448 #endif
449 #endif
450 #endif
451 #endif
452 #endif
453 #endif
454 #endif
455 #endif
456 #endif
457 #endif
458 #endif
459 #endif
460 #endif
461 #endif
462 #endif
463 #endif
464 #endif
465 #endif
466 #endif
467 #endif
468 #endif
469 #endif
470 #endif
471 #endif
472 #endif
473 #endif
474 #endif
475 #endif
476 #endif
477 #endif
478 #endif
479 #endif
480 #endif
481 #endif
482 #endif
483 #endif
484 #endif
485 #endif
486 #endif
487 #endif
488 #endif
489 #endif
490 #endif
491 #endif
492 #endif
493 #endif
494 #endif
495 #endif
496 #endif
497 #endif
498 #endif
499 #endif
500 #endif
501 #endif
502 #endif
503 #endif
504 #endif
505 #endif
506 #endif
507 #endif
508 #endif
509 #endif
510 #endif
511 #endif
512 #endif
513 #endif
514 #endif
515 #endif
516 #endif
517 #endif
518 #endif
519 #endif
520 #endif
521 #endif
522 #endif
523 #endif
524 #endif
525 #endif
526 #endif
527 #endif
528 #endif
529 #endif
530 #endif
531 #endif
532 #endif
533 #endif
534 #endif
535 #endif
536 #endif
537 #endif
538 #endif
539 #endif
540 #endif
541 #endif
542 #endif
543 #endif
544 #endif
545 #endif
546 #endif
547 #endif
548 #endif
549 #endif
550 #endif
551 #endif
552 #endif
553 #endif
554 #endif
555 #endif
556 #endif
557 #endif
558 #endif
559 #endif
560 #endif
561 #endif
562 #endif
563 #endif
564 #endif
565 #endif
566 #endif
567 #endif
568 #endif
569 #endif
570 #endif
571 #endif
572 #endif
573 #endif
574 #endif
575 #endif
576 #endif
577 #endif
578 #endif
579 #endif
580 #endif
581 #endif
582 #endif
583 #endif
584 #endif
585 #endif
586 #endif
587 #endif
588 #endif
589 #endif
590 #endif
591 #endif
592 #endif
593 #endif
594 #endif
595 #endif
596 #endif
597 #endif
598 #endif
599 #endif
600 #endif
601 #endif
602 #endif
603 #endif
604 #endif
605 #endif
606 #endif
607 #endif
608 #endif
609 #endif
610 #endif
611 #endif
612 #endif
613 #endif
614 #endif
615 #endif
616 #endif
617 #endif
618 #endif
619 #endif
620 #endif
621 #endif
622 #endif
623 #endif
624 #endif
625 #endif
626 #endif
627 #endif
628 #endif
629 #endif
630 #endif
631 #endif
632 #endif
633 #endif
634 #endif
635 #endif
636 #endif
637 #endif
638 #endif
639 #endif
640 #endif
641 #endif
642 #endif
643 #endif
644 #endif
645 #endif
646 #endif
647 #endif
648 #endif
649 #endif
650 #endif
651 #endif
652 #endif
653 #endif
654 #endif
655 #endif
656 #endif
657 #endif
658 #endif
659 #endif
660 #endif
661 #endif
662 #endif
663 #endif
664 #endif
665 #endif
666 #endif
667 #endif
668 #endif
669 #endif
670 #endif
671 #endif
672 #endif
673 #endif
674 #endif
675 #endif
676 #endif
677 #endif
678 #endif
679 #endif
680 #endif
681 #endif
682 #endif
683 #endif
684 #endif
685 #endif
686 #endif
687 #endif
688 #endif
689 #endif
690 #endif
691 #endif
692 #endif
693 #endif
694 #endif
695 #endif
696 #endif
697 #endif
698 #endif
699 #endif
700 #endif
701 #endif
702 #endif
703 #endif
704 #endif
705 #endif
706 #endif
707 #endif
708 #endif
709 #endif
710 #endif
711 #endif
712 #endif
713 #endif
714 #endif
715 #endif
716 #endif
717 #endif
718 #endif
719 #endif
720 #endif
721 #endif
722 #endif
723 #endif
724 #endif
725 #endif
726 #endif
727 #endif
728 #endif
729 #endif
730 #endif
731 #endif
732 #endif
733 #endif
734 #endif
735 #endif
736 #endif
737 #endif
738 #endif
739 #endif
740 #endif
741 #endif
742 #endif
743 #endif
744 #endif
745 #endif
746 #endif
747 #endif
748 #endif
749 #endif
750 #endif
751 #endif
752 #endif
753 #endif
754 #endif
755 #endif
756 #endif
757 #endif
758 #endif
759 #endif
760 #endif
761 #endif
762 #endif
763 #endif
764 #endif
765 #endif
766 #endif
767 #endif
768 #endif
769 #endif
770 #endif
771 #endif
772 #endif
773 #endif
774 #endif
775 #endif
776 #endif
777 #endif
778 #endif
779 #endif
780 #endif
781 #endif
782 #endif
783 #endif
784 #endif
785 #endif
786 #endif
787 #endif
788 #endif
789 #endif
790 #endif
791 #endif
792 #endif
793 #endif
794 #endif
795 #endif
796 #endif
797 #endif
798 #endif
799 #endif
800 #endif
801 #endif
802 #endif
803 #endif
804 #endif
805 #endif
806 #endif
807 #endif
808 #endif
809 #endif
810 #endif
811 #endif
812 #endif
813 #endif
814 #endif
815 #endif
816 #endif
817 #endif
818 #endif
819 #endif
820 #endif
821 #endif
822 #endif
823 #endif
824 #endif
825 #endif
826 #endif
827 #endif
828 #endif
829 #endif
830 #endif
831 #endif
832 #endif
833 #endif
834 #endif
835 #endif
836 #endif
837 #endif
838 #endif
839 #endif
840 #endif
841 #endif
842 #endif
843 #endif
844 #endif
845 #endif
846 #endif
847 #endif
848 #endif
849 #endif
850 #endif
851 #endif
852 #endif
853 #endif
854 #endif
855 #endif
856 #endif
857 #endif
858 #endif
859 #endif
860 #endif
861 #endif
862 #endif
863 #endif
864 #endif
865 #endif
866 #endif
867 #endif
868 #endif
869 #endif
870 #endif
871 #endif
872 #endif
873 #endif
874 #endif
875 #endif
876 #endif
877 #endif
878 #endif
879 #endif
880 #endif
881 #endif
882 #endif
883 #endif
884 #endif
885 #endif
886 #endif
887 #endif
888 #endif
889 #endif
890 #endif
891 #endif
892 #endif
893 #endif
894 #endif
895 #endif
896 #endif
897 #endif
898 #endif
899 #endif
900 #endif
901 #endif
902 #endif
903 #endif
904 #endif
905 #endif
906 #endif
907 #endif
908 #endif
909 #endif
910 #endif
911 #endif
912 #endif
913 #endif
914 #endif
915 #endif
916 #endif
917 #endif
918 #endif
919 #endif
920 #endif
921 #endif
922 #endif
923 #endif
924 #endif
925 #endif
926 #endif
927 #endif
928 #endif
929 #endif
930 #endif
931 #endif
932 #endif
933 #endif
934 #endif
935 #endif
936 #endif
937 #endif
938 #endif
939 #endif
940 #endif
941 #endif
942 #endif
943 #endif
944 #endif
945 #endif
946 #endif
947 #endif
948 #endif
949 #endif
950 #endif
951 #endif
952 #endif
953 #endif
954 #endif
955 #endif
956 #endif
957 #endif
958 #endif
959 #endif
960 #endif
961 #endif
962 #endif
963 #endif
964 #endif
965 #endif
966 #endif
967 #endif
968 #endif
969 #endif
970 #endif
971 #endif
972 #endif
973 #endif
974 #endif
975 #endif
976 #endif
977 #endif
978 #endif
979 #endif
980 #endif
981 #endif
982 #endif
983 #endif
984 #endif
985 #endif
986 #endif
987 #endif
988 #endif
989 #endif
990 #endif
991 #endif
992 #endif
993 #endif
994 #endif
995 #endif
996 #endif
997 #endif
998 #endif
999 #endif
1000 #endif
1001 #endif
1002 #endif
1003 #endif
1004 #endif
1005 #endif
1006 #endif
1007 #endif
1008 #endif
1009 #endif
1010 #endif
1011 #endif
1012 #endif
1013 #endif
1014 #endif
1015 #endif
1016 #endif
1017 #endif
1018 #endif
1019 #endif
1020 #endif
1021 #endif
1022 #endif
1023 #endif
1024 #endif
1025 #endif
1026 #endif
1027 #endif
1028 #endif
1029 #endif
1030 #endif
1031 #endif
1032 #endif
1033 #endif
1034 #endif
1035 #endif
1036 #endif
1037 #endif
1038 #endif
1039 #endif
1040 #endif
1041 #endif
1042 #endif
1043 #endif
1044 #endif
1045 #endif
1046 #endif
1047 #endif
1048 #endif
1049 #endif
1050 #endif
1051 #endif
1052 #endif
1053 #endif
1054 #endif
1055 #endif
1056 #endif
1057 #endif
1058 #endif
1059 #endif
1060 #endif
1061 #endif
1062 #endif
1063 #endif
1064 #endif
1065 #endif
1066 #endif
1067 #endif
1068 #endif
1069 #endif
1070 #endif
1071 #endif
1072 #endif
1073 #endif
1074 #endif
1075 #endif
1076 #endif
1077 #endif
1078 #endif
1079 #endif
1080 #endif
1081 #endif
1082 #endif
1083 #endif
1084 #endif
1085 #endif
1086 #endif
1087 #endif
1088 #endif
1089 #endif
1090 #endif
1091 #endif
1092 #endif
1093 #endif
1094 #endif
1095 #endif
1096 #endif
1097 #endif
1098 #endif
1099 #endif
1100 #endif
1101 #endif
1102 #endif
1103 #endif
1104 #endif
1105 #endif
1106 #endif
1107 #endif
1108 #endif
1109 #endif
1110 #endif
1111 #endif
1112 #endif
1113 #endif
1114 #endif
1115 #endif
1116 #endif
1117 #endif
1118 #endif
1119 #endif
1120 #endif
1121 #endif
1122 #endif
1123 #endif
1124 #endif
1125 #endif
1126 #endif
1127 #endif
1128 #endif
1129 #endif
1130 #endif
1131 #endif
1132 #endif
1133 #endif
1134 #endif
1135 #endif
1136 #endif
1137 #endif
1138 #endif
1139 #endif
1140 #endif
1141 #endif
1142 #endif
1143 #endif
1144 #endif
1145 #endif
1146 #endif
1147 #endif
1148 #endif
1149 #endif
1150 #endif
1151 #endif
1152 #endif
1153 #endif
1154 #endif
1155 #endif
1156 #endif
1157 #endif
1158 #endif
1159 #endif
1160 #endif
1161 #endif
1162 #endif
1163 #endif
1164 #endif
1165 #endif
1166 #endif
1167 #endif
1168 #endif
1169 #endif
1170 #endif
1171 #endif
1172 #endif
1173 #endif
1174 #endif
1175 #endif
1176 #endif
1177 #endif
1178 #endif
1179 #endif
1180 #endif
1181 #endif
1182 #endif
1183 #endif
1184 #endif
1185 #endif
1186 #endif
1187 #endif
1188 #endif
1189 #endif
1190 #endif
1191 #endif
1192 #endif
1193 #endif
1194 #endif
1195 #endif
1196 #endif
1197 #endif
1198 #endif
1199 #endif
1200 #endif
1201 #endif
1202 #endif
1203 #endif
1204 #endif
1205 #endif
1206 #endif
1207 #endif
1208 #endif
1209 #endif
1210 #endif
1211 #endif
1212 #endif
1213 #endif
1214 #endif
1215 #endif
1216 #endif
1217 #endif
1218 #endif
1219 #endif
1220 #endif
1221 #endif
1222 #endif
1223 #endif
1224 #endif
1225 #endif
1226 #endif
1227 #endif
1228 #endif
1229 #endif
1230 #endif
1231 #endif
1232 #endif
1233 #endif
1234 #endif
1235 #endif
1236 #endif
1237 #endif
1238 #endif
1239 #endif
1240 #endif
1241 #endif
1242 #endif
1243 #endif
1244 #endif
1245 #endif
1246 #endif
1247 #endif
1248 #endif
1249 #endif
1250 #endif
1251 #endif
1252 #endif
1253 #endif
1254 #endif
1255 #endif
1256 #endif
1257 #endif
1258 #endif
1259 #endif
1260 #endif
1261 #endif
1262 #endif
1263 #endif
1264 #endif
1265 #endif
1266 #endif
1267 #endif
1268 #endif
1269 #endif
1270 #endif
1271 #endif
1272 #endif
1273 #endif
1274 #endif
1275 #endif
1276 #endif
1277 #endif
1278 #endif
1279 #endif
1280 #endif
1281 #endif
1282 #endif
1283 #endif
1284 #endif
1285 #endif
1286 #endif
1287 #endif
1288 #endif
1289 #endif
1290 #endif
1291 #endif
1292 #endif
1293 #endif
1294 #endif
1295 #endif
1296 #endif
1297 #endif
1298 #endif
1299 #endif
1300 #endif
1301 #endif
1302 #endif
1303 #endif
1304 #endif
1305 #endif
1306 #endif
1307 #endif
1308 #endif
1309 #endif
1310 #endif
1311 #endif
1312 #endif
1313 #endif
1314 #endif
1315 #endif
1316 #endif
1317 #endif
1318 #endif
1319 #endif
1320 #endif
1321 #endif
1322 #endif
1323 #endif
1324 #endif
1325 #endif
1326 #endif
1327 #endif
1328 #endif
1329 #endif
1330 #endif
1331 #endif
1332 #endif
1333 #endif
1334 #endif
1335 #endif
1336 #endif
1337 #endif
1338 #endif
1339 #endif
1340 #endif
1341 #endif
1342 #endif
1343 #endif
1344 #endif
1345 #endif
1346 #endif
1347 #endif
1348 #endif
1349 #endif
1350 #endif
1351 #endif
1352 #endif
1353 #endif
1354 #endif
1355 #endif
1356 #endif
1357 #endif
1358 #endif
1359 #endif
1360 #endif
1361 #endif
1362 #endif
1363 #endif
1364 #endif
1365 #endif
1366 #endif
1367 #endif
1368 #endif
1369 #endif
1370 #endif
1371 #endif
1372 #endif
1373 #endif
1374 #endif
1375 #endif
1376 #endif
1377 #endif
1378 #endif
1379 #endif
1380 #endif
1381 #endif
1382 #endif
1383 #endif
1384 #endif
1385 #endif
1386 #endif
1387 #endif
1388 #endif
1389 #endif
1390 #endif
1391 #endif
1392 #endif
1393 #endif
1394 #endif
1395 #endif
1396 #endif
1397 #endif
1398 #endif
1399 #endif
1400 #endif
1401 #endif
1402 #endif
1403 #endif
1404 #endif
1405 #endif
1406 #endif
1407 #endif
1408 #endif
1409 #endif
1410 #endif
1411 #endif
1412 #endif
1413 #endif
1414 #endif
1415 #endif
1416 #endif
1417 #endif
1418 #endif
1419 #endif
1420 #endif
1421 #endif
1422 #endif
1423 #endif
1424 #endif
1425 #endif
1426 #endif
1427 #endif
1428 #endif
1429 #endif
1430 #endif
1431 #endif
1432 #endif
1433 #endif
1434 #endif
1435 #endif
1436 #endif
1437 #endif
1438 #endif
1439 #endif
1440 #endif
1441 #endif
1442 #endif
1443 #endif
1444 #endif
1445 #endif
1446 #endif
1447 #endif
1448 #endif
1449 #endif
1450 #endif
1451 #endif
1452 #endif
1453 #endif
1454 #endif
1455 #endif
1456 #endif
1457 #endif
1458 #endif
1459 #endif
1460 #endif
1461 #endif
1462 #endif
1463 #endif
1464 #endif
1465 #endif
1466 #endif
1467 #endif
1468 #endif
1469 #endif
1470 #endif
1471 #endif
1472 #endif
1473 #endif
1474 #endif
1475 #endif
1476 #endif
1477 #endif
1478 #endif
1479 #endif
1480 #endif
1481 #endif
1482 #endif
1483 #endif
1484 #endif
1485 #endif
1486 #endif
1487 #endif
1488 #endif
1489 #endif
1490 #endif
1491 #endif
1492 #endif
1493 #endif
1494 #endif
1495 #endif
1496 #endif
1497 #endif
1498 #endif
1499 #endif
1500 #endif
1501 #endif
1502 #endif
1503 #endif
1504 #endif
1505 #endif
1506 #endif
1507 #endif
1508 #endif
1509 #endif
1510 #endif
1511 #endif
1512 #endif
1513 #endif
1514 #endif
1515 #endif
1516 #endif
1517 #endif
1518 #endif
1519 #endif
1520 #endif
1521 #endif
1522 #endif
1523 #endif
1524 #endif
1525 #endif
1526 #endif
1527 #endif
1528 #endif
1529 #endif
1530 #endif
1531 #endif
1532 #endif
1533 #endif
1534 #endif
1535 #endif
1536 #endif
1537 #endif
1538 #endif
1539 #endif
1540 #endif
1541 #endif
1542 #endif
1543 #endif
1544 #endif
1545 #endif
1546 #endif
1547 #endif
1548 #endif
1549 #endif
1550 #endif
1551 #endif
1552 #endif
1553 #endif
1554 #endif
1555 #endif
1556 #endif
1557 #endif
1558 #endif
1559 #endif
1560 #endif
1561 #endif
1562 #endif
1563 #endif
1564 #endif
1565 #endif
1566 #endif
1567 #endif
1568 #endif
1569 #endif
1570 #endif
1571 #endif
1572 #endif
1573 #endif
1574 #endif
1575 #endif
1576 #endif
1577 #endif
1578 #endif
1579 #endif
1580 #endif
1581 #endif
1582 #endif
1583 #endif
1584 #endif
1585 #endif
1586 #endif
1587 #endif
1588 #endif
1589 #endif
1590 #endif
1591 #endif
1592 #endif
1593 #endif
1594 #endif
1595 #endif
1596 #endif
1597 #endif
1598 #endif
1599 #endif
1600 #endif
1601 #endif
1602 #endif
1603 #endif
1604 #endif
1605 #endif
1606 #endif
1607 #endif
1608 #endif
1609 #endif
1610 #endif
1611 #endif
1612 #endif
1613 #endif
1614 #endif
1615 #endif
1616 #endif
1617 #endif
1618 #endif
1619 #endif
1620 #endif
1621 #endif
1622 #endif
1623 #endif
1624 #endif
1625 #endif
1626 #endif
1627 #endif
1628 #endif
1629 #endif
1630 #endif
1631 #endif
1632 #endif
1633 #endif
1634 #endif
1635 #endif
1636 #endif
1637 #endif
1638 #endif
1639 #endif
1640 #endif
1641 #endif
1642 #endif
1643 #endif
1644 #endif
1645 #endif
1646 #endif
1647 #endif
1648 #endif
1649 #endif
1650 #endif
1651 #endif
1652 #endif
1653 #endif
1654 #endif
1655 #endif
1656 #endif
1657 #endif
1658 #endif
1659 #endif
1660 #endif
1661 #endif
1662 #endif
1663 #endif
1664 #endif
1665 #endif
1666 #endif
1667 #endif
1668 #endif
1669 #endif
1670 #endif
1671 #endif
1672 #endif
1673 #endif
1674 #endif
1675 #endif
1676 #endif
1677 #endif
1678 #endif
1679 #endif
1680 #endif
1681 #endif
1682 #endif
1683 #endif
1684 #endif
1685 #endif
1686 #endif
1687 #endif
1688 #endif
1689 #endif
1690 #endif
1691 #endif
1692 #endif
1693 #endif
1694 #endif
1695 #endif
1696 #endif
1697 #endif
1698 #endif
1699 #endif
1700 #endif
1701 #endif
1702 #endif
1703 #endif
1704 #endif
1705 #endif
1706 #endif
1707 #endif
1708 #endif
1709 #endif
1710 #endif
1711 #endif
1712 #endif
1713 #endif
1714 #endif
1715 #endif
1716 #endif
1717 #endif
1718 #endif
1719 #endif
1720 #endif
1721 #endif
1722 #endif
1723 #endif
1724 #endif
1725 #endif
1726 #endif
1727 #endif
1728 #endif
1729 #endif
1730 #endif
1731 #endif
1732 #endif
1733 #endif
1734 #endif
1735 #endif
1736 #endif
1737 #endif
17
```

```

*****
10018 Fri Jul 11 11:43:04 2014
new/usr/src/lib/libc/port/locale/utf8.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <errno.h>
31 #include <limits.h>
32 #include "runetype.h"
33 #include <stdlib.h>
34 #include <string.h>
35 #include <wchar.h>
36 #include "mblocal.h"

38 static size_t _UTF8_mbrtowc(wchar_t *_RESTRICT_KYWD,
39 const char *_RESTRICT_KYWD,
40 size_t, mbstate_t *_RESTRICT_KYWD);
41 static int _UTF8_mbsinit(const mbstate_t *);
42 static size_t _UTF8_mbsnrtowcs(wchar_t *_RESTRICT_KYWD,
43 const char **_RESTRICT_KYWD, size_t, size_t,
44 mbstate_t *_RESTRICT_KYWD);
45 static size_t _UTF8_wcrtomb(char *_RESTRICT_KYWD, wchar_t,
46 mbstate_t *_RESTRICT_KYWD);
47 static size_t _UTF8_wcsnrtombs(char *_RESTRICT_KYWD,
48 const wchar_t **_RESTRICT_KYWD,
49 size_t, size_t, mbstate_t *_RESTRICT_KYWD);

51 typedef struct {
52     wchar_t ch;
53     int want;
54     wchar_t lbound;
55 } _UTF8State;

57 void

```

```

58 _UTF8_init(struct lc_ctype *lct)
59 int
60 _UTF8_init(_RuneLocale *rl)
61 {
62     lct->lc_mbrtowc = _UTF8_mbrtowc;
63     lct->lc_wcrtomb = _UTF8_wcrtomb;
64     lct->lc_mbsinit = _UTF8_mbsinit;
65     lct->lc_mbsnrtowcs = _UTF8_mbsnrtowcs;
66     lct->lc_wcsnrtombs = _UTF8_wcsnrtombs;
67     lct->lc_is_ascii = 0;
68     lct->lc_max_mblen = 4;
69     __mbrtowc = _UTF8_mbrtowc;
70     __wcrtomb = _UTF8_wcrtomb;
71     __mbsinit = _UTF8_mbsinit;
72     __mbsnrtowcs = _UTF8_mbsnrtowcs;
73     __wcsnrtombs = _UTF8_wcsnrtombs;
74     _CurrentRuneLocale = rl;

76     charset_is_ascii = 0;

78     /*
79      * In theory up to 6 bytes can be used for the encoding,
80      * but only encodings with more than 4 bytes are illegal.
81      */
82     __ctype[520] = 4;
83     /*
84      * Note that the other CSWIDTH members are nonsensical for this
85      * this coding. They only are valid with EUC codings.
86      */

87     return (0);
88 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libc/port/locale/wcrtomb.c

1

\*\*\*\*\*

1962 Fri Jul 11 11:43:04 2014

new/usr/src/lib/libc/port/locale/wcrtomb.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <wchar.h>
31 #include <locale.h>
32 #include <xlocale.h>
33 #include "localeimpl.h"
34 #include "lctype.h"
35 #include "mblocal.h"

37 size_t
38 wcrtomb_l(char * _RESTRICT_KYWD s, wchar_t wc, mbstate_t * _RESTRICT_KYWD ps,
39 locale_t loc)
33 wcrtomb(char * _RESTRICT_KYWD s, wchar_t wc, mbstate_t * _RESTRICT_KYWD ps)
40 {
41     static mbstate_t mbs;

43     if (ps == NULL)
44         ps = &mbs;
45     return (loc->ctype->lc_wcrtomb(s, wc, ps));
46 }

48 size_t
49 wcrtomb(char * _RESTRICT_KYWD s, wchar_t wc, mbstate_t * _RESTRICT_KYWD ps)
50 {
51     return (wcrtomb_l(s, wc, ps, uselocale(NULL)));
39     return (__wcrtomb(s, wc, ps));
52 }

_____unchanged_portion_omitted_
```

new/usr/src/lib/libc/port/locale/wcscasecmp.c

1

```
*****
2099 Fri Jul 11 11:43:04 2014
new/usr/src/lib/libc/port/locale/wcscasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright (c) 1992, 2010, Oracle and/or its affiliates. All rights reserved.
24  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
25  */
26
27 /*
28  * Compare strings ignoring case difference.
29  * returns: s1>s2: >0 s1==s2: 0 s1<s2: <0
30  * All letters are converted to the lowercase and compared.
31  */
32
33 #include "lint.h"
34 #include <stdlib.h>
35 #include <wchar.h>
36 #include <locale.h>
37 #include "libc.h"
38
39 /* Legacy Sun interfaces */
40 #pragma weak wcscasecmp = wcscasecmp
41 #pragma weak wcsncasecmp = wcsncasecmp
42
43 int
44 wcscasecmp_l(const wchar_t *s1, const wchar_t *s2, locale_t loc)
45 {
46     if (s1 == s2)
47         return (0);
48
49     while (tolower_l(*s1, loc) == tolower_l(*s2, loc)) {
50         if (*s1 == 0)
51             return (0);
52         s1++;
53         s2++;
54     }
55     return (tolower_l(*s1, loc) - tolower_l(*s2, loc));
56 }
57
58 int
```

new/usr/src/lib/libc/port/locale/wcscasecmp.c

2

```
59 wcscasecmp(const wchar_t *s1, const wchar_t *s2)
60 {
61     return (wcscasecmp_l(s1, s2,uselocale(NULL)));
62 }
63
64 int
65 wcsncasecmp_l(const wchar_t *s1, const wchar_t *s2, size_t n, locale_t loc)
66 {
67     if (s1 == s2)
68         return (0);
69
70     while ((tolower_l(*s1, loc) == tolower_l(*s2, loc)) && n-- > 0) {
71         if (*s1 == 0)
72             return (0);
73         s1++;
74         s2++;
75     }
76     return (tolower_l(*s1, loc) - tolower_l(*s2, loc));
77 }
78
79 int
80 wcsncasecmp(const wchar_t *s1, const wchar_t *s2, size_t n)
81 {
82     return (wcsncasecmp_l(s1, s2, n,uselocale(NULL)));
83 }
```



```

*****
4894 Fri Jul 11 11:43:04 2014
new/usr/src/lib/libc/port/locale/wcscoll.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002 Tim J. Robbins
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <errno.h>
31 #include <stdlib.h>
32 #include <string.h>
33 #include <wchar.h>
34 #include <assert.h>
35 #include "collate.h"
36 #include "localeimpl.h"

38 int
39 wcscoll_l(const wchar_t *ws1, const wchar_t *ws2, locale_t loc)
40 {
41     int len1, len2, pri1, pri2, ret;
42     wchar_t *tr1 = NULL, *tr2 = NULL;
43     int direc, pass;
44     const struct lc_collate *lcc = loc->collate;

46     if (lcc->lc_is_posix)
47         collate_info_t *info = _collate_info;

48     if (_collate_load_error)
49         /*
50          * Locale has no special collating order or could not be
51          * loaded, do a fast binary comparison.
52          */
53         return (wcscmp(ws1, ws2));

54     ret = 0;

```

```

55 /*
56  * Once upon a time we had code to try to optimize this, but
57  * it turns out that you really can't make many assumptions
58  * safely. You absolutely have to run this pass by pass,
59  * because some passes will be ignored for a given character,
60  * while others will not. Simpler locales will benefit from
61  * having fewer passes, and most comparisons should resolve
62  * during the primary pass anyway.
63  *
64  * Note that we do one final extra pass at the end to pick
65  * up UNDEFINED elements. There is special handling for them.
66  */
67     for (pass = 0; pass <= lcc->lc_directive_count; pass++) {
68         for (pass = 0; pass <= info->directive_count; pass++) {
69             const int32_t *st1 = NULL;
70             const int32_t *st2 = NULL;
71             int32_t *st1 = NULL;
72             int32_t *st2 = NULL;
73             const wchar_t *w1 = ws1;
74             const wchar_t *w2 = ws2;

75             /* special pass for UNDEFINED */
76             if (pass == lcc->lc_directive_count) {
77                 if (pass == info->directive_count) {
78                     direc = DIRECTION_FORWARD | DIRECTION_UNDEFINED;
79                 } else {
80                     direc = lcc->lc_directive[pass];
81                     direc = info->directive[pass];
82                 }
83             }

84             if (direc & DIRECTION_BACKWARD) {
85                 wchar_t *bp, *fp, c;
86                 if ((tr1 = wcsdup(w1)) == NULL)
87                     goto fail;
88                 bp = tr1;
89                 fp = tr1 + wcslen(tr1) - 1;
90                 while (bp < fp) {
91                     c = *bp;
92                     *bp++ = *fp;
93                     *fp-- = c;
94                 }
95                 if ((tr2 = wcsdup(w2)) == NULL)
96                     goto fail;
97                 bp = tr2;
98                 fp = tr2 + wcslen(tr2) - 1;
99                 while (bp < fp) {
100                     c = *bp;
101                     *bp++ = *fp;
102                     *fp-- = c;
103                 }
104                 w1 = tr1;
105                 w2 = tr2;
106             }

107             if (direc & DIRECTION_POSITION) {
108                 while ((*w1 || st1) && (*w2 || st2)) {
109                     pri1 = pri2 = 0;
110                     _collate_lookup(lcc, w1, &len1, &pri1, pass,
111                                     &st1);
112                     _collate_lookup(w1, &len1, &pri1, pass, &st1);
113                     if (pri1 <= 0) {
114                         if (pri1 < 0) {
115                             errno = EINVAL;
116                             goto fail;
117                         }
118                     }
119                 }

```

```

115         pril = COLLATE_MAX_PRIORITY;
116     }
117     _collate_lookup(lcc, w2, &len2, &pri2, pass,
118                    &st2);
119     _collate_lookup(w2, &len2, &pri2, pass, &st2);
120     if (pri2 <= 0) {
121         if (pri2 < 0) {
122             errno = EINVAL;
123             goto fail;
124         }
125         pri2 = COLLATE_MAX_PRIORITY;
126     }
127     if (pril != pri2) {
128         ret = pril - pri2;
129         goto end;
130     }
131     w1 += len1;
132     w2 += len2;
133 } else {
134     while ((*w1 || st1) && (*w2 || st2)) {
135         pril = pri2 = 0;
136         while (*w1) {
137             _collate_lookup(lcc, w1, &len1,
138                            _collate_lookup(w1, &len1,
139                                             &pril, pass, &st1);
140             if (pril > 0)
141                 break;
142             if (pril < 0) {
143                 errno = EINVAL;
144                 goto fail;
145             }
146             w1 += len1;
147         }
148         while (*w2) {
149             _collate_lookup(lcc, w2, &len2,
150                            _collate_lookup(w2, &len2,
151                                             &pri2, pass, &st2);
152             if (pri2 > 0)
153                 break;
154             if (pri2 < 0) {
155                 errno = EINVAL;
156                 goto fail;
157             }
158             w2 += len2;
159         }
160         if (!pril || !pri2)
161             break;
162         if (pril != pri2) {
163             ret = pril - pri2;
164             goto end;
165         }
166         w1 += len1;
167         w2 += len2;
168     }
169     if (!*w1) {
170         if (*w2) {
171             ret = -(int)*w2;
172             goto end;
173         }
174     } else {
175         ret = *w1;
176         goto end;
177     }

```

```

178         ret = 0;
179     }
180 end:
181     if (tr1)
182         free(tr1);
183     if (tr2)
184         free(tr2);
185
186     return (ret);
187
188 fail:
189     ret = wcscmp(ws1, ws2);
190     goto end;
191 }
192
193
194 int
195 wcscoll(const wchar_t *ws1, const wchar_t *ws2)
196 {
197     return (wcscoll_l(ws1, ws2, uselocale(NULL)));
198 }

```

\_\_\_\_\_unchanged\_portion\_omitted\_\_\_\_\_

```

*****
3551 Fri Jul 11 11:43:04 2014
new/usr/src/lib/libc/port/locale/wcsnrtoombs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stdlib.h>
32 #include <string.h>
33 #include <locale.h>
34 #include <wchar.h>
35 #include "mblocal.h"
36 #include "localeimpl.h"
37 #include "lctype.h"

39 size_t
40 wcsnrtoombs_l(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
41              size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps, locale_t loc)
42 wcsnrtoombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
43              size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps)
44 {
45     static mbstate_t mbs;
46
47     if (ps == NULL)
48         ps = &mbs;
49     return (loc->ctype->lc_wcsnrtoombs(dst, src, nwc, len, ps));
50     return (__wcsnrtoombs(dst, src, nwc, len, ps));
51 }

52 size_t
53 wcsnrtoombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
54             __wcsnrtoombs_std(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
55                             size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps)
56 {
57     return (wcsnrtoombs_l(dst, src, nwc, len, ps, uselocale(NULL)));

```

```

55 }

57 size_t
58 __wcsnrtoombs_std(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
59                  size_t nwc, size_t len, mbstate_t *_RESTRICT_KYWD ps,
60                  wchar_t_pfn_t pwcrtomb)
61 {
62     mbstate_t mbsbak;
63     char buf[MB_LEN_MAX];
64     const wchar_t *s;
65     size_t nbytes;
66     size_t nb;

68     s = *src;
69     nbytes = 0;

71     if (dst == NULL) {
72         while (nwc-- > 0) {
73             if ((nb = pwcrtomb(buf, *s, ps)) == (size_t)-1)
74                 if ((nb = __wctomb(buf, *s, ps)) == (size_t)-1)
75                     /* Invalid character - wctomb() sets errno. */
76                     return ((size_t)-1);
77                 else if (*s == L'\0')
78                     return (nbytes + nb - 1);
79                 s++;
80                 nbytes += nb;
81         }
82         return (nbytes);
83     }

84     while (len > 0 && nwc-- > 0) {
85         if (len > (size_t)MB_CUR_MAX) {
86             /* Enough space to translate in-place. */
87             if ((nb = pwcrtomb(dst, *s, ps)) == (size_t)-1) {
88                 if ((nb = __wctomb(dst, *s, ps)) == (size_t)-1) {
89                     *src = s;
90                     return ((size_t)-1);
91                 }
92             } else {
93                 /*
94                  * May not be enough space; use temp. buffer.
95                  *
96                  * We need to save a copy of the conversion state
97                  * here so we can restore it if the multibyte
98                  * character is too long for the buffer.
99                  */
100                mbsbak = *ps;
101                if ((nb = pwcrtomb(buf, *s, ps)) == (size_t)-1) {
102                    if ((nb = __wctomb(buf, *s, ps)) == (size_t)-1) {
103                        *src = s;
104                        return ((size_t)-1);
105                    }
106                }
107                if (nb > (int)len) {
108                    /* MB sequence for character won't fit. */
109                    *ps = mbsbak;
110                    break;
111                }
112                (void) memcpy(dst, buf, nb);
113            }
114            if (*s == L'\0') {
115                *src = NULL;
116                return (nbytes + nb - 1);
117            }
118            s++;
119            dst += nb;
120            len -= nb;

```

new/usr/src/lib/libc/port/locale/wcsnrtombs.c

3

```
118         nbytes += nb;
119     }
120     *src = s;
121     return (nbytes);
122 }
unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/wcsrtombs.c

1

\*\*\*\*\*

2097 Fri Jul 11 11:43:04 2014

new/usr/src/lib/libc/port/locale/wcsrtombs.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stdlib.h>
32 #include <string.h>
33 #include <wchar.h>
34 #include "localeimpl.h"
35 #include "lctype.h"
36 #include "mblocal.h"

38 size_t
39 wcsrtombs_l(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
40             size_t len, mbstate_t *_RESTRICT_KYWD ps, locale_t loc)
36 wcsrtombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
37            size_t len, mbstate_t *_RESTRICT_KYWD ps)
41 {
42     static mbstate_t mbs;

44     if (ps == NULL)
45         ps = &mbs;
46     return (loc->ctype->lc_wcsnrtoombs(dst, src, ULONG_MAX, len, ps));
47 }

49 size_t
50 wcsrtombs(char *_RESTRICT_KYWD dst, const wchar_t **_RESTRICT_KYWD src,
51           size_t len, mbstate_t *_RESTRICT_KYWD ps)
52 {
53     return (wcsrtombs_l(dst, src, len, ps, uselocale(NULL)));
43     return (__wcsnrtoombs(dst, src, ULONG_MAX, len, ps));
54 }

    unchanged_portion_omitted
```

new/usr/src/lib/libc/port/locale/wcstombs.c

1

\*\*\*\*\*

2054 Fri Jul 11 11:43:05 2014

new/usr/src/lib/libc/port/locale/wcstombs.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stdlib.h>
32 #include <wchar.h>
33 #include "mblocal.h"
34 #include "localeimpl.h"
35 #include "lctype.h"

37 size_t
38 wcstombs_l(char *_RESTRICT KYWD s, const wchar_t *_RESTRICT KYWD pwcs,
39 size_t n, locale_t loc)
35 wcstombs(char *_RESTRICT KYWD s, const wchar_t *_RESTRICT KYWD pwcs, size_t n)
40 {
41     static const mbstate_t initial = { 0 };
42     mbstate_t mbs;
43     const wchar_t *pwcs;

45     mbs = initial;
46     pwcs = pwcs;
47     return (loc->ctype->lc_wcsnrtombs(s, &pwcs, ULONG_MAX, n, &mbs));
48 }

50 size_t
51 wcstombs(char *_RESTRICT KYWD s, const wchar_t *_RESTRICT KYWD pwcs, size_t n)
52 {
53     return (wcstombs_l(s, pwcs, n, uselocale(NULL)));
54     return (__wcsnrtombs(s, &pwcs, ULONG_MAX, n, &mbs));
54 }

_unchanged_portion_omitted_
```

new/usr/src/lib/libc/port/locale/wcswidth.c

1

```
*****
2530 Fri Jul 11 11:43:05 2014
new/usr/src/lib/libc/port/locale/wcswidth.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1989, 1993
5  * The Regents of the University of California. All rights reserved.
6  * (c) UNIX System Laboratories, Inc.
7  * All or some portions of this file are derived from material licensed
8  * to the University of California by American Telephone and Telegraph
9  * Co. or Unix System Laboratories, Inc. and are reproduced herein with
10 * the permission of UNIX System Laboratories, Inc.
11 *
12 * This code is derived from software contributed to Berkeley by
13 * Paul Borman at Krystal Technologies.
14 *
15 * Redistribution and use in source and binary forms, with or without
16 * modification, are permitted provided that the following conditions
17 * are met:
18 * 1. Redistributions of source code must retain the above copyright
19 * notice, this list of conditions and the following disclaimer.
20 * 2. Redistributions in binary form must reproduce the above copyright
21 * notice, this list of conditions and the following disclaimer in the
22 * documentation and/or other materials provided with the distribution.
23 * 4. Neither the name of the University nor the names of its contributors
24 * may be used to endorse or promote products derived from this software
25 * without specific prior written permission.
26 *
27 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
28 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
29 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
30 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
31 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
32 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
33 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
34 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
35 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
36 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
37 * SUCH DAMAGE.
38 */

40 #include "lint.h"
41 #include <locale.h>
42 #include <wchar.h>
43 #include <xlocale.h>

45 int
46 wcswidth_l(const wchar_t *pwcs, size_t n, locale_t loc)
47 wcswidth(const wchar_t *pwcs, size_t n)
48 {
49     wchar_t wc;
50     int len, l;

51     len = 0;
52     while (n-- > 0 && (wc = *pwcs++) != L'\0') {
53         if ((l = wcwidth_l(wc, loc)) < 0)
54             if (l = wcwidth(wc)) < 0)
55                 return (-1);
56         len += l;
57     }
58 }
```

new/usr/src/lib/libc/port/locale/wcswidth.c

2

```
57     return (len);
58 }

60 int
61 wcswidth(const wchar_t *pwcs, size_t n)
62 {
63     return (wcswidth_l(pwcs, n, uselocale(NULL)));
64 }
_____unchanged_portion_omitted_
```

new/usr/src/lib/libc/port/locale/wcsxfrm.c

1

```
*****
2507 Fri Jul 11 11:43:05 2014
new/usr/src/lib/libc/port/locale/wcsxfrm.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1995 Alex Tatmanjants <alex@elvisti.kiev.ua>
5  *      at Electronni Visti IA, Kiev, Ukraine.
6  *      All rights reserved.
7  *
8  * Redistribution and use in source and binary forms, with or without
9  * modification, are permitted provided that the following conditions
10 * are met:
11 * 1. Redistributions of source code must retain the above copyright
12 * notice, this list of conditions and the following disclaimer.
13 * 2. Redistributions in binary form must reproduce the above copyright
14 * notice, this list of conditions and the following disclaimer in the
15 * documentation and/or other materials provided with the distribution.
16 *
17 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND
18 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
19 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
20 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE
21 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
22 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
23 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
24 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
25 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
26 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
27 * SUCH DAMAGE.
28 */

30 #include "lint.h"
31 #include <stdlib.h>
32 #include <string.h>
33 #include <wchar.h>
34 #include <assert.h>
35 #include "collate.h"

37 #define WCS_XFRM_OFFSET 1

39 size_t
40 wcsxfrm_l(wchar_t *_RESTRICT_KYWD dest,
41           const wchar_t *_RESTRICT_KYWD src, size_t len, locale_t loc)
39 wcsxfrm(wchar_t *_RESTRICT_KYWD dest,
40         const wchar_t *_RESTRICT_KYWD src, size_t len)
42 {
43     size_t slen;
44     const struct lc_collate *lcc = loc->collate;

46     if (*src == L'\0') {
47         if (len != 0)
48             *dest = L'\0';
49         return (0);
50     }

52     if ((lcc->lc_is_posix) ||
53         ((slen = _collate_wxfrm(lcc, src, dest, len)) == (size_t)-1)) {
50     if ((_collate_load_error) ||
51         ((slen = _collate_wxfrm(src, dest, len)) == (size_t)-1)) {
54         goto error;

```

new/usr/src/lib/libc/port/locale/wcsxfrm.c

2

```
55     }

57     /* Add null termination at the correct location. */
58     if (len > slen) {
59         dest[slen] = 0;
60     } else if (len != 0) {
61         dest[len-1] = 0;
62     }

64     return (slen);

66 error:
67     slen = wcslen(src);
68     if (slen < len)
69         (void) wcsncpy(dest, src);
70     else if (len != 0) {
71         (void) wcsncpy(dest, src, len - 1);
72         dest[len - 1] = L'\0';
73     }
74     return (slen);
75 }

77 size_t
78 wcsxfrm(wchar_t *_RESTRICT_KYWD dest,
79         const wchar_t *_RESTRICT_KYWD src, size_t len)
80 {
81     return (wcsxfrm_l(dest, src, len, uselocale(NULL)));
82 }

unchanged_portion_omitted_

```



new/usr/src/lib/libc/port/locale/wctob.c

1

\*\*\*\*\*

1913 Fri Jul 11 11:43:05 2014

new/usr/src/lib/libc/port/locale/wctob.c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <limits.h>
31 #include <stdio.h>
32 #include <wchar.h>
33 #include <locale.h>
34 #include <xlocale.h>
35 #include "mblocal.h"

37 int
38 wctob_l(wint_t c, locale_t loc)
39 wctob(wint_t c)
40 {
41     static const mbstate_t initial = { 0 };
42     mbstate_t mbs = initial;
43     char buf[MB_LEN_MAX];

44     if (c == WEOF || wcrctomb_l(buf, c, &mbs, loc) != 1)
45     if (c == WEOF || __wcrctomb(buf, c, &mbs) != 1)
46         return (EOF);
47     return ((unsigned char)*buf);

49 int
50 wctob(wint_t c)
51 {
52     return (wctob_l(c,uselocale(NULL)));
53 }

unchanged_portion_omitted_
```

```

*****
1961 Fri Jul 11 11:43:05 2014
new/usr/src/lib/libc/port/locale/wctomb.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002-2004 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <stdlib.h>
31 #include <wchar.h>
32 #include <locale.h>
33 #include <xlocale.h>
34 #include "mblocal.h"

36 int
37 wctomb_l(char *s, wchar_t wchar, locale_t loc)
38 {
39     mbsstate_t mbs = { 0 };
40     static const mbsstate_t initial = { 0 };
41     static mbsstate_t mbs;
42     size_t rval;

43     if (s == NULL) {
44         /* No support for state dependent encodings. */
45         mbs = initial;
46         return (0);
47     }
48     if ((rval = wcrctomb_l(s, wchar, &mbs, loc)) == (size_t)-1)
49         if ((rval = __wcrctomb(s, wchar, &mbs)) == (size_t)-1)
50             return (-1);
51     return ((int)rval);
52 }

51 int
52 wctomb(char *s, wchar_t wchar)
53 {

```

```

54     return (wctomb_l(s, wchar, uselocale(NULL)));
55 }
_____unchanged_portion_omitted_

```

new/usr/src/lib/libc/port/locale/wctrans.c

1

```
*****
2642 Fri Jul 11 11:43:05 2014
new/usr/src/lib/libc/port/locale/wctrans.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */
29 #include "lint.h"
30 #include <note.h>
31 #include <errno.h>
32 #include <string.h>
33 #include <wctype.h>
34 #include <locale.h>
36 enum {
37     _WCT_ERROR      = 0,
38     _WCT_TOLOWER    = 1,
39     _WCT_TOUPPER    = 2
40 };
42 wint_t
43 towctrans_l(wint_t wc, wctrans_t desc, locale_t loc)
44 {
45     switch (desc) {
46     case _WCT_TOLOWER:
47         wc = towlower_l(wc, loc);
48         wc = towlower(wc);
49         break;
50     case _WCT_TOUPPER:
51         wc = towupper_l(wc, loc);
52         wc = towupper(wc);
53         break;
54     case _WCT_ERROR:
55     default:
56         errno = EINVAL;
57     }
58 }
```

new/usr/src/lib/libc/port/locale/wctrans.c

2

```
55     break;
56 }
58     return (wc);
59 }
61 wint_t
62 towctrans(wint_t wc, wctrans_t desc)
63 {
64     return (towctrans_l(wc, desc, uselocale(NULL)));
65 }
67 /*
68  * For *now* we don't support locale sensitive transforms besides toupper
69  * and tolower.
70 */
71 wctrans_t
72 wctrans_l(const char *charclass, locale_t loc)
73 wctrans(const char *charclass)
74 {
75     struct {
76         const char    *name;
77         wctrans_t     trans;
78     } ccls[] = {
79         { "tolower",    _WCT_TOLOWER },
80         { "toupper",    _WCT_TOUPPER },
81         { NULL,         _WCT_ERROR },
82     };
83     int i;
84     _NOTE(ARGUNUSED(loc));
85     i = 0;
86     while (ccls[i].name != NULL && strcmp(ccls[i].name, charclass) != 0)
87         i++;
88     if (ccls[i].trans == _WCT_ERROR)
89         errno = EINVAL;
90     return (ccls[i].trans);
91 }
93 wctrans_t
94 wctrans(const char *charclass)
95 {
96     return (wctrans_l(charclass, uselocale(NULL)));
97 }
98 }
_____unchanged_portion_omitted_____
```

```

*****
2693 Fri Jul 11 11:43:05 2014
new/usr/src/lib/libc/port/locale/wctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 2002 Tim J. Robbins.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms, with or without
8  * modification, are permitted provided that the following conditions
9  * are met:
10 * 1. Redistributions of source code must retain the above copyright
11 * notice, this list of conditions and the following disclaimer.
12 * 2. Redistributions in binary form must reproduce the above copyright
13 * notice, this list of conditions and the following disclaimer in the
14 * documentation and/or other materials provided with the distribution.
15 *
16 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
17 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
18 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
19 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
20 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
22 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
23 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
24 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
25 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
26 * SUCH DAMAGE.
27 */

29 #include "lint.h"
30 #include <ctype.h>
31 #include <string.h>
32 #include <wctype.h>
33 #include <note.h>
34 #include <locale.h>
35 #include "_ctype.h"

37 /*
38  * For now, we don't support locale specific character classes. This is
39  * a capability that needs to be added (locales should be able to define
40  * their own character classes.)
41  */
42 wctype_t
43 wctype_l(const char *property, locale_t loc)
44 wctype(const char *property)
44 {
45     static const struct {
46         const char      *name;
47         wctype_t        mask;
48     } props[] = {
49         { "alnum",      _CTYPE_A|_CTYPE_D },
50         { "alpha",      _CTYPE_A },
51         { "blank",      _CTYPE_B },
52         { "cntrl",      _CTYPE_C },
53         { "digit",      _CTYPE_D },
54         { "graph",      _CTYPE_G },
55         { "lower",      _CTYPE_L },
56         { "print",      _CTYPE_R },
57         { "punct",      _CTYPE_P },

```

```

58     { "space",        _CTYPE_S },
59     { "upper",        _CTYPE_U },
60     { "xdigit",       _CTYPE_X },
61     { "ideogram",     _CTYPE_I },          /* BSD extension */
62     { "special",      _CTYPE_T },          /* BSD extension */
63     { "phonogram",    _CTYPE_Q },          /* BSD extension */
64     { "rune",         -1 },                /* BSD extension */
65     { NULL,           0 },                 /* Default */
66     };
67     int i;
68     _NOTE(ARGUNUSED(loc));

70     i = 0;
71     while (props[i].name != NULL && strcmp(props[i].name, property) != 0)
72         i++;

74     return (props[i].mask);
75 }

77 wctype_t
78 wctype(const char *property)
79 {
80     return (wctype_l(property, uselocale(NULL)));
81 }

unchanged_portion_omitted

```

new/usr/src/lib/libc/port/locale/wcwidth.c

1

```
*****
3005 Fri Jul 11 11:43:06 2014
new/usr/src/lib/libc/port/locale/wcwidth.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright (c) 1989, 1993
4  * The Regents of the University of California. All rights reserved.
5  * (c) UNIX System Laboratories, Inc.
6  * All or some portions of this file are derived from material licensed
7  * to the University of California by American Telephone and Telegraph
8  * Co. or Unix System Laboratories, Inc. and are reproduced herein with
9  * the permission of UNIX System Laboratories, Inc.
10 *
11 * This code is derived from software contributed to Berkeley by
12 * Paul Borman at Krystal Technologies.
13 *
14 * Redistribution and use in source and binary forms, with or without
15 * modification, are permitted provided that the following conditions
16 * are met:
17 * 1. Redistributions of source code must retain the above copyright
18 * notice, this list of conditions and the following disclaimer.
19 * 2. Redistributions in binary form must reproduce the above copyright
20 * notice, this list of conditions and the following disclaimer in the
21 * documentation and/or other materials provided with the distribution.
22 * 4. Neither the name of the University nor the names of its contributors
23 * may be used to endorse or promote products derived from this software
24 * without specific prior written permission.
25 *
26 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
27 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
28 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
29 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
30 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
31 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
32 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
33 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
34 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
35 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
36 * SUCH DAMAGE.
37 */
39 /*
40 * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
41 * Use is subject to license terms.
42 */
44 #include "lint.h"
45 #include <wchar.h>
46 #include "_ctype.h"
47 #include "runetype.h"
48 #include "localeimpl.h"
50 #undef wcwidth
52 int
53 wcwidth_l(wchar_t wc, locale_t loc)
54 {
55     unsigned int x;
56     const _RuneLocale *rl = loc->runelocale;
```

new/usr/src/lib/libc/port/locale/wcwidth.c

2

```
58     if (wc == 0)
59         return (0);
61     x = ((wc < 0 || wc >= _CACHED_RUNES) ? __runetype(rl, wc) :
62         rl->__runetype[wc]) & (_CTYPE_SWM|_CTYPE_R);
58     x = ((wc < 0 || wc >= _CACHED_RUNES) ? __runetype(wc) :
59         _CurrentRuneLocale->__runetype[wc]) & (_CTYPE_SWM|_CTYPE_R);
64     if ((x & _CTYPE_SWM) != 0)
65         return ((x & _CTYPE_SWM) >> _CTYPE_SWS);
66     return ((x & _CTYPE_R) != 0 ? 1 : -1);
67 }
69 int
70 wcwidth(wchar_t wc)
71 {
72     return (wcwidth_l(wc, uselocale(NULL)));
73 }
75 #pragma weak _scrwidth = scrwidth
77 /*
78 * This is a Solaris extension. It never returns a negative width, even for
79 * non-printable characters. It is used internally by the printf
80 * implementation for %ws.
81 */
82 int
83 scrwidth(wchar_t wc)
84 {
85     int v = wcwidth(wc);
86     return (v > 0 ? v : 0);
87 }
_____unchanged_portion_omitted_____
```

```

*****
56290 Fri Jul 11 11:43:06 2014
new/usr/src/lib/libc/port/mapfile-vers
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 2006, 2010, Oracle and/or its affiliates. All rights reserved.
23 #
24 # Copyright 2010 Nexenta Systems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 #
27 # Copyright (c) 2012 by Delphix. All rights reserved.
28 # Copyright (c) 2012, Joyent, Inc. All rights reserved.
29 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
30 # Copyright (c) 2013 Gary Mills
31 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
32 #

33 #
34 # MAPFILE HEADER START
35 #
36 # WARNING: STOP NOW. DO NOT MODIFY THIS FILE.
37 # Object versioning must comply with the rules detailed in
38 #
39 #     usr/src/lib/README.mapfiles
40 #
41 # You should not be making modifications here until you've read the most current
42 # copy of that file. If you need help, contact a gatekeeper for guidance.
43 #
44 # MAPFILE HEADER END
45 #

47 $mapfile_version 2

49 #
50 # All function names added to this or any other libc mapfile
51 # must be placed under the 'protected:' designation.
52 # The 'global:' designation is used *only* for data
53 # items and for the members of the malloc() family.
54 #

56 #
57 # README README README README README README: how to update this file

```

```

58 # 1) each version of Solaris/OpenSolaris gets a version number.
59 # (Actually since Solaris is actually a series of OpenSolaris releases
60 # we'll just use OpenSolaris for this exercise.)
61 # OpenSolaris 2008.11 gets 1.23
62 # OpenSolaris 2009.04 gets 1.24
63 # etc.
64 # 2) each project integration uses a unique version number.
65 # PSARC/2008/123 gets 1.24.1
66 # PSARC/2008/456 gets 1.24.2
67 # etc.
68 #

71 # Mnemonic conditional input identifiers:
72 #
73 # - amd64, i386, sparc32, sparcv9: Correspond to ISA subdirectories used to
74 # hold per-platform code. Note however that we use 'sparc32' instead of
75 # 'sparc'. Since '_sparc' is predefined to apply to, all sparc platforms,
76 # naming the 32-bit version 'sparc' would be too likely to cause errors.
77 #
78 # - lf64: Defined on platforms that offer the 32-bit largefile APIs
79 #
80 $if _ELF32
81 $add lf64
82 $endif
83 $if _sparc && _ELF32
84 $add sparc32
85 $endif
86 $if _sparc && _ELF64
87 $add sparcv9
88 $endif
89 $if _x86 && _ELF32
90 $add i386
91 $endif
92 $if _x86 && _ELF64
93 $add amd64
94 $endif

96 SYMBOL_VERSION ILLUMOS_0.8 { # POSIX 2008 newlocale and friends
97     protected:
98         __global_locale;
99         __mb_cur_max;
100         __mb_cur_max_l;
101         btowc_l;
102         duplocale;
103         fgetwc_l;
104         freelocale;
105         getwc_l;
106         isalnum_l;
107         isalpha_l;
108         isblank_l;
109         iscntrl_l;
110         isdigit_l;
111         isgraph_l;
112         islower_l;
113         isprint_l;
114         ispunct_l;
115         isspace_l;
116         isupper_l;
117         iswideogram;
118         iswideogram_l;
119         iswnumber;
120         iswnumber_l;
121         iswhexnumber;
122         iswhexnumber_l;
123         iswphonogram;

```

```
124 iswphonogram_l;
125 iswspecial;
126 iswspecial_l;
127 iswalnum_l;
128 iswalphal_l;
129 iswblank_l;
130 iswcntrl_l;
131 iswctype_l;
132 iswdigit_l;
133 iswgraph_l;
134 iswlower_l;
135 iswprint_l;
136 iswpunct_l;
137 iswspace_l;
138 iswupper_l;
139 mblen_l;
140 mbrlen_l;
141 mbsinit_l;
142 mbsnrtowcs;
143 mbsnrtowcs_l;
144 mbsrtowcs_l;
145 mbstowcs_l;
146 mbtowc_l;
147 newlocale;
148 nl_langinfo_l;
149 strcasecmp_l;
150 strcasestr_l;
151 strcoll_l;
152 strfmon_l;
153 strftime_l;
154 strncasecmp_l;
155 strtptime_l;
156 strxfrm_l;
157 tolower_l;
158 toupper_l;
159 tolower_l;
160 toupper_l;
161 towctrans_l;
162uselocale;
163 wctomb_l;
164 wscasecmp_l;
165 wscoll_l;
166 wcsncasecmp_l;
167 wcsrtombs_l;
168 wcstombs_l;
169 wcswidth_l;
170 wcsxfrm_l;
171 wctob_l;
172 wctomb_l;
173 wctrans_l;
174 wctype_l;
175 wcwidth_l;
176 } ILLUMOS_0.7;

178 SYMBOL_VERSION ILLUMOS_0.7 { # Illumos additions
179     protected:
180         _glob_ext;
181         _globfree_ext;
182 } ILLUMOS_0.6;
unchanged_portion_omitted
```

```

*****
25827 Fri Jul 11 11:43:06 2014
new/usr/src/lib/libc/sparc/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2012, Joyent, Inc. All rights reserved.
24 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
25 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
26 #
27 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
28 # Use is subject to license terms.
29 #

31 LIBCDIR=      $(SRC)/lib/libc
32 LIB_PIC=      libc_pic.a
33 VERS=         .1
34 CPP=          /usr/lib/cpp
35 TARGET_ARCH=  sparc

37 # objects are grouped by source directory

39 # Symbol capabilities objects.
40 EXTPICS=      \
41      $(LIBCDIR)/capabilities/sun4u/sparc/pics/symcap.o \
42      $(LIBCDIR)/capabilities/sun4u-opl/sparc/pics/symcap.o \
43      $(LIBCDIR)/capabilities/sun4u-us3-hwcap1/sparc/pics/symcap.o \
44      $(LIBCDIR)/capabilities/sun4u-us3-hwcap2/sparc/pics/symcap.o \
45      $(LIBCDIR)/capabilities/sun4v-hwcap1/sparc/pics/symcap.o \
46      $(LIBCDIR)/capabilities/sun4v-hwcap2/sparc/pics/symcap.o

48 # local objects
49 STRETS=       \
50      stret1.o  \
51      stret2.o  \
52      stret4.o

54 CRTOBSJ=     \
55      _ftou.o   \
56      cerror.o  \
57      cerror64.o \
58      hwmuldiv.o

```

```

60 DYNOBJS=     \
61      _rtbootld.o

63 FPOBJS=      \
64      _D_cplx_div.o \
65      _D_cplx_div_ix.o \
66      _D_cplx_div_rx.o \
67      _D_cplx_mul.o \
68      _F_cplx_div.o \
69      _F_cplx_div_ix.o \
70      _F_cplx_div_rx.o \
71      _F_cplx_mul.o \
72      _Q_add.o \
73      _Q_cmp.o \
74      _Q_cmpe.o \
75      _Q_cplx_div.o \
76      _Q_cplx_div_ix.o \
77      _Q_cplx_div_rx.o \
78      _Q_cplx_lr_div.o \
79      _Q_cplx_lr_div_ix.o \
80      _Q_cplx_lr_div_rx.o \
81      _Q_cplx_lr_mul.o \
82      _Q_cplx_mul.o \
83      _Q_div.o \
84      _Q_dtoq.o \
85      _Q_fcc.o \
86      _Q_itoq.o \
87      _Q_lltoq.o \
88      _Q_mul.o \
89      _Q_neg.o \
90      _Q_qtod.o \
91      _Q_qtoi.o \
92      _Q_qtos.o \
93      _Q_qtou.o \
94      _Q_scl.o \
95      _Q_set_except.o \
96      _Q_sqrt.o \
97      _Q_stoq.o \
98      _Q_sub.o \
99      _Q_ulltoq.o \
100     _Q_utoq.o \
101     __quad_mag.o

103 FPASMOBJS=   \
104     _Q_get_rp_rd.o \
105     fpgetmask.o \
106     fpgetrnd.o \
107     fpgetsticky.o \
108     fpsetmask.o \
109     fpsetrnd.o \
110     fpsetsticky.o

112 $(__GNUC)FPASMOBJS += \
113     __quad.o

115 ATOMICOBJS=  \
116     atomic.o

118 XATTROBJS=   \
119     xattr_common.o

121 COMOBJS=     \
122     bcmp.o \
123     bcopy.o \
124     bzero.o

```



```

125      bsearch.o          \
126      memccpy.o         \
127      qsort.o           \
128      strtol.o          \
129      strtoul.o         \
130      strtoll.o         \
131      strtoull.o        \

133 DTRACEOBS=           \
134      dtrace_data.o     \

136 GENOBS=              \
137      _getsp.o          \
138      _xregs_clrptr.o   \
139      abs.o             \
140      alloca.o          \
141      ascii_strcasecmp.o \
142      byteorder.o       \
143      cuexit.o          \
144      ecvt.o            \
145      errlst.o          \
146      getctxt.o         \
147      ladd.o            \
148      lmul.o            \
149      lock.o            \
150      lshif1.o          \
151      lsign.o           \
152      lsub.o            \
153      makectxt.o        \
154      memchr.o          \
155      memcmp.o          \
156      new_list.o        \
157      setjmp.o          \
158      siginfolst.o      \
159      siglongjmp.o      \
160      smt_pause.o       \
161      sparc_data.o      \
162      strchr.o          \
163      strcmp.o          \
164      strlcpy.o         \
165      strncmp.o         \
166      strncpy.o         \
167      strnlen.o         \
168      swapctxt.o        \
169      sync_instruction_memory.o

171 # sysobjs that contain large-file interfaces
172 COMSYSOBS64=          \
173      fstatvfs64.o      \
174      getdents64.o      \
175      getrlimit64.o     \
176      lseek64.o         \
177      mmap64.o          \
178      pread64.o         \
179      pwrite64.o        \
180      setrlimit64.o     \
181      statvfs64.o

183 SYSOBS64=

185 COMSYSOBS=           \
186      __clock_timer.o   \
187      __getloadavg.o    \
188      __rusagesys.o     \
189      __signotify.o     \
190      __sigrt.o         \

```

```

191      __time.o          \
192      _lgrp_home_fast.o \
193      _lgrpsys.o        \
194      _nfssys.o         \
195      _portfs.o         \
196      _pset.o           \
197      _rpcsys.o         \
198      _sigaction.o      \
199      _so_accept.o      \
200      _so_bind.o        \
201      _so_connect.o     \
202      _so_getpeername.o \
203      _so_getsockname.o \
204      _so_getsockopt.o  \
205      _so_listen.o      \
206      _so_recv.o        \
207      _so_recvfrom.o    \
208      _so_recvmsg.o     \
209      _so_send.o        \
210      _so_sendmsg.o     \
211      _so_sendto.o      \
212      _so_setsockopt.o  \
213      _so_shutdown.o    \
214      _so_socket.o      \
215      _so_socketpair.o  \
216      _sockconfig.o     \
217      acct.o            \
218      acl.o             \
219      adjtime.o         \
220      alarm.o           \
221      brk.o             \
222      chdir.o           \
223      chroot.o          \
224      cladm.o           \
225      close.o           \
226      execve.o          \
227      exit.o            \
228      facl.o            \
229      fchdir.o          \
230      fchroot.o         \
231      fdsync.o          \
232      fpathconf.o       \
233      fstatfs.o         \
234      fstatvfs.o        \
235      getcpuid.o        \
236      getdents.o        \
237      getegid.o         \
238      geteuid.o         \
239      getgid.o          \
240      getgroups.o       \
241      gethrtime.o       \
242      getitimer.o       \
243      getmsg.o          \
244      getpid.o          \
245      getpmsg.o         \
246      getppid.o         \
247      getrlimit.o       \
248      getuid.o          \
249      gtty.o            \
250      install_utrap.o   \
251      ioctl.o           \
252      kaio.o            \
253      kill.o            \
254      llseek.o          \
255      lseek.o           \
256      memcntl.o        \

```

```

257 mincore.o \
258 mmap.o \
259 mmapobjsys.o \
260 modctl.o \
261 mount.o \
262 mprotect.o \
263 munmap.o \
264 nice.o \
265 ntp_adjtime.o \
266 ntp_gettime.o \
267 p_online.o \
268 pathconf.o \
269 pause.o \
270 pcsample.o \
271 pipe2.o \
272 pollsys.o \
273 pread.o \
274 prioctlset.o \
275 processor_bind.o \
276 processor_info.o \
277 profil.o \
278 putmsg.o \
279 putpmsg.o \
280 pwrite.o \
281 read.o \
282 readv.o \
283 resolvepath.o \
284 seteguid.o \
285 setgid.o \
286 setgroups.o \
287 setitimer.o \
288 setreid.o \
289 setrlimit.o \
290 setuid.o \
291 sigaltstk.o \
292 sigprocmsk.o \
293 sigsendset.o \
294 sigsuspend.o \
295 statfs.o \
296 statvfs.o \
297 stty.o \
298 sync.o \
299 sysconfig.o \
300 sysfs.o \
301 sysinfo.o \
302 syslwp.o \
303 times.o \
304 ulimit.o \
305 umask.o \
306 umount2.o \
307 utssys.o \
308 uucopy.o \
309 vhangup.o \
310 waitid.o \
311 write.o \
312 writev.o \
313 yield.o \

315 SYSOBJS= \
316 __clock_gettime.o \
317 __getcontext.o \
318 __lwp_mutex_unlock.o \
319 __stack_grow.o \
320 __uadmin.o \
321 door.o \
322 forkx.o \

```

```

323 forkallx.o \
324 gettimeofday.o \
325 ptrace.o \
326 syscall.o \
327 tls_get_addr.o \
328 uadmin.o \
329 umount.o \
330 uname.o \
331 vforkx.o \

333 # objects under $(LIBCDIR)/port which contain transitional large file interfaces
334 PORTGEN64= \
335 __xftw64.o \
336 attropen64.o \
337 ftw64.o \
338 mkstemp64.o \
339 nftw64.o \
340 tell64.o \
341 truncate64.o \

343 # objects from source under $(LIBCDIR)/port
344 PORTFP= \
345 __flt_decim.o \
346 __flt_rounds.o \
347 __tbl_10_b.o \
348 __tbl_10_h.o \
349 __tbl_10_s.o \
350 __tbl_2_b.o \
351 __tbl_2_h.o \
352 __tbl_2_s.o \
353 __tbl_fdq.o \
354 __tbl_tens.o \
355 __x_power.o \
356 __base_sup.o \
357 aconvert.o \
358 decimal_bin.o \
359 double_decim.o \
360 econvert.o \
361 fconvert.o \
362 file_decim.o \
363 finite.o \
364 fp_data.o \
365 func_decim.o \
366 gconvert.o \
367 hex_bin.o \
368 ieee_globals.o \
369 pack_float.o \
370 sigfpe.o \
371 string_decim.o \
372 ashldi3.o \
373 ashrdi3.o \
374 cmpdi2.o \
375 divdi3.o \
376 floatdidf.o \
377 floatdisf.o \
378 floatundidf.o \
379 floatundisf.o \
380 lshrdi3.o \
381 moddi3.o \
382 muldi3.o \
383 qdivrem.o \
384 ucmpdi2.o \
385 udivdi3.o \
386 umoddi3.o \

388 PORTGEN= \

```

```

389     _env_data.o      \|
390     _ftoll.o        \|
391     _ftoull.o       \|
392     _xftw.o         \|
393     a64l.o          \|
394     abort.o         \|
395     addsev.o        \|
396     ascii_strncasecmp.o \|
397     assert.o        \|
398     atof.o          \|
399     atoi.o          \|
400     atol.o          \|
401     atoll.o         \|
402     attrat.o        \|
403     attropen.o      \|
404     atexit.o        \|
405     atfork.o        \|
406     basename.o     \|
407     calloc.o        \|
408     catgets.o       \|
409     catopen.o       \|
410     cfgetispeed.o  \|
411     cfgetospeed.o  \|
412     cfree.o         \|
413     cfsetispeed.o  \|
414     cfsetospeed.o  \|
415     cftime.o        \|
416     clock.o         \|
417     closedir.o     \|
418     closefrom.o    \|
419     confstr.o       \|
420     crypt.o         \|
421     csetlen.o       \|
422     ctime.o         \|
423     ctime_r.o      \|
424     daemon.o       \|
425     deflt.o         \|
426     directio.o     \|
427     dirname.o      \|
428     div.o           \|
429     drand48.o       \|
430     dup.o           \|
431     env_data.o     \|
432     err.o           \|
433     errno.o        \|
434     euclen.o       \|
435     event_port.o   \|
436     execvp.o       \|
437     fattach.o      \|
438     fdetach.o      \|
439     fdopendir.o    \|
440     ffs.o          \|
441     fls.o           \|
442     fmtmsg.o       \|
443     ftime.o        \|
444     ftok.o         \|
445     ftw.o          \|
446     gcvt.o         \|
447     getauxv.o      \|
448     getcwd.o       \|
449     getdate_err.o  \|
450     getdtblsize.o  \|
451     getenv.o       \|
452     getexecname.o  \|
453     getgrnam.o     \|
454     getgrnam_r.o   \|

```

```

455     gethostid.o    \|
456     gethostname.o  \|
457     gethz.o        \|
458     getisax.o      \|
459     getloadavg.o   \|
460     getlogin.o     \|
461     getmntent.o    \|
462     getnetgrent.o  \|
463     get_nprocs.o   \|
464     getopt.o       \|
465     getopt_long.o  \|
466     getpagesize.o  \|
467     getpw.o        \|
468     getpwnam.o     \|
469     getpwnam_r.o   \|
470     getrusage.o    \|
471     getspent.o     \|
472     getspent_r.o   \|
473     getsubopt.o    \|
474     gettxt.o       \|
475     getusershell.o \|
476     getut.o        \|
477     getutx.o       \|
478     getvfsent.o    \|
479     getwd.o        \|
480     getwidth.o     \|
481     getxby_door.o  \|
482     gtxt.o         \|
483     hsearch.o      \|
484     iconv.o        \|
485     imaxabs.o      \|
486     imaxdiv.o      \|
487     index.o        \|
488     initgroups.o   \|
489     insque.o       \|
490     isaexec.o      \|
491     isastream.o    \|
492     isatty.o       \|
493     killpg.o       \|
494     klpdlib.o      \|
495     l64a.o         \|
496     lckpwdf.o      \|
497     lconstants.o   \|
498     ldivide.o      \|
499     lexpl0.o       \|
500     lfind.o        \|
501     lfmt.o         \|
502     lfmt_log.o     \|
503     llabs.o        \|
504     lldiv.o        \|
505     llog10.o       \|
506     lltostr.o      \|
507     localtime.o    \|
508     lsearch.o      \|
509     madvise.o      \|
510     malloc.o       \|
511     memalign.o     \|
512     memmem.o       \|
513     mkdev.o        \|
514     mkttemp.o      \|
515     mkfifo.o       \|
516     mkstemp.o      \|
517     mktemp.o       \|
518     mlock.o        \|
519     mlockall.o     \|
520     mon.o          \|

```

```

521      msync.o           \|
522      munlock.o        \|
523      munlockall.o    \|
524      ndbm.o           \|
525      nftw.o           \|
526      nlspath_checks.o \|
527      nsparse.o        \|
528      nss_common.o     \|
529      nss_dbdefs.o     \|
530      nss_deffinder.o  \|
531      opendir.o        \|
532      opt_data.o       \|
533      perror.o         \|
534      pfmt.o           \|
535      pfmt_data.o     \|
536      pfmt_print.o    \|
537      pipe.o           \|
538      plock.o          \|
539      poll.o           \|
540      posix_fadvise.o  \|
541      posix_fallocate.o \|
542      posix_madvise.o  \|
543      posix_memalign.o \|
544      priocntl.o       \|
545      privlib.o        \|
546      priv_str_xlate.o \|
547      psiginfo.o      \|
548      psignal.o        \|
549      pt.o             \|
550      putpwent.o       \|
551      putspent.o       \|
552      raise.o          \|
553      rand.o           \|
554      random.o         \|
555      rctlops.o        \|
556      readdir.o        \|
557      readdir_r.o      \|
558      realpath.o       \|
559      reboot.o         \|
560      regexpr.o        \|
561      remove.o         \|
562      rewinddir.o      \|
563      rindex.o         \|
564      scandir.o        \|
565      seekdir.o        \|
566      select.o         \|
567      select_large_fdset.o \|
568      setlabel.o       \|
569      setpriority.o    \|
570      settimeofday.o   \|
571      sh_locks.o       \|
572      sigflag.o        \|
573      siglist.o        \|
574      sigsend.o        \|
575      sigsetops.o      \|
576      signal.o         \|
577      stack.o          \|
578      stpcpy.o         \|
579      stpncpy.o        \|
580      str2sig.o        \|
581      strcase_ucharmp.o \|
582      strcat.o         \|
583      strchrnul.o     \|
584      strcspn.o        \|
585      strdup.o         \|
586      strerror.o       \|

```

```

587      strlcat.o        \|
588      strncat.o        \|
589      strndup.o        \|
590      strpbrk.o        \|
591      strrchr.o        \|
592      strsep.o         \|
593      strsignal.o      \|
594      strspn.o         \|
595      strstr.o         \|
596      strtod.o         \|
597      strtointmax.o    \|
598      strtok.o         \|
599      strtok_r.o       \|
600      strtoumax.o     \|
601      swab.o           \|
602      swapctl.o        \|
603      sysconf.o        \|
604      syslog.o         \|
605      tcdrain.o        \|
606      tcflow.o         \|
607      tcflush.o        \|
608      tcgetattr.o     \|
609      tcgetpgrp.o     \|
610      tcgetsid.o       \|
611      tcsendbreak.o   \|
612      tcsetattr.o     \|
613      tcsetpgrp.o     \|
614      tell.o           \|
615      telldir.o        \|
616      tfind.o          \|
617      time_data.o      \|
618      time_gdata.o    \|
619      tls_data.o       \|
620      truncate.o       \|
621      tsdalloc.o       \|
622      tsearch.o        \|
623      ttyname.o        \|
624      ttyslot.o        \|
625      ualarm.o         \|
626      ucred.o          \|
627      valloc.o         \|
628      vlfmt.o          \|
629      vpfmt.o          \|
630      waitpid.o        \|
631      walkstack.o      \|
632      wdata.o          \|
633      xgetwidth.o     \|
634      xpg4.o           \|
635      xpg6.o           \|
637      PORTPRINT_W=    \|
638      doprnt_w.o      \|
640      PORTPRINT=      \|
641      asprintf.o       \|
642      doprnt.o         \|
643      fprintf.o        \|
644      printf.o         \|
645      snprintf.o       \|
646      sprintf.o        \|
647      vfprintf.o       \|
648      vprintf.o        \|
649      vsnprintf.o      \|
650      vsprintf.o       \|
651      vwprintf.o       \|
652      wprintf.o        \|

```

```

654 # c89 variants to support 32-bit size of c89 u/intmax_t (32-bit libc only)
655 PORTPRINT_C89= \
656     vfprintf_c89.o \
657     vprintf_c89.o \
658     vsnprintf_c89.o \
659     vsprintf_c89.o \
660     vwprintf_c89.o

662 PORTSTDIO_C89= \
663     vscanf_c89.o \
664     vwscanf_c89.o

666 # portable stdio objects that contain large file interfaces.
667 # Note: fopen64 is a special case, as we build it small.
668 PORTSTDIO64= \
669     fopen64.o \
670     fpos64.o

672 PORTSTDIO_W= \
673     doscan_w.o

675 PORTSTDIO= \
676     __extensions.o \
677     _endopen.o \
678     _filbuf.o \
679     _findbuf.o \
680     _flsbuf.o \
681     _wrtchk.o \
682     clearerr.o \
683     ctermid.o \
684     ctermid_r.o \
685     cuserid.o \
686     data.o \
687     doscan.o \
688     fdopen.o \
689     feof.o \
690     ferror.o \
691     fgetc.o \
692     fgets.o \
693     fileno.o \
694     flockf.o \
695     flush.o \
696     fopen.o \
697     fpos.o \
698     fputc.o \
699     fputs.o \
700     fread.o \
701     fseek.o \
702     fseeko.o \
703     ftell.o \
704     ftello.o \
705     fwrite.o \
706     getc.o \
707     getchar.o \
708     getline.o \
709     getpass.o \
710     gets.o \
711     getw.o \
712     popen.o \
713     putc.o \
714     putchar.o \
715     puts.o \
716     putw.o \
717     rewind.o \
718     scanf.o \

```

```

719     setbuf.o \
720     setbuffer.o \
721     setvbuf.o \
722     system.o \
723     tempnam.o \
724     tmpfile.o \
725     tmpnam_r.o \
726     ungetc.o \
727     mse.o \
728     vscanf.o \
729     vwscanf.o \
730     wscanf.o

732 PORTI18N= \
733     getwchar.o \
734     putwchar.o \
735     putws.o \
736     strcasecmp.o \
737     strcasestr.o \
738     strncasecmp.o \
739     strtows.o \
740     wcsnlen.o \
741     wcstoimax.o \
742     wcstol.o \
743     wcstoul.o \
744     wcs wcs.o \
745     wscasecmp.o \
746     wscat.o \
747     wscr.o \
748     wscmp.o \
749     wscpy.o \
750     wscspn.o \
751     wsdup.o \
752     wslen.o \
753     wsncasecmp.o \
754     wsnat.o \
755     wsncmp.o \
756     wsncpy.o \
757     wspbkr.o \
758     wprintf.o \
759     wsrchr.o \
760     wsscanf.o \
761     wssp.o \
762     wstod.o \
763     wstok.o \
764     wstol.o \
765     wstoll.o \
766     wxfrm.o \
767     wmemchr.o \
768     wmemcmp.o \
769     wmemcpy.o \
770     wmemmove.o \
771     wmemset.o \
772     wcsstr.o \
773     gettext.o \
774     gettext_real.o \
775     gettext_util.o \
776     gettext_gnu.o \
777     plural_parser.o \
778     wdresolve.o \
779     _ctype.o \
780     isascii.o \
781     isdigit.o \
782     toascii.o

778 PORTI18N_COND= \

```

```

779      wcstol_longlong.o  \
780      wcstoul_longlong.o

782 PORTLOCALE=          \
783      big5.o            \
784      btowc.o           \
785      collate.o         \
786      collcmp.o        \
787      euc.o             \
788      fnmatch.o        \
789      fgetwc.o          \
790      fgetws.o          \
791      fix_grouping.o   \
792      fputwc.o          \
793      fputws.o          \
794      fwide.o           \
795      gb18030.o         \
796      gb2312.o         \
797      gbk.o             \
798      getdate.o         \
799      isdigit.o         \
800      iswctype.o       \
801      ldpair.o          \
802      lmessages.o      \
803      lnumeric.o       \
804      lmonetary.o      \
805      localeimpl.o    \
806      localeconv.o     \
807      mbftowc.o        \
808      mblen.o           \
809      mbrlen.o          \
810      mbrtowc.o        \
811      mbsinit.o        \
812      mbsnrto wcs.o    \
813      mbsrtowcs.o      \
814      mbstowcs.o       \
815      mbtowc.o          \
816      mskanji.o        \
817      nextwctype.o     \
818      nl_langinfo.o    \
819      none.o            \
820      regcomp.o         \
821      regfree.o         \
822      regerror.o       \
823      regexec.o        \
824      rune.o           \
825      runetype.o       \
826      setlocale.o      \
827      setrunelocale.o  \
828      strcasecmp.o    \
829      strcasestr.o    \
830      strcoll.o         \
831      strfmon.o         \
832      strftime.o       \
833      strncasecmp.o   \
834      strtptime.o      \
835      strxfrm.o         \
836      table.o           \
837      timelocal.o      \
838      tolower.o         \
839      towlower.o       \
840      ungetwc.o        \
841      utf8.o           \
842      wctomb.o         \
843      wcscasecmp.o    \
844      wcscoll.o        \

```

```

845      wcsftime.o        \
846      wcsnrtoombs.o    \
847      wcsrtombs.o      \
848      wcstombs.o       \
849      wcswidth.o       \
850      wcsxfrm.o        \
851      wctob.o          \
852      wctomb.o         \
853      wctrans.o        \
854      wctype.o         \
855      wcwidth.o        \
856      wscoll.o

858 AIOOBJS=              \
859      aio.o             \
860      aio_alloc.o      \
861      posix_aio.o

863 RTOBJS=               \
864      clock_timer.o    \
865      mqueue.o         \
866      pos4obj.o        \
867      sched.o          \
868      sem.o            \
869      shm.o            \
870      sigev_thread.o

872 TPOOLBJS=             \
873      thread_pool.o

875 THREADSOBJS=         \
876      alloc.o          \
877      assfail.o        \
878      cancel.o         \
879      door_calls.o     \
880      tmem.o           \
881      pthr_attr.o      \
882      pthr_barrier.o   \
883      pthr_cond.o      \
884      pthr_mutex.o     \
885      pthr_rwlock.o   \
886      pthread.o        \
887      rwlock.o         \
888      scalls.o         \
889      sema.o           \
890      sigaction.o      \
891      spawn.o          \
892      synchron.o       \
893      tdb_agent.o      \
894      thr.o             \
895      thread_interface.o \
896      tls.o            \
897      tsd.o

899 THREADSMACHOBJS=     \
900      machdep.o

902 THREADSASMOBJS=     \
903      asm_subr.o

905 UNICODEOBJS=        \
906      u8_textprep.o   \
907      uconv.o

909 UNWINDMACHOBJS=     \
910      unwind.o

```

```

912 UNWINDASMOBJS=      \
913     unwind_frame.o

915 # objects that implement the transitional large file API
916 PORTSYS64=          \
917     lockf64.o        \
918     stat64.o

920 PORTSYS=            \
921     _autofssys.o     \
922     access.o         \
923     acctctl.o        \
924     bsd_signal.o     \
925     chmod.o          \
926     chown.o          \
927     corectl.o        \
928     exacctsyst.o     \
929     execl.o           \
930     execl.e.o        \
931     execv.o          \
932     fcntl.o          \
933     getpagesizes.o   \
934     getpeerucred.o   \
935     inst_sync.o      \
936     issetugid.o      \
937     label.o          \
938     link.o           \
939     lockf.o          \
940     lwp.o            \
941     lwp_cond.o       \
942     lwp_rwlock.o     \
943     lwp_sigmask.o    \
944     meminfosys.o     \
945     mkdir.o          \
946     mknod.o          \
947     msgsys.o         \
948     nfssys.o         \
949     open.o           \
950     pgrpssys.o       \
951     posix_sigwait.o  \
952     ppriv.o          \
953     psetsys.o        \
954     rctlsys.o        \
955     readlink.o       \
956     rename.o         \
957     sbrk.o           \
958     semsys.o         \
959     set_erro.o       \
960     sharefs.o        \
961     shmsys.o         \
962     sidsys.o         \
963     siginterrupt.o   \
964     signal.o         \
965     sigpending.o     \
966     sigstack.o       \
967     stat.o           \
968     symlink.o        \
969     tasksys.o        \
970     time.o           \
971     time_util.o      \
972     ucontext.o       \
973     unlink.o         \
974     ustat.o          \
975     utimesys.o      \
976     zone.o

```

```

978 PORTREGEX=        \
979     glob.o          \
980     regcmp.o        \
981     regex.o         \
982     wordexp.o

984 VALUES= values-Xa.o

986 MOSTOBSJS=        \
987     $(STRETS)       \
988     $(CRTOBSJS)     \
989     $(DYNOBJS)      \
990     $(FPOBJS)       \
991     $(FPASMOBJS)    \
992     $(ATOMICOBJS)   \
993     $(XATTROBJS)    \
994     $(COMOBSJS)     \
995     $(DTRACEOBSJS)  \
996     $(GENOBSJS)     \
997     $(PRFOBSJS)     \
998     $(PORTFP)       \
999     $(PORTGEN)      \
1000    $(PORTGEN64)     \
1001    $(PORTI18N)      \
1002    $(PORTI18N_COND) \
1003    $(PORTLOCALE)    \
1004    $(PORTPRINT)     \
1005    $(PORTPRINT_C89) \
1006    $(PORTPRINT_W)   \
1007    $(PORTREGEX)     \
1008    $(PORTSTDIO)     \
1009    $(PORTSTDIO64)   \
1010    $(PORTSTDIO_C89) \
1011    $(PORTSTDIO_W)   \
1012    $(PORTSYS)       \
1013    $(PORTSYS64)     \
1014    $(AIOOBSJS)      \
1015    $(RTOBSJS)       \
1016    $(TPOOLOBSJS)    \
1017    $(THREADSOBSJS)  \
1018    $(THREADSMACHOBSJS) \
1019    $(THREADSASMOBSJS) \
1020    $(UNICODEOBSJS)  \
1021    $(UNWINDMACHOBSJS) \
1022    $(UNWINDASMOBSJS) \
1023    $(COMSYSOBSJS)   \
1024    $(SYSOBSJS)      \
1025    $(COMSYSOBSJS64) \
1026    $(SYSOBSJS64)   \
1027    $(VALUES)

1029 TRACEOBSJS=      \
1030     plockstat.o

1032 # NOTE: libc.so.1 must be linked with the minimal crti.o and crtn.o
1033 # modules whose source is provided in the $(SRC)/lib/common directory.
1034 # This must be done because otherwise the Sun C compiler would insert
1035 # its own versions of these modules and those versions contain code
1036 # to call out to C++ initialization functions. Such C++ initialization
1037 # functions can call back into libc before thread initialization is
1038 # complete and this leads to segmentation violations and other problems.
1039 # Since libc contains no C++ code, linking with the minimal crti.o and
1040 # crtn.o modules is safe and avoids the problems described above.
1041 OBJECTS= $(CRTI) $(MOSTOBSJS) $(CRTN)
1042 CRTSRCS= ../../common/sparc

```

```

1044 # include common library definitions
1045 include $(SRC)/lib/Makefile.lib

1047 # we need to override the default SONAME here because we might
1048 # be building a variant object (still libc.so.1, but different filename)
1049 SONAME = libc.so.1

1051 CFLAGS += $(CCVERBOSE)

1053 # This is necessary to avoid problems with calling _ex_unwind().
1054 # We probably don't want any inlining anyway.
1055 CFLAGS += -xinline=

1057 CERRWARN += -_gcc=-Wno-parentheses
1058 CERRWARN += -_gcc=-Wno-switch
1059 CERRWARN += -_gcc=-Wno-uninitialized
1060 CERRWARN += -_gcc=-Wno-unused-value
1061 CERRWARN += -_gcc=-Wno-unused-label
1062 CERRWARN += -_gcc=-Wno-unused-variable
1063 CERRWARN += -_gcc=-Wno-type-limits
1064 CERRWARN += -_gcc=-Wno-char-subscripts
1065 CERRWARN += -_gcc=-Wno-clobbered
1066 CERRWARN += -_gcc=-Wno-unused-function
1067 CERRWARN += -_gcc=-Wno-address

1069 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
1070 # enables ASSERT() checking in the threads portion of the library.
1071 # This is automatically enabled for DEBUG builds, not for non-debug builds.
1072 THREAD_DEBUG =
1073 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

1075 # Make string literals read-only to save memory.
1076 CFLAGS += $(XSTRCONST)

1078 ALTPICS= $(TRACEOBSJS:=pics/%)

1080 $(DYNLIB) := BUILD.SO = $(LD) -o $@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) $(EXTPICS)

1082 MAPFILES = $(LIBCDIR)/port/mapfile-vers

1084 CFLAGS += $(EXTN_CFLAGS)
1085 CPPFLAGS= -D_REENTRANT -Dsparc $(EXTN_CPPFLAGS) $(THREAD_DEBUG) \
1086 -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1087 ASFLAGS= $(EXTN_ASFLAGS) -K pic -P -D__STDC__ -D_ASM $(CPPFLAGS) $(sparc_)

1089 # As a favor to the dtrace syscall provider, libc still calls the
1090 # old syscall traps that have been obsoleted by the *at() interfaces.
1091 # Delete this to compile libc using only the new *at() system call traps
1092 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1094 # Inform the run-time linker about libc specialized initialization
1095 RTLDINFO = -z rtldinfo=tls_rtldinfo
1096 DYNFLAGS += $(RTLDINFO)

1098 # Force libc's internal references to be resolved immediately upon loading
1099 # in order to avoid critical region problems. Since almost all libc symbols
1100 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1101 DYNFLAGS += -znw

1103 DYNFLAGS += -e __rtboot
1104 DYNFLAGS += $(EXTN_DYNFLAGS)

1106 # Inform the kernel about the initial DTrace area (in case
1107 # libc is being used as the interpreter / runtime linker).
1108 DTRACE_DATA = -zdtrace=dtrace_data

```

```

1109 DYNFLAGS += $(DTRACE_DATA)

1111 # DTrace needs an executable data segment.
1112 MAPFILE.NED=

1114 BUILD.s= $(AS) $(ASFLAGS) $< -o $@

1116 # Override this top level flag so the compiler builds in its native
1117 # C99 mode. This has been enabled to support the complex arithmetic
1118 # added to libc.
1119 C99MODE= $(C99_ENABLE)

1121 # libc method of building an archive
1122 # The "$(GREP) -v ' L '" part is necessary only until
1123 # lorder is fixed to ignore thread-local variables.
1124 BUILD.AR= $(RM) $@ ; \
1125 $(AR) q $@ '$(LORDER) $(MOSTOBSJS:=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR)

1127 # extra files for the clean target
1128 CLEANFILES= \
1129 $(LIBCDIR)/port/gen/errlst.c \
1130 $(LIBCDIR)/port/gen/new_list.c \
1131 assym.h \
1132 genassym \
1133 $(LIBCBASE)/crt/_rtld.s \
1134 $(LIBCBASE)/crt/_rtbootld.s \
1135 pics/_rtbootld.o \
1136 pics/crti.o \
1137 pics/crtn.o \
1138 $(ALTPICS)

1140 CLOBBERFILES += $(LIB_PIC)

1142 # list of C source for lint
1143 SRCS= \
1144 $(ATOMICOBJS:=.o=$(SRC)/common/atomic/%.c) \
1145 $(XATTROBJS:=.o=$(SRC)/common/xattr/%.c) \
1146 $(COMOBSJS:=.o=$(SRC)/common/util/%.c) \
1147 $(DTRACEOBSJS:=.o=$(SRC)/common/dtrace/%.c) \
1148 $(PORTFP:=.o=$(LIBCDIR)/port/fp/%.c) \
1149 $(PORTGEN:=.o=$(LIBCDIR)/port/gen/%.c) \
1150 $(PORTI18N:=.o=$(LIBCDIR)/port/i18n/%.c) \
1151 $(PORTLOCALE:=.o=$(LIBCDIR)/port/locale/%.c) \
1152 $(PORTPRINT:=.o=$(LIBCDIR)/port/print/%.c) \
1153 $(PORTREGEX:=.o=$(LIBCDIR)/port/regex/%.c) \
1154 $(PORTSTDIO:=.o=$(LIBCDIR)/port/stdio/%.c) \
1155 $(PORTSYS:=.o=$(LIBCDIR)/port/sys/%.c) \
1156 $(AIOOBJS:=.o=$(LIBCDIR)/port/aio/%.c) \
1157 $(RTOBJS:=.o=$(LIBCDIR)/port/rt/%.c) \
1158 $(TPOOLBJS:=.o=$(LIBCDIR)/port/tpool/%.c) \
1159 $(THREADSOBJS:=.o=$(LIBCDIR)/port/threads/%.c) \
1160 $(THREADSMACHOBJS:=.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1161 $(UNICODEOBJS:=.o=$(SRC)/common/unicode/%.c) \
1162 $(UNWINDMACHOBJS:=.o=$(LIBCDIR)/port/unwind/%.c) \
1163 $(FPOBJS:=.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1164 $(LIBCBASE)/crt/_ftou.c \
1165 $(LIBCBASE)/gen/_xregs_clrprtr.c \
1166 $(LIBCBASE)/gen/byteorder.c \
1167 $(LIBCBASE)/gen/ecvt.c \
1168 $(LIBCBASE)/gen/getctxt.c \
1169 $(LIBCBASE)/gen/lmul.c \
1170 $(LIBCBASE)/gen/makectxt.c \
1171 $(LIBCBASE)/gen/signfolst.c \
1172 $(LIBCBASE)/gen/signlongjmp.c \
1173 $(LIBCBASE)/gen/swapctxt.c \
1174 $(LIBCBASE)/sys/ptrace.c

```



```

1175      $(LIBCBASE)/sys/uadmin.c

1177 # conditional assignments
1178 $(DYNLIB) := CRTI = crti.o
1179 $(DYNLIB) := CRTN = crtn.o

1181 # Files which need the threads .il inline template
1182 TIL=
1183     aio.o
1184     alloc.o
1185     assfail.o
1186     atexit.o
1187     atfork.o
1188     cancel.o
1189     door_calls.o
1190     err.o
1191     errno.o
1192     getctxt.o
1193     lwp.o
1194     ma.o
1195     machdep.o
1196     posix_aio.o
1197     pthr_attr.o
1198     pthr_barrier.o
1199     pthr_cond.o
1200     pthr_mutex.o
1201     pthr_rwlock.o
1202     pthread.o
1203     rand.o
1204     rwlock.o
1205     scalls.o
1206     sched.o
1207     sema.o
1208     sigaction.o
1209     sigev_thread.o
1210     spawn.o
1211     stack.o
1212     swapctxt.o
1213     synch.o
1214     tdb_agent.o
1215     thr.o
1216     thread_interface.o
1217     thread_pool.o
1218     tls.o
1219     tsd.o
1220     unwind.o

1222 $(TIL:%=pics/%) := CFLAGS += $(LIBCBASE)/threads/sparc.il

1224 # special kludge for inlines with 'cas':
1225 pics/rwlock.o pics/synch.o pics/lwp.o pics/door_calls.o := \
1226     sparc_CFLAGS += -_gcc=-Wa,-xarch=v8plus

1228 # Files in port/fp subdirectory that need base.il inline template
1229 IL=
1230     __flt_decim.o
1231     decimal_bin.o

1233 $(IL:%=pics/%) := CFLAGS += $(LIBCBASE)/fp/base.il

1235 # Files in fp subdirectory which need __quad.il inline template
1236 QIL=
1237     _Q_add.o
1238     _Q_cmp.o
1239     _Q_cmpe.o
1240     _Q_div.o

```

```

1241     _Q_dtoq.o
1242     _Q_fcc.o
1243     _Q_mul.o
1244     _Q_qtod.o
1245     _Q_qtoi.o
1246     _Q_qtos.o
1247     _Q_qtou.o
1248     _Q_sqrt.o
1249     _Q_stoq.o
1250     _Q_sub.o

1252 $(QIL:%=pics/%) := CFLAGS += $(LIBCDIR)/$(MACH)/fp/__quad.il
1253 pics/_Q%.o := sparc_COPTFLAG = -xO4 -dalign
1254 pics/__quad%.o := sparc_COPTFLAG = -xO4 -dalign

1256 # large-file-aware components that should be built large

1258 $(COMSYSOBS64:%=pics/%) := \
1259     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1261 $(SYSOBS64:%=pics/%) := \
1262     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1264 $(PORTGEN64:%=pics/%) := \
1265     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1267 $(PORTSTDIO64:%=pics/%) := \
1268     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1270 $(PORTSYS64:%=pics/%) := \
1271     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1273 $(PORTSTDIO_W:%=pics/%) := \
1274     CPPFLAGS += -D_WIDE

1276 $(PORTPRINT_W:%=pics/%) := \
1277     CPPFLAGS += -D_WIDE

1279 # printf/scanf functions to support c89-sized intmax_t variables
1280 $(PORTPRINT_C89:%=pics/%) := \
1281     CPPFLAGS += -D_C89_INTMAX32

1283 $(PORTSTDIO_C89:%=pics/%) := \
1284     CPPFLAGS += -D_C89_INTMAX32

1286 $(PORTI18N_COND:%=pics/%) := \
1287     CPPFLAGS += -D_WCS_LONGLONG

1289 # Files which need extra optimization
1290 pics/getenv.o := sparc_COPTFLAG = -xO4

1292 .KEEP_STATE:

1294 all: $(LIBS) $(LIB_PIC)

1296 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1297 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1298 lint := LINTFLAGS += -mn

1300 lint:
1301     @echo $(LINT.c) ... $(LDLIBS)
1302     @$$(LINT.c) $(SRCS) $(LDLIBS)

1304 $(LINTLIB) := SRCS=$(LIBCDIR)/port/l1ib-lc
1305 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1306 $(LINTLIB) := LINTFLAGS=-nvx

```

```

1308 # object files that depend on inline template
1309 $(TIL:%=pics/%): $(LIBCBASE)/threads/sparc.il
1310 $(IL:%=pics/%): $(LIBCBASE)/fp/base.il
1311 $(QL:%=pics/%): $(LIBCDIR)/$(MACH)/fp/__quad.il

1313 # include common libc targets
1314 include $(LIBCDIR)/Makefile.targ

1316 # We need to strip out all CTF and DOF data from the static library
1317 $(LIB_PIC) := DIR = pics
1318 $(LIB_PIC): pics $$ (PICS)
1319     $(BUILD.AR)
1320     $(MCS) -d -n .SUNW_ctf $@ > /dev/null 2>&l
1321     $(MCS) -d -n .SUNW_dof $@ > /dev/null 2>&l
1322     $(AR) -ts $@ > /dev/null
1323     $(POST_PROCESS_A)

1325 # special cases
1326 $(STRETS:%=pics/%): $(LIBCBASE)/crt/stret.s
1327     $(AS) $(ASFLAGS) -DSTRET$(@F:stret%.o=) $(LIBCBASE)/crt/stret.s -o $@
1328     $(POST_PROCESS_O)

1330 $(LIBCBASE)/crt/_rtbootld.s: $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.
1331     $(CC) $(CPPFLAGS) $(CTF_FLAGS) -O -S -K pic \
1332     $(LIBCBASE)/crt/_rtld.c -o $(LIBCBASE)/crt/_rtld.s
1333     $(CAT) $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.s > $@
1334     $(RM) $(LIBCBASE)/crt/_rtld.s

1336 # partially built from C source
1337 pics/_rtbootld.o: $(LIBCBASE)/crt/_rtbootld.s
1338     $(AS) $(ASFLAGS) $(LIBCBASE)/crt/_rtbootld.s -o $@
1339     $(CTFCONVERT_O)

1341 ASSYMDEP_OBJS= \
1342     _lwp_mutex_unlock.o \
1343     _stack_grow.o \
1344     asm_subr.o \
1345     setjmp.o \
1346     smt_pause.o \
1347     tls_get_addr.o \
1348     unwind_frame.o \
1349     vforkx.o

1351 $(ASSYMDEP_OBJS:%=pics/%) := CPPFLAGS += -I.

1353 $(ASSYMDEP_OBJS:%=pics/%): assym.h

1355 # assym.h build rules

1357 assym.h := CFLAGS += -g

1359 GENASSYM_C = $(LIBCDIR)/$(MACH)/genassym.c

1361 genassym: $(GENASSYM_C)
1362     $(NATIVECC) -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc \
1363     $(CPPFLAGS.native) -o $@ $(GENASSYM_C)

1365 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in

1367 assym.h: $(OFFSETS) genassym
1368     $(OFFSETS_CREATE) <$(OFFSETS) >$@
1369     ./genassym >>$@

1371 # derived C source and related explicit dependencies
1372 $(LIBCDIR)/port/gen/errlst.c + \

```

```

1373 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/
1374     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist

1376 pics/errlst.o: $(LIBCDIR)/port/gen/errlst.c

1378 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c

```

```

*****
24490 Fri Jul 11 11:43:06 2014
new/usr/src/lib/libc/sparcv9/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2012, Joyent, Inc. All rights reserved.
24 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
25 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
26 #
27 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
28 # Use is subject to license terms.
29 #

31 LIBCDIR=      $(SRC)/lib/libc
32 LIB_PIC=      libc_pic.a
33 VERS=        .1
34 CPP=         /usr/lib/cpp
35 TARGET_ARCH=  sparc

37 # objects are grouped by source directory

39 # Symbol capabilities objects.
40 EXTPICS=      \
41      $(LIBCDIR)/capabilities/sun4u/sparcv9/pics/symcap.o \
42      $(LIBCDIR)/capabilities/sun4u-opl/sparcv9/pics/symcap.o \
43      $(LIBCDIR)/capabilities/sun4u-us3-hwcap1/sparcv9/pics/symcap.o \
44      $(LIBCDIR)/capabilities/sun4u-us3-hwcap2/sparcv9/pics/symcap.o \
45      $(LIBCDIR)/capabilities/sun4v-hwcap1/sparcv9/pics/symcap.o \
46      $(LIBCDIR)/capabilities/sun4v-hwcap2/sparcv9/pics/symcap.o

48 # local objects
49 STRETS=

51 CRTOBJS=     \
52      __align_cpy_2.o \
53      __align_cpy_4.o \
54      __align_cpy_8.o \
55      _ftou.o \
56      cerror.o

58 DYNOBJS=

```

```

60 FPOBJS=      \
61      _D_cplx_div.o \
62      _D_cplx_div_ix.o \
63      _D_cplx_div_rx.o \
64      _D_cplx_mul.o \
65      _F_cplx_div.o \
66      _F_cplx_div_ix.o \
67      _F_cplx_div_rx.o \
68      _F_cplx_mul.o \
69      _Q_add.o \
70      _Q_cmp.o \
71      _Q_cmpe.o \
72      _Q_cplx_div.o \
73      _Q_cplx_div_ix.o \
74      _Q_cplx_div_rx.o \
75      _Q_cplx_lr_div.o \
76      _Q_cplx_lr_div_ix.o \
77      _Q_cplx_lr_div_rx.o \
78      _Q_cplx_lr_mul.o \
79      _Q_cplx_mul.o \
80      _Q_div.o \
81      _Q_dtoq.o \
82      _Q_fcc.o \
83      _Q_itoq.o \
84      _Q_mul.o \
85      _Q_neg.o \
86      _Q_qtod.o \
87      _Q_qtoi.o \
88      _Q_qtos.o \
89      _Q_qtou.o \
90      _Q_scl.o \
91      _Q_sqrt.o \
92      _Q_stoq.o \
93      _Q_sub.o \
94      _Q_utoq.o

96 FPOBJS64=   \
97      __Qp_qtox.o \
98      __Qp_qtoux.o \
99      __Qp_xtoq.o \
100     __Qp_uxtoq.o \
101     __dtoul.o \
102     __ftoul.o

104 FPASMOBJS=  \
105     _Q_get_rp_rd.o \
106     __quad_mag64.o \
107     fpgetmask.o \
108     fpgetrnd.o \
109     fpgetsticky.o \
110     fpsetmask.o \
111     fpsetrnd.o \
112     fpsetsticky.o

114 $(__GNUCC)FPASMOBJS += \
115     __quad.o

117 ATOMICOBJS= \
118     atomic.o

120 XATTROBJS=  \
121     xattr_common.o

123 COMOBJS=   \
124     bcmp.o

```

```

125      bcopy.o           \
126      bsearch.o        \
127      bzero.o          \
128      memccpy.o        \
129      qsort.o          \
130      strtol.o         \
131      strtoul.o        \
132      strtoll.o        \
133      strtoull.o       \

```

```

135 GENOBJS= \
136  _getsp.o \
137  _xregs_clrptr.o \
138  abs.o \
139  alloca.o \
140  ascii_strcasecmp.o \
141  byteorder.o \
142  cuexit.o \
143  ecvt.o \
144  getctxt.o \
145  lock.o \
146  makectxt.o \
147  memchr.o \
148  memcmp.o \
149  new_list.o \
150  setjmp.o \
151  siginfolst.o \
152  siglongjmp.o \
153  smt_pause.o \
154  sparc_data.o \
155  strchr.o \
156  strcmp.o \
157  strlcpy.o \
158  strncmp.o \
159  strncpy.o \
160  strnlen.o \
161  swapctxt.o \
162  sync_instruction_memory.o \

```

```

164 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
165 # This macro should ALWAYS be empty; native APIs are already 'large file'.
166 COMSYSOBS64=

```

```

168 SYSOBS64=
170 COMSYSOBS64= \
171  __clock_timer.o \
172  __getloadavg.o \
173  __rusagesys.o \
174  __signotify.o \
175  __sigrt.o \
176  __time.o \
177  __lgrp_home_fast.o \
178  __lgrpsys.o \
179  __nfssys.o \
180  __portfs.o \
181  __pset.o \
182  __rpcsys.o \
183  __sigaction.o \
184  __so_accept.o \
185  __so_bind.o \
186  __so_connect.o \
187  __so_getpeername.o \
188  __so_getsockname.o \
189  __so_getsockopt.o \
190  __so_listen.o \

```

```

191      _so_recv.o \
192      _so_recvfrom.o \
193      _so_recvmsg.o \
194      _so_send.o \
195      _so_sendmsg.o \
196      _so_sendto.o \
197      _so_setsockopt.o \
198      _so_shutdown.o \
199      _so_socket.o \
200      _so_socketpair.o \
201      _sockconfig.o \
202      acct.o \
203      acl.o \
204      adjtime.o \
205      alarm.o \
206      brk.o \
207      chdir.o \
208      chroot.o \
209      cladm.o \
210      close.o \
211      execve.o \
212      exit.o \
213      facl.o \
214      fchdir.o \
215      fchroot.o \
216      fdsync.o \
217      fpathconf.o \
218      fstatfs.o \
219      fstatvfs.o \
220      getcpuid.o \
221      getdents.o \
222      getegid.o \
223      geteuid.o \
224      getgid.o \
225      getgroups.o \
226      gethrtime.o \
227      getitimer.o \
228      getmsg.o \
229      getpid.o \
230      getpmsg.o \
231      getppid.o \
232      getrlimit.o \
233      getuid.o \
234      gtty.o \
235      install_utrap.o \
236      ioctl.o \
237      kaio.o \
238      kill.o \
239      llseek.o \
240      lseek.o \
241      memcntl.o \
242      mincore.o \
243      mmap.o \
244      mmapobjsys.o \
245      modctl.o \
246      mount.o \
247      mprotect.o \
248      munmap.o \
249      nice.o \
250      ntp_adjtime.o \
251      ntp_gettime.o \
252      p_online.o \
253      pathconf.o \
254      pause.o \
255      pcsample.o \
256      pipe2.o \

```

```

257 pollsys.o \
258 pread.o \
259 priocntlset.o \
260 processor_bind.o \
261 processor_info.o \
262 profil.o \
263 putmsg.o \
264 putpmsg.o \
265 pwrite.o \
266 read.o \
267 readv.o \
268 resolvepath.o \
269 seteguid.o \
270 setgid.o \
271 setgroups.o \
272 setitimer.o \
273 setreid.o \
274 setrlimit.o \
275 setuid.o \
276 sigaltstk.o \
277 sigprocmsk.o \
278 sigsendset.o \
279 sigsuspend.o \
280 statfs.o \
281 statvfs.o \
282 stty.o \
283 sync.o \
284 sysconfig.o \
285 sysfs.o \
286 sysinfo.o \
287 syslwp.o \
288 times.o \
289 ulimit.o \
290 umask.o \
291 umount2.o \
292 utssys.o \
293 uucopy.o \
294 vhangup.o \
295 waitid.o \
296 write.o \
297 writev.o \
298 yield.o \

300 SYSOBJS= \
301 __clock_gettime.o \
302 __getcontext.o \
303 __uadmin.o \
304 __lwp_mutex_unlock.o \
305 __stack_grow.o \
306 door.o \
307 forkx.o \
308 forkallx.o \
309 gettimeofday.o \
310 sparc_utrap_install.o \
311 syscall.o \
312 tls_get_addr.o \
313 uadmin.o \
314 umount.o \
315 uname.o \
316 vforkx.o \

```

```

318 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
319 # This macro should ALWAYS be empty; native APIs are already 'large file'.
320 PORTGEN64=

```

```

322 # objects from source under $(LIBCDIR)/port

```

```

323 PORTFP= \
324 __flt_decim.o \
325 __flt_rounds.o \
326 __tbl_10_b.o \
327 __tbl_10_h.o \
328 __tbl_10_s.o \
329 __tbl_2_b.o \
330 __tbl_2_h.o \
331 __tbl_2_s.o \
332 __tbl_fdq.o \
333 __tbl_tens.o \
334 __x_power.o \
335 __base_sup.o \
336 aconvert.o \
337 decimal_bin.o \
338 double_decim.o \
339 econvert.o \
340 fconvert.o \
341 file_decim.o \
342 finite.o \
343 fp_data.o \
344 func_decim.o \
345 gconvert.o \
346 hex_bin.o \
347 ieee_globals.o \
348 pack_float.o \
349 sigfpe.o \
350 string_decim.o \

352 PORTGEN= \
353 __env_data.o \
354 __xftw.o \
355 a64l.o \
356 abort.o \
357 addsev.o \
358 ascii_strncasecmp.o \
359 assert.o \
360 attrat.o \
361 atof.o \
362 atoi.o \
363 atol.o \
364 atoll.o \
365 attropen.o \
366 atexit.o \
367 atfork.o \
368 basename.o \
369 calloc.o \
370 catgets.o \
371 catopen.o \
372 cfgetispeed.o \
373 cfgetospeed.o \
374 cfree.o \
375 cfsetispeed.o \
376 cfsetospeed.o \
377 cftime.o \
378 clock.o \
379 closedir.o \
380 closefrom.o \
381 confstr.o \
382 crypt.o \
383 csetlen.o \
384 ctime.o \
385 ctime_r.o \
386 daemon.o \
387 deflt.o \
388 directio.o \

```

```

389     dirname.o           \|
390     div.o               \|
391     drand48.o          \|
392     dup.o              \|
393     env_data.o         \|
394     err.o              \|
395     errno.o           \|
396     euclen.o          \|
397     event_port.o     \|
398     execvp.o          \|
399     fattach.o         \|
400     fdetach.o         \|
401     fdopendir.o      \|
402     ffs.o             \|
403     fls.o             \|
404     fmtmsg.o          \|
405     ftime.o           \|
406     ftok.o            \|
407     ftw.o             \|
408     gcvt.o            \|
409     getauxv.o         \|
410     getcwd.o          \|
411     getdate_err.o    \|
412     getdtblsize.o    \|
413     getenv.o         \|
414     getexecname.o    \|
415     getgrnam.o       \|
416     getgrnam_r.o     \|
417     gethostid.o      \|
418     gethostname.o    \|
419     gethz.o           \|
420     getisax.o         \|
421     getloadavg.o     \|
422     getlogin.o        \|
423     getmntent.o      \|
424     getnetgrent.o    \|
425     get_nprocs.o     \|
426     getopt.o         \|
427     getopt_long.o    \|
428     getpagesize.o    \|
429     getpw.o           \|
430     getpwnam.o       \|
431     getpwnam_r.o     \|
432     getrusage.o      \|
433     getspent.o       \|
434     getspent_r.o     \|
435     getsubopt.o      \|
436     gettxt.o         \|
437     getusershell.o  \|
438     getut.o           \|
439     getutx.o         \|
440     getvfsent.o      \|
441     getwd.o           \|
442     getwidth.o       \|
443     getxby_door.o    \|
444     gtxt.o           \|
445     hsearch.o        \|
446     iconv.o          \|
447     imaxabs.o        \|
448     imaxdiv.o        \|
449     index.o          \|
450     initgroups.o     \|
451     insque.o         \|
452     isaexec.o        \|
453     isastream.o      \|
454     isatty.o         \|

```

```

455     killpg.o          \|
456     klpdlib.o         \|
457     l64a.o           \|
458     lckpwdf.o        \|
459     lconstants.o     \|
460     ldivide.o        \|
461     lexpl0.o         \|
462     lfind.o          \|
463     lfmt.o           \|
464     lfmt_log.o       \|
465     lldiv.o          \|
466     llog10.o         \|
467     lltostr.o        \|
468     lmath.o          \|
469     localtime.o     \|
470     lsearch.o        \|
471     madvise.o        \|
472     malloc.o         \|
473     memalign.o       \|
474     memmem.o         \|
475     mkdev.o          \|
476     mkdtemp.o        \|
477     mkfifo.o         \|
478     mkstemp.o        \|
479     mktemp.o         \|
480     mlock.o          \|
481     mlockall.o      \|
482     mon.o            \|
483     msync.o          \|
484     munlock.o        \|
485     munlockall.o    \|
486     ndbm.o           \|
487     nftw.o           \|
488     nlspath_checks.o \|
489     nsparse.o        \|
490     nss_common.o     \|
491     nss_dbdefs.o     \|
492     nss_deffinder.o  \|
493     opendir.o        \|
494     opt_data.o       \|
495     perror.o         \|
496     pfmt.o           \|
497     pfmt_data.o     \|
498     pfmt_print.o    \|
499     pipe.o           \|
500     plock.o          \|
501     poll.o           \|
502     posix_fadvise.o  \|
503     posix_fallocate.o \|
504     posix_madvise.o  \|
505     posix_memalign.o \|
506     priocntl.o       \|
507     privlib.o        \|
508     priv_str_xlate.o \|
509     psiginfo.o       \|
510     psignal.o        \|
511     pt.o             \|
512     putpwent.o       \|
513     putspent.o       \|
514     raise.o          \|
515     rand.o           \|
516     random.o         \|
517     rctlops.o        \|
518     readdir.o        \|
519     readdir_r.o      \|
520     realpath.o       \|

```

```

521     reboot.o           \|
522     regexpr.o          \|
523     remove.o           \|
524     rewinddir.o       \|
525     rindex.o           \|
526     scandir.o          \|
527     seekdir.o          \|
528     select.o           \|
529     setlabel.o         \|
530     setpriority.o      \|
531     settimeofday.o     \|
532     sh_locks.o         \|
533     sigflag.o          \|
534     siglist.o          \|
535     sigsend.o          \|
536     sigsetops.o        \|
537     ssignal.o          \|
538     stack.o            \|
539     stpcpy.o           \|
540     stpncpy.o          \|
541     str2sig.o          \|
542     strcase_ormap.o    \|
543     strcat.o           \|
544     strchrnul.o        \|
545     strcspn.o          \|
546     strdup.o           \|
547     strerror.o         \|
548     strlcat.o          \|
549     strncat.o          \|
550     strndup.o          \|
551     strpbrk.o          \|
552     strrchr.o          \|
553     strsep.o           \|
554     strsignal.o        \|
555     strspn.o           \|
556     strstr.o           \|
557     strtod.o           \|
558     strtoumax.o        \|
559     strtok.o           \|
560     strtok_r.o         \|
561     strtoumax.o        \|
562     swab.o             \|
563     swapctl.o          \|
564     sysconf.o          \|
565     syslog.o           \|
566     tcdrain.o          \|
567     tcflow.o           \|
568     tcflush.o          \|
569     tcgetattr.o        \|
570     tcgetpgrp.o        \|
571     tcgetsid.o         \|
572     tcseendbreak.o     \|
573     tcsetattr.o        \|
574     tcsetpgrp.o        \|
575     tell.o             \|
576     telldir.o          \|
577     tfind.o            \|
578     time_data.o        \|
579     time_gdata.o       \|
580     tls_data.o         \|
581     truncate.o         \|
582     tsdalloc.o         \|
583     tsearch.o          \|
584     ttyname.o          \|
585     ttyslot.o          \|
586     ualarm.o           \|

```

```

587     ucred.o           \|
588     valloc.o           \|
589     vlfmt.o            \|
590     vpfmt.o            \|
591     waitpid.o          \|
592     walkstack.o        \|
593     wdata.o            \|
594     xgetwidth.o        \|
595     xpg4.o             \|
596     xpg6.o             \|
598     PORTPRINT_W=      \|
599     doprnt_w.o        \|
601     PORTPRINT=        \|
602     asprintf.o         \|
603     doprnt.o           \|
604     fprintf.o          \|
605     printf.o           \|
606     snprintf.o         \|
607     sprintf.o          \|
608     vfprintf.o         \|
609     vprintf.o          \|
610     vsnprintf.o        \|
611     vsprintf.o         \|
612     vwprintf.o         \|
613     wprintf.o          \|
615     # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
616     # This macro should ALWAYS be empty; native APIs are already 'large file'.
617     PORTSTDIO64=
619     PORTSTDIO_W=       \|
620     doscan_w.o         \|
622     PORTSTDIO=        \|
623     __extensions.o     \|
624     __endopen.o        \|
625     __filbuf.o         \|
626     __findbuf.o        \|
627     __flsbuf.o         \|
628     __wrtchk.o         \|
629     clearerr.o         \|
630     ctermid.o          \|
631     ctermid_r.o        \|
632     cuserid.o          \|
633     data.o             \|
634     doscan.o           \|
635     fdopen.o           \|
636     feof.o             \|
637     ferrord.o          \|
638     fgetc.o            \|
639     fgets.o            \|
640     fileno.o           \|
641     flockf.o           \|
642     flush.o            \|
643     fopen.o            \|
644     fpos.o             \|
645     fputc.o            \|
646     fputs.o            \|
647     fread.o            \|
648     fseek.o            \|
649     fseeko.o           \|
650     ftell.o            \|
651     ftello.o           \|
652     fwrite.o           \|

```

```

653      getc.o           \|
654      getchar.o        \|
655      getline.o        \|
656      getpass.o        \|
657      gets.o           \|
658      getw.o           \|
659      popen.o          \|
660      putc.o           \|
661      putchar.o        \|
662      puts.o           \|
663      putw.o           \|
664      rewind.o         \|
665      scanf.o          \|
666      setbuf.o         \|
667      setbuffer.o     \|
668      setvbuf.o       \|
669      system.o        \|
670      tempnam.o        \|
671      tmpfile.o        \|
672      tmpnam_r.o       \|
673      ungetc.o         \|
674      mse.o            \|
675      vscanf.o         \|
676      vwscanf.o       \|
677      wscanf.o        \|

679 PORTI18N=          \|
680      getwchar.o       \|
681      putwchar.o      \|
682      putws.o          \|
683      strcasecmp.o     \|
684      strcasestr.o    \|
685      strtows.o        \|
686      wcsnlen.o        \|
687      wcstoimax.o     \|
688      wcstol.o         \|
689      wcstoul.o        \|
690      wcs wcs.o        \|
691      wscasecmp.o     \|
692      wscat.o          \|
693      wchr.o           \|
694      wscmp.o          \|
695      wscpy.o          \|
696      wscspn.o         \|
697      wsdup.o          \|
698      wslen.o          \|
699      wscasecmp.o     \|
700      wscat.o          \|
701      wscmp.o          \|
702      wscpy.o          \|
703      wscspn.o         \|
704      wsdup.o          \|
705      wslen.o          \|
706      wscasecmp.o     \|
707      wscat.o          \|
708      wscmp.o          \|
709      wscpy.o          \|
710      wscspn.o         \|
711      wsdup.o          \|
712      wslen.o          \|
713      wscasecmp.o     \|

```

```

714      wcsstr.o         \|
715      gettext.o        \|
716      gettext_real.o  \|
717      gettext_util.o  \|
718      gettext_gnu.o   \|
719      plural_parser.o  \|
720      wdresolve.o      \|
721      _ctype.o         \|
722      isascii.o        \|
723      isdigit.o        \|
724      toascii.o        \|

725 PORTI18N_COND=     \|
726      wcstol_longlong.o \|
727      wcstoul_longlong.o \|

729 PORTLOCALE=        \|
730      big5.o           \|
731      btowc.o          \|
732      collate.o        \|
733      collcmp.o        \|
734      euc.o            \|
735      fnmatch.o        \|
736      fgetwc.o         \|
737      fgetws.o         \|
738      fix_grouping.o   \|
739      fputwc.o         \|
740      fputws.o         \|
741      fwide.o          \|
742      gb18030.o        \|
743      gb2312.o         \|
744      gbk.o            \|
745      getdate.o        \|
746      isdigit.o        \|
747      iswctype.o       \|
748      ldpert.o         \|
749      lmessages.o     \|
750      lnumeric.o       \|
751      lmonetary.o      \|
752      localeconv.o     \|
753      localeimpl.o    \|
754      mbftowc.o        \|
755      mblen.o          \|
756      mbrlen.o         \|
757      mbrtowc.o        \|
758      mbsinit.o        \|
759      mbsnrtowcs.o     \|
760      mbsrtowcs.o      \|
761      mbstowcs.o       \|
762      mbtowc.o         \|
763      mskanji.o        \|
764      nextwctype.o     \|
765      nl_langinfo.o    \|
766      none.o           \|
767      regcomp.o        \|
768      regfree.o        \|
769      regerror.o       \|
770      regexec.o        \|
771      rune.o           \|
772      runetype.o       \|
773      setlocale.o      \|
774      setrunelocale.o  \|
775      strcasecmp.o     \|
776      strcasestr.o    \|
777      strcoll.o        \|
778      strfmon.o        \|

```



```

779      strftime.o      \
780 strncasecmp.o      \
781      strptime.o      \
782      strxfrm.o       \
783      table.o         \
784      timelocal.o     \
785      tolower.o       \
786      towlower.o      \
787      ungetwc.o       \
788      utf8.o          \
789      wctomb.o        \
790 wscasecmp.o        \
791      wcscoll.o       \
792      wcsftime.o      \
793      wcsnrtombs.o    \
794      wcsrtombs.o     \
795      wcstombs.o      \
796      wcswidth.o      \
797      wcsxfrm.o       \
798      wctob.o         \
799      wctomb.o        \
800      wctrans.o       \
801      wctype.o        \
802      wcwidth.o       \
803      wscoll.o       \
\
805 AIOBJS=             \
806      aio.o           \
807      aio_alloc.o    \
808      posix_aio.o    \
\
810 RTOBJS=             \
811      clock_timer.o  \
812      mqueue.o       \
813      pos4obj.o      \
814      sched.o        \
815      sem.o          \
816      shm.o          \
817      sigev_thread.o \
\
819 TPOOLBJS=          \
820      thread_pool.o \
\
822 THREADSOBJS=       \
823      alloc.o         \
824      assfail.o      \
825      cancel.o       \
826      door_calls.o   \
827      tmem.o         \
828      pthr_attr.o    \
829      pthr_barrier.o \
830      pthr_cond.o    \
831      pthr_mutex.o   \
832      pthr_rwlock.o  \
833      pthread.o       \
834      rwlock.o       \
835      scalls.o       \
836      sema.o         \
837      sigaction.o    \
838      spawn.o        \
839      synch.o        \
840      tdb_agent.o    \
841      thr.o          \
842      thread_interface.o \
843      tls.o          \
844      tsd.o

```

```

846 THREADSMACHOBJS=   \
847      machdep.o      \
\
849 THREADSASMOBJS=    \
850      asm_subr.o     \
\
852 UNICODEOBJS=       \
853      u8_textprep.o  \
854      uconv.o        \
\
856 UNWINDMACHOBJS=    \
857      unwind.o       \
\
859 UNWINDASMOBJS=     \
860      unwind_frame.o \
\
862 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
863 # This macro should ALWAYS be empty; native APIs are already 'large file'.
864 PORTSYS64=
\
866 PORTSYS=            \
867      _autofssys.o   \
868      access.o       \
869      acctctl.o      \
870      bsd_signal.o   \
871      chmod.o        \
872      chown.o        \
873      corectl.o      \
874      exacctsyst.o   \
875      execl.o        \
876      execl.o        \
877      execv.o        \
878      fcntl.o        \
879      getpagesizes.o \
880      getpeerucred.o \
881      inst_sync.o    \
882      issetugid.o    \
883      label.o        \
884      link.o         \
885      lockf.o        \
886      lwp.o          \
887      lwp_cond.o     \
888      lwp_rwlock.o   \
889      lwp_sigmask.o  \
890      meminfosys.o   \
891      mkdir.o        \
892      mknod.o        \
893      msgsys.o       \
894      nfssys.o       \
895      open.o         \
896      pgrpsys.o     \
897      posix_sigwait.o \
898      ppriv.o        \
899      psetsys.o      \
900      rctlsys.o     \
901      readlink.o    \
902      rename.o       \
903      sbrk.o         \
904      semsys.o       \
905      set_errno.o    \
906      sharefs.o      \
907      shmsys.o       \
908      sidsys.o       \
909      siginterrupt.o \
910      signal.o

```

```

911 sigpending.o \
912 sigstack.o \
913 stat.o \
914 symlink.o \
915 tasksys.o \
916 time.o \
917 time_util.o \
918 ucontext.o \
919 unlink.o \
920 ustat.o \
921 utimesys.o \
922 zone.o \

924 PORTREGEX= \
925 glob.o \
926 regcmp.o \
927 regex.o \
928 wordexp.o \

930 VALUES= values-Xa.o \

932 MOSTOBSJS= \
933 $(STRETS) \
934 $(CRTOBSJS) \
935 $(DYNOBJS) \
936 $(FPOBJS) \
937 $(FPOBJS64) \
938 $(FPASMOBJS) \
939 $(ATOMICOBJS) \
940 $(XATTROBJS) \
941 $(COMOBJS) \
942 $(GENOBJS) \
943 $(PRFOBJS) \
944 $(PORTFP) \
945 $(PORTGEN) \
946 $(PORTGEN64) \
947 $(PORTI18N) \
948 $(PORTI18N_COND) \
949 $(PORTLOCALE) \
950 $(PORTPRINT) \
951 $(PORTPRINT_W) \
952 $(PORTREGEX) \
953 $(PORTSTDIO) \
954 $(PORTSTDIO64) \
955 $(PORTSTDIO_W) \
956 $(PORTSYS) \
957 $(PORTSYS64) \
958 $(AIOOBJS) \
959 $(RTOBJS) \
960 $(TPOOLBJS) \
961 $(THREADSOBJS) \
962 $(THREADSMACHOBJS) \
963 $(THREADSASMOBJS) \
964 $(UNICODEOBJS) \
965 $(UNWINDMACHOBJS) \
966 $(UNWINDASMOBJS) \
967 $(COMSYSOBJS) \
968 $(SYSOBJS) \
969 $(COMSYSOBJS64) \
970 $(SYSOBJS64) \
971 $(VALUES) \

973 TRACEOBJS= \
974 plockstat.o \

976 # NOTE: libc.so.1 must be linked with the minimal crti.o and crtn.o

```

```

977 # modules whose source is provided in the $(SRC)/lib/common directory.
978 # This must be done because otherwise the Sun C compiler would insert
979 # its own versions of these modules and those versions contain code
980 # to call out to C++ initialization functions. Such C++ initialization
981 # functions can call back into libc before thread initialization is
982 # complete and this leads to segmentation violations and other problems.
983 # Since libc contains no C++ code, linking with the minimal crti.o and
984 # crtn.o modules is safe and avoids the problems described above.
985 OBJECTS= $(CRTI) $(MOSTOBSJS) $(CRTN)
986 CRTSRCS= ../../common/sparcv9

988 # include common library definitions
989 include $(SRC)/lib/Makefile.lib
990 include $(SRC)/lib/Makefile.lib.64

992 # we need to override the default SONAME here because we might
993 # be building a variant object (still libc.so.1, but different filename)
994 SONAME = libc.so.1

996 CFLAGS64 += $(CCVERBOSE)

998 # This is necessary to avoid problems with calling _ex_unwind().
999 # We probably don't want any inlining anyway.
1000 CFLAGS64 += -xinline=

1002 CERRWARN += -_gcc=-Wno-parentheses
1003 CERRWARN += -_gcc=-Wno-switch
1004 CERRWARN += -_gcc=-Wno-uninitialized
1005 CERRWARN += -_gcc=-Wno-unused-value
1006 CERRWARN += -_gcc=-Wno-unused-label
1007 CERRWARN += -_gcc=-Wno-unused-variable
1008 CERRWARN += -_gcc=-Wno-type-limits
1009 CERRWARN += -_gcc=-Wno-char-subscripts
1010 CERRWARN += -_gcc=-Wno-clobbered
1011 CERRWARN += -_gcc=-Wno-unused-function
1012 CERRWARN += -_gcc=-Wno-address

1014 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
1015 # enables ASSERT() checking in the threads portion of the library.
1016 # This is automatically enabled for DEBUG builds, not for non-debug builds.
1017 THREAD_DEBUG =
1018 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

1020 # Make string literals read-only to save memory.
1021 CFLAGS64 += $(XSTRCONST)

1023 ALTPICS= $(TRACEOBJS:%=pics/%)

1025 $(DYNLIB) := BUILD.SO = $(LD) -o $$@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) $(EXTPICS)

1027 MAPFILES = $(LIBCDIR)/port/mapfile-vers

1029 sparcv9_C_PICFLAGS= -K PIC
1030 CFLAGS64 += $(EXTN_CFLAGS)
1031 CPPFLAGS= -D_REENTRANT -Dsparc $(EXTN_CPPFLAGS) $(THREAD_DEBUG) \
1032 -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1033 ASFLAGS= $(EXTN_ASFLAGS) -K PIC -P -D__STDC__ -D__ASM -D__sparcv9 $(CPPFLA
1034 $(sparcv9_AS_XARCH)

1036 # As a favor to the dtrace syscall provider, libc still calls the
1037 # old syscall traps that have been obsoleted by the *at() interfaces.
1038 # Delete this to compile libc using only the new *at() system call traps
1039 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1041 # Inform the run-time linker about libc specialized initialization
1042 RTLDINFO = -z rtldinfo=tlstldinfo

```

```

1043 DYNFLAGS +=      $(RTLDINFO)

1045 # Force libc's internal references to be resolved immediately upon loading
1046 # in order to avoid critical region problems.  Since almost all libc symbols
1047 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1048 DYNFLAGS +=      -znov

1050 DYNFLAGS +=      $(EXTN_DYNFLAGS)

1052 BUILD.s=         $(AS) $(ASFLAGS) $< -o $@

1054 # Override this top level flag so the compiler builds in its native
1055 # C99 mode.  This has been enabled to support the complex arithmetic
1056 # added to libc.
1057 C99MODE=         $(C99_ENABLE)

1059 # libc method of building an archive
1060 # The "$(GREP) -v ' L '" part is necessary only until
1061 # lorder is fixed to ignore thread-local variables.
1062 BUILD.AR=        $(RM) $@ ; \
1063                $(AR) q $@ `$(LORDER) $(MOSTOBSJS:%=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR

1065 # extra files for the clean target
1066 CLEANFILES=      \
1067                $(LIBCDIR)/port/gen/errlst.c \
1068                $(LIBCDIR)/port/gen/new_list.c \
1069                assym.h \
1070                genassym \
1071                pics/crti.o \
1072                pics/crtn.o \
1073                $(ALTPICS)

1075 CLOBBERFILES += $(LIB_PIC)

1077 # list of C source for lint
1078 SRCS=            \
1079                $(ATOMICOBJS:%.o=$(SRC)/common/atomic/%.c) \
1080                $(XATTROBJS:%.o=$(SRC)/common/xattr/%.c) \
1081                $(COMOBJS:%.o=$(SRC)/common/util/%.c) \
1082                $(PORTFP:%.o=$(LIBCDIR)/port/fp/%.c) \
1083                $(PORTGEN:%.o=$(LIBCDIR)/port/gen/%.c) \
1084                $(PORTI18N:%.o=$(LIBCDIR)/port/i18n/%.c) \
1085                $(PORTLOCALE:%.o=$(LIBCDIR)/port/locale/%.c) \
1086                $(PORTPRINT:%.o=$(LIBCDIR)/port/print/%.c) \
1087                $(PORTREGEX:%.o=$(LIBCDIR)/port/regex/%.c) \
1088                $(PORTSTDIO:%.o=$(LIBCDIR)/port/stdio/%.c) \
1089                $(PORTSYS:%.o=$(LIBCDIR)/port/sys/%.c) \
1090                $(AIOOBJS:%.o=$(LIBCDIR)/port/aio/%.c) \
1091                $(RTOBJS:%.o=$(LIBCDIR)/port/rt/%.c) \
1092                $(TPOOLBJS:%.o=$(LIBCDIR)/port/tpool/%.c) \
1093                $(THREADSOBJS:%.o=$(LIBCDIR)/port/threads/%.c) \
1094                $(THREADSMACHOBJS:%.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1095                $(UNICODEOBJS:%.o=$(SRC)/common/unicode/%.c) \
1096                $(UNWINDMACHOBJS:%.o=$(LIBCDIR)/port/unwind/%.c) \
1097                $(FPOBJS:%.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1098                $(FPOBJS64:%.o=$(LIBCBASE)/fp/%.c) \
1099                $(LIBCBASE)/crt/_ftou.c \
1100                $(LIBCBASE)/gen/_xregs_clrptr.c \
1101                $(LIBCBASE)/gen/byteorder.c \
1102                $(LIBCBASE)/gen/ecvt.c \
1103                $(LIBCBASE)/gen/getctxt.c \
1104                $(LIBCBASE)/gen/makectxt.c \
1105                $(LIBCBASE)/gen/siginfofst.c \
1106                $(LIBCBASE)/gen/siglongjmp.c \
1107                $(LIBCBASE)/gen/swapctxt.c

```

```

1109 # conditional assignments
1110 $(DYNLIB) := CRTI = crt.i.o
1111 $(DYNLIB) := CRTN = crtn.o

1113 # Files which need the threads .il inline template
1114 TIL=            \
1115                aio.o \
1116                alloc.o \
1117                assfail.o \
1118                atexit.o \
1119                atfork.o \
1120                cancel.o \
1121                door_calls.o \
1122                err.o \
1123                errno.o \
1124                getctxt.o \
1125                lwp.o \
1126                ma.o \
1127                machdep.o \
1128                posix_aio.o \
1129                pthr_attr.o \
1130                pthr_barrier.o \
1131                pthr_cond.o \
1132                pthr_mutex.o \
1133                pthr_rwlock.o \
1134                pthread.o \
1135                rand.o \
1136                rwlock.o \
1137                scalls.o \
1138                sched.o \
1139                sema.o \
1140                sigaction.o \
1141                sigev_thread.o \
1142                spawn.o \
1143                stack.o \
1144                swapctxt.o \
1145                synch.o \
1146                tdb_agent.o \
1147                thr.o \
1148                thread_interface.o \
1149                thread_pool.o \
1150                tls.o \
1151                tsd.o \
1152                unwind.o

1154 $(TIL:%=pics%) := CFLAGS64 += $(LIBCBASE)/threads/sparcv9.il

1156 # Files in fp, port/fp subdirectories that need base.il inline template
1157 IL=            \
1158                __flt_decim.o \
1159                decimal_bin.o

1161 $(IL:%=pics%) := CFLAGS64 += $(LIBCBASE)/fp/base.il

1163 # Files in fp subdirectory which need __quad.il inline template
1164 QIL=            \
1165                _Q_add.o \
1166                _Q_cmp.o \
1167                _Q_cmpe.o \
1168                _Q_div.o \
1169                _Q_dtoq.o \
1170                _Q_fcc.o \
1171                _Q_mul.o \
1172                _Q_qtod.o \
1173                _Q_qtoi.o \
1174                _Q_qtos.o

```

```

1175     _Q_qtoui.o           \
1176     _Q_sqrt.o           \
1177     _Q_stoq.o           \
1178     _Q_sub.o            \
1179     _Qp_qtox.o          \
1180     _Qp_qtoux.o         \

1182 $(QIL:%=pics/%) := CFLAGS64 += $(LIBCDIR)/$(MACH)/fp/__quad.il
1183 pics/_Op%.o := CFLAGS64 += -I$(LIBCDIR)/$(MACH)/fp
1184 pics/_Q%.o := sparcv9_COPTFLAG = -xO4 -xchip=ultra

1186 # Files in crt subdirectory which need muldiv64.il inline template
1187 #CIL= mul64.o divrem64.o
1188 #$(CIL:%=pics/%) := CFLAGS += $(LIBCBASE)/crt/mul64.il

1190 # large-file-aware components that should be built large

1192 #$(COMSYSOBS64:%=pics/%) := \
1193 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1195 #$(SYSOBS64:%=pics/%) := \
1196 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1198 #$(PORTGEN64:%=pics/%) := \
1199 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1201 #$(PORTSTDIO64:%=pics/%) := \
1202 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1204 #$(PORTSYS64:%=pics/%) := \
1205 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1207 $(PORTSTDIO_W:%=pics/%) := \
1208 CPPFLAGS += -D_WIDE

1210 $(PORTPRINT_W:%=pics/%) := \
1211 CPPFLAGS += -D_WIDE

1213 $(PORTI18N_COND:%=pics/%) := \
1214 CPPFLAGS += -D_WCS_LOGLONG

1216 # Files which need extra optimization
1217 pics/getenv.o := sparcv9_COPTFLAG = -xO4

1219 .KEEP_STATE:

1221 all: $(LIBS) $(LIB_PIC)

1223 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1224 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1225 lint := LINTFLAGS64 += -mn

1227 lint:
1228 @echo $(LINT.c) ... $(LDLIBS)
1229 @$$(LINT.c) $(SRCS) $(LDLIBS)

1231 $(LINTLIB) := SRCS=$(LIBCDIR)/port/l1ib-1c
1232 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1233 $(LINTLIB) := LINTFLAGS64=-nvx -m64

1235 # object files that depend on inline template
1236 $(TIL:%=pics/%) := $(LIBCBASE)/threads/sparcv9.il
1237 $(IL:%=pics/%) := $(LIBCBASE)/fp/base.il
1238 $(QIL:%=pics/%) := $(LIBCDIR)/$(MACH)/fp/__quad.il
1239 #$(CIL:%=pics/%) := $(LIBCBASE)/crt/muldiv64.il

```

```

1241 # include common libc targets
1242 include $(LIBCDIR)/Makefile.targ

1244 # We need to strip out all CTF and DOF data from the static library
1245 $(LIB_PIC) := DIR = pics
1246 $(LIB_PIC): pics $$ (PICS)
1247     $(BUILD.AR)
1248     $(MCS) -d -n .SUNW_ctf $$@ > /dev/null 2>&1
1249     $(MCS) -d -n .SUNW_dof $$@ > /dev/null 2>&1
1250     $(AR) -ts $$@ > /dev/null
1251     $(POST_PROCESS_A)

1253 # special cases
1254 #$(STRETS:%=pics/%) := crt/stret.s
1255 # $(AS) $(ASFLAGS) -DSTRET$(@F:stret%.o=) crt/stret.s -o $$@
1256 # $(POST_PROCESS_O)

1258 #crt/_rtbootld.s: crt/_rtboot.s crt/_rtld.c
1259 # $(CC) $(CPPFLAGS) -O -S -K pic crt/_rtld.c -o crt/_rtld.s
1260 # $(CAT) crt/_rtboot.s crt/_rtld.s > $$@
1261 # $(RM) crt/_rtld.s

1263 ASSYMDEP_OBJS= \
1264     _lwp_mutex_unlock.o \
1265     _stack_grow.o \
1266     asm_subr.o \
1267     setjmp.o \
1268     smt_pause.o \
1269     tls_get_addr.o \
1270     unwind_frame.o \
1271     vforkx.o

1273 $(ASSYMDEP_OBJS:%=pics/%) := CPPFLAGS += -I.

1275 $(ASSYMDEP_OBJS:%=pics/%) := assym.h

1277 # assym.h build rules

1279 assym.h := CFLAGS64 += -g

1281 GENASSYM_C = $(LIBCDIR)/$(MACH)/genassym.c

1283 genassym: $(GENASSYM_C)
1284     $(NATIVECC) -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc \
1285     $(CPPFLAGS.native) -o $$@ $(GENASSYM_C)

1287 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in

1289 assym.h: $(OFFSETS) genassym
1290     $(OFFSETS_CREATE) <$(OFFSETS) >$$@
1291     ./genassym >>$$@

1293 # derived C source and related explicit dependencies
1294 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/
1295     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist

1297 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c

```

```

*****
2452 Fri Jul 11 11:43:06 2014
new/usr/src/lib/libumem/common/stub_stand.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
24 * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * Copyright (c) 2012, Joyent, Inc. All rights reserved.
29 */

31 /*
32 * Stubs for the standalone to reduce the dependence on external libraries
33 */

35 #include <string.h>
36 #include "misc.h"

38 /*ARGSUSED*/
39 int
40 cond_init(cond_t *cvp, int type, void *arg)
41 {
42     return (0);
43 }
  unchanged_portion_omitted_

145 /*ARGSUSED*/
146 void
147 _tmem_set_cleanup(void (*f)(int, void *))
148 {
149 }

151 int
152 isspace(int c)
153 {
154     switch (c) {
155         case ' ':
156         case '\t':
157         case '\n':

```

```

158     case '\r':
159     case '\f':
160     case '\v':
161         return (1);
162     }
163     return (0);
164 }

```

unchanged\_portion\_omitted\_

```

*****
64012 Fri Jul 11 11:43:07 2014
new/usr/src/man/man3c/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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
9 # at http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
15 # Copyright 2013, OmniTI Computer Consulting, Inc. All rights reserved.
16 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
17 #
18 #
19 include      $(SRC)/Makefile.master
20 #
21 MANSECT=     3c
22 #
23 MANFILES=    ___fbufsize.3c
24               ___longjmp.3c
25               ___stack_grow.3c
26               a64l.3c
27               abort.3c
28               abs.3c
29               addsev.3c
30               addseverity.3c
31               aio_cancel.3c
32               aio_error.3c
33               aio_fsync.3c
34               aio_read.3c
35               aio_return.3c
36               aio_suspend.3c
37               aio_waitn.3c
38               aio_write.3c
39               aiocancel.3c
40               aioread.3c
41               aiowait.3c
42               assert.3c
43               atexit.3c
44               atomic_add.3c
45               atomic_and.3c
46               atomic_bits.3c
47               atomic_cas.3c
48               atomic_dec.3c
49               atomic_inc.3c
50               atomic_ops.3c
51               atomic_or.3c
52               atomic_swap.3c
53               attropen.3c
54               basename.3c
55               bsd_signal.3c
56               bsearch.3c
57               bstring.3c
58               btowc.3c

```

```

59               catgets.3c
60               catopen.3c
61               cfgetispeed.3c
62               cfsetispeed.3c
63               clock.3c
64               clock_nanosleep.3c
65               clock_settime.3c
66               closedir.3c
67               closefrom.3c
68               cond_init.3c
69               confstr.3c
70               crypt.3c
71               crypt_genhash_impl.3c
72               crypt_gensalt.3c
73               crypt_gensalt_impl.3c
74               cset.3c
75               ctermid.3c
76               ctime.3c
77               ctype.3c
78               cuserid.3c
79               daemon.3c
80               decimal_to_floating.3c
81               difftime.3c
82               directio.3c
83               dirfd.3c
84               dirname.3c
85               div.3c
86               dladdr.3c
87               dlclose.3c
88               dldump.3c
89               dlerror.3c
90               dlinfo.3c
91               dlopen.3c
92               dlsym.3c
93               door_bind.3c
94               door_call.3c
95               door_create.3c
96               door_cred.3c
97               door_getparam.3c
98               door_info.3c
99               door_return.3c
100              door_revoke.3c
101              door_server_create.3c
102              door_ucred.3c
103              drand48.3c
104              dup2.3c
105              econvert.3c
106              ecvt.3c
107              enable_extended_FILE_stdio.3c
108              encrypt.3c
109              end.3c
110              err.3c
111              euclen.3c
112              exit.3c
113              fattach.3c
114              fclose.3c
115              fdasync.3c
116              fdetach.3c
117              fdopen.3c
118              ferror.3c
119              fflush.3c
120              ffs.3c
121              fgetattr.3c
122              fgetc.3c
123              fgetpos.3c
124              fgetwc.3c

```

```

125 floating_to_decimal.3c
126 flockfile.3c
127 fmtmsg.3c
128 fnmatch.3c
129 fopen.3c
130 fpgetround.3c
131 fputc.3c
132 fputwc.3c
133 fputws.3c
134 fread.3c
135 freopen.3c
136 fseek.3c
137 fsetpos.3c
138 fsync.3c
139 ftell.3c
140 ftime.3c
141 ftok.3c
142 ftw.3c
143 fwide.3c
144 fwprintf.3c
145 fwrite.3c
146 fwscanf.3c
147 getcuid.3c
148 getcwd.3c
149 getdate.3c
150 getdtablesize.3c
151 getenv.3c
152 getexecname.3c
153 getgrnam.3c
154 gethostid.3c
155 gethostname.3c
156 gethrtime.3c
157 getline.3c
158 getloadavg.3c
159 getlogin.3c
160 getmntent.3c
161 getnetgrent.3c
162 getopt.3c
163 getpagesize.3c
164 getpagesizes.3c
165 getpass.3c
166 getpeercred.3c
167 getpriority.3c
168 getpw.3c
169 getpwnam.3c
170 getrusage.3c
171 gets.3c
172 getspnam.3c
173 getsubopt.3c
174 gettext.3c
175 gettimeofday.3c
176 gettxt.3c
177 getusershell.3c
178 getutent.3c
179 getutxent.3c
180 getvfsent.3c
181 getwc.3c
182 getwchar.3c
183 getwd.3c
184 getwidth.3c
185 getws.3c
186 getzoneid.3c
187 glob.3c
188 grantpt.3c
189 hsearch.3c
190 iconv.3c

```

```

191 iconv_close.3c
192 iconv_open.3c
193 imaxabs.3c
194 imaxdiv.3c
195 index.3c
196 initgroups.3c
197 insque.3c
198 is_system_labeled.3c
199 isaexec.3c
200 isastream.3c
201 isatty.3c
202 isnand.3c
203 iswalph.3c
204 iswctype.3c
205 killpg.3c
206 lckpddf.3c
207 lfmt.3c
208 lio_listio.3c
209 localeconv.3c
210 lockf.3c
211 lsearch.3c
212 madvise.3c
213 makecontext.3c
214 makedev.3c
215 malloc.3c
216 mblen.3c
217 mbrlen.3c
218 mbrtowc.3c
219 mbsrinit.3c
220 mbsrtowcs.3c
221 mbtowc.3c
222 membar_ops.3c
223 memory.3c
224 mkfifo.3c
225 mkstemp.3c
226 mktemp.3c
227 mkttime.3c
228 mlock.3c
229 mlockall.3c
230 monitor.3c
231 mq_close.3c
232 mq_getattr.3c
233 mq_notify.3c
234 mq_open.3c
235 mq_receive.3c
236 mq_send.3c
237 mq_setattr.3c
238 mq_unlink.3c
239 msync.3c
240 mutex_init.3c
241 nanosleep.3c
242 ndbm.3c
243 newlocale.3c
244 nl_langinfo.3c
245 offsetof.3c
246 opendir.3c
247 perror.3c
248 pfmt.3c
249 plock.3c
250 popen.3c
251 port_alert.3c
252 port_associate.3c
253 port_create.3c
254 port_get.3c
255 port_send.3c
256 posix_fadvise.3c

```

```

257     posix_fallocate.3c //
258     posix_madvise.3c //
259     posix_memalign.3c //
260     posix_openpt.3c //
261     posix_spawn.3c //
262     posix_spawn_file_actions_addclose.3c //
263     posix_spawn_file_actions_addclosefrom_np.3c //
264     posix_spawn_file_actions_adddup2.3c //
265     posix_spawn_file_actions_destroy.3c //
266     posix_spawn_pipe_np.3c //
267     posix_spawnattr_destroy.3c //
268     posix_spawnattr_getflags.3c //
269     posix_spawnattr_getpgroup.3c //
270     posix_spawnattr_getschedparam.3c //
271     posix_spawnattr_getschedpolicy.3c //
272     posix_spawnattr_getsigdefault.3c //
273     posix_spawnattr_getsigignore_np.3c //
274     posix_spawnattr_getsigmask.3c //
275     printf.3c //
276     priv_addset.3c //
277     priv_set.3c //
278     priv_str_to_set.3c //
279     pset_getloadavg.3c //
280     psignal.3c //
281     pthread_atfork.3c //
282     pthread_attr_getdetachstate.3c //
283     pthread_attr_getguardsize.3c //
284     pthread_attr_getinheritsched.3c //
285     pthread_attr_getschedparam.3c //
286     pthread_attr_getschedpolicy.3c //
287     pthread_attr_getscope.3c //
288     pthread_attr_getstack.3c //
289     pthread_attr_getstackaddr.3c //
290     pthread_attr_getstacksize.3c //
291     pthread_attr_init.3c //
292     pthread_barrier_destroy.3c //
293     pthread_barrier_wait.3c //
294     pthread_barrierattr_destroy.3c //
295     pthread_barrierattr_getpshared.3c //
296     pthread_cancel.3c //
297     pthread_cleanup_pop.3c //
298     pthread_cleanup_push.3c //
299     pthread_cond_init.3c //
300     pthread_cond_signal.3c //
301     pthread_cond_wait.3c //
302     pthread_condattr_getclock.3c //
303     pthread_condattr_getpshared.3c //
304     pthread_condattr_init.3c //
305     pthread_create.3c //
306     pthread_detach.3c //
307     pthread_equal.3c //
308     pthread_exit.3c //
309     pthread_getconcurrency.3c //
310     pthread_getschedparam.3c //
311     pthread_getspecific.3c //
312     pthread_join.3c //
313     pthread_key_create.3c //
314     pthread_key_delete.3c //
315     pthread_kill.3c //
316     pthread_mutex_getprioceiling.3c //
317     pthread_mutex_init.3c //
318     pthread_mutex_lock.3c //
319     pthread_mutex_timedlock.3c //
320     pthread_mutexattr_getprioceiling.3c //
321     pthread_mutexattr_getprotocol.3c //
322     pthread_mutexattr_getpshared.3c //

```

```

323     pthread_mutexattr_gettype.3c //
324     pthread_mutexattr_init.3c //
325     pthread_once.3c //
326     pthread_rwlock_init.3c //
327     pthread_rwlock_rdlock.3c //
328     pthread_rwlock_timedrdlock.3c //
329     pthread_rwlock_timedwrlock.3c //
330     pthread_rwlock_unlock.3c //
331     pthread_rwlock_wrlock.3c //
332     pthread_rwlockattr_getpshared.3c //
333     pthread_rwlockattr_init.3c //
334     pthread_self.3c //
335     pthread_setcancelstate.3c //
336     pthread_setcanceltype.3c //
337     pthread_setschedprio.3c //
338     pthread_sigmask.3c //
339     pthread_spin_destroy.3c //
340     pthread_spin_lock.3c //
341     pthread_spin_unlock.3c //
342     pthread_testcancel.3c //
343     ptrace.3c //
344     ptsname.3c //
345     putenv.3c //
346     putpwent.3c //
347     puts.3c //
348     putspent.3c //
349     putws.3c //
350     qsort.3c //
351     raise.3c //
352     rand.3c //
353     random.3c //
354     rctl_walk.3c //
355     rctlblk_set_value.3c //
356     re_comp.3c //
357     readdir.3c //
358     realpath.3c //
359     reboot.3c //
360     regcmp.3c //
361     regcomp.3c //
362     remove.3c //
363     rewind.3c //
364     rewinddir.3c //
365     rlock.3c //
366     scandir.3c //
367     scanf.3c //
368     sched_get_priority_max.3c //
369     sched_getparam.3c //
370     sched_getscheduler.3c //
371     sched_rr_get_interval.3c //
372     sched_setparam.3c //
373     sched_setscheduler.3c //
374     sched_yield.3c //
375     schedctl_init.3c //
376     seekdir.3c //
377     select.3c //
378     sem_close.3c //
379     sem_destroy.3c //
380     sem_getvalue.3c //
381     sem_init.3c //
382     sem_open.3c //
383     sem_post.3c //
384     sem_timedwait.3c //
385     sem_unlink.3c //
386     sem_wait.3c //
387     semaphore.3c //
388     setbuf.3c //

```



```

389         setbuffer.3c
390         setcat.3c
391         setenv.3c
392         setjmp.3c
393         setkey.3c
394         setlabel.3c
395         setlocale.3c
396         shm_open.3c
397         shm_unlink.3c
398         sigfpe.3c
399         siginterrupt.3c
400         signal.3c
401         sigqueue.3c
402         sigsetops.3c
403         sigstack.3c
404         sigwaitinfo.3c
405         sleep.3c
406         ssignal.3c
407         stack_getbounds.3c
408         stack_inbounds.3c
409         stack_setbounds.3c
410         stack_violation.3c
411         stdio.3c
412         str2sig.3c
413         strcoll.3c
414         strerror.3c
415         strfmon.3c
416         strftime.3c
417         string.3c
418         string_to_decimal.3c
419         strptime.3c
420         strsignal.3c
421         strtod.3c
422         strtoumax.3c
423         strtol.3c
424         strtoul.3c
425         strtows.3c
426         strxfrm.3c
427         swab.3c
428         sync_instruction_memory.3c
429         sysconf.3c
430         syslog.3c
431         system.3c
432         tdrain.3c
433         tcflow.3c
434         tcflush.3c
435         tcgetattr.3c
436         togetpgrp.3c
437         togetsid.3c
438         tcsendbreak.3c
439         tcsetattr.3c
440         tcsetpgrp.3c
441         tell.3c
442         telldir.3c
443         termios.3c
444         thr_create.3c
445         thr_exit.3c
446         thr_getconcurrency.3c
447         thr_getprio.3c
448         thr_join.3c
449         thr_keycreate.3c
450         thr_kill.3c
451         thr_main.3c
452         thr_min_stack.3c
453         thr_self.3c
454         thr_sigsetmask.3c

```

```

455         thr_stksegment.3c
456         thr_suspend.3c
457         thr_yield.3c
458         timer_create.3c
459         timer_delete.3c
460         timer_settime.3c
461         timeradd.3c
462         tmpfile.3c
463         tmpnam.3c
464         toascii.3c
465         tolower.3c
466         toupper.3c
467         towlower.3c
468         toupper.3c
469         truncate.3c
470         tsearch.3c
471         ttyname.3c
472         ttyslot.3c
473         u8_strcmp.3c
474         u8_textprep_str.3c
475         u8_validate.3c
476         ualarm.3c
477         uconv_ul6tou32.3c
478         ucred_get.3c
479         ungetc.3c
480         ungetwc.3c
481         unlockpt.3c
482         unsetenv.3c
483         uselocale.3c
484         usleep.3c
485         vfwprintf.3c
486         vlfmt.3c
487         vpfmt.3c
488         vprintf.3c
489         vsyslog.3c
490         wait.3c
491         wait3.3c
492         waitpid.3c
493         walkcontext.3c
494         wctomb.3c
495         wcscoll.3c
496         wcsftime.3c
497         wcsrtombs.3c
498         wcsstr.3c
499         wcstod.3c
500         wcstoumax.3c
501         wcstol.3c
502         wcstoul.3c
503         wcstring.3c
504         wcswidth.3c
505         wcsxfrm.3c
506         wctob.3c
507         wctomb.3c
508         wctrans.3c
509         wctype.3c
510         wcwidth.3c
511         wmemchr.3c
512         wmemcmp.3c
513         wmemcpy.3c
514         wmemmove.3c
515         wmemset.3c
516         wordexp.3c
517         wsprintf.3c
518         wsscanf.3c
519         wstring.3c

```

```

521 MANLINKS=      FD_CLR.3c      \
522                FD_ISSET.3c     \
523                FD_SET.3c       \
524                FD_ZERO.3c      \
525                __flbf.3c       \
526                __fpending.3c   \
527                __fpurge.3c     \
528                __freadable.3c  \
529                __freanding.3c  \
530                __fsetlocking.3c \
531                __fwritable.3c  \
532                __fwriting.3c   \
533                _edata.3c       \
534                _end.3c         \
535                _etext.3c       \
536                _exithandle.3c  \
537                _flushlbf.3c    \
538                _setjmp.3c      \
539                addrtosymstr.3c \
540                aiowrite.3c     \
541                alloca.3c       \
542                alphasort.3c    \
543                ascftime.3c     \
544                asctime.3c      \
545                asctime_r.3c    \
546                asprintf.3c     \
547                atof.3c         \
548                atoi.3c         \
549                atol.3c         \
550                atoll.3c        \
551                atomic_add_16.3c \
552                atomic_add_16_nv.3c \
553                atomic_add_32.3c \
554                atomic_add_32_nv.3c \
555                atomic_add_64.3c \
556                atomic_add_64_nv.3c \
557                atomic_add_8.3c \
558                atomic_add_8_nv.3c \
559                atomic_add_char.3c \
560                atomic_add_char_nv.3c \
561                atomic_add_int.3c \
562                atomic_add_int_nv.3c \
563                atomic_add_long.3c \
564                atomic_add_long_nv.3c \
565                atomic_add_ptr.3c \
566                atomic_add_ptr_nv.3c \
567                atomic_add_short.3c \
568                atomic_add_short_nv.3c \
569                atomic_and_16.3c \
570                atomic_and_16_nv.3c \
571                atomic_and_32.3c \
572                atomic_and_32_nv.3c \
573                atomic_and_64.3c \
574                atomic_and_64_nv.3c \
575                atomic_and_8.3c \
576                atomic_and_8_nv.3c \
577                atomic_and_uchar.3c \
578                atomic_and_uchar_nv.3c \
579                atomic_and_uint.3c \
580                atomic_and_uint_nv.3c \
581                atomic_and_ulong.3c \
582                atomic_and_ulong_nv.3c \
583                atomic_and_ushort.3c \
584                atomic_and_ushort_nv.3c \
585                atomic_cas_16.3c \
586                atomic_cas_32.3c \

```

```

587                atomic_cas_64.3c \
588                atomic_cas_8.3c \
589                atomic_cas_ptr.3c \
590                atomic_cas_uchar.3c \
591                atomic_cas_uint.3c \
592                atomic_cas_ulong.3c \
593                atomic_cas_ushort.3c \
594                atomic_clear_long_excl.3c \
595                atomic_dec_16.3c \
596                atomic_dec_16_nv.3c \
597                atomic_dec_32.3c \
598                atomic_dec_32_nv.3c \
599                atomic_dec_64.3c \
600                atomic_dec_64_nv.3c \
601                atomic_dec_8.3c \
602                atomic_dec_8_nv.3c \
603                atomic_dec_ptr.3c \
604                atomic_dec_ptr_nv.3c \
605                atomic_dec_uchar.3c \
606                atomic_dec_uchar_nv.3c \
607                atomic_dec_uint.3c \
608                atomic_dec_uint_nv.3c \
609                atomic_dec_ulong.3c \
610                atomic_dec_ulong_nv.3c \
611                atomic_dec_ushort.3c \
612                atomic_dec_ushort_nv.3c \
613                atomic_inc_16.3c \
614                atomic_inc_16_nv.3c \
615                atomic_inc_32.3c \
616                atomic_inc_32_nv.3c \
617                atomic_inc_64.3c \
618                atomic_inc_64_nv.3c \
619                atomic_inc_8.3c \
620                atomic_inc_8_nv.3c \
621                atomic_inc_ptr.3c \
622                atomic_inc_ptr_nv.3c \
623                atomic_inc_uchar.3c \
624                atomic_inc_uchar_nv.3c \
625                atomic_inc_uint.3c \
626                atomic_inc_uint_nv.3c \
627                atomic_inc_ulong.3c \
628                atomic_inc_ulong_nv.3c \
629                atomic_inc_ushort.3c \
630                atomic_inc_ushort_nv.3c \
631                atomic_or_16.3c \
632                atomic_or_16_nv.3c \
633                atomic_or_32.3c \
634                atomic_or_32_nv.3c \
635                atomic_or_64.3c \
636                atomic_or_64_nv.3c \
637                atomic_or_8.3c \
638                atomic_or_8_nv.3c \
639                atomic_or_uchar.3c \
640                atomic_or_uchar_nv.3c \
641                atomic_or_uint.3c \
642                atomic_or_uint_nv.3c \
643                atomic_or_ulong.3c \
644                atomic_or_ulong_nv.3c \
645                atomic_or_ushort.3c \
646                atomic_or_ushort_nv.3c \
647                atomic_set_long_excl.3c \
648                atomic_swap_16.3c \
649                atomic_swap_32.3c \
650                atomic_swap_64.3c \
651                atomic_swap_8.3c \
652                atomic_swap_ptr.3c \

```

```

653         atomic_swap_uchar.3c
654         atomic_swap_uint.3c
655         atomic_swap_ulong.3c
656         atomic_swap_ushort.3c
657         backtrace.3c
658         backtrace_symbols.3c
659         backtrace_symbols_fd.3c
660         bcmp.3c
661         bcopy.3c
662         bind_textdomain_codeset.3c
663         bindtextdomain.3c
664 btowc.1.3c
665         bzero.3c
666         calloc.3c
667         catclose.3c
668         cfgetospeed.3c
669         cfsetospeed.3c
670         cftime.3c
671         clearerr.3c
672         clock_getres.3c
673         clock_gettime.3c
674         closelog.3c
675         cond_broadcast.3c
676         cond_destroy.3c
677         cond_reltimedwait.3c
678         cond_signal.3c
679         cond_timedwait.3c
680         cond_wait.3c
681         csetcol.3c
682         csetlen.3c
683         csetno.3c
684         ctermid_r.3c
685         ctime_r.3c
686         dbm_clearerr.3c
687         dbm_close.3c
688         dbm_delete.3c
689         dbm_error.3c
690         dbm_fetch.3c
691         dbm_firstkey.3c
692         dbm_nextkey.3c
693         dbm_open.3c
694         dbm_store.3c
695         dcgettext.3c
696         dcgettext.3c
697         decimal_to_double.3c
698         decimal_to_extended.3c
699         decimal_to_quadruple.3c
700         decimal_to_single.3c
701         dgettext.3c
702         dladdr1.3c
703         dlmopen.3c
704         dngettext.3c
705         door_setparam.3c
706         door_unbind.3c
707         double_to_decimal.3c
708         dup3.3c
709 duplocale.3c
710         edata.3c
711         endgrent.3c
712         endnetgrent.3c
713         endpwent.3c
714         endspent.3c
715         endusershell.3c
716         endutent.3c
717         endutxent.3c
718         erand48.3c

```

```

719         errno.3c
720         errx.3c
721         etext.3c
722         euccol.3c
723         eukscol.3c
724         extended_to_decimal.3c
725         fconvert.3c
726         fcvt.3c
727         fdopendir.3c
728         fdwalk.3c
729         feof.3c
730         fgetgrent.3c
731         fgetgrent_r.3c
732         fgetpwent.3c
733         fgetpwent_r.3c
734         fgets.3c
735         fgetspent.3c
736         fgetspent_r.3c
737 fgetwc.1.3c
738         fgetws.3c
739         file_to_decimal.3c
740         fileno.3c
741         finite.3c
742         fpclass.3c
743         fpgetmask.3c
744         fpgetsticky.3c
745         fprintf.3c
746         fpsetmask.3c
747         fpsetround.3c
748         fpsetsticky.3c
749         fputs.3c
750         free.3c
751 freelocale.3c
752         fscanf.3c
753         fseeko.3c
754         fsetattr.3c
755         ftello.3c
756         ftruncate.3c
757         ftrylockfile.3c
758         func_to_decimal.3c
759         funlockfile.3c
760         gconvert.3c
761         gcvt.3c
762         getattrat.3c
763        getc.3c
764        getc_unlocked.3c
765         getchar.3c
766         getchar_unlocked.3c
767         getdelim.3c
768         getextmntent.3c
769         getgrent.3c
770         getgrent_r.3c
771         getgrgid.3c
772         getgrgid_r.3c
773         getgrnam_r.3c
774         gethomedgroup.3c
775         gethrvtime.3c
776         getlogin_r.3c
777         getmntany.3c
778         getnetgrent_r.3c
779         getpassphrase.3c
780         getpwent.3c
781         getpwent_r.3c
782         getpwnam_r.3c
783         getpwuid.3c
784         getpwuid_r.3c

```

```

785 getspent.3c
786 getspent_r.3c
787 getspnam_r.3c
788 getutid.3c
789 getutline.3c
790 getutmp.3c
791 getutmpx.3c
792 getutxid.3c
793 getutxline.3c
794 getvfsany.3c
795 getvfsfile.3c
796 getvfsspec.3c
797 getw.3c
798 getwc_1.3c
799 getwchar_1.3c
800 getzoneidbyname.3c
801 getzonenamebyid.3c
802 globfree.3c
803 gmtime.3c
804 gmtime_r.3c
805 gsignal.3c
806 hasmntopt.3c
807 hcreate.3c
808 hdestroy.3c
809 initstate.3c
810 innetgr.3c
811 isalnum.3c
812 isalnum_1.3c
813 isalpha.3c
814 isalpha_1.3c
815 isascii.3c
816 isblank.3c
817 isblank_1.3c
818 iscntrl.3c
819 iscntrl_1.3c
820 isdigit.3c
821 isdigit_1.3c
822 isenglish.3c
823 isgraph.3c
824 isgraph_1.3c
825 isideogram.3c
826 islower.3c
827 islower_1.3c
828 isnanf.3c
829 isnumber.3c
830 isphonogram.3c
831 isprint.3c
832 isprint_1.3c
833 ispunct.3c
834 ispunct_1.3c
835 isspace.3c
836 isspace_1.3c
837 isspecial.3c
838 isupper.3c
839 isupper_1.3c
840 iswalnum.3c
841 iswalnum_1.3c
842 iswalpha_1.3c
843 iswascii.3c
844 iswblank.3c
845 iswblank_1.3c
846 iswcntrl.3c
847 iswcntrl_1.3c
848 iswctype_1.3c
849 iswdigit.3c
850 iswdigit_1.3c

```

```

851 iswgraph.3c
852 iswgraph_1.3c
853 iswideogram.3c
854 iswideogram_1.3c
855 iswhexnumber.3c
856 iswhexnumber_1.3c
857 iswlower.3c
858 iswlower_1.3c
859 iswnumber.3c
860 iswnumber_1.3c
861 iswphonogram.3c
862 iswphonogram_1.3c
863 iswprint.3c
864 iswprint_1.3c
865 iswpunct.3c
866 iswpunct_1.3c
867 iswspace.3c
868 iswspace_1.3c
869 iswspecial.3c
870 iswspecial_1.3c
871 iswupper.3c
872 iswupper_1.3c
873 iswxdigit.3c
874 iswxdigit_1.3c
875 isxdigit.3c
876 isxdigit_1.3c
877 jrand48.3c
878 l64a.3c
879 labs.3c
880 lcong48.3c
881 ldiv.3c
882 lfind.3c
883 llabs.3c
884 lldiv.3c
885 lltostr.3c
886 localtime.3c
887 localtime_r.3c
888 longjmp.3c
889 lrand48.3c
890 major.3c
891 mblen_1.3c
892 mbrlen_1.3c
893 mbrtowc_1.3c
894 mbsinit_1.3c
895 mbsnrtowcs.3c
896 mbsnrtowcs_1.3c
897 mbsrtowcs_1.3c
898 mbstowcs.3c
899 mbstowcs_1.3c
900 mbtowc_1.3c
901 memalign.3c
902 membar_consumer.3c
903 membar_enter.3c
904 membar_exit.3c
905 membar_producer.3c
906 memccpy.3c
907 memchr.3c
908 memcmp.3c
909 memcpy.3c
910 memmove.3c
911 memset.3c
912 minor.3c
913 mkdtemp.3c
914 mkostemp.3c
915 mkostemps.3c
916 mkstemp.3c

```

```

917      mq_reltimedreceive_np.3c      //
918      mq_reltimedsend_np.3c        //
919      mq_timedreceive.3c           //
920      mq_timedsend.3c             //
921      mrand48.3c                  //
922      munlock.3c                  //
923      munlockall.3c               //
924      mutex_consistent.3c         //
925      mutex_destroy.3c            //
926      mutex_lock.3c               //
927      mutex_trylock.3c            //
928      mutex_unlock.3c             //
929      nl_langinfo.1.3c            //
930      nftw.3c                     //
931      ngettext.3c                 //
932      nrand48.3c                  //
933      openlog.3c                  //
934      pclose.3c                   //
935      port_dissociate.3c           //
936      port_getn.3c                //
937      port_sendn.3c               //
938      posix_spawn_file_actions_addopen.3c //
939      posix_spawn_file_actions_init.3c //
940      posix_spawnattr_init.3c      //
941      posix_spawnattr_setflags.3c  //
942      posix_spawnattr_setpgroup.3c //
943      posix_spawnattr_setschedparam.3c //
944      posix_spawnattr_setschedpolicy.3c //
945      posix_spawnattr_setsigdefault.3c //
946      posix_spawnattr_setsigignore_np.3c //
947      posix_spawnattr_setsigmask.3c //
948      posix_spawnnp.3c             //
949      printstack.3c                //
950      priv_allocset.3c             //
951      priv_basicset.3c             //
952      priv_copysset.3c             //
953      priv_delset.3c               //
954      priv_emptyset.3c             //
955      priv_fillset.3c              //
956      priv_freerset.3c             //
957      priv_getbyname.3c            //
958      priv_getbynum.3c             //
959      priv_getsetbyname.3c         //
960      priv_getsetbynum.3c          //
961      priv_gettext.3c              //
962      priv_ineffect.3c             //
963      priv_intersect.3c            //
964      priv_inverse.3c              //
965      priv_isemptyset.3c           //
966      priv_isequalset.3c           //
967      priv_isfullset.3c            //
968      priv_ismember.3c             //
969      priv_issubset.3c             //
970      priv_set_to_str.3c           //
971      priv_union.3c                //
972      pselect.3c                   //
973      psiginfo.3c                  //
974      pthread_attr_destroy.3c       //
975      pthread_attr_setdetachstate.3c //
976      pthread_attr_setguardsize.3c //
977      pthread_attr_setinheritsched.3c //
978      pthread_attr_setschedparam.3c //
979      pthread_attr_setschedpolicy.3c //
980      pthread_attr_setscope.3c     //
981      pthread_attr_setstack.3c     //
982      pthread_attr_setstackaddr.3c //

```

```

983      pthread_attr_setstacksize.3c //
984      pthread_barrier_init.3c      //
985      pthread_barrierattr_init.3c  //
986      pthread_barrierattr_setshared.3c //
987      pthread_cond_broadcast.3c    //
988      pthread_cond_destroy.3c      //
989      pthread_cond_reltimedwait_np.3c //
990      pthread_cond_timedwait.3c    //
991      pthread_condattr_destroy.3c  //
992      pthread_condattr_setclock.3c //
993      pthread_condattr_setshared.3c //
994      pthread_key_create_once_np.3c //
995      pthread_mutex_destroy.3c     //
996      pthread_mutex_reltimedlock_np.3c //
997      pthread_mutex_setprioceiling.3c //
998      pthread_mutex_trylock.3c     //
999      pthread_mutex_unlock.3c      //
1000     pthread_mutexattr_destroy.3c  //
1001     pthread_mutexattr_setprioceiling.3c //
1002     pthread_mutexattr_setprotocol.3c //
1003     pthread_mutexattr_setpshared.3c //
1004     pthread_mutexattr_settype.3c   //
1005     pthread_rwlock_destroy.3c      //
1006     pthread_rwlock_reltimedrdlock_np.3c //
1007     pthread_rwlock_reltimedwrlck_np.3c //
1008     pthread_rwlock_tryrdlock.3c    //
1009     pthread_rwlock_trywrlock.3c    //
1010     pthread_rwlockattr_destroy.3c  //
1011     pthread_rwlockattr_setpshared.3c //
1012     pthread_setconcurrency.3c      //
1013     pthread_setschedparam.3c       //
1014     pthread_setspecific.3c         //
1015     pthread_spin_init.3c           //
1016     pthread_spin_trylock.3c        //
1017     putc.3c                        //
1018     putc_unlocked.3c               //
1019     putchar.3c                     //
1020     putchar_unlocked.3c            //
1021     putmntent.3c                   //
1022     pututline.3c                   //
1023     pututxline.3c                  //
1024     putw.3c                        //
1025     putwc.3c                       //
1026     putwchar.3c                    //
1027     qeconvert.3c                   //
1028     qfconvert.3c                   //
1029     qgconvert.3c                   //
1030     quadruple_to_decimal.3c        //
1031     rand_r.3c                      //
1032     rctlblk_get_enforced_value.3c   //
1033     rctlblk_get_firing_time.3c     //
1034     rctlblk_get_global_action.3c    //
1035     rctlblk_get_global_flags.3c    //
1036     rctlblk_get_local_action.3c    //
1037     rctlblk_get_local_flags.3c     //
1038     rctlblk_get_privilege.3c       //
1039     rctlblk_get_recipient_pid.3c   //
1040     rctlblk_get_value.3c           //
1041     rctlblk_set_local_action.3c    //
1042     rctlblk_set_local_flags.3c     //
1043     rctlblk_set_privilege.3c       //
1044     rctlblk_set_recipient_pid.3c   //
1045     rctlblk_size.3c                //
1046     re_exec.3c                     //
1047     readdir_r.3c                   //
1048     realloc.3c                     //

```

```

1049         regerror.3c
1050         regex.3c
1051         regexec.3c
1052         regfree.3c
1053         remque.3c
1054         resetmnttab.3c
1055         rindex.3c
1056         rw_rlock.3c
1057         rw_tryrdlock.3c
1058         rw_trywrlock.3c
1059         rw_unlock.3c
1060         rw_wlock.3c
1061         rwlock_destroy.3c
1062         rwlock_init.3c
1063         sched_get_priority_min.3c
1064         schedctl_exit.3c
1065         schedctl_lookup.3c
1066         schedctl_start.3c
1067         schedctl_stop.3c
1068         seconvert.3c
1069         seed48.3c
1070         sem_reltimedwait_np.3c
1071         sem_trywait.3c
1072         sema_destroy.3c
1073         sema_init.3c
1074         sema_post.3c
1075         sema_trywait.3c
1076         sema_wait.3c
1077         setattrat.3c
1078         setgrent.3c
1079         sethostname.3c
1080         setlinebuf.3c
1081         setlogmask.3c
1082         setnetgrent.3c
1083         setpriority.3c
1084         setpwent.3c
1085         setspent.3c
1086         setstate.3c
1087         settimeofday.3c
1088         setusershell.3c
1089         setutent.3c
1090         setutxent.3c
1091         setvbuf.3c
1092         sfconvert.3c
1093         sgconvert.3c
1094         sig2str.3c
1095         sigaddset.3c
1096         sigdelset.3c
1097         sigemptyset.3c
1098         sigfillset.3c
1099         sighold.3c
1100         sigignore.3c
1101         sigismember.3c
1102         siglongjmp.3c
1103         sigpause.3c
1104         sigrelse.3c
1105         sigset.3c
1106         sigsetjmp.3c
1107         sigtimedwait.3c
1108         single_to_decimal.3c
1109         snprintf.3c
1110         sprintf.3c
1111         srand.3c
1112         srand48.3c
1113         srandom.3c
1114         sscanf.3c

```

```

1115         stderr.3c
1116         stdin.3c
1117         stdout.3c
1118         strcasecmp.3c
1119         strcasecmp.1.3c
1120         strcat.3c
1121         strchr.3c
1122         strcmp.3c
1123         strcoll.1.3c
1124         strcpy.3c
1125         strespn.3c
1126         strdup.3c
1127         strerror_r.3c
1128         strfmon.1.3c
1129         strftime.1.3c
1130         strlcat.3c
1131         strlcpy.3c
1132         strlen.3c
1133         strncasecmp.3c
1134         strncasecmp.1.3c
1135         strncat.3c
1136         strncmp.3c
1137         strncpy.3c
1138         strnlen.3c
1139         strpbrk.3c
1140         strptime.1.3c
1141         strrchr.3c
1142         strsep.3c
1143         strspn.3c
1144         strstr.3c
1145         strtod.3c
1146         strtok.3c
1147         strtok_r.3c
1148         strtold.3c
1149         strtoll.3c
1150         strtoull.3c
1151         strtoumax.3c
1152         strxfrm.1.3c
1153         swapcontext.3c
1154         swprintf.3c
1155         swscanf.3c
1156         tdelete.3c
1157         tempnam.3c
1158         textdomain.3c
1159         tfind.3c
1160         thr_continue.3c
1161         thr_getspecific.3c
1162         thr_keycreate_once.3c
1163         thr_setconcurrency.3c
1164         thr_setprio.3c
1165         thr_setspecific.3c
1166         timer_getoverrun.3c
1167         timer_gettime.3c
1168         timerclear.3c
1169         timercmp.3c
1170         timerisset.3c
1171         timersub.3c
1172         tmpnam_r.3c
1173         tolower.1.3c
1174         toupper.1.3c
1175         towctrans.3c
1176         towctrans.1.3c
1177         towlower.1.3c
1178         toupper.1.3c
1179         ttyname_r.3c
1180         twalk.3c

```

```

1181 tzset.3c
1182 uconv_ul6tou8.3c
1183 uconv_u32tou16.3c
1184 uconv_u32tou8.3c
1185 uconv_u8tou16.3c
1186 uconv_u8tou32.3c
1187 ucred_free.3c
1188 ucred_getegid.3c
1189 ucred_geteuid.3c
1190 ucred_getgroups.3c
1191 ucred_getlabel.3c
1192 ucred_getpflags.3c
1193 ucred_getpid.3c
1194 ucred_getprivset.3c
1195 ucred_getprojid.3c
1196 ucred_getrgid.3c
1197 ucred_getruid.3c
1198 ucred_getsgid.3c
1199 ucred_getsuid.3c
1200 ucred_getzoneid.3c
1201 ucred_size.3c
1202 ulckpwdf.3c
1203 ulltostr.3c
1204 unordered.3c
1205 updwtmp.3c
1206 updwtmpx.3c
1207 utmpname.3c
1208 utmpxname.3c
1209 valloc.3c
1210 vasprintf.3c
1211 verr.3c
1212 verrx.3c
1213 vfprintf.3c
1214 vfscanf.3c
1215 vfwscanf.3c
1216 vscanf.3c
1217 vsnprintf.3c
1218 vsprintf.3c
1219 vsscanf.3c
1220 vswprintf.3c
1221 vswscanf.3c
1222 vwarn.3c
1223 vwarnx.3c
1224 vwprintf.3c
1225 vwscanf.3c
1226 wait4.3c
1227 warn.3c
1228 warnx.3c
1229 watof.3c
1230 watoi.3c
1231 watol.3c
1232 watoll.3c
1233 wrtomb.1.3c
1234 wscat.3c
1235 wscr.3c
1236 wscmp.3c
1237 wscoll.1.3c
1238 wscpy.3c
1239 wscspn.3c
1240 wsetno.3c
1241 wslen.3c
1242 wsnecat.3c
1243 wscnmp.3c
1244 wscncpy.3c
1245 wspbkr.3c
1246 wscrchr.3c

```

```

1247 wcsspn.3c
1248 wcstof.3c
1249 wcstok.3c
1250 wcstold.3c
1251 wcstoll.3c
1252 wcstoull.3c
1253 wcstoumax.3c
1254 wswcs.3c
1255 wswidth.1.3c
1256 wctob.1.3c
1257 wctomb.1.3c
1258 wctrans.1.3c
1259 wctype.1.3c
1260 wcwidth.1.3c
1261 windex.3c
1262 wordfree.3c
1263 wprintf.3c
1264 wrindex.3c
1265 wscanf.3c
1266 wscasecmp.3c
1267 wscat.3c
1268 wscr.3c
1269 wscmp.3c
1270 wscoll.3c
1271 wscoll.3c
1272 wscpy.3c
1273 wscspn.3c
1274 wsdup.3c
1275 wslen.3c
1276 wscasecmp.3c
1277 wsnecat.3c
1278 wscnmp.3c
1279 wscncpy.3c
1280 wspbkr.3c
1281 wscrchr.3c
1282 wcsspn.3c
1283 wstod.3c
1284 wstok.3c
1285 wstol.3c
1286 wstostr.3c
1287 wsxfrm.3c

1289 __flbf.3c := LINKSRC = __fbufsize.3c
1290 __fpending.3c := LINKSRC = __fbufsize.3c
1291 __fpurge.3c := LINKSRC = __fbufsize.3c
1292 __freadable.3c := LINKSRC = __fbufsize.3c
1293 __freading.3c := LINKSRC = __fbufsize.3c
1294 __fsetlocking.3c := LINKSRC = __fbufsize.3c
1295 __fwritable.3c := LINKSRC = __fbufsize.3c
1296 __fwriting.3c := LINKSRC = __fbufsize.3c
1297 __flushlbf.3c := LINKSRC = __fbufsize.3c

1299 _setjmp.3c := LINKSRC = _longjmp.3c

1301 l64a.3c := LINKSRC = a64l.3c

1303 labs.3c := LINKSRC = abs.3c
1304 llabs.3c := LINKSRC = abs.3c

1306 aiowrite.3c := LINKSRC = aioread.3c

1308 atomic_add_16.3c := LINKSRC = atomic_add.3c
1309 atomic_add_16_nv.3c := LINKSRC = atomic_add.3c
1310 atomic_add_32.3c := LINKSRC = atomic_add.3c
1311 atomic_add_32_nv.3c := LINKSRC = atomic_add.3c
1312 atomic_add_64.3c := LINKSRC = atomic_add.3c

```

```

1313 atomic_add_64_nv.3c      := LINKSRC = atomic_add.3c
1314 atomic_add_8_3c         := LINKSRC = atomic_add.3c
1315 atomic_add_8_nv.3c       := LINKSRC = atomic_add.3c
1316 atomic_add_char.3c       := LINKSRC = atomic_add.3c
1317 atomic_add_char_nv.3c    := LINKSRC = atomic_add.3c
1318 atomic_add_int.3c        := LINKSRC = atomic_add.3c
1319 atomic_add_int_nv.3c     := LINKSRC = atomic_add.3c
1320 atomic_add_long.3c       := LINKSRC = atomic_add.3c
1321 atomic_add_long_nv.3c    := LINKSRC = atomic_add.3c
1322 atomic_add_ptr.3c        := LINKSRC = atomic_add.3c
1323 atomic_add_ptr_nv.3c     := LINKSRC = atomic_add.3c
1324 atomic_add_short.3c      := LINKSRC = atomic_add.3c
1325 atomic_add_short_nv.3c   := LINKSRC = atomic_add.3c
1326 atomic_and_16_3c        := LINKSRC = atomic_and.3c
1327 atomic_and_16_nv.3c     := LINKSRC = atomic_and.3c
1328 atomic_and_32_3c        := LINKSRC = atomic_and.3c
1329 atomic_and_32_nv.3c     := LINKSRC = atomic_and.3c
1330 atomic_and_64_3c        := LINKSRC = atomic_and.3c
1331 atomic_and_64_nv.3c     := LINKSRC = atomic_and.3c
1332 atomic_and_8_3c         := LINKSRC = atomic_and.3c
1333 atomic_and_8_nv.3c      := LINKSRC = atomic_and.3c
1334 atomic_and_uchar.3c     := LINKSRC = atomic_and.3c
1335 atomic_and_uchar_nv.3c  := LINKSRC = atomic_and.3c
1336 atomic_and_uint.3c      := LINKSRC = atomic_and.3c
1337 atomic_and_uint_nv.3c   := LINKSRC = atomic_and.3c
1338 atomic_and_ulong.3c     := LINKSRC = atomic_and.3c
1339 atomic_and_ulong_nv.3c  := LINKSRC = atomic_and.3c
1340 atomic_and_ushort.3c    := LINKSRC = atomic_and.3c
1341 atomic_and_ushort_nv.3c := LINKSRC = atomic_and.3c

1343 atomic_clear_long_excl.3c := LINKSRC = atomic_bits.3c
1344 atomic_set_long_excl.3c  := LINKSRC = atomic_bits.3c

1346 atomic_cas_16_3c        := LINKSRC = atomic_cas.3c
1347 atomic_cas_32_3c        := LINKSRC = atomic_cas.3c
1348 atomic_cas_64_3c        := LINKSRC = atomic_cas.3c
1349 atomic_cas_8_3c         := LINKSRC = atomic_cas.3c
1350 atomic_cas_ptr.3c       := LINKSRC = atomic_cas.3c
1351 atomic_cas_uchar.3c     := LINKSRC = atomic_cas.3c
1352 atomic_cas_uint.3c     := LINKSRC = atomic_cas.3c
1353 atomic_cas_ulong.3c     := LINKSRC = atomic_cas.3c
1354 atomic_cas_ushort.3c    := LINKSRC = atomic_cas.3c

1356 atomic_dec_16_3c       := LINKSRC = atomic_dec.3c
1357 atomic_dec_16_nv.3c    := LINKSRC = atomic_dec.3c
1358 atomic_dec_32_3c       := LINKSRC = atomic_dec.3c
1359 atomic_dec_32_nv.3c    := LINKSRC = atomic_dec.3c
1360 atomic_dec_64_3c       := LINKSRC = atomic_dec.3c
1361 atomic_dec_64_nv.3c    := LINKSRC = atomic_dec.3c
1362 atomic_dec_8_3c        := LINKSRC = atomic_dec.3c
1363 atomic_dec_8_nv.3c     := LINKSRC = atomic_dec.3c
1364 atomic_dec_ptr.3c      := LINKSRC = atomic_dec.3c
1365 atomic_dec_ptr_nv.3c   := LINKSRC = atomic_dec.3c
1366 atomic_dec_uchar.3c    := LINKSRC = atomic_dec.3c
1367 atomic_dec_uchar_nv.3c := LINKSRC = atomic_dec.3c
1368 atomic_dec_uint.3c     := LINKSRC = atomic_dec.3c
1369 atomic_dec_uint_nv.3c  := LINKSRC = atomic_dec.3c
1370 atomic_dec_ulong.3c    := LINKSRC = atomic_dec.3c
1371 atomic_dec_ulong_nv.3c := LINKSRC = atomic_dec.3c
1372 atomic_dec_ushort.3c   := LINKSRC = atomic_dec.3c
1373 atomic_dec_ushort_nv.3c := LINKSRC = atomic_dec.3c

1375 atomic_inc_16_3c       := LINKSRC = atomic_inc.3c
1376 atomic_inc_16_nv.3c   := LINKSRC = atomic_inc.3c
1377 atomic_inc_32_3c      := LINKSRC = atomic_inc.3c
1378 atomic_inc_32_nv.3c   := LINKSRC = atomic_inc.3c

```

```

1379 atomic_inc_64_3c       := LINKSRC = atomic_inc.3c
1380 atomic_inc_64_nv.3c    := LINKSRC = atomic_inc.3c
1381 atomic_inc_8_3c        := LINKSRC = atomic_inc.3c
1382 atomic_inc_8_nv.3c    := LINKSRC = atomic_inc.3c
1383 atomic_inc_ptr.3c      := LINKSRC = atomic_inc.3c
1384 atomic_inc_ptr_nv.3c   := LINKSRC = atomic_inc.3c
1385 atomic_inc_uchar.3c   := LINKSRC = atomic_inc.3c
1386 atomic_inc_uchar_nv.3c := LINKSRC = atomic_inc.3c
1387 atomic_inc_uint.3c    := LINKSRC = atomic_inc.3c
1388 atomic_inc_uint_nv.3c  := LINKSRC = atomic_inc.3c
1389 atomic_inc_ulong.3c   := LINKSRC = atomic_inc.3c
1390 atomic_inc_ulong_nv.3c := LINKSRC = atomic_inc.3c
1391 atomic_inc_ushort.3c   := LINKSRC = atomic_inc.3c
1392 atomic_inc_ushort_nv.3c := LINKSRC = atomic_inc.3c

1394 atomic_or_16_3c        := LINKSRC = atomic_or.3c
1395 atomic_or_16_nv.3c     := LINKSRC = atomic_or.3c
1396 atomic_or_32_3c       := LINKSRC = atomic_or.3c
1397 atomic_or_32_nv.3c    := LINKSRC = atomic_or.3c
1398 atomic_or_64_3c       := LINKSRC = atomic_or.3c
1399 atomic_or_64_nv.3c    := LINKSRC = atomic_or.3c
1400 atomic_or_8_3c         := LINKSRC = atomic_or.3c
1401 atomic_or_8_nv.3c     := LINKSRC = atomic_or.3c
1402 atomic_or_uchar.3c    := LINKSRC = atomic_or.3c
1403 atomic_or_uchar_nv.3c := LINKSRC = atomic_or.3c
1404 atomic_or_uint.3c     := LINKSRC = atomic_or.3c
1405 atomic_or_uint_nv.3c  := LINKSRC = atomic_or.3c
1406 atomic_or_ulong.3c   := LINKSRC = atomic_or.3c
1407 atomic_or_ulong_nv.3c := LINKSRC = atomic_or.3c
1408 atomic_or_ushort.3c   := LINKSRC = atomic_or.3c
1409 atomic_or_ushort_nv.3c := LINKSRC = atomic_or.3c

1411 atomic_swap_16_3c      := LINKSRC = atomic_swap.3c
1412 atomic_swap_32_3c    := LINKSRC = atomic_swap.3c
1413 atomic_swap_64_3c    := LINKSRC = atomic_swap.3c
1414 atomic_swap_8_3c     := LINKSRC = atomic_swap.3c
1415 atomic_swap_ptr.3c   := LINKSRC = atomic_swap.3c
1416 atomic_swap_uchar.3c := LINKSRC = atomic_swap.3c
1417 atomic_swap_uint.3c  := LINKSRC = atomic_swap.3c
1418 atomic_swap_ulong.3c := LINKSRC = atomic_swap.3c
1419 atomic_swap_ushort.3c := LINKSRC = atomic_swap.3c

1421 bcmp.3c               := LINKSRC = bstring.3c
1422 bcopy.3c              := LINKSRC = bstring.3c
1423 bzero.3c              := LINKSRC = bstring.3c

1425 btowc_1.3c           := LINKSRC = btowc.3c

1427 catclose.3c         := LINKSRC = catopen.3c

1429 cfgetospeed.3c     := LINKSRC = cfgetispeed.3c

1431 cfsetospeed.3c     := LINKSRC = cfsetispeed.3c

1433 clock_getres.3c     := LINKSRC = clock_settime.3c
1434 clock_gettime.3c   := LINKSRC = clock_settime.3c

1436 fdwalk.3c           := LINKSRC = closefrom.3c

1438 cond_broadcast.3c   := LINKSRC = cond_init.3c
1439 cond_destroy.3c    := LINKSRC = cond_init.3c
1440 cond_reltimedwait.3c := LINKSRC = cond_init.3c
1441 cond_signal.3c     := LINKSRC = cond_init.3c
1442 cond_timedwait.3c := LINKSRC = cond_init.3c
1443 cond_wait.3c       := LINKSRC = cond_init.3c

```



```

1445 csetcol.3c      := LINKSRC = cset.3c
1446 csetlen.3c     := LINKSRC = cset.3c
1447 csetno.3c      := LINKSRC = cset.3c
1448 wcsetno.3c     := LINKSRC = cset.3c

1450 ctermid_r.3c   := LINKSRC = ctermid.3c

1452 asctime.3c     := LINKSRC = ctime.3c
1453 asctime_r.3c   := LINKSRC = ctime.3c
1454 ctime_r.3c     := LINKSRC = ctime.3c
1455 gmtime.3c     := LINKSRC = ctime.3c
1456 gmtime_r.3c   := LINKSRC = ctime.3c
1457 localtime.3c  := LINKSRC = ctime.3c
1458 localtime_r.3c := LINKSRC = ctime.3c
1459 tzset.3c      := LINKSRC = ctime.3c

1461 isalnum.3c     := LINKSRC = ctype.3c
1462 isalnum_1.3c   := LINKSRC = ctype.3c
1463 isalpha.3c     := LINKSRC = ctype.3c
1464 isalpha_1.3c   := LINKSRC = ctype.3c
1465 isascii.3c     := LINKSRC = ctype.3c
1466 isblank.3c    := LINKSRC = ctype.3c
1467 isblank_1.3c  := LINKSRC = ctype.3c
1468 iscntrl.3c    := LINKSRC = ctype.3c
1469 iscntrl_1.3c  := LINKSRC = ctype.3c
1470 isdigit.3c    := LINKSRC = ctype.3c
1471 isdigit_1.3c  := LINKSRC = ctype.3c
1472 isgraph.3c   := LINKSRC = ctype.3c
1473 isgraph_1.3c := LINKSRC = ctype.3c
1474 islower.3c   := LINKSRC = ctype.3c
1475 islower_1.3c := LINKSRC = ctype.3c
1476 isprint.3c   := LINKSRC = ctype.3c
1477 isprint_1.3c := LINKSRC = ctype.3c
1478 ispunct.3c   := LINKSRC = ctype.3c
1479 ispunct_1.3c := LINKSRC = ctype.3c
1480 isspace.3c   := LINKSRC = ctype.3c
1481 isspace_1.3c := LINKSRC = ctype.3c
1482 isupper.3c   := LINKSRC = ctype.3c
1483 isupper_1.3c := LINKSRC = ctype.3c
1484 isxdigit.3c := LINKSRC = ctype.3c
1485 isxdigit_1.3c := LINKSRC = ctype.3c

1487 decimal_to_double.3c := LINKSRC = decimal_to_floating.3c
1488 decimal_to_extended.3c := LINKSRC = decimal_to_floating.3c
1489 decimal_to_quadruple.3c := LINKSRC = decimal_to_floating.3c
1490 decimal_to_single.3c := LINKSRC = decimal_to_floating.3c

1492 ldiv.3c       := LINKSRC = div.3c
1493 lldiv.3c      := LINKSRC = div.3c

1495 dladdr1.3c   := LINKSRC = dladdr.3c

1497 dlmopen.3c  := LINKSRC = dlopen.3c

1499 door_unbind.3c := LINKSRC = door_bind.3c

1501 door_setparam.3c := LINKSRC = door_getparam.3c

1503 erand48.3c   := LINKSRC = drand48.3c
1504 jrand48.3c   := LINKSRC = drand48.3c
1505 lcong48.3c   := LINKSRC = drand48.3c
1506 lrand48.3c   := LINKSRC = drand48.3c
1507 mrand48.3c   := LINKSRC = drand48.3c
1508 nrand48.3c   := LINKSRC = drand48.3c
1509 seed48.3c    := LINKSRC = drand48.3c
1510 srand48.3c   := LINKSRC = drand48.3c

```

```

1512 dup3.3c      := LINKSRC = dup2.3c

1514 fconvert.3c  := LINKSRC = econvert.3c
1515 gconvert.3c  := LINKSRC = econvert.3c
1516 qeconvert.3c := LINKSRC = econvert.3c
1517 qfconvert.3c := LINKSRC = econvert.3c
1518 qgconvert.3c := LINKSRC = econvert.3c
1519 seconvert.3c := LINKSRC = econvert.3c
1520 sfconvert.3c := LINKSRC = econvert.3c
1521 sgconvert.3c := LINKSRC = econvert.3c

1523 fcvt.3c     := LINKSRC = ecvt.3c
1524 gcvt.3c     := LINKSRC = ecvt.3c

1526 _edata.3c   := LINKSRC = end.3c
1527 _end.3c     := LINKSRC = end.3c
1528 _etext.3c   := LINKSRC = end.3c
1529 edata.3c    := LINKSRC = end.3c
1530 etext.3c    := LINKSRC = end.3c

1532 errx.3c     := LINKSRC = err.3c
1533 verr.3c     := LINKSRC = err.3c
1534 verrx.3c    := LINKSRC = err.3c
1535 vwarn.3c    := LINKSRC = err.3c
1536 vwarnx.3c   := LINKSRC = err.3c
1537 warn.3c     := LINKSRC = err.3c
1538 warnx.3c    := LINKSRC = err.3c

1540 euccol.3c   := LINKSRC = euclen.3c
1541 eucscol.3c := LINKSRC = euclen.3c

1543 _exithandle.3c := LINKSRC = exit.3c

1545 clearerr.3c := LINKSRC = ferror.3c
1546 feof.3c     := LINKSRC = ferror.3c
1547 fileno.3c   := LINKSRC = ferror.3c

1549 fsetattr.3c := LINKSRC = fgetattr.3c
1550 getattr.3c  := LINKSRC = fgetattr.3c
1551 setattr.3c  := LINKSRC = fgetattr.3c

1553 getc.3c     := LINKSRC = fgetc.3c
1554 getc_unlocked.3c := LINKSRC = fgetc.3c
1555 getchar.3c := LINKSRC = fgetc.3c
1556 getchar_unlocked.3c := LINKSRC = fgetc.3c
1557 getw.3c     := LINKSRC = fgetc.3c

1559 fgetwc_1.3c := LINKSRC = fgetwc.3c

1561 double_to_decimal.3c := LINKSRC = floating_to_decimal.3c
1562 extended_to_decimal.3c := LINKSRC = floating_to_decimal.3c
1563 quadruple_to_decimal.3c := LINKSRC = floating_to_decimal.3c
1564 single_to_decimal.3c := LINKSRC = floating_to_decimal.3c

1566 ftrylockfile.3c := LINKSRC = flockfile.3c
1567 funlockfile.3c := LINKSRC = flockfile.3c

1569 fpgetmask.3c    := LINKSRC = fpgetround.3c
1570 fpgetsticky.3c := LINKSRC = fpgetround.3c
1571 fpsetmask.3c   := LINKSRC = fpgetround.3c
1572 fpsetround.3c := LINKSRC = fpgetround.3c
1573 fpsetsticky.3c := LINKSRC = fpgetround.3c

1575 putc.3c        := LINKSRC = fputc.3c
1576 putc_unlocked.3c := LINKSRC = fputc.3c

```

```

1577 putchar.3c      := LINKSRC = fputc.3c
1578 putchar_unlocked.3c := LINKSRC = fputc.3c
1579 putw.3c         := LINKSRC = fputc.3c

1581 putwc.3c        := LINKSRC = fputwc.3c
1582 putwchar.3c    := LINKSRC = fputwc.3c

1584 fseeko.3c      := LINKSRC = fseek.3c

1586 ftello.3c     := LINKSRC = ftell.3c

1588 nftw.3c        := LINKSRC = ftw.3c

1590 swprintf.3c    := LINKSRC = fwprintf.3c
1591 wprintf.3c     := LINKSRC = fwprintf.3c

1593 wscanf.3c      := LINKSRC = fwsanf.3c
1594 vfwscanf.3c    := LINKSRC = fwsanf.3c
1595 vswscanf.3c    := LINKSRC = fwsanf.3c
1596 wscanf.3c      := LINKSRC = fwsanf.3c
1597 wscanf.3c      := LINKSRC = fwsanf.3c

1599 gethomegroup.3c := LINKSRC = getcpuid.3c

1601 endgrent.3c     := LINKSRC = getgrnam.3c
1602 fgetgrent.3c   := LINKSRC = getgrnam.3c
1603 fgetgrent_r.3c := LINKSRC = getgrnam.3c
1604 getgrent.3c    := LINKSRC = getgrnam.3c
1605 getgrent_r.3c  := LINKSRC = getgrnam.3c
1606 getgrgid.3c    := LINKSRC = getgrnam.3c
1607 getgrgid_r.3c := LINKSRC = getgrnam.3c
1608 getgrnam_r.3c  := LINKSRC = getgrnam.3c
1609 setgrent.3c    := LINKSRC = getgrnam.3c

1611 sethostname.3c := LINKSRC = gethostname.3c

1613 gethrvtime.3c := LINKSRC = gethrtime.3c

1615 getdelim.3c   := LINKSRC = getline.3c

1617 getlogin_r.3c := LINKSRC = getlogin.3c

1619 getextmntent.3c := LINKSRC = getmntent.3c
1620 getmntany.3c   := LINKSRC = getmntent.3c
1621 hasmntopt.3c  := LINKSRC = getmntent.3c
1622 putmntent.3c  := LINKSRC = getmntent.3c
1623 resetmnttab.3c := LINKSRC = getmntent.3c

1625 endnetgrent.3c := LINKSRC = getnetgrent.3c
1626 getnetgrent_r.3c := LINKSRC = getnetgrent.3c
1627 innetgr.3c     := LINKSRC = getnetgrent.3c
1628 setnetgrent.3c := LINKSRC = getnetgrent.3c

1630 getpassphrase.3c := LINKSRC = getpass.3c

1632 setpriority.3c  := LINKSRC = getpriority.3c

1634 endpwent.3c    := LINKSRC = getpwnam.3c
1635 fgetpwent.3c   := LINKSRC = getpwnam.3c
1636 fgetpwent_r.3c := LINKSRC = getpwnam.3c
1637 getpwent.3c    := LINKSRC = getpwnam.3c
1638 getpwent_r.3c  := LINKSRC = getpwnam.3c
1639 getpwnam_r.3c  := LINKSRC = getpwnam.3c
1640 getpwuid.3c    := LINKSRC = getpwnam.3c
1641 getpwuid_r.3c  := LINKSRC = getpwnam.3c
1642 setpwent.3c    := LINKSRC = getpwnam.3c

```

```

1644 fgets.3c       := LINKSRC = gets.3c

1646 endspent.3c    := LINKSRC = getspnam.3c
1647 fgetsent.3c   := LINKSRC = getspnam.3c
1648 fgetsent_r.3c  := LINKSRC = getspnam.3c
1649 getspent.3c    := LINKSRC = getspnam.3c
1650 getspent_r.3c  := LINKSRC = getspnam.3c
1651 getspnam_r.3c  := LINKSRC = getspnam.3c
1652 setspent.3c    := LINKSRC = getspnam.3c

1654 bind_textdomain_codeset.3c := LINKSRC = gettext.3c
1655 bindtextdomain.3c          := LINKSRC = gettext.3c
1656 dcgettext.3c               := LINKSRC = gettext.3c
1657 dcngettext.3c              := LINKSRC = gettext.3c
1658 dgettext.3c                := LINKSRC = gettext.3c
1659 dngettext.3c               := LINKSRC = gettext.3c
1660 ngettext.3c                := LINKSRC = gettext.3c
1661 textdomain.3c              := LINKSRC = gettext.3c

1663 settimeofday.3c           := LINKSRC = gettimeofday.3c

1665 endusershell.3c         := LINKSRC = getusershell.3c
1666 setusershell.3c        := LINKSRC = getusershell.3c

1668 endutent.3c             := LINKSRC = getutent.3c
1669 getutid.3c              := LINKSRC = getutent.3c
1670 getutline.3c            := LINKSRC = getutent.3c
1671 pututline.3c           := LINKSRC = getutent.3c
1672 setutent.3c            := LINKSRC = getutent.3c
1673 utmpname.3c             := LINKSRC = getutent.3c

1675 endutxent.3c           := LINKSRC = getutxent.3c
1676 getutmp.3c             := LINKSRC = getutxent.3c
1677 getutmpx.3c            := LINKSRC = getutxent.3c
1678 getutxid.3c             := LINKSRC = getutxent.3c
1679 getutxline.3c          := LINKSRC = getutxent.3c
1680 pututxline.3c          := LINKSRC = getutxent.3c
1681 setutxent.3c           := LINKSRC = getutxent.3c
1682 updwtmp.3c             := LINKSRC = getutxent.3c
1683 updwtmpx.3c            := LINKSRC = getutxent.3c
1684 utmpxname.3c           := LINKSRC = getutxent.3c

1686 getvfsany.3c            := LINKSRC = getvfsent.3c
1687 getvfsfile.3c         := LINKSRC = getvfsent.3c
1688 getvfsspec.3c         := LINKSRC = getvfsent.3c

1690 getwc_1.3c           := LINKSRC = getwc.3c
1692 getwchar_1.3c       := LINKSRC = getwchar.3c

1694 fgetws.3c              := LINKSRC = getws.3c

1696 getzoneidbyname.3c     := LINKSRC = getzoneid.3c
1697 getzonenamebyid.3c    := LINKSRC = getzoneid.3c

1699 globfree.3c            := LINKSRC = glob.3c

1701 hcreate.3c              := LINKSRC = hsearch.3c
1702 hdestroy.3c            := LINKSRC = hsearch.3c

1704 rindex.3c              := LINKSRC = index.3c

1706 remque.3c              := LINKSRC = insque.3c

1708 finite.3c              := LINKSRC = isnand.3c

```

```

1709 fpclass.3c      := LINKSRC = isnand.3c
1710 isnanf.3c       := LINKSRC = isnand.3c
1711 unordered.3c    := LINKSRC = isnand.3c

1713 isenglish.3c    := LINKSRC = iswalpha.3c
1714 isideogram.3c  := LINKSRC = iswalpha.3c
1715 isnumber.3c     := LINKSRC = iswalpha.3c
1716 isphonogram.3c := LINKSRC = iswalpha.3c
1717 isspecial.3c   := LINKSRC = iswalpha.3c
1718 iswalnum.3c     := LINKSRC = iswalpha.3c
1719 iswalnum_1.3c  := LINKSRC = iswalpha.3c
1720 iswalpha_1.3c  := LINKSRC = iswalpha.3c
1721 iswascii.3c    := LINKSRC = iswalpha.3c
1722 iswblank.3c    := LINKSRC = iswalpha.3c
1723 iswblank_1.3c  := LINKSRC = iswalpha.3c
1724 iswcntrl.3c   := LINKSRC = iswalpha.3c
1725 iswcntrl_1.3c := LINKSRC = iswalpha.3c
1726 iswdigit.3c   := LINKSRC = iswalpha.3c
1727 iswdigit_1.3c := LINKSRC = iswalpha.3c
1728 iswgraph.3c    := LINKSRC = iswalpha.3c
1729 iswgraph_1.3c := LINKSRC = iswalpha.3c
1730 iswhexnumber.3c := LINKSRC = iswalpha.3c
1731 iswhexnumber_1.3c := LINKSRC = iswalpha.3c
1732 iswideogram.3c := LINKSRC = iswalpha.3c
1733 iswideogram_1.3c := LINKSRC = iswalpha.3c
1734 iswlower.3c   := LINKSRC = iswalpha.3c
1735 iswlower_1.3c := LINKSRC = iswalpha.3c
1736 iswnumber.3c  := LINKSRC = iswalpha.3c
1737 iswnumber_1.3c := LINKSRC = iswalpha.3c
1738 iswphonogram.3c := LINKSRC = iswalpha.3c
1739 iswphonogram_1.3c := LINKSRC = iswalpha.3c
1740 iswprint.3c    := LINKSRC = iswalpha.3c
1741 iswprint_1.3c := LINKSRC = iswalpha.3c
1742 iswpunct.3c   := LINKSRC = iswalpha.3c
1743 iswpunct_1.3c := LINKSRC = iswalpha.3c
1744 iswspace.3c   := LINKSRC = iswalpha.3c
1745 iswspace_1.3c := LINKSRC = iswalpha.3c
1746 iswspecial.3c := LINKSRC = iswalpha.3c
1747 iswspecial_1.3c := LINKSRC = iswalpha.3c
1748 iswupper.3c   := LINKSRC = iswalpha.3c
1749 iswupper_1.3c := LINKSRC = iswalpha.3c
1750 iswxdigit.3c  := LINKSRC = iswalpha.3c
1751 iswxdigit_1.3c := LINKSRC = iswalpha.3c

1753 iswctype_1.3c := LINKSRC = iswctype.3c

1755 ulckpddf.3c   := LINKSRC = lckpddf.3c

1757 lfind.3c      := LINKSRC = lsearch.3c

1759 swapcontext.3c := LINKSRC = makecontext.3c

1761 major.3c     := LINKSRC = makedev.3c
1762 minor.3c     := LINKSRC = makedev.3c

1764 alloca.3c    := LINKSRC = malloc.3c
1765 calloc.3c    := LINKSRC = malloc.3c
1766 free.3c      := LINKSRC = malloc.3c
1767 memalign.3c  := LINKSRC = malloc.3c
1768 realloc.3c   := LINKSRC = malloc.3c
1769 valloc.3c    := LINKSRC = malloc.3c

1771 mblen_1.3c   := LINKSRC = mblen.3c

1773 mbrlen_1.3c  := LINKSRC = mbrlen.3c

```

```

1775 mbrtowc_1.3c := LINKSRC = mbrtowc.3c

1777 mbsinit_1.3c := LINKSRC = mbsinit.3c

1779 mbsnrtowcs.3c := LINKSRC = mbsrtowcs.3c
1780 mbsnrtowcs_1.3c := LINKSRC = mbsrtowcs.3c
1781 mbsrtowcs_1.3c := LINKSRC = mbsrtowcs.3c
1782 mbstowcs.3c    := LINKSRC = mbsrtowcs.3c
1783 mbstowcs_1.3c := LINKSRC = mbsrtowcs.3c

1785 mbtowc_1.3c   := LINKSRC = mbtowc.3c

1787 membar_consumer.3c := LINKSRC = membar_ops.3c
1788 membar_enter.3c   := LINKSRC = membar_ops.3c
1789 membar_exit.3c   := LINKSRC = membar_ops.3c
1790 membar_producer.3c := LINKSRC = membar_ops.3c

1792 memccpy.3c       := LINKSRC = memory.3c
1793 memchr.3c        := LINKSRC = memory.3c
1794 memcmp.3c        := LINKSRC = memory.3c
1795 memcpy.3c       := LINKSRC = memory.3c
1796 memmove.3c      := LINKSRC = memory.3c
1797 memset.3c       := LINKSRC = memory.3c

1799 mkdtemp.3c      := LINKSRC = mkstemp.3c
1800 mkostemp.3c     := LINKSRC = mkstemp.3c
1801 mkostemps.3c   := LINKSRC = mkstemp.3c
1802 mkstemp.3c     := LINKSRC = mkstemp.3c

1804 munlock.3c     := LINKSRC = mlock.3c

1806 munlockall.3c := LINKSRC = mlockall.3c

1808 mq_reltimedreceive_np.3c := LINKSRC = mq_receive.3c
1809 mq_timedreceive.3c      := LINKSRC = mq_receive.3c

1811 mq_reltimedsend_np.3c   := LINKSRC = mq_send.3c
1812 mq_timedsend.3c       := LINKSRC = mq_send.3c

1814 mutex_consistent.3c    := LINKSRC = mutex_init.3c
1815 mutex_destroy.3c      := LINKSRC = mutex_init.3c
1816 mutex_lock.3c        := LINKSRC = mutex_init.3c
1817 mutex_trylock.3c     := LINKSRC = mutex_init.3c
1818 mutex_unlock.3c      := LINKSRC = mutex_init.3c

1820 dbm_clearerr.3c        := LINKSRC = ndbm.3c
1821 dbm_close.3c         := LINKSRC = ndbm.3c
1822 dbm_delete.3c        := LINKSRC = ndbm.3c
1823 dbm_error.3c         := LINKSRC = ndbm.3c
1824 dbm_fetch.3c        := LINKSRC = ndbm.3c
1825 dbm_firstkey.3c     := LINKSRC = ndbm.3c
1826 dbm_nextkey.3c     := LINKSRC = ndbm.3c
1827 dbm_open.3c        := LINKSRC = ndbm.3c
1828 dbm_store.3c       := LINKSRC = ndbm.3c

1830 duplocale.3c         := LINKSRC = newlocale.3c
1831 freelocale.3c       := LINKSRC = newlocale.3c

1833 nl_langinfo_1.3c    := LINKSRC = nl_langinfo.3c

1835 fdopendir.3c       := LINKSRC = opendir.3c

1837 errno.3c           := LINKSRC = perror.3c

1839 pclose.3c          := LINKSRC = popen.3c

```

```

1841 port_dissociate.3c      := LINKSRC = port_associate.3c
1843 port_getn.3c           := LINKSRC = port_get.3c
1845 port_sendn.3c         := LINKSRC = port_send.3c
1847 posix_spawnnp.3c      := LINKSRC = posix_spawn.3c
1849 posix_spawn_file_actions_addopen.3c := LINKSRC = posix_spawn_file_actions_ad
1850 posix_spawn_file_actions_init.3c   := LINKSRC = posix_spawn_file_actions_de
1852 posix_spawnattr_init.3c          := LINKSRC = posix_spawnattr_destroy.3c
1854 posix_spawnattr_setflags.3c      := LINKSRC = posix_spawnattr_getflags.3c
1856 posix_spawnattr_setpgroup.3c     := LINKSRC = posix_spawnattr_getpgroup.3
1858 posix_spawnattr_setschedparam.3c := LINKSRC = posix_spawnattr_getschedpar
1860 posix_spawnattr_setschedpolicy.3c := LINKSRC = posix_spawnattr_getschedpol
1862 posix_spawnattr_setsigdefault.3c := LINKSRC = posix_spawnattr_getsigdefau
1864 posix_spawnattr_setsigignore_np.3c := LINKSRC = posix_spawnattr_getsigignor
1866 posix_spawnattr_setsigmask.3c   := LINKSRC = posix_spawnattr_getsigmask.
1868 asprintf.3c                 := LINKSRC = printf.3c
1869 fprintf.3c                  := LINKSRC = printf.3c
1870 snprintf.3c                 := LINKSRC = printf.3c
1871 sprintf.3c                   := LINKSRC = printf.3c
1873 priv_allocset.3c            := LINKSRC = priv_addset.3c
1874 priv_basicset.3c            := LINKSRC = priv_addset.3c
1875 priv_copyset.3c            := LINKSRC = priv_addset.3c
1876 priv_delset.3c             := LINKSRC = priv_addset.3c
1877 priv_emptyset.3c           := LINKSRC = priv_addset.3c
1878 priv_fillset.3c            := LINKSRC = priv_addset.3c
1879 priv_freerset.3c           := LINKSRC = priv_addset.3c
1880 priv_intersect.3c          := LINKSRC = priv_addset.3c
1881 priv_inverse.3c            := LINKSRC = priv_addset.3c
1882 priv_ismptyset.3c          := LINKSRC = priv_addset.3c
1883 priv_isequalset.3c         := LINKSRC = priv_addset.3c
1884 priv_isfullset.3c          := LINKSRC = priv_addset.3c
1885 priv_ismember.3c           := LINKSRC = priv_addset.3c
1886 priv_issubset.3c           := LINKSRC = priv_addset.3c
1887 priv_union.3c              := LINKSRC = priv_addset.3c
1889 priv_ineffect.3c           := LINKSRC = priv_set.3c
1891 priv_getbyname.3c          := LINKSRC = priv_str_to_set.3c
1892 priv_getbynum.3c           := LINKSRC = priv_str_to_set.3c
1893 priv_getsetbyname.3c       := LINKSRC = priv_str_to_set.3c
1894 priv_getsetbynum.3c        := LINKSRC = priv_str_to_set.3c
1895 priv_gettext.3c           := LINKSRC = priv_str_to_set.3c
1896 priv_set_to_str.3c         := LINKSRC = priv_str_to_set.3c
1898 psiginfo.3c                := LINKSRC = psignal.3c
1900 pthread_attr_setdetachstate.3c := LINKSRC = pthread_attr_getdetachstate
1902 pthread_attr_setguardsize.3c := LINKSRC = pthread_attr_getguardsize.3
1904 pthread_attr_setinheritsched.3c := LINKSRC = pthread_attr_getinheritsche
1906 pthread_attr_setschedparam.3c := LINKSRC = pthread_attr_getschedparam.

```

```

1908 pthread_attr_setschedpolicy.3c := LINKSRC = pthread_attr_getschedpolicy
1910 pthread_attr_setscope.3c        := LINKSRC = pthread_attr_getscope.3c
1911 pthread_attr_setstack.3c        := LINKSRC = pthread_attr_getstack.3c
1913 pthread_attr_setstackaddr.3c    := LINKSRC = pthread_attr_getstackaddr.3
1915 pthread_attr_setstacksize.3c    := LINKSRC = pthread_attr_getstacksize.3
1917 pthread_attr_destroy.3c         := LINKSRC = pthread_attr_init.3c
1919 pthread_barrier_init.3c         := LINKSRC = pthread_barrier_destroy.3c
1921 pthread_barrierattr_init.3c     := LINKSRC = pthread_barrierattr_destroy
1923 pthread_barrierattr_setpshared.3c := LINKSRC = pthread_barrierattr_getpsha
1925 pthread_cond_destroy.3c         := LINKSRC = pthread_cond_init.3c
1927 pthread_cond_broadcast.3c       := LINKSRC = pthread_cond_signal.3c
1929 pthread_cond_reltimedwait_np.3c := LINKSRC = pthread_cond_wait.3c
1930 pthread_cond_timedwait.3c       := LINKSRC = pthread_cond_wait.3c
1932 pthread_condattr_setclock.3c    := LINKSRC = pthread_condattr_getclock.3
1934 pthread_condattr_setpshared.3c  := LINKSRC = pthread_condattr_getpshared
1936 pthread_condattr_destroy.3c     := LINKSRC = pthread_condattr_init.3c
1938 pthread_setconcurrency.3c       := LINKSRC = pthread_getconcurrency.3c
1940 pthread_setschedparam.3c        := LINKSRC = pthread_getschedparam.3c
1942 pthread_setspecific.3c          := LINKSRC = pthread_getspecific.3c
1944 pthread_key_create_once_np.3c   := LINKSRC = pthread_key_create.3c
1946 pthread_mutex_setprioceiling.3c := LINKSRC = pthread_mutex_getprioceilin
1948 pthread_mutex_destroy.3c        := LINKSRC = pthread_mutex_init.3c
1950 pthread_mutex_trylock.3c         := LINKSRC = pthread_mutex_lock.3c
1951 pthread_mutex_unlock.3c         := LINKSRC = pthread_mutex_lock.3c
1953 pthread_mutex_reltimedlock_np.3c := LINKSRC = pthread_mutex_timedlock.3c
1955 pthread_mutexattr_setprioceiling.3c := LINKSRC = pthread_mutexattr_getprioce
1957 pthread_mutexattr_setprotocol.3c := LINKSRC = pthread_mutexattr_getprotoc
1959 pthread_mutexattr_setpshared.3c := LINKSRC = pthread_mutexattr_getpshare
1961 pthread_mutexattr_settype.3c    := LINKSRC = pthread_mutexattr_gettype.3
1963 pthread_mutexattr_destroy.3c    := LINKSRC = pthread_mutexattr_init.3c
1965 pthread_rwlock_destroy.3c       := LINKSRC = pthread_rwlock_init.3c
1967 pthread_rwlock_tryrdlock.3c    := LINKSRC = pthread_rwlock_rdlock.3c
1969 pthread_rwlock_reltimedrdlock_np.3c := LINKSRC = pthread_rwlock_timedrdlock.
1971 pthread_rwlock_reltimedwrlock_np.3c := LINKSRC = pthread_rwlock_timedwrlock.

```

```

1973 pthread_rwlock_trywrlock.3c      := LINKSRC = pthread_rwlock_wrlock.3c
1975 pthread_rwlockattr_setpshared.3c := LINKSRC = pthread_rwlockattr_getpshar
1977 pthread_rwlockattr_destroy.3c     := LINKSRC = pthread_rwlockattr_init.3c
1979 pthread_spin_init.3c              := LINKSRC = pthread_spin_destroy.3c
1981 pthread_spin_trylock.3c           := LINKSRC = pthread_spin_lock.3c
1983 fputs.3c                         := LINKSRC = puts.3c
1985 rand_r.3c                         := LINKSRC = rand.3c
1986 srand.3c                         := LINKSRC = rand.3c
1988 initstate.3c                     := LINKSRC = random.3c
1989 setstate.3c                      := LINKSRC = random.3c
1990 srandom.3c                       := LINKSRC = random.3c
1992 rctlblk_get_enforced_value.3c      := LINKSRC = rctlblk_set_value.3c
1993 rctlblk_get_firing_time.3c        := LINKSRC = rctlblk_set_value.3c
1994 rctlblk_get_global_action.3c       := LINKSRC = rctlblk_set_value.3c
1995 rctlblk_get_global_flags.3c       := LINKSRC = rctlblk_set_value.3c
1996 rctlblk_get_local_action.3c       := LINKSRC = rctlblk_set_value.3c
1997 rctlblk_get_local_flags.3c        := LINKSRC = rctlblk_set_value.3c
1998 rctlblk_get_privilege.3c          := LINKSRC = rctlblk_set_value.3c
1999 rctlblk_get_recipient_pid.3c      := LINKSRC = rctlblk_set_value.3c
2000 rctlblk_get_value.3c              := LINKSRC = rctlblk_set_value.3c
2001 rctlblk_set_local_action.3c       := LINKSRC = rctlblk_set_value.3c
2002 rctlblk_set_local_flags.3c        := LINKSRC = rctlblk_set_value.3c
2003 rctlblk_set_privilege.3c          := LINKSRC = rctlblk_set_value.3c
2004 rctlblk_set_recipient_pid.3c      := LINKSRC = rctlblk_set_value.3c
2005 rctlblk_size.3c                  := LINKSRC = rctlblk_set_value.3c
2007 re_exec.3c                      := LINKSRC = re_comp.3c
2009 readdir_r.3c                    := LINKSRC = readdir.3c
2011 regex.3c                        := LINKSRC = regcmp.3c
2013 regerror.3c                     := LINKSRC = regcomp.3c
2014 regexec.3c                      := LINKSRC = regcomp.3c
2015 regfree.3c                      := LINKSRC = regcomp.3c
2017 rw_rdlock.3c                   := LINKSRC = rwlock.3c
2018 rw_tryrdlock.3c                 := LINKSRC = rwlock.3c
2019 rw_trywrlock.3c                 := LINKSRC = rwlock.3c
2020 rw_unlock.3c                    := LINKSRC = rwlock.3c
2021 rw_wrlock.3c                    := LINKSRC = rwlock.3c
2022 rwlock_destroy.3c               := LINKSRC = rwlock.3c
2023 rwlock_init.3c                  := LINKSRC = rwlock.3c
2025 alphasort.3c                   := LINKSRC = scandir.3c
2027 fscanf.3c                       := LINKSRC = scanf.3c
2028 sscanf.3c                       := LINKSRC = scanf.3c
2029 vfscanf.3c                      := LINKSRC = scanf.3c
2030 vscanf.3c                       := LINKSRC = scanf.3c
2031 vsscanf.3c                      := LINKSRC = scanf.3c
2033 sched_get_priority_min.3c        := LINKSRC = sched_get_priority_max.3c
2035 schedctl_exit.3c                 := LINKSRC = schedctl_init.3c
2036 schedctl_lookup.3c               := LINKSRC = schedctl_init.3c
2037 schedctl_start.3c                := LINKSRC = schedctl_init.3c
2038 schedctl_stop.3c                 := LINKSRC = schedctl_init.3c

```

```

2040 FD_CLR.3c                        := LINKSRC = select.3c
2041 FD_ISSET.3c                     := LINKSRC = select.3c
2042 FD_SET.3c                       := LINKSRC = select.3c
2043 FD_ZERO.3c                      := LINKSRC = select.3c
2044 pselect.3c                      := LINKSRC = select.3c
2046 sem_reltimedwait_np.3c           := LINKSRC = sem_timedwait.3c
2048 sem_trywait.3c                  := LINKSRC = sem_wait.3c
2050 sema_destroy.3c                 := LINKSRC = semaphore.3c
2051 sema_init.3c                   := LINKSRC = semaphore.3c
2052 sema_post.3c                   := LINKSRC = semaphore.3c
2053 sema_trywait.3c                := LINKSRC = semaphore.3c
2054 sema_wait.3c                   := LINKSRC = semaphore.3c
2056 setvbuf.3c                     := LINKSRC = setbuf.3c
2058 setlinebuf.3c                  := LINKSRC = setbuffer.3c
2060 longjmp.3c                      := LINKSRC = setjmp.3c
2061 siglongjmp.3c                  := LINKSRC = setjmp.3c
2062 sigsetjmp.3c                   := LINKSRC = setjmp.3c
2064 sighold.3c                     := LINKSRC = signal.3c
2065 sigignore.3c                   := LINKSRC = signal.3c
2066 sigpause.3c                    := LINKSRC = signal.3c
2067 sigrelse.3c                    := LINKSRC = signal.3c
2068 sigset.3c                      := LINKSRC = signal.3c
2070 sigaddset.3c                   := LINKSRC = sigsetops.3c
2071 sigdelset.3c                   := LINKSRC = sigsetops.3c
2072 sigemptyset.3c                 := LINKSRC = sigsetops.3c
2073 sigfillset.3c                   := LINKSRC = sigsetops.3c
2074 sigismember.3c                  := LINKSRC = sigsetops.3c
2076 sigtimedwait.3c                := LINKSRC = sigwaitinfo.3c
2078 gsignal.3c                     := LINKSRC = signal.3c
2080 stderr.3c                      := LINKSRC = stdio.3c
2081 stdin.3c                       := LINKSRC = stdio.3c
2082 stdout.3c                      := LINKSRC = stdio.3c
2084 sig2str.3c                     := LINKSRC = str2sig.3c
2086 strcoll.1.3c                   := LINKSRC = strcoll.3c
2088 strerror_r.3c                   := LINKSRC = strerror.3c
2090 strfmon.1.3c                   := LINKSRC = strfmon.3c
2092 ascftime.3c                     := LINKSRC = strftime.3c
2093 cftime.3c                       := LINKSRC = strftime.3c
2094 strftime.1.3c                   := LINKSRC = strftime.3c
2096 strcasecmp.3c                   := LINKSRC = string.3c
2097 strcascmp.1.3c                   := LINKSRC = string.3c
2098 strcat.3c                       := LINKSRC = string.3c
2099 strchr.3c                       := LINKSRC = string.3c
2100 strcmp.3c                       := LINKSRC = string.3c
2101 strcpy.3c                       := LINKSRC = string.3c
2102 strcspn.3c                      := LINKSRC = string.3c
2103 strdup.3c                       := LINKSRC = string.3c
2104 strlcat.3c                      := LINKSRC = string.3c

```

```

2105 strcpy.3c      := LINKSRC = string.3c
2106 strlen.3c     := LINKSRC = string.3c
2107 strncasecmp.3c := LINKSRC = string.3c
2108 strncasecmp_1.3c := LINKSRC = string.3c
2109 strncat.3c     := LINKSRC = string.3c
2110 strncmp.3c    := LINKSRC = string.3c
2111 strncpy.3c   := LINKSRC = string.3c
2112 strnlen.3c   := LINKSRC = string.3c
2113 strpbrk.3c   := LINKSRC = string.3c
2114 strrchr.3c   := LINKSRC = string.3c
2115 strsep.3c    := LINKSRC = string.3c
2116 strspn.3c   := LINKSRC = string.3c
2117 strstr.3c   := LINKSRC = string.3c
2118 strtok.3c   := LINKSRC = string.3c
2119 strtok_r.3c  := LINKSRC = string.3c

2121 file_to_decimal.3c := LINKSRC = string_to_decimal.3c
2122 func_to_decimal.3c := LINKSRC = string_to_decimal.3c

2124 strptime_1.3c      := LINKSRC = strptime.3c

2126 atof.3c          := LINKSRC = strtod.3c
2127 strtof.3c        := LINKSRC = strtod.3c
2128 strtold.3c       := LINKSRC = strtod.3c

2130 strtoumax.3c    := LINKSRC = strtoumax.3c

2132 atoi.3c         := LINKSRC = strtol.3c
2133 atol.3c         := LINKSRC = strtol.3c
2134 atoll.3c        := LINKSRC = strtol.3c
2135 lltostr.3c      := LINKSRC = strtol.3c
2136 strtoll.3c     := LINKSRC = strtol.3c
2137 ulltostr.3c    := LINKSRC = strtol.3c

2139 strtoull.3c    := LINKSRC = strtoul.3c

2141 wstostr.3c     := LINKSRC = strtows.3c

2143 strxfrm_1.3c      := LINKSRC = strxfrm.3c

2145 closelog.3c   := LINKSRC = syslog.3c
2146 openlog.3c   := LINKSRC = syslog.3c
2147 setlogmask.3c := LINKSRC = syslog.3c

2149 thr_setconcurrency.3c := LINKSRC = thr_getconcurrency.3c

2151 thr_setprio.3c := LINKSRC = thr_getprio.3c

2153 thr_getspecific.3c := LINKSRC = thr_keycreate.3c
2154 thr_keycreate_once.3c := LINKSRC = thr_keycreate.3c
2155 thr_setspecific.3c := LINKSRC = thr_keycreate.3c

2157 thr_continue.3c := LINKSRC = thr_suspend.3c

2159 timer_getoverrun.3c := LINKSRC = timer_settime.3c
2160 timer_gettime.3c   := LINKSRC = timer_settime.3c

2162 timerclear.3c     := LINKSRC = timeradd.3c
2163 timercmp.3c      := LINKSRC = timeradd.3c
2164 timerisset.3c   := LINKSRC = timeradd.3c
2165 timersub.3c     := LINKSRC = timeradd.3c

2167 tmpnam.3c        := LINKSRC = tmpnam.3c
2168 tmpnam_r.3c     := LINKSRC = tmpnam.3c

2170 tolower_1.3c     := LINKSRC = tolower.3c

```

```

2172 toupper_1.3c     := LINKSRC = toupper.3c

2174 tolower_1.3c    := LINKSRC = tolower.3c

2176 toupper_1.3c    := LINKSRC = toupper.3c

2178 ftruncate.3c    := LINKSRC = truncate.3c

2180 tdelete.3c     := LINKSRC = tsearch.3c
2181 tfind.3c      := LINKSRC = tsearch.3c
2182 twalk.3c      := LINKSRC = tsearch.3c

2184 ttyname_r.3c   := LINKSRC = ttyname.3c

2186 uconv_u16tou8.3c := LINKSRC = uconv_u16tou32.3c
2187 uconv_u32tou16.3c := LINKSRC = uconv_u16tou32.3c
2188 uconv_u32tou8.3c := LINKSRC = uconv_u16tou32.3c
2189 uconv_u8tou16.3c := LINKSRC = uconv_u16tou32.3c
2190 uconv_u8tou32.3c := LINKSRC = uconv_u16tou32.3c

2192 ucred_free.3c   := LINKSRC = ucred_get.3c
2193 ucred_getegid.3c := LINKSRC = ucred_get.3c
2194 ucred_geteuid.3c := LINKSRC = ucred_get.3c
2195 ucred_getgroups.3c := LINKSRC = ucred_get.3c
2196 ucred_getlabel.3c := LINKSRC = ucred_get.3c
2197 ucred_getpflags.3c := LINKSRC = ucred_get.3c
2198 ucred_getpid.3c := LINKSRC = ucred_get.3c
2199 ucred_getprivset.3c := LINKSRC = ucred_get.3c
2200 ucred_getprojid.3c := LINKSRC = ucred_get.3c
2201 ucred_getrgid.3c := LINKSRC = ucred_get.3c
2202 ucred_getruid.3c := LINKSRC = ucred_get.3c
2203 ucred_getsgid.3c := LINKSRC = ucred_get.3c
2204 ucred_getsuid.3c := LINKSRC = ucred_get.3c
2205 ucred_getzoneid.3c := LINKSRC = ucred_get.3c
2206 ucred_size.3c   := LINKSRC = ucred_get.3c

2208 vswprintf.3c   := LINKSRC = vfwprintf.3c
2209 vwprintf.3c   := LINKSRC = vfwprintf.3c

2211 vasprintf.3c  := LINKSRC = vprintf.3c
2212 vfprintf.3c   := LINKSRC = vprintf.3c
2213 vsnprintf.3c := LINKSRC = vprintf.3c
2214 vsprintf.3c   := LINKSRC = vprintf.3c

2216 wait4.3c     := LINKSRC = wait3.3c

2218 addrtosymstr.3c := LINKSRC = walkcontext.3c
2219 backtrace.3c   := LINKSRC = walkcontext.3c
2220 backtrace_symbols.3c := LINKSRC = walkcontext.3c
2221 backtrace_symbols_fd.3c := LINKSRC = walkcontext.3c
2222 printstack.3c := LINKSRC = walkcontext.3c

2224 wrtomb_1.3c     := LINKSRC = wrtomb.3c

2226 wscoll_1.3c    := LINKSRC = wscoll.3c
2227 wscoll.3c     := LINKSRC = wscoll.3c

2229 watof.3c      := LINKSRC = wctod.3c
2230 wctof.3c     := LINKSRC = wctod.3c
2231 wcstold.3c   := LINKSRC = wctod.3c
2232 wctod.3c    := LINKSRC = wctod.3c

2234 wcstoumax.3c := LINKSRC = wcstoumax.3c

2236 watoi.3c    := LINKSRC = wctol.3c

```

```

2237 watol.3c      := LINKSRC = wctol.3c
2238 watoll.3c     := LINKSRC = wctol.3c
2239 wctoll.3c      := LINKSRC = wctol.3c
2240 wstol.3c       := LINKSRC = wctol.3c

2242 wcstoull.3c   := LINKSRC = wcstoul.3c

2244 wscat.3c       := LINKSRC = wcstring.3c
2245 wscchr.3c     := LINKSRC = wcstring.3c
2246 wscmp.3c       := LINKSRC = wcstring.3c
2247 wscopy.3c     := LINKSRC = wcstring.3c
2248 wscspn.3c     := LINKSRC = wcstring.3c
2249 wslen.3c       := LINKSRC = wcstring.3c
2250 wscncat.3c    := LINKSRC = wcstring.3c
2251 wscncmp.3c    := LINKSRC = wcstring.3c
2252 wscncpy.3c    := LINKSRC = wcstring.3c
2253 wscpbrk.3c    := LINKSRC = wcstring.3c
2254 wscrchr.3c    := LINKSRC = wcstring.3c
2255 wcsspn.3c     := LINKSRC = wcstring.3c
2256 wctok.3c      := LINKSRC = wcstring.3c
2257 wswcs.3c      := LINKSRC = wcstring.3c
2258 windex.3c     := LINKSRC = wcstring.3c
2259 wrindex.3c    := LINKSRC = wcstring.3c
2260 wscat.3c       := LINKSRC = wcstring.3c
2261 wschr.3c       := LINKSRC = wcstring.3c
2262 wscmp.3c       := LINKSRC = wcstring.3c
2263 wscopy.3c      := LINKSRC = wcstring.3c
2264 wscspn.3c     := LINKSRC = wcstring.3c
2265 wslen.3c      := LINKSRC = wcstring.3c
2266 wscat.3c       := LINKSRC = wcstring.3c
2267 wscncmp.3c    := LINKSRC = wcstring.3c
2268 wscncpy.3c    := LINKSRC = wcstring.3c
2269 wscpbrk.3c    := LINKSRC = wcstring.3c
2270 wscrchr.3c    := LINKSRC = wcstring.3c
2271 wssp.3c       := LINKSRC = wcstring.3c
2272 wstok.3c      := LINKSRC = wcstring.3c
2273 wscasecmp.3c   := LINKSRC = wstring.3c
2274 wscol.3c      := LINKSRC = wstring.3c
2275 wsdup.3c      := LINKSRC = wstring.3c
2276 wscasecmp.3c := LINKSRC = wstring.3c

2278 wscwidth_1.3c := LINKSRC = wscwidth.3c

2280 wsxfrm.3c     := LINKSRC = wsxfrm.3c

2282 wctob_1.3c    := LINKSRC = wctob.3c

2284 wctomb_1.3c   := LINKSRC = wctomb.3c

2286 towctrans.3c  := LINKSRC = wctrans.3c
2287 towctrans_1.3c := LINKSRC = wctrans.3c
2288 wctrans_1.3c  := LINKSRC = wctrans.3c

2290 wctype_1.3c   := LINKSRC = wctype.3c

2292 wcwidth_1.3c := LINKSRC = wcwidth.3c

2294 wordfree.3c  := LINKSRC = wordexp.3c

2110 wscasecmp.3c := LINKSRC = wstring.3c
2111 wscol.3c      := LINKSRC = wstring.3c
2112 wsdup.3c      := LINKSRC = wstring.3c
2113 wscasecmp.3c := LINKSRC = wstring.3c

2296 .KEEP_STATE:

```

```

2298 include      $(SRC)/man/Makefile.man
2300 install:      $(ROOTMANFILES) $(ROOTMANLINKS)

```

\*\*\*\*\*

3683 Fri Jul 11 11:43:07 2014

new/usr/src/man/man3c/btowc.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions Co
3  .\" Copyright (c) 2014, Joyent, Inc. All rights reserved.
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH BTOWC 3C \"Jun 27, 2014\"
12 .TH BTOWC 3C \"Jul 24, 2002\"
13 .SH NAME
14 btowc, btowc_l \- single-byte to wide-character conversion
15 btowc \- single-byte to wide-character conversion
16 .SH SYNOPSIS
17 .LP
18 .nf
19 #include <stdio.h>
20 #include <wchar.h>
21
22 \fBwint_t\fR \fBbtowc\fR(\fBint\fR \fBic\fR);
23 .fi
24 .LP
25 .nf
26 #include <stdio.h>
27 #include <wchar.h>
28 #include <xlocale.h>
29
30 \fBwint_t\fR \fBbtowc_l\fR(\fBint\fR \fBic\fR, \fBlocale_t\fR \fBiloc\fR);
31 .fi
32 .SH DESCRIPTION
33 .sp
34 .LP
35 The \fBbtowc()\fR and \fBbtowc_l()\fR functions determines whether
36 \fBic\fR constitutes a valid (one-byte) character in the initial shift
37 state.
38 The \fBbtowc()\fR function determines whether \fBic\fR constitutes a valid
39 (one-byte) character in the initial shift state.
40 .sp
41 .LP
42 The behavior of \fBbtowc()\fR is affected by the \fBLC_CTYPE\fR category of the
43 current locale. See \fBenviron\fR(5). The function \fBbtowc_l()\fR does
44 not use the current locale, and instead operates on the locale
45 specified by \fBiloc\fR.
46 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
47 current locale. See \fBenviron\fR(5).
48 .SH RETURN VALUES
49 .sp
50 .LP
51 The \fBbtowc()\fR function returns \fBWEOF\fR if \fBic\fR has the value
52 \fBWEOF\fR or if \fB(unsigned char)\fR \fBic\fR does not constitute a valid
53 (one-byte) character in the initial shift state. Otherwise, it returns the
54 wide-character representation of that character.
55 .SH ERRORS
56 .sp
57 .LP
58 No errors are defined.

```

```

49 .SH ATTRIBUTES
50 .sp
51 .LP
52 See \fBAttributes\fR(5) for descriptions of the following attributes:
53 .sp
54 .sp
55 .TS
56 ATTRIBUTE TYPE    ATTRIBUTE VALUE
57 -
58 Interface Stability    See below.
59 Interface Stability    Standard
60 -
61 MT-Level              MT-Safe
62 MT-Level              MT-Safe with exceptions
63 .TE
64 .LP
65 The
66 .B btowc()
67 function is Standard. The
68 .B btowc_l()
69 function is Uncommitted.
70 .SH SEE ALSO
71 .sp
72 .LP
73 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C), \fBwctob\fR(3C),
74 \fBAttributes\fR(5), \fBenviron\fR(5), \fBstandards\fR(5)
75 \fBsetlocale\fR(3C), \fBwctob\fR(3C), \fBAttributes\fR(5), \fBenviron\fR(5),
76 \fBstandards\fR(5)
77 .SH NOTES
78 .sp
79 .LP
80 The \fBbtowc()\fR function can be used safely in multithreaded applications, as
81 long as \fBsetlocale\fR(3C) is not being called to change the locale.

```



```

*****
9515 Fri Jul 11 11:43:07 2014
new/usr/src/man/man3c/ctype.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****

```

```

1  \" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright 1989 AT&T. Copyright (c) 1992, X/Open Company Limited All Rights
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH CTYPE 3C "Jun 27, 2014"
12 .TH CTYPE 3C "Jan 28, 2005"
13 .SH NAME
14 ctype, isalpha, isalnum, isascii, isblank, iscntrl, isdigit, islower,
15 isprint, isspace, isupper, ispunct, isgraph, isxdigit, isalpha_1,
16 isalnum_1, isblank_1, iscntrl_1, isdigit_1, islower_1, isprint_1,
17 isspace_1, isupper_1, ispunct_1, isgraph_1 \- character handling
18 ctype, isalpha, isalnum, isascii, isblank, iscntrl, isdigit, islower, isprint,
19 isspace, isupper, ispunct, isgraph, isxdigit \- character handling
20 .SH SYNOPSIS
21 .LP
22 .nf
23 #include <ctype.h>
24 .LP
25 .nf
26 \fBint\fR \fBisalpha\fR(\fBint\fR \fIc\fR);
27 .fi
28 .LP
29 .nf
30 \fBint\fR \fBisalnum\fR(\fBint\fR \fIc\fR);
31 .fi
32 .LP
33 .nf
34 \fBint\fR \fBisascii\fR(\fBint\fR \fIc\fR);
35 .fi
36 .LP
37 .nf
38 \fBint\fR \fBisblank\fR(\fBint\fR \fIc\fR);
39 .fi
40 .LP
41 .nf
42 \fBint\fR \fBisctrl\fR(\fBint\fR \fIc\fR);
43 .fi
44 .LP
45 .nf
46 \fBint\fR \fBisdigit\fR(\fBint\fR \fIc\fR);
47 .fi
48 .LP

```

```

49 .nf
50 \fBint\fR \fBislower\fR(\fBint\fR \fIc\fR);
51 .fi
52 .LP
53 .nf
54 \fBint\fR \fBisprint\fR(\fBint\fR \fIc\fR);
55 .fi
56 .LP
57 .nf
58 \fBint\fR \fBispunct\fR(\fBint\fR \fIc\fR);
59 .fi
60 .LP
61 .nf
62 \fBint\fR \fBisspace\fR(\fBint\fR \fIc\fR);
63 .fi
64 .LP
65 .nf
66 \fBint\fR \fBisupper\fR(\fBint\fR \fIc\fR);
67 .fi
68 .LP
69 .nf
70 \fBint\fR \fBisxdigit\fR(\fBint\fR \fIc\fR);
71 .fi
72 .LP
73 .nf
74 \fBint\fR \fBisalpha_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
75 .fi
76 .LP
77 .nf
78 \fBint\fR \fBisalnum_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
79 .fi
80 .LP
81 .nf
82 \fBint\fR \fBisblank_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
83 .fi
84 .LP
85 .nf
86 \fBint\fR \fBiscntrl_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
87 .fi
88 .LP
89 .nf
90 \fBint\fR \fBisdigit_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
91 .fi
92 .LP
93 .nf
94 \fBint\fR \fBisgraph_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
95 .fi
96 .LP
97 .nf
98 \fBint\fR \fBislower_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
99 .fi
100 .LP
101 .nf
102 \fBint\fR \fBisprint_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
103 .fi
104 .LP
105 .nf
106 \fBint\fR \fBispunct_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
107 .fi
108 .LP
109 .nf

```

```

110 \fBint\fr \fBisspace_1\fr(\fBint\fr \fIc\fr, \fBlocale_t\fr \fIloc\fr);
111 .fi
112 .LP
113 .nf
114 \fBint\fr \fBisupper_1\fr(\fBint\fr \fIc\fr, \fBlocale_t\fr \fIloc\fr);
115 .fi

116 .SH DESCRIPTION
117 .sp
118 These functions classify character-coded integer values. Each is a
119 predicate returning non-zero for true, \fB0\fr for false. The behavior
120 of these macros, except \fBisascii()\fr, is affected by the current
121 locale (see \fBsetlocale\fr(3C) and \fBuselocale\fr(3C)). To modify
122 the behavior, change the \fBLC_TYPE\fr category in \fBsetlocale()\fr, that is,
123 \fBsetlocale(\fr\fBLC_TYPE\fr, \fInewlocale\fr). In the "C" locale,
124 or in a locale where character type information is not defined,
125 characters are classified according to the rules of the \fBUS-ASCII\fr
126 7-bit coded character set.
127 .sp
128 These macros classify character-coded integer values. Each is a predicate
129 returning non-zero for true, \fB0\fr for false. The behavior of these macros,
130 except \fBisascii()\fr, is affected by the current locale (see
131 \fBsetlocale\fr(3C)). To modify the behavior, change the \fBLC_TYPE\fr category
132 in \fBsetlocale()\fr, that is, \fBsetlocale(\fr\fBLC_TYPE\fr,
133 \fInewlocale\fr). In the "C" locale, or in a locale where character type
134 information is not defined, characters are classified according to the rules of
135 the \fBUS-ASCII\fr 7-bit coded character set.
136 .sp
137 .LP
138 The functions \fBisalnum_1()\fr, \fBisalpha_1()\fr, \fBisblank_1()\fr,
139 \fBisctrl_1()\fr, \fBisdigit_1()\fr, \fBislower_1()\fr,
140 \fBisprint_1()\fr, \fBispunct_1()\fr, \fBisspace_1()\fr,
141 \fBisupper_1()\fr, all behave identically to their counterparts without
142 the '\fB_1\fr' prefix, except that instead of acting on the current
143 locale, that perform the test on the locale specified in the argument
144 \fIloc\fr.
145 .LP
146 The \fBisascii()\fr macro is defined on all integer values. The rest are
147 defined only where the argument is an \fBint\fr, the value of which is
148 representable as an \fBunsigned char\fr, or \fBEOF\fr, which is defined by the
149 <\fBstdio.h\fr> header and represents end-of-file.
150 .sp
151 .LP
152 Functions exist for all the macros defined below. To get the function form, the
153 macro name must be undefined (for example, \fB#undef isdigit\fr).
154 .sp
155 .LP
156 For macros described with \fBDefault\fr and \fBStandard conforming\fr versions,
157 standard-conforming behavior is provided for standard-conforming applications
158 (see \fBstandards\fr(5)) and for applications that define
159 \fB_XPG4_CHAR_CLASS__\fr before including <\fBctype.h\fr>.
160 .SS "Default"
161 .sp
162 .ne 2
163 .na
164 \fB\fBisalpha()\fr\fr
165 .ad
166 .RS 13n
167 Tests for any character for which \fBisupper()\fr or \fBislower()\fr is true.
168 .RE
169 .SS "Standard conforming"
170 .sp
171 .ne 2
172 .na
173 \fB\fBisalpha()\fr\fr

```

```

174 .ad
175 .RS 13n
176 Tests for any character for which \fBisupper()\fr or \fBislower()\fr is true,
177 or any character that is one of the current locale-defined set of characters
178 for which none of \fBisctrl()\fr, \fBisdigit()\fr, \fBispunct()\fr, or
179 \fBisspace()\fr is true. In "C" locale, \fBisalpha()\fr returns true only for
180 the characters for which \fBisupper()\fr or \fBislower()\fr is true.
181 .RE
182 .sp
183 .ne 2
184 .na
185 \fB\fBisalnum()\fr\fr
186 .ad
187 .RS 13n
188 Tests for any character for which \fBisalpha()\fr or \fBisdigit()\fr is true
189 (letter or digit).
190 .RE
191 .sp
192 .ne 2
193 .na
194 \fB\fBisascii()\fr\fr
195 .ad
196 .RS 13n
197 Tests for any ASCII character, code between \fB0\fr and \fB0177\fr inclusive.
198 .RE
199 .sp
200 .ne 2
201 .na
202 \fB\fBisblank()\fr\fr
203 .ad
204 .RS 13n
205 Tests whether \fIc\fr is a character of class blank in the current locale. This
206 macro/function is not available to applications conforming to standards prior
207 to SUSv3. See \fBstandards\fr(5)
208 .RE
209 .sp
210 .ne 2
211 .na
212 \fB\fBisctrl()\fr\fr
213 .ad
214 .RS 13n
215 Tests for any ``control character`` as defined by the character set.
216 .RE
217 .sp
218 .ne 2
219 .na
220 \fB\fBisgraph()\fr\fr
221 .ad
222 .RS 13n
223 Tests for any character for which \fBispunct()\fr, \fBisupper()\fr,
224 \fBislower()\fr, and \fBisdigit()\fr is true.

```

```

190 .RE
192 .SS "Standard conforming"
193 .sp
194 .ne 2
195 .na
196 \fB\fBisgraph()\fR\fR
197 .ad
198 .RS 13n
207 Tests for any character for which \fBisalnum()\fR and \fBispunct()\fR are true,
208 or any character in the current locale-defined "graph" class which is neither a
209 space ("\|") nor a character for which \fBisctrnl()\fR is true.
210 .RE
212 .sp
213 .ne 2
214 .na
215 \fB\fBislower()\fR\fR
216 .ad
217 .RS 13n
218 Tests for any character that is a lower-case letter or is one of the current
219 locale-defined set of characters for which none of \fBisctrnl()\fR,
220 \fBisdigit()\fR, \fBispunct()\fR, \fBispace()\fR, or \fBisupper()\fR is true.
221 In the "C" locale, \fBislower()\fR returns true only for the characters defined
222 as lower-case \fBASCII\fR characters.
223 .RE
217 .SS "Default"
225 .sp
226 .ne 2
227 .na
228 \fB\fBisprint()\fR\fR
229 .ad
230 .RS 13n
224 Tests for any character for which \fBispunct()\fR, \fBisupper()\fR,
225 \fBislower()\fR, \fBisdigit()\fR, and the space character ("\|") is true.
226 .RE
228 .SS "Standard conforming"
229 .sp
230 .ne 2
231 .na
232 \fB\fBisprint()\fR\fR
233 .ad
234 .RS 13n
231 Tests for any character for which \fBisctrnl()\fR is false, and
232 \fBisalnum()\fR, \fBisgraph()\fR, \fBispunct()\fR, the space character ("\|"),
233 and the characters in the current locale-defined "print" class are true.
234 .RE
236 .sp
237 .ne 2
238 .na
239 \fB\fBispunct()\fR\fR
240 .ad
241 .RS 13n
242 Tests for any printing character which is neither a space ("\|") nor a
243 character for which \fBisalnum()\fR or \fBisctrnl()\fR is true.
244 .RE
246 .sp
247 .ne 2
248 .na
249 \fB\fBispace()\fR\fR
250 .ad
251 .RS 13n

```

```

252 Tests for any space, tab, carriage-return, newline, vertical-tab or form-feed
253 (standard white-space characters) or for one of the current locale-defined set
254 of characters for which \fBisalnum()\fR is false. In the "C" locale,
255 \fBispace()\fR returns true only for the standard white-space characters.
256 .RE
258 .sp
259 .ne 2
260 .na
261 \fB\fBisupper()\fR\fR
262 .ad
263 .RS 13n
264 Tests for any character that is an upper-case letter or is one of the current
265 locale-defined set of characters for which none of \fBisctrnl()\fR,
266 \fBisdigit()\fR, \fBispunct()\fR, \fBispace()\fR, or \fBislower()\fR is true.
267 In the "C" locale, \fBisupper()\fR returns true only for the characters defined
268 as upper-case \fBASCII\fR characters.
269 .RE
275 .SS "Default"
271 .sp
272 .ne 2
273 .na
274 \fB\fBisxdigit()\fR\fR
275 .ad
276 .RS 14n
277 Tests for any hexadecimal-digit character (\fB[0\mi9]\fR, \fB[A\miF]\fR, or
278 \fB[a\mif]\fR).
284 .RE
286 .SS "Standard conforming"
287 .sp
288 .ne 2
289 .na
290 \fB\fBisxdigit()\fR\fR
291 .ad
292 .RS 14n
293 Tests for any hexadecimal-digit character (\fB[0\mi9]\fR, \fB[A\miF]\fR, or
278 \fB[a\mif]\fR or the current locale-defined sets of characters representing
279 the hexadecimal digits \fB10\fR to \fB15\fR inclusive). In the "C" locale, only
280 .sp
281 .in +2
282 .nf
283 0 1 2 3 4 5 6 7 8 9 A B C D E F a b c d e f
284 .fi
285 .in -2
287 are included.
288 .RE
290 .SH RETURN VALUES
307 .sp
291 .LP
292 If the argument to any of the character handling macros is not in the domain of
293 the function, the result is undefined. Otherwise, the macro or function returns
294 non-zero if the classification is \fBTRUE\fR and \fB0\fR if the classification
295 is \fBFALSE\fR.
313 .SH USAGE
314 .sp
315 .LP
316 These macros or functions can be used safely in multithreaded applications, as
317 long as \fBsetlocale(3C)\fR is not being called to change the locale.
296 .SH ATTRIBUTES
297 .sp
298 .LP
299 See \fBattributes(5)\fR for descriptions of the following attributes:

```

```
322 .sp
324 .sp
300 .TS
301 box;
302 c | c
303 l | l .
327 l l
328 l l .
304 ATTRIBUTE TYPE ATTRIBUTE VALUE
305 -
306 CSI Enabled
307 -
308 Interface Stability Standard
309 -
310 MT-Level MT-Safe
332 MT-Level MT-Safe with exceptions
311 .TE

313 .SH SEE ALSO
336 .sp
314 .LP
315 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
316 \fBstdio\fR(3C), \fBascii\fR(5), \fBenviron\fR(5),
338 \fBsetlocale\fR(3C), \fBstdio\fR(3C), \fBascii\fR(5), \fBenviron\fR(5),
317 \fBstandards\fR(5)
```

```

*****
6350 Fri Jul 11 11:43:07 2014
new/usr/src/man/man3c/fgetwc.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \" te
2  \\ Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  \\ Copyright (c) 2003, X/Open Company Limited. All Rights Reserved. Portions
4  \\ Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  \\ http://www.opengroup.org/bookstore/.
6  \\ The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  \\ This notice shall appear on any product containing this material.
8  \\ The contents of this file are subject to the terms of the Common Development
9  \\ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 \\ When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH FGETWC 3C "Jun 24, 2014"
12 .SH NAME
13 fgetwc, fgetwc_l - get a wide-character code from a stream
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <stdio.h>
18 #include <wchar.h>
19
20 \fBwint_t\fR \fBfgetwc\fR(\fBFILE * \fR \fR \fR \fR \fR \fR \fR \fR);
21 \fBwint_t\fR \fBfgetwc_l\fR(\fBFILE * \fR \fR \fR \fR \fR \fR \fR \fR);
22 .fi
23 .LP
24 .nf
25 #include <stdio.h>
26 #include <wchar.h>
27 #include <xlocale.h>
28
29 \fBwint_t\fR \fBfgetwc_l\fR(\fBFILE * \fR \fR \fR \fR \fR \fR \fR \fR, \fR \fR \fR \fR \fR \fR \fR \fR);
30 .fi
31 .SH DESCRIPTION
32 .sp
33 .LP
34 The \fBfgetwc()\fR and \fBfgetwc_l()\fR functions obtain the next
35 character (if present) from the input stream pointed to by \fR \fR \fR \fR \fR \fR \fR \fR,
36 convert that to the corresponding wide-character code and advance the
37 associated file position indicator for the stream (if defined).
38 Whereas \fBfgetwc()\fR uses the current locale, \fBfgetwc_l()\fR uses the
39 locale specified by \fR \fR \fR \fR \fR \fR \fR \fR.
40 The \fBfgetwc()\fR function obtains the next character (if present) from the
41 input stream pointed to by \fR \fR \fR \fR \fR \fR \fR \fR, converts that to the corresponding
42 wide-character code and advances the associated file position indicator for the
43 stream (if defined).
44 .sp
45 .LP
46 If an error occurs, the resulting value of the file position indicator for the
47 stream is indeterminate.
48 .sp
49 .LP
50 The \fBfgetwc()\fR and \fBfgetwc_l()\fR functions may mark the \fR \fR \fR \fR \fR \fR \fR \fR
51 field of the file
52 The \fBfgetwc()\fR function may mark the \fR \fR \fR \fR \fR \fR \fR \fR field of the file
53 associated with \fR \fR \fR \fR \fR \fR \fR \fR for update. The \fR \fR \fR \fR \fR \fR \fR \fR field will be
54 marked for update by the first successful execution of \fBfgetwc()\fR,
55 \fBfgetc()\fR(3C), \fBfgetc_s()\fR(3C), \fBfgetws()\fR(3C), \fBfread()\fR(3C),

```

```

49 \fBfscanf()\fR(3C), \fBgetc()\fR(3C), \fBgetchar()\fR(3C), \fBgets()\fR(3C), or
50 \fBscanf()\fR(3C) using \fR \fR \fR \fR \fR \fR \fR \fR that returns data not supplied by a prior
51 call to \fBungetc()\fR(3C) or \fBungetwc()\fR(3C).
52 .SH RETURN VALUES
53 .sp
54 .LP
55 Upon successful completion both functions return the
56 Upon successful completion the \fBfgetwc()\fR function returns the
57 wide-character code of the character read from the input stream pointed to by
58 \fR \fR \fR \fR \fR \fR \fR \fR converted to a type \fR \fR \fR \fR \fR \fR \fR \fR.
59 .sp
60 .LP
61 For standard-conforming (see \fR \fR \fR \fR \fR \fR \fR \fR) applications, if the
62 end-of-file indicator for the stream is set, \fBfgetwc()\fR and
63 \fBfgetwc_l()\fR return \fR \fR \fR \fR \fR \fR \fR \fR whether or not the stream is at
64 end-of-file.
65 end-of-file indicator for the stream is set, \fBfgetwc()\fR returns \fR \fR \fR \fR \fR \fR \fR \fR
66 whether or not the stream is at end-of-file.
67 .sp
68 .LP
69 If a read error occurs, the error indicator for the stream is set,
70 \fBfgetwc()\fR and \fBfgetwc_l()\fR returns \fR \fR \fR \fR \fR \fR \fR \fR and sets
71 \fBfBerrno()\fR to indicate the error.
72 \fBfgetwc()\fR returns \fR \fR \fR \fR \fR \fR \fR \fR and sets \fR \fR \fR \fR \fR \fR \fR \fR to indicate the error.
73 .sp
74 .LP
75 If an encoding error occurs, the error indicator for the stream is set,
76 \fBfgetwc()\fR and \fBfgetwc_l()\fR return \fR \fR \fR \fR \fR \fR \fR \fR, and \fBfBerrno()\fR is
77 set to indicate the error.
78 \fBfgetwc()\fR returns \fR \fR \fR \fR \fR \fR \fR \fR, and \fBfBerrno()\fR is set to indicate the
79 error.
80 .SH ERRORS
81 .sp
82 .LP
83 The \fBfgetwc()\fR and \fBfgetwc_l()\fR functions will fail if data needs to be
84 read and:
85 .TP
86 .B EAGAIN
87 The \fBfgetwc()\fR function will fail if data needs to be read and:
88 .sp
89 .ne 2
90 .na
91 \fB \fR \fR \fR \fR \fR \fR \fR \fR
92 .ad
93 .RS 14n
94 The \fR \fR \fR \fR \fR \fR \fR \fR flag is set for the file descriptor underlying
95 \fR \fR \fR \fR \fR \fR \fR \fR and the process would be delayed in the \fBfgetwc()\fR or
96 \fBfgetwc_l()\fR operation.
97 .TP
98 .B EBADF
99 \fR \fR \fR \fR \fR \fR \fR \fR and the process would be delayed in the \fBfgetwc()\fR operation.
100 .RE
101 .sp
102 .ne 2
103 .na
104 \fB \fR \fR \fR \fR \fR \fR \fR \fR
105 .ad
106 .RS 14n
107 The file descriptor underlying \fR \fR \fR \fR \fR \fR \fR \fR is not a valid file descriptor open
108 for reading.
109 .TP
110 .B EINTR
111 .RE
112 .sp

```

```

87 .ne 2
88 .na
89 \fB\fBEINTR\fR\fR
90 .ad
91 .RS 14n
85 The read operation was terminated due to the receipt of a signal, and no data
86 was transferred.
87 .TP
88 .B EIO
94 .RE

96 .sp
97 .ne 2
98 .na
99 \fB\fBEIO\fR\fR
100 .ad
101 .RS 14n
89 A physical I/O error has occurred, or the process is in a background process
90 group attempting to read from its controlling terminal and either the process
91 is ignoring or blocking the \fBSIGTTIN\fR signal or the process group is
92 orphaned.
93 .TP
94 .B EOVERFLOW
106 .RE

108 .sp
109 .ne 2
110 .na
111 \fB\fBEVERFLOW\fR \fR
112 .ad
113 .RS 14n
95 The file is a regular file and an attempt was made to read at or beyond the
96 offset maximum associated with the corresponding \fIstream\fR.
116 .RE

118 .sp
97 .LP
98 The \fBfgetwc()\fR and \fBfgetwc_l()\fR functions may fail if:
99 .TP
100 .B ENOMEM
101 Insufficient memory is available.
102 .TP
103 .B ENXIO
120 The \fBfgetwc()\fR function may fail if:
121 .sp
122 .ne 2
123 .na
124 \fB\fBENOMEM\fR\fR
125 .ad
126 .RS 10n
127 Insufficient storage space is available.
128 .RE

130 .sp
131 .ne 2
132 .na
133 \fB\fBENXIO\fR\fR
134 .ad
135 .RS 10n
104 A request was made of a non-existent device, or the request was outside the
105 capabilities of the device.
106 .TP
107 .B EILSEQ
138 .RE

140 .sp

```

```

141 .ne 2
142 .na
143 \fB\fBEILSEQ\fR\fR
144 .ad
145 .RS 10n
108 The data obtained from the input stream does not form a valid character.
147 .RE

109 .SH USAGE
110 .sp
111 .LP
112 The \fBferror\fR(3C) or \fBfeof\fR(3C) functions must be used to distinguish
113 between an error condition and an end-of-file condition.
114 .SH ATTRIBUTES
115 .sp
116 .LP
117 See \fBattributes\fR(5) for descriptions of the following attributes:
158 .sp

160 .sp
118 .TS
119 box;
120 c | c
163 l | l
121 l | l .
122 ATTRIBUTE TYPE ATTRIBUTE VALUE
123 -
124 CSI Enabled
125 -
126 Interface Stability See below.
169 Interface Stability Standard
127 -
128 MT-Level MT-Safe
171 MT-Level MT-Safe with exceptions
129 .TE

131 .LP
132 The
133 \fBfgetwc()\fR
134 function is Standard. The
135 \fBfgetwc_l()\fR
136 function is Uncommitted.
137 .SH SEE ALSO
175 .sp
138 .LP
139 \fBfeof\fR(3C), \fBferror\fR(3C), \fBfgetc\fR(3C), \fBfgets\fR(3C),
140 \fBfgetws\fR(3C), \fBfopen\fR(3C), \fBfread\fR(3C), \fBfscanf\fR(3C),
141 \fBgetc\fR(3C), \fBgetchar\fR(3C), \fBgets\fR(3C), \fBscanf\fR(3C),
142 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBungetc\fR(3C), \fBungetwc\fR(3C),
143 \fBuselocale\fR(3C), \fBattributes\fR(5),
180 \fBsetlocale\fR(3C), \fBungetc\fR(3C), \fBungetwc\fR(3C), \fBattributes\fR(5),
144 \fBstandards\fR(5)

```

\*\*\*\*\*

3858 Fri Jul 11 11:43:07 2014

new/usr/src/man/man3c/getwc.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH GETWC 3C "Jul 11, 2014"
10 .TH GETWC 3C "Jul 24, 2002"
12 .SH NAME
13 getwc, getwc_l \- get wide character from a stream
12 getwc \- get wide character from a stream
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <stdio.h>
18 #include <wchar.h>

20 \fBwint_t\fR \fBgetwc\fR(\fBFILE\fR \fI*stream\fR);
21 .fi
22 .LP
23 .nf
24 #include <stdio.h>
25 #include <wchar.h>
26 #include <xlocale.h>

28 \fBwint_t\fR \fBgetwc_l\fR(\fBFILE\fR \fI*stream\fR, \fBlocale_t\fR \fIloc\fR);
29 .fi

31 .SH DESCRIPTION
32 .sp
33 .LP
34 The \fBgetwc()\fR function is equivalent to \fBfgetwc\fR(3C), except that if it
35 is implemented as a macro it may evaluate \fIstream\fR more than once, so the
36 argument should never be an expression with side effects.
37 .LP
38 The \fBgetwc_l()\fR function is similar to \fBgetwc()\fR, except instead of
39 acting on the current locale, it uses the locale specified by \fIloc\fR.
40 .SH RETURN VALUES
29 .sp
41 .LP
42 Refer to \fBfgetwc\fR(3C).
43 .SH ERRORS
33 .sp
44 .LP
45 Refer to \fBfgetwc\fR(3C).
46 .SH USAGE
37 .sp
47 .LP
48 This interface is provided to align with some current implementations and with
49 possible future \fBISO\fR standards.
41 .sp
50 .LP
51 Because they may be implemented as macros, these functions may treat incorrectly
52 a \fIstream\fR argument with side effects. In particular,
```

```

53 \fBgetwc\fR(*\fIstream\fR) may not work as expected. Therefore, use of these
54 functions is not recommended; \fBfgetwc\fR(3C) and \fBfgetwc_l\fR(3C)
55 should be used instead.
43 Because it may be implemented as a macro, \fBgetwc()\fR may treat incorrectly a
44 \fIstream\fR argument with side effects. In particular,
45 \fBgetwc\fR(*\fIstream\fR) may not work as expected. Therefore, use of this
46 function is not recommended; \fBfgetwc\fR(3C) should be used instead.
56 .SH ATTRIBUTES
48 .sp
57 .LP
58 See \fBattributes\fR(5) for descriptions of the following attributes:
51 .sp

53 .sp
59 .TS
60 box;
61 c | c
62 l | l .
63 ATTRIBUTE TYPE ATTRIBUTE VALUE
64 -
65 Interface Stability See below.
60 Interface Stability Standard
66 -
67 MT-Level MT-Safe
68 .TE

70 .LP
71 The
72 .B getwc()
73 function is Standard. The
74 .B getwc_l()
75 function is Uncommitted.
76 .SH SEE ALSO
66 .sp
77 .LP
78 \fBfgetwc\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
79 \fBattributes\fR(5), \fBstandards\fR(5)
68 \fBfgetwc\fR(3C), \fBattributes\fR(5), \fBstandards\fR(5)
```

\*\*\*\*\*

3333 Fri Jul 11 11:43:07 2014

new/usr/src/man/man3c/getwchar.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1 \" te
2 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3 .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions C
4 .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5 .\" http://www.opengroup.org/bookstore/.
6 .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7 .\" This notice shall appear on any product containing this material.
8 .\" The contents of this file are subject to the terms of the Common Development
9 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH GETWCHAR 3C "Jul 24, 2002"
12 .SH NAME
13 getwchar \- get wide character from stdin stream
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <wchar.h>

19 \fBwint_t\fR \fBgetwchar\fR(void)
20 .fi
21 .LP
22 .nf
23 #include <wchar.h>
24 #include <xlocale.h>

26 \fBwint_t\fR \fBgetwchar_l\fR(locale_t loc)
27 .fi
28 .SH DESCRIPTION
29 .sp
30 .LP
31 The \fBgetwchar()\fR function is equivalent to \fBgetwc\fR(\fBstdin\fR).
32 .LP
33 The \fBgetwchar_l()\fR function is equivalent to \fBgetwc\fR(\fBstdin\fR, \fBilloc\
34 The \fBgetwchar()\fR function is equivalent to \fBgetwc(stdin)\fR.
35 .SH RETURN VALUES
36 .sp
37 .LP
38 Refer to \fBfgetwc\fR(3C).
39 .SH ERRORS
40 .sp
41 .LP
42 Refer to \fBfgetwc\fR(3C).
43 .SH USAGE
44 .sp
45 .LP
46 If the \fBwint_t\fR value returned by \fBgetwchar()\fR is stored into a
47 variable of type \fBwchar_t\fR and then compared against the \fBwint_t\fR macro
48 \fBWEOF\fR, the comparison may never succeed because \fBwchar_t\fR is defined
49 as unsigned.
50 .SH ATTRIBUTES
51 .sp
52 .LP
53 See \fBattributes\fR(5) for descriptions of the following attributes:
54 .sp
55 .sp
56 .TS
57 box;
```

```

50 c | c
51 l | l .
52 ATTRIBUTE TYPE ATTRIBUTE VALUE
53 -
54 Interface Stability See below.
55 Interface Stability Standard
56 -
57 MT-Level MT-Safe
58 .TE

59 .LP
60 The \fBgetwchar()\fR function is Standard. The \fBgetwchar_l()\fR function
61 is Uncommitted.
62 .SH SEE ALSO
63 .sp
64 \fBfgetwc\fR(3C), \fBgetwc\fR(3C), \fBattributes\fR(5), \fBstandards\fR(5)
```



\*\*\*\*\*

11588 Fri Jul 11 11:43:08 2014

new/usr/src/man/man3c/iswalpha.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \'  te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH ISWALPHA 3C "Jun 23, 2014"
12 .TH ISWALPHA 3C "Apr 19, 2004"
13 .SH NAME
14 iswalpha, isenglish, isideogram, iswideogram, isnumber, iswnumber,
15 isphonogram, iswphonogram, isspecial, iswspecial, iswalnum, iswascii,
16 iswblank, iswcntrl, iswdigit, iswgraph, iswlower, iswprint, iswpunct,
17 iswspace, iswupper, iswxdigit, iswhexnumber, iswalpha_1, iswideogram_1,
18 iswnumber_1, iswphonogram_1, iswspecial_1, iswalnum_1, iswascii_1,
19 iswblank_1, iswcntrl_1, iswdigit_1, iswgraph_1, iswlower_1, iswprint_1,
20 iswpunct_1, iswspace_1, iswupper_1, iswhexnumber_1 \- wide-character
   code classification functions

12 iswalpha, isenglish, isideogram, isnumber, isphonogram, isspecial, iswalnum,
13 iswascii, iswblank, iswcntrl, iswdigit, iswgraph, iswlower, iswprint, iswpunct,
14 iswspace, iswupper, iswxdigit \- wide-character code classification functions
22 .SH SYNOPSIS
23 .LP
24 .nf
25 #include <wchar.h>
26 #include <wctype.h>

28 \fBint\fR \fBiswalpha\fR(\fBwint_t\fR \fRIwc\fR);
29 .fi

30 .LP
31 .nf
32 \fBint\fR \fBisenglish\fR(\fBwint_t\fR \fRIwc\fR);
33 .fi

34 .LP
35 .nf
36 \fBint\fR \fBisideogram\fR(\fBwint_t\fR \fRIwc\fR);
37 .fi

38 .LP
39 .nf
40 \fBint\fR \fBisnumber\fR(\fBwint_t\fR \fRIwc\fR);
41 .fi

42 .LP
43 .nf
44 \fBint\fR \fBisphonogram\fR(\fBwint_t\fR \fRIwc\fR);
45 .fi

46 .LP
47 .nf
48 \fBint\fR \fBisspecial\fR(\fBwint_t\fR \fRIwc\fR);
49 .fi

```

```

50 .LP
51 .nf
52 \fBint\fR \fBiswalnum\fR(\fBwint_t\fR \fRIwc\fR);
53 .fi

54 .LP
55 .nf
56 \fBint\fR \fBiswascii\fR(\fBwint_t\fR \fRIwc\fR);
57 .fi

58 .LP
59 .nf
60 \fBint\fR \fBiswblank\fR(\fBwint_t\fR \fRIwc\fR);
61 .fi

62 .LP
63 .nf
64 \fBint\fR \fBiswcntrl\fR(\fBwint_t\fR \fRIwc\fR);
65 .fi

66 .LP
67 .nf
68 \fBint\fR \fBiswdigit\fR(\fBwint_t\fR \fRIwc\fR);
69 .fi

70 .LP
71 .nf
72 \fBint\fR \fBiswgraph\fR(\fBwint_t\fR \fRIwc\fR);
73 .fi

74 .LP
75 .nf
76 \fBint\fR \fBiswlower\fR(\fBwint_t\fR \fRIwc\fR);
77 .fi

78 .LP
79 .nf
80 \fBint\fR \fBiswprint\fR(\fBwint_t\fR \fRIwc\fR);
81 .fi

82 .LP
83 .nf
84 \fBint\fR \fBiswpunct\fR(\fBwint_t\fR \fRIwc\fR);
85 .fi

86 .LP
87 .nf
88 \fBint\fR \fBiswspace\fR(\fBwint_t\fR \fRIwc\fR);
89 .fi

90 .LP
91 .nf
92 \fBint\fR \fBiswupper\fR(\fBwint_t\fR \fRIwc\fR);
93 .fi

94 .LP
95 .nf
96 \fBint\fR \fBiswxdigit\fR(\fBwint_t\fR \fRIwc\fR);
97 .fi
98 .LP
99 .nf
100 \fBint\fR \fBiswhexnumber\fR(\fBwint_t\fR \fRIwc\fR);
101 .fi
102 .LP
103 .nf

```

```

104 \fBint\fr \fBiswideogram\fr(\fBwint_t\fr \fIwc\fr);
105 .fi
106 .LP
107 .nf
108 \fBint\fr \fBiswnumber\fr(\fBwint_t\fr \fIwc\fr);
109 .fi
110 .LP
111 .nf
112 \fBint\fr \fBiswphonogram\fr(\fBwint_t\fr \fIwc\fr);
113 .fi
114 .LP
115 .nf
116 \fBint\fr \fBiswspecial\fr(\fBwint_t\fr \fIwc\fr);
117 .fi
118 .LP
119 .nf
120 \fBint\fr \fBiswalpha_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
121 .fi
122 .LP
123 .nf
124 \fBint\fr \fBiswideogram_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
125 .fi
126 .LP
127 .nf
128 \fBint\fr \fBiswnumber_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
129 .fi
130 .LP
131 .nf
132 \fBint\fr \fBiswphonogram_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
133 .fi
134 .LP
135 .nf
136 \fBint\fr \fBiswspecial_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
137 .fi
138 .LP
139 .nf
140 \fBint\fr \fBiswalnum_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
141 .fi
142 .LP
143 .nf
144 \fBint\fr \fBiswascii_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
145 .fi
146 .LP
147 .nf
148 \fBint\fr \fBiswblank_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
149 .fi
150 .LP
151 .nf
152 \fBint\fr \fBiswcntrl_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
153 .fi
154 .LP
155 .nf
156 \fBint\fr \fBiswdigit_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
157 .fi
158 .LP
159 .nf
160 \fBint\fr \fBiswgraph_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
161 .fi
162 .LP
163 .nf
164 \fBint\fr \fBiswlower_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
165 .fi
166 .LP
167 .nf
168 \fBint\fr \fBiswprint_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
169 .fi

```

```

170 .LP
171 .nf
172 \fBint\fr \fBiswpunct_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
173 .fi
174 .LP
175 .nf
176 \fBint\fr \fBiswspace_l\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
177 .fi
178 .LP
179 .nf
180 \fBint\fr \fBiswupper\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
181 .fi
182 .LP
183 .nf
184 \fBint\fr \fBiswxdigit\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
185 .fi
186 .LP
187 .nf
188 #include <xlocale.h>

190 \fBint\fr \fBiswhexnumber\fr(\fBwint_t\fr \fIwc\fr, \fBlocale_t\fr \fIloc\fr);
191 .fi

193 .SH DESCRIPTION
110 .sp
194 .LP
195 These functions test whether \fIwc\fr is a wide-character code representing a
196 character of a particular class defined in the \fBLC_TYPE\fr category of the
197 current locale or the locale specified by \fIloc\fr.
114 current locale.
115 .sp
198 .LP
199 The functions \fBiswalpha_l()\fr, \fBiswideogram_l()\fr, \fBiswnumber_l()\fr,
200 \fBiswphonogram_l()\fr, \fBiswspecial_l()\fr, \fBiswalnum_l()\fr,
201 \fBiswascii_l()\fr, \fBiswblank_l()\fr, \fBiswcntrl_l()\fr, \fBiswdigit_l()\fr,
202 \fBiswgraph_l()\fr, \fBiswlower_l()\fr, \fBiswprint_l()\fr, \fBiswpunct_l()\fr,
203 \fBiswspace_l()\fr, \fBiswupper_l()\fr, all behave identically to their
204 counterparts without the '\fB_l\fr' suffix, but instead of operating on the
205 current locale, they operate on the locale specified by \fIloc\fr.
206 .LP
207 The functions, \fBiswideogram()\fr, \fBiswnumber()\fr, \fBiswphonogram()\fr,
208 \fBiswspecial()\fr, and \fBiswhexnumber()\fr, are respectively identical to
209 \fBisideogram()\fr, \fBisnumber()\fr, \fBisphonogram()\fr, \fBissspecial()\fr,
210 \fBiswxdigit()\fr respectively. They are provided for compatability purposes.
211 .LP
212 In all cases, \fIwc\fr is a \fBwint_t\fr, the value of which must be a
213 wide-character code corresponding to a valid character in the current locale or
214 must equal the value of the macro \fBWEOF\fr. If the argument has any other
215 values, the behavior is undefined.
216 .sp
217 .ne 2
218 .na
219 \fB\fBiswalpha(\fr\fr\fr\fr\fr)\fr\fr
220 .ad
221 .RS 19n
222 Tests whether \fIwc\fr is a wide-character code representing a character of
223 class "alpha" in the program's current locale.
224 .RE

226 .sp
227 .ne 2
228 .na
229 \fB\fBisenglish(\fr\fr\fr\fr\fr)\fr\fr
230 .ad
231 .RS 19n
232 Tests whether \fIwc\fr is a wide-character code representing an English

```

```

233 language character, excluding \fBASCII\fr characters.
234 .RE

236 .sp
237 .ne 2
238 .na
239 \fB\fBisideogram(\fr\frIwc\fr\frB)\fr\fr
240 .ad
241 .RS 19n
242 Tests whether \frIwc\fr is a wide-character code representing an ideographic
243 language character, excluding \fBASCII\fr characters.
244 .RE

246 .sp
247 .ne 2
248 .na
249 \fB\fBisnumber(\fr\frIwc\fr\frB)\fr\fr
250 .ad
251 .RS 19n
252 Tests whether \frIwc\fr is a wide-character code representing digit [0\mi9],
253 excluding \fBASCII\fr characters.
254 .RE

256 .sp
257 .ne 2
258 .na
259 \fB\fBisphonogram(\fr\frIwc\fr\frB)\fr\fr
260 .ad
261 .RS 19n
262 Tests whether \frIwc\fr is a wide-character code representing a phonetic
263 language character, excluding \fBASCII\fr characters.
264 .RE

266 .sp
267 .ne 2
268 .na
269 \fB\fBisspecial(\fr\frIwc\fr\frB)\fr\fr
270 .ad
271 .RS 19n
272 Tests whether \frIwc\fr is a wide-character code representing a special language
273 character, excluding \fBASCII\fr characters.
274 .RE

276 .sp
277 .ne 2
278 .na
279 \fB\fBiswalnum(\fr\frIwc\fr\frB)\fr\fr
280 .ad
281 .RS 19n
282 Tests whether \frIwc\fr is a wide-character code representing a character of
283 class "alpha" or "digit" in the program's current locale.
284 .RE

286 .sp
287 .ne 2
288 .na
289 \fB\fBiswascii(\fr\frIwc\fr\frB)\fr\fr
290 .ad
291 .RS 19n
292 Tests whether \frIwc\fr is a wide-character code representing an \fBASCII\fr
293 character.
294 .RE

296 .sp
297 .ne 2
298 .na

```

```

299 \fB\fBiswblank(\fr\frIwc\fr\frB)\fr\fr
300 .ad
301 .RS 19n
302 Tests whether \frIwc\fr is a wide-character code representing a character of
303 class "blank" in the program's current locale. This function is not available
304 to applications conforming to standards prior to SUSv3. See \frBstandards\fr(5).
305 .RE

307 .sp
308 .ne 2
309 .na
310 \fB\fBiswlower(\fr\frIwc\fr\frB)\fr\fr
311 .ad
312 .RS 19n
313 Tests whether \frIwc\fr is a wide-character code representing a character of
314 class "lower" in the program's current locale.
315 .RE

317 .sp
318 .ne 2
319 .na
320 \fB\fBiswcntrl(\fr\frIwc\fr\frB)\fr\fr
321 .ad
322 .RS 19n
323 Tests whether \frIwc\fr is a wide-character code representing a character of
324 class "cntrl" in the program's current locale.
325 .RE

327 .sp
328 .ne 2
329 .na
330 \fB\fBiswdigit(\fr\frIwc\fr\frB)\fr\fr
331 .ad
332 .RS 19n
333 Tests whether \frIwc\fr is a wide-character code representing a character of
334 class "digit" in the program's current locale.
335 .RE

337 .sp
338 .ne 2
339 .na
340 \fB\fBiswgraph(\fr\frIwc\fr\frB)\fr\fr
341 .ad
342 .RS 19n
343 Tests whether \frIwc\fr is a wide-character code representing a character of
344 class "graph" in the program's current locale.
345 .RE

347 .sp
348 .ne 2
349 .na
350 \fB\fBiswprint(\fr\frIwc\fr\frB)\fr\fr
351 .ad
352 .RS 19n
353 Tests whether \frIwc\fr is a wide-character code representing a character of
354 class "print" in the program's current locale.
355 .RE

357 .sp
358 .ne 2
359 .na
360 \fB\fBiswpunct(\fr\frIwc\fr\frB)\fr\fr
361 .ad
362 .RS 19n
363 Tests whether \frIwc\fr is a wide-character code representing a character of
364 class "punct" in the program's current locale.

```

```

365 .RE
367 .sp
368 .ne 2
369 .na
370 \fB\fBiswspace(\fR\fIwc\fR\fB)\fR\fR
371 .ad
372 .RS 19n
373 Tests whether \fIwc\fR is a wide-character code representing a character of
374 class "space" in the program's current locale.
375 .RE
377 .sp
378 .ne 2
379 .na
380 \fB\fBiswupper(\fR\fIwc\fR\fB)\fR\fR
381 .ad
382 .RS 19n
383 Tests whether \fIwc\fR is a wide-character code representing a character of
384 class "upper" in the program's current locale.
385 .RE
387 .sp
388 .ne 2
389 .na
390 \fB\fBiswxdigit(\fR\fIwc\fR\fB)\fR\fR
391 .ad
392 .RS 19n
393 Tests whether \fIwc\fR is a wide-character code representing a character of
394 class "xdigit" in the program's current locale.
395 .RE
397 .SH ATTRIBUTES
398 .sp
399 .LP
400 See \fBattributes\fR(5) for descriptions of the following attributes:
401 .sp
402 box;
403 

|   |  |   |
|---|--|---|
| c |  | c |
| l |  | l |
| l |  | l |


404 l | l .
405 ATTRIBUTE TYPE ATTRIBUTE VALUE
406 -
407 CSI Enabled
408 -
409 Interface Stability See below.
410 -
411 MT-Level MT-Safe
412 MT-Level MT-Safe with exceptions
413 .TE
414 .sp
415 The \fBiswalpha()\fR, \fBiswalnum()\fR, \fBiswblank()\fR, \fBiswcntrl()\fR,
416 \fBiswdigit()\fR, \fBiswgraph()\fR, \fBiswlower()\fR, \fBiswprint()\fR,
417 \fBiswpunct()\fR, \fBiswspace()\fR, \fBiswupper()\fR, and \fBiswxdigit()\fR
418 functions are Standard.
419 .LP
420 The \fBiswalpha_l()\fR, \fBiswalnum_l()\fR, \fBiswblank_l()\fR, \fBiswcntrl_l()\fR,
421 \fBiswdigit_l()\fR, \fBiswgraph_l()\fR, \fBiswlower_l()\fR, \fBiswprint_l()\fR,
422 \fBiswpunct_l()\fR, \fBiswspace_l()\fR, and \fBiswupper_l()\fR
423 functions are Standard.
424 .SH SEE ALSO

```

```

329 .sp
329 .LP
426 \fBlocaledef\fR(1), \fBnewlocale\fR(3C),
427 \fBsetlocale\fR(3C), \fBstdio\fR(3C), \fBuselocale\fR(3C),
428 \fBascii\fR(5), \fBattributes\fR(5), \fBstandards\fR(5)
331 \fBlocaledef\fR(1), \fBsetlocale\fR(3C), \fBstdio\fR(3C), \fBascii\fR(5),
332 \fBattributes\fR(5), \fBstandards\fR(5)

```

```

*****
4951 Fri Jul 11 11:43:08 2014
new/usr/src/man/man3c/iswctype.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \' te
2  \. Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  \. Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  \. Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  \. http://www.opengroup.org/bookstore/.
6  \. The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  \. This notice shall appear on any product containing this material.
8  \. The contents of this file are subject to the terms of the Common Development
9  \. You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 \. When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH ISWCTYPE 3C "Jul 26, 2014"
10 .TH ISWCTYPE 3C "Jul 24, 2002"
12 .SH NAME
13 iswctype \- test character for specified class
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <wchar.h>

19 \fBint\fR \fBiswctype\fR(\fBwint_t\fR \fIwc\fR, \fBwctype_t\fR \fIcharclass\fR);
20 .fi
21 .LP
22 .nf
23 \fBint\fR \fBiswctype\fR(\fBwint_t\fR \fIwc\fR, \fBwctype_t\fR \fIcharclass\fR,
24 .fi

26 .SH DESCRIPTION
22 .sp
27 .LP
28 The
29 .B iswctype()
30 and
31 .B iswctype_l()
32 functions determine whether the wide-character code
33 .I wc
34 is a member of the character class
35 .IR charclass ,
36 returning
37 .B TRUE
38 or
39 .BR FALSE .
40 These functions are defined on
41 .B WEOF
42 and wide-character codes corresponding to the valid character encodings in the
43 current locale (or
44 .I loc
45 for
46 .BR iswctype_l() .)
47 If the
48 .I wc
49 argument is not in the domain of the function,
50 the result is undefined. If the value of
51 .I charclass
52 is invalid (that is,
53 not obtained by a call to
54 .B wctype (3C)
55 or obtained from a different locale), the result is indeterminate.
56 .LP

```

```

57 Whereas
58 .B iswctype()
59 operates in the current locale,
60 .B iswctype_l()
61 operates in the locale specified with
62 .IR loc .
24 The \fBiswctype()\fR function determines whether the wide-character code
25 \fIwc\fR has the character class \fIcharclass\fR, returning \fBTRUE\fR or
26 \fBFALSE\fR. The \fBiswctype()\fR function is defined on \fBWEOF\fR and
27 wide-character codes corresponding to the valid character encodings in the
28 current locale. If the \fIwc\fR argument is not in the domain of the function,
29 the result is undefined. If the value of \fIcharclass\fR is invalid (that is,
30 not obtained by a call to \fBwctype\fR(3C) or \fIcharclass\fR is invalidated
31 by a subsequent call to \fBsetlocale\fR(3C) that has affected category
32 \fBLC_TYPE\fR), the result is indeterminate.
63 .SH RETURN VALUES
34 .sp
64 .LP
65 These functions return
66 .B 0
67 for
68 .B FALSE
69 and non-zero for
70 .BR TRUE .
36 The \fBiswctype()\fR function returns \fB0\fR for \fBFALSE\fR and non-zero for
37 \fBTRUE\fR.
71 .SH USAGE
39 .sp
72 .LP
73 There are twelve strings that are reserved for the standard character classes:
42 .sp

44 .sp
74 .TS
75 box;
46 l | l | l
76 l | l | l .
77 "alnum" "alpha" "blank"
78 -
79 "cntrl" "digit" "graph"
80 -
81 "lower" "print" "punct"
82 -
83 "space" "upper" "xdigit"
84 .TE

57 .sp
86 .LP
87 In the table below, the functions in the left column are equivalent to the
88 functions in the right column.
61 .sp

63 .sp
89 .TS
90 box;
91 l | l .
92 \fBiswalnum\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("alnum"))
93 \fBiswalpha\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("alpha"))
94 \fBiswcntrl\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("cntrl"))
95 \fBiswdigit\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("digit"))
96 \fBiswgraph\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("graph"))
97 \fBiswlower\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("lower"))
98 \fBiswprint\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("print"))
99 \fBiswpunct\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("punct"))
100 \fBiswspace\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("space"))
101 \fBiswupper\fR(\fIwc\fR) \fBiswctype\fR(\fIwc\fR, \fBwctype\fR("upper"))

```

```

102 \fBiswxdigit\fr(\fIwc\fr) \fBiswctype\fr(\fIwc\fr, \fBwctype\fr("xdigit"))
65 l l
66 l l .
67 \fBiswalnum(\fr\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"alnum"\fb))\fr
68 iswalpha(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"alpha"\fb))\fr
69 iswcntrl(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"cntrl"\fb))\fr
70 iswdigit(\fi wc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"digit"\fb))\fr
71 iswgraph(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"graph"\fb))\fr
72 iswlower(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"lower"\fb))\fr
73 iswprint(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"print"\fb))\fr
74 iswpunct(\fiwc\fr\fb)\fr iswctype(\fi wc,\fr\fb wctype(\fr"punct"\fb))\fr
75 iswspace(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"space"\fb))\fr
76 iswupper(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"upper"\fb))\fr
77 iswxdigit(\fiwc\fr\fb)\fr iswctype(\fiwc,\fr\fb wctype(\fr"xdigit"\fb))\fr
103 .TE

80 .sp
105 .LP
106 The call
83 .sp
107 .LP
108 \fBiswctype\fr(\fIwc\fr, \fBwctype\fr("blank"))
85 \fBiswctype(\fr\fiwc,\fr\fb wctype(\fr"blank"\fb))\fr
86 .sp
109 .LP
110 does not have an equivalent \fBisw*(\|\)\fr function.
111 .SH ATTRIBUTES
90 .sp
112 .LP
113 See
114 .BR attributes (5)
115 for descriptions of the following attributes:
92 See \fBattributes\fr(5) for descriptions of the following attributes:
93 .sp

95 .sp
116 .TS
117 box;
118 c | c
119 l | l .
120 ATTRIBUTE TYPE ATTRIBUTE VALUE
121 -
122 CSI Enabled
123 -
124 Interface Stability Standard
125 -
126 MT-Level MT-Safe
106 MT-Level MT-Safe with exceptions
127 .TE

129 .SH SEE ALSO
110 .sp
130 .LP
131 .BR iswalpha (3C),
132 .BR newlocale (3C),
133 .BR setlocale (3C),
134 .BR uselocale (3C),
135 .BR wctype (3C),
136 .BR attributes (5),
137 .BR standards (5)
112 \fBiswalpha\fr(3C), \fBsetlocale\fr(3C), \fBwctype\fr(3C), \fBattributes\fr(5),
113 \fBenviron\fr(5), \fBstandards\fr(5)

```

\*\*\*\*\*

4360 Fri Jul 11 11:43:08 2014

new/usr/src/man/man3c/mblen.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  '\" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright 1989 AT&T Copyright (c) 1992, X/Open Company Limited All Rights
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH MBLLEN 3C "Jun 24, 2014"
12 .TH MBLLEN 3C "Jul 24, 2002"
13 .SH NAME
14 mblen, mblen_l \- get number of bytes in a character
15 mblen \- get number of bytes in a character
16 .SH SYNOPSIS
17 .LP
18 .nf
19 \fBint\fR \fBmblen\fR(\fBconst char *\fR\fIs\fR, \fBsize_t\fR \fIn\fR);
20 .fi
21 .LP
22 .nf
23 #include <stdlib.h>
24 #include <xlocale.h>

26 \fBint\fR \fBmblen_l\fR(\fBconst char *\fR\fIs\fR, \fBsize_t\fR \fIn\fR, \fBloca
27 .fi

29 .SH DESCRIPTION
30 .sp
31 .LP
32 If \fIs\fR is not a null pointer, \fBmblen()\fR and \fBmblen_l()\fR
33 determine the number of bytes constituting the character pointed to by
34 \fIs\fR. The call
35 .IP
36 \fBmblen(\fIs\fR, \fIn\fR)\fR;
37 If \fIs\fR is not a null pointer, \fBmblen()\fR determines the number of bytes
38 constituting the character pointed to by \fIs\fR. It is equivalent to:
39 .sp
40 .LP
41 is equivalent to:
42 .IP
43 \fBmblen(\fIs\fR, \fIn\fR)\fR;
44 \fBmblen_l(\fIs\fR, \fIn\fR, \fBlocale_t\fR)\fR;
45 .sp
46 .LP
47 A call with \fIs\fR as a null pointer causes this function to return \fB0\fR.
48 .LP
49 Whereas the behavior of the \fBmblen()\fR function is affected by the
50 \fBLC_CTYPE\fR category of the current locale, the behavior of
51 \fBmblen_l()\fR is affected by the \fBLC_CTYPE\fR category of the
52 locale specified by \fBlocale_t\fR.
53 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
54 current locale.
55 .SH RETURN VALUES
56 .sp
57 .LP

```

```

49 .LP
50 If \fIs\fR is a null pointer, \fBmblen()\fR and \fBmblen_l()\fR returns \fB0\fR.
51 a null pointer, \fBmblen()\fR and \fBmblen_l()\fR returns \fB0\fR (if \fIs\fR po
52 If \fIs\fR is a null pointer, \fBmblen()\fR returns \fB0\fR. It \fIs\fR is not
53 a null pointer, \fBmblen()\fR returns \fB0\fR (if \fIs\fR points to the null
54 byte), the number of bytes that constitute the character (if the next \fIn\fR
55 or fewer bytes form a valid character), or \fB(mil)\fR (if they do not form a
56 valid character) and may set \fBerrno\fR to indicate the error. In no case will
57 the value returned be greater than \fIn\fR or the value of the \fBMB_CUR_MAX\fR
58 macro.
59 .SH ERRORS
60 .sp
61 .LP
62 The \fBmblen()\fR and \fBmblen_l()\fR functions may fail if:
63 The \fBmblen()\fR function may fail if:
64 .sp
65 .ne 2
66 .na
67 \fB(mil)\fR
68 .ad
69 .RS 10n
70 Invalid character sequence is detected.
71 .RE

72 .SH USAGE
73 .sp
74 .LP
75 The \fBmblen()\fR function can be used safely in multithreaded applications, as
76 long as \fBsetlocale(3C)\fR is not being called to change the locale.
77 .SH ATTRIBUTES
78 .sp
79 .LP
80 See \fBattributes(5)\fR for descriptions of the following attributes:
81 .sp

82 .sp
83 .TS
84 box;
85 c | c
86 l | l .
87 ATTRIBUTE TYPE ATTRIBUTE VALUE
88 -
89 CSI Enabled
90 -
91 Interface Stability See below.
92 Interface Stability Standard
93 -
94 MT-Level MT-Safe
95 MT-Level MT-Safe with exceptions
96 .TE

97 .LP
98 The
99 .B mblen()
100 function is Standard. The
101 .B mblen_l()
102 function is Uncommitted.
103 .SH SEE ALSO
104 .sp
105 .LP
106 \fBmblen(3C)\fR, \fBmblen_l(3C)\fR, \fBnewlocale(3C)\fR, \fBsetlocale(3C)\fR,
107 \fBwctomb(3C)\fR, \fBwctomb(3C)\fR, \fBuselocale(3C)\fR, \fBattributes(5)\fR,
108 \fBstandards(5)\fR
109 \fBmblen(3C)\fR, \fBmblen_l(3C)\fR, \fBsetlocale(3C)\fR, \fBwctomb(3C)\fR,
110 \fBwctomb(3C)\fR, \fBattributes(5)\fR, \fBstandards(5)\fR

```

```

*****
6081 Fri Jul 11 11:43:08 2014
new/usr/src/man/man3c/mbrlen.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH MBRLN 3C \"Jun 21, 2014\"
10 .TH MBRLN 3C \"Nov 1, 2003\"
12 .SH NAME
13 mbrlen, mbrlen_1 \- get number of bytes in a character (restartable)
12 mbrlen \- get number of bytes in a character (restartable)
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <wchar.h>

19 \fBsize_t\fR \fBmbrlen\fR(\fBconst char *restrict\fR \fBis\fR, \fBsize_t\fR \fBin\
20 .fi
21 .LP
22 .nf
23 #include <wchar.h>
24 #include <xlocale.h>

26 \fBsize_t\fR \fBmbrlen_1\fR(\fBconst char *restrict\fR \fBis\fR, \fBsize_t\fR \fBi
27 \fBlocale_t\fR \fBilloc\fR);
28 .fi
29 .SH DESCRIPTION
22 .sp
30 .LP
31 If \fBis\fR is not a null pointer, \fBmbrlen()\fR and \fBmbrlen_1()\fR determine
32 the number of bytes constituting the character pointed to by \fBis\fR.
33 The call
34 If \fBis\fR is not a null pointer, \fBmbrlen()\fR determines the number of
35 bytes constituting the character pointed to by \fBis\fR. It is equivalent to:
26 .sp
27 .in +2
34 .nf
35 .IP
36 \fBmbrlen\fR(\fBis\fR, \fBin\fR, \fBips\fR);
37 .fi
38 .LP
39 is equivalent to:
40 .nf
41 .IP
42 mbstate_t internal;
43 \fBmbrtowc\fR(NULL, \fBis\fR, \fBin\fR, \fBips\fR != NULL ? \fBips\fR : &internal);
30 \fBmbrtowc\fR(NULL, s, n, ps != NULL ? ps : &internal);
44 .fi
32 .in -2

34 .sp
45 .LP
46 If \fBips\fR is a null pointer, the \fBmbrlen()\fR and \fBmbrlen_1()\fR
47 functions use their own

```

```

36 If \fBips\fR is a null pointer, the \fBmbrlen()\fR function uses its own
48 internal \fBmbstate_t\fR object, which is initialized at program startup to the
49 initial conversion state. Otherwise, the \fBmbstate_t\fR object pointed to
50 by \fBips\fR is used to completely describe the current conversion state of the
51 associated character sequence. The implementation will behave as if no function
52 defined in the Reference Manual calls \fBmbrlen()\fR.
40 associated character sequence. Solaris will behave as if no function defined in
41 the Solaris Reference Manual calls \fBmbrlen()\fR.
42 .sp
43 .LP
54 The behavior of \fBmbrlen()\fR is affected by the \fBLC_CTYPE\fR category of the
55 current locale. See \fBenviron\fR(5). The behavior of \fBmbrlen_1()\fR does not
56 use the current environment and instead uses the locale specified by \fBlocale\fR.
44 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
45 current locale. See \fBenviron\fR(5).
57 .SH RETURN VALUES
47 .sp
48 .LP
59 The \fBmbrlen()\fR and \fBmbrlen_1()\fR functions return the first of the
60 following that applies:
49 The \fBmbrlen()\fR function returns the first of the following that applies:
61 .sp
62 .ne 2
63 .na
64 \fB0\fR
65 .ad
66 .RS 17n
67 If the next \fBin\fR or fewer bytes complete the character that corresponds to
68 the null wide-character.
69 .RE

71 .sp
72 .ne 2
73 .na
74 \fBpositive\fR
75 .ad
76 .RS 17n
77 If the next \fBin\fR or fewer bytes complete a valid character; the value
78 returned is the number of bytes that complete the character.
79 .RE

81 .sp
82 .ne 2
83 .na
84 \fB(size_t)\fR
85 .ad
86 .RS 17n
87 If the next \fBin\fR bytes contribute to an incomplete but potentially valid
88 character, and all \fBin\fR bytes have been processed. When \fBin\fR has at least
89 the value of the \fBMB_CUR_MAX\fR macro, this case can only occur if \fBis\fR
90 points at a sequence of redundant shift sequences (for implementations with
91 state-dependent encodings).
92 .RE

94 .sp
95 .ne 2
96 .na
97 \fB(size_t)\fR
98 .ad
99 .RS 17n
100 If an encoding error occurs, in which case the next \fBin\fR or fewer bytes do
101 not contribute to a complete and valid character. In this case, \fBEBLSEQ\fR
102 is stored in \fBerrno\fR and the conversion state is undefined.
103 .RE

105 .SH ERRORS

```



```

95 .sp
106 .LP
107 The \fBmbrlen() and \fBmbrlen_l() functions may fail if:
97 The \fBmbrlen() function may fail if:
108 .sp
109 .ne 2
110 .na
111 \fB\FBEINVAL\fR\fR
112 .ad
113 .RS 10n
114 The \fIps\fR argument points to an object that contains an invalid conversion
115 state.
116 .RE

118 .sp
119 .ne 2
120 .na
121 \fB\FBEILSEQ\fR\fR
122 .ad
123 .RS 10n
124 Invalid character sequence is detected.
125 .RE

127 .SH ATTRIBUTES
118 .sp
128 .LP
129 See \fBattributes\fR(5) for descriptions of the following attributes:
121 .sp

123 .sp
130 .TS
131 box;
132 c | c
133 l | l .
134 ATTRIBUTE TYPE ATTRIBUTE VALUE
135 -
136 Interface Stability See below.
130 Interface Stability Standard
137 -
138 MT-Level MT-Safe
132 MT-Level See NOTES below
139 .TE

141 .LP
142 The
143 .B mbrlen()
144 function is Standard. The
145 .B mbrlen_l()
146 function is Uncommitted.
147 .SH SEE ALSO
136 .sp
148 .LP
149 \fBmbrtowc\fR(3C), \fBmbsinit\fR(3C), \fBnewlocale\fR(3C),
150 \fBsetlocale\fR(3C), \fBattributes\fR(5),
138 \fBmbrtowc\fR(3C), \fBmbsinit\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),
151 \fBenviron\fR(5), \fBstandards\fR(5)
152 .SH NOTES
153 .sp
154 .LP
155 If \fIps\fR is not a null pointer, \fBmbrlen()\fR uses the \fBmbstate_t\fR
156 object pointed to by \fIps\fR and the function can be used safely in
157 multithreaded applications, as long as \fBsetlocale\fR(3C) is not being called
158 to change the locale or a per-thread locale has been installed on the calling
159 thread with \fBuselocale\fR(3C). If \fIps\fR is a null pointer, \fBmbrlen()\fR
160 uses its internal \fBmbstate_t\fR object and the function is Unsafe in
161 multithreaded applications.

```

```

146 to change the locale. If \fIps\fR is a null pointer, \fBmbrlen()\fR uses its
147 internal \fBmbstate_t\fR object and the function is Unsafe in multithreaded
148 applications.

```

\*\*\*\*\*

6227 Fri Jul 11 11:43:08 2014

new/usr/src/man/man3c/mbrtowc.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH MBRTOWC 3C "Jun 23, 2014"
12 .SH NAME
13 mbrtowc, mbrtowc_l \- convert a character to a wide-character code (restartable)
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <wchar.h>

19 \fBsize_t\fR \fBmbrtowc\fR(\fBwchar_t *restrict\fR \fBfipwc\fR, \fBfBconst char *res
20 \fBfBmbstate_t *restrict\fR \fBfips\fR);
21 .fi
22 .LP
23 .nf
24 #include <wchar.h>
25 #include <xlocale.h>

27 \fBsize_t\fR \fBmbrtowc_l\fR(\fBwchar_t *restrict\fR \fBfipwc\fR, \fBfBconst char *r
28 \fBfBmbstate_t *restrict\fR \fBfips\fR, \fBfBlocale_t\fR \fBfiloc\fR);
29 .fi
30 .SH DESCRIPTION
31 .LP
32 .If
33 .I s
34 is a null pointer, the
35 .B mbrtowc()
36 function is equivalent to the call:
37 .sp
38 .in +2
39 .nf
40 \fBmbrtowc\fR(NULL, "", 1, \fBfips\fR)
41 .fi
42 .in -2

34 .sp
43 .LP
44 Likewise, if
45 .I s
46 is a null pointer, the
47 .B mbrtowc_l()
48 function is equivalent to the call:
36 In this case, the values of the arguments \fBfipwc\fR and \fBfipwc\fR are ignored.
49 .sp

```

```

50 .in +2
51 .nf
52 \fBmbrtowc_l\fR(NULL, "", 1, \fBfips\fR, \fBfiloc\fR);
53 .fi
54 .in -2
55 .LP
56 In these cases, the values of the arguments
57 .I pwc
58 and
59 .I n
60 are ignored.
61 .LP
62 .If
63 .I s
64 is not a null pointer, these functions inspect at most
65 .I n
66 bytes beginning at the byte pointed to by
67 .I s
68 to determine the number of bytes needed to complete the next character
69 (including any shift sequences). If the functions determine that the next
70 character is completed,
71 they determine the value of the corresponding wide-character and then, if
72 .I pwc
73 is not a null pointer, stores that value in the object pointed to by
74 .IR pwc .
75 If the corresponding wide-character is the null wide-character, the
39 If \fBfips\fR is not a null pointer, the \fBmbrtowc()\fR function inspects at most
40 \fBfipwc\fR bytes beginning at the byte pointed to by \fBfips\fR to determine the
41 number of bytes needed to complete the next character (including any shift
42 sequences). If the function determines that the next character is completed,
43 it determines the value of the corresponding wide-character and then, if
44 \fBfipwc\fR is not a null pointer, stores that value in the object pointed to by
45 \fBfipwc\fR. If the corresponding wide-character is the null wide-character, the
76 resulting state described is the initial conversion state.
47 .sp
48 .LP
49 .If
50 .I ps
51 is a null pointer, these functions use their own
52 internal
53 .B mbstate_t
54 object, which is initialized at program startup to the
55 initial conversion state. Otherwise, the
56 .B mbstate_t
57 object pointed to by
58 .I ps
59 is used to completely describe the current conversion state of the
60 associated character sequence. The system will behave as if no function defined
61 in the Reference Manual calls
62 .B mbrtowc()
63 or
64 .BR mbrtowc_l() .
49 If \fBfips\fR is a null pointer, the \fBmbrtowc()\fR function uses its own
50 internal \fBmbstate_t\fR object, which is initialized at program startup to the
51 initial conversion state. Otherwise, the \fBmbstate_t\fR object pointed to
52 by \fBfips\fR is used to completely describe the current conversion state of the
53 associated character sequence. Solaris will behave as if no function defined in
54 the Solaris Reference Manual calls \fBmbrtowc()\fR.
55 .sp
56 .LP
57 The behavior of
58 .B mbrtowc()
59 is affected by the
60 .B LC_CTYPE
61 category of the current locale. The
100 .B mbrtowc_l()

```

```

101 function is affected by the
102 .B LC_CTYPE
103 category of the specified
104 .I loc
105 locale object. See
106 .B environ (5).
107 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
108 current locale. See \fBenviron\fR(5).
109 .SH RETURN VALUES
110 .sp
111 .LP
112 The
113 .B mbrtowc()
114 and
115 .B mbrtowc_l()
116 functions return the first of the following that applies:
117 .IP \fB0\fR
118 If the next
119 .I n
120 or fewer bytes complete the character that corresponds to
121 The \fBmbrtowc()\fR function returns the first of the following that applies:
122 .sp
123 .ne 2
124 .na
125 \fB\fB0\fR\fR
126 .ad
127 .RS 17n
128 If the next \fIn\fR or fewer bytes complete the character that corresponds to
129 the null wide-character (which is the value stored).
130 .IP \fBpositive\fR
131 If the next
132 .I n
133 or fewer bytes complete a valid character (which is the
134 .RE
135 .sp
136 .ne 2
137 .na
138 \fB\fBpositive\fR\fR
139 .ad
140 .RS 17n
141 If the next \fIn\fR or fewer bytes complete a valid character (which is the
142 value stored); the value returned is the number of bytes that complete the
143 character.
144 .IP \fB(size_t)\(mi2\fR
145 If the next
146 .I n
147 bytes contribute to an incomplete but potentially valid
148 character, and all
149 .I n
150 bytes have been processed (no value is stored).
151 When
152 .I n
153 has at least the value of the
154 .B MB_CUR_MAX
155 macro, this case can only occur if
156 .I s
157 points at a sequence of redundant shift sequences
158 .RE
159 .sp
160 .ne 2
161 .na
162 \fB\fB(size_t)\(mi2\fR\fR
163 .ad
164 .RS 17n

```

```

165 .sp
166 .ne 2
167 .na
168 \fB\fB(size_t)\(mi1\fR\fR
169 .ad
170 .RS 17n
171 If an encoding error occurs, in which case the next \fIn\fR or fewer bytes do
172 not contribute to a complete and valid character (no value is stored). In
173 this case,
174 .B EILSEQ
175 is stored in
176 .B errno
177 and the conversion state is undefined.
178 this case, \fBEILSEQ\fR is stored in \fBerrno\fR and the conversion state is
179 undefined.
180 .RE
181 .SH ERRORS
182 .sp
183 .LP
184 The
185 .B mbrtowc()
186 and
187 .B mbrtowc_l()
188 functions may fail if:
189 .IP \fBEINVAL\fR
190 The
191 .I ps
192 argument points to an object that contains an invalid conversion
193 The \fBmbrtowc()\fR function may fail if:
194 .sp
195 .ne 2
196 .na
197 \fB\fBEINVAL\fR\fR
198 .ad
199 .RS 10n
200 The \fIps\fR argument points to an object that contains an invalid conversion
201 state.
202 .IP \fBEILSEQ\fR
203 .RE
204 .sp
205 .ne 2
206 .na
207 \fB\fBEILSEQ\fR\fR
208 .ad
209 .RS 10n
210 Invalid character sequence is detected.
211 .RE
212 .SH ATTRIBUTES
213 .sp
214 .LP
215 See \fBattributes\fR(5) for descriptions of the following attributes:
216 .sp
217 .sp
218 .sp
219 .TS

```

```

166 box;
167 c | c
168 l | l .
169 ATTRIBUTE TYPE ATTRIBUTE VALUE
170 _
171 Interface Stability See below.
145 Interface Stability Standard
172 _
173 MT-Level See below.
147 MT-Level See NOTES below
174 .TE

176 .LP
177 The
178 .B mbrtowc()
179 function is Standard. The
180 .B mbrtowc_l()
181 function is Uncommitted.
182 .LP
183 If
184 .I ps
185 is a null pointer, these functions are Unsafe for use in
186 multithreaded applications. Otherwise they are MT-Safe.
187 .SH SEE ALSO
151 .sp
188 .LP
189 .BR mbsinit (3C),
190 .BR newlocale (3C),
191 .BR setlocale (3C),
192 .BR uselocale (3C),
193 .BR attributes (5),
194 .BR environ (5),
195 .BR standards (5)
153 \fBmbsinit\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5), \fBenviron\fR(5),
154 \fBstandards\fR(5)
155 .SH NOTES
156 .sp
157 .LP
158 If \fIps\fR is not a null pointer, \fBmbrtowc()\fR uses the \fBmbstate_t\fR
159 object pointed to by \fIps\fR and the function can be used safely in
160 multithreaded applications, as long as \fBsetlocale\fR(3C) is not being called
161 to change the locale. If \fIps\fR is a null pointer, \fBmbrtowc()\fR uses its
162 internal \fBmbstate_t\fR object and the function is Unsafe in multithreaded
163 applications.

```

\*\*\*\*\*

4564 Fri Jul 11 11:43:08 2014

new/usr/src/man/man3c/mbsinit.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions Co
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH MBSINIT 3C \"Jun 23, 2014\"
10 .TH MBSINIT 3C \"Jul 24, 2002\"
12 .SH NAME
13 mbsinit,mbsinit_1 \- determine conversion object status
12 mbsinit \- determine conversion object status
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <wchar.h>

19 \fBint\fR \fBmbsinit\fR(\fBconst mbstate_t *\fR\fR);
20 .fi
21 .LP
22 .nf
23 #include <wchar.h>
24 #include <xlocale.h>

26 \fBint\fR \fBmbsinit_1\fR(\fBconst mbstate_t *\fR\fR, \fBlocale_t\fR \fR)
27 .fi
28 .SH DESCRIPTION
29 .sp
29 .LP
30 If \fBps\fR is not a null pointer, the \fBmbsinit()\fR function determines
31 whether the object pointed to by \fBps\fR describes an initial conversion state.
32 .LP
33 The function \fBmbsinit_1()\fR behaves identically to \fBmbsinit()\fR except the
34 current locale is ignored and the locale \fR is used instead.
35 whether the object pointed to by \fBps\fR describes an initial conversion
36 state.
35 .SH RETURN VALUES
36 .sp
36 .LP
37 The \fBmbsinit()\fR function returns non-zero if \fBps\fR is a null pointer,
38 or if the pointed-to object describes an initial conversion state; otherwise,
39 it returns \fB0\fR.
33 .sp
40 .LP
41 If an \fBmbstate_t\fR object is altered by any of the functions described as
42 "restartable", and is then used with a different character sequence, or in the
43 other conversion direction, or with a different \fBLC_CTYPE\fR category setting
44 than an earlier function calls, the behavior is undefined. See
45 \fBviron\fR(5).
46 .SH ERRORS
47 .sp
47 .LP
48 No errors are defined.
49 .SH USAGE
45 .sp

```

```

50 .LP
51 The \fBmbstate_t\fR object is used to describe the current conversion state
52 from a particular character sequence to a wide-character sequence (or vice
53 versa) under the rules of a particular setting of the \fBLC_CTYPE\fR category
54 of the current locale.
51 .sp
55 .LP
56 The initial conversion state corresponds, for a conversion in either direction,
57 to the beginning of a new character sequence in the initial shift state. A
58 zero-valued \fBmbstate_t\fR object is at least one way to describe an initial
59 conversion state. A zero-valued \fBmbstate_t\fR object can be used to initiate
60 conversion involving any character sequence, in any \fBLC_CTYPE\fR category
61 setting.
62 .SH ATTRIBUTES
60 .sp
63 .LP
64 See \fBattributes\fR(5) for descriptions of the following attributes:
63 .sp

65 .sp
65 .TS
66 box;
67 c | c
68 l | l .
69 ATTRIBUTE TYPE ATTRIBUTE VALUE
70 -
71 Interface Stability See below.
72 Interface Stability Standard
72 -
73 MT-Level MT-Safe
74 MT-Level MT-Safe with exceptions
74 .TE

76 .LP
77 The
78 .B mbsinit()
79 function is Standard. The
80 .B mbsinit_1()
81 function is Uncommitted.
82 .SH SEE ALSO
78 .sp
83 .LP
84 \fBmbrlen\fR(3C), \fBmbrtowc\fR(3C), \fBmbsrtowcs\fR(3C),
85 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C),
86 \fBwrtomb\fR(3C), \fBwrtombs\fR(3C), \fBuselocale\fR(), \fBattributes\fR(5),
80 \fBmbrlen\fR(3C), \fBmbrtowc\fR(3C), \fBmbsrtowcs\fR(3C), \fBsetlocale\fR(3C),
81 \fBwrtomb\fR(3C), \fBwrtombs\fR(3C), \fBattributes\fR(5), \fBviron\fR(5),
87 \fBstandards\fR(5)
83 .SH NOTES
84 .sp
85 .LP
86 The \fBmbsinit()\fR function can be used safely in multithreaded applications,
87 as long as \fBsetlocale\fR(3C) is not being called to change the locale.

```



```

105 .ad
106 .RS 10n
107 The \fIps\fR argument points to an object that contains an invalid conversion
108 state.
109 .RE

111 .sp
112 .ne 2
113 .na
114 \fB\FBEILSEQ\fR\fR
115 .ad
116 .RS 10n
117 Invalid character sequence is detected.
118 .RE

119 .SH ATTRIBUTES
120 .sp
120 .LP
121 See \fBattributes\fR(5) for descriptions of the following attributes:
105 .sp

107 .sp
122 .TS
123 box;
124 c | c
125 l | l .
126 ATTRIBUTE TYPE ATTRIBUTE VALUE
127 -
128 Interface Stability See below.
114 Interface Stability Standard
129 -
130 MT-Level See below.
116 MT-Level See NOTES below
131 .TE

133 .LP
134 The
135 .B mbsnrtowcs()
136 and
137 .B mbsrtowcs()
138 functions are Standard. The
139 .B mbsnrtowcs_l()
140 and
141 .B mbsrtowcs_l()
142 functions are Uncommitted.
143 .LP
144 If \fIps\fR is not a null pointer, these functions use the \fBmbsstate_t\fR
145 object pointed to by \fIps\fR and can be used safely in
146 multithreaded applications, otherwise they use an
147 internal \fBmbsstate_t\fR object and are Unsafe in multithreaded
148 applications.
149 .SH SEE ALSO
120 .sp
150 .LP
151 \fBmbrtowc\fR(3C), \fBmbsinit\fR(3C), \fBnewlocale\fR(3C),
152 \fBsetlocale\fR(3C), \fBuselocale()\fR, \fBattributes\fR(5),
122 \fBmbrtowc\fR(3C), \fBmbsinit\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),
153 \fBenviron\fR(5), \fBstandards\fR(5)
124 .SH NOTES
125 .sp
126 .LP
127 If \fIps\fR is not a null pointer, \fBmbsrtowcs()\fR uses the \fBmbsstate_t\fR
128 object pointed to by \fIps\fR and the function can be used safely in
129 multithreaded applications, as long as \fBsetlocale\fR(3C) is not being called
130 to change the locale. If \fIps\fR is a null pointer, \fBmbsrtowcs()\fR uses its
131 internal \fBmbsstate_t\fR object and the function is Unsafe in multithreaded

```

```

132 applications.

```

\*\*\*\*\*

4702 Fri Jul 11 11:43:09 2014

new/usr/src/man/man3c/mbtowc.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  '\" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH MBTOWC 3C "Jun 23, 2014"
12 .TH MBTOWC 3C "Nov 1, 2003"
13 .SH NAME
14 mbtowc, mbtowc_l \- convert a character to a wide-character code
15 .LP
16 .nf
17 #include <stdlib.h>

19 \fBint\fR \fBmbtowc\fR(\fBwchar_t *restrict\fR \fBipwc\fR, \fBconst char *restric
20 .fi
21 .LP
22 .nf
23 #include <stdlib.h>
24 #include <xlocale.h>

26 \fBint\fR \fBmbtowc_l\fR(\fBwchar_t *restrict\fR \fBipwc\fR, \fBconst char *restr
27 \fBlocale_t\fR \fBiloc\fR);
28 .fi
29 .SH DESCRIPTION
30 .sp
31 .LP
32 If \fBis\fR is not a null pointer, \fBmbtowc()\fR determines the number of the
33 bytes that constitute the character pointed to by \fBis\fR. It then determines
34 the wide-character code for the value of type \fBwchar_t\fR that corresponds to
35 that character. (The value of the wide-character code corresponding to the null
36 byte is 0.) If the character is valid and \fBipwc\fR is not a null pointer,
37 \fBmbtowc()\fR stores the wide-character code in the object pointed to by
38 \fBipwc\fR.
39 .sp
40 .LP
41 A call with \fBis\fR as a null pointer causes this function to return \fB0\fR.
42 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
43 current locale. At most \fBin\fR bytes of the array pointed to by \fBis\fR will
44 be examined.
45 .LP
46 The function \fBmbtowc_l()\fR behaves identically to \fBmbtowc()\fR, except
47 instead of using the current locale, it uses the locale as specified by
48 \fBiloc\fR.
49 .SH RETURN VALUES
50 .sp
51 .LP
52 If \fBis\fR is a null pointer, these functions return \fB0\fR. If \fBis\fR is not
53 a null pointer, they return \fB0\fR (if \fBis\fR points to the null
54 character), \fB0\fR (if \fBis\fR points to a character that is not a valid character),
55 or \fB0\fR (if \fBis\fR points to a character that is not a valid character).
56 If \fBis\fR is a null pointer, \fBmbtowc()\fR returns \fB0\fR. If \fBis\fR is not
57 a null pointer, \fBmbtowc()\fR returns \fB0\fR (if \fBis\fR points to the null
58 character), the number of bytes that constitute the converted character (if the next

```

```

52 \fBin\fR or fewer bytes form a valid character), or \fB0\fR (if \fBis\fR points to a
53 character that is not a valid character).
54 .sp
55 .LP
56 In no case will the value returned be greater than \fBin\fR or the value of the
57 \fBMB_CUR_MAX\fR macro.
58 .SH ERRORS
59 .sp
60 .LP
61 The \fBmbtowc()\fR and \fBmbtowc_l()\fR functions may fail if:
62 .sp
63 .LP
64 The \fBmbtowc()\fR function may fail if:
65 .sp
66 .LP
67 .ne 2
68 .na
69 \fBEBEILSEQ\fR
70 .ad
71 .RS 10n
72 Invalid character sequence is detected.
73 .RE

74 .SH USAGE
75 .sp
76 .LP
77 The \fBmbtowc()\fR function can be used safely in multithreaded applications,
78 as long as \fBsetlocale(3C)\fR is not being called to change the locale.
79 .SH ATTRIBUTES
80 .sp
81 .LP
82 See \fBattributes(5)\fR for descriptions of the following attributes:
83 .sp

84 .sp
85 .TS
86 box;
87 c | c
88 l | l
89 l | l .
90 ATTRIBUTE TYPE ATTRIBUTE VALUE
91 -
92 CSI Enabled
93 -
94 Interface Stability See below.
95 Interface Stability Standard
96 -
97 MT-Level MT-Safe
98 MT-Level MT-Safe with exceptions
99 .TE

100 .LP
101 The
102 function \fBmbtowc()\fR
103 is Standard.
104 The
105 function \fBmbtowc_l()\fR
106 is Uncommitted.
107 .SH SEE ALSO
108 .sp
109 .LP
110 \fBmbrlen(3C)\fR, \fBmbsrtowcs(3C)\fR,
111 \fBnewlocale(3C)\fR, \fBsetlocale(3C)\fR, \fBuselocale(3C)\fR,
112 \fBwcstombs(3C)\fR, \fBwctomb(3C)\fR, \fBattributes(5)\fR, \fBstandards(5)\fR,
113 \fBmbrlen(3C)\fR, \fBmbsrtowcs(3C)\fR, \fBsetlocale(3C)\fR, \fBwcstombs(3C)\fR,
114 \fBwctomb(3C)\fR, \fBattributes(5)\fR, \fBstandards(5)\fR

```



```

*****
4310 Fri Jul 11 11:43:09 2014
new/usr/src/man/man3c/newlocale.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  '\" te
2  .\"
3  .\" This file and its contents are supplied under the terms of the
4  .\" Common Development and Distribution License (\"CDDL\"), version 1.0.
5  .\" You may only use this file in accordance with the terms of version
6  .\" 1.0 of the CDDL.
7  .\"
8  .\" A full copy of the text of the CDDL should have accompanied this
9  .\" source. A copy of the CDDL is also available via the Internet at
10 .\" http://www.illumos.org/license/CDDL.
11 .\"
12 .\"
13 .\" Copyright (c) 2014 Joyent, Inc. All rights reserved.
14 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 .\"
16 .TH NEWLOCALE 3C \"Jun 23, 2014\"
17 .SH NAME
18 duplocale, freelocale, newlocale \- create, duplicate, and destroy locale object
19 .SH SYNOPSIS
20 .LP
21 .nf
22 #include <locale.h>
23
24 \fBlocale_t\fR \fBnewlocale\fR(\fBint\fR \fBcategory_mask\fR, \fBconst char *\fR
25 \fBlocale_t\fR \fBibase\fR);
26 .fi
27 .LP
28 .nf
29 \fBlocale_t\fR \fBduplocale\fR(\fBlocale_t\fR \fBiloc\fR);
30 .fi
31 .LP
32 .nf
33 \fBvoid\fR \fBfreelocale\fR(\fBlocale_t\fR \fBiloc\fR);
34 .fi
35 .SH DESCRIPTION
36 .LP
37 These functions manipulate locale objects that can be used
38 .BR uselocale (3C)
39 and functions that take arguments of type
40 .BR locale_t .
41 .LP
42 The function
43 .B newlocale()
44 can be used to create a new locale object. It can also be used to modify an
45 existing locale object, the new locale object will be a replacement for the
46 modified locale object. To create a new locale, the argument
47 .I base
48 should be passed the special argument
49 .RB ( locale_t )0.
50 This will use a copy of the current global locale as a starting point. To modify
51 an existing locale object, it should be passed in as the argument
52 .IR base .
53 The new locale object is constructed by taking the categories specified in
54 .I category_mask
55 from the locale specified by the string
56 .IR locale ,
57 and filling in the remaining categories from the locale
58 .IR base .

```

```

59 When
60 .B newlocale()
61 returns, callers must no longer use
62 .IR base
63 and assume that
64 .BR freelocale (3C)
65 has been called on it. In addition to locales defined on the system, the
66 following three locales may always be passed in as the string
67 .IR locale :
68 .TP
69 \"C\"
70 Specifies the traditional UNIX system behavior.
71 .TP
72 \"POSIX\"
73 An alternate name fo the locale \"C\".
74 .TP
75 \"\"
76 Indicates that the locale should be processed based in the values in the
77 environment. See
78 .BR setlocale (3C)
79 and
80 .BR environ (5)
81 for more information.
82 .LP
83 The value of
84 .I category_mask
85 is a bitwise-inclusive or of the following macros which correspond to categories
86 as defined in
87 .BR locale (5)
88 and
89 .BR environ (5):
90 .TP
91 .B LC_CTYPE_MASK
92 Character classification and case conversion.
93 .TP
94 .B LC_NUMERIC_MASK
95 Numeric formatting.
96 .TP
97 .B LC_TIME_MASK
98 Date and time formatting.
99 .TP
100 .B LC_COLLATE_MASK
101 Collation order.
102 .TP
103 .B LC_MONETARY_MASK
104 Monetary formatting.
105 .TP
106 .B LC_MESSAGES_MASK
107 Formats of informative and diagnostic messages and interactive responses.
108 .TP
109 .B LC_ALL_MASK
110 Mask of all categories.
111 .LP
112 The function
113 .B duplocale()
114 duplicates the locale object specified by
115 .IR loc .
116 If the locale object passed is
117 .BR LC_GLOBAL_LOCALE ,
118 .B duplocale()
119 creates a copy of the current global locale as defined through calls to
120 .BR setlocale (3C).
121 .LP
122 The function
123 .B freelocale()
124 removes and releases all resources associated with the locale object

```

```

125 .IR loc .
126 Programs must not call
127 .B freelocale()
128 on
129 .BR LC_GLOBAL_LOCALE .
130 .SH RETURN VALUES
131 .LP
132 On success, the functions
133 .B newlocale()
134 and
135 .B duplocale()
136 return a new locale object that can be used with functions that take a
137 .BR locale_t .
138 Locale objects created this way should be freed with
139 .BR freelocale() .
140 On error, the functions
141 .B newlocale()
142 and
143 .B duplocale()
144 return
145 .BR (locale_t) 0
146 and
147 .B errno
148 is set to indicate the error. The
149 .B freelocale()
150 function does not set
151 .B errno.
152 .SH ERRORS
153 .LP
154 The
155 .B newlocale()
156 and
157 .B duplocale()
158 functions will fail if:
159 .TP
160 .B ENOMEM
161 Insufficient memory was available to create the locale object or to load the
162 requested locale data.
163 .LP
164 The
165 .B newlocale()
166 function will fail if:
167 .TP
168 .B EINVAL
169 An unknown bit is specified in
170 .IR category_mask .
171 .TP
172 .B ENOENT
173 Locale data was not found for a category specified in
174 .SH ATTRIBUTES
175 .TS
176 box;
177 c | c
178 1 | 1 .
179 ATTRIBUTE TYPE ATTRIBUTE VALUE
180 -
181 Interface Stability Standard
182 -
183 MT-Level Safe
184 .TE

186 .SH SEE ALSO
187 .BR locale (1),
188 .BR setlocale (3C),
189 .BR uselocale (3C),
190 .BR environ (5),

```

```

191 .BR locale (5)

```

```

*****
4081 Fri Jul 11 11:43:09 2014
new/usr/src/man/man3c/nl_langinfo.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright 1989 AT&T Copyright (c) 2002, Sun Microsystems, Inc. All Rights
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH NL_LANGINFO 3C "Jun 24, 2014"
10 .TH NL_LANGINFO 3C "Jul 24, 2002"
12 .SH NAME
13 nl_langinfo, nl_langinfo_l \- language information
12 nl_langinfo \- language information
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <langinfo.h>

19 \fBchar *\fR\fBnl_langinfo\fR(\fBnl_item\fR \fIitem\fR);
20 .fi
21 .LP
22 .nf
23 \fBchar *\fR\fBnl_langinfo_l\fR(\fBnl_item\fR \fIitem\fR, \fBlocale_t\fR \fIloc\
24 .fi

25 .SH DESCRIPTION
22 .sp
26 .LP
27 The \fBnl_langinfo()\fR function returns a pointer to a null-terminated string
28 containing information relevant to a particular language or cultural area
29 defined in the programs locale. The manifest constant names and values of
30 \fIitem\fR are defined by <\fBlanginfo.h\fR>. For example:
31 .IP
32 \fBnl_langinfo\fR(\fBABDAY_1\fR);
28 .sp
33 .LP
30 \fBnl_langinfo (ABDAY_1);\fR
31 .sp
32 .LP
34 would return a pointer to the string "\fBDim\fR" if the identified language was
35 French and a French locale was correctly installed; or "\fBSun\fR" if the
36 identified language was English.
37 .LP
38 The function \fBnl_langinfo_l()\fR behaves identically to \fBnl_langinfo()\fR,
39 except instead of acting in the current locale, it instead acts in the locale
40 specified by \fIloc\fR.
41 .SH RETURN VALUES
37 .sp
42 .LP
43 If the locale has not been set or is invalid, or if data for a
39 If \fBsetlocale\fR(3C) has not been called successfully, or if data for a
44 supported language is either not available, or if \fIitem\fR is not defined
45 therein, then these functions return a pointer to the corresponding string
46 in the C locale. In all locales, they return a pointer to an
41 therein, then \fBnl_langinfo()\fR returns a pointer to the corresponding string
42 in the C locale. In all locales, \fBnl_langinfo()\fR returns a pointer to an

```

```

47 empty string if \fIitem\fR contains an invalid setting.
44 .SH USAGE
45 .sp
46 .LP
47 The \fBnl_langinfo()\fR function can be used safely in multithreaded
48 applications, as long as \fBsetlocale\fR(3C) is not being called to change the
49 locale.
48 .SH ATTRIBUTES
51 .sp
49 .LP
50 See \fBattributes\fR(5) for descriptions of the following attributes:
54 .sp

56 .sp
51 .TS
52 box;
53 c | c
54 l | l .
55 ATTRIBUTE TYPE ATTRIBUTE VALUE
56 _
57 CSI Enabled
58 _
59 Interface Stability Standard
60 _
61 MT-Level MT-Safe
67 MT-Level MT-Safe with exceptions
62 .TE

64 .SH SEE ALSO
71 .sp
75 .LP
66 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
67 \fBlanginfo.h\fR(3HEAD), \fBnl_types.h\fR(3HEAD),
73 \fBsetlocale\fR(3C), \fBlanginfo.h\fR(3HEAD), \fBnl_types.h\fR(3HEAD),
68 \fBattributes\fR(5), \fBstandards\fR(5)
69 .SH WARNINGS
76 .sp
70 .LP
71 The array pointed to by the return value should not be modified by the program.
72 Subsequent calls to these functions may overwrite the array.
79 Subsequent calls to \fBnl_langinfo()\fR may overwrite the array.

```

```

*****
4626 Fri Jul 11 11:43:09 2014
new/usr/src/man/man3c/strcoll.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 2003, Sun Microsystems, Inc. All Rights Reserved.
4  .\" Copyright 1989 AT&T
5  .\" Portions Copyright (c) 2001, the Institute of Electrical and Electronics Eng
6  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
7  .\" http://www.opengroup.org/bookstore/.
8  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
9  .\" This notice shall appear on any product containing this material.
10 .\" The contents of this file are subject to the terms of the Common Development
11 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
12 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
13 .TH STRCOLL 3C "Jun 23, 2014"
14 .TH STRCOLL 3C "Nov 1, 2003"
15 .SH NAME
16 strcoll, strcoll_l \- string collation
17 .LP
18 .nf
19 #include <string.h>

21 \fBint\fR \fBstrcoll\fR(\fBconst char *\fR\fIs1\fR, \fBconst char *\fR\fIs2\fR);
22 .fi
23 .LP
24 .nf
25 \fBint\fR \fBstrcoll_l\fR(\fBconst char *\fR\fIs1\fR, \fBconst char *\fR\fIs2\fR
26 .fi

28 .SH DESCRIPTION
29 .sp
30 Both \fBstrcoll()\fR and \fBstrxfrm()\fR(3C) provide for locale-specific string
31 sorting. \fBstrcoll()\fR is intended for applications in which the number of
32 comparisons per string is small. When strings are to be compared a number of
33 times, \fBstrxfrm()\fR(3C) is a more appropriate function because the
34 transformation process occurs only once.
35 .sp
36 .LP
37 The \fBstrcoll_l()\fR function behaves
38 identically to \fBstrcoll()\fR, except instead of operating in the current
39 locale, it operates in the locale specified by \fBlocale\fR.
40 .LP
41 The \fBstrcoll()\fR function does not change the setting of \fBerrno\fR if
42 successful.
43 .sp
44 .LP
45 Since no return value is reserved to indicate an error, an application wishing
46 to check for error situations should set \fBerrno\fR to 0, then call
47 \fBstrcoll()\fR, then check \fBerrno\fR.
48 .SH RETURN VALUES
49 .sp
50 .LP
51 Upon successful completion, \fBstrcoll()\fR returns an integer greater than,
52 equal to, or less than zero in direct correlation to whether string \fIs1\fR is
53 greater than, equal to, or less than the string \fIs2\fR. The comparison is
54 based on strings interpreted as appropriate to the locale
55 based on strings interpreted as appropriate to the program's locale for

```

```

52 category \fBLC_COLLATE\fR (see \fBsetlocale()\fR(3C)).
53 .sp
54 .LP
55 On error, \fBstrcoll()\fR may set \fBerrno\fR, but no return value is reserved
56 to indicate an error.
57 .SH ERRORS
58 .sp
59 .LP
60 The \fBstrcoll()\fR and \fBstrcoll_l()\fR functions may fail if:
61 .sp
62 .ne 2
63 .na
64 \fBEBINVAL\fR
65 .ad
66 .RS 10n
67 The \fIs1\fR or \fIs2\fR arguments contain characters outside the domain of the
68 collating sequence.
69 .RE

70 .SH FILES
71 .IP \fB/usr/lib/locale/\fR\fIlocale\fR/\fB/LC_COLLATE/*\fR
72 collation database for \fIlocale\fR
73 .sp
74 .ne 2
75 .na
76 \fB/usr/lib/locale/\fR\fIlocale\fR/\fR/\fIlocale\fR.so.*\fR
77 .ad
78 .sp .6
79 .RS 4n
80 \fBLC_COLLATE\fR database for \fIlocale\fR
81 .RE

82 .SH ATTRIBUTES
83 .sp
84 .LP
85 See \fBattributes\fR(5) for descriptions of the following attributes:
86 .sp

87 .sp
88 .TS
89 box;
90 c | c
91 l | l .
92 ATTRIBUTE TYPE ATTRIBUTE VALUE
93 _
94 CSI Enabled
95 _
96 Interface Stability Standard
97 _
98 MT-Level MT-Safe
99 MT-Level MT-Safe with exceptions
100 .TE

101 .sp
102 .LP
103 The \fBstrcoll()\fR function can be used safely in multithreaded applications,
104 as long as \fBsetlocale()\fR(3C) is not being called to change the locale.
105 .SH SEE ALSO
106 .sp
107 .LP
108 \fBlocaledef()\fR(1), \fBnewlocale()\fR(3C), \fBsetlocale()\fR(3C), \fBstring()\fR(3C),
109 \fBstrxfrm()\fR(3C), \fBuselocale()\fR(3C),
110 \fBlocaledef()\fR(1), \fBsetlocale()\fR(3C), \fBstring()\fR(3C), \fBstrxfrm()\fR(3C),
111 \fBwsxfrm()\fR(3C), \fBattributes()\fR(5), \fBenviron()\fR(5), \fBstandards()\fR(5)

```



```
125 the terminating null character) fits in
126 .I maxsize
127 bytes, then the number of
128 bytes placed in the buffer (excluding the terminating null character) is
129 returned.
130 .LP
131 If the result of expansion exceeds
132 .I maxsize
133 bytes, then the value \mil is returned, and
134 .I errno
135 is set to
136 .BR E2BIG .
137 .SH NOTES
138 The result of formatting a value that is not a rational number (e.g. +NaN) is
139 unspecified.
140 .SH ATTRIBUTES
141 .LP
142 See
143 .BR attributes (5)
144 for descriptions of the following attributes:
145 .TS
146 box;
147 c | c
148 l | l .
149 ATTRIBUTE TYPE ATTRIBUTE VALUE
150 -
151 CSI Enabled
152 -
153 Interface Stability Standard
154 -
155 MT-Level MT-Safe
156 .TE
158 .SH SEE ALSO
159 .BR setlocale (3C),
160 .BR uselocale (3C),
161 .BR locale (3HEAD),
162 .BR attributes (5),
163 .BR standards (5)
```

```

*****
16204 Fri Jul 11 11:43:09 2014
new/usr/src/man/man3c/strftime.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 \" te
2 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3 .\" Copyright (c) 2014 Gary Mills
4 .\" Copyright (c) 2008, Sun Microsystems, Inc. All Rights Reserved.
5 .\" Copyright 1989 AT&T
6 .\" Portions Copyright (c) 1992, X/Open Company Limited. All Rights Reserved.
7 .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
8 .\" http://www.opengroup.org/bookstore/.
9 .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
10 .\" This notice shall appear on any product containing this material.
11 .\" The contents of this file are subject to the terms of the Common Development
12 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
13 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
14 .TH STRFTIME 3C \"Jun 24, 2014\"
13 .TH STRFTIME 3C \"Jan 3, 2014\"
15 .SH NAME
16 strftime, strftime_l cftime, ascftime \- convert date and time to string
15 strftime, cftime, ascftime \- convert date and time to string
17 .SH SYNOPSIS
18 .LP
19 .nf
20 #include <time.h>

22 \fBsize_t\fR \fBstrftime\fR(\fBchar *restrict\fR \fRis\fR, \fBsize_t\fR \fBmaxsiz
23 \fBconst char *restrict\fR \fBformat\fR,
24 \fBconst struct tm *restrict\fR \fBt\fR);
25 .fi

26 .LP
27 .nf
28 \fBsize_t\fR \fBstrftime_l\fR(\fBchar *restrict\fR \fRis\fR, \fBsize_t\fR \fBmaxs
29 \fBconst char *restrict\fR \fBformat\fR,
30 \fBconst struct tm *restrict\fR \fBt\fR, \fBlocale_t\fR \fBilloc\fR);
31 .fi
32 .LP
33 .nf
34 \fBint\fR \fBcftime\fR(\fBchar *\fR \fRis\fR, \fBchar *\fR \fBformat\fR, \fBconst t
35 .fi

36 .LP
37 .nf
38 \fBint\fR \fBascftime\fR(\fBchar *\fR \fRis\fR, \fBconst char *\fR \fBformat\fR,
39 \fBconst struct tm *\fR \fBt\fR);
40 .fi

42 .SH DESCRIPTION
43 .sp
44 .LP
45 The \fBstrftime()\fR, \fBstrftime_l()\fR, \fBascftime()\fR, and \fBcftime()\fR
46 functions place
47 The \fBstrftime()\fR, \fBascftime()\fR, and \fBcftime()\fR functions place
48 bytes into the array pointed to by \fRis\fR as controlled by the string pointed
49 to by \fBformat\fR. The \fBformat\fR string consists of zero or more conversion
49 specifications and ordinary characters. A conversion specification consists of
50 a '\fB%\fR' (percent) character and one or two terminating conversion
51 characters that determine the conversion specification's behavior. All
52 ordinary characters (including the terminating null byte) are copied unchanged
53 into the array pointed to by \fRis\fR. If copying takes place between objects

```

```

54 that overlap, the behavior is undefined. For \fBstrftime()\fR, no more than
55 \fBmaxsize\fR bytes are placed into the array. The \fBstrftime_l()\fR function
56 behaves identically to \fBstrftime()\fR function, but instead of operating in
57 the current locale, it operates in the locale specified by \fBilloc\fR.
49 \fBmaxsize\fR bytes are placed into the array.
50 .sp
51 .LP
52 If \fBformat\fR is \fB(char *)0\fR, then the locale's default format is used.
60 For \fBstrftime()\fR the default format is the same as \fB%c\fR; for
61 \fBcftime()\fR and \fBascftime()\fR the default format is the same as \fB%+\fR.
62 \fBcftime()\fR and \fBascftime()\fR first try to use the value of the
63 environment variable \fBCFTIME\fR, and if that is undefined or empty, the
64 default format is used.
58 .sp
65 .LP
66 Each conversion specification is replaced by appropriate characters as
67 described in the following list. The appropriate characters are determined by
68 the \fBLC_TIME\fR category of the program's locale and by the values contained
69 in the structure pointed to by \fBt\fR for \fBstrftime()\fR and
70 \fBascftime()\fR, and by the time represented by \fBt\fR for
71 \fBcftime()\fR.
72 .sp
73 .ne 2
74 .na
75 \fB\F%\fR
76 .ad
77 .RS 6n
78 Same as \fB%\fR.
79 .RE

81 .sp
82 .ne 2
83 .na
84 \fB\Fa\fR
85 .ad
86 .RS 6n
87 Locale's abbreviated weekday name.
88 .RE

90 .sp
91 .ne 2
92 .na
93 \fB\F%A\fR
94 .ad
95 .RS 6n
96 Locale's full weekday name.
97 .RE

99 .sp
100 .ne 2
101 .na
102 \fB\Fb\fR
103 .ad
104 .RS 6n
105 Locale's abbreviated month name.
106 .RE

108 .sp
109 .ne 2
110 .na
111 \fB\F%B\fR
112 .ad
113 .RS 6n
114 Locale's full month name.
115 .RE

```

```

117 .sp
118 .ne 2
119 .na
120 \fB\fB%c\fR\fR
121 .ad
122 .RS 6n
123 Locale's appropriate date and time representation.
124 In the C locale, this format is:
125 .sp
126 .in +2
127 .nf
128 %a %b %e %H:%M:%S %Y
129 .fi
130 .in -2
131 .sp
132 Other locales may have different locale-specific formats.
133 .RE

135 .sp
136 .ne 2
137 .na
138 \fB\fB%C\fR\fR
139 .ad
140 .RS 6n
141 Century number (the year divided by 100 and truncated to an integer as a
142 decimal number [01,99]).
143 .RE

145 .sp
146 .ne 2
147 .na
148 \fB\fB%d\fR\fR
149 .ad
150 .RS 6n
151 Day of month [01,31].
152 .RE

154 .sp
155 .ne 2
156 .na
157 \fB\fB%D\fR\fR
158 .ad
159 .RS 6n
160 Date as \fB%m\fR/\fB%d\fR/\fB%y\fR.
161 .RE

163 .sp
164 .ne 2
165 .na
166 \fB\fB%e\fR\fR
167 .ad
168 .RS 6n
169 Day of month [1,31]; single digits are preceded by a space.
170 .RE

172 .sp
173 .ne 2
174 .na
175 \fB\fB%F\fR\fR
176 .ad
177 .RS 6n
178 Equivalent to \fB%Y\fR-\fB%m\fR-\fB%d\fR (the ISO 8601:2000 standard date
179 format).
180 .RE

182 .sp

```

```

183 .ne 2
184 .na
185 \fB\fB%g\fR\fR
186 .ad
187 .RS 6n
188 Week-based year within century [00,99].
189 .RE

191 .sp
192 .ne 2
193 .na
194 \fB\fB%G\fR\fR
195 .ad
196 .RS 6n
197 Week-based year, including the century [0000,9999].
198 .RE

200 .sp
201 .ne 2
202 .na
203 \fB\fB%h\fR\fR
204 .ad
205 .RS 6n
206 Locale's abbreviated month name.
207 .RE

209 .sp
210 .ne 2
211 .na
212 \fB\fB%H\fR\fR
213 .ad
214 .RS 6n
215 Hour (24-hour clock) [00,23].
216 .RE

218 .sp
219 .ne 2
220 .na
221 \fB\fB%I\fR\fR
222 .ad
223 .RS 6n
224 Hour (12-hour clock) [01,12].
225 .RE

227 .sp
228 .ne 2
229 .na
230 \fB\fB%j\fR\fR
231 .ad
232 .RS 6n
233 Day number of year [001,366].
234 .RE

236 .sp
237 .ne 2
238 .na
239 \fB\fB%k\fR\fR
240 .ad
241 .RS 6n
242 Hour (24-hour clock) [0,23]; single digits are preceded by a space.
243 .RE

245 .sp
246 .ne 2
247 .na
248 \fB\fB%l\fR\fR

```



```

249 .ad
250 .RS 6n
251 Hour (12-hour clock) [1,12]; single digits are preceded by a space.
252 .RE

254 .sp
255 .ne 2
256 .na
257 \fB\fB%m\fR\fR
258 .ad
259 .RS 6n
260 Month number [01,12].
261 .RE

263 .sp
264 .ne 2
265 .na
266 \fB\fB%M\fR\fR
267 .ad
268 .RS 6n
269 Minute [00,59].
270 .RE

272 .sp
273 .ne 2
274 .na
275 \fB\fB%n\fR\fR
276 .ad
277 .RS 6n
278 Insert a NEWLINE.
279 .RE

281 .sp
282 .ne 2
283 .na
284 \fB\fB%p\fR\fR
285 .ad
286 .RS 6n
287 Locale's equivalent of either a.m. or p.m.
288 .RE

290 .sp
291 .ne 2
292 .na
293 \fB\fB%r\fR\fR
294 .ad
295 .RS 6n
296 Appropriate time representation in 12-hour clock format with \fB%p\fR.
297 .RE

299 .sp
300 .ne 2
301 .na
302 \fB\fB%R\fR\fR
303 .ad
304 .RS 6n
305 Time as \fB%H\fR:\fB%M\fR.
306 .RE

308 .sp
309 .ne 2
310 .na
311 \fB\fB%s\fR\fR
312 .ad
313 .RS 6n
314 Seconds since 00:00:00 UTC, January 1, 1970.

```

```

315 .RE

317 .sp
318 .ne 2
319 .na
320 \fB\fB%S\fR\fR
321 .ad
322 .RS 6n
323 Seconds [00,60]; the range of values is [00,60] rather than [00,59] to allow
324 for the occasional leap second.
325 .RE

327 .sp
328 .ne 2
329 .na
330 \fB\fB%t\fR\fR
331 .ad
332 .RS 6n
333 Insert a TAB.
334 .RE

336 .sp
337 .ne 2
338 .na
339 \fB\fB%T\fR\fR
340 .ad
341 .RS 6n
342 Time as \fB%H\fR:\fB%M\fR:\fB%S\fR.
343 .RE

345 .sp
346 .ne 2
347 .na
348 \fB\fB%u\fR\fR
349 .ad
350 .RS 6n
351 Weekday as a decimal number [1,7], with 1 representing Monday. See \fBNOTES\fR
352 below.
353 .RE

355 .sp
356 .ne 2
357 .na
358 \fB\fB%U\fR\fR
359 .ad
360 .RS 6n
361 Week number of year as a decimal number [00,53], with Sunday as the first day
362 of week 1.
363 .RE

365 .sp
366 .ne 2
367 .na
368 \fB\fB%v\fR\fR
369 .ad
370 .RS 6n
371 Date as \fB%e\fR-\fB%b\fR-\fB%Y\fR.
372 .RE

374 .sp
375 .ne 2
376 .na
377 \fB\fB%V\fR\fR
378 .ad
379 .RS 6n
380 The ISO 8601 week number as a decimal number [01,53]. In the ISO 8601

```

381 week-based system, weeks begin on a Monday and week 1 of the year is the week  
 382 that includes both January 4th and the first Thursday of the year. If the  
 383 first Monday of January is the 2nd, 3rd, or 4th, the preceding days are part of  
 384 the last week of the preceding year. See \fBNOTES\fR below.  
 385 .RE

387 .sp  
 388 .ne 2  
 389 .na  
 390 \fB\fB%w\fR\fR  
 391 .ad  
 392 .RS 6n  
 393 Weekday as a decimal number [0,6], with 0 representing Sunday.  
 394 .RE

396 .sp  
 397 .ne 2  
 398 .na  
 399 \fB\fB%W\fR\fR  
 400 .ad  
 401 .RS 6n  
 402 Week number of year as a decimal number [00,53], with Monday as the first day  
 403 of week 1.  
 404 .RE

406 .sp  
 407 .ne 2  
 408 .na  
 409 \fB\fB%x\fR\fR  
 410 .ad  
 411 .RS 6n  
 412 Locale's appropriate date representation.  
 413 .RE

415 .sp  
 416 .ne 2  
 417 .na  
 418 \fB\fB%X\fR\fR  
 419 .ad  
 420 .RS 6n  
 421 Locale's appropriate time representation.  
 422 .RE

424 .sp  
 425 .ne 2  
 426 .na  
 427 \fB\fB%y\fR\fR  
 428 .ad  
 429 .RS 6n  
 430 Year within century [00,99].  
 431 .RE

433 .sp  
 434 .ne 2  
 435 .na  
 436 \fB\fB%Y\fR\fR  
 437 .ad  
 438 .RS 6n  
 439 Year, including the century (for example 1993).  
 440 .RE

442 .sp  
 443 .ne 2  
 444 .na  
 445 \fB\fB%z\fR\fR  
 446 .ad

447 .RS 6n  
 448 Replaced by offset from UTC in ISO 8601:2000 standard format (\fB+h:mm\fR or  
 449 \fB-h:mm\fR), or by no characters if no time zone is determinable. For example,  
 450 "-0430" means 4 hours 30 minutes behind UTC (west of Greenwich). If  
 451 \fBtm\_isdst\fR is zero, the standard time offset is used. If \fBtm\_isdst\fR is  
 452 greater than zero, the daylight savings time offset is used. If \fBtm\_isdst\fR  
 453 is negative, no characters are returned.  
 454 .RE

456 .sp  
 457 .ne 2  
 458 .na  
 459 \fB\fB%Z\fR\fR  
 460 .ad  
 461 .RS 6n  
 462 Time zone name or abbreviation, or no bytes if no time zone information exists.  
 463 .RE

465 .sp  
 466 .ne 2  
 467 .na  
 468 \fB\fB%+\fR\fR  
 469 .ad  
 470 .RS 6n  
 471 Locale's date and time representation as produced by \fBdate\fR(1).  
 472 .RE

468 .sp  
 474 .LP  
 475 If a conversion specification does not correspond to any of the above or to any  
 476 of the modified conversion specifications listed below, the behavior is  
 477 undefined and \fB0\fR is returned.

473 .sp  
 478 .LP  
 479 The difference between \fB%U\fR and \fB%W\fR (and also between modified  
 480 conversion specifications \fB%OU\fR and \fB%OW\fR) lies in which day is counted  
 481 as the first of the week. Week number 1 is the first week in January starting  
 482 with a Sunday for \fB%U\fR or a Monday for \fB%W\fR. Week number 0 contains  
 483 those days before the first Sunday or Monday in January for \fB%U\fR and  
 484 \fB%W\fR, respectively.

485 .SS "Modified Conversion Specifications"

482 .sp  
 486 .LP  
 487 Some conversion specifications can be modified by the \fB%E\fR and \fB%O\fR  
 488 modifiers to indicate that an alternate format or specification should be used  
 489 rather than the one normally used by the unmodified conversion specification.  
 490 If the alternate format or specification does not exist in the current locale,  
 491 the behavior will be as if the unmodified specification were used.

492 .sp  
 493 .ne 2  
 494 .na  
 495 \fB\fB%Ec\fR\fR  
 496 .ad  
 497 .RS 7n  
 498 Locale's alternate appropriate date and time representation.  
 499 .RE

501 .sp  
 502 .ne 2  
 503 .na  
 504 \fB\fB%EC\fR\fR  
 505 .ad  
 506 .RS 7n  
 507 Name of the base year (period) in the locale's alternate representation.  
 508 .RE

```

510 .sp
511 .ne 2
512 .na
513 \fB\fB%Eg\fR\fR
514 .ad
515 .RS 7n
516 Offset from \fB%EC\fR of the week-based year in the locale's alternative
517 representation.
518 .RE

520 .sp
521 .ne 2
522 .na
523 \fB\fB%EG\fR\fR
524 .ad
525 .RS 7n
526 Full alternative representation of the week-based year.
527 .RE

529 .sp
530 .ne 2
531 .na
532 \fB\fB%Ex\fR\fR
533 .ad
534 .RS 7n
535 Locale's alternate date representation.
536 .RE

538 .sp
539 .ne 2
540 .na
541 \fB\fB%EX\fR\fR
542 .ad
543 .RS 7n
544 Locale's alternate time representation.
545 .RE

547 .sp
548 .ne 2
549 .na
550 \fB\fB%Ey\fR\fR
551 .ad
552 .RS 7n
553 Offset from \fB%EC\fR (year only) in the locale's alternate representation.
554 .RE

556 .sp
557 .ne 2
558 .na
559 \fB\fB%EY\fR\fR
560 .ad
561 .RS 7n
562 Full alternate year representation.
563 .RE

565 .sp
566 .ne 2
567 .na
568 \fB\fB%Od\fR\fR
569 .ad
570 .RS 7n
571 Day of the month using the locale's alternate numeric symbols.
572 .RE

574 .sp
575 .ne 2

```

```

576 .na
577 \fB\fB%Oe\fR\fR
578 .ad
579 .RS 7n
580 Same as \fB%Od\fR.
581 .RE

583 .sp
584 .ne 2
585 .na
586 \fB\fB%Og\fR\fR
587 .ad
588 .RS 7n
589 Week-based year (offset from \fB%C\fR) in the locale's alternate representation
590 and using the locale's alternate numeric symbols.
591 .RE

593 .sp
594 .ne 2
595 .na
596 \fB\fB%OH\fR\fR
597 .ad
598 .RS 7n
599 Hour (24-hour clock) using the locale's alternate numeric symbols.
600 .RE

602 .sp
603 .ne 2
604 .na
605 \fB\fB%OI\fR\fR
606 .ad
607 .RS 7n
608 Hour (12-hour clock) using the locale's alternate numeric symbols.
609 .RE

611 .sp
612 .ne 2
613 .na
614 \fB\fB%Om\fR\fR
615 .ad
616 .RS 7n
617 Month using the locale's alternate numeric symbols.
618 .RE

620 .sp
621 .ne 2
622 .na
623 \fB\fB%OM\fR\fR
624 .ad
625 .RS 7n
626 Minutes using the locale's alternate numeric symbols.
627 .RE

629 .sp
630 .ne 2
631 .na
632 \fB\fB%OS\fR\fR
633 .ad
634 .RS 7n
635 Seconds using the locale's alternate numeric symbols.
636 .RE

638 .sp
639 .ne 2
640 .na
641 \fB\fB%Ou\fR\fR

```

```

642 .ad
643 .RS 7n
644 Weekday as a number in the locale's alternate numeric symbols.
645 .RE

647 .sp
648 .ne 2
649 .na
650 \fB\fB%OU\fR\fR
651 .ad
652 .RS 7n
653 Week number of the year (Sunday as the first day of the week) using the
654 locale's alternate numeric symbols.
655 .RE

657 .sp
658 .ne 2
659 .na
660 \fB\fB%Ow\fR\fR
661 .ad
662 .RS 7n
663 Number of the weekday (Sunday=0) using the locale's alternate numeric symbols.
664 .RE

666 .sp
667 .ne 2
668 .na
669 \fB\fB%OW\fR\fR
670 .ad
671 .RS 7n
672 Week number of the year (Monday as the first day of the week) using the
673 locale's alternate numeric symbols.
674 .RE

676 .sp
677 .ne 2
678 .na
679 \fB\fB%Oy\fR\fR
680 .ad
681 .RS 7n
682 Year (offset from \fB%C\fR) in the locale's alternate representation and using
683 the locale's alternate numeric symbols.
684 .RE

686 .SS "Selecting the Output Language"
687 .LP
688 These routines produce output that is formatted according to the \fBLC_TIME\fR
689 locale category. They use either the current locale, or in the case of
690 \fBstrptime_l()\fR, the locale supplied by \fIloc\fR.
691 By default, the output of \fBstrptime()\fR, \fBcftime()\fR, and
692 \fBascftime()\fR appear in U.S. English. The user can request that the output
693 of \fBstrptime()\fR, \fBcftime()\fR, or \fBascftime()\fR be in a specific
694 language by setting the \fBLC_TIME\fR category using \fBsetlocale()\fR.
695 "Time Zone"
696 .SS "Time Zone"
697 .LP
698 Local time zone information is used as though \fBtzset\fR(3C) were called.
699 .SH RETURN VALUES
700 .sp
701 .LP
702 These functions return the
703 The \fBstrptime()\fR, \fBcftime()\fR, and \fBascftime()\fR functions return the
704 number of characters placed into the array pointed to by \fIis\fR, not including
705 the terminating null character. If the total number of resulting characters
706 including the terminating null character is more than \fImaxsize\fR,

```

```

700 \fBstrptime()\fR returns \fB0\fR and the contents of the array are
701 indeterminate.
702 .SH EXAMPLES
703 .LP
704 \fBExample 1\fR An example of the \fBstrptime()\fR function.
705 .sp
706 .LP
707 The following example illustrates the use of \fBstrptime()\fR for the
708 \fBPOSIX\fR locale. It shows what the string in \fIstr\fR would look like if
709 the structure pointed to by \fItmptr\fR contains the values corresponding to
710 Thursday, August 28, 1986 at 12:44:36.
711 .sp
712 .in +2
713 .nf
714 .IP
715 \fBstrptime()\fR(\fIstr\fR, \fIstrsize\fR, "%A %b %d %j", \fItmptr\fR);
716 \fBstrptime(str, strsize, "%A %b %d %j", tmptr)\fR
717 .fi
718 .in -2

720 .sp
721 .LP
722 This results in \fIstr\fR containing "Thursday Aug 28 240".

724 .SH ATTRIBUTES
725 .sp
726 .LP
727 See \fBattributes\fR(5) for descriptions of the following attributes:
728 .sp

730 .sp
731 .TS
732 box;
733 c | c
734 l | l .
735 ATTRIBUTE TYPE ATTRIBUTE VALUE
736 -
737 CSI Enabled
738 -
739 Interface Stability See below.
740 Interface Stability Committed
741 -
742 MT-Level MT-Safe
743 -
744 Standard See below.
745 .TE

746 .sp
747 .LP
748 The \fBstrptime()\fR and \fBstrptime_l()\fR functions are Standard.
749 \fBcftime()\fR and \fBascftime()\fR functions are Committed.
750 .LP
751 For \fBstrptime()\fR and \fBstrptime_l()\fR, see \fBstandards\fR(5).
752 For \fBstrptime()\fR, see \fBstandards\fR(5).
753 .SH SEE ALSO
754 .sp
755 .LP
756 \fBdate\fR(1), \fBctime\fR(3C), \fBmktime\fR(3C),
757 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBstrptime\fR(3C), \fBtzset\fR(3C),
758 \fBuselocale\fR(3C), \fBTIMEZONE\fR(4), \fBzoneinfo\fR(4),
759 \fBdate\fR(1), \fBctime\fR(3C), \fBmktime\fR(3C), \fBsetlocale\fR(3C),
760 \fBstrptime\fR(3C), \fBtzset\fR(3C), \fBTIMEZONE\fR(4), \fBzoneinfo\fR(4),
761 \fBattributes\fR(5), \fBenviron\fR(5), \fBstandards\fR(5)
762 .SH NOTES
763 .sp

```

746 .LP  
747 The conversion specification for `\fB%V\fR` was changed in the Solaris 7 release.  
748 This change was based on the public review draft of the ISO C9x standard at  
749 that time. Previously, the specification stated that if the week containing 1  
750 January had fewer than four days in the new year, it became week 53 of the  
751 previous year. The ISO C9x standard committee subsequently recognized that that  
752 specification had been incorrect.  
764 .sp  
753 .LP  
754 The conversion specifications for `\fB%g\fR`, `\fB%G\fR`, `\fB%Eg\fR`, `\fB%EG\fR`, and  
755 `\fB%Og\fR` were added in the Solaris 7 release. This change was based on the  
756 **public review draft of the ISO C9x standard at that time. The `\fB%g\fR` and**  
757 **`\fB%G\fR` specifications were adopted in the formal standard. The other two**  
758 **were not, and should not be used in portable applications.**  
768 *public review draft of the ISO C9x standard at that time. These specifications*  
769 *are evolving. If the ISO C9x standard is finalized with a different*  
770 *conclusion, these specifications will change to conform to the ISO C9x standard*  
771 *decision.*  
772 .sp  
759 .LP  
760 The conversion specification for `\fB%u\fR` was changed in the Solaris 8 release.  
761 This change was based on the XPG4 specification.  
776 .sp  
762 .LP  
763 If using the `\fB%Z\fR` specifier and `\fBzoneinfo\fR` timezones and if the input  
764 date is outside the range 20:45:52 UTC, December 13, 1901 to 03:14:07 UTC,  
765 January 19, 2038, the timezone name may not be correct.  
781 .sp  
766 .LP  
767 The conversion specification for `\fB%+\fR` was added in illumos.  
768 It is not part of any standard, although it is available on a number  
769 of other platforms.  
770 Its use is discouraged for conforming applications.

\*\*\*\*\*

21323 Fri Jul 11 11:43:09 2014

new/usr/src/man/man3c/string.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\ Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\ Copyright (c) 2008, Sun Microsystems, Inc. All Rights Reserved.
4  .\ Copyright 1989 AT&T
5  .\ Portions Copyright (c) 1994 Man-cgi 1.15, Panagiotis Christias (christia@sof
6  .\ Portions Copyright (c) 1996-2008 Modified for NetBSD by Kimmo Suominen (kimm
7  .\ Portions Copyright (c) 1992, X/Open Company Limited. All Rights Reserved.
8  .\ Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
9  .\ http://www.opengroup.org/bookstore/.
10 .\ The Institute of Electrical and Electronics Engineers and The Open Group, ha
11 .\ This notice shall appear on any product containing this material.
12 .\ The contents of this file are subject to the terms of the Common Development
13 .\ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
14 .\ When distributing Covered Code, include this CDDL HEADER in each file and in
15 .TH STRING 3C "Jun 21, 2013"
16 .SH NAME
17 string, strcasecmp, strcasecmp_l, strncasecmp, strncasecmp_l, strcat, strncat,
18 strlcat, strchr, strrchr,
19 string, strcasecmp, strncasecmp, strcat, strncat, strlcat, strchr, strrchr,
20 strcmp, strncmp, strcpy, strncpy, strlcpy, strcspn, strspn, strdup, strlen,
21 strnlen, strpbrk, strsep, strstr, strtok, strtok_r - string operations
22 .SH SYNOPSIS
23 .LP
24 #include <strings.h>
25
26 \fBint\fR \fBstrncasecmp\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
27 .fi
28 .LP
29 .nf
30 \fBint\fR \fBstrncasecmp_l\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
31 .fi
32 .LP
33 .nf
34 \fBint\fR \fBstrncasecmp_l\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
35 .fi
36 .LP
37 .nf
38 \fBint\fR \fBstrncasecmp_l\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI1
39 .fi
40 .LP
41 .nf
42 #include <string.h>
43
44 \fBchar *\fR \fBstrncat\fR(\fBchar *restrict\fR \fI1\fR, \fBconst char *restrict\fR
45 .fi
46 .LP
47 .nf
48 \fBchar *\fR \fBstrncat\fR(\fBchar *restrict\fR \fI1\fR, \fBconst char *restrict
49 .fi
50 .LP
51 .nf
52 \fBsize_t\fR \fBstrlcat\fR(\fBchar *\fR\fI1\fR, \fBconst char *\fR\fIsrc\fR, \

```

```

53 .fi
54 .LP
55 .nf
56 \fBchar *\fR \fBstrchr\fR(\fBconst char *\fR\fI1\fR, \fBint\fR \fIc\fR);
57 .fi
58 .LP
59 .nf
60 \fBchar *\fR \fBstrrchr\fR(\fBconst char *\fR\fI1\fR, \fBint\fR \fIc\fR);
61 .fi
62 .LP
63 .nf
64 \fBint\fR \fBstrcmp\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR);
65 .fi
66 .LP
67 .nf
68 \fBint\fR \fBstrncmp\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR,
69 .fi
70 .LP
71 .nf
72 \fBchar *\fR \fBstrcpy\fR(\fBchar *restrict\fR \fI1\fR, \fBconst char *restrict\fR
73 .fi
74 .LP
75 .nf
76 \fBchar *\fR \fBstrncpy\fR(\fBchar *restrict\fR \fI1\fR, \fBconst char *restrict
77 .fi
78 .LP
79 .nf
80 \fBsize_t\fR \fBstrlcpy\fR(\fBchar *\fR\fI1\fR, \fBconst char *\fR\fIsrc\fR, \
81 .fi
82 .LP
83 .nf
84 \fBsize_t\fR \fBstrcspn\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
85 .fi
86 .LP
87 .nf
88 \fBsize_t\fR \fBstrspn\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
89 .fi
90 .LP
91 .nf
92 \fBchar *\fR \fBstrdup\fR(\fBconst char *\fR\fI1\fR);
93 .fi
94 .LP
95 .nf
96 \fBsize_t\fR \fBstrlen\fR(\fBconst char *\fR\fI1\fR);
97 .fi
98 .LP
99 .nf
100 \fBsize_t\fR \fBstrnlen\fR(\fBconst char *\fR\fI1\fR, \fBsize_t\fR \fIn\fR);
101 .fi
102 .LP
103 .nf
104 \fBchar *\fR \fBstrpbrk\fR(\fBconst char *\fR\fI1\fR, \fBconst char *\fR\fI2\fR
105 .fi

```

```

106 .LP
107 .nf
108 \fBchar *\fR\fBstrsep\fR(\fBchar **\fR\fIstringp\fR, \fBconst char *\fR\fI delim\
109 .fi

110 .LP
111 .nf
112 \fBchar *\fR\fBstrchr\fR(\fBconst char *\fR\fI s1\fR, \fBconst char *\fR\fI s2\fR)
113 .fi

114 .LP
115 .nf
116 \fBchar *\fR\fBstrtok\fR(\fBchar *restrict\fR \fI s1\fR, \fBconst char *restrict\
117 .fi

118 .LP
119 .nf
120 \fBchar *\fR\fBstrtok_r\fR(\fBchar *restrict\fR \fI s1\fR, \fBconst char *restrict
121 \fBchar **restrict\fR \fI lastp\fR);
122 .fi

123 .SS "ISO C++"
124 .LP
125 .nf
126 #include <string.h>

128 \fBconst char *\fR\fBstrchr\fR(\fBconst char *\fR\fI s1\fR, \fBint\fR \fI c\fR);
129 .fi

130 .LP
131 .nf
132 \fBconst char *\fR\fBstrbrk\fR(\fBconst char *\fR\fI s1\fR, \fBconst char *\fR\fI
133 .fi

134 .LP
135 .nf
136 \fBconst char *\fR\fBstrrchr\fR(\fBconst char *\fR\fI s1\fR, \fBint\fR \fI c\fR);
137 .fi

138 .LP
139 .nf
140 \fBconst char *\fR\fBstrstr\fR(\fBconst char *\fR\fI s1\fR, \fBconst char *\fR\fI
141 .fi

142 .LP
143 .nf
144 #include <cstring>

146 \fBchar *std::\fR\fBstrchr\fR(\fBchar *\fR\fI s1\fR, \fBint\fR \fI c\fR);
147 .fi

148 .LP
149 .nf
150 \fBchar *std::\fR\fBstrbrk\fR(\fBchar *\fR\fI s1\fR, \fBconst char *\fR\fI s2\fR)
151 .fi

152 .LP
153 .nf
154 \fBchar *std::\fR\fBstrrchr\fR(\fBchar *\fR\fI s1\fR, \fBint\fR \fI c\fR);
155 .fi

156 .LP
157 .nf
158 \fBchar *std::\fR\fBstrstr\fR(\fBchar *\fR\fI s1\fR, \fBconst char *\fR\fI s2\fR);
159 .fi

```

```

160 .SH DESCRIPTION
161 .sp
162 .LP
163 The arguments \fI s1\fR, \fI s2\fR, and \fI s3\fR point to strings (arrays of
164 characters terminated by a null character). The \fBstrcat()\fR,
165 \fBstrncat()\fR, \fBstrlcat()\fR, \fBstrcpy()\fR, \fBstrncpy()\fR,
166 \fBstrncpy()\fR, \fBstrsep()\fR, \fBstrtok()\fR, and \fBstrtok_r()\fR functions
167 all alter their first argument. Additionally, the \fBstrcat()\fR and
168 \fBstrcpy()\fR functions do not check for overflow of the array.
169 .SS "\fBstrcasecmp()\fR, \fBstrncasecmp()\fR"
170 .sp
171 .LP
172 The \fBstrcasecmp()\fR and \fBstrncasecmp()\fR functions are case-insensitive
173 versions of \fBstrcmp()\fR and \fBstrncmp()\fR respectively, described below.
174 .LP
175 The \fBstrcasecmp()\fR and \fBstrncasecmp()\fR functions compare two strings
176 byte-by-byte, after
177 converting each upper-case character to lower-case (as determined by the
178 \fBLC_TYPE\fR category of the current locale). Note that neither the contents
179 pointed to by \fI s1\fR nor \fI s2\fR are modified.
180 .LP
181 The functions return an integer
182 greater than, equal to, or less than 0, if the string pointed to by \fI s1\fR
183 is greater than, equal to, or less than the string pointed to by \fI s2\fR
184 respectively. The sign of a non-zero return value is determined by the sign of
185 the difference between the values of the first pair of bytes that differ in the
186 string.
187 .SS "\fBstrcasecmp_l()\fR, \fBstrncasecmp_l()\fR"
188 .sp
189 .LP
190 The \fBstrcasecmp_l()\fR and \fBstrncasecmp_l()\fR functions behave identically
191 to \fBstrcasecmp()\fR and \fBstrncasecmp()\fR, except instead of operating in
192 the current locale, they instead operate in the locale specified by \fI loc.
193 They assume the \fBASCII\fR character set and ignore differences in case when
194 comparing lower and upper case characters.
195 .SS "\fBstrcat()\fR, \fBstrncat()\fR, \fBstrlcat()\fR"
196 .sp
197 .LP
198 The \fBstrcat()\fR function appends a copy of string \fI s2\fR, including the
199 terminating null character, to the end of string \fI s1\fR. The \fBstrncat()\fR
200 function appends at most \fI n\fR characters. Each returns a pointer to the
201 null-terminated result. The initial character of \fI s2\fR overrides the null
202 character at the end of \fI s1\fR. If copying takes place between objects that
203 overlap, the behavior of \fBstrcat()\fR, \fBstrncat()\fR, and \fBstrlcat()\fR
204 is undefined.
205 .sp
206 .LP
207 The \fBstrlcat()\fR function appends at most
208 (\fI dstsize\fR-\fBstrlen()\fR(\fI dst)-1) characters of \fI src\fR to \fI dst\fR
209 (\fI dstsize\fR being the size of the string buffer \fI dst\fR). If the string
210 pointed to by \fI dst\fR contains a null-terminated string that fits into
211 \fI dstsize\fR bytes when \fBstrlcat()\fR is called, the string pointed to by
212 \fI dst\fR will be a null-terminated string that fits in \fI dstsize\fR bytes
213 (including the terminating null character) when it completes, and the initial
214 character of \fI src\fR will override the null character at the end of
215 \fI dst\fR. If the string pointed to by \fI dst\fR is longer than \fI dstsize\fR
216 bytes when \fBstrlcat()\fR is called, the string pointed to by \fI dst\fR will
217 not be changed. The function returns
218 \fBmin()\fR(\fI dstsize\fR,\fBstrlen()\fR(\fI dst))+\fBstrlen()\fR(\fI src).
219 Buffer overflow can be checked as follows:
220 .sp
221 .in +2
222 .nf
223 if (strlcat(dst, src, dstsize) >= dstsize)

```

```

220         return \(mil;
221 .fi
222 .in -2

223 .SS "\fBstrchr()\fR, \fBstrrchr()\fR"
224 .sp
225 The \fBstrchr()\fR function returns a pointer to the first occurrence of
226 \fIc\fR (converted to a \fBchar\fR) in string \fIs\fR, or a null pointer if
227 \fIc\fR does not occur in the string. The \fBstrrchr()\fR function returns a
228 pointer to the last occurrence of \fIc\fR. The null character terminating a
229 string is considered to be part of the string.
230 .SS "\fBstrcmp()\fR, \fBstrncmp()\fR"
231 .sp
232 The \fBstrcmp()\fR function compares two strings byte-by-byte, according to the
233 ordering of your machine's character set. The function returns an integer
234 greater than, equal to, or less than 0, if the string pointed to by \fIs1\fR
235 is greater than, equal to, or less than the string pointed to by \fIs2\fR
236 respectively. The sign of a non-zero return value is determined by the sign of
237 the difference between the values of the first pair of bytes that differ in the
238 strings being compared. The \fBstrncmp()\fR function makes the same comparison
239 but looks at a maximum of \fIn\fR bytes. Bytes following a null byte are not
240 compared.
241 .SS "\fBstrcpy()\fR, \fBstrncpy()\fR, \fBstrncpy()\fR"
242 .sp
243 The \fBstrcpy()\fR function copies string \fIs2\fR to \fIs1\fR, including the
244 terminating null character, stopping after the null character has been copied.
245 The \fBstrncpy()\fR function copies exactly \fIn\fR bytes, truncating \fIs2\fR
246 or adding null characters to \fIs1\fR if necessary. The result will not be
247 null-terminated if the length of \fIs2\fR is \fIn\fR or more. Each function
248 returns \fIs1\fR. If copying takes place between objects that overlap, the
249 behavior of \fBstrcpy()\fR, \fBstrncpy()\fR, and \fBstrncpy()\fR is undefined.
250 .sp
251 The \fBstrncpy()\fR function copies at most \fIdstsize\fR(mil characters
252 (\fIdstsize\fR being the size of the string buffer \fIdst\fR) from \fIsrc\fR
253 to \fIdst\fR, truncating \fIsrc\fR if necessary. The result is always
254 null-terminated. The function returns \fBstrlen\fR(\fIsrc\fR). Buffer overflow
255 can be checked as follows:
256 .sp
257 .in +2
258 .nf
259 if (strlen(dst, src, dstsize) >= dstsize)
260     return \(mil;
261 .fi
262 .in -2

264 .SS "\fBstrcspn()\fR, \fBstrspn()\fR"
265 .sp
266 The \fBstrcspn()\fR function returns the length of the initial segment of
267 string \fIs1\fR that consists entirely of characters not from string \fIs2\fR.
268 The \fBstrspn()\fR function returns the length of the initial segment of string
269 \fIs1\fR that consists entirely of characters from string \fIs2\fR.
270 .SS "\fBstrdup()\fR"
271 .sp
272 The \fBstrdup()\fR function returns a pointer to a new string that is a
273 duplicate of the string pointed to by \fIs1\fR. The returned pointer can be
274 passed to \fBfree()\fR. The space for the new string is obtained using
275 \fBmalloc\fR(3C). If the new string cannot be created, a null pointer is
276 returned and \fBerrno\fR may be set to \fBENOMEM\fR to indicate that the
277 storage space available is insufficient.
278 .SS "\fBstrlen()\fR, \fBstrnlen()\fR"

```

```

279 .sp
280 The \fBstrlen()\fR function returns the number of bytes in \fIs\fR, not
281 including the terminating null character.
282 .sp
283 The \fBstrnlen()\fR function returns the smaller of \fIn\fR or the number of
284 bytes in \fIs\fR, not including the terminating null character. The
285 \fBstrnlen()\fR function never examines more than \fIn\fR bytes of the string
286 pointed to by \fIs\fR.
287 .SS "\fBstrpbrk()\fR"
288 .sp
289 The \fBstrpbrk()\fR function returns a pointer to the first occurrence in
290 string \fIs1\fR of any character from string \fIs2\fR, or a null pointer if no
291 character from \fIs2\fR exists in \fIs1\fR.
292 .SS "\fBstrsep()\fR"
293 .sp
294 The \fBstrsep()\fR function locates, in the null-terminated string referenced
295 by *\fIstringp\fR, the first occurrence of any character in the string
296 \fIdelim\fR (or the terminating '\e0' character) and replaces it with a '\e0'.
297 The location of the next character after the delimiter character (or
298 \fINULL\fR, if the end of the string was reached) is stored in *\fIstringp\fR.
299 The original value of *\fIstringp\fR is returned.
300 .sp
301 An 'empty' field (one caused by two adjacent delimiter characters) can be
302 detected by comparing the location referenced by the pointer returned by
303 \fBstrsep()\fR to '\e0'.
304 .sp
305 If *\fIstringp\fR is initially \fINULL\fR, \fBstrsep()\fR returns \fINULL\fR.
306 .SS "\fBstrstr()\fR"
307 .sp
308 The \fBstrstr()\fR function locates the first occurrence of the string \fIs2\fR
309 (excluding the terminating null character) in string \fIs1\fR and returns a
310 pointer to the located string, or a null pointer if the string is not found. If
311 \fIs2\fR points to a string with zero length (that is, the string \fB"\fR),
312 the function returns \fIs1\fR.
313 .SS "\fBstrtok()\fR"
314 .sp
315 A sequence of calls to \fBstrtok()\fR breaks the string pointed to by \fIs1\fR
316 into a sequence of tokens, each of which is delimited by a byte from the string
317 pointed to by \fIs2\fR. The first call in the sequence has \fIs1\fR as its
318 first argument, and is followed by calls with a null pointer as their first
319 argument. The separator string pointed to by \fIs2\fR can be different from
320 call to call.
321 .sp
322 The first call in the sequence searches the string pointed to by \fIs1\fR for
323 the first byte that is not contained in the current separator string pointed to
324 by \fIs2\fR. If no such byte is found, then there are no tokens in the string
325 pointed to by \fIs1\fR and \fBstrtok()\fR returns a null pointer. If such a
326 byte is found, it is the start of the first token.
327 .sp
328 The \fBstrtok()\fR function then searches from there for a byte that is
329 contained in the current separator string. If no such byte is found, the
330 current token extends to the end of the string pointed to by \fIs1\fR, and
331 subsequent searches for a token return a null pointer. If such a byte is found,
332 it is overwritten by a null byte that terminates the current token. The
333 \fBstrtok()\fR function saves a pointer to the following byte in
334 thread-specific data, from which the next search for a token starts.

```



```

357 .sp
358 .LP
359 Each subsequent call, with a null pointer as the value of the first argument,
360 starts searching from the saved pointer and behaves as described above.
361 .sp
362 .LP
363 See Example 1, 2, and 3 in the \fBEXAMPLES\fR section for examples of
364 \fBstrtok()\fR usage and the explanation in \fBNOTES\fR.
365 .SS "\fBstrtok_r()\fR"
366 .sp
367 .LP
368 The \fBstrtok_r()\fR function considers the null-terminated string \fIstr\fR as
369 a sequence of zero or more text tokens separated by spans of one or more
370 characters from the separator string \fIsep\fR. The argument \fIlasts\fR points
371 to a user-provided pointer which points to stored information necessary for
372 \fBstrtok_r()\fR to continue scanning the same string.
373 .sp
374 .LP
375 In the first call to \fBstrtok_r()\fR, \fIstr\fR points to a null-terminated
376 string, \fIsep\fR to a null-terminated string of separator characters, and the
377 value pointed to by \fIlasts\fR is ignored. The \fBstrtok_r()\fR function
378 returns a pointer to the first character of the first token, writes a null
379 character into \fIstr\fR immediately following the returned token, and updates
380 the pointer to which \fIlasts\fR points.
381 .sp
382 .LP
383 In subsequent calls, \fIstr\fR is a null pointer and \fIlasts\fR is unchanged
384 from the previous call so that subsequent calls move through the string
385 \fIstr\fR, returning successive tokens until no tokens remain. The separator
386 string \fIsep\fR can be different from call to call. When no token remains in
387 \fIstr\fR, a null pointer is returned.
388 .sp
389 .LP
390 See Example 3 in the \fBEXAMPLES\fR section for an example of \fBstrtok_r()\fR
391 usage and the explanation in \fBNOTES\fR.
392 .SH EXAMPLES
393 .LP
394 \fBExample 1 \fRSearch for word separators.
395 .sp
396 .LP
397 The following example searches for tokens separated by space characters.
398 .sp
399 .in +2
400 .nf
401 #include <string.h>
402 \&...
403 char *token;
404 char line[] = "LINE TO BE SEPARATED";
405 char *search = " ";
406
407 /* Token will point to "LINE". */
408 token = strtok(line, search);
409
410 /* Token will point to "TO". */
411 token = strtok(NULL, search);
412 .fi
413 .in -2
414
415 .LP
416 \fBExample 2 \fRBreak a Line.
417 .sp
418 .LP
419 The following example uses strtok to break a line into two character strings
420 separated by any combination of SPACES, TABS, or NEWLINES.

```

```

421 .sp
422 .in +2
423 .nf
424 #include <string.h>
425 \&...
426 struct element {
427     char *key;
428     char *data;
429 };
430 \&...
431 char line[LINE_MAX];
432 char *key, *data;
433 \&...
434 key = strtok(line, "\n");
435 data = strtok(NULL, "\n");
436 .fi
437 .in -2
438
439 .LP
440 \fBExample 3 \fRSearch for tokens.
441 .sp
442 .LP
443 The following example uses both \fBstrtok()\fR and \fBstrtok_r()\fR to search
444 for tokens separated by one or more characters from the string pointed to by
445 the second argument, "\fR".
446 .sp
447 .in +2
448 .nf
449 #define __EXTENSIONS__
450 #include <stdio.h>
451 #include <string.h>
452
453 int
454 main() {
455     char *buf="5/90/45";
456     char *token;
457     char *lasts;
458
459     printf("tokenizing \e%s\e with strtok():\en", buf);
460     if ((token = strtok(buf, "/")) != NULL) {
461         printf("token = %s\e\n", token);
462         while ((token = strtok(NULL, "/")) != NULL) {
463             printf("token = \e%s\e\n", token);
464         }
465     }
466
467     buf = "//5//90//45//";
468     printf("\entokenizing \e%s\e with strtok_r():\en", buf);
469     if ((token = strtok_r(buf, "/", &lasts)) != NULL) {
470         printf("token = \e%s\e\n", token);
471         while ((token = strtok_r(NULL, "/", &lasts)) != NULL) {
472             printf("token = \e%s\e\n", token);
473         }
474     }
475 }
476 .fi
477 .in -2
478
479 .sp
480 .LP
481 When compiled and run, this example produces the following output:
482 .sp
483 .in +2
484 .nf

```

```

457 tokenizing "5/90/45" with \fBstrtok()\fR:
458 token = "5"
459 token = "90"
460 token = "45"

462 tokenizing "//5/90//45//" with \fBstrtok_r()\fR:
463 token = "5"
464 token = "90"
465 token = "45"
466 .fi
467 .in -2

469 .SH ATTRIBUTES
502 .sp
470 .LP
471 See \fBattributes\fR(5) for descriptions of the following attributes:
505 .sp

507 .sp
472 .TS
473 box;
474 c | c
475 l | l .
476 ATTRIBUTE TYPE      ATTRIBUTE VALUE
477 -
478 Interface Stability      See below.
514 Interface Stability      Committed
479 -
480 MT-Level             See below.
481 -
482 Standard             See below.
483 .TE

521 .sp
485 .LP
486 The
487 \fBstrlcat()\fR, \fBstrncpy()\fR, and \fBstrsep()\fR functions are Committed.
488 All the rest are Standard.
489 .LP
490 The \fBstrtok()\fR and \fBstrdup()\fR functions are MT-Safe. The remaining
491 functions are Async-Signal-Safe.
525 .sp
492 .LP
493 For all except \fBstrlcat()\fR, \fBstrncpy()\fR, and \fBstrsep()\fR, see
494 \fBstandards\fR(5).
495 .SH SEE ALSO
530 .sp
496 .LP
497 \fBmalloc\fR(3C),
498 \fBnewlocale(3C), \fBsetlocale\fR(3C), \fBstrxfrm\fR(3C), \fBuselocale\fR(3C),
499 \fBattributes\fR(5), \fBstandards\fR(5)
532 \fBmalloc\fR(3C), \fBsetlocale\fR(3C), \fBstrxfrm\fR(3C), \fBattributes\fR(5),
533 \fBstandards\fR(5)
500 .SH NOTES
535 .sp
501 .LP
502 When compiling multithreaded applications, the \fB_REENTRANT\fR flag must be
503 defined on the compile line. This flag should only be used in multithreaded
504 applications.
540 .sp
505 .LP
506 A single-threaded application can gain access to \fBstrtok_r()\fR only by
507 defining \fB__EXTENSIONS_\fR or by defining \fB_POSIX_C_SOURCE\fR to a value
508 greater than or equal to 199506L.
545 .sp
509 .LP

```

```

510 Except where noted otherwise, all of these functions assume the default
511 locale ``C.`` For some locales,
547 All of these functions assume the default locale ``C.`` For some locales,
512 \fBstrxfrm\fR(3C) should be applied to the strings before they are passed to
513 the functions.
550 .sp
514 .LP
515 The \fBstrtok()\fR function is safe to use in multithreaded applications
516 because it saves its internal state in a thread-specific data area. However,
517 its use is discouraged, even for single-threaded applications. The
518 \fBstrtok_r()\fR function should be used instead.
556 .sp
519 .LP
520 Do not pass the address of a character string literal as the argument \fBis1\fR
521 to either \fBstrtok()\fR or \fBstrtok_r()\fR. Similarly, do not pass a pointer
522 to the address of a character string literal as the argument \fBistringp\fR to
523 \fBstrsep()\fR. These functions can modify the storage pointed to by \fBis1\fR
524 in the case of \fBstrtok()\fR and \fBstrtok_r()\fR or *\fBistringp\fR in the
525 case of \fBstrsep()\fR. The C99 standard specifies that attempting to modify
526 the storage occupied by a string literal results in undefined behavior. This
527 allows compilers (including \fBgcc\fR and the Sun Studio compilers when the
528 \fB-xstrconst\fR flag is used) to place string literals in read-only memory.
529 Note that in Example 1 above, this problem is avoided because the variable
530 \fBiline\fR is declared as a writable array of type \fBbchar\fR that is
531 initialized by a string literal rather than a pointer to \fBbchar\fR that points
532 to a string literal.

```

\*\*\*\*\*

16993 Fri Jul 11 11:43:10 2014

new/usr/src/man/man3c/strptime.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \" te
2  \\ Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  \\ Copyright (c) 1992, X/Open Company Limited. All Rights Reserved.
4  \\ Portions Copyright (c) 2007, Sun Microsystems, Inc. All Rights Reserved.
5  \\ Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
6  \\ http://www.opengroup.org/bookstore/.
7  \\ The Institute of Electrical and Electronics Engineers and The Open Group, ha
8  \\ This notice shall appear on any product containing this material.
9  \\ The contents of this file are subject to the terms of the Common Development
10 \\ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
11 \\ When distributing Covered Code, include this CDDL HEADER in each file and in
12 .TH STRPTIME 3C "Jun 27, 2014"
13 .TH STRPTIME 3C "Aug 27, 2007"
14 .SH NAME
15 strptime, strptime_1 \- date and time conversion
16 .LP
17 .nf
18 #include <time.h>

20 \fBchar *\fR\fBstrptime\fR(\fBconst char *restrict\fR \fIbuf\fR,
21 \fBconst char *restrict\fR \fIformat\fR, \fBstruct tm *restrict\fR \fIitm\fR
22 .fi
23 .LP
24 .nf
25 #include <time.h>
26 #include <xlocale.h>

28 \fBchar *\fR\fBstrptime_1\fR(\fBconst char *restrict\fR \fIbuf\fR,
29 \fBconst char *restrict\fR \fIformat\fR, \fBstruct tm *restrict\fR \fIitm\fR
30 \fBlocale_t\fR \fIloc\fR);
31 .fi
32 .SS "Non-zeroing Behavior"
33 .LP
34 .nf
35 \fBcc\fR [\fIflag\fR...] \fIfile\fR... \fB-D_STRPTIME_DONTZERO\fR [\fIlibrary\f
36 #include <time.h>

38 \fBchar *\fR\fBstrptime\fR(\fBconst char *restrict\fR \fIbuf\fR,
39 \fBconst char *restrict\fR \fIformat\fR, \fBstruct tm *restrict\fR \fIitm\fR
40 .fi

42 .SH DESCRIPTION
43 .sp
44 The \fBstrptime()\fR function converts the character string pointed to by
45 \fIbuf\fR to values which are stored in the \fBtm\fR structure pointed to by
46 \fIitm\fR, using the format specified by \fIformat\fR. The \fBstrptime_1()\fR
47 function is identical to \fBstrptime()\fR except instead of acting in the
48 current locale, it acts in the locale specified by the argument \fIloc\fR.
49 \fIitm\fR, using the format specified by \fIformat\fR.
49 .sp
49 .LP
50 The \fIformat\fR argument is composed of zero or more conversion
51 specifications. Each conversion specification is composed of a "%" (percent)
52 character followed by one or two conversion characters which specify the
53 replacement required. One or more white space characters (as specified by

```

```

54 \fBisspace\fR(3C)) may precede or follow a conversion specification. There must
55 be white-space or other non-alphanumeric characters between any two conversion
56 specifications.
57 .sp
58 A non-zeroing version of \fBstrptime()\fR, described below under \fBNon-zeroing
59 Behavior\fR, is provided if \fB_STRPTIME_DONTZERO\fR is defined.
60 .SS "Conversion Specifications"
61 .sp
62 The following conversion specifications are supported:
63 .sp
64 .ne 2
65 .na
66 \fB%\fR
67 .ad
68 .RS 6n
69 Same as \fB%\fR.
70 .RE

72 .sp
73 .ne 2
74 .na
75 \fB%a\fR
76 .ad
77 .RS 6n
78 Day of week, using the locale's weekday names; either the abbreviated or full
79 name may be specified.
80 .RE

82 .sp
83 .ne 2
84 .na
85 \fB%A\fR
86 .ad
87 .RS 6n
88 Same as \fB%a\fR.
89 .RE

91 .sp
92 .ne 2
93 .na
94 \fB%b\fR
95 .ad
96 .RS 6n
97 Month, using the locale's month names; either the abbreviated or full name may
98 be specified.
99 .RE

101 .sp
102 .ne 2
103 .na
104 \fB%B\fR
105 .ad
106 .RS 6n
107 Same as \fB%b\fR.
108 .RE

110 .sp
111 .ne 2
112 .na
113 \fB%c\fR
114 .ad
115 .RS 6n
116 Locale's appropriate date and time representation.
117 .RE

```

```

119 .sp
120 .ne 2
121 .na
122 \fB\fB%C\fR\fR
123 .ad
124 .RS 6n
125 Century number (the year divided by 100 and truncated to an integer as a
126 decimal number [1,99]); single digits are preceded by 0. If \fB%C\fR is used
127 without the \fB%y\fR specifier, \fBstrptime()\fR assumes the year offset is
128 zero in whichever century is specified. Note the behavior of \fB%C\fR in the
129 absence of \fB%y\fR is not specified by any of the standards or specifications
130 described on the \fBstandards\fR(5) manual page, so portable applications
131 should not depend on it. This behavior may change in a future release.
132 .RE

134 .sp
135 .ne 2
136 .na
137 \fB\fB%d\fR\fR
138 .ad
139 .RS 6n
140 Day of month [1,31]; leading zero is permitted but not required.
141 .RE

143 .sp
144 .ne 2
145 .na
146 \fB\fB%D\fR\fR
147 .ad
148 .RS 6n
149 Date as \fB%m\fR/\fB%d\fR/\fB%y\fR.
150 .RE

152 .sp
153 .ne 2
154 .na
155 \fB\fB%e\fR\fR
156 .ad
157 .RS 6n
158 Same as \fB%d\fR.
159 .RE

161 .sp
162 .ne 2
163 .na
164 \fB\fB%h\fR\fR
165 .ad
166 .RS 6n
167 Same as \fB%b\fR.
168 .RE

170 .sp
171 .ne 2
172 .na
173 \fB\fB%H\fR\fR
174 .ad
175 .RS 6n
176 Hour (24-hour clock) [0,23]; leading zero is permitted but not required.
177 .RE

179 .sp
180 .ne 2
181 .na
182 \fB\fB%I\fR\fR
183 .ad

```

```

184 .RS 6n
185 Hour (12-hour clock) [1,12]; leading zero is permitted but not required.
186 .RE

188 .sp
189 .ne 2
190 .na
191 \fB\fB%j\fR\fR
192 .ad
193 .RS 6n
194 Day number of the year [1,366]; leading zeros are permitted but not required.
195 .RE

197 .sp
198 .ne 2
199 .na
200 \fB\fB%m\fR\fR
201 .ad
202 .RS 6n
203 Month number [1,12]; leading zero is permitted but not required.
204 .RE

206 .sp
207 .ne 2
208 .na
209 \fB\fB%M\fR\fR
210 .ad
211 .RS 6n
212 Minute [0-59]; leading zero is permitted but not required.
213 .RE

215 .sp
216 .ne 2
217 .na
218 \fB\fB%n\fR\fR
219 .ad
220 .RS 6n
221 Any white space.
222 .RE

224 .sp
225 .ne 2
226 .na
227 \fB\fB%p\fR\fR
228 .ad
229 .RS 6n
230 Locale's equivalent of either a.m. or p.m.
231 .RE

233 .sp
234 .ne 2
235 .na
236 \fB\fB%r\fR\fR
237 .ad
238 .RS 6n
239 Appropriate time representation in the 12-hour clock format with \fB%p\fR.
240 .RE

242 .sp
243 .ne 2
244 .na
245 \fB\fB%R\fR\fR
246 .ad
247 .RS 6n
248 Time as \fB%H\fR:\fB%M\fR.
249 .RE

```

```

251 .SS "SUSv3"
252 .sp
253 .ne 2
254 .na
255 \fB\fB%S\fR\fR
256 .ad
257 .RS 6n
258 Seconds [0,60]; leading zero is permitted but not required. The range of values
259 is [00,60] rather than [00,59] to allow for the occasional leap second.
260 .RE

262 .SS "Default and other standards"
263 .sp
264 .ne 2
265 .na
266 \fB\fB%S\fR\fR
267 .ad
268 .RS 6n
269 Seconds [0,61]; leading zero is permitted but not required. The range of values
270 is [00,61] rather than [00,59] to allow for the occasional leap second and even
271 more occasional double leap second.
272 .RE

274 .sp
275 .ne 2
276 .na
277 \fB\fB%t\fR\fR
278 .ad
279 .RS 6n
280 Any white space.
281 .RE

283 .sp
284 .ne 2
285 .na
286 \fB\fB%T\fR\fR
287 .ad
288 .RS 6n
289 Time as \fB%H\fR:\fB%M\fR:\fB%S\fR.
290 .RE

292 .sp
293 .ne 2
294 .na
295 \fB\fB%U\fR\fR
296 .ad
297 .RS 6n
298 Week number of the year as a decimal number [0,53], with Sunday as the first
299 day of the week; leading zero is permitted but not required.
300 .RE

302 .sp
303 .ne 2
304 .na
305 \fB\fB%w\fR\fR
306 .ad
307 .RS 6n
308 Weekday as a decimal number [0,6], with 0 representing Sunday.
309 .RE

311 .sp
312 .ne 2
313 .na
314 \fB\fB%W\fR\fR
315 .ad

```

```

316 .RS 6n
317 Week number of the year as a decimal number [0,53], with Monday as the first
318 day of the week; leading zero is permitted but not required.
319 .RE

321 .sp
322 .ne 2
323 .na
324 \fB\fB%x\fR\fR
325 .ad
326 .RS 6n
327 Locale's appropriate date representation.
328 .RE

330 .sp
331 .ne 2
332 .na
333 \fB\fB%X\fR\fR
334 .ad
335 .RS 6n
336 Locale's appropriate time representation.
337 .RE

339 .sp
340 .ne 2
341 .na
342 \fB\fB%y\fR\fR
343 .ad
344 .RS 6n
345 Year within century. When a century is not otherwise specified, values in the
346 range 69-99 refer to years in the twentieth century (1969 to 1999 inclusive);
347 values in the range 00-68 refer to years in the twenty-first century (2000 to
348 2068 inclusive).
349 .RE

351 .sp
352 .ne 2
353 .na
354 \fB\fB%Y\fR\fR
355 .ad
356 .RS 6n
357 Year, including the century (for example, 1993).
358 .RE

360 .sp
361 .ne 2
362 .na
363 \fB\fB%Z\fR\fR
364 .ad
365 .RS 6n
366 Time zone name or no characters if no time zone exists.
367 .RE

369 .SS "Modified Conversion Specifications"
370 .LP
371 Some conversion specifications can be modified by the \fB\E\fR and \fB\O\fR
372 modifier characters to indicate that an alternate format or specification
373 should be used rather than the one normally used by the unmodified
374 specification. If the alternate format or specification does not exist in the
375 current locale, the behavior will be as if the unmodified conversion
376 specification were used.
377 .sp
378 .ne 2
379 .na
380 \fB\fB%Ec\fR\fR

```

```

381 .ad
382 .RS 7n
383 Locale's alternate appropriate date and time representation.
384 .RE

386 .sp
387 .ne 2
388 .na
389 \fB\fB%EC\fR\fR
390 .ad
391 .RS 7n
392 Name of the base year (era) in the locale's alternate representation.
393 .RE

395 .sp
396 .ne 2
397 .na
398 \fB\fB%Ex\fR\fR
399 .ad
400 .RS 7n
401 Locale's alternate date representation.
402 .RE

404 .sp
405 .ne 2
406 .na
407 \fB\fB%EX\fR\fR
408 .ad
409 .RS 7n
410 Locale's alternate time representation.
411 .RE

413 .sp
414 .ne 2
415 .na
416 \fB\fB%Ey\fR\fR
417 .ad
418 .RS 7n
419 Offset from \fB%EC\fR (year only) in the locale's alternate representation.
420 .RE

422 .sp
423 .ne 2
424 .na
425 \fB\fB%EY\fR\fR
426 .ad
427 .RS 7n
428 Full alternate year representation.
429 .RE

431 .sp
432 .ne 2
433 .na
434 \fB\fB%Od\fR\fR
435 .ad
436 .RS 7n
437 Day of the month using the locale's alternate numeric symbols.
438 .RE

440 .sp
441 .ne 2
442 .na
443 \fB\fB%Oe\fR\fR
444 .ad
445 .RS 7n
446 Same as \fB%Od\fR.

```

```

447 .RE

449 .sp
450 .ne 2
451 .na
452 \fB\fB%OH\fR\fR
453 .ad
454 .RS 7n
455 Hour (24-hour clock) using the locale's alternate numeric symbols.
456 .RE

458 .sp
459 .ne 2
460 .na
461 \fB\fB%OI\fR\fR
462 .ad
463 .RS 7n
464 Hour (12-hour clock) using the locale's alternate numeric symbols.
465 .RE

467 .sp
468 .ne 2
469 .na
470 \fB\fB%Om\fR\fR
471 .ad
472 .RS 7n
473 Month using the locale's alternate numeric symbols.
474 .RE

476 .sp
477 .ne 2
478 .na
479 \fB\fB%OM\fR\fR
480 .ad
481 .RS 7n
482 Minutes using the locale's alternate numeric symbols.
483 .RE

485 .sp
486 .ne 2
487 .na
488 \fB\fB%OS\fR\fR
489 .ad
490 .RS 7n
491 Seconds using the locale's alternate numeric symbols.
492 .RE

494 .sp
495 .ne 2
496 .na
497 \fB\fB%OU\fR\fR
498 .ad
499 .RS 7n
500 Week number of the year (Sunday as the first day of the week) using the
501 locale's alternate numeric symbols.
502 .RE

504 .sp
505 .ne 2
506 .na
507 \fB\fB%Ow\fR\fR
508 .ad
509 .RS 7n
510 Number of the weekday (Sunday=0) using the locale's alternate numeric symbols.
511 .RE

```

```

513 .sp
514 .ne 2
515 .na
516 \fB\fB%OW\fR\fR
517 .ad
518 .RS 7n
519 Week number of the year (Monday as the first day of the week) using the
520 locale's alternate numeric symbols.
521 .RE

523 .sp
524 .ne 2
525 .na
526 \fB\fB%Oy\fR\fR
527 .ad
528 .RS 7n
529 Year (offset from \fB%C\fR) in the locale's alternate representation and using
530 the locale's alternate numeric symbols.
531 .RE

533 .SS "General Specifications"
528 .sp
534 .LP
535 A conversion specification that is an ordinary character is executed by
536 scanning the next character from the buffer. If the character scanned from the
537 buffer differs from the one comprising the specification, the specification
538 fails, and the differing and subsequent characters remain unscanned.
539 .sp
540 .LP
541 A series of specifications composed of \fB%n\fR, \fB%t\fR, white-space
542 characters or any combination is executed by scanning up to the first character
543 that is not white space (which remains unscanned), or until no more characters
544 can be scanned. White space is defined by \fB\bisspace\fR(3C).
545 .sp
546 .LP
547 Any other conversion specification is executed by scanning characters until a
548 character matching the next specification is scanned, or until no more
549 characters can be scanned. These characters, except the one matching the next
550 specification, are then compared to the locale values associated with the
551 conversion specifier. If a match is found, values for the appropriate
552 \fB%itm\fR structure members are set to values corresponding to the locale
553 information. If no match is found, \fB%strptime()\fR fails and no more
554 characters are scanned.
555 .sp
556 .LP
557 The month names, weekday names, era names, and alternate numeric symbols can
558 consist of any combination of upper and lower case letters. The user can
559 request that the input date or time specification be in a specific language by
560 setting the \fB%BLC_TIME\fR category using \fB%setlocale\fR(3C).
561 .SS "Non-zeroing Behavior"
562 .sp
563 .LP
564 In addition to the behavior described above by various standards, the Solaris
565 implementation of \fB%strptime()\fR provides the following extensions. These may
566 change at any time in the future. Portable applications should not depend on
567 these extended features:
568 .RS +4
569 .TP
570 .ie t \(\bu
571 .el o
572 If \fB%STRPTIME_DONTZERO\fR is not defined, the \fB%Btm struct\fR is zeroed on
573 entry and \fB%strptime()\fR updates the fields of the \fB%Btm struct\fR associated
574 with the specifiers in the format string.
575 .RE
576 .RS +4
577 .TP

```

```

576 .ie t \(\bu
577 .el o
578 If \fB%STRPTIME_DONTZERO\fR is defined, \fB%strptime()\fR does not zero the
579 \fB%Btm struct\fR on entry. Additionally, for some specifiers, \fB%strptime()\fR
580 will use some values in the input \fB%Btm struct\fR to recalculate the date and
581 re-assign the appropriate members of the \fB%Btm struct\fR.
582 .RE
583 .LP
584 The following describes extended features regardless of whether
585 \fB%STRPTIME_DONTZERO\fR is defined or not defined:
586 .RS +4
587 .TP
588 .ie t \(\bu
589 .el o
590 If \fB%j\fR is specified, \fB%Btm_yday\fR is set; if year is given, and if month
591 and day are not given, \fB%strptime()\fR calculates and sets \fB%Btm_mon\fR,
592 \fB%Btm_mday\fR, and \fB%Btm_year\fR.
593 .RE
594 .RS +4
595 .TP
596 .ie t \(\bu
597 .el o
598 If \fB%U\fR or \fB%W\fR is specified and if weekday and year are given and
599 month and day of month are not given, \fB%strptime()\fR calculates and sets
600 \fB%Btm_mon\fR, \fB%Btm_mday\fR, \fB%Btm_wday\fR, and \fB%Btm_year\fR.
601 .RE
602 .sp
603 .LP
604 The following describes extended features when \fB%STRPTIME_DONTZERO\fR is not
605 defined:
606 .RS +4
607 .TP
608 .ie t \(\bu
609 .el o
610 If \fB%C\fR is specified and \fB%y\fR is not specified, \fB%strptime()\fR assumes
611 0 as the year offset, then calculates the year, and assigns \fB%Btm_year\fR.
612 .RE
613 .sp
614 .LP
615 The following describes extended features when \fB%STRPTIME_DONTZERO\fR is
616 defined:
617 .RS +4
618 .TP
619 .ie t \(\bu
620 .el o
621 If \fB%C\fR is specified and \fB%y\fR is not specified, \fB%strptime()\fR
622 assumes the year offset of the year value of the \fB%Btm_year\fR member of the
623 input \fB%Btm struct\fR, then calculates the year and assigns \fB%Btm_year\fR.
624 .RE
625 .RS +4
626 .TP
627 .ie t \(\bu
628 .el o
629 If \fB%j\fR is specified and neither \fB%y\fR, \fB%Y\fR, nor \fB%C\fR are
630 specified, and neither month nor day of month are specified, \fB%strptime()\fR
631 assumes the year value given by the value of the \fB%Btm_year\fR field of the
632 input \fB%Btm struct\fR. Then, in addition to setting \fB%Btm_yday\fR,
633 \fB%strptime()\fR uses day-of-year and year values to calculate the month and
634 day-of-month, and assigns \fB%Btm_month\fR and \fB%Btm_mday\fR.
635 .RE
636 .RS +4
637 .TP
638 .ie t \(\bu
639 .el o
640 If \fB%U\fR or \fB%W\fR is specified, and if weekday and/or year are not given,

```

```

639 and month and day of month are not given, \fBstrptime()\fR will assume the
640 weekday value and/or the year value as the value of the \fBtm_wday\fR field
641 and/or \fBtm_year\fR field of the input \fBtm_struct\fR. Then,
642 \fBstrptime()\fR will calculate the month and day-of-month and assign
643 \fBtm_monh\fR, \fBtm_mday\fR, and/or \fBtm_year\fR.
644 .RE
645 .RS +4
646 .TP
647 .ie t \(\bu
648 .el o
649 If \fB%p\fR is specified and if hour is not specified, \fBstrptime()\fR will
650 reference, and if needed, update the \fBtm_hour\fR member. If the \fBam_pm\fR
651 input is p.m. and the input \fBtm_hour\fR value is between 0 - 11,
652 \fBstrptime()\fR will add 12 hours and update \fBtm_hour\fR. If the
653 \fBam_pm\fR input is a.m. and input \fBtm_hour\fR value is between 12 - 23,
654 \fBstrptime()\fR will subtract 12 hours and update \fBtm_hour\fR.
655 .RE
656 .SH RETURN VALUES
657 .sp
658 .LP
659 Upon successful completion, \fBstrptime()\fR returns a pointer to the character
660 following the last character parsed. Otherwise, a null pointer is returned.
661 .SH USAGE
662 .sp
663 .LP
664 Several "same as" formats, and the special processing of white-space characters
665 are provided in order to ease the use of identical \fIformat\fR strings for
666 \fBstrptime()\fR and \fBstrptime()\fR.
667 .sp
668 .LP
669 The \fBstrptime()\fR function tries to calculate \fBtm_year\fR, \fBtm_mon\fR,
670 and \fBtm_mday\fR when given incomplete input. This allows the \fBstruct tm\fR
671 created by \fBstrptime()\fR to be passed to \fBmktime()\fR(3C) to produce a
672 \fBtime_t\fR value for dates and times that are representable by a
673 \fBtime_t\fR. As an example, since \fBmktime()\fR ignores \fBtm_yday\fR,
674 \fBstrptime()\fR calculates \fBtm_mon\fR and \fBtm_mday\fR as well as filling
675 in \fBtm_yday\fR when \fB%j\fR is specified without otherwise specifying a
676 month and day within month.
677 .SH ATTRIBUTES
678 .sp
679 .LP
680 See \fBattributes()\fR(5) for descriptions of the following attributes:
681 .sp
682 .sp
683 .TS
684 box;
685 c | c
686 l | l .
687 ATTRIBUTE TYPE ATTRIBUTE VALUE
688 -
689 CSI Enabled
690 -
691 Interface Stability See below.
692 Interface Stability Committed
693 -
694 MT-Level MT-Safe
695 -
696 Standard See \fBstandards()\fR(5) for \fBstrptime()\fR.
697 Standard See \fBstandards()\fR(5).
698 .TE
699 .LP
700 The \fBstrptime()\fR function is Standard. The \fBstrptime_l()\fR function
701 is Uncommitted.
702 .SH SEE ALSO

```

```

700 .sp
701 .LP
702 \fBctime()\fR(3C), \fBgetdate()\fR(3C), \fBisspace()\fR(3C), \fBmktime()\fR(3C),
703 \fBnewlocale()\fR(3C),
704 \fBsetlocale()\fR(3C), \fBstrptime()\fR(3C), \fBuselocale()\fR(3C),
705 \fBattributes()\fR(5), \fBenviron()\fR(5),
706 \fBsetlocale()\fR(3C), \fBstrptime()\fR(3C), \fBattributes()\fR(5), \fBenviron()\fR(5),
707 \fBstandards()\fR(5)

```



\*\*\*\*\*

5300 Fri Jul 11 11:43:10 2014

new/usr/src/man/man3c/strxfrm.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  '\" te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright 1989 AT&T. Copyright (c) 2003, Sun Microsystems, Inc. All Rights
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH STRXFRM 3C "Jun 21, 2014"
12 .TH STRXFRM 3C "Dec 10, 2003"
13 .SH NAME
14 strxfrm, strxfrm_l \- string transformation
15 .SH SYNOPSIS
16 .LP
17 #include <string.h>
18
19 \fBsize_t\fR \fBstrxfrm\fR(\fBchar *restrict\fR \fIis1\fR, \fBconst char *restric
20 .fi
21 .LP
22 .nf
23 \fBsize_t\fR \fBstrxfrm_l\fR(\fBchar *restrict\fR \fIis1\fR, \fBconst char *restr
24 \fBlocale_t\fR \fIiloc\fR);
25 .fi
26 .SH DESCRIPTION
27 .sp
28 The \fBstrxfrm()\fR function transforms the string pointed to by \fIis2\fR and
29 places the resulting string into the array pointed to by \fIis1\fR. The
30 transformation is such that if \fBstrncpy\fR(3C) is applied to two transformed
31 strings, it returns a value greater than, equal to or less than \fB0\fR,
32 corresponding to the result of \fBstrcoll\fR(3C) applied to the same two
33 original strings. No more than \fIn\fR bytes are placed into the resulting
34 array pointed to by \fIis1\fR, including the terminating null byte. If \fIn\fR
35 is \fB0\fR, \fIis1\fR is permitted to be a null pointer. If copying takes place
36 between objects that overlap, the behavior is undefined.
37 .sp
38 The \fBstrxfrm()\fR function does not change the setting of \fBerrno\fR if
39 successful.
40 .sp
41 Since no return value is reserved to indicate an error, an application wishing
42 to check for error situations should set \fBerrno\fR to 0, then call
43 \fBstrxfrm()\fR, then check \fBerrno\fR.
44 .LP
45 The \fBstrxfrm_l()\fR functions behaves identically to the function
46 \fBstrxfrm()\fR, except instead of operating in the current locale it
47 operates in the locale specified by the argument \fIiloc\fR.
48 .SH RETURN VALUES
49 .sp
50 Upon successful completion, \fBstrxfrm()\fR returns the length of the
51 transformed string (not including the terminating null byte). If the value

```

```

52 returned is \fIn\fR or more, the contents of the array pointed to by \fIis1\fR
53 are indeterminate.
54 .sp
55 On error, \fBstrxfrm()\fR may set \fBerrno\fR but no return value is reserved
56 to indicate the error.
57 .SH USAGE
58 .sp
59 The transformation function is such that two transformed strings can be ordered
60 by \fBstrncpy\fR(3C) as appropriate to collating sequence information in the
61 locale (category \fBLC_COLLATE\fR).
62 .sp
63 The fact that when \fIn\fR is \fB0\fR, \fIis1\fR is permitted to be a null
64 pointer, is useful to determine the size of the \fIis1\fR array prior to making
65 the transformation.
66 .SH EXAMPLES
67 .LP
68 \fBExample 1\fR \fR sample of using the \fBstrxfrm()\fR function.
69 .LP
70 The value of the following expression is the size of the array needed to hold
71 the transformation of the string pointed to by \fIis\fR.
72
73 .sp
74 .in +2
75 .nf
76 l + strxfrm(NULL, s, 0);
77 .fi
78 .in -2
79
80 .SH FILES
81 .IP \fB/usr/lib/locale/\fR\fIlocale\fR/\fB/LC_COLLATE/*\fR
82 collation database for \fIlocale\fR
83 .sp
84 .ne 2
85 .na
86 \fB/usr/lib/locale/\fIlocale\fR/\fIlocale\fR.so.*\fR
87 .ad
88 .sp .6
89 .RS 4n
90 \fBLC_COLLATE\fR database for \fIlocale\fR
91 .RE
92
93 .SH ATTRIBUTES
94 .sp
95 See \fBattributes\fR(5) for descriptions of the following attributes:
96 .sp
97 .TS
98 box;
99 c | c
100 l | l .
101 ATTRIBUTE TYPE ATTRIBUTE VALUE
102 CSI Enabled
103 -
104 Interface Stability Standard
105 -
106 MT-Level MT-Safe
107 MT-Level MT-Safe with exceptions
108 .TE

```

```
110 .sp
111 .LP
112 The \fBstrxfrm()\fR function can be used safely in a multithreaded application,
113 as long as \fBsetlocale\fR(3C) is not being called to change the locale.
99 .SH SEE ALSO
115 .sp
100 .LP
101 \fBlocaledef\fR(1), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBstrcmp\fR(3C),
102 \fBstrcoll\fR(3C), \fBuselocale\fR(3C),
117 \fBlocaledef\fR(1), \fBsetlocale\fR(3C), \fBstrcmp\fR(3C), \fBstrcoll\fR(3C),
103 \fBwscoll\fR(3C), \fBattributes\fR(5), \fBenviron\fR(5), \fBstandards\fR(5)
```

```

*****
2469 Fri Jul 11 11:43:10 2014
new/usr/src/man/man3c/tolower.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 \" te
2 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3 .\" Copyright (c) 2002, Sun Microsystems, Inc. All Rights Reserved
4 .\" The contents of this file are subject to the terms of the Common Development
5 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
6 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
7 .TH TOLOWER 3C \"Jun 21, 2014\"
8 .TH TOLOWER 3C \"Aug 14, 2002\"
9 .SH NAME
9 tolower, tolower_1 \- transliterate upper-case characters to lower-case
8 tolower \- transliterate upper-case characters to lower-case
10 .SH SYNOPSIS
11 .LP
12 .nf
13 #include <ctype.h>

15 \fBint\fR \fBtolower\fR(\fBint\fR \fIc\fR);
16 .fi
17 .LP
18 .nf
19 \fBint\fR \fBtolower_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIiloc\fR);
20 .fi

22 .SH DESCRIPTION
18 .sp
23 .LP
24 The \fBtolower()\fR function has as a domain a type \fBint\fR, the value of
25 which is representable as an \fBunsigned char\fR or the value of \fBEOF\fR. If
26 the argument has any other value, the argument is returned unchanged. If the
27 argument of \fBtolower()\fR represents an upper-case letter, and there exists a
28 corresponding lower-case letter (as defined by character type information in
29 the locale category \fB\fR\fBLC_CTYPE\fR\fB), \fR the result is the
25 the program locale category \fB\fR\fBLC_CTYPE\fR\fB), \fR the result is the
30 corresponding lower-case letter. All other arguments in the domain are returned
31 unchanged.
32 .LP
33 The function \fBtolower_1()\fR behaves identically to \fBtolower()\fR, except
34 instead of operating in the current locale, it operates in the locale
35 specified by \fIiloc\fR.
36 .SH RETURN VALUES
29 .sp
37 .LP
38 On successful completion, \fBtolower()\fR returns the lower-case letter
39 corresponding to the argument passed. Otherwise, it returns the argument
40 unchanged.
41 .SH ERRORS
35 .sp
42 .LP
43 No errors are defined.
44 .SH ATTRIBUTES
39 .sp
45 .LP
46 See \fBAttributes\fR(5) for descriptions of the following attributes:
42 .sp

44 .sp
47 .TS
48 box;

```

```

49 c | c
50 l | l .
51 ATTRIBUTE TYPE ATTRIBUTE VALUE
52 -
53 CSI Enabled
54 -
55 Interface Stability Standard
56 -
57 MT-Level MT-Safe
58 .TE

60 .SH SEE ALSO
59 .sp
61 .LP
62 \fBtolower\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C),
63 \fBuselocale\fR(3C), \fBattributes\fR(5), \fBstandards\fR(5)
61 \fBtolower\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),
62 \fBstandards\fR(5)

```

```

*****
2411 Fri Jul 11 11:43:10 2014
new/usr/src/man/man3c/toupper.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 \" te
2.\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3.\" Copyright (c) 2002, Sun Microsystems, Inc. All Rights Reserved
4.\" The contents of this file are subject to the terms of the Common Development
5.\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
6.\" When distributing Covered Code, include this CDDL HEADER in each file and in
7.TH TOUPPER 3C "Jun 21, 2014"
6.TH TOUPPER 3C "Aug 14, 2002"
8.SH NAME
9 toupper, toupper_1 \- transliterate lower-case characters to upper-case
8 toupper \- transliterate lower-case characters to upper-case
10 .SH SYNOPSIS
11 .LP
12 .nf
13 #include <ctype.h>

15 \fBint\fR \fBtoupper\fR(\fBint\fR \fIc\fR);
16 .fi
17 .LP
18 .nf
19 \fBint\fR \fBtoupper_1\fR(\fBint\fR \fIc\fR, \fBlocale_t\fR \fIloc\fR);
20 .fi

22 .SH DESCRIPTION
18 .sp
23 .LP
24 The \fBtoupper()\fR function has as a domain a type \fBint\fR, the value of
25 which is representable as an \fBunsigned char\fR or the value of \fBEOF\fR. If
26 the argument has any other value, the argument is returned unchanged. If the
27 argument of \fBtoupper()\fR represents a lower-case letter, and there exists a
28 corresponding upper-case letter (as defined by character type information in
29 the locale category \fBLC_CTYPE\fR), the result is the corresponding
25 the program locale category \fBLC_CTYPE\fR), the result is the corresponding
30 upper-case letter. All other arguments in the domain are returned unchanged.
31 .LP
32 The function \fBtoupper_1()\fR behaves identically to \fBtoupper()\fR, except
33 instead of operating in the current locale, it operates in the locale
34 specified by \fIloc\fR.
35 .SH RETURN VALUES
28 .sp
36 .LP
37 On successful completion, \fBtoupper()\fR returns the upper-case letter
38 corresponding to the argument passed.
39 .SH ERRORS
33 .sp
40 .LP
41 No errors are defined.
42 .SH ATTRIBUTES
37 .sp
43 .LP
44 See \fBattributes\fR(5) for descriptions of the following attributes:
40 .sp

42 .sp
45 .TS
46 box;
47 c | c
48 1 | 1 .

```

```

49 ATTRIBUTE TYPE ATTRIBUTE VALUE
50 _
51 CSI Enabled
52 _
53 Interface Stability Standard
54 _
55 MT-Level MT-Safe
56 .TE

58 .SH SEE ALSO
57 .sp
59 .LP
60 \fBtoupper\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C),
61 \fBuselocale\fR(3C),
62 \fBattributes\fR(5), \fBstandards\fR(5)
59 \fBtoupper\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),
60 \fBstandards\fR(5)

```

```

*****
1767 Fri Jul 11 11:43:10 2014
new/usr/src/man/man3c/towlower.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 '\ ' te
2 .\"
3 .\" This file and its contents are supplied under the terms of the
4 .\" Common Development and Distribution License ("CDDL"), version 1.0.
5 .\" You may only use this file in accordance with the terms of version
6 .\" 1.0 of the CDDL.
7 .\"
8 .\" A full copy of the text of the CDDL should have accompanied this
9 .\" source. A copy of the CDDL is also available via the Internet at
10 .\" http://www.illumos.org/license/CDDL.
11 .\"
12 .\"
13 .\" Copyright (c) 2014 Joyent, Inc. All rights reserved.
14 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 .\"
16 .TH TOWLOWER 3C "Jun 21, 2014"
17 .SH NAME
18 tolower, tolower_l \- transliterate upper-case wide characters to lower-case
19 .SH SYNOPSIS
20 .LP
21 .nf
22 #include <wctype.h>

24 \fBwint_t\fR \fBtolower\fR(\fBwint_t\fR \fIwc\fR);
25 .fi
26 .LP
27 .nf
28 \fBwint_t\fR \fBtolower_l\fR(\fBwint_t\fR \fIwc\fR, \fBlocale_t\fR \fIloc\fR);
29 .nf
30 .SH DESCRIPTION
31 The function
32 .BR tolower()
33 is the wide character equivalent of the function
34 .BR tolower (3C).
35 It converts the upper-case wide character
36 .I wc
37 to the equivalent lower-case
38 wide character, if one exists. If one does not exist, it returns
39 .I wc
40 unchanged.
41 .LP
42 The function
43 .B tolower_l()
44 is equivalent to the function
45 .BR tolower() ,
46 but instead of operating in the current locale, operates in the
47 locale specified by
48 .IR loc .
49 .SH RETURN VALUES
50 On successful completion,
51 .B tolower()
52 and
53 .B tolower_l()
54 return the lower-case character that corresponds to the argument passed.
55 Otherwise, they return the argument unchanged.
56 .SH ERRORS
57 No errors are defined.
58 .SH ATTRIBUTES

```

```

59 .TS
60 box;
61 c | c
62 l | l .
63 ATTRIBUTE TYPE ATTRIBUTE VALUE
64 _
65 Interface Stability Standard
66 _
67 MT-Level MT-Safe
68 .TE

70 .SH SEE ALSO
71 .BR newlocale (3C),
72 .BR setlocale (3C),
73 .BR towupper (3C),
74 .BR uselocale (3C),
75 .BR locale (5)

```

```

*****
1765 Fri Jul 11 11:43:10 2014
new/usr/src/man/man3c/towupper.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 '\ ' te
2 .\"
3 .\" This file and its contents are supplied under the terms of the
4 .\" Common Development and Distribution License ("CDDL"), version 1.0.
5 .\" You may only use this file in accordance with the terms of version
6 .\" 1.0 of the CDDL.
7 .\"
8 .\" A full copy of the text of the CDDL should have accompanied this
9 .\" source. A copy of the CDDL is also available via the Internet at
10 .\" http://www.illumos.org/license/CDDL.
11 .\"
12 .\"
13 .\" Copyright (c) 2014 Joyent, Inc. All rights reserved.
14 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 .\"
16 .TH TOWUPPER 3C "Jun 21, 2014"
17 .SH NAME
18 towupper, towupper_1 \- transliterate lower-case wide characters to upper-case
19 .SH SYNOPSIS
20 .LP
21 .nf
22 #include <wctype.h>

24 \fBwint_t\fR \fBtowupper\fR(\fBwint_t\fR \fIwc\fR);
25 .fi
26 .LP
27 .nf
28 \fBwint_t\fR \fBtowupper_1\fR(\fBwint_t\fR \fIwc\fR, \fBlocale_t\fR \fIloc\fR);
29 .fi
30 .SH DESCRIPTION
31 The function
32 .BR towupper()
33 is the wide character equivalent of the function
34 .BR toupper(3C).
35 It converts the lower-case wide character
36 .I wc
37 to the equivalent upper-case
38 wide character, if one exists. If one does not exist, it returns
39 .I wc
40 unchanged.
41 .LP
42 The function
43 .B towupper_1()
44 is equivalent to the function
45 .BR toupper() ,
46 but instead of operating in the current locale, operates in the
47 locale specified by
48 .IR loc .
49 .SH RETURN VALUES
50 On successful completion,
51 .B towupper()
52 and
53 .B towupper_1()
54 return the upper-case character that corresponds to the argument passed.
55 Otherwise, they return the argument unchanged.
56 .SH ERRORS
57 No errors are defined.
58 .SH ATTRIBUTES

```

```

59 .TS
60 box;
61 c | c
62 l | l .
63 ATTRIBUTE TYPE ATTRIBUTE VALUE
64 _
65 Interface Stability Standard
66 _
67 MT-Level MT-Safe
68 .TE

70 .SH SEE ALSO
71 .BR newlocale(3C),
72 .BR setlocale(3C),
73 .BR toupper(3C),
74 .BR uselocale(3C),
75 .BR locale(5)

```

\*\*\*\*\*

2324 Fri Jul 11 11:43:10 2014  
 new/usr/src/man/man3c/uselocale.3c  
 2964 need POSIX 2008 locale object support  
 Reviewed by: Robert Mustacchi <rm@joyent.com>  
 Reviewed by: Gordon Ross <gordon.ross@nexenta.com>  
 Approved by: TBD

\*\*\*\*\*

```

1  \" te
2  \"
3  \" This file and its contents are supplied under the terms of the
4  \" Common Development and Distribution License (\"CDDL\"), version 1.0.
5  \" You may only use this file in accordance with the terms of version
6  \" 1.0 of the CDDL.
7  \"
8  \" A full copy of the text of the CDDL should have accompanied this
9  \" source. A copy of the CDDL is also available via the Internet at
10 \" http://www.illumos.org/license/CDDL.
11 \"
12 \"
13 \" Copyright (c) 2014 Joyent, Inc. All rights reserved.
14 \" Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 \"
16 .TH USELOCALE 3C \"Jun 24, 2014\"

18 .SH NAME
19 uselocale \- get and set the locale for an individual thread
20 .SH SYNOPSIS
21 .LP
22 .nf
23 #include <locale.h>

25 \fBlocale_t\fR \fBuselocale\fR(\fBlocale_t\fR \fBilloc\fR);
26 .fi
27 .SH DESCRIPTION
28 .LP
29 The
30 .B uselocale()
31 function is used to obtain and set the current locale for a thread. When
32 a thread is created, it uses the global locale as specified by calls to
33 .BR setlocale() .
34 .LP
35 If
36 .BR (locale_t) 0
37 is supplied for
38 .IR loc ,
39 then no change is made to the thread's locale setting. This can be used
40 to query the thread's locale without making any change.
41 .LP
42 If
43 .B LC_GLOBAL_LOCALE
44 is supplied for
45 .IR loc ,
46 then the thread will use the global locale, undoing the effect of any
47 prior call to establish a thread-specific locale.
48 .LP
49 Otherwise the thread will use the supplied
50 .I loc
51 locale object as a thread-specific locale. Changes to the global
52 locale, or to the locale of any other thread, will not affect this thread.
53 .LP
54 Locale objects for use with
55 .B uselocale()
56 can be created with the functions
57 .BR duplocale (3C)
58 and

```

```

59 .BR newlocale (3C).
60 .SH RETURN VALUES
61 Upon successful completion, the
62 .B uselocale()
63 function always returns the previous locale that was set. If no locale
64 was previously set, the global locale,
65 .BR LC_GLOBAL_LOCALE ,
66 is returned. On failure, the
67 .B uselocale()
68 function returns
69 .BR (locale_t) 0,
70 and sets
71 .B errno
72 to indicate the error.
73 .SH ERRORS
74 .TP
75 .B EINVAL
76 An invalid locale was encountered or an internal error occurred that
77 caused the system to be unable to update the locale.
78 .SH ATTRIBUTES
79 .TS
80 box;
81 c | c
82 l | l .
83 ATTRIBUTE TYPE ATTRIBUTE VALUE
84 -
85 Interface Stability Standard
86 -
87 MT-Level MT-Safe
88 .TE

90 .SH SEE ALSO
91 .BR locale (1),
92 .BR duplocale (3C),
93 .BR newlocale (3C),
94 .BR setlocale (3C),
95 .BR locale (5)

```

\*\*\*\*\*

5376 Fri Jul 11 11:43:11 2014  
 new/usr/src/man/man3c/wcrtomb.3c  
 2964 need POSIX 2008 locale object support  
 Reviewed by: Robert Mustacchi <rm@joyent.com>  
 Reviewed by: Gordon Ross <gordon.ross@nexenta.com>  
 Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCRTOMB 3C \"Jun 24, 2014\"
12 .SH NAME
13 wcrtomb, wcrtomb_l \- convert a wide-character code to a character (restartable)
14 wcrtomb \- convert a wide-character code to a character (restartable)
15 .LP
16 .nf
17 #include <stdio.h>

19 \fBsize_t\fR \fBwcrtomb\fR(\fBchar *restrict\fR \fBis\fR, \fBwchar_t\fR \fBw\fR,
20 \fBfi
21 .LP
22 .nf
23 #include <stdio.h>
24 #include <xlocale.h>

26 \fBsize_t\fR \fBwcrtomb_l\fR(\fBchar *restrict\fR \fBis\fR, \fBwchar_t\fR \fBw\fR,
27 \fBlocale_t\fR \fBiloc\fR);
28 .fi
29 .SH DESCRIPTION
30 .LP
31 If \fBis\fR is a null pointer, the \fBwcrtomb()\fR function is equivalent to the
32 call:
33 .IP
34 \fBwcrtomb\fR(\fBibuf\fR, L'\e0', \fBis\fR);
35 .LP
36 where \fBibuf\fR is an internal buffer.
37 .LP
38 If \fBis\fR is not a null pointer, the \fBwcrtomb()\fR function determines the
39 number of bytes needed to represent the character that corresponds to the
40 wide-character given by \fBw\fR (including any shift sequences), and stores
41 the resulting bytes in the array whose first element is pointed to by \fBis\fR.
42 At most \fBMB_CUR_MAX\fR bytes are stored. If \fBw\fR is a null
43 wide-character, a null byte is stored, preceded by any shift sequence needed to
44 restore the initial shift state. The resulting state described is the initial
45 conversion state.
46 .sp
47 .LP
48 If \fBis\fR is a null pointer, the \fBwcrtomb()\fR function uses its own
49 internal \fBmbstate_t\fR object, which is initialized at program startup to the
50 initial conversion state. Otherwise, the \fBmbstate_t\fR object pointed to

```

```

50 by \fBis\fR is used to completely describe the current conversion state of the
51 associated character sequence. The system will behave as if no function
52 defined in the Reference Manual calls \fBwcrtomb()\fR.
53 associated character sequence. Solaris will behave as if no function defined in
54 the Solaris Reference Manual calls \fBwcrtomb()\fR.
55 .sp
56 .LP
57 The behavior of \fBwcrtomb()\fR is affected by the \fBLC_CTYPE\fR category of th
58 current locale. See \fBsetenv\fR(5). The function \fBwcrtomb_l()\fR behaves
59 identically to \fBwcrtomb()\fR, except instead of operating in the current
60 locale, it operates in the locale specified by \fBiloc\fR.
61 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
62 current locale. See \fBsetenv\fR(5).
63 .SH RETURN VALUES
64 .sp
65 .LP
66 The \fBwcrtomb()\fR function returns the number of bytes stored in the array
67 object (including any shift sequences). When \fBw\fR is not a valid
68 wide-character, an encoding error occurs. In this case, the function stores
69 the value of the macros \fBMB_ERRNO\fR in \fBerrno\fR and returns
70 \fB(size_t)\fR(\fBm\fR; the conversion state is undefined.
71 .SH ERRORS
72 .sp
73 .LP
74 The \fBwcrtomb()\fR function may fail if:
75 .sp
76 .ne 2
77 .na
78 \fBEBEINVAL\fR
79 .ad
80 .RS 10n
81 The \fBis\fR argument points to an object that contains an invalid conversion
82 state.
83 .RE

85 .SH USAGE
86 .sp
87 .LP
88 If \fBis\fR is not a null pointer, \fBwcrtomb()\fR uses the \fBmbstate_t\fR
89 object pointed to by \fBis\fR and the function can be used safely in
90 multithreaded applications, as long as \fBsetlocale\fR(3C) is not being called
91 to change the locale. If \fBis\fR is a null pointer, \fBwcrtomb()\fR uses its
92 internal \fBmbstate_t\fR object and the function is Unsafe in multithreaded
93 applications.
94 .SH ATTRIBUTES
95 .sp
96 .LP
97 See \fBattributes\fR(5) for descriptions of the following attributes:
98 .sp

100 .sp
101 .TS
102 box;
103 c | c
104 l | l .
105 ATTRIBUTE TYPE ATTRIBUTE VALUE
106 _

```



```
95 Interface Stability      See below.  
107 Interface Stability      Standard  
96  
97 MT-Level                See below.  
109 MT-Level                 See NOTES below  
98 .TE  
  
100 .LP  
101 The \fBwcrtomb()\fR function is Standard. The  
102 \fBwcrtomb_l()\fR function is Uncommitted.  
103 .LP  
104 If \fR is a null pointer, these functions should be considered Unsafe  
105 for use in multithreaded applications. Otherwise, they are MT-Safe.  
106 .SH SEE ALSO  
113 .sp  
107 .LP  
108 \fBmbsinit\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C),  
109 \fBuselocale\fR(3C), \fBattributes\fR(5),  
115 \fBmbsinit\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),  
110 \fBstandards\fR(5), \fBenviron\fR(5)
```

\*\*\*\*\*

4736 Fri Jul 11 11:43:11 2014

new/usr/src/man/man3c/wcscoll.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCSCOLL 3C "Jun 25, 2014"
12 .TH WCSCOLL 3C "Aug 14, 2002"
13 .SH NAME
14 wcscoll, wscoll_1 wscoll \- wide character string comparison using collating in
15 wscoll, wscoll \- wide character string comparison using collating information
16 .SH SYNOPSIS
17 .LP
18 .nf
19 \fBint\fR \fBwcscoll\fR(\fBconst wchar_t *\fR\fIws1\fR, \fBconst wchar_t *\fR\fI
20 .fi

21 .LP
22 .nf
23 \fBint\fR \fBwcscoll_1\fR(\fBconst wchar_t *\fR\fIws1\fR, \fBconst wchar_t *\fR
24 .fi
25 .LP
26 .nf
27 \fBint\fR \fBwscoll\fR(\fBconst wchar_t *\fR\fIws1\fR, \fBconst wchar_t *\fR\fIw
28 .fi

29 .SH DESCRIPTION
30 .sp
31 .LP
32 The \fBwcscoll()\fR, \fBwcscoll_1()\fR, and \fBwscoll()\fR functions compare
33 the wide character
34 The \fBwcscoll()\fR and \fBwscoll()\fR functions compare the wide character
35 string pointed to by \fIws1\fR to the wide character string pointed to by
36 \fIws2\fR, both interpreted as appropriate to the \fBLC_COLLATE\fR locale
37 category.
38 \fIws2\fR, both interpreted as appropriate to the \fBLC_COLLATE\fR category of
39 the current locale.
40 .sp
41 .LP
42 These functions do not change the setting of \fBerrno\fR if successful.
43 The \fBwcscoll()\fR and \fBwscoll()\fR functions do not change the setting of
44 \fBerrno\fR if successful.
45 .sp
46 .LP
47 An application wanting to check for error situations should set \fBerrno\fR to
48 0 before calling these functions. If \fBerrno\fR is non-zero
49 0 before calling \fBwcscoll()\fR or \fBwscoll()\fR. If \fBerrno\fR is non-zero
50 on return, an error has occurred.
51 .LP
52 The function \fBwscoll_1()\fR behaves identically to \fBwscoll()\fR, except
53 instead of operating in the current locale, it operates in the locale
54 specified by \fBlocale\fR.

```

46 .SH RETURN VALUES

47 .sp

48 .LP

49 Upon successful completion, these functions return an

45 Upon successful completion, \fBwcscoll()\fR and \fBwscoll()\fR return an  
49 integer greater than, equal to, or less than 0, depending upon whether the wide  
50 character string pointed to by \fIws1\fR is greater than, equal to, or less  
51 than the wide character string pointed to by \fIws2\fR, when both are

52 interpreted as appropriate to the the current locale, or the locale

53 specified by \fBlocale\fR. On error,

54 they set \fBerrno\fR, but no return value is reserved to indicate an error.

49 interpreted as appropriate to the current locale. On error, \fBwcscoll()\fR and  
50 \fBwscoll()\fR may set \fBerrno\fR, but no return value is reserved to indicate  
51 an error.

55 .SH ERRORS

56 .sp

57 .LP

58 The \fBwcscoll()\fR, \fBwcscoll\_1()\fR and \fBwscoll()\fR functions may fail if:

55 The \fBwcscoll()\fR and \fBwscoll()\fR functions may fail if:

58 .sp

59 .ne 2

60 .na

61 \fBEBINVAL\fR

62 .ad

63 .RS 10n

64 The \fIws1\fR or \fIws2\fR arguments contain wide character codes outside the  
65 domain of the collating sequence.

66 .RE

67 .SH USAGE

68 .sp

69 .LP

70 The \fBwcsxfrm\fR(3C) and \fBwscmp\fR(3C) functions should be used for sorting

71 large lists, or when performing many comparisons on the same strings.

72 large lists.

73 .SH ATTRIBUTES

74 .sp

75 .LP

76 See \fBattributes\fR(5) for descriptions of the following attributes:

77 .sp

78 .TS

79 box;

80 c | c

81 l | l

82 l | l

83 l | l

84 .

79 ATTRIBUTE TYPE ATTRIBUTE VALUE

80 -

81 CSI Enabled

82 -

83 Interface Stability See below.

86 Interface Stability \fBwcscoll()\fR is Standard

84 -

85 MT-Level MT-Safe

88 MT-Level MT-Safe with exceptions

86 .TE

91 .sp

92 .LP

93 The \fBwcscoll()\fR and \fBwcscoll\_1()\fR functions are Standard.

89 The \fBwscoll()\fR function is Committed.

93 The \fBwcscoll()\fR and \fBwscoll()\fR functions can be used safely in

94 multithreaded applications as long as \fBsetlocale\fR(3C) is not being called

95 to change the locale.

91 .SH SEE ALSO

```
97 .sp
92 .LP
93 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
94 \fBwcscmp\fR(3C), \fBwcsxfrm\fR(3C), \fBattributes\fR(5),
99 \fBsetlocale\fR(3C), \fBwcscmp\fR(3C), \fBwcsxfrm\fR(3C), \fBattributes\fR(5),
95 \fBstandards\fR(5)
```

```

*****
6198 Fri Jul 11 11:43:11 2014
new/usr/src/man/man3c/wcsrtombs.3c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 \" te
2 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3 .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4 .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5 .\" http://www.opengroup.org/bookstore/.
6 .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7 .\" This notice shall appear on any product containing this material.
8 .\" The contents of this file are subject to the terms of the Common Development
9 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCSRTOMBS 3C \"Jun 25, 2014\"
12 .SH WCSRTOMBS 3C \"Nov 1, 2003\"
13 .SH NAME
14 wcsrombs, wcsrombs_1 \- convert a wide-character string to a character string
15 wcsrombs \- convert a wide-character string to a character string
16 (restartable)
17 .SH SYNOPSIS
18 .LP
19 .nf
20 #include <wchar.h>
21
22 \fBsize_t\fR \fBwcsrtombs\fR(\fBchar *restrict\fR \fBidst\fR,
23 \fBconst wchar_t **restrict\fR \fIsrc\fR, \fBsize_t\fR \fIlen\fR,
24 \fBmbstate_t *restrict\fR \fIps\fR);
25 .fi
26 .LP
27 .nf
28 #include <wchar.h>
29 #include <xlocale.h>
30
31 \fBsize_t\fR \fBwcsrtombs_1\fR(\fBchar *restrict\fR \fBidst\fR,
32 \fBconst wchar_t **restrict\fR \fIsrc\fR, \fBsize_t\fR \fIlen\fR,
33 \fBmbstate_t *restrict\fR \fIps\fR, \fBlocale_t\fR \fIloc\fR);
34 .fi
35 .SH DESCRIPTION
36 .sp
37 .LP
38 The \fBwcsrtombs()\fR function converts a sequence of wide-characters from the
39 array indirectly pointed to by \fIsrc\fR into a sequence of corresponding
40 characters, beginning in the conversion state described by the object pointed
41 to by \fIps\fR. If \fBidst\fR is not a null pointer, the converted characters
42 are then stored into the array pointed to by \fBidst\fR. Conversion continues up
43 to and including a terminating null wide-character, which is also stored.
44 Conversion stops earlier in the following cases:
45 .RS +4
46 .TP
47 .ie t \(\bu
48 .el o
49 When a code is reached that does not correspond to a valid character.
50 .RE
51 .RS +4
52 .TP
53 .ie t \(\bu
54 .el o
55 When the next character would exceed the limit of \fIlen\fR total bytes to be
56 stored in the array pointed to by \fBidst\fR (and \fBidst\fR is not a null
57 pointer).
58 .RE

```

```

48 .sp
49 .LP
50 Each conversion takes place as if by a call to the \fBwcsrtomb()\fR function.
51 .sp
52 .LP
53 If \fBidst\fR is not a null pointer, the pointer object pointed to by \fIsrc\fR
54 is assigned either a null pointer (if conversion stopped due to reaching a
55 terminating null wide-character) or the address just past the last
56 wide-character converted (if any). If conversion stopped due to reaching a
57 terminating null wide-character, the resulting state described is the initial
58 conversion state.
59 .sp
60 .LP
61 If \fIps\fR is a null pointer, the \fBwcsrtombs()\fR function uses its own
62 internal \fBmbstate_t\fR object, which is initialized at program startup to the
63 initial conversion state. Otherwise, the \fBmbstate_t\fR object pointed to by
64 \fIps\fR is used to completely describe the current conversion state of the
65 associated character sequence. The system will behave as if no function defined
66 in the Reference Manual calls any of these functions.
67 associated character sequence. Solaris will behave as if no function defined in
68 the Solaris Reference Manual calls \fBwcsrtombs()\fR.
69 .sp
70 .LP
71 The behavior of \fBwcsrtombs()\fR is affected by the \fBLC_CTYPE\fR category of
72 the current locale. See \fBenviron\fR(5).
73 .LP
74 The function \fBwcsrtombs_1()\fR behaves identically to \fBwcsrtombs\fR, except
75 instead of operating in the current locale, it operates in the locale
76 specified by \fIloc\fR.
77 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
78 current locale. See \fBenviron\fR(5).
79 .SH RETURN VALUES
80 .sp
81 .LP
82 If conversion stops because a code is reached that does not correspond to a
83 valid character, an encoding error occurs. In this case, these
84 functions store the value of the macro \fBEILSEQ\fR in \fBerrno\fR and return
85 \fB(size_t)\(mil\fR; the conversion state is undefined. Otherwise, they return
86 valid character, an encoding error occurs. In this case, the \fBwcsrtombs()\fR
87 function stores the value of the macro \fBEILSEQ\fR in \fBerrno\fR and returns
88 \fB(size_t)\(mil\fR; the conversion state is undefined. Otherwise, it returns
89 the number of bytes in the resulting character sequence, not including the
90 terminating null (if any).
91 .SH ERRORS
92 .sp
93 .LP
94 The \fBwcsrtombs()\fR and \fBwcsrtombs_1()\fR functions may fail if:
95 The \fBwcsrtombs()\fR function may fail if:
96 .sp
97 .ne 2
98 .na
99 \fB\BEINVAL\fR
100 .ad
101 .RS 10n
102 The \fIps\fR argument points to an object that contains an invalid conversion
103 state.
104 .RE
105 .sp
106 .ne 2
107 .na
108 \fB\BEILSEQ\fR
109 .ad
110 .RS 10n
111 A wide-character code does not correspond to a valid character.
112 .RE

```

```

103 .SH USAGE
104 .sp
105 .LP
106 If \fIps is not a null pointer, \fBwcsrtombs() uses the \fBmbstate_t
107 object pointed to by \fIps and the function can be used safely in
108 multithreaded applications, as long as \fBsetlocale(3C) is not being called
109 to change the locale. If \fIps is a null pointer, \fBwcsrtombs() uses its
110 internal \fBmbstate_t object and the function is Unsafe in multithreaded
111 applications.
112 .SH ATTRIBUTES
113 .sp
114 .LP
115 See \fBattributes(5) for descriptions of the following attributes:
116 .sp
117 .sp
118 .TS
119 box;
120 c | c
121 l | l .
122 ATTRIBUTE TYPE ATTRIBUTE VALUE
123 ---
124 Interface Stability See below.
125 Interface Stability Standard
126 ---
127 MT-Level See below.
128 MT-Level See NOTES below
129 .TE
130 .LP
131 The \fBwcsrtombs() function is Standard. The \fBwcsrtombs_l()
132 function is Uncommitted.
133 .LP
134 If \fIps is a null pointer, these functions should be considered Unsafe
135 for use in multithreaded applications. Otherwise, they are MT-Safe.
136 .SH SEE ALSO
137 .sp
138 .LP
139 \fBmbsinit(3C), \fBnewlocale(3C), \fBsetlocale(3C), \fBuselocale(3C)
140 \fBwcrctomb(3C), \fBattributes(5),
141 \fBmbsinit(3C), \fBsetlocale(3C), \fBwcrctomb(3C), \fBattributes(5),
142 \fBenviron(5), \fBstandards(5)

```

\*\*\*\*\*

3878 Fri Jul 11 11:43:11 2014

new/usr/src/man/man3c/wcswidth.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1 \" te
2.\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3.\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved.
4.\" Portions Copyright (c) 2002, Sun Microsystems, Inc. All Rights Reserved
5.\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
6.\" http://www.opengroup.org/bookstore/.
7.\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
8.\" This notice shall appear on any product containing this material.
9.\" The contents of this file are subject to the terms of the Common Development
10.\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
11.\" When distributing Covered Code, include this CDDL HEADER in each file and in
12.\" .TH WCSWIDTH 3C "Jul 11, 2014"
13.\" .TH WCSWIDTH 3C "Aug 14, 2002"
14.\" .SH NAME
15.\" wcswidth, wcswidth_l - number of column positions of a wide-character string
16.\" - number of column positions of a wide-character string
17.\" .SH SYNOPSIS
18.\" .LP
19.\" .nf
20.\" #include <wchar.h>
21.\" #include <xlocale.h>
22.\"
23.\" \fBint\fR \fBwcswidth\fR(\fBconst wchar_t * \fR\fIpwcs\fR, \fBsize_t\fR \fIn\fR);
24.\" .fi
25.\" .LP
26.\" \fBint\fR \fBwcswidth_l\fR(\fBconst wchar_t * \fR\fIpwcs\fR, \fBsize_t\fR \fIn\fR
27.\" .fi
28.\"
29.\" .SH DESCRIPTION
30.\" .sp
31.\" .LP
32.\" The \fBwcswidth() function determines the number of column positions
33.\" required for \fIn wide-character codes (or fewer than \fIn wide-character
34.\" codes if a null wide-character code is encountered before \fIn or
35.\" wide-character codes are exhausted) in the string pointed to by \fIpwcs.
36.\" .SH RETURN VALUES
37.\" .sp
38.\" .LP
39.\" The \fBwcswidth() function either returns \fB0 (if \fIpwcs points to a
40.\" null wide-character code), or returns the number of column positions to be
41.\" occupied by the wide-character string pointed to by \fIpwcs, or returns
42.\" \fB\mi (if any of the first \fIn wide-character codes in the
43.\" wide-character string pointed to by \fIpwcs is not a printing wide-character
44.\" code).
45.\" .LP
46.\" The function \fBwcwidth_l() behaves identically to \fBwcwidth(), except
47.\" instead of operating in the current environemtn, it operates in the environment
48.\" specified by \fIloc.
49.\" .SH ERRORS
50.\" .sp
51.\" .LP
52.\" No errors are defined.
53.\" .SH ATTRIBUTES
54.\" .sp
55.\" .LP

```

54 See `\fBattributes\fR(5)` for descriptions of the following attributes:

46 .sp

48 .sp

55 .TS

56 box;

57 c | c

58 l | l .

59 ATTRIBUTE TYPE ATTRIBUTE VALUE

60 \_

61 CSI Enabled

62 \_

63 Interface Stability Standard

64 \_

65 MT-Level MT-Safe with exceptions

66 .TE

68 .SH SEE ALSO

63 .sp

69 .LP

70 `\fBnewlocale\fR(3C)`, `\fBsetlocale\fR(3C)`, `\fBuselocale\fR(3C)`,

71 `\fBwcwidth\fR(3C)`, `\fBattributes\fR(5)`, `\fBstandards\fR(5)`

65 `\fBsetlocale\fR(3C)`, `\fBwcwidth\fR(3C)`, `\fBattributes\fR(5)`, `\fBstandards\fR(5)`

\*\*\*\*\*

3704 Fri Jul 11 11:43:11 2014

new/usr/src/man/man3c/wctob.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions Co
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCTOB 3C "Jun 25, 2014"
10 .TH WCTOB 3C "Aug 14, 2002"
12 .SH NAME
13 wctob, wctob_l \- wide-character to single-byte conversion
12 wctob \- wide-character to single-byte conversion
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <stdio.h>
18 #include <wchar.h>

20 \fBint\fR \fBwctob\fR(\fBwint_t\fR \fBIc\fR);
21 .fi
22 .LP
23 .nf
24 #include <stdio.h>
25 #include <wchar.h>
26 #include <xlocale.h>

28 \fBint\fR \fBwctob_l\fR(\fBwint_t\fR \fBIc\fR, \fBlocale_t\fR \fBIloc\fR);
29 .fi
30 .SH DESCRIPTION
23 .sp
31 .LP
32 The \fBwctob()\fR function determines whether \fBIc\fR corresponds to a member
33 of the extended character set whose character representation is a single byte
34 when in the initial shift state.
28 .sp
35 .LP
36 The behavior of \fBwctob()\fR is affected by the \fBLC_CTYPE\fR category of the
30 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
37 current locale. See \fBenviron\fR(5)
38 .LP
39 The function \fBwctob_l()\fR behaves identically to \fBwctob()\fR, except
40 instead of operating in the current locale, it operates in the locale
41 specified by \fBIloc\fR.
42 .SH RETURN VALUES
33 .sp
43 .LP
44 These functions return \fBEOF\fR if \fBIc\fR does not correspond to
45 a character with length one in the initial shift state. Otherwise, they return
35 The \fBwctob()\fR function returns \fBEOF\fR if \fBIc\fR does not correspond to
36 a character with length one in the initial shift state. Otherwise, it returns
46 the single-byte representation of that character.
47 .SH ERRORS
39 .sp
48 .LP
49 No errors are defined.

```

```

50 .SH ATTRIBUTES
43 .sp
51 .LP
52 See \fBattributes\fR(5) for descriptions of the following attributes:
46 .sp

48 .sp
53 .TS
54 box;
55 c | c
56 l | l .
57 ATTRIBUTE TYPE      ATTRIBUTE VALUE
58 -
59 Interface Stability      See below
55 Interface Stability      Standard
60 -
61 MT-Level                  MT-Safe
57 MT-Level                  MT-Safe with exceptions
62 .TE

64 .LP
65 The \fBwctob()\fR function is Standard. The \fBwctob_l()\fR function
66 is Uncommitted.
67 .SH SEE ALSO
61 .sp
68 .LP
69 \fBbtowc\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
70 \fBattributes\fR(5), \fBenviron\fR(5), \fBstandards\fR(5)
63 \fBbtowc\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5), \fBenviron\fR(5),
64 \fBstandards\fR(5)
65 .SH NOTES
66 .sp
67 .LP
68 The \fBwctob()\fR function can be used safely in multithreaded applications, as
69 long as \fBsetlocale\fR(3C) is not being called to change the locale.

```

\*\*\*\*\*

4213 Fri Jul 11 11:43:11 2014

new/usr/src/man/man3c/wctomb.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved. Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCTOMB 3C "Jun 25, 2014"
10 .TH WCTOMB 3C "Aug 14, 2002"
12 .SH NAME
13 wctomb, wctomb_l \- convert a wide-character code to a character
12 wctomb \- convert a wide-character code to a character
14 .SH SYNOPSIS
15 .LP
16 .nf
17 #include <stdlib.h>

19 \fBint\fR \fBwctomb\fR(\fBchar * \fR\fIis\fR, \fBwchar_t\fR \fIwchar\fR);
20 .fi
21 .LP
22 .nf
23 #include <stdlib.h>
24 #include <xlocale.h>

26 \fBint\fR \fBwctomb_l\fR(\fBchar * \fR\fIis\fR, \fBwchar_t\fR \fIwchar\fR, \fBlocale_t\fR \fIloc\fR);
27 .fi
28 .SH DESCRIPTION
29 .sp
29 .LP
30 The \fBwctomb()\fR function determines the number of bytes needed to represent
31 the character corresponding to the wide-character code whose value is
32 \fIwchar\fR. It stores the character representation (possibly multiple bytes)
33 in the array object pointed to by \fIis\fR (if \fIis\fR is not a null pointer).
34 At most \fBMB_CUR_MAX\fR bytes are stored.
29 .sp
35 .LP
36 A call with \fIis\fR as a null pointer causes this function to return \fB0\fR.
37 The behavior of this function is affected by the \fBLC_CTYPE\fR category of the
38 current locale.
39 .LP
40 The function \fBwctomb_l()\fR behaves identically to \fBwctomb()\fR, except
41 instead of operating in the current locale, it operates in the locale
42 specified by \fIloc\fR.
43 .SH RETURN VALUES
44 .sp
44 .LP
45 If \fIis\fR is a null pointer, \fBwctomb()\fR returns \fB0\fR value. If \fIis\fR
46 is not a null pointer, \fBwctomb()\fR returns \fB(mil\fR if the value of
47 \fIwchar\fR does not correspond to a valid character, or returns the number of
48 bytes that constitute the character corresponding to the value of \fIwchar\fR.
41 .sp
49 .LP
50 In no case will the value returned be greater than the value of the
51 \fBMB_CUR_MAX\fR macro.
52 .SH ERRORS

```

```

46 .sp
53 .LP
54 No errors are defined.
49 .SH USAGE
50 .sp
51 .LP
52 The \fBwctomb()\fR function can be used safely in a multithreaded application,
53 as long as \fBsetlocale\fR(3C) is not being called to change the locale.
55 .SH ATTRIBUTES
55 .sp
56 .LP
57 See \fBattributes\fR(5) for descriptions of the following attributes:
58 .sp

60 .sp
58 .TS
59 box;
60 c | c
61 l | l .
62 ATTRIBUTE TYPE ATTRIBUTE VALUE
63 -
64 CSI Enabled
65 -
66 Interface Stability See below.
69 Interface Stability Standard
67 -
68 MT-Level MT-Safe
71 MT-Level MT-Safe with exceptions
69 .TE

71 .LP
72 The \fBwctomb()\fR function is Standard. The \fBwctomb_l()\fR function
73 is Uncommitted.
74 .SH SEE ALSO
75 .sp
75 .LP
76 \fBmblen\fR(3C), \fBmbstowcs\fR(3C), \fBmbtowc\fR(3C),
77 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),
77 \fBmblen\fR(3C), \fBmbstowcs\fR(3C), \fBmbtowc\fR(3C), \fBsetlocale\fR(3C),
78 \fBwcstombs\fR(3C), \fBattributes\fR(5), \fBstandards\fR(5)

```



\*\*\*\*\*

2973 Fri Jul 11 11:43:11 2014

new/usr/src/man/man3c/wctrans.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  '\" te
2  .\"
3  .\" This file and its contents are supplied under the terms of the
4  .\" Common Development and Distribution License ("CDDL"), version 1.0.
5  .\" You may only use this file in accordance with the terms of version
6  .\" 1.0 of the CDDL.
7  .\"
8  .\" A full copy of the text of the CDDL should have accompanied this
9  .\" source. A copy of the CDDL is also available via the Internet at
10 .\" http://www.illumos.org/license/CDDL.
11 .\"
12 .\"
13 .\" Copyright (c) 2014 Joyent, Inc. All rights reserved.
14 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 .\"
16 .TH WCTRANS 3C "Jun 25, 2014"

18 .SH NAME
19 towctrans, towctrans_l, wctrans, wctrans_l \- define and perform transliteration
20 .SH SYNOPSIS
21 .LP
22 .nf
23 #include <wctype.h>

25 \fBwint_t\fR \fBtowctrans\fR(\fBwint_t\fR \fBIwc\fR, \fBwctrans_t\fR \fBIdesc\fR);
26 .fi
27 .LP
28 .nf
29 \fBwint_t\fR \fBtowctrans_l\fR(\fBwint_t\fR \fBIwc\fR, \fBwctrans_t\fR \fBIdesc\fR
30 .fi
31 .LP
32 .nf
33 \fBwctrans_t\fR \fBwctrans\fR(\fBconst char *\fR \fBtraclass\fR);
34 .fi
35 .LP
36 .nf
37 \fBwctrans_t\fR \fBwctrans_l\fR(\fBconst char *\fR \fBtraclass\fR, \fBlocale_t\fR
38 .fi
39 .SH DESCRIPTION
40 .LP
41 The functions
42 .B wctrans()
43 and
44 .B wctrans_l()
45 are used to obtain a handle to a table that maps one set of wide
46 characters to another. They return an object of type
47 .I wctrans_t
48 which can be used with the functions
49 .B towctrans()
50 and
51 .BR towctrans_l() .
52 The valid set of classes that are available depends on the locale.
53 The following names are valid in all locales:
54 .TP
55 "tolower"
56 Conversion from upper case to lower case characters.
57 .TP
58 "toupper"

```

```

59 Conversion from lower case to upper case characters.
60 .LP
61 The
62 .B towctrans()
63 and
64 .B towctrans_l()
65 functions convert the wide character
66 .IR wc
67 based on the conversion table specified by
68 .IR desc .
69 .LP
70 The functions
71 .B towctrans_l()
72 and
73 .B wctrans_l()
74 are equivalent to the functions
75 .B towctrans()
76 and
77 .BR wctrans() ,
78 but instead of operating in the current locale, they operate on the
79 locale specified by
80 .IR loc .
81 .SH RETURN VALUES
82 On successful completion,
83 .B towctrans()
84 and
85 .B towctrans_l()
86 functions return the character that corresponds to the argument passed
87 through the mapping table described by
88 .IR desc .
89 Otherwise, they return the character unchanged and set
90 .BR errno .
91 On successful completion,
92 .B wctrans()
93 and
94 .B wctrans_l()
95 functions return a non-zero identifier for
96 .IR traclass .
97 On failure, they return zero and set
98 .BR errno .
99 .SH ERRORS
100 .LP
101 The
102 .B wctrans()
103 and
104 .B wctrans_l()
105 functions will fail if:
106 .TP
107 .B EINVAL
108 The mapping class specified by
109 .I traclass
110 does not exist or is invalid.
111 .LP
112 The
113 .B towctrans()
114 and
115 .B towctrans_l()
116 functions will fail if:
117 .TP
118 .B EINVAL
119 The mapping class specified by
120 .I desc
121 is invalid.
122 .SH ATTRIBUTES
123 .TS
124 box;

```

```
125 c | c
126 l | l .
127 ATTRIBUTE TYPE ATTRIBUTE VALUE
128 _
129 Interface Stability Standard
130 _
131 MT-Level MT-Safe
132 .TE
```

```
134 .SH SEE ALSO
135 .BR newlocale (3C),
136 .BR setlocale (3C),
137 .BR tolower (3C),
138 .BR toupper (3C),
139 .BR environ (5),
140 .BR locale (5)
```

\*\*\*\*\*

4240 Fri Jul 11 11:43:12 2014

new/usr/src/man/man3c/wctype.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved Portions C
4  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
5  .\" http://www.opengroup.org/bookstore/.
6  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7  .\" This notice shall appear on any product containing this material.
8  .\" The contents of this file are subject to the terms of the Common Development
9  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
10 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
11 .TH WCTYPE 3C "Jun 25, 2014"
12 .TH WCTYPE 3C "Aug 14, 2002"
13 .SH NAME
14 wctype, wctype_l \- define character class
15 .SH SYNOPSIS
16 .LP
17 #include <wchar.h>
18
19 \fBwctype_t\fR \fBwctype\fR(\fBconst char *\fR\fIcharclass\fR);
20 .fi
21 .LP
22 .nf
23 \fBwctype_t\fR \fBwctype_l\fR(\fBconst char *\fR\fIcharclass\fR, \fBlocale_t\fR
24 .fi
25
26 .SH DESCRIPTION
27 .sp
28 The \fBwctype()\fR function is defined for valid character class names as
29 defined in the current locale. The \fIcharclass\fR is a string identifying a
30 generic character class for which codeset-specific type information is
31 required. The following character class names are defined in all locales:
32 .IP
33 .sp
34 .sp
35 .TS
36 l l l
37 l l l .
38 "alnum" "alpha" "blank"
39 "cntrl" "digit" "graph"
40 "lower" "print" "punct"
41 "space" "upper" "xdigit"
42 alnum alpha blank
43 cntrl digit graph
44 lower print punct
45 space upper xdigit
46 .TE
47
48 .sp
49 .LP
50 Additional character class names defined in the locale definition file
51 (category \fBLC_CTYPE\fR) can also be specified.
52 .sp
53 .LP
54 The function returns a value of type \fBwctype_t\fR, which can be used as the

```

```

47 second argument to subsequent calls of \fBiswctype\fR(3C). The \fBwctype()\fR
48 function determines values of \fBwctype_t\fR according to the rules of the coded
49 character set defined by character type information in the current locale
50 second argument to subsequent calls of \fBiswctype\fR(3C). \fBwctype()\fR
51 determines values of \fBwctype_t\fR according to the rules of the coded
52 character set defined by character type information in the program's locale
53 (category \fBLC_CTYPE\fR). The values returned by \fBwctype()\fR are valid
54 only in the locale, or locales with the same \fBLC_CTYPE\fR category.
55 .LP
56 The function \fBwctype_l()\fR behaves identically to \fBwctype()\fR, except
57 instead of operating in the current locale, it operates in the locale
58 specified by \fIloc\fR.
59 until a call to \fBsetlocale\fR(3C) that modifies the category \fBLC_CTYPE\fR.
60 .SH RETURN VALUES
61 .sp
62 .LP
63 These functions return \fB0\fR if the given character class name
64 is not valid for the locale (category \fBLC_CTYPE\fR); otherwise it
65 The \fBwctype()\fR function returns \fB0\fR if the given character class name
66 is not valid for the current locale (category \fBLC_CTYPE\fR); otherwise it
67 returns an object of type \fBwctype_t\fR that can be used in calls to
68 \fBiswctype\fR(3C).
69 \fBiswctype()\fR.
70 .SH ATTRIBUTES
71 .sp
72 .LP
73 See \fBattributes\fR(5) for descriptions of the following attributes:
74 .sp
75 .sp
76 .TS
77 box;
78 c | c
79 l | l .
80 ATTRIBUTE TYPE ATTRIBUTE VALUE
81 _
82 CSI Enabled
83 _
84 Interface Stability Standard
85 _
86 MT-Level MT-Safe
87 MT-Level MT-Safe with exceptions
88 .TE
89
90 .SH SEE ALSO
91 .sp
92 .LP
93 \fBiswctype\fR(3C), \fBnewlocale\fR(3C), \fBsetlocale\fR(3C),
94 \fBuselocale\fR(3C),
95 \fBattributes\fR(5),
96 \fBiswctype\fR(3C), \fBsetlocale\fR(3C), \fBattributes\fR(5),
97 \fBstandards\fR(5)

```

\*\*\*\*\*

3730 Fri Jul 11 11:43:12 2014

new/usr/src/man/man3c/wcwidth.3c

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1  \' te
2  .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  .\" Copyright (c) 1992, X/Open Company Limited. All Rights Reserved.
4  .\" Portions Copyright (c) 2002, Sun Microsystems, Inc. All Rights Reserved.
5  .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
6  .\" http://www.opengroup.org/bookstore/.
7  .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
8  .\" This notice shall appear on any product containing this material.
9  .\" The contents of this file are subject to the terms of the Common Development
10 .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
11 .\" When distributing Covered Code, include this CDDL HEADER in each file and in
12 .TH WCWIDTH 3C "Jun 25, 2014"
13 .TH WCWIDTH 3C "Aug 14, 2002"
14 .SH NAME
15 wcwidth, wcwidth_l \- number of column positions of a wide-character code
16 .SH SYNOPSIS
17 .LP
18 #include <wchar.h>
19
20 \fBint\fR \fBwcwidth\fR(\fBwchar_t\fR \fBIwc\fR);
21 .fi
22 .LP
23 .nf
24 #include <wchar.h>
25 #include <xlocale.h>
26
27 \fBint\fR \fBwcwidth_l\fR(\fBwchar_t\fR \fBIwc\fR, \fBllocale_t\fR \fBIloc\fR);
28 .fi
29 .SH DESCRIPTION
30 .sp
31 .LP
32 The \fBwcwidth()\fR function determines the number of column positions required
33 for the wide character \fBIwc\fR. The value of \fBIwc\fR must be a character
34 representable as a \fBwchar_t\fR, and must be a wide-character code
35 corresponding to a valid character in the current locale.
36 .LP
37 The function \fBwcwidth_l()\fR behaves identically to \fBwcwidth()\fR, except
38 instead of operating in the current locale, it operates in the locale
39 specified by \fBIloc\fR.
40 .SH RETURN VALUES
41 .sp
42 .LP
43 These functions either return \fB0\fR (if \fBIwc\fR is a null
44 wide-character code), or the number of column positions to be occupied
45 by the wide-character code \fBIwc\fR, or \fB\mi1\fR (if \fBIwc\fR does
46 not correspond to a printing wide-character code).
47 .SH ERRORS
48 .sp
49 .LP
50 No errors are defined.
51 .SH ATTRIBUTES
52 .sp
53 .LP

```

50 See \fBattributes\fR(5) for descriptions of the following attributes:

44 .sp

46 .sp

51 .TS

52 box;

53 c | c

54 l | l .

55 ATTRIBUTE TYPE ATTRIBUTE VALUE

56 \_

57 CSI Enabled

58 \_

59 **Interface Stability** See below.

55 *Interface Stability* Standard

60 \_

61 **MT-Level** MT-Safe

57 *MT-Level* MT-Safe with exceptions

62 .TE

64 .LP

65 The \fBwcwidth()\fR function is Standard. The \fBwcwidth\_l()\fR function

66 is Uncommitted.

67 .SH SEE ALSO

61 .sp

68 .LP

69 \fBnewlocale\fR(3C), \fBsetlocale\fR(3C), \fBuselocale\fR(3C),

70 \fBwcwidth\fR(3C), \fBattributes\fR(5),

63 \fBsetlocale\fR(3C), \fBwcwidth\fR(3C), \fBattributes\fR(5),

71 \fBstandards\fR(5)

```

*****
7622 Fri Jul 11 11:43:12 2014
new/usr/src/man/man3head/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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
9 # at http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
15 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
16 #
17 #
18 include      $(SRC)/Makefile.master
19 #
20 MANSECT=     3head
21 #
22 MANFILES=    acct.h.3head      //
23              aio.h.3head      //
24              ar.h.3head        //
25              archives.h.3head //
26              assert.h.3head   //
27              complex.h.3head  //
28              cpio.h.3head     //
29              dirent.h.3head   //
30              errno.h.3head    //
31              fcntl.h.3head    //
32              fenv.h.3head     //
33              float.h.3head    //
34              floatingpoint.h.3head //
35              fmtmsg.h.3head   //
36              fnmatch.h.3head  //
37              ftw.h.3head      //
38              glob.h.3head     //
39              grp.h.3head      //
40              iconv.h.3head    //
41              if.h.3head       //
42              in.h.3head       //
43              inet.h.3head     //
44              inttypes.h.3head //
45              ipc.h.3head     //
46              iso646.h.3head   //
47              langinfo.h.3head //
48              libgen.h.3head   //
49              libintl.h.3head  //
50              limits.h.3head   //
51              locale.h.3head   //
52              math.h.3head     //
53              mman.h.3head     //
54              monetary.h.3head //
55              mqueue.h.3head   //
56              msg.h.3head      //
57              ndbm.h.3head     //
58              netdb.h.3head    //

```

```

59              nl_types.h.3head //
60              poll.h.3head     //
61              pthread.h.3head  //
62              pwd.h.3head      //
63              regex.h.3head   //
64              resource.h.3head //
65              sched.h.3head    //
66              search.h.3head   //
67              select.h.3head   //
68              sem.h.3head      //
69              semaphore.h.3head //
70              setjmp.h.3head   //
71              shm.h.3head      //
72              siginfo.h.3head  //
73              signal.h.3head   //
74              socket.h.3head   //
75              spawn.h.3head    //
76              stat.h.3head     //
77              statvfs.h.3head  //
78              stdbool.h.3head  //
79              stddef.h.3head   //
80              stdint.h.3head   //
81              stdio.h.3head    //
82              stdlib.h.3head   //
83              string.h.3head   //
84              strings.h.3head //
85              stropts.h.3head  //
86              syslog.h.3head   //
87              tar.h.3head      //
88              tcp.h.3head      //
89              termios.h.3head  //
90              tgmath.h.3head   //
91              time.h.3head     //
92              timeb.h.3head    //
93              times.h.3head    //
94              types.h.3head    //
95              types32.h.3head  //
96              ucontext.h.3head //
97              uio.h.3head     //
98              ulimit.h.3head   //
99              un.h.3head      //
100            unistd.h.3head    //
101             utime.h.3head     //
102             utmpx.h.3head     //
103             utsname.h.3head   //
104             values.h.3head    //
105             wait.h.3head      //
106             wchar.h.3head     //
107             wctype.h.3head   //
108             wordexp.h.3head   //
109             xlocale.h.3head   //
110             wordexp.h.3head   //
111 MANLINKS=    acct.3head      //
112             aio.3head        //
113             ar.3head         //
114             archives.3head   //
115             assert.3head     //
116             complex.3head    //
117             cpio.3head       //
118             dirent.3head     //
119             errno.3head      //
120             fcntl.3head     //
121             fenv.3head       //
122             float.3head      //
123             floatingpoint.3head //

```

```

124      fmtmsg.3head      //
125      fnmatch.3head    //
126      ftw.3head         //
127      glob.3head       //
128      grp.3head        //
129      iconv.3head      //
130      if.3head         //
131      in.3head         //
132      inet.3head       //
133      inttypes.3head   //
134      ipc.3head        //
135      iso646.3head     //
136      langinfo.3head   //
137      libgen.3head     //
138      libintl.3head    //
139      limits.3head     //
140      locale.3head     //
141      math.3head       //
142      mman.3head       //
143      monetary.3head   //
144      mqueue.3head     //
145      msg.3head        //
146      ndbm.3head      //
147      netdb.3head     //
148      nl_types.3head   //
149      poll.3head       //
150      pthread.3head    //
151      pwd.3head        //
152      regex.3head     //
153      resource.3head   //
154      sched.3head     //
155      search.3head     //
156      select.3head     //
157      sem.3head        //
158      semaphore.3head //
159      setjmp.3head     //
160      shm.3head        //
161      siginfo.3head    //
162      signal.3head     //
163      socket.3head    //
164      spawn.3head     //
165      stat.3head       //
166      statvfs.3head    //
167      stdbool.3head    //
168      stddef.3head     //
169      stdint.3head     //
170      stdio.3head     //
171      stdlib.3head    //
172      string.3head     //
173      strings.3head    //
174      stropts.3head    //
175      syslog.3head    //
176      tar.3head        //
177      tcp.3head        //
178      termios.3head    //
179      tgmth.3head     //
180      time.3head      //
181      timeb.3head     //
182      times.3head     //
183      types.3head     //
184      types32.3head    //
185      ucontext.3head   //
186      uio.3head        //
187      ulimit.3head    //
188      un.3head         //
189     unistd.3head     //

```

```

190      utime.3head      //
191      utmpx.3head     //
192      utsname.3head   //
193      values.3head    //
194      wait.3head      //
195      wchar.3head     //
196      wctype.3head    //
197      wordexp.3head   //
198      xlocale.3head   //
199      wordexp.3head   //
200      acct.3head      := LINKSRC = acct.h.3head
201      aio.3head       := LINKSRC = aio.h.3head
202      ar.3head        := LINKSRC = ar.h.3head
203      archives.3head := LINKSRC = archives.h.3head
204      assert.3head   := LINKSRC = assert.h.3head
205      complex.3head  := LINKSRC = complex.h.3head
206      cpio.3head     := LINKSRC = cpio.h.3head
207      dirent.3head   := LINKSRC = dirent.h.3head
208      errno.3head    := LINKSRC = errno.h.3head
209      fcntl.3head    := LINKSRC = fcntl.h.3head
210      fenv.3head     := LINKSRC = fenv.h.3head
211      float.3head    := LINKSRC = float.h.3head
212      floatingpoint.3head := LINKSRC = floatingpoint.h.3head
213      fmtmsg.3head   := LINKSRC = fmtmsg.h.3head
214      fnmatch.3head := LINKSRC = fnmatch.h.3head
215      ftw.3head      := LINKSRC = ftw.h.3head
216      glob.3head    := LINKSRC = glob.h.3head
217      grp.3head     := LINKSRC = grp.h.3head
218      iconv.3head   := LINKSRC = iconv.h.3head
219      if.3head      := LINKSRC = if.h.3head
220      in.3head      := LINKSRC = in.h.3head
221      inet.3head    := LINKSRC = inet.h.3head
222      inttypes.3head := LINKSRC = inttypes.h.3head
223      ipc.3head     := LINKSRC = ipc.h.3head
224      iso646.3head  := LINKSRC = iso646.h.3head
225      langinfo.3head := LINKSRC = langinfo.h.3head
226      libgen.3head  := LINKSRC = libgen.h.3head
227      libintl.3head := LINKSRC = libintl.h.3head
228      limits.3head  := LINKSRC = limits.h.3head
229      locale.3head  := LINKSRC = locale.h.3head
230      math.3head    := LINKSRC = math.h.3head
231      mman.3head    := LINKSRC = mman.h.3head
232      monetary.3head := LINKSRC = monetary.h.3head
233      mqueue.3head  := LINKSRC = mqueue.h.3head
234      msg.3head     := LINKSRC = msg.h.3head
235      ndbm.3head    := LINKSRC = ndbm.h.3head
236      netdb.3head   := LINKSRC = netdb.h.3head
237      nl_types.3head := LINKSRC = nl_types.h.3head
238      poll.3head    := LINKSRC = poll.h.3head
239      pthread.3head := LINKSRC = pthread.h.3head
240      pwd.3head     := LINKSRC = pwd.h.3head
241      regex.3head   := LINKSRC = regex.h.3head
242      resource.3head := LINKSRC = resource.h.3head
243      sched.3head   := LINKSRC = sched.h.3head
244      search.3head  := LINKSRC = search.h.3head
245      select.3head  := LINKSRC = select.h.3head
246      sem.3head     := LINKSRC = sem.h.3head
247      semaphore.3head := LINKSRC = semaphore.h.3head
248      setjmp.3head  := LINKSRC = setjmp.h.3head
249      shm.3head     := LINKSRC = shm.h.3head
250      siginfo.3head := LINKSRC = siginfo.h.3head
251      signal.3head  := LINKSRC = signal.h.3head
252      socket.3head  := LINKSRC = socket.h.3head
253      spawn.3head   := LINKSRC = spawn.h.3head
254      stat.3head    := LINKSRC = stat.h.3head

```

```
255 statvfs.3head      := LINKSRC = statvfs.h.3head
256 stdbool.3head     := LINKSRC = stdbool.h.3head
257 stddef.3head      := LINKSRC = stddef.h.3head
258 stdint.3head      := LINKSRC = stdint.h.3head
259 stdio.3head       := LINKSRC = stdio.h.3head
260 stdlib.3head      := LINKSRC = stdlib.h.3head
261 string.3head      := LINKSRC = string.h.3head
262 strings.3head     := LINKSRC = strings.h.3head
263 stropts.3head     := LINKSRC = stropts.h.3head
264 syslog.3head      := LINKSRC = syslog.h.3head
265 tar.3head         := LINKSRC = tar.h.3head
266 tcp.3head         := LINKSRC = tcp.h.3head
267 termios.3head     := LINKSRC = termios.h.3head
268 tgmth.3head       := LINKSRC = tgmth.h.3head
269 time.3head        := LINKSRC = time.h.3head
270 timeb.3head       := LINKSRC = timeb.h.3head
271 times.3head       := LINKSRC = times.h.3head
272 types.3head       := LINKSRC = types.h.3head
273 types32.3head     := LINKSRC = types32.h.3head
274 ucontext.3head    := LINKSRC = ucontext.h.3head
275 uio.3head         := LINKSRC = uio.h.3head
276 ulimit.3head      := LINKSRC = ulimit.h.3head
277 un.3head          := LINKSRC = un.h.3head
278 unistd.3head      := LINKSRC = unistd.h.3head
279 utime.3head       := LINKSRC = utime.h.3head
280 utmpx.3head       := LINKSRC = utmpx.h.3head
281 utsname.3head     := LINKSRC = utsname.h.3head
282 values.3head      := LINKSRC = values.h.3head
283 wait.3head        := LINKSRC = wait.h.3head
284 wchar.3head       := LINKSRC = wchar.h.3head
285 wctype.3head     := LINKSRC = wctype.h.3head
286 wordexp.3head     := LINKSRC = wordexp.h.3head
287 xlocale.3head    := LINKSRC = xlocale.h.3head

289 .KEEP_STATE:

291 include            $(SRC)/man/Makefile.man

293 install:          $(ROOTMANFILES) $(ROOTMANLINKS)
```

```

*****
4826 Fri Jul 11 11:43:12 2014
new/usr/src/man/man3head/locale.h.3head
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \" te
2  \\ Copyright (c) 2001, The IEEE and The Open Group. All Rights Reserved. Portio
3  \\ Copyright (c) 2014, Joyent, Inc. All rights reserved.
4  \\ Copyright 2014 Garrett D'Amore <garrett@damore.org>
5  \\ Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
6  \\ http://www.opengroup.org/bookstore/.
7  \\ The Institute of Electrical and Electronics Engineers and The Open Group, ha
8  \\ This notice shall appear on any product containing this material.
9  \\ The contents of this file are subject to the terms of the Common Development
10 \\ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
11 \\ When distributing Covered Code, include this CDDL HEADER in each file and in
12 .TH LOCALE.H 3HEAD "Jun 23, 2004"
10 .TH LOCALE.H 3HEAD "Sep 10, 2004"
13 .SH NAME
14 locale.h, locale \\- category macros
15 .SH SYNOPSIS
16 .LP
17 .nf
18 #include <fBlocale.h\fr>
19 .fi

20 .SH DESCRIPTION
20 .sp
21 .LP
22 The <fBlocale.h\fr> header provides a definition for the \fBlconv\fr
23 structure, which includes the following members. (See the definition of
24 \fBLC_MONETARY\fr in \fBlocale\fr(5).)
25 .sp
26 .in +2
27 .nf
28 char      *currency_symbol
29 char      *decimal_point
30 char      frac_digits
31 char      *grouping
32 char      *int_curr_symbol
33 char      int_frac_digits
34 char      int_n_cs_precedes
35 char      int_n_sep_by_space
36 char      int_n_sign_posn
37 char      int_p_cs_precedes
38 char      int_p_sep_by_space
39 char      int_p_sign_posn
40 char      *mon_decimal_point
41 char      *mon_grouping
42 char      *mon_thousands_sep
43 char      *negative_sign
44 char      n_cs_precedes
45 char      n_sep_by_space
46 char      n_sign_posn
47 char      *positive_sign
48 char      p_cs_precedes
49 char      p_sep_by_space
50 char      p_sign_posn
51 char      *thousands_sep
52 .fi
53 .in -2

55 .sp

```

```

54 .LP
55 The <fBlocale.h\fr> header defines \fINULL\fr (as defined in <fBstddef.h\fr>)
56 and the following as macros:
57 .sp
58 .in +2
59 .nf
60 LC_ALL
61 LC_COLLATE
62 LC_CTYPE
63 LC_MESSAGES
64 LC_MONETARY
65 LC_NUMERIC
66 LC_TIME
67 .fi
68 .in -2

72 .sp
69 .LP
70 The preceding expand to distinct integer constant expressions, for use as the
71 first argument to the \fBsetlocale()\fr function. See \fBsetlocale\fr(3C).
72 .LP
73 The <fBlocale.h\fr> header also defines the following macros, for use with
74 \fBnewlocale\fr(3C):
75 .sp
76 .in +2
77 .nf
78 LC_ALL_MASK
79 LC_COLLATE_MASK
80 LC_CTYPE_MASK
81 LC_MESSAGES_MASK
82 LC_MONETARY_MASK
83 LC_NUMERIC_MASK
84 LC_TIME_MASK
85 .fi
86 .in -2
87 .LP
88 In addition, to facilitate the use of per-thread locales with the
89 .BR uselocale (3C)
90 function, the
91 .RB < locale.h >
92 header defines the following type and macro:
93 .sp
94 .in +2
95 .nf
96 locale_t
97 .fi
98 .in -2
99 .sp
100 .RS 6n
101 An opaque type that refers to a locale constructed through the use of
102 the
103 .BR newlocale (3C)
104 function.
105 .RE

107 .sp
108 .in +2
109 .nf
110 LC_GLOBAL_LOCALE
111 .fi
112 .in -2
113 .sp
114 .RS 6n
115 A macro which expands to a
116 .B locale_t
117 that represents the current global locale as set by

```



```
118 .BR setlocale (3C).
119 .RE

121 .LP
122 Additional macro definitions, beginning with the characters \fBLC_\fR and an
123 uppercase letter, can also be specified here.
124 .SH ATTRIBUTES
   81 .sp
125 .LP
126 See \fBattributes\fR(5) for descriptions of the following attributes:
   84 .sp

   86 .sp
127 .TS
128 box;
129 c | c
130 l | l .
131 ATTRIBUTE TYPE ATTRIBUTE VALUE
132 _
133 Interface Stability Standard
134 .TE

136 .SH SEE ALSO
137 .sp
138 .LP
139 .BR newlocale (3C),
140 .BR setlocale (3C),
141 .BR uselocale (3C),
142 .BR localeconv (3C),
143 .BR stddef.h (3HEAD),
144 .BR attributes (5),
145 .BR locale (5),
146 .BR standards (5)
   99 \fBsetlocale\fR(3C), \fBlocaleconv\fR(3C), \fBstddef.h\fR(3HEAD),
  100 \fBattributes\fR(5), \fBlocale\fR(5), \fBstandards\fR(5)
```

```
*****
```

```
1421 Fri Jul 11 11:43:12 2014
```

```
new/usr/src/man/man3head/xlocale.h.3head
```

```
2964 need POSIX 2008 locale object support
```

```
Reviewed by: Robert Mustacchi <rm@joyent.com>
```

```
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
```

```
Approved by: TBD
```

```
*****
```

```
1  '\" te
2  .\"
3  .\" This file and its contents are supplied under the terms of the
4  .\" Common Development and Distribution License ("CDDL"), version 1.0.
5  .\" You may only use this file in accordance with the terms of version
6  .\" 1.0 of the CDDL.
7  .\"
8  .\" A full copy of the text of the CDDL should have accompanied this
9  .\" source. A copy of the CDDL is also available via the Internet at
10 .\" http://www.illumos.org/license/CDDL.
11 .\"
12 .\"
13 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
14 .\"
15 .TH XLOCALE.H 3HEAD "Jun 26, 2014"
16 .SH NAME
17 xlocale.h, xlocale - extended locale support
18 .SH SYNOPSIS
19 .LP
20 .nf
21 #include <fBxlocale.h\fR>
22 .fi
23 .SH DESCRIPTION
24 .LP
25 The
26 .RB < xlocale.h >
27 header supplies prototypes for various functions that operate
28 using the
29 .B locale_t
30 structure but which are not specified as part of any formal standard.
31 Their location in this header follows the convention established by
32 other operating systems.
33 .LP
34 As none of the definitions supplied in this header, nor this
35 header itself, are specified in any standard, their use should
36 be avoided in portable applications.
37 .SH ATTRIBUTES
38 .LP
39 See
40 .BR attributes (5)
41 for descriptions of the following attributes:
42 .TS
43 box;
44 c | c
45 l | l .
46 ATTRIBUTE TYPE    ATTRIBUTE VALUE
47 -
48 Interface Stability    Uncommitted
49 .TE

51 .SH SEE ALSO
52 .LP
53 .BR newlocale (3C),
54 .BR setlocale (3C),
55 .BR uselocale (3C),
56 .BR locale (3HEAD),
57 .BR attributes (5),
58 .BR locale (5),
```

```
59 .BR standards (5)
```

\*\*\*\*\*

33980 Fri Jul 11 11:43:12 2014

new/usr/src/man/man3lib/libc.3lib

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```

1 \" te
2 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
3 .\" Copyright (c) 2009, Sun Microsystems, Inc. All rights reserved.
4 .\" The contents of this file are subject to the terms of the Common Development
5 .\" See the License for the specific language governing permissions and limitat
6 .\" the fields enclosed by brackets \"[]\" replaced with your own identifying info
7 .\" Copyright 2011 by Delphix. All rights reserved.
8 .TH LIBC 3LIB \"Jul 1, 2014\"
9 .SH LIBC 3LIB \"Sep 15, 2009\"
10 .SH NAME
11 libc \- C library
12 .SH DESCRIPTION
13 .sp
14 .LP
15 Functions in this library provide various facilities defined by System V, ANSI
16 C, POSIX, and so on. See \fBstandards\fR(5). In addition, those facilities
17 previously defined in the internationalization and the wide-character libraries
18 are now defined in this library, as are the facilities previously defined in
19 the multithreading libraries, \fBlibthread\fR and \fBlibpthread\fR.
20 .SH INTERFACES
21 .sp
22 .LP
23 The shared object \fBlibc.so.1\fR provides the public interfaces defined below.
24 See \fBintro\fR(3) for additional information on shared object interfaces.
25 .sp
26 .sp
27 .TS
28 l l
29 l l .
30 \fB__locl\fR \fB__errno\fR
31 \fB__builtin_alloca\fR \fB__ctype\fR
32 \fB__fbufsize\fR \fB__flbf\fR
33 \fB__flt_rounds\fR \fB__fpending\fR
34 \fB__fpurge\fR \fB__freadable\fR
35 \fB__freading\fR \fB__fsetlocking\fR
36 \fB__fwritable\fR \fB__fwriting\fR
37 \fB__huge_val\fR \fB__iob\fR
38 \fB__locl\fR \fB__major\fR
39 \fB__makedev\fR \fB__mb_cur_max_1\fR
40 \fB__minor\fR
41 \fB__nsw_extended_action\fR \fB__nsw_freeconfig\fR
42 \fB__nsw_getconfig\fR \fB__posix_asctime_r\fR
43 \fB__posix_ctime_r\fR \fB__posix_getgrgid_r\fR
44 \fB__posix_getgrnam_r\fR \fB__posix_getlogin_r\fR
45 \fB__posix_getpwnam_r\fR \fB__posix_getpwuid_r\fR
46 \fB__posix_sigwait\fR \fB__posix_ttyname_r\fR
47 \fB__prioctl\fR \fB__prioctlset\fR
48 \fB__pthread_cleanup_pop\fR \fB__pthread_cleanup_push\fR
49 \fB__sysconf_xpg5\fR \fB__xpg4\fR
50 \fB__xpg4_putmsg\fR \fB__xpg4_putpmsg\fR
51 \fB_Exit\fR \fB_altzone\fR
52 \fB_assert\fR \fB_cleanup\fR
53 \fB__ctype\fR \fB__daylight\fR
54 \fB__environ\fR \fB__exit\fR
55 \fB__exithandle\fR \fB__filbuf\fR
56 \fB__flsbuf\fR \fB__flushlbf\fR

```

```

57 \fB_getdate_err\fR \fB_getdate_err_addr\fR
58 \fB_iob\fR \fB_isnan\fR
59 \fB_isnand\fR \fB_lwp_cond_broadcast\fR
60 \fB_lwp_cond_reltimedwait\fR \fB_lwp_cond_signal\fR
61 \fB_lwp_cond_timedwait\fR \fB_lwp_cond_wait\fR
62 \fB_lwp_continue\fR \fB_lwp_info\fR
63 \fB_lwp_kill\fR \fB_lwp_mutex_lock\fR
64 \fB_lwp_mutex_trylock\fR \fB_lwp_mutex_unlock\fR
65 \fB_lwp_self\fR \fB_lwp_sema_init\fR
66 \fB_lwp_sema_post\fR \fB_lwp_sema_trywait\fR
67 \fB_lwp_sema_wait\fR \fB_lwp_suspend\fR
68 \fB_lwp_suspend2\fR \fB_modf\fR
69 \fB_nextafter\fR \fB_nsc_trydoorcall\fR
70 \fB_nss_XbyY_buf_alloc\fR \fB_nss_XbyY_buf_free\fR
71 \fB_nss_netdb_aliases\fR \fB_numeric\fR
72 \fB_scalb\fR \fB_sibuf\fR
73 \fB_sobuf\fR \fB_stack_grow\fR
74 \fB_sys_buslist\fR \fB_sys_cldlist\fR
75 \fB_sys_fpelist\fR \fB_sys_illlist\fR
76 \fB_sys_segvlst\fR \fB_sys_siginfoflst\fR
77 \fB_sys_siglist\fR \fB_sys_siglistn\fR
78 \fB_sys_siglistp\fR \fB_sys_traplist\fR
79 \fB_timezone\fR \fB_tolower\fR
80 \fB_toupper\fR \fB_tzname\fR
81 \fB_xftw\fR \fB_xfR
82 \fBa64l\fR \fBabort\fR
83 \fBabsl\fR \fBaccess\fR
84 \fBacct\fR \fBacl\fR
85 \fBaddrtosymstr\fR \fBaddsevl\fR
86 \fBaddseverity\fR \fBadjtime\fR
87 \fBaio_cancel\fR \fBaio_error\fR
88 \fBaio_fsync\fR \fBaio_read\fR
89 \fBaio_return\fR \fBaio_suspend\fR
90 \fBaio_waitn\fR \fBaio_write\fR
91 \fBaioCancel\fR \fBaioread\fR
92 \fBaiowait\fR \fBaiowrite\fR
93 \fBbalarm\fR \fBbalphasort\fR
94 \fBbaltzone\fR \fBbasctime\fR
95 \fBbasctime\fR \fBbasctime_r\fR
96 \fBbasprintf\fR
97 \fBbatexit\fR \fBbatof\fR
98 \fBbatoi\fR \fBbatol\fR
99 \fBbatoll\fR \fBbatomic_add_16\fR
100 \fBbatomic_add_16_nv\fR \fBbatomic_add_32\fR
101 \fBbatomic_add_32_nv\fR \fBbatomic_add_64\fR
102 \fBbatomic_add_64_nv\fR \fBbatomic_add_8\fR
103 \fBbatomic_add_8_nv\fR \fBbatomic_add_char\fR
104 \fBbatomic_add_char_nv\fR \fBbatomic_add_int\fR
105 \fBbatomic_add_int_nv\fR \fBbatomic_add_long\fR
106 \fBbatomic_add_long_nv\fR \fBbatomic_add_ptr\fR
107 \fBbatomic_add_ptr_nv\fR \fBbatomic_add_short\fR
108 \fBbatomic_add_short_nv\fR \fBbatomic_and_16\fR
109 \fBbatomic_and_16_nv\fR \fBbatomic_and_32\fR
110 \fBbatomic_and_32_nv\fR \fBbatomic_and_64\fR
111 \fBbatomic_and_64_nv\fR \fBbatomic_and_8\fR
112 \fBbatomic_and_8_nv\fR \fBbatomic_and_uchar\fR
113 \fBbatomic_and_uchar_nv\fR \fBbatomic_and_uint\fR
114 \fBbatomic_and_uint_nv\fR \fBbatomic_and_ulong\fR
115 \fBbatomic_and_ulong_nv\fR \fBbatomic_and_ushort\fR
116 \fBbatomic_and_ushort_nv\fR \fBbatomic_cas_16\fR
117 \fBbatomic_cas_32\fR \fBbatomic_cas_64\fR
118 \fBbatomic_cas_8\fR \fBbatomic_cas_ptr\fR
119 \fBbatomic_cas_uchar\fR \fBbatomic_cas_uint\fR
120 \fBbatomic_cas_ulong\fR \fBbatomic_cas_ushort\fR
121 \fBbatomic_clear_long_excl\fR \fBbatomic_dec_16\fR
122 \fBbatomic_dec_16_nv\fR \fBbatomic_dec_32\fR

```

```

123 \fBatonic_dec_32_nv\fr \fBatonic_dec_64\fr
124 \fBatonic_dec_64_nv\fr \fBatonic_dec_8\fr
125 \fBatonic_dec_8_nv\fr \fBatonic_dec_ptr\fr
126 \fBatonic_dec_ptr_nv\fr \fBatonic_dec_uchar\fr
127 \fBatonic_dec_uchar_nv\fr \fBatonic_dec_uint\fr
128 \fBatonic_dec_uint_nv\fr \fBatonic_dec_ulong\fr
129 \fBatonic_dec_ulong_nv\fr \fBatonic_dec_ushort\fr
130 \fBatonic_dec_ushort_nv\fr \fBatonic_inc_16\fr
131 \fBatonic_inc_16_nv\fr \fBatonic_inc_32\fr
132 \fBatonic_inc_32_nv\fr \fBatonic_inc_64\fr
133 \fBatonic_inc_64_nv\fr \fBatonic_inc_8\fr
134 \fBatonic_inc_8_nv\fr \fBatonic_inc_ptr\fr
135 \fBatonic_inc_ptr_nv\fr \fBatonic_inc_uchar\fr
136 \fBatonic_inc_uchar_nv\fr \fBatonic_inc_uint\fr
137 \fBatonic_inc_uint_nv\fr \fBatonic_inc_ulong\fr
138 \fBatonic_inc_ulong_nv\fr \fBatonic_inc_ushort\fr
139 \fBatonic_inc_ushort_nv\fr \fBatonic_or_16\fr
140 \fBatonic_or_16_nv\fr \fBatonic_or_32\fr
141 \fBatonic_or_32_nv\fr \fBatonic_or_64\fr
142 \fBatonic_or_64_nv\fr \fBatonic_or_8\fr
143 \fBatonic_or_8_nv\fr \fBatonic_or_uchar\fr
144 \fBatonic_or_uchar_nv\fr \fBatonic_or_uint\fr
145 \fBatonic_or_uint_nv\fr \fBatonic_or_ulong\fr
146 \fBatonic_or_ulong_nv\fr \fBatonic_or_ushort\fr
147 \fBatonic_or_ushort_nv\fr \fBatonic_set_long_excl\fr
148 \fBatonic_swap_16\fr \fBatonic_swap_32\fr
149 \fBatonic_swap_64\fr \fBatonic_swap_8\fr
150 \fBatonic_swap_ptr\fr \fBatonic_swap_uchar\fr
151 \fBatonic_swap_uint\fr \fBatonic_swap_ulong\fr
152 \fBatonic_swap_ushort\fr \fBatropen\fr
153 \fBbacktrace\fr \fBbacktrace_symbols\fr
154 \fBbacktrace_symbols_fd\fr \fBbasename\fr
155 \fBbcmp\fr \fBbcopy\fr
156 \fBbindtextdomain\fr \fBbind_textdomain_codeset\fr
157 \fBbrk\fr \fBbsd_signal\fr
158 \fBbsearch\fr
159 \fBbtowc\fr \fBbtowc_l\fr
156 \fBbsearch\fr \fBbtowc\fr
160 \fBbzero\fr \fBcalloc\fr
161 \fBcatclose\fr \fBcatgets\fr
162 \fBcatopen\fr \fBcfgetispeed\fr
163 \fBcfgetospeed\fr \fBcfsetispeed\fr
164 \fBcfsetospeed\fr \fBcftime\fr
165 \fBchdir\fr \fBchmod\fr
166 \fBchown\fr \fBchroot\fr
167 \fBclearerr\fr \fBclock\fr
168 \fBclock_getres\fr \fBclock_gettime\fr
169 \fBclock_nanosleep\fr \fBclock_settime\fr
170 \fBclose\fr \fBclosedir\fr
171 \fBclosefrom\fr \fBcloselog\fr
172 \fBcond_broadcast\fr \fBcond_destroy\fr
173 \fBcond_init\fr \fBcond_reltimedwait\fr
174 \fBcond_signal\fr \fBcond_timedwait\fr
175 \fBcond_wait\fr \fBconfstr\fr
176 \fBcreat\fr \fBcrypt\fr
177 \fBcrypt_genhash_impl\fr \fBcrypt_gensalt\fr
178 \fBcrypt_gensalt_impl\fr \fBcsetcol\fr
179 \fBcsetlen\fr \fBctermid\fr
180 \fBctermid_r\fr \fBctime\fr
181 \fBctime_r\fr \fBcuserid\fr
182 \fBdaemon\fr
183 \fBdaylight\fr \fBdbm_clearerr\fr
184 \fBdbm_close\fr \fBdbm_delete\fr
185 \fBdbm_error\fr \fBdbm_fetch\fr
186 \fBdbm_firstkey\fr \fBdbm_nextkey\fr
187 \fBdbm_open\fr \fBdbm_store\fr

```

```

188 \fBdcgettext\fr \fBdcngettext\fr
189 \fBdecimal_to_double\fr \fBdecimal_to_extended\fr
190 \fBdecimal_to_quadruple\fr \fBdecimal_to_single\fr
191 \fBdgettext\fr \fBdifftime\fr
192 \fBdirectio\fr \fBdirfd\fr
193 \fBdirname\fr \fBdiv\fr
194 \fBdladdr\fr \fBdladdr1\fr
195 \fBdlclose\fr \fBdlclose\fr
196 \fBdlerror\fr \fBdlinfo\fr
197 \fBdlmopen\fr \fBdlopen\fr
198 \fBdlsym\fr \fBdngettext\fr
199 \fBdoor_bind\fr \fBdoor_call\fr
200 \fBdoor_create\fr \fBdoor_cred\fr
201 \fBdoor_getparam\fr \fBdoor_info\fr
202 \fBdoor_return\fr \fBdoor_revoke\fr
203 \fBdoor_server_create\fr \fBdoor_setparam\fr
204 \fBdoor_ucred\fr \fBdoor_unbind\fr
205 \fBdouble_to_decimal\fr \fBdrand48\fr
206 \fBdup\fr \fBdup2\fr
207 \fBduplocale\fr
208 \fBconvert\fr \fBecvt\fr
209 \fBenable_extended_FILE_stdio\fr
210 \fBencrypt\fr \fBendgrent\fr
211 \fBendnetgrent\fr \fBendpwent\fr
212 \fBendspent\fr \fBendusershell\fr
213 \fBendutent\fr \fBendutxent\fr
214 \fBenvron\fr \fBerand48\fr
215 \fBerr\fr \fBerrno\fr
216 \fBerrx\fr \fBeuocol\fr
217 \fBeuclen\fr \fBeucocol\fr
218 \fBexecl\fr \fBexecl\fr
219 \fBexeclp\fr \fBexecv\fr
220 \fBexecve\fr \fBexecvp\fr
221 \fBexit\fr \fBextended_to_decimal\fr
222 \fBfaccessat\fr
223 \fBfacl\fr \fBfattach\fr
224 \fBfchdir\fr \fBfchmod\fr
225 \fBfchown\fr \fBfchownat\fr
226 \fBfchroot\fr \fBfclose\fr
227 \fBfcntl\fr \fBfconvert\fr
228 \fBfcvt\fr \fBfdatasync\fr
229 \fBfdetach\fr
230 \fBfdopen\fr \fBfdopendir\fr
231 \fBfdwalk\fr \fBfeof\fr
232 \fBferror\fr \fBfflush\fr
233 \fBffs\fr \fBfgetatr\fr
234 \fBfgetc\fr
235 \fBfgetgrent\fr \fBfgetgrent_r\fr
236 \fBfgetpos\fr \fBfgetpwent\fr
237 \fBfgetpwent_r\fr \fBfgets\fr
238 \fBfgetspent\fr \fBfgetspent_r\fr
239 \fBfgetwc\fr \fBfgetwc_l\fr
240 \fBfgetws\fr
235 \fBfgetwc\fr \fBfgetws\fr
241 \fBfile_to_decimal\fr \fBfileno\fr
242 \fBfinite\fr \fBflockfile\fr
243 \fBfmtmsg\fr \fBfnmatch\fr
244 \fBfopen\fr \fBfork\fr
245 \fBfork1\fr \fBforkall\fr
246 \fBforkallx\fr \fBforkx\fr
247 \fBfpathconf\fr \fBfpclass\fr
248 \fBfpgetmask\fr \fBfpgetround\fr
249 \fBfpgetsticky\fr \fBfprintf\fr
250 \fBfpsetmask\fr \fBfpsetround\fr
251 \fBfpsetsticky\fr \fBfputc\fr
252 \fBfputs\fr \fBfputwc\fr

```

```

253 \fBfputws\fr \fBfread\fr
254 \fBffree\fr \fBffreelocale\fr
255 \fBffreopen\fr
249 \fBffree\fr \fBffreopen\fr
256 \fBfexp\fr \fBfscanf\fr
257 \fBfseek\fr \fBfseeko\fr
258 \fBfsetattr\fr
259 \fBfsetpos\fr \fBfstat\fr
260 \fBfstatat\fr \fBfstatfs\fr
261 \fBfstatvfs\fr \fBfsync\fr
262 \fBftell\fr \fBftello\fr
263 \fBftime\fr \fBftok\fr
264 \fBftruncate\fr \fBftrylockfile\fr
265 \fBftw\fr \fBfunc_to_decimal\fr
266 \fBfunlockfile\fr \fBfutimens\fr
267 \fBfutimesat\fr
268 \fBfwide\fr \fBfwprintf\fr
269 \fBfwrite\fr \fBfwscanf\fr
270 \fBgconvert\fr \fBgcvt\fr
271 \fBgetacct\fr \fBgetatrat\fr
272 \fBgetc\fr
273 \fBgetc_unlocked\fr \fBgetchar\fr
274 \fBgetchar_unlocked\fr \fBgetcontext\fr
275 \fBgetcpuid\fr \fBgetcwd\fr
276 \fBgetdate\fr \fBgetdate_err\fr
277 \fBgetdents\fr \fBgetdtablesize\fr
278 \fBgetegid\fr \fBgetenv\fr
279 \fBgeteuid\fr \fBgetexecname\fr
280 \fBgetextmntent\fr \fBgetgid\fr
281 \fBgetgrent\fr \fBgetgrent_r\fr
282 \fBgetgrgid\fr \fBgetgrgid_r\fr
283 \fBgetgrnam\fr \fBgetgrnam_r\fr
284 \fBgetgroups\fr \fBgethome\group\fr
285 \fBgethostid\fr \fBgethostname\fr
286 \fBgethrtime\fr \fBgethrtime\fr
287 \fBgetisax\fr \fBgetitimer\fr
288 \fBgetloadavg\fr \fBgetlogin\fr
289 \fBgetlogin_r\fr \fBgetmntany\fr
290 \fBgetmntent\fr \fBgetmsg\fr
291 \fBgetnetgrent\fr \fBgetnetgrent_r\fr
292 \fBgetopt\fr \fBgetopt_clip\fr
293 \fBgetopt_long\fr \fBgetopt_long_only\fr
294 \fBgetpagesize\fr \fBgetpagesizes\fr
295 \fBgetpass\fr \fBgetpassphrase\fr
296 \fBgetpeerucred\fr \fBgetpflags\fr
297 \fBgetpgid\fr \fBgetpgrp\fr
298 \fBgetpid\fr \fBgetpmsg\fr
299 \fBgetppid\fr \fBgetppriv\fr
300 \fBgetpriority\fr \fBgetprojid\fr
301 \fBgetpw\fr \fBgetpwent\fr
302 \fBgetpwent_r\fr \fBgetpwnam\fr
303 \fBgetpwnam_r\fr \fBgetpwuid\fr
304 \fBgetpwuid_r\fr \fBgetrctl\fr
305 \fBgetrlimit\fr \fBgetrusage\fr
306 \fBgets\fr \fBgetsid\fr
307 \fBgetspent\fr \fBgetspent_r\fr
308 \fBgetspnam\fr \fBgetspnam_r\fr
309 \fBgetsubopt\fr \fBgettaskid\fr
310 \fBgetttext\fr \fBgettimeofday\fr
311 \fBgetttxt\fr \fBgetuid\fr
312 \fBgetusershell\fr \fBgetustack\fr
313 \fBgetutent\fr \fBgetutid\fr
314 \fBgetutline\fr \fBgetutmp\fr
315 \fBgetutmpx\fr \fBgetutxent\fr
316 \fBgetutxid\fr \fBgetutxline\fr
317 \fBgetvfsany\fr \fBgetvfssent\fr

```

```

318 \fBgetvfssfile\fr \fBgetvfsspec\fr
319 \fBgetw\fr
320 \fBgetwc\fr \fBgetwc_l\fr
321 \fBgetwchar\fr \fBgetwchar_l\fr
322 \fBgetwd\fr
323 \fBgetw\fr \fBgetwc\fr
324 \fBgetwchar\fr \fBgetwd\fr
323 \fBgetwidth\fr \fBgetws\fr
324 \fBgetzoneid\fr \fBgetzoneidbyname\fr
325 \fBgetzonenamebyid\fr \fBglob\fr
326 \fBglobfree\fr \fBgmtime\fr
327 \fBgmtime_r\fr \fBgrantpt\fr
328 \fBgsignal\fr \fBhasmntopt\fr
329 \fBhcreate\fr \fBhdestroy\fr
330 \fBhsearch\fr \fBiconv\fr
331 \fBiconv_close\fr \fBiconv_open\fr
332 \fBimaxabs\fr \fBimaxdiv\fr
333 \fBindex\fr \fBinitgroups\fr
334 \fBinitstate\fr \fBinnnetgr\fr
335 \fBinsque\fr \fBioctl\fr
336 \fBis_system_labeled\fr
337 \fBisaexec\fr
338 \fBisalnum\fr \fBisalnum_l\fr
339 \fBisalpha\fr \fBisalpha_l\fr
340 \fBisascii\fr \fBisastream\fr
341 \fBisatty\fr
342 \fBisblank\fr \fBisblank_l\fr
343 \fBiscntrl\fr \fBiscntrl_l\fr
344 \fBisdigit\fr \fBisdigit_l\fr
345 \fBisenglish\fr
346 \fBisgraph\fr \fBisgraph_l\fr
347 \fBisideogram\fr
348 \fBislower\fr \fBislower_l\fr
349 \fBisnan\fr
329 \fBisaexec\fr \fBisalnum\fr
330 \fBisalpha\fr \fBisascii\fr
331 \fBisastream\fr \fBisatty\fr
332 \fBisblank\fr \fBiscntrl\fr
333 \fBisdigit\fr \fBisenglish\fr
334 \fBisgraph\fr \fBisideogram\fr
335 \fBislower\fr \fBisnan\fr
350 \fBisnand\fr \fBisnanf\fr
351 \fBisnumber\fr \fBisphonogram\fr
352 \fBisprint\fr \fBisprint_l\fr
353 \fBispunct\fr \fBispunct_l\fr
354 \fBissetugid\fr
355 \fBissspace\fr \fBissspace_l\fr
356 \fBissspecial\fr
357 \fBisupper\fr \fBisupper_l\fr
358 \fBiswalnum\fr \fBiswalnum_l\fr
359 \fBiswalphal\fr \fBiswalphal_l\fr
360 \fBiswblank\fr \fBiswblank_l\fr
361 \fBiswcntrl\fr \fBiswcntrl_l\fr
362 \fBiswctype\fr \fBiswctype_l\fr
363 \fBiswdigit\fr \fBiswdigit_l\fr
364 \fBiswideogram\fr \fBiswideogram_l\fr
365 \fBiswgraph\fr \fBiswgraph_l\fr
366 \fBiswhexnumber\fr \fBiswhexnumber_l\fr
367 \fBiswlower\fr \fBiswlower_l\fr
368 \fBiswnumber\fr \fBiswnumber_l\fr
369 \fBiswphonogram\fr \fBiswphonogram_l\fr
370 \fBiswprint\fr \fBiswprint_l\fr
371 \fBiswpunct\fr \fBiswpunct_l\fr
372 \fBiswspace\fr \fBiswspace_l\fr
373 \fBiswsspecial\fr \fBiswsspecial_l\fr
374 \fBiswupper\fr \fBiswupper_l\fr

```

```

375 \fBiswxdigit\fr \fBiswxdigit_l\fr
376 \fBisxdigit\fr \fBisxdigit_l\fr
338 \fBisprint\fr \fBispunct\fr
339 \fBissetugid\fr \fBissspace\fr
340 \fBisspecial\fr \fBisupper\fr
341 \fBiswalnum\fr \fBiswalpha\fr
342 \fBiswblank\fr \fBiswcntrl\fr
343 \fBiswctype\fr \fBiswdigit\fr
344 \fBiswgraph\fr \fBiswlower\fr
345 \fBiswprint\fr \fBiswpunct\fr
346 \fBiswspace\fr \fBiswupper\fr
347 \fBiswxdigit\fr \fBisxdigit\fr
377 \fBjrand48\fr \fBkill\fr
378 \fBkillpg\fr \fB164a\fr
379 \fBblabs\fr \fB1add\fr
380 \fBblchown\fr \fB1ckpww\fr
381 \fBblcong48\fr \fB1dexp\fr
382 \fBldivide\fr \fB1expl0\fr
383 \fB1find\fr \fB1fmt\fr
384 \fB1ink\fr \fB1io_listio\fr
385 \fB1llabs\fr
386 \fB1lldiv\fr \fB1llog10\fr
387 \fB1llseek\fr \fB1lltostr\fr
388 \fB1localeconv\fr \fB1localtime\fr
389 \fB1localtime_r\fr \fB1lockf\fr
390 \fB1logb\fr \fB1lone\fr
391 \fB1longjmp\fr \fB1lrand48\fr
392 \fB1search\fr \fB1seek\fr
393 \fB1shiftl\fr \fB1stat\fr
394 \fB1sub\fr \fB1ten\fr
395 \fB1zero\fr \fB1madvise\fr
396 \fB1makecontext\fr \fB1makeutx\fr
397 \fB1malloc\fr
398 \fB1mblen\fr \fB1mblen_l\fr
399 \fB1mbrlen\fr \fB1mbrlen_l\fr
400 \fB1mbrtowc\fr \fB1mbrtowc_l\fr
401 \fB1mbsinit\fr \fB1mbsinit_l\fr
402 \fB1mbsnrtowcs\fr \fB1mbsnrtowcs_l\fr
403 \fB1mbsrtowcs\fr \fB1mbsrtowcs_l\fr
404 \fB1mbstowcs\fr \fB1mbstowcs_l\fr
405 \fB1mbtowc\fr \fB1mbtowc_l\fr
368 \fB1malloc\fr \fB1mblen\fr
369 \fB1mbrlen\fr \fB1mbrtowc\fr
370 \fB1mbsinit\fr \fB1mbsrtowcs\fr
371 \fB1mbstowcs\fr \fB1mbtowc\fr
406 \fB1memalign\fr \fB1membar_consumer\fr
407 \fB1membar_enter\fr \fB1membar_exit\fr
408 \fB1membar_producer\fr \fB1memccpy\fr
409 \fB1memchr\fr \fB1memcmp\fr
410 \fB1memcntl\fr \fB1memcpy\fr
411 \fB1meminfo\fr \fB1memmove\fr
412 \fB1memset\fr \fB1mincore\fr
413 \fB1mkdir\fr \fB1mkfifo\fr
414 \fB1mknod\fr \fB1mkstemp\fr
415 \fB1mktemp\fr \fB1mktime\fr
416 \fB1mlock\fr \fB1mlockall\fr
417 \fB1mmap\fr \fB1mmapobj\fr
418 \fB1modctl\fr
419 \fB1modf\fr \fB1modff\fr
420 \fB1modutx\fr \fB1monitor\fr
421 \fB1mount\fr \fB1mprotect\fr
422 \fB1mq_close\fr \fB1mq_getattr\fr
423 \fB1mq_notify\fr \fB1mq_open\fr
424 \fB1mq_receive\fr \fB1mq_reltimedreceive_np\fr
425 \fB1mq_reltimedsend_np\fr \fB1mq_send\fr
426 \fB1mq_setattr\fr \fB1mq_timedreceive\fr

```

```

427 \fBmq_timedsend\fr \fBmq_unlink\fr
428 \fBmrand48\fr \fBmsgctl\fr
429 \fBmsgget\fr \fBmsgids\fr
430 \fBmsgrcv\fr \fBmsgsnap\fr
431 \fBmsgsnd\fr \fBmsync\fr
432 \fBmunlock\fr \fBmunlockall\fr
433 \fBmunmap\fr \fBmutex_consistent\fr
434 \fBmutex_destroy\fr \fBmutex_init\fr
435 \fBmutex_lock\fr \fBmutex_trylock\fr
436 \fBmutex_unlock\fr \fBnanosleep\fr
437 \fBnxtafter\fr \fBnewlocale\fr
403 \fBnxtafter\fr
438 \fBnfs_getfh\fr \fBnftw\fr
439 \fBngettext\fr \fBnice\fr
440 \fBnl_langinfo\fr \fBnl_langinfo_l\fr
441 \fBnrand48\fr
406 \fBnl_langinfo\fr \fBnrand48\fr
442 \fBnss_default_finders\fr \fBnss_delete\fr
443 \fBnss_endent\fr \fBnss_getent\fr
444 \fBnss_search\fr \fBnss_setent\fr
445 \fBntp_adjtime\fr \fBntp_gettime\fr
446 \fBopen\fr \fBopenat\fr
447 \fBopendir\fr \fBopenlog\fr
448 \fBoptarg\fr \fBopterr\fr
449 \fBoptind\fr \fBoptopt\fr
450 \fBp_online\fr \fBpathconf\fr
451 \fBpause\fr \fBpclose\fr
452 \fBpccsample\fr \fBperror\fr
453 \fBpfmt\fr \fBpipe\fr
454 \fBplock\fr \fBpoll\fr
455 \fBpopen\fr \fBpport_alert\fr
456 \fBpport_associate\fr \fBpport_create\fr
457 \fBpport_dissociate\fr \fBpport_get\fr
458 \fBpport_getn\fr \fBpport_send\fr
459 \fBpport_sendn\fr \fBpposix_fadvise\fr
460 \fBpposix_fallocate\fr \fBpposix_madvise\fr
461 \fBpposix_memalign\fr \fBpposix_openpt\fr
462 \fBpposix_spawn\fr \fBpposix_spawn_file_actions_addclose\fr
463 \fBpposix_spawn_file_actions_addclosefrom_np\fr \fBpposix_spawn_file_actions_add\fr
464 \fBpposix_spawn_file_actions_addopen\fr
465 \fBpposix_spawn_file_actions_destroy\fr
466 \fBpposix_spawn_file_actions_init\fr
467 \fBpposix_spawn_pipe_np\fr
468 \fBpposix_spawnattr_destroy\fr
469 \fBpposix_spawnattr_getflags\fr \fBpposix_spawnattr_getpgroup\fr
470 \fBpposix_spawnattr_getschedparam\fr \fBpposix_spawnattr_getschedpolicy\fr
471 \fBpposix_spawnattr_getsigdefault\fr \fBpposix_spawnattr_getsigignore_np\fr
472 \fBpposix_spawnattr_getsigmask\fr \fBpposix_spawnattr_init\fr
473 \fBpposix_spawnattr_setflags\fr \fBpposix_spawnattr_setpgroup\fr
474 \fBpposix_spawnattr_setschedparam\fr \fBpposix_spawnattr_setschedpolicy\fr
475 \fBpposix_spawnattr_setsigdefault\fr \fBpposix_spawnattr_setsigignore_np\fr
476 \fBpposix_spawnattr_setsigmask\fr \fBpposix_spawnnp\fr
477 \fBpread\fr \fBprintf\fr
478 \fBprintstack\fr \fBpriocntl\fr
479 \fBpriocntlset\fr \fBpriv_addset\fr
480 \fBpriv_allocset\fr \fBpriv_copyset\fr
481 \fBpriv_delset\fr \fBpriv_emptyset\fr
482 \fBpriv_fillset\fr \fBpriv_freeset\fr
483 \fBpriv_getbyname\fr \fBpriv_getbynum\fr
484 \fBpriv_getsetbyname\fr \fBpriv_getsetbynum\fr
485 \fBpriv_gettext\fr \fBpriv_ineffect\fr
486 \fBpriv_intersect\fr \fBpriv_inverse\fr
487 \fBpriv_isemptyset\fr \fBpriv_isequalset\fr
488 \fBpriv_isfullset\fr \fBpriv_ismember\fr
489 \fBpriv_issubset\fr \fBpriv_set\fr
490 \fBpriv_set_to_str\fr \fBpriv_str_to_set\fr

```

```

491 \fBpriv_union\fR      \fBprocessor_bind\fR
492 \fBprocessor_info\fR   \fBprofil\fR
493 \fBpselect\fR          \fBpset_assign\fR
494 \fBpset_bind\fR        \fBpset_create\fR
495 \fBpset_destroy\fR     \fBpset_getattr\fR
496 \fBpset_getloadavg\fR  \fBpset_info\fR
497 \fBpset_list\fR        \fBpset_setattr\fR
498 \fBpsiginfo\fR         \fBpsignal\fR
499 \fBpthead_atfork\fR    \fBpthead_attr_destroy\fR
500 \fBpthead_attr_getdetachstate\fR \fBpthead_attr_getguardsize\fR
501 \fBpthead_attr_getinheritsched\fR \fBpthead_attr_getschedparam\fR
502 \fBpthead_attr_getschedpolicy\fR \fBpthead_attr_getscope\fR
503 \fBpthead_attr_getstack\fR \fBpthead_attr_getstackaddr\fR
504 \fBpthead_attr_getstacksize\fR \fBpthead_attr_init\fR
505 \fBpthead_attr_setdetachstate\fR \fBpthead_attr_setguardsize\fR
506 \fBpthead_attr_setinheritsched\fR \fBpthead_attr_setschedparam\fR
507 \fBpthead_attr_setschedpolicy\fR \fBpthead_attr_setscope\fR
508 \fBpthead_attr_setstack\fR \fBpthead_attr_setstackaddr\fR
509 \fBpthead_attr_setstacksize\fR \fBpthead_barrier_destroy\fR
510 \fBpthead_barrier_init\fR \fBpthead_barrier_wait\fR
511 \fBpthead_barrierattr_destroy\fR \fBpthead_barrierattr_getpshared\fR
512 \fBpthead_barrierattr_init\fR \fBpthead_barrierattr_setpshared\fR
513 \fBpthead_cancel\fR    \fBpthead_cond_broadcast\fR
514 \fBpthead_cond_destroy\fR \fBpthead_cond_init\fR
515 \fBpthead_cond_timedwait_np\fR \fBpthead_cond_signal\fR
516 \fBpthead_cond_timedwait\fR \fBpthead_cond_wait\fR
517 \fBpthead_condattr_destroy\fR \fBpthead_condattr_getclock\fR
518 \fBpthead_condattr_getpshared\fR \fBpthead_condattr_init\fR
519 \fBpthead_condattr_setclock\fR \fBpthead_condattr_setpshared\fR
520 \fBpthead_create\fR     \fBpthead_detach\fR
521 \fBpthead_equal\fR      \fBpthead_exit\fR
522 \fBpthead_getconcurrency\fR \fBpthead_getschedparam\fR
523 \fBpthead_getspecific\fR \fBpthead_join\fR
524 \fBpthead_key_create\fR  \fBpthead_key_create_once_np\fR
525 \fBpthead_key_delete\fR
526 \fBpthead_kill\fR       \fBpthead_mutex_consistent\fR
527 \fBpthead_mutex_destroy\fR \fBpthead_mutex_getprioceiling\fR
528 \fBpthead_mutex_init\fR  \fBpthead_mutex_lock\fR
529 \fBpthead_mutex_reltimedlock_np\fR \fBpthead_mutex_setprioceiling\fR
530 \fBpthead_mutex_timedlock\fR \fBpthead_mutex_trylock\fR
531 \fBpthead_mutex_unlock\fR \fBpthead_mutexattr_destroy\fR
532 \fBpthead_mutexattr_getprioceiling\fR \fBpthead_mutexattr_getprotocol\fR
533 \fBpthead_mutexattr_getpshared\fR \fBpthead_mutexattr_getrobust\fR
534 \fBpthead_mutexattr_gettype\fR \fBpthead_mutexattr_init\fR
535 \fBpthead_mutexattr_setprioceiling\fR \fBpthead_mutexattr_setprotocol\fR
536 \fBpthead_mutexattr_setpshared\fR \fBpthead_mutexattr_setrobust\fR
537 \fBpthead_mutexattr_settype\fR \fBpthead_once\fR
538 \fBpthead_rwlock_destroy\fR \fBpthead_rwlock_init\fR
539 \fBpthead_rwlock_rdlock\fR \fBpthead_rwlock_reltimedrdlock_np\fR
540 \fBpthead_rwlock_reltimedwrllock_np\fR \fBpthead_rwlock_timedrdlock\fR
541 \fBpthead_rwlock_timedwrllock\fR \fBpthead_rwlock_tryrdlock\fR
542 \fBpthead_rwlock_trywrlock\fR \fBpthead_rwlock_unlock\fR
543 \fBpthead_rwlock_wrlock\fR \fBpthead_rwlockattr_destroy\fR
544 \fBpthead_rwlockattr_getpshared\fR \fBpthead_rwlockattr_init\fR
545 \fBpthead_rwlockattr_setpshared\fR \fBpthead_self\fR
546 \fBpthead_setcancelstate\fR \fBpthead_setcanceltype\fR
547 \fBpthead_setconcurrency\fR \fBpthead_setspecific\fR
548 \fBpthead_sigmask\fR     \fBpthead_setschedparam\fR
549 \fBpthead_setschedprio\fR \fBpthead_spin_destroy\fR
550 \fBpthead_spin_init\fR   \fBpthead_spin_lock\fR
551 \fBpthead_spin_trylock\fR \fBpthead_spin_unlock\fR
552 \fBpthead_testcancel\fR  \fBpthead_testcancel\fR
553 \fBputacct\fR           \fBputc\fR
554 \fBputc_unlocked\fR     \fBputcchar\fR
555 \fBputcchar_unlocked\fR \fBputcenv\fR
556 \fBputmsg\fR            \fBputpmsg\fR

```

```

557 \fBputpwent\fR        \fBputs\fR
558 \fBputspent\fR         \fBpututline\fR
559 \fBpututxline\fR      \fBputw\fR
560 \fBputwc\fR            \fBputwchar\fR
561 \fBputws\fR            \fBpwrite\fR
562 \fBqecconvert\fR      \fBqecvt\fR
563 \fBqfconvert\fR       \fBqfcvt\fR
564 \fBqgconvert\fR       \fBqgcvt\fR
565 \fBqsort\fR            \fBquadruple_to_decimal\fR
566 \fBraise\fR            \fBrand\fR
567 \fBrand_r\fR           \fBrandom\fR
568 \fBrcctl_walk\fR       \fBrcctlblk_get_enforced_value\fR
569 \fBrcctlblk_get_firing_time\fR \fBrcctlblk_get_global_action\fR
570 \fBrcctlblk_get_global_flags\fR \fBrcctlblk_get_local_action\fR
571 \fBrcctlblk_get_local_flags\fR \fBrcctlblk_get_privilege\fR
572 \fBrcctlblk_get_recipient_pid\fR \fBrcctlblk_get_value\fR
573 \fBrcctlblk_set_local_action\fR \fBrcctlblk_set_local_flags\fR
574 \fBrcctlblk_set_privilege\fR \fBrcctlblk_set_recipient_pid\fR
575 \fBrcctlblk_set_value\fR \fBrcctlblk_size\fR
576 \fBre_comp\fR         \fBre_exec\fR
577 \fBread\fR             \fBreaddir\fR
578 \fBreaddir_r\fR       \fBreadlink\fR
579 \fBreadv\fR            \fBrealloc\fR
580 \fBrealpath\fR         \fBreboot\fR
581 \fBregcmp\fR           \fBregcomp\fR
582 \fBregerror\fR        \fBregex\fR
583 \fBregexec\fR          \fBregfree\fR
584 \fBremove\fR           \fBremove\fR
585 \fBrename\fR           \fBrenameat\fR
586 \fBresetmnttab\fR     \fBresolvepath\fR
587 \fBrewind\fR           \fBrewinddir\fR
588 \fBrindex\fR           \fBrmkdir\fR
589 \fBbrw_rdlock\fR       \fBbrw_read_held\fR
590 \fBbrw_tryrdlock\fR   \fBbrw_trywrllock\fR
591 \fBbrw_unlock\fR       \fBbrw_write_held\fR
592 \fBbrw_wrllock\fR     \fBbrwlock_destroy\fR
593 \fBbrwlock_init\fR    \fBbsbrk\fR
594 \fBscalb\fR            \fBscandir\fR
595 \fBscanf\fR            \fBsched_get_priority_max\fR
596 \fBsched_get_priority_min\fR \fBsched_getparam\fR
597 \fBsched_getscheduler\fR \fBsched_rr_get_interval\fR
598 \fBsched_setparam\fR   \fBsched_setscheduler\fR
599 \fBsched_yield\fR      \fBschedctl_exit\fR
600 \fBschedctl_init\fR    \fBschedctl_lookup\fR
601 \fBschedctl_start\fR   \fBschedctl_stop\fR
602 \fBseconvert\fR       \fBseed48\fR
603 \fBseekdir\fR          \fBselect\fR
604 \fBsem_close\fR        \fBsem_destroy\fR
605 \fBsem_getvalue\fR     \fBsem_init\fR
606 \fBsem_open\fR         \fBsem_post\fR
607 \fBsem_reltimedwait_np\fR \fBsem_timedwait\fR
608 \fBsem_trywait\fR     \fBsem_unlink\fR
609 \fBsem_wait\fR
610 \fBsema_destroy\fR     \fBsema_held\fR
611 \fBsema_init\fR        \fBsema_post\fR
612 \fBsema_trywait\fR    \fBsema_wait\fR
613 \fBsemctl\fR           \fBsemget\fR
614 \fBsemids\fR           \fBsemop\fR
615 \fBsemtimedop\fR      \fBsetattrat\fR
616 \fBsetbuf\fR
617 \fBsetbuffer\fR        \fBsetcat\fR
618 \fBsetcontext\fR       \fBsetegid\fR
619 \fBsetenv\fR           \fBseteuid\fR
620 \fBsetgid\fR           \fBsetgrent\fR
621 \fBsetgroups\fR        \fBsethostname\fR
622 \fBsetitimer\fR        \fBsetjmp\fR

```

```

623 \fBsetkey\fR      \fBsetlabel\fR
624 \fBsetlinebuf\fR   \fBsetlocale\fR
625 \fBsetlogmask\fR   \fBsetnetgrent\fR
626 \fBsetpflags\fR    \fBsetpgid\fR
627 \fBsetpgrp\fR      \fBsetppriv\fR
628 \fBsetpriority\fR   \fBsetpwent\fR
629 \fBsetrotl\fR       \fBsetregid\fR
630 \fBsetreuid\fR      \fBsetrlimit\fR
631 \fBsetsid\fR        \fBsetspent\fR
632 \fBsetstate\fR      \fBsettaskid\fR
633 \fBsettimeofday\fR \fBsetuid\fR
634 \fBsetusershell\fR \fBsetustack\fR
635 \fBsetutent\fR      \fBsetutxent\fR
636 \fBsetvbuf\fR       \fBsfconvert\fR
637 \fBsgconvert\fR     \fBshm_open\fR
638 \fBshm_unlink\fR    \fBshmattr\fR
639 \fBshmctl\fR         \fBshmdt\fR
640 \fBshmget\fR         \fBshmids\fR
641 \fBsig2str\fR        \fBsigaction\fR
642 \fBsigaddset\fR      \fBsigaltstack\fR
643 \fBsigdelset\fR     \fBsigemptyset\fR
644 \fBsigfillset\fR     \fBsigpev\fR
645 \fBsighold\fR        \fBsigignore\fR
646 \fBsiginterrupt\fR   \fBsigismember\fR
647 \fBsiglongjmp\fR    \fBsignal\fR
648 \fBsigpause\fR       \fBsigpending\fR
649 \fBsigprocmask\fR    \fBsigqueuev\fR
650 \fBsigrelse\fR
651 \fBsigsend\fR        \fBsigsendset\fR
652 \fBsigset\fR         \fBsigsetjmp\fR
653 \fBsigstack\fR       \fBsigsuspend\fR
654 \fBsigtimedwait\fR   \fBsigwait\fR
655 \fBsigwaitinfo\fR    \fBsingle_to_decimal\fR
656 \fBsleep\fR          \fBsnprintf\fR
657 \fBsprintf\fR        \fBsrnd\fR
658 \fBsrnd48\fR         \fBsrnd48_r\fR
659 \fBsscanf\fR          \fBssignal\fR
660 \fBstack_getbounds\fR \fBstack_inbounds\fR
661 \fBstack_setbounds\fR \fBstack_violation\fR
662 \fBstat\fR           \fBstatfs\fR
663 \fBstatvfs\fR        \fBstime\fR
664 \fBstr2sig\fR
665 \fBstrcasecmp\fR     \fBstrcasecmp_l\fR
666 \fBstrcat\fR         \fBstrchr\fR
667 \fBstrcat_r\fR       \fBstrchr_r\fR
668 \fBstrcoll\fR        \fBstrcoll_r\fR
669 \fBstrcpy\fR         \fBstrcpy_r\fR
670 \fBstrdup\fR         \fBstrdup_r\fR
671 \fBstrerror_r\fR
672 \fBstrerror_r_r\fR  \fBstrfmon_l\fR
673 \fBstrftime\fR      \fBstrptime_l\fR
674 \fBstring_to_decimal\fR
675 \fBstring_to_decimal_r\fR \fBstrfmon\fR
676 \fBstrncat\fR        \fBstrncat_r\fR
677 \fBstrncat_r\fR      \fBstrncmp\fR
678 \fBstrncpy\fR        \fBstrncpy_r\fR
679 \fBstrptime\fR       \fBstrptime_l\fR
680 \fBstrrchr\fR        \fBstrsep\fR
681 \fBstrsignal\fR      \fBstrsignal_r\fR
682 \fBstrsignal_r\fR   \fBstrtod\fR

```

```

683 \fBstrtof\fR        \fBstrtoimax\fR
684 \fBstrtok\fR          \fBstrtok_r\fR
685 \fBstrtol\fR          \fBstrtol_r\fR
686 \fBstrtoll\fR         \fBstrtoll_r\fR
687 \fBstrtoul\fR         \fBstrtoul_r\fR
688 \fBstrxfrm\fR         \fBstrxfrm_r\fR
689 \fBswab\fR            \fBswapcontext\fR
690 \fBswaptbl\fR         \fBswprintf\fR
691 \fBswscanf\fR         \fBsymlink\fR
692 \fBsync\fR           \fBsync_instruction_memory\fR
693 \fBsysconf\fR         \fBsysfs\fR
694 \fBsysinfo\fR         \fBsyslog\fR
695 \fBsystem\fR          \fBtcdrain\fR
696 \fBtcfld\fR           \fBtcfld_r\fR
697 \fBtcfldset\fR        \fBtcfldset_r\fR
698 \fBtcfldset_r\fR     \fBtcfldset_r_r\fR
699 \fBtcfldset_r_r\fR   \fBtcfldset_r_r_r\fR
700 \fBtcfldset_r_r_r\fR \fBtcfldset_r_r_r_r\fR
701 \fBtelldir\fR         \fBtempnam\fR
702 \fBtextdomain\fR      \fBtfind\fR
703 \fBthr_continue\fR    \fBthr_create\fR
704 \fBthr_exit\fR        \fBthr_getconcurrency\fR
705 \fBthr_getprio\fR     \fBthr_getspecific\fR
706 \fBthr_join\fR        \fBthr_keycreate\fR
707 \fBthr_keycreate_once\fR
708 \fBthr_kill\fR        \fBthr_main\fR
709 \fBthr_min_stack\fR   \fBthr_self\fR
710 \fBthr_setconcurrency\fR \fBthr_setprio\fR
711 \fBthr_setspecific\fR \fBthr_sigsetmask\fR
712 \fBthr_stksegment\fR \fBthr_suspend\fR
713 \fBthr_yield\fR       \fBtime\fR
714 \fBtimer_create\fR    \fBtimer_delete\fR
715 \fBtimer_getoverrun\fR \fBtimer_gettime\fR
716 \fBtimer_settime\fR
717 \fBtimes\fR           \fBtimezone\fR
718 \fBtmpfile\fR         \fBtmpnam\fR
719 \fBtmpnam_r\fR        \fBtoascii\fR
720 \fBtolower\fR         \fBtolower_l\fR
721 \fBtoupper\fR         \fBtoupper_l\fR
722 \fBtowctrans\fR       \fBtowctrans_l\fR
723 \fBtowlower\fR        \fBtowlower_l\fR
724 \fBtowupper\fR        \fBtowupper_l\fR
725 \fBtruncate\fR
726 \fBtsearch\fR         \fBttyname\fR
727 \fBttyname_r\fR       \fBttyname_r_r\fR
728 \fBtwalk\fR           \fBtzname\fR
729 \fBtzset\fR           \fBu8_strcmp\fR
730 \fBu8_textprep_str\fR \fBu8_validate\fR
731 \fBuadmin\fR
732 \fBualarm\fR         \fBuconv_u16tou32\fR
733 \fBuconv_u16tou8\fR   \fBuconv_u32tou16\fR
734 \fBuconv_u32tou8\fR   \fBuconv_u8tou16\fR
735 \fBuconv_u8tou32\fR   \fBucred_free\fR
736 \fBucred_get\fR       \fBucred_getegid\fR
737 \fBucred_geteuid\fR   \fBucred_getgroups\fR
738 \fBucred_getpflags\fR \fBucred_getpid\fR
739 \fBucred_getprivset\fR \fBucred_getprojid\fR
740 \fBucred_getrgid\fR   \fBucred_getruid\fR
741 \fBucred_getsgid\fR   \fBucred_getsuid\fR
742 \fBucred_getzoneid\fR \fBucred_size\fR
743 \fBulckpwdf\fR        \fBulimit\fR
744 \fBulltostr\fR        \fBumask\fR
745 \fBumount\fR          \fBumount2\fR

```



```

746 \fBuname\fR      \fBungetc\fR
747 \fBungetwc\fR     \fBunlink\fR
748 \fBunlinkat\fR    \fBunlockpt\fR
749 \fBunordered\fR   \fBunsetenv\fR
750 \fBupdwtmp\fR     \fBupdwtmpx\fR
751 \fBuselocale\fR
752 \fBusleep\fR      \fBustat\fR
753 \fButime\fR       \fButimensat\fR
754 \fButimes\fR     \fButmpname\fR
755 \fButmpxname\fR   \fBuucopy\fR
756 \fBvalloc\fR     \fBvasprintf\fR
757 \fBverror\fR     \fBverrx\fR
758 \fBvfork\fR      \fBvforkx\fR
759 \fBvfprintf\fR   \fBvfscanf\fR
760 \fBvfwprintf\fR  \fBvfwscanf\fR
761 \fBvhangup\fR   \fBvlfmt\fR
762 \fBvprintf\fR   \fBvprintf\fR
763 \fBvscanf\fR    \fBvsnprintf\fR
764 \fBvsprintf\fR  \fBvsscanf\fR
765 \fBvswprintf\fR \fBvswscanf\fR
766 \fBvsyslog\fR   \fBvwarn\fR
767 \fBvwarnx\fR    \fBvwprintf\fR
768 \fBvwscanf\fR  \fBwait\fR
769 \fBwait3\fR     \fBwait4\fR
770 \fBwaitid\fR    \fBwaitpid\fR
771 \fBwalkcontext\fR \fBwarn\fR
772 \fBwarnx\fR     \fBwatoll\fR
773 \fBwcr tomb\fR   \fBwcr tomb_1\fR
774 \fBwcscat\fR
775 \fBwchr tomb\fR \fBwcscat\fR
776 \fBwchr chr\fR  \fBwcscmp\fR
777 \fBwccoll\fR
778 \fBwccoll_1\fR \fBwccoll_1\fR
779 \fBwccopy\fR
780 \fBwccoll\fR    \fBwccopy\fR
781 \fBwccspn\fR   \fBwccsftime\fR
782 \fBwccslen\fR  \fBwccsncat\fR
783 \fBwccscmp\fR \fBwccsncpy\fR
784 \fBwccsbrk\fR  \fBwccsrchr\fR
785 \fBwccrtombs\fR \fBwccrtombs_1\fR
786 \fBwccsspfn\fR
787 \fBwccrtombs\fR \fBwccsspfn\fR
788 \fBwccsstr\fR  \fBwccstod\fR
789 \fBwccstof\fR  \fBwccstoimax\fR
790 \fBwccstok\fR  \fBwccstol\fR
791 \fBwccstold\fR \fBwccstoll\fR
792 \fBwccstombs\fR \fBwccstombs_1\fR
793 \fBwccstoul\fR
794 \fBwccstombs\fR \fBwccstoul\fR
795 \fBwccstoull\fR \fBwccstoumax\fR
796 \fBwccswcs\fR
797 \fBwccswidth\fR \fBwccswidth_1\fR
798 \fBwccsxfm\fR  \fBwccsxfm_1\fR
799 \fBwccctob\fR  \fBwccctob_1\fR
800 \fBwccctomb\fR \fBwccctomb_1\fR
801 \fBwccctrans\fR \fBwccctrans_1\fR
802 \fBwccctype\fR \fBwccctype_1\fR
803 \fBwccwidth\fR \fBwccwidth_1\fR
804 \fBwccwics\fR  \fBwccwidth\fR
805 \fBwccsxfm\fR \fBwccctob\fR
806 \fBwccctomb\fR \fBwccctrans\fR
807 \fBwccctype\fR \fBwccwidth\fR
808 \fBwccmemchr\fR \fBwccmemcmp\fR
809 \fBwccmemcpy\fR \fBwccmemmove\fR
810 \fBwccmemset\fR \fBwccwordexp\fR
811 \fBwccwordfree\fR \fBwccprintf\fR
812 \fBwccwract\fR \fBwccwrite\fR

```

```

804 \fBwritev\fR     \fBwscanf\fR
805 \fBwscasecmp\fR  \fBwscat\fR
806 \fBwscchr\fR     \fBwscmp\fR
807 \fBwscoll\fR    \fBwscoll\fR
808 \fBwscpy\fR      \fBwscspn\fR
809 \fBwsdup\fR      \fBwslen\fR
810 \fBwscasecmp\fR  \fBwsncat\fR
811 \fBwsncmp\fR     \fBwsncpy\fR
812 \fBwspbrk\fR    \fBwsprintf\fR
813 \fBwsrchr\fR    \fBwsscanf\fR
814 \fBwsspfn\fR    \fBwstod\fR
815 \fBwstok\fR     \fBwstol\fR
816 \fBwstoll\fR   \fBwstostr\fR
817 \fBwsxfrm\fR   \fByield\fR
818 .TE

820 .sp
821 .LP
822 The following interfaces are unique to the 32-bit version of this library:
823 .sp

825 .sp
826 .TS
827 l l
828 l l .
829 \fB__div64\fR     \fB__mul64\fR
830 \fB__posix_readdir_r\fR \fB__rem64\fR
831 \fB__udiv64\fR    \fB__urem64\fR
832 \fB__bufendtab\fR \fB__lastbuf\fR
833 \fB__s_fcntl\fR  \fB__sys_nsig\fR
834 \fB__xftw64\fR   \fB__baio_cancel64\fR
835 \fB__baio_error64\fR \fB__baio_fsync64\fR
836 \fB__baio_read64\fR \fB__baio_return64\fR
837 \fB__baio_suspend64\fR \fB__baio_waitn64\fR
838 \fB__baio_write64\fR \fB__creat64\fR
839 \fB__bfgetpos64\fR \fB__bfopen64\fR
840 \fB__bfreopen64\fR \fB__bfseeko64\fR
841 \fB__bfsetpos64\fR \fB__bfstat64\fR
842 \fB__bfstatvfs64\fR \fB__bftello64\fR
843 \fB__bftruncate64\fR \fB__bftw64\fR
844 \fB__bgetdents64\fR \fB__bgetrlimit64\fR
845 \fB__blio_listio64\fR \fB__blockf64\fR
846 \fB__blseek64\fR   \fB__blstat64\fR
847 \fB__bmkstemp64\fR \fB__bmmmap64\fR
848 \fB__bnftw64\fR   \fB__bopen64\fR
849 \fB__bpread64\fR  \fB__bptrace\fR
850 \fB__bpwrite64\fR \fB__breaddir64\fR
851 \fB__breaddir64_r\fR \fB__bs_fcntl\fR
852 \fB__bs_ioctl\fR   \fB__bselect_large_fdset\fR
853 \fB__bsetrlimit64\fR \fB__bstat64\fR
854 \fB__bstatvfs64\fR \fB__bsys_errlist\fR
855 \fB__bsys_nerr\fR \fB__btell64\fR
856 \fB__btmpfile64\fR \fB__btruncate64\fR
857 .TE

859 .sp
860 .LP
861 The following interfaces are unique to the 32-bit SPARC version of this
862 library:
863 .sp

865 .sp
866 .TS
867 l l
868 l l .
869 \fB__div\fR       \fB__mul\fR

```

```

870 \fB\&.rem\fR      \fB\&.stret1\fR
871 \fB\&.stret2\fR   \fB\&.stret4\fR
872 \fB\&.stret8\fR   \fB\&.udiv\fR
873 \fB\&.umul\fR     \fB\&.urem\fR
874 \fB_Q_add\fR       \fB_Q_cmp\fR
875 \fB_Q_cmpe\fR      \fB_Q_div\fR
876 \fB_Q_dtoq\fR     \fB_Q_feq\fR
877 \fB_Q_fge\fR       \fB_Q_fgt\fR
878 \fB_Q_fle\fR       \fB_Qflt\fR
879 \fB_Q_fne\fR       \fB_Q_itoq\fR
880 \fB_Q_lltoq\fR    \fB_Q_mul\fR
881 \fB_Q_neg\fR       \fB_Q_qtod\fR
882 \fB_Q_qtoi\fR     \fB_Q_qtoll\fR
883 \fB_Q_qtos\fR     \fB_Q_qtou\fR
884 \fB_Q_qtoull\fR   \fB_Q_sqrt\fR
885 \fB_Q_stoq\fR     \fB_Q_sub\fR
886 \fB_Q_ulltoq\fR   \fB_Q_utoq\fR
887 \fB_dtoll\fR      \fB_dtou\fR
888 \fB_dtoull\fR     \fB_ftoll\fR
889 \fB_ftou\fR       \fB_ftoull\fR
890 \fB_umul64\fR
891 .TE

```

```

893 .sp
894 .LP
895 The following interfaces are unique to the 32-bit x86 version of this library:
896 .sp

```

```

898 .sp
899 .TS
900 l l
901 l l .
902 \fB__fpstart\fR \fB__fp_hw\fR
903 \fB__fpstart\fR \fB__fxstat\fR
904 \fB__lxstat\fR  \fB__nuname\fR
905 \fB__thr_errno_addr\fR \fB__xmknod\fR
906 \fB__xstat\fR   \fB__nuname\fR
907 .TE

```

```

909 .sp
910 .LP
911 The following interfaces are unique to the 64-bit SPARC version of this
912 library:
913 .sp

```

```

915 .sp
916 .TS
917 l l
918 l l .
919 \fB_Qp_add\fR      \fB_Qp_cmp\fR
920 \fB_Qp_cmpe\fR     \fB_Qp_div\fR
921 \fB_Qp_dtoq\fR     \fB_Qp_feq\fR
922 \fB_Qp_fge\fR      \fB_Qp_fgt\fR
923 \fB_Qp_fle\fR      \fB_Qpflt\fR
924 \fB_Qp_fne\fR      \fB_Qp_itoq\fR
925 \fB_Qp_mul\fR      \fB_Qp_neg\fR
926 \fB_Qp_qtod\fR     \fB_Qp_qtoi\fR
927 \fB_Qp_qtos\fR     \fB_Qp_qtoui\fR
928 \fB_Qp_qtoux\fR    \fB_Qp_qtox\fR
929 \fB_Qp_sqrt\fR     \fB_Qp_stoq\fR
930 \fB_Qp_sub\fR      \fB_Qp_uitoq\fR
931 \fB_Qp_uptoq\fR    \fB_Qp_xtoq\fR
932 \fB__align_cpy_1\fR \fB__align_cpy_16\fR
933 \fB__align_cpy_2\fR \fB__align_cpy_4\fR
934 \fB__align_cpy_8\fR \fB__dtoul\fR
935 \fB__ftoul\fR      \fB__sparc_utrap_install\fR

```

```

936 .TE

938 .SH FILES
939 .sp
940 .ne 2
941 .na
942 \fB\fb/lib/libc.so.1\fR
943 .ad
944 .RS 27n
945 shared object
946 .RE

948 .sp
949 .ne 2
950 .na
951 \fB\fb/lib/64/libc.so.1\fR
952 .ad
953 .RS 27n
954 64-bit shared object
955 .RE

957 .sp
958 .ne 2
959 .na
960 \fB\fb/lib/c_synonyms.so.1\fR
961 .ad
962 .RS 27n
963 A compatibility library to provide access to obsolete \fBlibc\fR synonym
964 symbols
965 .RE

967 .sp
968 .ne 2
969 .na
970 \fB\fb/lib/64/c_synonyms.so.1\fR
971 .ad
972 .RS 27n
973 A 64-bit compatibility library to provide access to obsolete \fBlibc\fR synonym
974 symbols
975 .RE

977 .SH ATTRIBUTES
978 .sp
979 .LP
980 See \fBattributes\fR(5) for descriptions of the following attributes:
981 .sp

983 .sp
984 .TS
985 box;
986 c | c
987 l | l .
988 ATTRIBUTE TYPE    ATTRIBUTE VALUE
989 MT-Level          Safe
990 .TE

992 .SH SEE ALSO
993 .sp
994 .LP
995 \fBfbpvs\fR(1), \fBfbIntro\fR(2), \fBfbIntro\fR(3), \fBattributes\fR(5),
996 \fBfb1f64\fR(5), \fBfbstandards\fR(5)
997 .SH NOTES
998 .sp
999 .LP
1000 The synonyms compatibility library, \fBfb_c_synonyms.so.1\fR, provides a mechanism
1001 to support old applications and libraries that were mistakenly built using

```

1002 now-obsolete synonym symbols from \fBlibc\fR.  
1003 .sp  
1004 .LP  
1005 Before the advent of direct binding (-B direct) \fBlibc\fR provided many  
1006 functions with two names. For example, \fBgetpwent()\fR and \fB\_getpwent()\fR.  
1007 These two names referred to exactly the same function in \fBlibc\fR. The  
1008 leading-underscore symbol was intended to be used by system libraries in order  
1009 to avoid conflicting with an application that might define its own version of  
1010 \fBgetpwent()\fR with completely different semantics. Standard-conforming  
1011 applications may not define and use function names with leading underscores.  
1012 .sp  
1013 .LP  
1014 Solaris system libraries are now built with direct binding. This means that a  
1015 system library that calls \fBgetpwent()\fR will bind directly to the instance  
1016 of \fBgetpwent()\fR in \fBlibc\fR, even if the application to which it is  
1017 linked defines a different \fBgetpwent()\fR for its own use. The application  
1018 binds to its instance of \fBgetpwent()\fR and there is no resulting conflict.  
1019 The direct binding mechanism is equally available to libraries not delivered  
1020 with Solaris.  
1021 .sp  
1022 .LP  
1023 As a result of this evolution, most of the leading-underscore synonym symbols  
1024 have been removed from \fBlibc\fR. This means that applications that call  
1025 these now-obsolete function names will cease to work. They will typically draw  
1026 the error:  
1027 .sp  
1028 .in +2  
1029 .nf  
1030 \$ ./application  
1031 ld.so.1: fatal: relocation error: symbol \_getpwent:  
1032 referenced symbol not found  
1033 Killed  
1034 .fi  
1035 .in -2  
1036 .sp  
  
1038 .sp  
1039 .LP  
1040 All of the old leading-underscore symbols have been copied to the synonyms  
1041 compatibility library. This library simply redirects the calls to the  
1042 non-underscore instances of the corresponding functions in \fBlibc\fR. Use it  
1043 as a pre-loaded object:  
1044 .sp  
1045 .in +2  
1046 .nf  
1047 \$ LD\_PRELOAD=c\_synonyms.so.1 ./application  
1048 .fi  
1049 .in -2  
1050 .sp  
  
1052 .sp  
1053 .LP  
1054 The synonyms compatibility library is intended neither to enable the generation  
1055 of applications that call the obsolete leading-underscore synonym functions,  
1056 nor to endorse this particular programming practice.

```

*****
16233 Fri Jul 11 11:43:12 2014
new/usr/src/man/man5/envIRON.5
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1  \' te
2  .\ Copyright 1989 AT&T
3  .\ Copyright (c) 2002, Sun Microsystems, Inc. All Rights Reserved
4  .\ Copyright (c) 2014, Joyent, Inc. All Rights Reserved
5  .\ Copyright 2014 Garrett D'Amore <garrett@damore.org>
6  .\ The contents of this file are subject to the terms of the Common Development
7  .\ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
8  .\ When distributing Covered Code, include this CDDL HEADER in each file and in
9  .TH ENVIRON 5 "Jun 26, 2014"
7  .TH ENVIRON 5 "Nov 19, 2002"
10 .SH NAME
11 environ \- user environment
12 .SH DESCRIPTION
13 .sp
14 .LP
15 When a process begins execution, one of the \fBexec\fR family of functions
16 makes available an array of strings called the environment; see \fBexec\fR(2).
17 By convention, these strings have the form \fIvariable=value\fR, for example,
18 \fBPATH=/sbin:/usr/sbin\fR. These environmental variables provide a way to make
19 information about a program's environment available to programs.
20 .sp
21 .LP
22 A name may be placed in the environment by the \fBexport\fR command and
23 \fIname\fR=\fIvalue\fR arguments in \fBsh\fR(1), or by one of the \fBexec\fR
24 functions. It is unwise to conflict with certain shell variables such as
25 \fBMAIL\fR, \fBPS1\fR, \fBPS2\fR, and \fBIFS\fR that are frequently exported by
26 \fB&.&.profile\fR files; see \fBprofile\fR(4).
27 .sp
28 .LP
29 The following environmental variables can be used by applications and are
30 expected to be set in the target run-time environment.
31 .sp
32 .ne 2
33 .na
34 \fBFBHOME\fR
35 .ad
36 .sp .6
37 .RS 4n
38 The name of the user's login directory, set by \fBlogin\fR(1) from the password
39 file; see \fBpasswd\fR(4).
40 .RE
41 .sp
42 .ne 2
43 .na
44 \fBFLANG\fR
45 .ad
46 .sp .6
47 .RS 4n
48 The string used to specify internationalization information that allows users
49 to work with different national conventions. The \fBsetlocale\fR(3C) and
50 \fBnewlocale\fR(3C) functions
51 check the \fBFLANG\fR environment variable when they are called with \fB""\fR as
52 to work with different national conventions. The \fBsetlocale\fR(3C) function
53 checks the \fBFLANG\fR environment variable when it is called with \fB""\fR as
54 the \fBlocale\fR argument. \fBFLANG\fR is used as the default locale if the
55 corresponding environment variable for a particular category is unset or null.
56 If, however, \fBLC_ALL\fR is set to a valid, non-empty value, its contents are

```

```

53 used to override both the \fBLANG\fR and the other \fBLC_*\fR variables. For
54 example, when invoked as \fBsetlocale(LC_CTYPE, "")\fR, \fBsetlocale()\fR will
55 query the \fBLC_CTYPE\fR environment variable first to see if it is set and
56 non-null. If \fBLC_CTYPE\fR is not set or null, then \fBsetlocale()\fR will
57 check the \fBLANG\fR environment variable to see if it is set and non-null. If
58 both \fBLANG\fR and \fBLC_CTYPE\fR are unset or \fBNULL\fR, the default "C"
59 locale will be used to set the \fBLC_CTYPE\fR category.
60 .sp
61 Most commands will invoke \fBsetlocale(LC_ALL, "")\fR prior to any other
62 processing. This allows the command to be used with different national
63 conventions by setting the appropriate environment variables. In addition, some
64 commands will use
65 .BR uselocale (3C)
66 to set a thread-specific locale.
67 .sp
68 conventions by setting the appropriate environment variables.
69 .sp
70 The following environment variables correspond to each category of
71 \fBsetlocale\fR(3C):
72 .sp
73 .ne 2
74 .na
75 \fBLC_ALL\fR
76 .ad
77 .sp .6
78 .RS 4n
79 If set to a valid, non-empty string value, override the values of \fBLANG\fR
80 and all the other \fBLC_*\fR variables.
81 .RE
82 .sp
83 .ne 2
84 .na
85 \fBLC_COLLATE\fR
86 .ad
87 .sp .6
88 .RS 4n
89 This category specifies the character collation sequence being used. The
90 information corresponding to this category is stored in a database created by
91 the \fBlocaledef\fR(1) command. This environment variable affects
92 \fBstrcoll\fR(3C) and \fBstrxfrm\fR(3C).
93 .RE
94 .sp
95 .ne 2
96 .na
97 \fBLC_CTYPE\fR
98 .ad
99 .sp .6
100 .RS 4n
101 This category specifies character classification, character conversion, and
102 widths of multibyte characters. When \fBLC_CTYPE\fR is set to a valid value,
103 the calling utility can display and handle text and file names containing valid
104 characters for that locale; Extended Unix Code (EUC) characters where any
105 individual character can be 1, 2, or 3 bytes wide; and EUC characters of 1, 2,
106 or 3 column widths. The default "C" locale corresponds to the 7-bit \fBASCII\fR
107 character set; only characters from ISO 8859-1 are valid. The information
108 corresponding to this category is stored in a database created by the
109 \fBlocaledef()\fR command. This environment variable is used by
110 \fBctype\fR(3C), \fBmblen\fR(3C), and many commands, such as \fBcat\fR(1),
111 \fBbed\fR(1), \fBls\fR(1), and \fBvi\fR(1).
112 .RE
113 .sp
114 .ne 2
115 .na
116 \fBLC_MESSAGES\fR
117 .ad

```

```

118 .ad
119 .sp .6
120 .RS 4n
121 This category specifies the language of the message database being used. For
122 example, an application may have one message database with French messages, and
123 another database with German messages. Message databases are created by the
124 \fBmkmsgs\fR(1) command. This environment variable is used by \fBexstr\fR(1),
125 \fBgetttx\fR(1), \fBsrchttx\fR(1), \fBgetttx\fR(3C), and \fBgetttx\fR(3C).
126 .RE

128 .sp
129 .ne 2
130 .na
131 \fB\FBLC_MONETARY\fR\fR
132 .ad
133 .sp .6
134 .RS 4n
135 This category specifies the monetary symbols and delimiters used for a
136 particular locale. The information corresponding to this category is stored in
137 a database created by the \fBlocaledef\fR(1) command. This environment variable
138 is used by \fBlocaleconv\fR(3C).
139 .RE

141 .sp
142 .ne 2
143 .na
144 \fB\FBLC_NUMERIC\fR\fR
145 .ad
146 .sp .6
147 .RS 4n
148 This category specifies the decimal and thousands delimiters. The information
149 corresponding to this category is stored in a database created by the
150 \fBlocaledef()\fR command. The default \fBC\fR locale corresponds to \fB"."\fR
151 as the decimal delimiter and no thousands delimiter. This environment variable
152 is used by \fBlocaleconv\fR(3C), \fBprintf\fR(3C), and \fBstrtod\fR(3C).
153 .RE

155 .sp
156 .ne 2
157 .na
158 \fB\FBLC_TIME\fR\fR
159 .ad
160 .sp .6
161 .RS 4n
162 This category specifies date and time formats. The information corresponding to
163 this category is stored in a database specified in \fBlocaledef()\fR. The
164 default \fBC\fR locale corresponds to U.S. date and time formats. This
165 environment variable is used by many commands and functions; for example:
166 \fBbat\fR(1), \fBcalendar\fR(1), \fBdate\fR(1), \fBstrftime\fR(3C), and
167 \fBgetdate\fR(3C).
168 .RE

170 .RE

172 .sp
173 .ne 2
174 .na
175 \fB\FBMSGVERB\fR\fR
176 .ad
177 .sp .6
178 .RS 4n
179 Controls which standard format message components \fBfmtmsg\fR selects when
180 messages are displayed to \fBstderr\fR; see \fBfmtmsg\fR(1) and
181 \fBfmtmsg\fR(3C).
182 .RE

```

```

184 .sp
185 .ne 2
186 .na
187 \fB\FBNETPATH\fR\fR
188 .ad
189 .sp .6
190 .RS 4n
191 A colon-separated list of network identifiers. A network identifier is a
192 character string used by the Network Selection component of the system to
193 provide application-specific default network search paths. A network identifier
194 must consist of non-null characters and must have a length of at least 1. No
195 maximum length is specified. Network identifiers are normally chosen by the
196 system administrator. A network identifier is also the first field in any
197 \fB/etc/netconfig\fR file entry. \fBNETPATH\fR thus provides a link into the
198 \fB/etc/netconfig\fR file and the information about a network contained in that
199 network's entry. \fB/etc/netconfig\fR is maintained by the system
200 administrator. The library routines described in \fBgetnetpath\fR(3NSL) access
201 the \fBNETPATH\fR environment variable.
202 .RE

204 .sp
205 .ne 2
206 .na
207 \fB\FBNLSPATH\fR\fR
208 .ad
209 .sp .6
210 .RS 4n
211 Contains a sequence of templates which \fBcatopen\fR(3C) and \fBgetttx\fR(3C)
212 use when attempting to locate message catalogs. Each template consists of an
213 optional prefix, one or more substitution fields, a filename and an optional
214 suffix. For example:
215 .sp
216 .in +2
217 .nf
218 NLSPATH="/system/nlslib/%N.cat"
219 .fi
220 .in -2
221 .sp

223 defines that \fBcatopen()\fR should look for all message catalogs in the
224 directory \fB/system/nlslib\fR, where the catalog name should be constructed
225 from the \fIname\fR parameter passed to \fBcatopen\fR(\|), \fB%N\fR, with the
226 suffix \fB\&.cat\fR.
227 .sp
228 Substitution fields consist of a \fB%\fR symbol, followed by a single-letter
229 keyword. The following keywords are currently defined:
230 .sp
231 .ne 2
232 .na
233 \fB%N\fR
234 .ad
235 .sp .6
236 .RS 4n
237 The value of the \fIname\fR parameter passed to \fBcatopen()\fR.
238 .RE

240 .sp
241 .ne 2
242 .na
243 \fB%L\fR
244 .ad
245 .sp .6
246 .RS 4n
247 The value of \fBLANG\fR or \fBLC_MESSAGES\fR.
248 .RE

```

```

250 .sp
251 .ne 2
252 .na
253 \fB%l\fR
254 .ad
255 .sp .6
256 .RS 4n
257 The language element from \fBLANG\fR or \fBLC_MESSAGES\fR.
258 .RE

260 .sp
261 .ne 2
262 .na
263 \fB%t\fR
264 .ad
265 .sp .6
266 .RS 4n
267 The territory element from \fBLANG\fR or \fBLC_MESSAGES\fR.
268 .RE

270 .sp
271 .ne 2
272 .na
273 \fB%c\fR
274 .ad
275 .sp .6
276 .RS 4n
277 The codeset element from \fBLANG\fR or \fBLC_MESSAGES\fR.
278 .RE

280 .sp
281 .ne 2
282 .na
283 \fB%%\fR
284 .ad
285 .sp .6
286 .RS 4n
287 A single \fB%\fR character.
288 .RE

290 An empty string is substituted if the specified value is not currently defined.
291 The separators "\fB_\fR" and "\fB&.\fR" are not included in \fB%t\fR and
292 \fB%c\fR substitutions.
293 .sp
294 Templates defined in \fBNLSPATH\fR are separated by colons (\fB:\fR). A leading
295 colon or two adjacent colons (\fB::\fR) is equivalent to specifying \fB%\fR.
296 For example:
297 .sp
298 .in +2
299 .nf
300 NLSPATH="%N.cat:/nlslib/%L/%N.cat"
301 .fi
302 .in -2
303 .sp

305 indicates to \fBcatopen()\fR that it should look for the requested message
306 catalog in \fIname\fR, \fIname\fR\fB&.cat\fR and
307 \fB/nlslib/$LANG/\fR\fIname\fR.cat. For \fBgettext()\fR, \fB%\fR automatically
308 maps to "messages".
309 .sp
310 If \fBNLSPATH\fR is unset or \fBINULL\fR, \fBcatopen()\fR and \fBgettext()\fR
311 call \fBsetlocale\fR(3C), which checks \fBLANG\fR and the \fBLC_*\fR
312 variables to locate the message catalogs.
313 .sp
314 \fBNLSPATH\fR will normally be set up on a system wide basis (in
315 \fB/etc/profile\fR) and thus makes the location and naming conventions

```

```

316 associated with message catalogs transparent to both programs and users.
317 .RE

319 .sp
320 .ne 2
321 .na
322 \fB\FPATH\fR
323 .ad
324 .sp .6
325 .RS 4n
326 The sequence of directory prefixes that \fBsh\fR(1), \fBtime\fR(1),
327 \fBnice\fR(1), \fBnohup\fR(1), and other utilities apply in searching for a
328 file known by an incomplete path name. The prefixes are separated by colons
329 (\fB:\fR). \fBlogin\fR(1) sets \fBFPATH=/usr/bin\fR. For more detail, see
330 \fBsh\fR(1).
331 .RE

333 .sp
334 .ne 2
335 .na
336 \fB\FSEV_LEVEL\fR
337 .ad
338 .sp .6
339 .RS 4n
340 Define severity levels and associate and print strings with them in standard
341 format error messages; see \fBaddseverity\fR(3C), \fBfmtmsg\fR(1), and
342 \fBfmtmsg\fR(3C).
343 .RE

345 .sp
346 .ne 2
347 .na
348 \fB\FTERM\fR
349 .ad
350 .sp .6
351 .RS 4n
352 The kind of terminal for which output is to be prepared. This information is
353 used by commands, such as \fBvi\fR(1), which may exploit special capabilities
354 of that terminal.
355 .RE

357 .sp
358 .ne 2
359 .na
360 \fB\FBTZ\fR
361 .ad
362 .sp .6
363 .RS 4n
364 Timezone information. The contents of this environment variable are used by the
365 functions \fBctime\fR(3C), \fBlocaltime\fR(3C), \fBstrftime\fR(3C), and
366 \fBmktime\fR(3C) to override the default timezone. The value of \fBFTZ\fR has
367 one of the two formats (spaces inserted for clarity):
368 .sp
369 .in +2
370 .nf
371 :characters
372 .fi
373 .in -2

375 or
376 .sp
377 .in +2
378 .nf
379 std offset dst offset, rule
380 .fi
381 .in -2

```

```

383 If \fBTZ\fR is of the first format (that is, if the first character is a colon
384 (:)), or if \fBTZ\fR is not of the second format, then \fBTZ\fR designates a
385 path to a timezone database file relative to \fB/usr/share/lib/zoneinfo/\fR,
386 ignoring a leading colon if one exists.
387 .sp
388 Otherwise, \fBTZ\fR is of the second form, which when expanded is as follows:
389 .sp
390 .in +2
391 .nf
392 \fIstdoffset\fR[\fIdst\fR[\fIoffset\fR][,\fIstart\fR[\fItime\fR],\fIend\fR[\fI
393 .fi
394 .in -2

396 .sp
397 .ne 2
398 .na
399 \fB\fIstd\fR and \fIdst\fR
400 .ad
401 .sp .6
402 .RS 4n
403 Indicate no less than three, nor more than {\fBTZNAME_MAX\fR}, bytes that are
404 the designation for the standard (\fIstd\fR) or the alternative (\fIdst\fR,
405 such as Daylight Savings Time) timezone. Only \fIstd\fR is required; if
406 \fIdst\fR is missing, then the alternative time does not apply in this
407 timezone. Each of these fields can occur in either of two formats, quoted or
408 unquoted:
409 .RS +4
410 .TP
411 .ie t \(\bu
412 .el o
413 In the quoted form, the first character is the less-than ('<') character and
414 the last character is the greater-than ('>') character. All characters between
415 these quoting characters are alphanumeric characters from the portable
416 character set in the current locale, the plus-sign ('+') character, or the
417 minus-sign ('-') character. The \fIstd\fR and \fIdst\fR fields in this case do
418 not include the quoting characters.
419 .RE
420 .RS +4
421 .TP
422 .ie t \(\bu
423 .el o
424 In the unquoted form, all characters in these fields are alphabetic characters
425 from the portable character set in the current locale.
426 .RE
427 The interpretation of these fields is unspecified if either field is less than
428 three bytes (except for the case when \fIdst\fR is missing), more than
429 {\fBTZNAME_MAX\fR} bytes, or if they contain characters other than those
430 specified.
431 .RE

433 .sp
434 .ne 2
435 .na
436 \fB\fIoffset\fR
437 .ad
438 .sp .6
439 .RS 4n
440 Indicate the value one must add to the local time to arrive at Coordinated
441 Universal Time. The offset has the form:
442 .sp
443 .in +2
444 .nf
445 \fIhh\fR[:\fImm\fR[:\fIss\fR]]
446 .fi
447 .in -2

```

```

448 .sp
449 .RE

450 The minutes (\fImm\fR) and seconds (\fIss\fR) are optional. The hour (\fIhh\fR)
451 is required and can be a single digit. The \fIoffset\fR following \fIstd\fR is
452 required. If no \fIoffset\fR follows \fIdst\fR, daylight savings time is
453 assumed to be one hour ahead of standard time. One or more digits can be used.
454 The value is always interpreted as a decimal number. The hour must be between 0
455 and 24, and the minutes (and seconds), if present, must be between 0 and 59.
456 Out of range values can cause unpredictable behavior. If preceded by a "-", the
457 timezone is east of the Prime Meridian. Otherwise, it is west of the Prime
458 Meridian (which can be indicated by an optional preceding "\fI+\fR" sign).
459 .RE

461 .sp
462 .ne 2
463 .na
464 \fB\fIstart\fR[\fItime\fR,\fIend\fR[\fItime\fR]
465 .ad
466 .sp .6
467 .RS 4n
468 Indicate when to change to and back from daylight savings time, where
469 \fIstart/time\fR describes when the change from standard time to daylight
470 savings time occurs, and \fIend/time\fR describes when the change back occurs.
471 Each \fItime\fR field describes when, in current local time, the change is
472 made.
473 .sp
474 The formats of \fIstart\fR and \fIend\fR are one of the following:
475 .sp
476 .ne 2
477 .na
478 \fB\fBJ\fR[\fIn\fR]
479 .ad
480 .sp .6
481 .RS 4n
482 The Julian day \fIn\fR (1 \(\le \fIn\fR \(\le 365). Leap days are not counted.
483 That is, in all years, February 28 is day 59 and March 1 is day 60. It is
484 impossible to refer to the occasional February 29.
485 .RE

487 .sp
488 .ne 2
489 .na
490 \fB\fIn\fR
491 .ad
492 .sp .6
493 .RS 4n
494 The zero-based Julian day (0 \(\le \fIn\fR \(\le 365). Leap days are counted, and
495 it is possible to refer to February 29.
496 .RE

498 .sp
499 .ne 2
500 .na
501 \fB\fBM\fR[\fIm.n.d\fR]
502 .ad
503 .sp .6
504 .RS 4n
505 The \fId\fRth day, (0 \(\le \fId\fR \(\le 6) of week \fIn\fR of month \fIm\fR of
506 the year (1 \(\le \fIn\fR \(\le 5, 1 \(\le \fIm\fR \(\le 12), where week 5 means
507 "the last \fId\fR-day in month \fIm\fR" which may occur in either the fourth or
508 the fifth week). Week 1 is the first week in which the \fId\fRth day occurs.
509 Day zero is Sunday.
510 .RE

512 Implementation specific defaults are used for \fIstart\fR and \fIend\fR if
513 these optional fields are not specified.

```

```
514 .sp
515 The \ftime\fr has the same format as \foffset\fr except that no leading sign
516 ("- " or "+ " ) is allowed. If \ftime\fr is not specified, the default value is
517 02:00:00.
518 .RE

520 .RE

522 .SH SEE ALSO
523 .LP
524 \fbcat\fr(1), \fbdate\fr(1), \fbed\fr(1), \fbfmtmsg\fr(1), \fblocaledef\fr(1),
525 \fblogin\fr(1), \fbls\fr(1), \fbmkmsg\fr(1), \fbnice\fr(1), \fbnohup\fr(1),
526 \fbsh\fr(1), \fbstod\fr(1), \fbtime\fr(1), \fbvi\fr(1), \fbexec\fr(2),
527 \fbaddseverity\fr(3C), \fbcatopen\fr(3C), \fbctime\fr(3C), \fbctype\fr(3C),
528 \fbfmtmsg\fr(3C), \fbgetdate\fr(3C), \fbgetnetpath\fr(3NSL), \fbgettext\fr(3C),
529 \fbgettxt\fr(3C), \fblocaleconv\fr(3C), \fbmblen\fr(3C), \fbmkttime\fr(3C),
530 \fbnewlocale\fr(3C),
531 \fbprintf\fr(3C), \fbsetlocale\fr(3C), \fbstrcoll\fr(3C), \fbstrftime\fr(3C),
532 \fbstrtod\fr(3C), \fbstrxfrm\fr(3C), \fbuselocale\fr(3C), \fbTIMEZONE\fr(4),
533 \fbnetconfig\fr(4), \fbpasswd\fr(4), \fbprofile\fr(4)
529 \fbstrtod\fr(3C), \fbstrxfrm\fr(3C), \fbTIMEZONE\fr(4), \fbnetconfig\fr(4),
530 \fbpasswd\fr(4), \fbprofile\fr(4)
```



new/usr/src/pkg/manifests/system-header.mf

1

```
*****
90062 Fri Jul 11 11:43:13 2014
new/usr/src/pkg/manifests/system-header.mf
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright (c) 2012 by Delphix. All rights reserved.
25 # Copyright 2012 Nexenta Systems, Inc. All rights reserved.
26 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
27 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
28 #
29 set name=pkg.fmri value=pkg:/system/header@$(PKGVERS)
30 set name=pkg.description \
31   value="SunOS C/C++ header files for general development of software"
32 set name=pkg.summary value="SunOS Header Files"
33 set name=info.classification value=org.opensolaris.category.2008:System/Core
34 set name=variant.arch value=$(ARCH)
35 dir path=usr group=sys
36 dir path=usr/include
37 $(i386_ONLY)dir path=usr/include/$(ARCH64)
38 $(i386_ONLY)dir path=usr/include/$(ARCH64)/sys
39 dir path=usr/include/arpa
40 dir path=usr/include/asm
41 dir path=usr/include/ast
42 dir path=usr/include/bsm
43 dir path=usr/include/dat
44 dir path=usr/include/des
45 dir path=usr/include/gssapi
46 dir path=usr/include/hal
47 $(i386_ONLY)dir path=usr/include/ia32
48 $(i386_ONLY)dir path=usr/include/ia32/sys
49 dir path=usr/include/inet
50 dir path=usr/include/inet/kssl
51 dir path=usr/include/ipp
52 dir path=usr/include/ipp/ipgpc
53 dir path=usr/include/iso
54 dir path=usr/include/kerberosv5
55 dir path=usr/include/libpolkit
56 dir path=usr/include/net
57 dir path=usr/include/netinet
```

new/usr/src/pkg/manifests/system-header.mf

2

```
58 dir path=usr/include/nfs
59 dir path=usr/include/protocols
60 dir path=usr/include/rpc
61 dir path=usr/include/rpcsvc
62 dir path=usr/include/sasl
63 dir path=usr/include/scsi
64 dir path=usr/include/scsi/plugins
65 dir path=usr/include/scsi/plugins/ses
66 dir path=usr/include/scsi/plugins/ses/framework
67 dir path=usr/include/scsi/plugins/ses/vendor
68 dir path=usr/include/scsi/plugins/smp
69 dir path=usr/include/scsi/plugins/smp/engine
70 dir path=usr/include/scsi/plugins/smp/framework
71 dir path=usr/include/security
72 dir path=usr/include/sharefs
73 dir path=usr/include/sys
74 dir path=usr/include/sys/av
75 dir path=usr/include/sys/contract
76 dir path=usr/include/sys/crypto
77 dir path=usr/include/sys/dktp
78 dir path=usr/include/sys/fc4
79 dir path=usr/include/sys/fm
80 dir path=usr/include/sys/fm/cpu
81 dir path=usr/include/sys/fm/fs
82 dir path=usr/include/sys/fm/io
83 $(sparc_ONLY)dir path=usr/include/sys/fpu
84 dir path=usr/include/sys/fs
85 dir path=usr/include/sys/hotplug
86 dir path=usr/include/sys/hotplug/pci
87 dir path=usr/include/sys/ib
88 dir path=usr/include/sys/ib/adapters
89 dir path=usr/include/sys/ib/adapters/hermon
90 dir path=usr/include/sys/ib/adapters/tavor
91 dir path=usr/include/sys/ib/clients
92 dir path=usr/include/sys/ib/clients/ibd
93 dir path=usr/include/sys/ib/clients/of
94 dir path=usr/include/sys/ib/clients/of/rdma
95 dir path=usr/include/sys/ib/clients/of/sol_ofs
96 dir path=usr/include/sys/ib/clients/of/sol_ucma
97 dir path=usr/include/sys/ib/clients/of/sol_umad
98 dir path=usr/include/sys/ib/clients/of/sol_uverbs
99 dir path=usr/include/sys/ib/ibnex
100 dir path=usr/include/sys/ib/ibt1
101 dir path=usr/include/sys/ib/ibt1/impl
102 dir path=usr/include/sys/ib/mgt
103 dir path=usr/include/sys/ib/mgt/ibmf
104 dir path=usr/include/sys/iso
105 dir path=usr/include/sys/lvm
106 dir path=usr/include/sys/proc
107 dir path=usr/include/sys/rsm
108 $(i386_ONLY)dir path=usr/include/sys/sata group=sys
109 dir path=usr/include/sys/scsi
110 dir path=usr/include/sys/scsi/adapters
111 dir path=usr/include/sys/scsi/conf
112 dir path=usr/include/sys/scsi/generic
113 dir path=usr/include/sys/scsi/impl
114 dir path=usr/include/sys/scsi/targets
115 dir path=usr/include/sys/sysevent
116 dir path=usr/include/sys/tsol
117 dir path=usr/include/tsol
118 dir path=usr/include/uuid
119 $(sparc_ONLY)dir path=usr/include/v7
120 $(sparc_ONLY)dir path=usr/include/v7/sys
121 $(sparc_ONLY)dir path=usr/include/v9
122 $(sparc_ONLY)dir path=usr/include/v9/sys
123 dir path=usr/include/vm
```

```

124 dir path=usr/platform group=sys
125 $(sparc_ONLY)dir path=usr/platform/SUNW,A70 group=sys
126 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-CP2300 group=sys
127 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-CP2300/include
128 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-CP3010 group=sys
129 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-CP3010/include
130 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-T12 group=sys
131 $(sparc_ONLY)dir path=usr/platform/SUNW,Netra-T4 group=sys
132 $(sparc_ONLY)dir path=usr/platform/SUNW,SPARC-Enterprise group=sys
133 $(sparc_ONLY)dir path=usr/platform/SUNW,Serverbladel1 group=sys
134 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Blade-100 group=sys
135 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Blade-1000 group=sys
136 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Blade-1500 group=sys
137 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Blade-2500 group=sys
138 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire group=sys
139 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-15000 group=sys
140 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-280R group=sys
141 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-480R group=sys
142 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-880 group=sys
143 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V215 group=sys
144 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V240 group=sys
145 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V250 group=sys
146 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V440 group=sys
147 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V445 group=sys
148 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V490 group=sys
149 $(sparc_ONLY)dir path=usr/platform/SUNW,Sun-Fire-V890 group=sys
150 $(sparc_ONLY)dir path=usr/platform/SUNW,Ultra-2 group=sys
151 $(sparc_ONLY)dir path=usr/platform/SUNW,Ultra-250 group=sys
152 $(sparc_ONLY)dir path=usr/platform/SUNW,Ultra-4 group=sys
153 $(sparc_ONLY)dir path=usr/platform/SUNW,Ultra-Enterprise group=sys
154 $(sparc_ONLY)dir path=usr/platform/SUNW,Ultra-Enterprise-10000 group=sys
155 $(sparc_ONLY)dir path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-40 group=sys
156 $(sparc_ONLY)dir path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-60 group=sys
157 $(sparc_ONLY)dir path=usr/platform/SUNW,UltraSPARC-IIi-Netract group=sys
158 $(i386_ONLY)dir path=usr/platform/i86pc group=sys
159 $(i386_ONLY)dir path=usr/platform/i86pc/include
160 $(i386_ONLY)dir path=usr/platform/i86pc/include/sys
161 $(i386_ONLY)dir path=usr/platform/i86pc/include/vm
162 $(i386_ONLY)dir path=usr/platform/i86xpv group=sys
163 $(i386_ONLY)dir path=usr/platform/i86xpv/include
164 $(i386_ONLY)dir path=usr/platform/i86xpv/include/sys
165 $(i386_ONLY)dir path=usr/platform/i86xpv/include/vm
166 $(sparc_ONLY)dir path=usr/platform/sun4u group=sys
167 $(sparc_ONLY)dir path=usr/platform/sun4u/include
168 $(sparc_ONLY)dir path=usr/platform/sun4u/include/sys
169 $(sparc_ONLY)dir path=usr/platform/sun4u/include/sys/i2c
170 $(sparc_ONLY)dir path=usr/platform/sun4u/include/sys/i2c/clients
171 $(sparc_ONLY)dir path=usr/platform/sun4u/include/sys/i2c/misc
172 $(sparc_ONLY)dir path=usr/platform/sun4u/include/vm
173 $(sparc_ONLY)dir path=usr/platform/sun4v group=sys
174 $(sparc_ONLY)dir path=usr/platform/sun4v/include
175 $(sparc_ONLY)dir path=usr/platform/sun4v/include/sys
176 $(sparc_ONLY)dir path=usr/platform/sun4v/include/vm
177 dir path=usr/share
178 dir path=usr/share/man
179 dir path=usr/share/man/man3head
180 dir path=usr/share/man/man4
181 dir path=usr/share/man/man5
182 dir path=usr/share/man/man7i
183 dir path=usr/share/src group=sys
184 dir path=usr/share/src/uts
185 $(i386_ONLY)dir path=usr/share/src/uts/i86pc
186 $(i386_ONLY)dir path=usr/share/src/uts/i86xpv
187 $(sparc_ONLY)dir path=usr/share/src/uts/sun4u
188 $(sparc_ONLY)dir path=usr/share/src/uts/sun4v
189 dir path=usr/xpg4

```

```

190 dir path=usr/xpg4/include
191 $(i386_ONLY)file path=usr/include/$(ARCH64)/sys/kdi_regs.h
192 $(i386_ONLY)file path=usr/include/$(ARCH64)/sys/privmregs.h
193 $(i386_ONLY)file path=usr/include/$(ARCH64)/sys/privregs.h
194 file path=usr/include/aio.h
195 file path=usr/include/alloca.h
196 file path=usr/include/apprtrace.h
197 file path=usr/include/apprtrace_impl.h
198 file path=usr/include/ar.h
199 file path=usr/include/archives.h
200 file path=usr/include/arpa/ftp.h
201 file path=usr/include/arpa/inet.h
202 file path=usr/include/arpa/nameser.h
203 file path=usr/include/arpa/nameser_compat.h
204 file path=usr/include/arpa/telnet.h
205 file path=usr/include/arpa/tftp.h
206 $(i386_ONLY)file path=usr/include/asm/atomic.h
207 $(i386_ONLY)file path=usr/include/asm/bitmap.h
208 $(i386_ONLY)file path=usr/include/asm/byteorder.h
209 $(i386_ONLY)file path=usr/include/asm/clock.h
210 $(i386_ONLY)file path=usr/include/asm/cpu.h
211 $(i386_ONLY)file path=usr/include/asm/cpudev.h
212 $(sparc_ONLY)file path=usr/include/asm/flush.h
213 $(i386_ONLY)file path=usr/include/asm/htable.h
214 $(i386_ONLY)file path=usr/include/asm/mmu.h
215 file path=usr/include/asm/sunddi.h
216 file path=usr/include/asm/thread.h
217 file path=usr/include/assert.h
218 file path=usr/include/ast/align.h
219 file path=usr/include/ast/ast.h
220 file path=usr/include/ast/ast_botch.h
221 file path=usr/include/ast/ast_ccode.h
222 file path=usr/include/ast/ast_common.h
223 file path=usr/include/ast/ast_dir.h
224 file path=usr/include/ast/ast_dirent.h
225 file path=usr/include/ast/ast_fcntl.h
226 file path=usr/include/ast/ast_float.h
227 file path=usr/include/ast/ast_fs.h
228 file path=usr/include/ast/ast_getopt.h
229 file path=usr/include/ast/ast_iconv.h
230 file path=usr/include/ast/ast_lib.h
231 file path=usr/include/ast/ast_limits.h
232 file path=usr/include/ast/ast_map.h
233 file path=usr/include/ast/ast_mmap.h
234 file path=usr/include/ast/ast_mode.h
235 file path=usr/include/ast/ast_namval.h
236 file path=usr/include/ast/ast_ndbm.h
237 file path=usr/include/ast/ast_nl_types.h
238 file path=usr/include/ast/ast_param.h
239 file path=usr/include/ast/ast_standards.h
240 file path=usr/include/ast/ast_std.h
241 file path=usr/include/ast/ast_stdio.h
242 file path=usr/include/ast/ast_sys.h
243 file path=usr/include/ast/ast_time.h
244 file path=usr/include/ast/ast_tty.h
245 file path=usr/include/ast/ast_version.h
246 file path=usr/include/ast/ast_vfork.h
247 file path=usr/include/ast/ast_wait.h
248 file path=usr/include/ast/ast_wchar.h
249 file path=usr/include/ast/ast_windows.h
250 file path=usr/include/ast/bytesex.h
251 file path=usr/include/ast/ccode.h
252 file path=usr/include/ast/cdt.h
253 file path=usr/include/ast/cmd.h
254 file path=usr/include/ast/cmdext.h
255 file path=usr/include/ast/debug.h

```

256 file path=usr/include/ast/dirent.h  
257 file path=usr/include/ast/dlldefs.h  
258 file path=usr/include/ast/dt.h  
259 file path=usr/include/ast/ndian.h  
260 file path=usr/include/ast/error.h  
261 file path=usr/include/ast/find.h  
262 file path=usr/include/ast/fnmatch.h  
263 file path=usr/include/ast/fnv.h  
264 file path=usr/include/ast/fs3d.h  
265 file path=usr/include/ast/fts.h  
266 file path=usr/include/ast/ftw.h  
267 file path=usr/include/ast/ftwalk.h  
268 file path=usr/include/ast/getopt.h  
269 file path=usr/include/ast/glob.h  
270 file path=usr/include/ast/hash.h  
271 file path=usr/include/ast/hashkey.h  
272 file path=usr/include/ast/hashpart.h  
273 file path=usr/include/ast/history.h  
274 file path=usr/include/ast/iconv.h  
275 file path=usr/include/ast/ip6.h  
276 file path=usr/include/ast/lc.h  
277 file path=usr/include/ast/ls.h  
278 file path=usr/include/ast/magic.h  
279 file path=usr/include/ast/magicid.h  
280 file path=usr/include/ast/mc.h  
281 file path=usr/include/ast/mime.h  
282 file path=usr/include/ast/mnt.h  
283 file path=usr/include/ast/modecanon.h  
284 file path=usr/include/ast/modex.h  
285 file path=usr/include/ast/namval.h  
286 file path=usr/include/ast/nl\_types.h  
287 file path=usr/include/ast/nval.h  
288 file path=usr/include/ast/option.h  
289 file path=usr/include/ast/preroot.h  
290 file path=usr/include/ast/proc.h  
291 file path=usr/include/ast/prototyped.h  
292 file path=usr/include/ast/re\_comp.h  
293 file path=usr/include/ast/recfmt.h  
294 file path=usr/include/ast/regex.h  
295 file path=usr/include/ast/regexp.h  
296 file path=usr/include/ast/sfdisc.h  
297 file path=usr/include/ast/sfio.h  
298 file path=usr/include/ast/sfio\_s.h  
299 file path=usr/include/ast/sfio\_t.h  
300 file path=usr/include/ast/shcmd.h  
301 file path=usr/include/ast/shell.h  
302 file path=usr/include/ast/sig.h  
303 file path=usr/include/ast/stack.h  
304 file path=usr/include/ast/stak.h  
305 file path=usr/include/ast/stdio.h  
306 file path=usr/include/ast/stk.h  
307 file path=usr/include/ast/sum.h  
308 file path=usr/include/ast/swap.h  
309 file path=usr/include/ast/tar.h  
310 file path=usr/include/ast/times.h  
311 file path=usr/include/ast/tm.h  
312 file path=usr/include/ast/tmx.h  
313 file path=usr/include/ast/tok.h  
314 file path=usr/include/ast/tv.h  
315 file path=usr/include/ast/usage.h  
316 file path=usr/include/ast/vdb.h  
317 file path=usr/include/ast/vecargs.h  
318 file path=usr/include/ast/vmalloc.h  
319 file path=usr/include/ast/wait.h  
320 file path=usr/include/ast/wchar.h  
321 file path=usr/include/ast/wordexp.h

322 file path=usr/include/atomic.h  
323 file path=usr/include/attr.h  
324 file path=usr/include/auth\_attr.h  
325 file path=usr/include/bsm/adt.h  
326 file path=usr/include/bsm/adt\_event.h  
327 file path=usr/include/bsm/audit.h  
328 file path=usr/include/bsm/audit\_kernel.h  
329 file path=usr/include/bsm/audit\_kevents.h  
330 file path=usr/include/bsm/audit\_record.h  
331 file path=usr/include/bsm/audit\_uevents.h  
332 file path=usr/include/bsm/devices.h  
333 file path=usr/include/bsm/libbsm.h  
334 file path=usr/include/config\_admin.h  
335 file path=usr/include/cpio.h  
336 file path=usr/include/crypt.h  
337 file path=usr/include/cryptoutil.h  
338 file path=usr/include/ctype.h  
339 file path=usr/include/curses.h  
340 file path=usr/include/dat/dat.h  
341 file path=usr/include/dat/dat\_error.h  
342 file path=usr/include/dat/dat\_platform\_specific.h  
343 file path=usr/include/dat/dat\_redirection.h  
344 file path=usr/include/dat/dat\_registry.h  
345 file path=usr/include/dat/dat\_vendor\_specific.h  
346 file path=usr/include/dat/udat.h  
347 file path=usr/include/dat/udat\_config.h  
348 file path=usr/include/dat/udat\_redirection.h  
349 file path=usr/include/dat/udat\_vendor\_specific.h  
350 file path=usr/include/default.h  
351 file path=usr/include/des/des.h  
352 file path=usr/include/des/desdata.h  
353 file path=usr/include/des/softdes.h  
354 file path=usr/include/device\_info.h  
355 file path=usr/include/devid.h  
356 file path=usr/include/devmgmt.h  
357 file path=usr/include/devpoll.h  
358 file path=usr/include/dial.h  
359 file path=usr/include/dirent.h  
360 file path=usr/include/dlfcn.h  
361 file path=usr/include/door.h  
362 file path=usr/include/elf.h  
363 file path=usr/include/err.h  
364 file path=usr/include/errno.h  
365 file path=usr/include/eti.h  
366 file path=usr/include/euc.h  
367 file path=usr/include/exacct.h  
368 file path=usr/include/exacct\_impl.h  
369 file path=usr/include/exec\_attr.h  
370 file path=usr/include/execinfo.h  
371 file path=usr/include/fatal.h  
372 file path=usr/include/fcntl.h  
373 file path=usr/include/float.h  
374 file path=usr/include/fmtmsg.h  
375 file path=usr/include/fnmatch.h  
376 file path=usr/include/form.h  
377 file path=usr/include/ftw.h  
378 file path=usr/include/gelf.h  
379 file path=usr/include/getopt.h  
380 file path=usr/include/getwidth.h  
381 file path=usr/include/glob.h  
382 file path=usr/include/grp.h  
383 file path=usr/include/gssapi/gssapi.h  
384 file path=usr/include/gssapi/gssapi\_ext.h  
385 file path=usr/include/hal/libhal-storage.h  
386 file path=usr/include/hal/libhal.h  
387 \$(i386\_ONLY)file path=usr/include/ia32/sys/asm\_linkage.h

```

388 $(i386_ONLY)file path=usr/include/ia32/sys/kdi_regs.h
389 $(i386_ONLY)file path=usr/include/ia32/sys/machtypes.h
390 $(i386_ONLY)file path=usr/include/ia32/sys/privmregs.h
391 $(i386_ONLY)file path=usr/include/ia32/sys/privregs.h
392 $(i386_ONLY)file path=usr/include/ia32/sys/psw.h
393 $(i386_ONLY)file path=usr/include/ia32/sys/pte.h
394 $(i386_ONLY)file path=usr/include/ia32/sys/reg.h
395 $(i386_ONLY)file path=usr/include/ia32/sys/stack.h
396 $(i386_ONLY)file path=usr/include/ia32/sys/trap.h
397 $(i386_ONLY)file path=usr/include/ia32/sys/traptrace.h
398 file path=usr/include/iconv.h
399 file path=usr/include/idmap.h
400 file path=usr/include/ieeefp.h
401 file path=usr/include/ifaddrs.h
402 file path=usr/include/inet/arp.h
403 file path=usr/include/inet/common.h
404 file path=usr/include/inet/ip.h
405 file path=usr/include/inet/ip6.h
406 file path=usr/include/inet/ip6_asp.h
407 file path=usr/include/inet/ip_arp.h
408 file path=usr/include/inet/ip_fhtable.h
409 file path=usr/include/inet/ip_if.h
410 file path=usr/include/inet/ip_ire.h
411 file path=usr/include/inet/ip_multi.h
412 file path=usr/include/inet/ip_netinfo.h
413 file path=usr/include/inet/ip_rts.h
414 file path=usr/include/inet/ip_stack.h
415 file path=usr/include/inet/ipclassifier.h
416 file path=usr/include/inet/ipdrop.h
417 file path=usr/include/inet/ipnet.h
418 file path=usr/include/inet/ipp_common.h
419 file path=usr/include/inet/kssl/ksslap.h
420 file path=usr/include/inet/led.h
421 file path=usr/include/inet/mi.h
422 file path=usr/include/inet/mib2.h
423 file path=usr/include/inet/nd.h
424 file path=usr/include/inet/optcom.h
425 file path=usr/include/inet/sctp_itf.h
426 file path=usr/include/inet/snmpcom.h
427 file path=usr/include/inet/tcp.h
428 file path=usr/include/inet/tcp_sack.h
429 file path=usr/include/inet/tcp_stack.h
430 file path=usr/include/inet/tcp_stats.h
431 file path=usr/include/inet/tunables.h
432 file path=usr/include/inet/wifi_ioctl.h
433 file path=usr/include/inttypes.h
434 file path=usr/include/ipmp.h
435 file path=usr/include/ipmp_admin.h
436 file path=usr/include/ipmp_mpathd.h
437 file path=usr/include/ipmp_query.h
438 file path=usr/include/ipp/ipgpc/ipgpc.h
439 file path=usr/include/ipp/ipp.h
440 file path=usr/include/ipp/ipp_config.h
441 file path=usr/include/ipp/ipp_impl.h
442 file path=usr/include/ipp/ippctl.h
443 file path=usr/include/iso/ctype_c99.h
443 file path=usr/include/iso/ctype_iso.h
444 file path=usr/include/iso/limits_iso.h
445 file path=usr/include/iso/locale_iso.h
446 file path=usr/include/iso/setjmp_iso.h
447 file path=usr/include/iso/signal_iso.h
448 file path=usr/include/iso/stdarg_c99.h
449 file path=usr/include/iso/stdarg_iso.h
450 file path=usr/include/iso/stddef_iso.h
451 file path=usr/include/iso/stdio_c99.h
452 file path=usr/include/iso/stdio_iso.h

```

```

453 file path=usr/include/iso/stdlib_c99.h
454 file path=usr/include/iso/stdlib_iso.h
455 file path=usr/include/iso/string_iso.h
456 file path=usr/include/iso/time_iso.h
457 file path=usr/include/iso/wchar_c99.h
458 file path=usr/include/iso/wchar_iso.h
460 file path=usr/include/iso/wctype_c99.h
459 file path=usr/include/iso/wctype_iso.h
460 file path=usr/include/iso646.h
461 file path=usr/include/kerberosv5/com_err.h
462 file path=usr/include/kerberosv5/krb5.h
463 file path=usr/include/kerberosv5/mit-sipb-copyright.h
464 file path=usr/include/kerberosv5/mit_copyright.h
465 file path=usr/include/klpd.h
466 file path=usr/include/kmfapi.h
467 file path=usr/include/kmftypes.h
468 file path=usr/include/kstat.h
469 file path=usr/include/kvm.h
470 file path=usr/include/langinfo.h
471 file path=usr/include/lastlog.h
472 file path=usr/include/lber.h
473 file path=usr/include/ldap.h
474 file path=usr/include/libcontract.h
475 file path=usr/include/libctf.h
476 file path=usr/include/libdevice.h
477 file path=usr/include/libdevinfo.h
478 file path=usr/include/libdldm.h
479 file path=usr/include/libdlbridge.h
480 file path=usr/include/libdlib.h
481 file path=usr/include/libdllink.h
482 file path=usr/include/libdmpi.h
483 file path=usr/include/libdvlan.h
484 file path=usr/include/libelf.h
485 $(i386_ONLY)file path=usr/include/libfdisk.h
486 file path=usr/include/libfstyp.h
487 file path=usr/include/libfstyp_module.h
488 file path=usr/include/libgen.h
489 file path=usr/include/libgrubgmt.h
490 file path=usr/include/libintl.h
491 file path=usr/include/libipmi.h
492 file path=usr/include/libipp.h
493 file path=usr/include/libnvpair.h
494 file path=usr/include/libnwm.h
495 file path=usr/include/libpolkit/libpolkit.h
496 file path=usr/include/librcm.h
497 file path=usr/include/libscf.h
498 file path=usr/include/libscf_priv.h
499 file path=usr/include/libshare.h
500 file path=usr/include/libsvm.h
501 file path=usr/include/libsysevent.h
502 file path=usr/include/libsysevent_impl.h
503 file path=usr/include/libtsnet.h
504 $(sparc_ONLY)file path=usr/include/libv12n.h
505 file path=usr/include/libw.h
506 file path=usr/include/libzfs.h
507 file path=usr/include/libzfs_core.h
508 file path=usr/include/libzoneinfo.h
509 file path=usr/include/limits.h
510 file path=usr/include/linenum.h
511 file path=usr/include/link.h
512 file path=usr/include/listen.h
513 file path=usr/include/locale.h
514 file path=usr/include/macros.h
515 file path=usr/include/maillock.h
516 file path=usr/include/malloc.h
517 file path=usr/include/md4.h

```

```

518 file path=usr/include/md5.h
519 file path=usr/include/mdiox.h
520 file path=usr/include/mdmn_changelog.h
521 file path=usr/include/memory.h
522 file path=usr/include/menu.h
523 file path=usr/include/meta.h
524 file path=usr/include/meta_basic.h
525 file path=usr/include/meta_runtime.h
526 file path=usr/include/metacl.h
527 file path=usr/include/metad.h
528 file path=usr/include/metadyn.h
529 file path=usr/include/metamed.h
530 file path=usr/include/metamhd.h
531 file path=usr/include/mhdx.h
532 file path=usr/include/mon.h
533 file path=usr/include/monetary.h
534 file path=usr/include/mp.h
535 file path=usr/include/mqueue.h
536 file path=usr/include/mtmalloc.h
537 file path=usr/include/nan.h
538 file path=usr/include/nbdbm.h
539 file path=usr/include/ndpd.h
540 file path=usr/include/net/af.h
541 file path=usr/include/net/bridge.h
542 file path=usr/include/net/if.h
543 file path=usr/include/net/if_arp.h
544 file path=usr/include/net/if_dl.h
545 file path=usr/include/net/if_types.h
546 file path=usr/include/net/pfkeyv2.h
547 file path=usr/include/net/pfpolicy.h
548 file path=usr/include/net/ppp-comp.h
549 file path=usr/include/net/ppp_defs.h
550 file path=usr/include/net/pppio.h
551 file path=usr/include/net/radix.h
552 file path=usr/include/net/route.h
553 file path=usr/include/net/trill.h
554 file path=usr/include/net/vjcompress.h
555 file path=usr/include/netconfig.h
556 file path=usr/include/netdb.h
557 file path=usr/include/netdir.h
558 file path=usr/include/netinet/arp.h
559 file path=usr/include/netinet/dhcp.h
560 file path=usr/include/netinet/dhcp6.h
561 file path=usr/include/netinet/icmp6.h
562 file path=usr/include/netinet/icmp_var.h
563 file path=usr/include/netinet/if_ether.h
564 file path=usr/include/netinet/igmp.h
565 file path=usr/include/netinet/igmp_var.h
566 file path=usr/include/netinet/in.h
567 file path=usr/include/netinet/in_pcb.h
568 file path=usr/include/netinet/in_system.h
569 file path=usr/include/netinet/in_var.h
570 file path=usr/include/netinet/ip.h
571 file path=usr/include/netinet/ip6.h
572 file path=usr/include/netinet/ip_icmp.h
573 file path=usr/include/netinet/ip_mroute.h
574 file path=usr/include/netinet/ip_var.h
575 file path=usr/include/netinet/pim.h
576 file path=usr/include/netinet/sctp.h
577 file path=usr/include/netinet/tcp.h
578 file path=usr/include/netinet/tcp_debug.h
579 file path=usr/include/netinet/tcp_fsm.h
580 file path=usr/include/netinet/tcp_seg.h
581 file path=usr/include/netinet/tcp_timer.h
582 file path=usr/include/netinet/tcp_var.h
583 file path=usr/include/netinet/tcpip.h

```

```

584 file path=usr/include/netinet/udp.h
585 file path=usr/include/netinet/udp_var.h
586 file path=usr/include/netinet/vrrp.h
587 file path=usr/include/nfs/auth.h
588 file path=usr/include/nfs/export.h
589 file path=usr/include/nfs/lm.h
590 file path=usr/include/nfs/mapid.h
591 file path=usr/include/nfs/mount.h
592 file path=usr/include/nfs/nfs.h
593 file path=usr/include/nfs/nfs4.h
594 file path=usr/include/nfs/nfs4_attr.h
595 file path=usr/include/nfs/nfs4_clnt.h
596 file path=usr/include/nfs/nfs4_db_impl.h
597 file path=usr/include/nfs/nfs4_idmap_impl.h
598 file path=usr/include/nfs/nfs4_kprot.h
599 file path=usr/include/nfs/nfs_acl.h
600 file path=usr/include/nfs/nfs_clnt.h
601 file path=usr/include/nfs/nfs_cmd.h
602 file path=usr/include/nfs/nfs_log.h
603 file path=usr/include/nfs/nfs_sec.h
604 file path=usr/include/nfs/nfsid_map.h
605 file path=usr/include/nfs/nfssys.h
606 file path=usr/include/nfs/rnode.h
607 file path=usr/include/nfs/rnode4.h
608 file path=usr/include/nl_types.h
609 file path=usr/include/nlist.h
610 file path=usr/include/note.h
611 file path=usr/include/nss_common.h
612 file path=usr/include/nss_dbdefs.h
613 file path=usr/include/nss_netdir.h
614 file path=usr/include/nsswitch.h
615 file path=usr/include/panel.h
616 file path=usr/include/paths.h
617 file path=usr/include/pcsample.h
618 file path=usr/include/pfmt.h
619 file path=usr/include/pkgdev.h
620 file path=usr/include/pkginfo.h
621 file path=usr/include/pkglocs.h
622 file path=usr/include/pkgstrct.h
623 file path=usr/include/pkgstrans.h
624 file path=usr/include/poll.h
625 file path=usr/include/port.h
626 file path=usr/include/priv.h
627 file path=usr/include/proc_service.h
628 file path=usr/include/procfs.h
629 file path=usr/include/prof.h
630 file path=usr/include/prof_attr.h
631 file path=usr/include/project.h
632 file path=usr/include/protocols/dumppstore.h
633 file path=usr/include/protocols/routed.h
634 file path=usr/include/protocols/rwhod.h
635 file path=usr/include/protocols/timed.h
636 file path=usr/include/pthread.h
637 file path=usr/include/pw.h
638 file path=usr/include/pwd.h
639 file path=usr/include/rcm_module.h
640 file path=usr/include/rctl.h
641 file path=usr/include/re_comp.h
642 file path=usr/include/regex.h
643 file path=usr/include/regexp.h
644 file path=usr/include/regexpr.h
645 file path=usr/include/resolv.h
646 file path=usr/include/rje.h
647 file path=usr/include/rp_plugin.h
648 file path=usr/include/rpc/auth.h
649 file path=usr/include/rpc/auth_des.h

```

```

650 file path=usr/include/rpc/auth_sys.h
651 file path=usr/include/rpc/auth_unix.h
652 file path=usr/include/rpc/bootparam.h
653 file path=usr/include/rpc/clnt.h
654 file path=usr/include/rpc/clnt_soc.h
655 file path=usr/include/rpc/clnt_stat.h
656 file path=usr/include/rpc/des_crypt.h
657 $(sparc_ONLY)file path=usr/include/rpc/ib.h
658 file path=usr/include/rpc/key_prot.h
659 file path=usr/include/rpc/nettype.h
660 file path=usr/include/rpc/pmap_clnt.h
661 file path=usr/include/rpc/pmap_prot.h
662 file path=usr/include/rpc/pmap_prot.x
663 file path=usr/include/rpc/pmap_rmt.h
664 file path=usr/include/rpc/raw.h
665 file path=usr/include/rpc/rpc.h
666 file path=usr/include/rpc/rpc_com.h
667 file path=usr/include/rpc/rpc_msg.h
668 file path=usr/include/rpc/rpc_rdma.h
669 file path=usr/include/rpc/rpc_sztypes.h
670 file path=usr/include/rpc/rpcb_clnt.h
671 file path=usr/include/rpc/rpcb_prot.h
672 file path=usr/include/rpc/rpcb_prot.x
673 file path=usr/include/rpc/rpcent.h
674 file path=usr/include/rpc/rpsec_gss.h
675 file path=usr/include/rpc/rpcsys.h
676 file path=usr/include/rpc/svc.h
677 file path=usr/include/rpc/svc_auth.h
678 file path=usr/include/rpc/svc_mt.h
679 file path=usr/include/rpc/svc_soc.h
680 file path=usr/include/rpc/types.h
681 file path=usr/include/rpc/xdr.h
682 file path=usr/include/rpcsvc/autofs_prot.h
683 file path=usr/include/rpcsvc/autofs_prot.x
684 file path=usr/include/rpcsvc/bootparam.h
685 file path=usr/include/rpcsvc/bootparam_prot.h
686 file path=usr/include/rpcsvc/bootparam_prot.x
687 file path=usr/include/rpcsvc/dbm.h
688 file path=usr/include/rpcsvc/key_prot.x
689 file path=usr/include/rpcsvc/mount.h
690 file path=usr/include/rpcsvc/mount.x
691 file path=usr/include/rpcsvc/nfs4_prot.h
692 file path=usr/include/rpcsvc/nfs4_prot.x
693 file path=usr/include/rpcsvc/nfs_acl.h
694 file path=usr/include/rpcsvc/nfs_acl.x
695 file path=usr/include/rpcsvc/nfs_prot.h
696 file path=usr/include/rpcsvc/nfs_prot.x
697 file path=usr/include/rpcsvc/nis.h
698 file path=usr/include/rpcsvc/nis.x
699 file path=usr/include/rpcsvc/nis_db.h
700 file path=usr/include/rpcsvc/nis_object.x
701 file path=usr/include/rpcsvc/nislib.h
702 file path=usr/include/rpcsvc/nlm_prot.h
703 file path=usr/include/rpcsvc/nlm_prot.x
704 file path=usr/include/rpcsvc/nsm_addr.h
705 file path=usr/include/rpcsvc/nsm_addr.x
706 file path=usr/include/rpcsvc/rex.h
707 file path=usr/include/rpcsvc/rex.x
708 file path=usr/include/rpcsvc/rpc_sztypes.h
709 file path=usr/include/rpcsvc/rpc_sztypes.x
710 file path=usr/include/rpcsvc/rquota.h
711 file path=usr/include/rpcsvc/rquota.x
712 file path=usr/include/rpcsvc/rstat.h
713 file path=usr/include/rpcsvc/rstat.x
714 file path=usr/include/rpcsvc/rusers.h
715 file path=usr/include/rpcsvc/rusers.x

```

```

716 file path=usr/include/rpcsvc/rwall.h
717 file path=usr/include/rpcsvc/rwall.x
718 file path=usr/include/rpcsvc/sm_inter.h
719 file path=usr/include/rpcsvc/sm_inter.x
720 file path=usr/include/rpcsvc/spray.h
721 file path=usr/include/rpcsvc/spray.x
722 file path=usr/include/rpcsvc/ufs_prot.h
723 file path=usr/include/rpcsvc/ufs_prot.x
724 file path=usr/include/rpcsvc/yp.x
725 file path=usr/include/rpcsvc/yp_prot.h
726 file path=usr/include/rpcsvc/ypclnt.h
727 file path=usr/include/rpcsvc/yppasswd.h
728 file path=usr/include/rpcsvc/ypupd.h
729 file path=usr/include/rsmapi.h
730 file path=usr/include/rtld_db.h
731 file path=usr/include/sac.h
732 file path=usr/include/sasl/prop.h
733 file path=usr/include/sasl/sasl.h
734 file path=usr/include/sasl/saslplug.h
735 file path=usr/include/sasl/saslutil.h
736 file path=usr/include/sched.h
737 file path=usr/include/schedctl.h
738 file path=usr/include/scsi/libscsi.h
739 file path=usr/include/scsi/libses.h
740 file path=usr/include/scsi/libses_plugin.h
741 file path=usr/include/scsi/libsmpl.h
742 file path=usr/include/scsi/libsmpl_plugin.h
743 file path=usr/include/scsi/plugins/ses/framework/libses.h
744 file path=usr/include/scsi/plugins/ses/framework/ses2.h
745 file path=usr/include/scsi/plugins/ses/framework/ses2_impl.h
746 file path=usr/include/scsi/plugins/ses/vendor/sun.h
747 file path=usr/include/sdp.h
748 file path=usr/include/search.h
749 file path=usr/include/secdb.h
750 file path=usr/include/security/auditd.h
751 file path=usr/include/security/cryptoki.h
752 file path=usr/include/security/pam_appl.h
753 file path=usr/include/security/pam_modules.h
754 file path=usr/include/security/pkcs11.h
755 file path=usr/include/security/pkcs11f.h
756 file path=usr/include/security/pkcs11t.h
757 file path=usr/include/semaphore.h
758 file path=usr/include/setjmp.h
759 file path=usr/include/sgtty.h
760 file path=usr/include/sha1.h
761 file path=usr/include/sha2.h
762 file path=usr/include/shadow.h
763 file path=usr/include/sharefs/share.h
764 file path=usr/include/sharefs/sharefs.h
765 file path=usr/include/sharefs/sharetab.h
766 file path=usr/include/signinfo.h
767 file path=usr/include/signal.h
768 file path=usr/include/sip.h
769 file path=usr/include/smbios.h
770 file path=usr/include/spawn.h
771 $(i386_ONLY)file path=usr/include/stack_unwind.h
772 file path=usr/include/stdarg.h
773 file path=usr/include/stdbool.h
774 file path=usr/include/stddef.h
775 file path=usr/include/stdint.h
776 file path=usr/include/stdio.h
777 file path=usr/include/stdio_ext.h
778 file path=usr/include/stdio_impl.h
779 file path=usr/include/stdio_tag.h
780 file path=usr/include/stdlib.h
781 file path=usr/include/storclass.h

```

```

782 file path=usr/include/string.h
783 file path=usr/include/strings.h
784 file path=usr/include/stropts.h
785 file path=usr/include/syms.h
786 file path=usr/include/synch.h
787 file path=usr/include/sys/acct.h
788 file path=usr/include/sys/acctctl.h
789 file path=usr/include/sys/acl.h
790 file path=usr/include/sys/acl_impl.h
791 file path=usr/include/sys/acpi_drv.h
792 file path=usr/include/sys/aio.h
793 file path=usr/include/sys/aio_impl.h
794 file path=usr/include/sys/aio_req.h
795 file path=usr/include/sys/aioch.h
796 file path=usr/include/sys/archsystem.h
797 file path=usr/include/sys/ascii.h
798 file path=usr/include/sys/asm_linkage.h
799 file path=usr/include/sys/asynch.h
800 file path=usr/include/sys/atomic.h
801 file path=usr/include/sys/attr.h
802 file path=usr/include/sys/autoconf.h
803 file path=usr/include/sys/auxv.h
804 file path=usr/include/sys/auxv_386.h
805 file path=usr/include/sys/auxv_SPARC.h
806 file path=usr/include/sys/av/iec61883.h
807 file path=usr/include/sys/avintr.h
808 file path=usr/include/sys/avl.h
809 file path=usr/include/sys/avl_impl.h
810 file path=usr/include/sys/bitmap.h
811 file path=usr/include/sys/bitset.h
812 file path=usr/include/sys/bl.h
813 file path=usr/include/sys/blkdev.h
814 file path=usr/include/sys/bofi.h
815 file path=usr/include/sys/bofi_impl.h
816 file path=usr/include/sys/bootconf.h
817 $(i386_ONLY)file path=usr/include/sys/bootregs.h
818 file path=usr/include/sys/bootstat.h
819 $(i386_ONLY)file path=usr/include/sys/bootsvcs.h
820 file path=usr/include/sys/bpp_io.h
821 file path=usr/include/sys/brand.h
822 file path=usr/include/sys/buf.h
823 file path=usr/include/sys/bufmod.h
824 file path=usr/include/sys/bustypes.h
825 file path=usr/include/sys/byteorder.h
826 file path=usr/include/sys/callb.h
827 file path=usr/include/sys/callo.h
828 file path=usr/include/sys/cap_util.h
829 file path=usr/include/sys/ccompile.h
830 file path=usr/include/sys/cdio.h
831 file path=usr/include/sys/cis.h
832 file path=usr/include/sys/cis_handlers.h
833 file path=usr/include/sys/cis_protos.h
834 file path=usr/include/sys/cladm.h
835 file path=usr/include/sys/class.h
836 file path=usr/include/sys/clconf.h
837 file path=usr/include/sys/cmlb.h
838 file path=usr/include/sys/cmn_err.h
839 $(sparc_ONLY)file path=usr/include/sys/cmpregs.h
840 file path=usr/include/sys/compress.h
841 file path=usr/include/sys/concvar.h
842 file path=usr/include/sys/concvar_impl.h
843 file path=usr/include/sys/conf.h
844 file path=usr/include/sys/consdev.h
845 file path=usr/include/sys/console.h
846 file path=usr/include/sys/consplat.h
847 file path=usr/include/sys/contract.h

```

```

848 file path=usr/include/sys/contract/device.h
849 file path=usr/include/sys/contract/device_impl.h
850 file path=usr/include/sys/contract/process.h
851 file path=usr/include/sys/contract/process_impl.h
852 file path=usr/include/sys/contract_impl.h
853 $(i386_ONLY)file path=usr/include/sys/controlregs.h
854 file path=usr/include/sys/copyops.h
855 file path=usr/include/sys/core.h
856 file path=usr/include/sys/corectl.h
857 file path=usr/include/sys/cpc_impl.h
858 file path=usr/include/sys/cpc_pcbe.h
859 file path=usr/include/sys/cpr.h
860 file path=usr/include/sys/cpu.h
861 file path=usr/include/sys/cpucaps.h
862 file path=usr/include/sys/cpucaps_impl.h
863 file path=usr/include/sys/cpupart.h
864 file path=usr/include/sys/cpuvar.h
865 file path=usr/include/sys/crc32.h
866 file path=usr/include/sys/cred.h
867 file path=usr/include/sys/cred_impl.h
868 file path=usr/include/sys/crtctl.h
869 file path=usr/include/sys/crypto/api.h
870 file path=usr/include/sys/crypto/common.h
871 file path=usr/include/sys/crypto/ioctl.h
872 file path=usr/include/sys/crypto/ioctladmin.h
873 file path=usr/include/sys/crypto/spi.h
874 file path=usr/include/sys/cs.h
875 file path=usr/include/sys/cs_priv.h
876 file path=usr/include/sys/cs_strings.h
877 file path=usr/include/sys/cs_stubs.h
878 file path=usr/include/sys/cs_types.h
879 file path=usr/include/sys/csiioctl.h
880 file path=usr/include/sys/ctf.h
881 file path=usr/include/sys/ctf_api.h
882 file path=usr/include/sys/ctfs.h
883 file path=usr/include/sys/ctfs_impl.h
884 file path=usr/include/sys/ctype.h
885 file path=usr/include/sys/cyclic.h
886 file path=usr/include/sys/cyclic_impl.h
887 file path=usr/include/sys/dacf.h
888 file path=usr/include/sys/dacf_impl.h
889 file path=usr/include/sys/damap.h
890 file path=usr/include/sys/damap_impl.h
891 file path=usr/include/sys/dc_ki.h
892 file path=usr/include/sys/ddi.h
893 file path=usr/include/sys/ddi_hp.h
894 file path=usr/include/sys/ddi_hp_impl.h
895 file path=usr/include/sys/ddi_impldefs.h
896 file path=usr/include/sys/ddi_implfuncs.h
897 file path=usr/include/sys/ddi_intr.h
898 file path=usr/include/sys/ddi_intr_impl.h
899 file path=usr/include/sys/ddi_isa.h
900 file path=usr/include/sys/ddi_obsolete.h
901 file path=usr/include/sys/ddi_periodic.h
902 file path=usr/include/sys/ddidevmap.h
903 file path=usr/include/sys/ddidmareq.h
904 file path=usr/include/sys/ddifm.h
905 file path=usr/include/sys/ddifm_impl.h
906 file path=usr/include/sys/ddimapreq.h
907 file path=usr/include/sys/ddipropdefs.h
908 file path=usr/include/sys/dditypes.h
909 file path=usr/include/sys/debug.h
910 $(i386_ONLY)file path=usr/include/sys/debugreg.h
911 file path=usr/include/sys/des.h
912 file path=usr/include/sys/devcache.h
913 file path=usr/include/sys/devcache_impl.h

```

```

914 file path=usr/include/sys/devctl.h
915 file path=usr/include/sys/devfm.h
916 file path=usr/include/sys/devid_cache.h
917 file path=usr/include/sys/devinfo_impl.h
918 file path=usr/include/sys/devops.h
919 file path=usr/include/sys/devpolicy.h
920 file path=usr/include/sys/devpoll.h
921 file path=usr/include/sys/dirent.h
922 file path=usr/include/sys/disp.h
923 file path=usr/include/sys/dkbad.h
924 file path=usr/include/sys/dkio.h
925 file path=usr/include/sys/dklabel.h
926 $(sparc_ONLY)file path=usr/include/sys/dkmpio.h
927 $(i386_ONLY)file path=usr/include/sys/dktp/altctr.h
928 $(i386_ONLY)file path=usr/include/sys/dktp/cmpkt.h
929 file path=usr/include/sys/dktp/dadkio.h
930 file path=usr/include/sys/dktp/fdisk.h
931 file path=usr/include/sys/dl.h
932 file path=usr/include/sys/dld.h
933 file path=usr/include/sys/dlpi.h
934 file path=usr/include/sys/dls_mgmt.h
935 $(i386_ONLY)file path=usr/include/sys/dma_engine.h
936 file path=usr/include/sys/dma_i8237A.h
937 file path=usr/include/sys/dnnc.h
938 file path=usr/include/sys/door.h
939 file path=usr/include/sys/door_data.h
940 file path=usr/include/sys/door_impl.h
941 file path=usr/include/sys/dumphdr.h
942 file path=usr/include/sys/ecppio.h
943 file path=usr/include/sys/ecppreg.h
944 file path=usr/include/sys/ecppsys.h
945 file path=usr/include/sys/ecppvar.h
946 file path=usr/include/sys/efi_partition.h
947 file path=usr/include/sys/elf.h
948 file path=usr/include/sys/elf_386.h
949 file path=usr/include/sys/elf_SPARC.h
950 file path=usr/include/sys/elf_amd64.h
951 file path=usr/include/sys/elf_notes.h
952 file path=usr/include/sys/elftypes.h
953 file path=usr/include/sys/epm.h
954 file path=usr/include/sys/errno.h
955 file path=usr/include/sys/errorq.h
956 file path=usr/include/sys/errorq_impl.h
957 file path=usr/include/sys/esunddi.h
958 file path=usr/include/sys/ethernet.h
959 file path=usr/include/sys/euc.h
960 file path=usr/include/sys/eucioctl.h
961 file path=usr/include/sys/exacct.h
962 file path=usr/include/sys/exacct_catalog.h
963 file path=usr/include/sys/exacct_impl.h
964 file path=usr/include/sys/exec.h
965 file path=usr/include/sys/exechdr.h
966 file path=usr/include/sys/fault.h
967 file path=usr/include/sys/fbio.h
968 file path=usr/include/sys/fbuf.h
969 file path=usr/include/sys/fc4/fc.h
970 file path=usr/include/sys/fc4/fc_transport.h
971 file path=usr/include/sys/fc4/fcal.h
972 file path=usr/include/sys/fc4/fcal_linkapp.h
973 file path=usr/include/sys/fc4/fcal_transport.h
974 file path=usr/include/sys/fc4/fcio.h
975 file path=usr/include/sys/fc4/fcp.h
976 file path=usr/include/sys/fc4/linkapp.h
977 file path=usr/include/sys/fcntl.h
978 file path=usr/include/sys/fdbuffer.h
979 file path=usr/include/sys/fdio.h

```

```

980 $(sparc_ONLY)file path=usr/include/sys/fdreg.h
981 $(sparc_ONLY)file path=usr/include/sys/fdvar.h
982 file path=usr/include/sys/feature_tests.h
983 file path=usr/include/sys/fem.h
984 file path=usr/include/sys/file.h
985 file path=usr/include/sys/filio.h
986 file path=usr/include/sys/flock.h
987 file path=usr/include/sys/flock_impl.h
988 $(sparc_ONLY)file path=usr/include/sys/fm/cpu/SPARC64-VI.h
989 $(sparc_ONLY)file path=usr/include/sys/fm/cpu/UltraSPARC-II.h
990 $(sparc_ONLY)file path=usr/include/sys/fm/cpu/UltraSPARC-III.h
991 $(sparc_ONLY)file path=usr/include/sys/fm/cpu/UltraSPARC-T1.h
992 file path=usr/include/sys/fm/fs/zfs.h
993 file path=usr/include/sys/fm/io/ddi.h
994 file path=usr/include/sys/fm/io/disk.h
995 file path=usr/include/sys/fm/io/opl_mc_fm.h
996 file path=usr/include/sys/fm/io/pci.h
997 file path=usr/include/sys/fm/io/scsi.h
998 file path=usr/include/sys/fm/io/sun4upci.h
999 file path=usr/include/sys/fm/protocol.h
1000 file path=usr/include/sys/fm/util.h
1001 file path=usr/include/sys/fork.h
1002 $(i386_ONLY)file path=usr/include/sys/fp.h
1003 $(sparc_ONLY)file path=usr/include/sys/fpu/fpu_simulator.h
1004 $(sparc_ONLY)file path=usr/include/sys/fpu/fpusystem.h
1005 $(sparc_ONLY)file path=usr/include/sys/fpu/globals.h
1006 $(sparc_ONLY)file path=usr/include/sys/fpu/ieee.h
1007 file path=usr/include/sys/frame.h
1008 file path=usr/include/sys/fs/autofs.h
1009 file path=usr/include/sys/fs/cacheofs_dir.h
1010 file path=usr/include/sys/fs/cacheofs_dlog.h
1011 file path=usr/include/sys/fs/cacheofs_filegrp.h
1012 file path=usr/include/sys/fs/cacheofs_fs.h
1013 file path=usr/include/sys/fs/cacheofs_fscache.h
1014 file path=usr/include/sys/fs/cacheofs_ioctl.h
1015 file path=usr/include/sys/fs/cacheofs_log.h
1016 file path=usr/include/sys/fs/decomp.h
1017 file path=usr/include/sys/fs/dv_node.h
1018 file path=usr/include/sys/fs/fifonode.h
1019 file path=usr/include/sys/fs/hsfs_isospec.h
1020 file path=usr/include/sys/fs/hsfs_node.h
1021 file path=usr/include/sys/fs/hsfs_rrip.h
1022 file path=usr/include/sys/fs/hsfs_spec.h
1023 file path=usr/include/sys/fs/hsfs_susp.h
1024 file path=usr/include/sys/fs/lofs_info.h
1025 file path=usr/include/sys/fs/lofs_node.h
1026 file path=usr/include/sys/fs/mntdata.h
1027 file path=usr/include/sys/fs/namenode.h
1028 file path=usr/include/sys/fs/pc_dir.h
1029 file path=usr/include/sys/fs/pc_fs.h
1030 file path=usr/include/sys/fs/pc_label.h
1031 file path=usr/include/sys/fs/pc_node.h
1032 file path=usr/include/sys/fs/pxfs_ki.h
1033 file path=usr/include/sys/fs/sdev_impl.h
1034 file path=usr/include/sys/fs/snnode.h
1035 file path=usr/include/sys/fs/swapnode.h
1036 file path=usr/include/sys/fs/tmp.h
1037 file path=usr/include/sys/fs/tmpnode.h
1038 file path=usr/include/sys/fs/udf_inode.h
1039 file path=usr/include/sys/fs/udf_volume.h
1040 file path=usr/include/sys/fs/ufs_acl.h
1041 file path=usr/include/sys/fs/ufs_bio.h
1042 file path=usr/include/sys/fs/ufs_filio.h
1043 file path=usr/include/sys/fs/ufs_fs.h
1044 file path=usr/include/sys/fs/ufs_fsdire.h
1045 file path=usr/include/sys/fs/ufs_inode.h

```



1046 file path=usr/include/sys/fs/ufs\_lockfs.h  
 1047 file path=usr/include/sys/fs/ufs\_log.h  
 1048 file path=usr/include/sys/fs/ufs\_mount.h  
 1049 file path=usr/include/sys/fs/ufs\_panic.h  
 1050 file path=usr/include/sys/fs/ufs\_prot.h  
 1051 file path=usr/include/sys/fs/ufs\_quota.h  
 1052 file path=usr/include/sys/fs/ufs\_snap.h  
 1053 file path=usr/include/sys/fs/ufs\_trans.h  
 1054 file path=usr/include/sys/fs/zfs.h  
 1055 file path=usr/include/sys/fs\_reparse.h  
 1056 file path=usr/include/sys/fs\_subr.h  
 1057 file path=usr/include/sys/fsid.h  
 1058 \$(sparc\_ONLY)file path=usr/include/sys/fsr.h  
 1059 file path=usr/include/sys/fss.h  
 1060 file path=usr/include/sys/fssnap.h  
 1061 file path=usr/include/sys/fssnap\_if.h  
 1062 file path=usr/include/sys/fsspriocntl.h  
 1063 file path=usr/include/sys/fstyp.h  
 1064 file path=usr/include/sys/fttrace.h  
 1065 file path=usr/include/sys/fix.h  
 1066 file path=usr/include/sys/fixpriocntl.h  
 1067 file path=usr/include/sys/gfs.h  
 1068 file path=usr/include/sys/gld.h  
 1069 file path=usr/include/sys/gldpriv.h  
 1070 file path=usr/include/sys/group.h  
 1071 file path=usr/include/sys/hdio.h  
 1072 file path=usr/include/sys/hook.h  
 1073 file path=usr/include/sys/hook\_event.h  
 1074 file path=usr/include/sys/hook\_impl.h  
 1075 file path=usr/include/sys/hotplug/hpcsvc.h  
 1076 file path=usr/include/sys/hotplug/hpctrl.h  
 1077 file path=usr/include/sys/hotplug/pci/pcicfg.h  
 1078 file path=usr/include/sys/hotplug/pci/pcihp.h  
 1079 file path=usr/include/sys/hwconf.h  
 1080 \$(i386\_ONLY)file path=usr/include/sys/hypervisor.h  
 1081 \$(i386\_ONLY)file path=usr/include/sys/i8272A.h  
 1082 file path=usr/include/sys/ia.h  
 1083 file path=usr/include/sys/iapriocntl.h  
 1084 file path=usr/include/sys/ib/adapters/hermon/hermon\_ioctl.h  
 1085 file path=usr/include/sys/ib/adapters/mlnx\_umap.h  
 1086 file path=usr/include/sys/ib/adapters/tavor/tavor\_ioctl.h  
 1087 file path=usr/include/sys/ib/clients/ibd/ibd.h  
 1088 file path=usr/include/sys/ib/clients/of/ofa\_solaris.h  
 1089 file path=usr/include/sys/ib/clients/of/ofed\_kernel.h  
 1090 file path=usr/include/sys/ib/clients/of/rdma/ib\_addr.h  
 1091 file path=usr/include/sys/ib/clients/of/rdma/ib\_user\_mad.h  
 1092 file path=usr/include/sys/ib/clients/of/rdma/ib\_user\_sa.h  
 1093 file path=usr/include/sys/ib/clients/of/rdma/ib\_user\_verbs.h  
 1094 file path=usr/include/sys/ib/clients/of/rdma/ib\_verbs.h  
 1095 file path=usr/include/sys/ib/clients/of/rdma/rdma\_cm.h  
 1096 file path=usr/include/sys/ib/clients/of/rdma/rdma\_user\_cm.h  
 1097 file path=usr/include/sys/ib/clients/of/sol\_ofs/sol\_cma.h  
 1098 file path=usr/include/sys/ib/clients/of/sol\_ofs/sol\_ib\_cma.h  
 1099 file path=usr/include/sys/ib/clients/of/sol\_ofs/sol\_kverb\_impl.h  
 1100 file path=usr/include/sys/ib/clients/of/sol\_ofs/sol\_ofs\_common.h  
 1101 file path=usr/include/sys/ib/clients/of/sol\_ucma/sol\_rdma\_user\_cm.h  
 1102 file path=usr/include/sys/ib/clients/of/sol\_ucma/sol\_ucma.h  
 1103 file path=usr/include/sys/ib/clients/of/sol\_umad/sol\_umad.h  
 1104 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs.h  
 1105 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs2ucma.h  
 1106 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs\_comp.h  
 1107 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs\_event.h  
 1108 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs\_hca.h  
 1109 file path=usr/include/sys/ib/clients/of/sol\_uverbs/sol\_uverbs\_qp.h  
 1110 file path=usr/include/sys/ib/ib\_pkt\_hdrs.h  
 1111 file path=usr/include/sys/ib/ib\_types.h

1112 file path=usr/include/sys/ib/ibnex/ibnex\_devctl.h  
 1113 file path=usr/include/sys/ib/ibtl/ibci.h  
 1114 file path=usr/include/sys/ib/ibtl/ibti.h  
 1115 file path=usr/include/sys/ib/ibtl/ibti\_cm.h  
 1116 file path=usr/include/sys/ib/ibtl/ibti\_common.h  
 1117 file path=usr/include/sys/ib/ibtl/ibtl\_ci\_types.h  
 1118 file path=usr/include/sys/ib/ibtl/ibtl\_status.h  
 1119 file path=usr/include/sys/ib/ibtl/ibtl\_types.h  
 1120 file path=usr/include/sys/ib/ibtl/ibvti.h  
 1121 file path=usr/include/sys/ib/ibtl/impl/ibtl\_util.h  
 1122 file path=usr/include/sys/ib/mgt/ib\_dm\_attr.h  
 1123 file path=usr/include/sys/ib/mgt/ib\_mad.h  
 1124 file path=usr/include/sys/ib/mgt/ibmf/ibmf.h  
 1125 file path=usr/include/sys/ib/mgt/ibmf/ibmf\_msg.h  
 1126 file path=usr/include/sys/ib/mgt/ibmf/ibmf\_saa.h  
 1127 file path=usr/include/sys/ib/mgt/ibmf/ibmf\_utils.h  
 1128 file path=usr/include/sys/ib/mgt/sa\_recs.h  
 1129 file path=usr/include/sys/ib/mgt/sm\_attr.h  
 1130 file path=usr/include/sys/ibpart.h  
 1131 file path=usr/include/sys/id32.h  
 1132 file path=usr/include/sys/id\_space.h  
 1133 file path=usr/include/sys/idmap.h  
 1134 file path=usr/include/sys/inline.h  
 1135 file path=usr/include/sys/instance.h  
 1136 file path=usr/include/sys/int\_const.h  
 1137 file path=usr/include/sys/int\_fmtio.h  
 1138 file path=usr/include/sys/int\_limits.h  
 1139 file path=usr/include/sys/int\_types.h  
 1140 file path=usr/include/sys/inttypes.h  
 1141 file path=usr/include/sys/ioccom.h  
 1142 file path=usr/include/sys/ioctl.h  
 1143 \$(i386\_ONLY)file path=usr/include/sys/iomulib.h  
 1144 file path=usr/include/sys/ipc.h  
 1145 file path=usr/include/sys/ipc\_impl.h  
 1146 file path=usr/include/sys/ipc\_rctl.h  
 1147 file path=usr/include/sys/isa\_defs.h  
 1148 file path=usr/include/sys/iso/signal\_iso.h  
 1149 file path=usr/include/sys/jioctl.h  
 1150 file path=usr/include/sys/kbd.h  
 1151 file path=usr/include/sys/kbdreg.h  
 1152 file path=usr/include/sys/kbio.h  
 1153 file path=usr/include/sys/kcpc.h  
 1154 file path=usr/include/sys/kd.h  
 1155 file path=usr/include/sys/kdi.h  
 1156 file path=usr/include/sys/kdi\_impl.h  
 1157 file path=usr/include/sys/kdi\_machimpl.h  
 1158 \$(i386\_ONLY)file path=usr/include/sys/kdi\_regs.h  
 1159 file path=usr/include/sys/kiconv.h  
 1160 file path=usr/include/sys/kidmap.h  
 1161 file path=usr/include/sys/klpd.h  
 1162 file path=usr/include/sys/klwp.h  
 1163 file path=usr/include/sys/kmem.h  
 1164 file path=usr/include/sys/kmem\_impl.h  
 1165 file path=usr/include/sys/kobj.h  
 1166 file path=usr/include/sys/kobj\_impl.h  
 1167 file path=usr/include/sys/ksocket.h  
 1168 file path=usr/include/sys/kstat.h  
 1169 file path=usr/include/sys/kstr.h  
 1170 file path=usr/include/sys/ksyms.h  
 1171 file path=usr/include/sys/ksynch.h  
 1172 file path=usr/include/sys/lc\_core.h  
 1173 file path=usr/include/sys/ldterm.h  
 1174 file path=usr/include/sys/lgrp.h  
 1175 file path=usr/include/sys/lgrp\_user.h  
 1176 file path=usr/include/sys/link.h  
 1177 file path=usr/include/sys/list.h

1178 file path=usr/include/sys/list\_impl.h  
 1179 file path=usr/include/sys/llcl.h  
 1180 file path=usr/include/sys/loadavg.h  
 1181 file path=usr/include/sys/localedef.h  
 1182 file path=usr/include/sys/lock.h  
 1183 file path=usr/include/sys/lockfs.h  
 1184 file path=usr/include/sys/lofi.h  
 1185 file path=usr/include/sys/log.h  
 1186 file path=usr/include/sys/logindmux.h  
 1187 file path=usr/include/sys/lvm/md\_basic.h  
 1188 file path=usr/include/sys/lvm/md\_convert.h  
 1189 file path=usr/include/sys/lvm/md\_crc.h  
 1190 file path=usr/include/sys/lvm/md\_hotspares.h  
 1191 file path=usr/include/sys/lvm/md\_mddb.h  
 1192 file path=usr/include/sys/lvm/md\_mdiox.h  
 1193 file path=usr/include/sys/lvm/md\_mhdx.h  
 1194 file path=usr/include/sys/lvm/md\_mirror.h  
 1195 file path=usr/include/sys/lvm/md\_mirror\_shared.h  
 1196 file path=usr/include/sys/lvm/md\_names.h  
 1197 file path=usr/include/sys/lvm/md\_notify.h  
 1198 file path=usr/include/sys/lvm/md\_raid.h  
 1199 file path=usr/include/sys/lvm/md\_rename.h  
 1200 file path=usr/include/sys/lvm/md\_sp.h  
 1201 file path=usr/include/sys/lvm/md\_stripe.h  
 1202 file path=usr/include/sys/lvm/md\_trans.h  
 1203 file path=usr/include/sys/lvm/mdio.h  
 1204 file path=usr/include/sys/lvm/mdmed.h  
 1205 file path=usr/include/sys/lvm/mdmn\_commd.h  
 1206 file path=usr/include/sys/lvm/mdvar.h  
 1207 file path=usr/include/sys/lwp.h  
 1208 file path=usr/include/sys/lwp\_timer\_impl.h  
 1209 file path=usr/include/sys/lwp\_upimutex\_impl.h  
 1210 file path=usr/include/sys/mac.h  
 1211 file path=usr/include/sys/mac\_ether.h  
 1212 file path=usr/include/sys/mac\_flow.h  
 1213 file path=usr/include/sys/mac\_provider.h  
 1214 file path=usr/include/sys/machelf.h  
 1215 file path=usr/include/sys/machlock.h  
 1216 file path=usr/include/sys/machsig.h  
 1217 file path=usr/include/sys/machtypes.h  
 1218 file path=usr/include/sys/map.h  
 1219 \$(i386\_ONLY)file path=usr/include/sys/mc.h  
 1220 \$(i386\_ONLY)file path=usr/include/sys/mc\_amd.h  
 1221 \$(i386\_ONLY)file path=usr/include/sys/mc\_intel.h  
 1222 \$(i386\_ONLY)file path=usr/include/sys/mca\_amd.h  
 1223 \$(i386\_ONLY)file path=usr/include/sys/mca\_x86.h  
 1224 file path=usr/include/sys/md4.h  
 1225 file path=usr/include/sys/md5.h  
 1226 file path=usr/include/sys/md5\_consts.h  
 1227 file path=usr/include/sys/mdi\_impldefs.h  
 1228 file path=usr/include/sys/mem.h  
 1229 file path=usr/include/sys/mem\_config.h  
 1230 file path=usr/include/sys/memlist.h  
 1231 file path=usr/include/sys/mhd.h  
 1232 file path=usr/include/sys/mii.h  
 1233 file path=usr/include/sys/miiregs.h  
 1234 file path=usr/include/sys/mkdev.h  
 1235 file path=usr/include/sys/mman.h  
 1236 file path=usr/include/sys/mmapobj.h  
 1237 file path=usr/include/sys/mntent.h  
 1238 file path=usr/include/sys/mntio.h  
 1239 file path=usr/include/sys/mnttab.h  
 1240 file path=usr/include/sys/modctl.h  
 1241 file path=usr/include/sys/mod.h  
 1242 file path=usr/include/sys/model.h  
 1243 file path=usr/include/sys/modhash.h

1244 file path=usr/include/sys/modhash\_impl.h  
 1245 file path=usr/include/sys/mount.h  
 1246 file path=usr/include/sys/mouse.h  
 1247 file path=usr/include/sys/msacct.h  
 1248 file path=usr/include/sys/msg.h  
 1249 file path=usr/include/sys/msg\_impl.h  
 1250 file path=usr/include/sys/msio.h  
 1251 file path=usr/include/sys/msreg.h  
 1252 file path=usr/include/sys/mtio.h  
 1253 file path=usr/include/sys/multidata.h  
 1254 file path=usr/include/sys/mutex.h  
 1255 \$(i386\_ONLY)file path=usr/include/sys/mutex\_impl.h  
 1256 file path=usr/include/sys/nbmlck.h  
 1257 file path=usr/include/sys/ndi\_impldefs.h  
 1258 file path=usr/include/sys/ndifm.h  
 1259 file path=usr/include/sys/netconfig.h  
 1260 file path=usr/include/sys/neti.h  
 1261 file path=usr/include/sys/netstack.h  
 1262 file path=usr/include/sys/nexusdefs.h  
 1263 file path=usr/include/sys/note.h  
 1264 file path=usr/include/sys/nvpair.h  
 1265 file path=usr/include/sys/nvpair\_impl.h  
 1266 file path=usr/include/sys/objfs.h  
 1267 file path=usr/include/sys/objfs\_impl.h  
 1268 file path=usr/include/sys/obpdefs.h  
 1269 file path=usr/include/sys/old\_procfs.h  
 1270 file path=usr/include/sys/open.h  
 1271 file path=usr/include/sys/openpromio.h  
 1272 file path=usr/include/sys/panic.h  
 1273 file path=usr/include/sys/param.h  
 1274 file path=usr/include/sys/pathconf.h  
 1275 file path=usr/include/sys/pathname.h  
 1276 file path=usr/include/sys/ptr.h  
 1277 file path=usr/include/sys/pbio.h  
 1278 file path=usr/include/sys/pcb.h  
 1279 file path=usr/include/sys/pccard.h  
 1280 file path=usr/include/sys/pci.h  
 1281 \$(i386\_ONLY)file path=usr/include/sys/pcic\_reg.h  
 1282 \$(i386\_ONLY)file path=usr/include/sys/pcic\_var.h  
 1283 file path=usr/include/sys/pcie.h  
 1284 file path=usr/include/sys/pcmcia.h  
 1285 file path=usr/include/sys/pctypes.h  
 1286 file path=usr/include/sys/pfmod.h  
 1287 file path=usr/include/sys/pg.h  
 1288 file path=usr/include/sys/pghw.h  
 1289 file path=usr/include/sys/phymem.h  
 1290 \$(i386\_ONLY)file path=usr/include/sys/pic.h  
 1291 \$(i386\_ONLY)file path=usr/include/sys/pit.h  
 1292 file path=usr/include/sys/pkp\_hash.h  
 1293 file path=usr/include/sys/pm.h  
 1294 \$(i386\_ONLY)file path=usr/include/sys/pmem.h  
 1295 file path=usr/include/sys/policy.h  
 1296 file path=usr/include/sys/poll.h  
 1297 file path=usr/include/sys/poll\_impl.h  
 1298 file path=usr/include/sys/pool.h  
 1299 file path=usr/include/sys/pool\_impl.h  
 1300 file path=usr/include/sys/pool\_pset.h  
 1301 file path=usr/include/sys/port.h  
 1302 file path=usr/include/sys/port\_impl.h  
 1303 file path=usr/include/sys/port\_kernel.h  
 1304 file path=usr/include/sys/ppmio.h  
 1305 file path=usr/include/sys/priocntl.h  
 1306 file path=usr/include/sys/priv.h  
 1307 file path=usr/include/sys/priv\_const.h  
 1308 file path=usr/include/sys/priv\_impl.h  
 1309 file path=usr/include/sys/priv\_names.h

```

1310 $(i386_ONLY)file path=usr/include/sys/privmregs.h
1311 $(i386_ONLY)file path=usr/include/sys/privregs.h
1312 file path=usr/include/sys/prnio.h
1313 file path=usr/include/sys/proc.h
1314 file path=usr/include/sys/proc/prdata.h
1315 file path=usr/include/sys/processor.h
1316 file path=usr/include/sys/procfs.h
1317 file path=usr/include/sys/procfs_isa.h
1318 file path=usr/include/sys/procset.h
1319 file path=usr/include/sys/project.h
1320 $(i386_ONLY)file path=usr/include/sys/prom_emul.h
1321 $(i386_ONLY)file path=usr/include/sys/prom_isa.h
1322 $(i386_ONLY)file path=usr/include/sys/prom_plat.h
1323 file path=usr/include/sys/promif.h
1324 file path=usr/include/sys/promimpl.h
1325 file path=usr/include/sys/protosw.h
1326 file path=usr/include/sys/prsysm.h
1327 file path=usr/include/sys/pset.h
1328 file path=usr/include/sys/psw.h
1329 $(i386_ONLY)file path=usr/include/sys/pte.h
1330 file path=usr/include/sys/ptem.h
1331 file path=usr/include/sys/ptms.h
1332 file path=usr/include/sys/ptyvar.h
1333 file path=usr/include/sys/queue.h
1334 file path=usr/include/sys/raidioctl.h
1335 file path=usr/include/sys/ramdisk.h
1336 file path=usr/include/sys/random.h
1337 file path=usr/include/sys/rctl.h
1338 file path=usr/include/sys/rctl_impl.h
1339 file path=usr/include/sys/rds.h
1340 file path=usr/include/sys/reboot.h
1341 file path=usr/include/sys/refstr.h
1342 file path=usr/include/sys/refstr_impl.h
1343 file path=usr/include/sys/reg.h
1344 file path=usr/include/sys/regset.h
1345 file path=usr/include/sys/resource.h
1346 file path=usr/include/sys/rliocntl.h
1347 file path=usr/include/sys/rsm/rsm.h
1348 file path=usr/include/sys/rsm/rsm_common.h
1349 file path=usr/include/sys/rsm/rsmapi_common.h
1350 file path=usr/include/sys/rsm/rsmka_path_int.h
1351 file path=usr/include/sys/rsm/rsmmdi.h
1352 file path=usr/include/sys/rsm/rsmmpi.h
1353 file path=usr/include/sys/rsm/rsmpi_driver.h
1354 file path=usr/include/sys/rt.h
1355 $(i386_ONLY)file path=usr/include/sys/rtc.h
1356 file path=usr/include/sys/rtpriocntl.h
1357 file path=usr/include/sys/rwlock.h
1358 file path=usr/include/sys/rwlock_impl.h
1359 file path=usr/include/sys/rwstlock.h
1360 file path=usr/include/sys/sad.h
1361 $(i386_ONLY)file path=usr/include/sys/sata/sata_defs.h
1362 $(i386_ONLY)file path=usr/include/sys/sata/sata_hba.h
1363 file path=usr/include/sys/schedctl.h
1364 $(sparc_ONLY)file path=usr/include/sys/scsi/adapters/ifpio.h
1365 file path=usr/include/sys/scsi/adapters/scsi_vhci.h
1366 $(sparc_ONLY)file path=usr/include/sys/scsi/adapters/sfvar.h
1367 file path=usr/include/sys/scsi/conf/autoconf.h
1368 file path=usr/include/sys/scsi/conf/device.h
1369 file path=usr/include/sys/scsi/generic/commands.h
1370 file path=usr/include/sys/scsi/generic/dad_mode.h
1371 file path=usr/include/sys/scsi/generic/inquiry.h
1372 file path=usr/include/sys/scsi/generic/message.h
1373 file path=usr/include/sys/scsi/generic/mode.h
1374 file path=usr/include/sys/scsi/generic/persist.h
1375 file path=usr/include/sys/scsi/generic/sense.h

```

```

1376 file path=usr/include/sys/scsi/generic/sff_frames.h
1377 file path=usr/include/sys/scsi/generic/smp_frames.h
1378 file path=usr/include/sys/scsi/generic/status.h
1379 file path=usr/include/sys/scsi/impl/commands.h
1380 file path=usr/include/sys/scsi/impl/inquiry.h
1381 file path=usr/include/sys/scsi/impl/mode.h
1382 file path=usr/include/sys/scsi/impl/scsi_reset_notify.h
1383 file path=usr/include/sys/scsi/impl/scsi_sas.h
1384 file path=usr/include/sys/scsi/impl/sense.h
1385 file path=usr/include/sys/scsi/impl/services.h
1386 file path=usr/include/sys/scsi/impl/smp_transport.h
1387 file path=usr/include/sys/scsi/impl/spc3_types.h
1388 file path=usr/include/sys/scsi/impl/status.h
1389 file path=usr/include/sys/scsi/impl/transport.h
1390 file path=usr/include/sys/scsi/impl/types.h
1391 file path=usr/include/sys/scsi/impl/uscsi.h
1392 file path=usr/include/sys/scsi/impl/usmp.h
1393 file path=usr/include/sys/scsi/scsi.h
1394 file path=usr/include/sys/scsi/scsi_address.h
1395 file path=usr/include/sys/scsi/scsi_ctl.h
1396 file path=usr/include/sys/scsi/scsi_fm.h
1397 file path=usr/include/sys/scsi/scsi_params.h
1398 file path=usr/include/sys/scsi/scsi_pkt.h
1399 file path=usr/include/sys/scsi/scsi_resource.h
1400 file path=usr/include/sys/scsi/scsi_types.h
1401 file path=usr/include/sys/scsi/scsi_watch.h
1402 file path=usr/include/sys/scsi/targets/sddef.h
1403 file path=usr/include/sys/scsi/targets/ses.h
1404 file path=usr/include/sys/scsi/targets/sesio.h
1405 file path=usr/include/sys/scsi/targets/sgendef.h
1406 file path=usr/include/sys/scsi/targets/smp.h
1407 $(sparc_ONLY)file path=usr/include/sys/scsi/targets/ssddef.h
1408 file path=usr/include/sys/scsi/targets/stdef.h
1409 $(i386_ONLY)file path=usr/include/sys/segment.h
1410 $(i386_ONLY)file path=usr/include/sys/segments.h
1411 file path=usr/include/sys/select.h
1412 file path=usr/include/sys/sem.h
1413 file path=usr/include/sys/sem_impl.h
1414 file path=usr/include/sys/sem_impl.h
1415 file path=usr/include/sys/semaphore.h
1416 file path=usr/include/sys/sendfile.h
1417 $(sparc_ONLY)file path=usr/include/sys/ser_async.h
1418 file path=usr/include/sys/ser_sync.h
1419 $(sparc_ONLY)file path=usr/include/sys/ser_zscc.h
1420 file path=usr/include/sys/serializer.h
1421 file path=usr/include/sys/session.h
1422 file path=usr/include/sys/sha1.h
1423 file path=usr/include/sys/sha2.h
1424 file path=usr/include/sys/share.h
1425 file path=usr/include/sys/shm.h
1426 file path=usr/include/sys/shm_impl.h
1427 file path=usr/include/sys/sid.h
1428 file path=usr/include/sys/siginfo.h
1429 file path=usr/include/sys/signal.h
1430 file path=usr/include/sys/sleepq.h
1431 file path=usr/include/sys/smbios.h
1432 file path=usr/include/sys/smbios_impl.h
1433 file path=usr/include/sys/smedia.h
1434 file path=usr/include/sys/subject.h
1435 $(sparc_ONLY)file path=usr/include/sys/social_cq_defs.h
1436 $(sparc_ONLY)file path=usr/include/sys/socialio.h
1437 $(sparc_ONLY)file path=usr/include/sys/socialmap.h
1438 $(sparc_ONLY)file path=usr/include/sys/socialreg.h
1439 $(sparc_ONLY)file path=usr/include/sys/socialvar.h
1440 file path=usr/include/sys/socket.h
1441 file path=usr/include/sys/socket_impl.h

```

1442 file path=usr/include/sys/socket\_proto.h  
 1443 file path=usr/include/sys/socketvar.h  
 1444 file path=usr/include/sys/sockio.h  
 1445 file path=usr/include/sys/spl.h  
 1446 file path=usr/include/sys/squeue.h  
 1447 file path=usr/include/sys/squeue\_impl.h  
 1448 file path=usr/include/sys/sservice.h  
 1449 file path=usr/include/sys/stack.h  
 1450 file path=usr/include/sys/stat.h  
 1451 file path=usr/include/sys/stat\_impl.h  
 1452 file path=usr/include/sys/statfs.h  
 1453 file path=usr/include/sys/statvfs.h  
 1454 file path=usr/include/sys/stdbool.h  
 1455 file path=usr/include/sys/stdint.h  
 1456 file path=usr/include/sys/stermio.h  
 1457 file path=usr/include/sys/stream.h  
 1458 file path=usr/include/sys/strft.h  
 1459 file path=usr/include/sys/strlog.h  
 1460 file path=usr/include/sys/strmdep.h  
 1461 file path=usr/include/sys/stropts.h  
 1462 file path=usr/include/sys/strredir.h  
 1463 file path=usr/include/sys/strstat.h  
 1464 file path=usr/include/sys/strsubr.h  
 1465 file path=usr/include/sys/strsun.h  
 1466 file path=usr/include/sys/strtty.h  
 1467 file path=usr/include/sys/sunddi.h  
 1468 file path=usr/include/sys/sunldi.h  
 1469 file path=usr/include/sys/sunldi\_impl.h  
 1470 file path=usr/include/sys/sunmdi.h  
 1471 file path=usr/include/sys/sunndi.h  
 1472 file path=usr/include/sys/sunpm.h  
 1473 file path=usr/include/sys/suntpi.h  
 1474 file path=usr/include/sys/suntty.h  
 1475 file path=usr/include/sys/swap.h  
 1476 file path=usr/include/sys/synch.h  
 1477 file path=usr/include/sys/syscall.h  
 1478 file path=usr/include/sys/sysconf.h  
 1479 file path=usr/include/sys/sysconfig.h  
 1480 file path=usr/include/sys/sysconfig\_impl.h  
 1481 file path=usr/include/sys/sysdc.h  
 1482 file path=usr/include/sys/sysdc\_impl.h  
 1483 file path=usr/include/sys/ysevent.h  
 1484 file path=usr/include/sys/ysevent/ap\_driver.h  
 1485 file path=usr/include/sys/ysevent/dev.h  
 1486 file path=usr/include/sys/ysevent/domain.h  
 1487 file path=usr/include/sys/ysevent/dr.h  
 1488 file path=usr/include/sys/ysevent/env.h  
 1489 file path=usr/include/sys/ysevent/eventdefs.h  
 1490 file path=usr/include/sys/ysevent/imp.h  
 1491 file path=usr/include/sys/ysevent/pwrctl.h  
 1492 file path=usr/include/sys/ysevent/svm.h  
 1493 file path=usr/include/sys/ysevent/vrrp.h  
 1494 file path=usr/include/sys/ysevent\_impl.h  
 1495 \$(i386\_ONLY)file path=usr/include/sys/sysi86.h  
 1496 file path=usr/include/sys/sysinfo.h  
 1497 file path=usr/include/sys/syslog.h  
 1498 file path=usr/include/sys/systemacros.h  
 1499 file path=usr/include/sys/systeminfo.h  
 1500 file path=usr/include/sys/system.h  
 1501 file path=usr/include/sys/t\_kuser.h  
 1502 file path=usr/include/sys/t\_lock.h  
 1503 file path=usr/include/sys/task.h  
 1504 file path=usr/include/sys/taskq.h  
 1505 file path=usr/include/sys/taskq\_impl.h  
 1506 file path=usr/include/sys/telioctl.h  
 1507 file path=usr/include/sys/termio.h

1508 file path=usr/include/sys/termios.h  
 1509 file path=usr/include/sys/termiox.h  
 1510 file path=usr/include/sys/thread.h  
 1511 file path=usr/include/sys/ticlts.h  
 1512 file path=usr/include/sys/ticots.h  
 1513 file path=usr/include/sys/ticotsord.h  
 1514 file path=usr/include/sys/tihdr.h  
 1515 file path=usr/include/sys/time.h  
 1516 file path=usr/include/sys/time\_impl.h  
 1517 file path=usr/include/sys/time\_std\_impl.h  
 1518 file path=usr/include/sys/timeb.h  
 1519 file path=usr/include/sys/timer.h  
 1520 file path=usr/include/sys/times.h  
 1521 file path=usr/include/sys/timex.h  
 1522 file path=usr/include/sys/timod.h  
 1523 file path=usr/include/sys/tirdwr.h  
 1524 file path=usr/include/sys/tiuser.h  
 1525 file path=usr/include/sys/tl.h  
 1526 file path=usr/include/sys/tnf.h  
 1527 file path=usr/include/sys/tnf\_com.h  
 1528 file path=usr/include/sys/tnf\_probe.h  
 1529 file path=usr/include/sys/tnf\_writer.h  
 1530 file path=usr/include/sys/todio.h  
 1531 file path=usr/include/sys/tpicommon.h  
 1532 file path=usr/include/sys/trap.h  
 1533 \$(i386\_ONLY)file path=usr/include/sys/traptrace.h  
 1534 file path=usr/include/sys/ts.h  
 1535 file path=usr/include/sys/tsol/label.h  
 1536 file path=usr/include/sys/tsol/label\_macro.h  
 1537 file path=usr/include/sys/tsol/priv.h  
 1538 file path=usr/include/sys/tsol/tndb.h  
 1539 file path=usr/include/sys/tsol/tsyscall.h  
 1540 file path=usr/include/sys/tsprioctl.h  
 1541 \$(i386\_ONLY)file path=usr/include/sys/tss.h  
 1542 file path=usr/include/sys/ttcompat.h  
 1543 file path=usr/include/sys/ttold.h  
 1544 file path=usr/include/sys/tty.h  
 1545 file path=usr/include/sys/ttychars.h  
 1546 file path=usr/include/sys/ttydev.h  
 1547 \$(sparc\_ONLY)file path=usr/include/sys/ttymux.h  
 1548 \$(sparc\_ONLY)file path=usr/include/sys/ttymuxuser.h  
 1549 file path=usr/include/sys/tuneable.h  
 1550 file path=usr/include/sys/turnstile.h  
 1551 file path=usr/include/sys/types.h  
 1552 file path=usr/include/sys/types32.h  
 1553 file path=usr/include/sys/tzfile.h  
 1554 file path=usr/include/sys/u8\_textprep.h  
 1555 file path=usr/include/sys/uadmin.h  
 1556 \$(i386\_ONLY)file path=usr/include/sys/ucode.h  
 1557 file path=usr/include/sys/ucontext.h  
 1558 file path=usr/include/sys/uiio.h  
 1559 file path=usr/include/sys/ulimit.h  
 1560 file path=usr/include/sys/un.h  
 1561 file path=usr/include/sys/unistd.h  
 1562 file path=usr/include/sys/user.h  
 1563 file path=usr/include/sys/ustat.h  
 1564 file path=usr/include/sys/utime.h  
 1565 file path=usr/include/sys/utrap.h  
 1566 file path=usr/include/sys/utsname.h  
 1567 file path=usr/include/sys/utssys.h  
 1568 file path=usr/include/sys/uuid.h  
 1569 file path=usr/include/sys/va\_impl.h  
 1570 file path=usr/include/sys/va\_list.h  
 1571 file path=usr/include/sys/var.h  
 1572 file path=usr/include/sys/varargs.h  
 1573 file path=usr/include/sys/vfs.h

```

1574 file path=usr/include/sys/vfs_opreg.h
1575 file path=usr/include/sys/vfstab.h
1576 file path=usr/include/sys/videodev2.h
1577 file path=usr/include/sys/visual_io.h
1578 file path=usr/include/sys/vm.h
1579 file path=usr/include/sys/vm_usage.h
1580 file path=usr/include/sys/vmem.h
1581 file path=usr/include/sys/vmem_impl.h
1582 file path=usr/include/sys/vmem_impl_user.h
1583 file path=usr/include/sys/vmparam.h
1584 file path=usr/include/sys/vmsystem.h
1585 file path=usr/include/sys/vnode.h
1586 file path=usr/include/sys/vt.h
1587 file path=usr/include/sys/vtdaemon.h
1588 file path=usr/include/sys/vtoc.h
1589 file path=usr/include/sys/vtrace.h
1590 file path=usr/include/sys/vuid_event.h
1591 file path=usr/include/sys/vuid_queue.h
1592 file path=usr/include/sys/vuid_state.h
1593 file path=usr/include/sys/vuid_store.h
1594 file path=usr/include/sys/vuid_wheel.h
1595 file path=usr/include/sys/wait.h
1596 file path=usr/include/sys/waitq.h
1597 file path=usr/include/sys/watchpoint.h
1598 $(i386_ONLY)file path=usr/include/sys/x86_archext.h
1599 $(i386_ONLY)file path=usr/include/sys/xen_errno.h
1600 file path=usr/include/sys/xti_inet.h
1601 file path=usr/include/sys/xti_osi.h
1602 file path=usr/include/sys/xti_xtiopt.h
1603 file path=usr/include/sys/zcons.h
1604 file path=usr/include/sys/zmod.h
1605 file path=usr/include/sys/zone.h
1606 $(sparc_ONLY)file path=usr/include/sys/zsdev.h
1607 file path=usr/include/syssexits.h
1608 file path=usr/include/syslog.h
1609 file path=usr/include/tar.h
1610 file path=usr/include/tcpd.h
1611 file path=usr/include/term.h
1612 file path=usr/include/termcap.h
1613 file path=usr/include/termio.h
1614 file path=usr/include/termios.h
1615 file path=usr/include/thread.h
1616 file path=usr/include/thread_db.h
1617 file path=usr/include/time.h
1618 file path=usr/include/tiuser.h
1619 file path=usr/include/tsol/label.h
1620 file path=usr/include/tzfile.h
1621 file path=usr/include/ucontext.h
1622 file path=usr/include/ucred.h
1623 file path=usr/include/uid_stp.h
1624 file path=usr/include/ulimit.h
1625 file path=usr/include/umem.h
1626 file path=usr/include/umem_impl.h
1627 file path=usr/include/unctrl.h
1628 file path=usr/include/unistd.h
1629 file path=usr/include/user_attr.h
1630 file path=usr/include/userdefs.h
1631 file path=usr/include/ustat.h
1632 file path=usr/include/utility.h
1633 file path=usr/include/utime.h
1634 file path=usr/include/utmp.h
1635 file path=usr/include/utmpx.h
1636 file path=usr/include/uuid/uuid.h
1637 $(sparc_ONLY)file path=usr/include/v7/sys/machpcb.h
1638 $(sparc_ONLY)file path=usr/include/v7/sys/machtrap.h
1639 $(sparc_ONLY)file path=usr/include/v7/sys/mutex_impl.h

```

```

1640 $(sparc_ONLY)file path=usr/include/v7/sys/privregs.h
1641 $(sparc_ONLY)file path=usr/include/v7/sys/prom_isa.h
1642 $(sparc_ONLY)file path=usr/include/v7/sys/psr.h
1643 $(sparc_ONLY)file path=usr/include/v7/sys/traptrace.h
1644 $(sparc_ONLY)file path=usr/include/v9/sys/asi.h
1645 $(sparc_ONLY)file path=usr/include/v9/sys/machpcb.h
1646 $(sparc_ONLY)file path=usr/include/v9/sys/machtrap.h
1647 $(sparc_ONLY)file path=usr/include/v9/sys/membar.h
1648 $(sparc_ONLY)file path=usr/include/v9/sys/mutex_impl.h
1649 $(sparc_ONLY)file path=usr/include/v9/sys/privregs.h
1650 $(sparc_ONLY)file path=usr/include/v9/sys/prom_isa.h
1651 $(sparc_ONLY)file path=usr/include/v9/sys/psr_compat.h
1652 $(sparc_ONLY)file path=usr/include/v9/sys/vis_simulator.h
1653 file path=usr/include/valtools.h
1654 file path=usr/include/values.h
1655 file path=usr/include/varargs.h
1656 file path=usr/include/vm/anon.h
1657 file path=usr/include/vm/as.h
1658 file path=usr/include/vm/faultcode.h
1659 file path=usr/include/vm/hat.h
1660 file path=usr/include/vm/kpm.h
1661 file path=usr/include/vm/page.h
1662 file path=usr/include/vm/pvn.h
1663 file path=usr/include/vm/rm.h
1664 file path=usr/include/vm/seg.h
1665 file path=usr/include/vm/seg_dev.h
1666 file path=usr/include/vm/seg_enum.h
1667 file path=usr/include/vm/seg_kmem.h
1668 file path=usr/include/vm/seg_kp.h
1669 file path=usr/include/vm/seg_kpm.h
1670 file path=usr/include/vm/seg_map.h
1671 file path=usr/include/vm/seg_spt.h
1672 file path=usr/include/vm/seg_vn.h
1673 file path=usr/include/vm/vpage.h
1674 file path=usr/include/vm/vpm.h
1675 file path=usr/include/volmgt.h
1676 file path=usr/include/wait.h
1677 file path=usr/include/wchar.h
1678 file path=usr/include/wchar_impl.h
1679 file path=usr/include/wctype.h
1680 file path=usr/include/widec.h
1681 file path=usr/include/wordexp.h
1682 file path=usr/include/xlocale.h
1683 file path=usr/include/xti.h
1684 file path=usr/include/xti_inet.h
1685 file path=usr/include/zone.h
1686 file path=usr/include/zonestat.h
1687 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/acpidev.h
1688 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/amd_iommu.h
1689 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/asm_misc.h
1690 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/clock.h
1691 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/cram.h
1692 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/ddi_subrdefs.h
1693 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/debug_info.h
1694 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/fastboot.h
1695 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/mach_mmu.h
1696 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machclock.h
1697 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machcpuvar.h
1698 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machparam.h
1699 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machprivregs.h
1700 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machsystem.h
1701 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machthread.h
1702 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/memmode.h
1703 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/pc_mmu.h
1704 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/psm.h
1705 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/psm_defs.h

```

```

1706 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/psm_modctl.h
1707 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/psm_types.h
1708 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/rm_platter.h
1709 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/sbd_ioctl.h
1710 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/smp_impldefs.h
1711 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/vm_machparam.h
1712 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/x_call.h
1713 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/xc_levels.h
1714 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/xsvc.h
1715 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/hat_i86.h
1716 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/hat_pte.h
1717 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/hment.h
1718 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/htable.h
1719 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/kboot_mmu.h
1720 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/balloon.h
1721 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/machprivregs.h
1722 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/xen_mmu.h
1723 $(i386_ONLY)file path=usr/platform/i86pc/include/sys/xpv_impl.h
1724 $(i386_ONLY)file path=usr/platform/i86pc/include/vm/seg_mf.h
1725 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/ac.h
1726 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/asynch.h
1727 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cheataregs.h
1728 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cherrystone.h
1729 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/clock.h
1730 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cmp.h
1731 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cpc_ultra.h
1732 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cpr_impl.h
1733 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cpu_impl.h
1734 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cpu_sgnblk_defs.h
1735 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cvt.h
1736 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/daktari.h
1737 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/ddi_subrdefs.h
1738 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/dvma.h
1739 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/ecc_kstat.h
1740 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/evprom.h
1741 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/envctrl.h
1742 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/envctrl_gen.h
1743 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/envctrl_ue250.h
1744 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/envctrl_ue450.h
1745 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/environ.h
1746 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/errclassify.h
1747 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/fhc.h
1748 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/gpio_87317.h
1749 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/hpc3130_events.h
1750 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/hpc3130.h
1751 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/i2c_client.h
1752 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/lm75.h
1753 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/max1617.h
1754 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/pcf8591.h
1755 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c_clients/ssc050.h
1756 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/i2c/misc/i2c_svc.h
1757 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/idprom.h
1758 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/intr.h
1759 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/intreg.h
1760 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/iocache.h
1761 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/iommu.h
1762 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/ivintr.h
1763 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/lom_io.h
1764 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machasi.h
1765 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machclock.h
1766 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machcpuvar.h
1767 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machparam.h
1768 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machsystem.h
1769 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/machthread.h
1770 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/mem_cache.h
1771 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/memlist_plat.h

```

```

1772 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/memnode.h
1773 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/mmu.h
1774 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/nexusdebug.h
1775 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/opl_hwdesc.h
1776 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/opl_module.h
1777 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/prom_debug.h
1778 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/prom_plat.h
1779 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/pte.h
1780 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sbd_ioctl.h
1781 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/scb.h
1782 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/scsb_led.h
1783 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/simmstat.h
1784 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/spitregs.h
1785 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sram.h
1786 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/starfire.h
1787 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sun4asi.h
1788 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sysctrl.h
1789 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sysioerr.h
1790 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/sysiosbus.h
1791 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/tod.h
1792 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/todmostek.h
1793 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/trapstat.h
1794 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/traptrace.h
1795 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/vis.h
1796 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/vm_machparam.h
1797 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/x_call.h
1798 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/xc_impl.h
1799 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/xsmach.h
1800 $(sparc_ONLY)file path=usr/platform/sun4u/include/vm/hat_sfmmu.h
1801 $(sparc_ONLY)file path=usr/platform/sun4u/include/vm/mach_sfmmu.h
1802 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/clock.h
1803 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/cmp.h
1804 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/cpc_ultra.h
1805 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/cpu_sgnblk_defs.h
1806 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/ddi_subrdefs.h
1807 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/ds_pri.h
1808 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/ds_snmp.h
1809 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/dvma.h
1810 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/evprom.h
1811 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/fcode.h
1812 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/hsvc.h
1813 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/hypervisor_api.h
1814 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/idprom.h
1815 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/intr.h
1816 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/intreg.h
1817 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/ivintr.h
1818 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machasi.h
1819 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machclock.h
1820 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machcpuvar.h
1821 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machintreg.h
1822 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machparam.h
1823 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machsystem.h
1824 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/machthread.h
1825 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/memlist_plat.h
1826 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/memnode.h
1827 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/mmu.h
1828 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/nexusdebug.h
1829 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/niagaraasi.h
1830 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/niagararegs.h
1831 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/niawdt.h
1832 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/pri.h
1833 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/prom_debug.h
1834 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/prom_plat.h
1835 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/pte.h
1836 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/qcn.h
1837 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/scb.h

```

```

1838 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/soft_state.h
1839 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/sun4asi.h
1840 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/tod.h
1841 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/trapstat.h
1842 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/traptrace.h
1843 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/vis.h
1844 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/vm_machparam.h
1845 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/x_call.h
1846 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/xc_impl.h
1847 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/zsmach.h
1848 $(sparc_ONLY)file path=usr/platform/sun4v/include/vm/hat_sfmmu.h
1849 $(sparc_ONLY)file path=usr/platform/sun4v/include/vm/mach_sfmmu.h
1850 file path=usr/share/man/man3head/acct.h.3head
1851 file path=usr/share/man/man3head/aio.h.3head
1852 file path=usr/share/man/man3head/ar.h.3head
1853 file path=usr/share/man/man3head/archives.h.3head
1854 file path=usr/share/man/man3head/assert.h.3head
1855 file path=usr/share/man/man3head/complex.h.3head
1856 file path=usr/share/man/man3head/cpio.h.3head
1857 file path=usr/share/man/man3head/dirent.h.3head
1858 file path=usr/share/man/man3head/errno.h.3head
1859 file path=usr/share/man/man3head/fcntl.h.3head
1860 file path=usr/share/man/man3head/fenv.h.3head
1861 file path=usr/share/man/man3head/float.h.3head
1862 file path=usr/share/man/man3head/floatingpoint.h.3head
1863 file path=usr/share/man/man3head/fmtmsg.h.3head
1864 file path=usr/share/man/man3head/fnmatch.h.3head
1865 file path=usr/share/man/man3head/ftw.h.3head
1866 file path=usr/share/man/man3head/glob.h.3head
1867 file path=usr/share/man/man3head/grp.h.3head
1868 file path=usr/share/man/man3head/iconv.h.3head
1869 file path=usr/share/man/man3head/if.h.3head
1870 file path=usr/share/man/man3head/in.h.3head
1871 file path=usr/share/man/man3head/inet.h.3head
1872 file path=usr/share/man/man3head/inttypes.h.3head
1873 file path=usr/share/man/man3head/ipc.h.3head
1874 file path=usr/share/man/man3head/iso646.h.3head
1875 file path=usr/share/man/man3head/langinfo.h.3head
1876 file path=usr/share/man/man3head/libgen.h.3head
1877 file path=usr/share/man/man3head/libintl.h.3head
1878 file path=usr/share/man/man3head/limits.h.3head
1879 file path=usr/share/man/man3head/locale.h.3head
1880 file path=usr/share/man/man3head/math.h.3head
1881 file path=usr/share/man/man3head/mman.h.3head
1882 file path=usr/share/man/man3head/monetary.h.3head
1883 file path=usr/share/man/man3head/mqueue.h.3head
1884 file path=usr/share/man/man3head/msg.h.3head
1885 file path=usr/share/man/man3head/ndbm.h.3head
1886 file path=usr/share/man/man3head/netdb.h.3head
1887 file path=usr/share/man/man3head/nl_types.h.3head
1888 file path=usr/share/man/man3head/poll.h.3head
1889 file path=usr/share/man/man3head/pthread.h.3head
1890 file path=usr/share/man/man3head/pwd.h.3head
1891 file path=usr/share/man/man3head/regex.h.3head
1892 file path=usr/share/man/man3head/resource.h.3head
1893 file path=usr/share/man/man3head/sched.h.3head
1894 file path=usr/share/man/man3head/search.h.3head
1895 file path=usr/share/man/man3head/select.h.3head
1896 file path=usr/share/man/man3head/sem.h.3head
1897 file path=usr/share/man/man3head/semaphore.h.3head
1898 file path=usr/share/man/man3head/setjmp.h.3head
1899 file path=usr/share/man/man3head/shm.h.3head
1900 file path=usr/share/man/man3head/signinfo.h.3head
1901 file path=usr/share/man/man3head/signal.h.3head
1902 file path=usr/share/man/man3head/socket.h.3head
1903 file path=usr/share/man/man3head/spawn.h.3head

```

```

1904 file path=usr/share/man/man3head/stat.h.3head
1905 file path=usr/share/man/man3head/statvfs.h.3head
1906 file path=usr/share/man/man3head/stdbool.h.3head
1907 file path=usr/share/man/man3head/stddef.h.3head
1908 file path=usr/share/man/man3head/stdint.h.3head
1909 file path=usr/share/man/man3head/stdio.h.3head
1910 file path=usr/share/man/man3head/stdlib.h.3head
1911 file path=usr/share/man/man3head/string.h.3head
1912 file path=usr/share/man/man3head/strings.h.3head
1913 file path=usr/share/man/man3head/stropts.h.3head
1914 file path=usr/share/man/man3head/syslog.h.3head
1915 file path=usr/share/man/man3head/tar.h.3head
1916 file path=usr/share/man/man3head/tcp.h.3head
1917 file path=usr/share/man/man3head/termios.h.3head
1918 file path=usr/share/man/man3head/tgmath.h.3head
1919 file path=usr/share/man/man3head/time.h.3head
1920 file path=usr/share/man/man3head/timeb.h.3head
1921 file path=usr/share/man/man3head/times.h.3head
1922 file path=usr/share/man/man3head/types.h.3head
1923 file path=usr/share/man/man3head/types32.h.3head
1924 file path=usr/share/man/man3head/ucontext.h.3head
1925 file path=usr/share/man/man3head/uio.h.3head
1926 file path=usr/share/man/man3head/ulimit.h.3head
1927 file path=usr/share/man/man3head/un.h.3head
1928 file path=usr/share/man/man3head/unistd.h.3head
1929 file path=usr/share/man/man3head/utime.h.3head
1930 file path=usr/share/man/man3head/utmpx.h.3head
1931 file path=usr/share/man/man3head/utshame.h.3head
1932 file path=usr/share/man/man3head/values.h.3head
1933 file path=usr/share/man/man3head/wait.h.3head
1934 file path=usr/share/man/man3head/wchar.h.3head
1935 file path=usr/share/man/man3head/wctype.h.3head
1936 file path=usr/share/man/man3head/wordexp.h.3head
1937 file path=usr/share/man/man3head/xlocale.h.3head
1938 file path=usr/share/man/man4/note.4
1939 file path=usr/share/man/man5/prof.5
1940 file path=usr/share/man/man7i/cdio.7i
1941 file path=usr/share/man/man7i/dkio.7i
1942 file path=usr/share/man/man7i/fbio.7i
1943 file path=usr/share/man/man7i/fdio.7i
1944 file path=usr/share/man/man7i/hdio.7i
1945 file path=usr/share/man/man7i/iec61883.7i
1946 file path=usr/share/man/man7i/mhd.7i
1947 file path=usr/share/man/man7i/mtio.7i
1948 file path=usr/share/man/man7i/prnio.7i
1949 file path=usr/share/man/man7i/quotactl.7i
1950 file path=usr/share/man/man7i/sesio.7i
1951 file path=usr/share/man/man7i/sockio.7i
1952 file path=usr/share/man/man7i/streamio.7i
1953 file path=usr/share/man/man7i/termio.7i
1954 file path=usr/share/man/man7i/termiox.7i
1955 file path=usr/share/man/man7i/uscsci.7i
1956 file path=usr/share/man/man7i/visual_io.7i
1957 file path=usr/share/man/man7i/vt.7i
1958 file path=usr/xpg4/include/curses.h
1959 file path=usr/xpg4/include/term.h
1960 file path=usr/xpg4/include/unctrl.h
1961 legacy pkg=SUNWhea \
1962 desc="SunOS C/C++ header files for general development of software" \
1963 name="SunOS Header Files"
1964 license cr_Sun license=cr_Sun
1965 license lic_CDDL license=lic_CDDL
1966 license license_in_headers license=license_in_headers
1967 license usr/src/lib/pkcs11/include/THIRDPARTYLICENSE \
1968 license=usr/src/lib/pkcs11/include/THIRDPARTYLICENSE
1969 link path=usr/include/iso/assert_iso.h target=../assert.h

```

```

1970 link path=usr/include/iso/errno_iso.h target=../errno.h
1971 link path=usr/include/iso/float_iso.h target=../float.h
1972 link path=usr/include/iso/iso646_iso.h target=../iso646.h
1973 $(sparc_ONLY)link path=usr/platform/SUNW,A70/include target=../sun4u/include
1974 $(sparc_ONLY)link path=usr/platform/SUNW,Netra-T12/include \
1975     target=../sun4u/include
1976 $(sparc_ONLY)link path=usr/platform/SUNW,Netra-T4/include \
1977     target=../sun4u/include
1978 $(sparc_ONLY)link path=usr/platform/SUNW,SPARC-Enterprise/include \
1979     target=../sun4u/include
1980 $(sparc_ONLY)link path=usr/platform/SUNW,Serverblad1/include \
1981     target=../sun4u/include
1982 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-100/include \
1983     target=../sun4u/include
1984 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-1000/include \
1985     target=../sun4u/include
1986 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-1500/include \
1987     target=../sun4u/include
1988 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-2500/include \
1989     target=../sun4u/include
1990 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-15000/include \
1991     target=../sun4u/include
1992 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-280R/include \
1993     target=../sun4u/include
1994 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-480R/include \
1995     target=../sun4u/include
1996 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-880/include \
1997     target=../sun4u/include
1998 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V215/include \
1999     target=../sun4u/include
2000 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V240/include \
2001     target=../sun4u/include
2002 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V250/include \
2003     target=../sun4u/include
2004 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V440/include \
2005     target=../sun4u/include
2006 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V445/include \
2007     target=../sun4u/include
2008 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V490/include \
2009     target=../sun4u/include
2010 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V890/include \
2011     target=../sun4u/include
2012 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire/include \
2013     target=../sun4u/include
2014 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-2/include \
2015     target=../sun4u/include
2016 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-250/include \
2017     target=../sun4u/include
2018 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-4/include \
2019     target=../sun4u/include
2020 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-Enterprise-10000/include \
2021     target=../sun4u/include
2022 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-Enterprise/include \
2023     target=../sun4u/include
2024 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-40/include \
2025     target=../sun4u/include
2026 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-60/include \
2027     target=../sun4u/include
2028 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-IIi-Netract/include \
2029     target=../sun4u/include
2030 link path=usr/share/man/man3head/acct.3head target=acct.h.3head
2031 link path=usr/share/man/man3head/aio.3head target=aio.h.3head
2032 link path=usr/share/man/man3head/ar.3head target=ar.h.3head
2033 link path=usr/share/man/man3head/archives.3head target=archives.h.3head
2034 link path=usr/share/man/man3head/assert.3head target=assert.h.3head
2035 link path=usr/share/man/man3head/complex.3head target=complex.h.3head

```

```

2036 link path=usr/share/man/man3head/cpio.3head target=cpio.h.3head
2037 link path=usr/share/man/man3head/dirent.3head target=dirent.h.3head
2038 link path=usr/share/man/man3head/errno.3head target=errno.h.3head
2039 link path=usr/share/man/man3head/fcntl.3head target=fcntl.h.3head
2040 link path=usr/share/man/man3head/fenv.3head target=fenv.h.3head
2041 link path=usr/share/man/man3head/float.3head target=float.h.3head
2042 link path=usr/share/man/man3head/floatingpoint.3head \
2043     target=floatingpoint.h.3head
2044 link path=usr/share/man/man3head/fmtmsg.3head target=fmtmsg.h.3head
2045 link path=usr/share/man/man3head/fnmatch.3head target=fnmatch.h.3head
2046 link path=usr/share/man/man3head/ftw.3head target=ftw.h.3head
2047 link path=usr/share/man/man3head/glob.3head target=glob.h.3head
2048 link path=usr/share/man/man3head/grp.3head target=grp.h.3head
2049 link path=usr/share/man/man3head/iconv.3head target=iconv.h.3head
2050 link path=usr/share/man/man3head/if.3head target=if.h.3head
2051 link path=usr/share/man/man3head/in.3head target=in.h.3head
2052 link path=usr/share/man/man3head/inet.3head target=inet.h.3head
2053 link path=usr/share/man/man3head/inttypes.3head target=inttypes.h.3head
2054 link path=usr/share/man/man3head/ipc.3head target=ipc.h.3head
2055 link path=usr/share/man/man3head/iso646.3head target=iso646.h.3head
2056 link path=usr/share/man/man3head/langinfo.3head target=langinfo.h.3head
2057 link path=usr/share/man/man3head/libgen.3head target=libgen.h.3head
2058 link path=usr/share/man/man3head/libintl.3head target=libintl.h.3head
2059 link path=usr/share/man/man3head/limits.3head target=limits.h.3head
2060 link path=usr/share/man/man3head/locale.3head target=locale.h.3head
2061 link path=usr/share/man/man3head/math.3head target=math.h.3head
2062 link path=usr/share/man/man3head/mman.3head target=mman.h.3head
2063 link path=usr/share/man/man3head/monetary.3head target=monetary.h.3head
2064 link path=usr/share/man/man3head/mqueue.3head target=mqueue.h.3head
2065 link path=usr/share/man/man3head/msg.3head target=msg.h.3head
2066 link path=usr/share/man/man3head/ndbm.3head target=ndbm.h.3head
2067 link path=usr/share/man/man3head/netdb.3head target=netdb.h.3head
2068 link path=usr/share/man/man3head/nl_types.3head target=nl_types.h.3head
2069 link path=usr/share/man/man3head/poll.3head target=poll.h.3head
2070 link path=usr/share/man/man3head/pthread.3head target=pthread.h.3head
2071 link path=usr/share/man/man3head/pwd.3head target=pwd.h.3head
2072 link path=usr/share/man/man3head/regex.3head target=regex.h.3head
2073 link path=usr/share/man/man3head/resource.3head target=resource.h.3head
2074 link path=usr/share/man/man3head/sched.3head target=sched.h.3head
2075 link path=usr/share/man/man3head/search.3head target=search.h.3head
2076 link path=usr/share/man/man3head/select.3head target=select.h.3head
2077 link path=usr/share/man/man3head/sem.3head target=sem.h.3head
2078 link path=usr/share/man/man3head/semaphore.3head target=semaphore.h.3head
2079 link path=usr/share/man/man3head/setjmp.3head target=setjmp.h.3head
2080 link path=usr/share/man/man3head/shm.3head target=shm.h.3head
2081 link path=usr/share/man/man3head/signinfo.3head target=signinfo.h.3head
2082 link path=usr/share/man/man3head/signal.3head target=signal.h.3head
2083 link path=usr/share/man/man3head/socket.3head target=socket.h.3head
2084 link path=usr/share/man/man3head/spawn.3head target=spawn.h.3head
2085 link path=usr/share/man/man3head/stat.3head target=stat.h.3head
2086 link path=usr/share/man/man3head/statvfs.3head target=statvfs.h.3head
2087 link path=usr/share/man/man3head/stdbool.3head target=stdbool.h.3head
2088 link path=usr/share/man/man3head/stddef.3head target=stddef.h.3head
2089 link path=usr/share/man/man3head/stdint.3head target=stdint.h.3head
2090 link path=usr/share/man/man3head/stdio.3head target=stdio.h.3head
2091 link path=usr/share/man/man3head/stdlib.3head target=stdlib.h.3head
2092 link path=usr/share/man/man3head/string.3head target=string.h.3head
2093 link path=usr/share/man/man3head/strings.3head target=strings.h.3head
2094 link path=usr/share/man/man3head/stropts.3head target=stropts.h.3head
2095 link path=usr/share/man/man3head/syslog.3head target=syslog.h.3head
2096 link path=usr/share/man/man3head/tar.3head target=tar.h.3head
2097 link path=usr/share/man/man3head/tcp.3head target=tc.h.3head
2098 link path=usr/share/man/man3head/termios.3head target=termios.h.3head
2099 link path=usr/share/man/man3head/tgmth.3head target=tgmth.h.3head
2100 link path=usr/share/man/man3head/time.3head target=time.h.3head
2101 link path=usr/share/man/man3head/timeb.3head target=timeb.h.3head

```



```
2102 link path=usr/share/man/man3head/times.3head target=times.h.3head
2103 link path=usr/share/man/man3head/types.3head target=types.h.3head
2104 link path=usr/share/man/man3head/types32.3head target=types32.h.3head
2105 link path=usr/share/man/man3head/ucontext.3head target=ucontext.h.3head
2106 link path=usr/share/man/man3head/uio.3head target=uio.h.3head
2107 link path=usr/share/man/man3head/ulimit.3head target=ulimit.h.3head
2108 link path=usr/share/man/man3head/un.3head target=un.h.3head
2109 link path=usr/share/man/man3head/unistd.3head target=unistd.h.3head
2110 link path=usr/share/man/man3head/utime.3head target=utime.h.3head
2111 link path=usr/share/man/man3head/utmpx.3head target=utmpx.h.3head
2112 link path=usr/share/man/man3head/utsname.3head target=utsname.h.3head
2113 link path=usr/share/man/man3head/values.3head target=values.h.3head
2114 link path=usr/share/man/man3head/wait.3head target=wait.h.3head
2115 link path=usr/share/man/man3head/wchar.3head target=wchar.h.3head
2116 link path=usr/share/man/man3head/wctype.3head target=wctype.h.3head
2117 link path=usr/share/man/man3head/wordexp.3head target=wordexp.h.3head
2118 link path=usr/share/man/man3head/xlocale.3head target=xlocale.h.3head
2119 $(i386_ONLY)link path=usr/share/src/uts/i86pc/sys \
2120     target=../../../../platform/i86pc/include/sys
2121 $(i386_ONLY)link path=usr/share/src/uts/i86pc/vm \
2122     target=../../../../platform/i86pc/include/vm
2123 $(i386_ONLY)link path=usr/share/src/uts/i86xpv/sys \
2124     target=../../../../platform/i86xpv/include/sys
2125 $(i386_ONLY)link path=usr/share/src/uts/i86xpv/vm \
2126     target=../../../../platform/i86xpv/include/vm
2127 $(sparc_ONLY)link path=usr/share/src/uts/sun4u/sys \
2128     target=../../../../platform/sun4u/include/sys
2129 $(sparc_ONLY)link path=usr/share/src/uts/sun4u/vm \
2130     target=../../../../platform/sun4u/include/vm
2131 $(sparc_ONLY)link path=usr/share/src/uts/sun4v/sys \
2132     target=../../../../platform/sun4v/include/sys
2133 $(sparc_ONLY)link path=usr/share/src/uts/sun4v/vm \
2134     target=../../../../platform/sun4v/include/vm
```

new/usr/src/pkg/manifests/system-library.man3c.inc

1

\*\*\*\*\*

73499 Fri Jul 11 11:43:13 2014

new/usr/src/pkg/manifests/system-library.man3c.inc

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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
9 # at http://www.illumos.org/license/CDDL.
10 #
```

```
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
15 # Copyright 2013 OmniTI Computer Consulting, Inc. All rights reserved.
16 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
17 #
```

```
19 file path=usr/share/man/man3c/_fbufsize.3c
20 file path=usr/share/man/man3c/_longjmp.3c
21 file path=usr/share/man/man3c/_stack_grow.3c
22 file path=usr/share/man/man3c/a64l.3c
23 file path=usr/share/man/man3c/abort.3c
24 file path=usr/share/man/man3c/abs.3c
25 file path=usr/share/man/man3c/addsev.3c
26 file path=usr/share/man/man3c/addseverity.3c
27 file path=usr/share/man/man3c/aio_cancel.3c
28 file path=usr/share/man/man3c/aio_error.3c
29 file path=usr/share/man/man3c/aio_fsync.3c
30 file path=usr/share/man/man3c/aio_read.3c
31 file path=usr/share/man/man3c/aio_return.3c
32 file path=usr/share/man/man3c/aio_suspend.3c
33 file path=usr/share/man/man3c/aio_waitn.3c
34 file path=usr/share/man/man3c/aio_write.3c
35 file path=usr/share/man/man3c/aiocancel.3c
36 file path=usr/share/man/man3c/aioread.3c
37 file path=usr/share/man/man3c/aiowait.3c
38 file path=usr/share/man/man3c/assert.3c
39 file path=usr/share/man/man3c/atexit.3c
40 file path=usr/share/man/man3c/atomic_add.3c
41 file path=usr/share/man/man3c/atomic_and.3c
42 file path=usr/share/man/man3c/atomic_bits.3c
43 file path=usr/share/man/man3c/atomic_cas.3c
44 file path=usr/share/man/man3c/atomic_dec.3c
45 file path=usr/share/man/man3c/atomic_inc.3c
46 file path=usr/share/man/man3c/atomic_ops.3c
47 file path=usr/share/man/man3c/atomic_or.3c
48 file path=usr/share/man/man3c/atomic_swap.3c
49 file path=usr/share/man/man3c/attropen.3c
50 file path=usr/share/man/man3c/basename.3c
51 file path=usr/share/man/man3c/bsd_signal.3c
52 file path=usr/share/man/man3c/bsearch.3c
53 file path=usr/share/man/man3c/bstring.3c
54 file path=usr/share/man/man3c/btowc.3c
55 file path=usr/share/man/man3c/catgets.3c
56 file path=usr/share/man/man3c/catopen.3c
57 file path=usr/share/man/man3c/cfgetispeed.3c
58 file path=usr/share/man/man3c/cfsetispeed.3c
```

new/usr/src/pkg/manifests/system-library.man3c.inc

2

```
59 file path=usr/share/man/man3c/clock.3c
60 file path=usr/share/man/man3c/clock_nanosleep.3c
61 file path=usr/share/man/man3c/clock_gettime.3c
62 file path=usr/share/man/man3c/closedir.3c
63 file path=usr/share/man/man3c/closefrom.3c
64 file path=usr/share/man/man3c/cond_init.3c
65 file path=usr/share/man/man3c/confstr.3c
66 file path=usr/share/man/man3c/crypt.3c
67 file path=usr/share/man/man3c/crypt_genhash_impl.3c
68 file path=usr/share/man/man3c/crypt_gensalt.3c
69 file path=usr/share/man/man3c/crypt_gensalt_impl.3c
70 file path=usr/share/man/man3c/cset.3c
71 file path=usr/share/man/man3c/ctermid.3c
72 file path=usr/share/man/man3c/ctime.3c
73 file path=usr/share/man/man3c/ctype.3c
74 file path=usr/share/man/man3c/cuserid.3c
75 file path=usr/share/man/man3c/daemon.3c
76 file path=usr/share/man/man3c/decimal_to_floating.3c
77 file path=usr/share/man/man3c/difftime.3c
78 file path=usr/share/man/man3c/directio.3c
79 file path=usr/share/man/man3c/dirfd.3c
80 file path=usr/share/man/man3c/dirname.3c
81 file path=usr/share/man/man3c/div.3c
82 file path=usr/share/man/man3c/dladdr.3c
83 file path=usr/share/man/man3c/dlclose.3c
84 file path=usr/share/man/man3c/dldump.3c
85 file path=usr/share/man/man3c/dlerror.3c
86 file path=usr/share/man/man3c/dlinfo.3c
87 file path=usr/share/man/man3c/dlopen.3c
88 file path=usr/share/man/man3c/dlsym.3c
89 file path=usr/share/man/man3c/door_bind.3c
90 file path=usr/share/man/man3c/door_call.3c
91 file path=usr/share/man/man3c/door_create.3c
92 file path=usr/share/man/man3c/door_cred.3c
93 file path=usr/share/man/man3c/door_getparam.3c
94 file path=usr/share/man/man3c/door_info.3c
95 file path=usr/share/man/man3c/door_return.3c
96 file path=usr/share/man/man3c/door_revoke.3c
97 file path=usr/share/man/man3c/door_server_create.3c
98 file path=usr/share/man/man3c/door_ucred.3c
99 file path=usr/share/man/man3c/drand48.3c
100 file path=usr/share/man/man3c/dup2.3c
101 file path=usr/share/man/man3c/econvert.3c
102 file path=usr/share/man/man3c/ecvt.3c
103 file path=usr/share/man/man3c/enable_extended_FILE_stdio.3c
104 file path=usr/share/man/man3c/encrypt.3c
105 file path=usr/share/man/man3c/end.3c
106 file path=usr/share/man/man3c/err.3c
107 file path=usr/share/man/man3c/euclen.3c
108 file path=usr/share/man/man3c/exit.3c
109 file path=usr/share/man/man3c/fattach.3c
110 file path=usr/share/man/man3c/fclose.3c
111 file path=usr/share/man/man3c/fdatasync.3c
112 file path=usr/share/man/man3c/fdetach.3c
113 file path=usr/share/man/man3c/fdopen.3c
114 file path=usr/share/man/man3c/ferror.3c
115 file path=usr/share/man/man3c/fflush.3c
116 file path=usr/share/man/man3c/ffs.3c
117 file path=usr/share/man/man3c/fgetattr.3c
118 file path=usr/share/man/man3c/fgetc.3c
119 file path=usr/share/man/man3c/fgetpos.3c
120 file path=usr/share/man/man3c/fgetwc.3c
121 file path=usr/share/man/man3c/floating_to_decimal.3c
122 file path=usr/share/man/man3c/flockfile.3c
123 file path=usr/share/man/man3c/fmtmsg.3c
124 file path=usr/share/man/man3c/fnmatch.3c
```

```

125 file path=usr/share/man/man3c/fopen.3c
126 file path=usr/share/man/man3c/fpgetround.3c
127 file path=usr/share/man/man3c/fputc.3c
128 file path=usr/share/man/man3c/fputwc.3c
129 file path=usr/share/man/man3c/fputws.3c
130 file path=usr/share/man/man3c/fread.3c
131 file path=usr/share/man/man3c/freopen.3c
132 file path=usr/share/man/man3c/fseek.3c
133 file path=usr/share/man/man3c/fsetpos.3c
134 file path=usr/share/man/man3c/fsync.3c
135 file path=usr/share/man/man3c/ftell.3c
136 file path=usr/share/man/man3c/ftime.3c
137 file path=usr/share/man/man3c/ftok.3c
138 file path=usr/share/man/man3c/ftw.3c
139 file path=usr/share/man/man3c/fwide.3c
140 file path=usr/share/man/man3c/fwprintf.3c
141 file path=usr/share/man/man3c/fwrite.3c
142 file path=usr/share/man/man3c/fwscanf.3c
143 file path=usr/share/man/man3c/getcpuid.3c
144 file path=usr/share/man/man3c/getcwd.3c
145 file path=usr/share/man/man3c/getdate.3c
146 file path=usr/share/man/man3c/getdtablesize.3c
147 file path=usr/share/man/man3c/getenv.3c
148 file path=usr/share/man/man3c/getexecname.3c
149 file path=usr/share/man/man3c/getgrnam.3c
150 file path=usr/share/man/man3c/gethostid.3c
151 file path=usr/share/man/man3c/gethostname.3c
152 file path=usr/share/man/man3c/gethrtime.3c
153 file path=usr/share/man/man3c/getline.3c
154 file path=usr/share/man/man3c/getloadavg.3c
155 file path=usr/share/man/man3c/getlogin.3c
156 file path=usr/share/man/man3c/getmntent.3c
157 file path=usr/share/man/man3c/getnetgrent.3c
158 file path=usr/share/man/man3c/getopt.3c
159 file path=usr/share/man/man3c/getpagesize.3c
160 file path=usr/share/man/man3c/getpagesizes.3c
161 file path=usr/share/man/man3c/getpass.3c
162 file path=usr/share/man/man3c/getpeerucred.3c
163 file path=usr/share/man/man3c/getpriority.3c
164 file path=usr/share/man/man3c/getpw.3c
165 file path=usr/share/man/man3c/getpwnam.3c
166 file path=usr/share/man/man3c/getrusage.3c
167 file path=usr/share/man/man3c/gets.3c
168 file path=usr/share/man/man3c/getspnam.3c
169 file path=usr/share/man/man3c/getsubopt.3c
170 file path=usr/share/man/man3c/gettext.3c
171 file path=usr/share/man/man3c/gettimeofday.3c
172 file path=usr/share/man/man3c/gettxt.3c
173 file path=usr/share/man/man3c/getusershell.3c
174 file path=usr/share/man/man3c/getutent.3c
175 file path=usr/share/man/man3c/getutxent.3c
176 file path=usr/share/man/man3c/getvfsent.3c
177 file path=usr/share/man/man3c/getwc.3c
178 file path=usr/share/man/man3c/getwchar.3c
179 file path=usr/share/man/man3c/getwd.3c
180 file path=usr/share/man/man3c/getwidth.3c
181 file path=usr/share/man/man3c/getws.3c
182 file path=usr/share/man/man3c/getzoneid.3c
183 file path=usr/share/man/man3c/glob.3c
184 file path=usr/share/man/man3c/grantpt.3c
185 file path=usr/share/man/man3c/hsearch.3c
186 file path=usr/share/man/man3c/iconv.3c
187 file path=usr/share/man/man3c/iconv_close.3c
188 file path=usr/share/man/man3c/iconv_open.3c
189 file path=usr/share/man/man3c/imaxabs.3c
190 file path=usr/share/man/man3c/imaxdiv.3c

```

```

191 file path=usr/share/man/man3c/index.3c
192 file path=usr/share/man/man3c/initgroups.3c
193 file path=usr/share/man/man3c/insque.3c
194 file path=usr/share/man/man3c/is_system_labeled.3c
195 file path=usr/share/man/man3c/isaexec.3c
196 file path=usr/share/man/man3c/isastream.3c
197 file path=usr/share/man/man3c/isatty.3c
198 file path=usr/share/man/man3c/isnand.3c
199 file path=usr/share/man/man3c/iswalpha.3c
200 file path=usr/share/man/man3c/iswctype.3c
201 file path=usr/share/man/man3c/killpg.3c
202 file path=usr/share/man/man3c/lckpwwf.3c
203 file path=usr/share/man/man3c/lfmt.3c
204 file path=usr/share/man/man3c/ljo_listio.3c
205 file path=usr/share/man/man3c/localeconv.3c
206 file path=usr/share/man/man3c/lockf.3c
207 file path=usr/share/man/man3c/lsearch.3c
208 file path=usr/share/man/man3c/madvise.3c
209 file path=usr/share/man/man3c/makecontext.3c
210 file path=usr/share/man/man3c/makedev.3c
211 file path=usr/share/man/man3c/malloc.3c
212 file path=usr/share/man/man3c/mblen.3c
213 file path=usr/share/man/man3c/mbrlen.3c
214 file path=usr/share/man/man3c/mbrtowc.3c
215 file path=usr/share/man/man3c/mbsinit.3c
216 file path=usr/share/man/man3c/mbsrtowcs.3c
217 file path=usr/share/man/man3c/mbtowc.3c
218 file path=usr/share/man/man3c/membar_ops.3c
219 file path=usr/share/man/man3c/memory.3c
220 file path=usr/share/man/man3c/mkfifo.3c
221 file path=usr/share/man/man3c/mkstemp.3c
222 file path=usr/share/man/man3c/mktemp.3c
223 file path=usr/share/man/man3c/mktime.3c
224 file path=usr/share/man/man3c/mlock.3c
225 file path=usr/share/man/man3c/mlockall.3c
226 file path=usr/share/man/man3c/monitor.3c
227 file path=usr/share/man/man3c/mq_close.3c
228 file path=usr/share/man/man3c/mq_getattr.3c
229 file path=usr/share/man/man3c/mq_notify.3c
230 file path=usr/share/man/man3c/mq_open.3c
231 file path=usr/share/man/man3c/mq_receive.3c
232 file path=usr/share/man/man3c/mq_send.3c
233 file path=usr/share/man/man3c/mq_setattr.3c
234 file path=usr/share/man/man3c/mq_unlink.3c
235 file path=usr/share/man/man3c/msync.3c
236 file path=usr/share/man/man3c/mutex_init.3c
237 file path=usr/share/man/man3c/nanosleep.3c
238 file path=usr/share/man/man3c/ndbm.3c
239 file path=usr/share/man/man3c/newlocale.3c
240 file path=usr/share/man/man3c/nl_langinfo.3c
241 file path=usr/share/man/man3c/offsetof.3c
242 file path=usr/share/man/man3c/opendir.3c
243 file path=usr/share/man/man3c/perror.3c
244 file path=usr/share/man/man3c/pfmt.3c
245 file path=usr/share/man/man3c/plock.3c
246 file path=usr/share/man/man3c/popen.3c
247 file path=usr/share/man/man3c/port_alert.3c
248 file path=usr/share/man/man3c/port_associate.3c
249 file path=usr/share/man/man3c/port_create.3c
250 file path=usr/share/man/man3c/port_get.3c
251 file path=usr/share/man/man3c/port_send.3c
252 file path=usr/share/man/man3c/posix_fadvise.3c
253 file path=usr/share/man/man3c/posix_fallocate.3c
254 file path=usr/share/man/man3c/posix_madvise.3c
255 file path=usr/share/man/man3c/posix_memalign.3c
256 file path=usr/share/man/man3c/posix_openpt.3c

```

```

257 file path=usr/share/man/man3c/posix_spawn.3c
258 file path=usr/share/man/man3c/posix_spawn_file_actions_addclose.3c
259 file path=usr/share/man/man3c/posix_spawn_file_actions_addclosefrom_np.3c
260 file path=usr/share/man/man3c/posix_spawn_file_actions_adddup2.3c
261 file path=usr/share/man/man3c/posix_spawn_file_actions_destroy.3c
262 file path=usr/share/man/man3c/posix_spawn_pipe_np.3c
263 file path=usr/share/man/man3c/posix_spawnattr_destroy.3c
264 file path=usr/share/man/man3c/posix_spawnattr_getflags.3c
265 file path=usr/share/man/man3c/posix_spawnattr_getpgroup.3c
266 file path=usr/share/man/man3c/posix_spawnattr_getschedparam.3c
267 file path=usr/share/man/man3c/posix_spawnattr_getschedpolicy.3c
268 file path=usr/share/man/man3c/posix_spawnattr_getsigdefault.3c
269 file path=usr/share/man/man3c/posix_spawnattr_getsigignore_np.3c
270 file path=usr/share/man/man3c/posix_spawnattr_getsigmask.3c
271 file path=usr/share/man/man3c/printf.3c
272 file path=usr/share/man/man3c/priv_addset.3c
273 file path=usr/share/man/man3c/priv_set.3c
274 file path=usr/share/man/man3c/priv_str_to_set.3c
275 file path=usr/share/man/man3c/pset_getloadavg.3c
276 file path=usr/share/man/man3c/psignal.3c
277 file path=usr/share/man/man3c/pthread_atfork.3c
278 file path=usr/share/man/man3c/pthread_attr_getdetachstate.3c
279 file path=usr/share/man/man3c/pthread_attr_getguardsize.3c
280 file path=usr/share/man/man3c/pthread_attr_getinheritsched.3c
281 file path=usr/share/man/man3c/pthread_attr_getschedparam.3c
282 file path=usr/share/man/man3c/pthread_attr_getschedpolicy.3c
283 file path=usr/share/man/man3c/pthread_attr_getscope.3c
284 file path=usr/share/man/man3c/pthread_attr_getstack.3c
285 file path=usr/share/man/man3c/pthread_attr_getstackaddr.3c
286 file path=usr/share/man/man3c/pthread_attr_getstacksize.3c
287 file path=usr/share/man/man3c/pthread_attr_init.3c
288 file path=usr/share/man/man3c/pthread_barrier_destroy.3c
289 file path=usr/share/man/man3c/pthread_barrier_wait.3c
290 file path=usr/share/man/man3c/pthread_barrierattr_destroy.3c
291 file path=usr/share/man/man3c/pthread_barrierattr_getpshared.3c
292 file path=usr/share/man/man3c/pthread_cancel.3c
293 file path=usr/share/man/man3c/pthread_cleanup_pop.3c
294 file path=usr/share/man/man3c/pthread_cleanup_push.3c
295 file path=usr/share/man/man3c/pthread_cond_init.3c
296 file path=usr/share/man/man3c/pthread_cond_signal.3c
297 file path=usr/share/man/man3c/pthread_cond_wait.3c
298 file path=usr/share/man/man3c/pthread_condattr_getclock.3c
299 file path=usr/share/man/man3c/pthread_condattr_getpshared.3c
300 file path=usr/share/man/man3c/pthread_condattr_init.3c
301 file path=usr/share/man/man3c/pthread_create.3c
302 file path=usr/share/man/man3c/pthread_detach.3c
303 file path=usr/share/man/man3c/pthread_equal.3c
304 file path=usr/share/man/man3c/pthread_exit.3c
305 file path=usr/share/man/man3c/pthread_getconcurrency.3c
306 file path=usr/share/man/man3c/pthread_getschedparam.3c
307 file path=usr/share/man/man3c/pthread_getspecific.3c
308 file path=usr/share/man/man3c/pthread_join.3c
309 file path=usr/share/man/man3c/pthread_key_create.3c
310 file path=usr/share/man/man3c/pthread_key_delete.3c
311 file path=usr/share/man/man3c/pthread_kill.3c
312 file path=usr/share/man/man3c/pthread_mutex_getprioceiling.3c
313 file path=usr/share/man/man3c/pthread_mutex_init.3c
314 file path=usr/share/man/man3c/pthread_mutex_lock.3c
315 file path=usr/share/man/man3c/pthread_mutex_timedlock.3c
316 file path=usr/share/man/man3c/pthread_mutexattr_getprioceiling.3c
317 file path=usr/share/man/man3c/pthread_mutexattr_getprotocol.3c
318 file path=usr/share/man/man3c/pthread_mutexattr_getpshared.3c
319 file path=usr/share/man/man3c/pthread_mutexattr_gettype.3c
320 file path=usr/share/man/man3c/pthread_mutexattr_init.3c
321 file path=usr/share/man/man3c/pthread_once.3c
322 file path=usr/share/man/man3c/pthread_rwlock_init.3c

```

```

323 file path=usr/share/man/man3c/pthread_rwlock_rdlock.3c
324 file path=usr/share/man/man3c/pthread_rwlock_timedrdlock.3c
325 file path=usr/share/man/man3c/pthread_rwlock_timedwrlock.3c
326 file path=usr/share/man/man3c/pthread_rwlock_unlock.3c
327 file path=usr/share/man/man3c/pthread_rwlock_wrlock.3c
328 file path=usr/share/man/man3c/pthread_rwlockattr_getpshared.3c
329 file path=usr/share/man/man3c/pthread_rwlockattr_init.3c
330 file path=usr/share/man/man3c/pthread_self.3c
331 file path=usr/share/man/man3c/pthread_setcancelstate.3c
332 file path=usr/share/man/man3c/pthread_setcanceltype.3c
333 file path=usr/share/man/man3c/pthread_setschedprio.3c
334 file path=usr/share/man/man3c/pthread_sigmask.3c
335 file path=usr/share/man/man3c/pthread_spin_destroy.3c
336 file path=usr/share/man/man3c/pthread_spin_lock.3c
337 file path=usr/share/man/man3c/pthread_spin_unlock.3c
338 file path=usr/share/man/man3c/pthread_testcancel.3c
339 file path=usr/share/man/man3c/ptrace.3c
340 file path=usr/share/man/man3c/ptsname.3c
341 file path=usr/share/man/man3c/putenv.3c
342 file path=usr/share/man/man3c/putpwent.3c
343 file path=usr/share/man/man3c/puts.3c
344 file path=usr/share/man/man3c/putspent.3c
345 file path=usr/share/man/man3c/putws.3c
346 file path=usr/share/man/man3c/qsort.3c
347 file path=usr/share/man/man3c/raise.3c
348 file path=usr/share/man/man3c/rand.3c
349 file path=usr/share/man/man3c/random.3c
350 file path=usr/share/man/man3c/rctl_walk.3c
351 file path=usr/share/man/man3c/rctlblk_set_value.3c
352 file path=usr/share/man/man3c/re_comp.3c
353 file path=usr/share/man/man3c/readdir.3c
354 file path=usr/share/man/man3c/realpath.3c
355 file path=usr/share/man/man3c/reboot.3c
356 file path=usr/share/man/man3c/regcmp.3c
357 file path=usr/share/man/man3c/regcomp.3c
358 file path=usr/share/man/man3c/remove.3c
359 file path=usr/share/man/man3c/rewind.3c
360 file path=usr/share/man/man3c/rewinddir.3c
361 file path=usr/share/man/man3c/rwlock.3c
362 file path=usr/share/man/man3c/scandir.3c
363 file path=usr/share/man/man3c/scanf.3c
364 file path=usr/share/man/man3c/sched_get_priority_max.3c
365 file path=usr/share/man/man3c/sched_getparam.3c
366 file path=usr/share/man/man3c/sched_getscheduler.3c
367 file path=usr/share/man/man3c/sched_rr_get_interval.3c
368 file path=usr/share/man/man3c/sched_setparam.3c
369 file path=usr/share/man/man3c/sched_setscheduler.3c
370 file path=usr/share/man/man3c/sched_yield.3c
371 file path=usr/share/man/man3c/schedctl_init.3c
372 file path=usr/share/man/man3c/seekdir.3c
373 file path=usr/share/man/man3c/select.3c
374 file path=usr/share/man/man3c/sem_close.3c
375 file path=usr/share/man/man3c/sem_destroy.3c
376 file path=usr/share/man/man3c/sem_getvalue.3c
377 file path=usr/share/man/man3c/sem_init.3c
378 file path=usr/share/man/man3c/sem_open.3c
379 file path=usr/share/man/man3c/sem_post.3c
380 file path=usr/share/man/man3c/sem_timedwait.3c
381 file path=usr/share/man/man3c/sem_unlink.3c
382 file path=usr/share/man/man3c/sem_wait.3c
383 file path=usr/share/man/man3c/semaphore.3c
384 file path=usr/share/man/man3c/setbuf.3c
385 file path=usr/share/man/man3c/setbuffer.3c
386 file path=usr/share/man/man3c/setcat.3c
387 file path=usr/share/man/man3c/setenv.3c
388 file path=usr/share/man/man3c/setjmp.3c

```

```

389 file path=usr/share/man/man3c/setkey.3c
390 file path=usr/share/man/man3c/setlabel.3c
391 file path=usr/share/man/man3c/setlocale.3c
392 file path=usr/share/man/man3c/shm_open.3c
393 file path=usr/share/man/man3c/shm_unlink.3c
394 file path=usr/share/man/man3c/sigfpe.3c
395 file path=usr/share/man/man3c/siginterrupt.3c
396 file path=usr/share/man/man3c/signal.3c
397 file path=usr/share/man/man3c/sigqueue.3c
398 file path=usr/share/man/man3c/sigsetops.3c
399 file path=usr/share/man/man3c/sigstack.3c
400 file path=usr/share/man/man3c/sigwaitinfo.3c
401 file path=usr/share/man/man3c/sleep.3c
402 file path=usr/share/man/man3c/ssignal.3c
403 file path=usr/share/man/man3c/stack_getbounds.3c
404 file path=usr/share/man/man3c/stack_inbounds.3c
405 file path=usr/share/man/man3c/stack_setbounds.3c
406 file path=usr/share/man/man3c/stack_violation.3c
407 file path=usr/share/man/man3c/stdio.3c
408 file path=usr/share/man/man3c/str2sig.3c
409 file path=usr/share/man/man3c/strcoll.3c
410 file path=usr/share/man/man3c/streerror.3c
411 file path=usr/share/man/man3c/strfmon.3c
412 file path=usr/share/man/man3c/strftime.3c
413 file path=usr/share/man/man3c/string.3c
414 file path=usr/share/man/man3c/string_to_decimal.3c
415 file path=usr/share/man/man3c/strptime.3c
416 file path=usr/share/man/man3c/strsignal.3c
417 file path=usr/share/man/man3c/strtod.3c
418 file path=usr/share/man/man3c/strtoimax.3c
419 file path=usr/share/man/man3c/strtol.3c
420 file path=usr/share/man/man3c/strtoul.3c
421 file path=usr/share/man/man3c/strtows.3c
422 file path=usr/share/man/man3c/strxfrm.3c
423 file path=usr/share/man/man3c/swab.3c
424 file path=usr/share/man/man3c/sync_instruction_memory.3c
425 file path=usr/share/man/man3c/sysconf.3c
426 file path=usr/share/man/man3c/syslog.3c
427 file path=usr/share/man/man3c/system.3c
428 file path=usr/share/man/man3c/tcdrain.3c
429 file path=usr/share/man/man3c/tcflow.3c
430 file path=usr/share/man/man3c/tcflush.3c
431 file path=usr/share/man/man3c/tcgetattr.3c
432 file path=usr/share/man/man3c/tcgetpgrp.3c
433 file path=usr/share/man/man3c/tcgetsid.3c
434 file path=usr/share/man/man3c/tcsendbreak.3c
435 file path=usr/share/man/man3c/tcsetattr.3c
436 file path=usr/share/man/man3c/tcsetpgrp.3c
437 file path=usr/share/man/man3c/tell.3c
438 file path=usr/share/man/man3c/telldir.3c
439 file path=usr/share/man/man3c/termios.3c
440 file path=usr/share/man/man3c/thr_create.3c
441 file path=usr/share/man/man3c/thr_exit.3c
442 file path=usr/share/man/man3c/thr_getconcurrency.3c
443 file path=usr/share/man/man3c/thr_getprio.3c
444 file path=usr/share/man/man3c/thr_join.3c
445 file path=usr/share/man/man3c/thr_keycreate.3c
446 file path=usr/share/man/man3c/thr_kill.3c
447 file path=usr/share/man/man3c/thr_main.3c
448 file path=usr/share/man/man3c/thr_min_stack.3c
449 file path=usr/share/man/man3c/thr_self.3c
450 file path=usr/share/man/man3c/thr_sigsetmask.3c
451 file path=usr/share/man/man3c/thr_stksegment.3c
452 file path=usr/share/man/man3c/thr_suspend.3c
453 file path=usr/share/man/man3c/thr_yield.3c
454 file path=usr/share/man/man3c/timer_create.3c

```

```

455 file path=usr/share/man/man3c/timer_delete.3c
456 file path=usr/share/man/man3c/timer_settime.3c
457 file path=usr/share/man/man3c/timeradd.3c
458 file path=usr/share/man/man3c/tmpfile.3c
459 file path=usr/share/man/man3c/tmpnam.3c
460 file path=usr/share/man/man3c/toascii.3c
461 file path=usr/share/man/man3c/tolower.3c
462 file path=usr/share/man/man3c/toupper.3c
463 file path=usr/share/man/man3c/towlower.3c
464 file path=usr/share/man/man3c/toupper.3c
465 file path=usr/share/man/man3c/truncate.3c
466 file path=usr/share/man/man3c/tsearch.3c
467 file path=usr/share/man/man3c/ttyname.3c
468 file path=usr/share/man/man3c/ttyslot.3c
469 file path=usr/share/man/man3c/u8_strcmp.3c
470 file path=usr/share/man/man3c/u8_textprep_str.3c
471 file path=usr/share/man/man3c/u8_validate.3c
472 file path=usr/share/man/man3c/ualarm.3c
473 file path=usr/share/man/man3c/uconv_ul6tou32.3c
474 file path=usr/share/man/man3c/ucred_get.3c
475 file path=usr/share/man/man3c/ungetc.3c
476 file path=usr/share/man/man3c/ungetwc.3c
477 file path=usr/share/man/man3c/unlockpt.3c
478 file path=usr/share/man/man3c/unsetenv.3c
479 file path=usr/share/man/man3c/uselocale.3c
480 file path=usr/share/man/man3c/usleep.3c
481 file path=usr/share/man/man3c/vfwprintf.3c
482 file path=usr/share/man/man3c/vlfmt.3c
483 file path=usr/share/man/man3c/vpfmt.3c
484 file path=usr/share/man/man3c/vprintf.3c
485 file path=usr/share/man/man3c/vsyslog.3c
486 file path=usr/share/man/man3c/wait.3c
487 file path=usr/share/man/man3c/wait3.3c
488 file path=usr/share/man/man3c/waitpid.3c
489 file path=usr/share/man/man3c/walkcontext.3c
490 file path=usr/share/man/man3c/wcrtomb.3c
491 file path=usr/share/man/man3c/wcsoll.3c
492 file path=usr/share/man/man3c/wcsftime.3c
493 file path=usr/share/man/man3c/wcsrtombs.3c
494 file path=usr/share/man/man3c/wcsstr.3c
495 file path=usr/share/man/man3c/wcstod.3c
496 file path=usr/share/man/man3c/wcstoimax.3c
497 file path=usr/share/man/man3c/wcstol.3c
498 file path=usr/share/man/man3c/wcstoul.3c
499 file path=usr/share/man/man3c/wcstring.3c
500 file path=usr/share/man/man3c/wcswidth.3c
501 file path=usr/share/man/man3c/wcsxfrm.3c
502 file path=usr/share/man/man3c/wctob.3c
503 file path=usr/share/man/man3c/wctomb.3c
504 file path=usr/share/man/man3c/wctrans.3c
505 file path=usr/share/man/man3c/wctype.3c
506 file path=usr/share/man/man3c/wcwidth.3c
507 file path=usr/share/man/man3c/wmemchr.3c
508 file path=usr/share/man/man3c/wmemcmp.3c
509 file path=usr/share/man/man3c/wmemcpy.3c
510 file path=usr/share/man/man3c/wmemmove.3c
511 file path=usr/share/man/man3c/wmemset.3c
512 file path=usr/share/man/man3c/wordexp.3c
513 file path=usr/share/man/man3c/wsprintf.3c
514 file path=usr/share/man/man3c/wscanf.3c
515 file path=usr/share/man/man3c/wstring.3c
516 link path=usr/share/man/man3c/FD_CLR.3c target=select.3c
517 link path=usr/share/man/man3c/FD_ISSET.3c target=select.3c
518 link path=usr/share/man/man3c/FD_SET.3c target=select.3c
519 link path=usr/share/man/man3c/FD_ZERO.3c target=select.3c
520 link path=usr/share/man/man3c/___fblk.3c target=___fbufsize.3c

```

```

521 link path=usr/share/man/man3c/_fpending.3c target=__fbufsize.3c
522 link path=usr/share/man/man3c/_fpurge.3c target=__fbufsize.3c
523 link path=usr/share/man/man3c/_freadable.3c target=__fbufsize.3c
524 link path=usr/share/man/man3c/_freading.3c target=__fbufsize.3c
525 link path=usr/share/man/man3c/_fsetlocking.3c target=__fbufsize.3c
526 link path=usr/share/man/man3c/_fwritable.3c target=__fbufsize.3c
527 link path=usr/share/man/man3c/_fwriting.3c target=__fbufsize.3c
528 link path=usr/share/man/man3c/_edata.3c target=end.3c
529 link path=usr/share/man/man3c/_end.3c target=end.3c
530 link path=usr/share/man/man3c/_etext.3c target=end.3c
531 link path=usr/share/man/man3c/_exithandle.3c target=exit.3c
532 link path=usr/share/man/man3c/_flushlbf.3c target=__fbufsize.3c
533 link path=usr/share/man/man3c/_setjmp.3c target=_longjmp.3c
534 link path=usr/share/man/man3c/addrtosymstr.3c target=walkcontext.3c
535 link path=usr/share/man/man3c/aiowrite.3c target=aioread.3c
536 link path=usr/share/man/man3c/alloca.3c target=malloc.3c
537 link path=usr/share/man/man3c/alphasort.3c target=scandir.3c
538 link path=usr/share/man/man3c/asctime.3c target=strftime.3c
539 link path=usr/share/man/man3c/asctime_r.3c target=ctime.3c
540 link path=usr/share/man/man3c/asprintf.3c target=printf.3c
541 link path=usr/share/man/man3c/atof.3c target=strtod.3c
542 link path=usr/share/man/man3c/atoi.3c target=strtol.3c
543 link path=usr/share/man/man3c/atol.3c target=strtol.3c
544 link path=usr/share/man/man3c/atoll.3c target=strtol.3c
545 link path=usr/share/man/man3c/atomic_add_16.3c target=atomic_add.3c
546 link path=usr/share/man/man3c/atomic_add_16_nv.3c target=atomic_add.3c
547 link path=usr/share/man/man3c/atomic_add_16_nv.3c target=atomic_add.3c
548 link path=usr/share/man/man3c/atomic_add_32.3c target=atomic_add.3c
549 link path=usr/share/man/man3c/atomic_add_32_nv.3c target=atomic_add.3c
550 link path=usr/share/man/man3c/atomic_add_64.3c target=atomic_add.3c
551 link path=usr/share/man/man3c/atomic_add_64_nv.3c target=atomic_add.3c
552 link path=usr/share/man/man3c/atomic_add_8.3c target=atomic_add.3c
553 link path=usr/share/man/man3c/atomic_add_8_nv.3c target=atomic_add.3c
554 link path=usr/share/man/man3c/atomic_add_char.3c target=atomic_add.3c
555 link path=usr/share/man/man3c/atomic_add_char_nv.3c target=atomic_add.3c
556 link path=usr/share/man/man3c/atomic_add_int.3c target=atomic_add.3c
557 link path=usr/share/man/man3c/atomic_add_int_nv.3c target=atomic_add.3c
558 link path=usr/share/man/man3c/atomic_add_long.3c target=atomic_add.3c
559 link path=usr/share/man/man3c/atomic_add_long_nv.3c target=atomic_add.3c
560 link path=usr/share/man/man3c/atomic_add_ptr.3c target=atomic_add.3c
561 link path=usr/share/man/man3c/atomic_add_ptr_nv.3c target=atomic_add.3c
562 link path=usr/share/man/man3c/atomic_add_short.3c target=atomic_add.3c
563 link path=usr/share/man/man3c/atomic_add_short_nv.3c target=atomic_add.3c
564 link path=usr/share/man/man3c/atomic_and_16.3c target=atomic_and.3c
565 link path=usr/share/man/man3c/atomic_and_16_nv.3c target=atomic_and.3c
566 link path=usr/share/man/man3c/atomic_and_32.3c target=atomic_and.3c
567 link path=usr/share/man/man3c/atomic_and_32_nv.3c target=atomic_and.3c
568 link path=usr/share/man/man3c/atomic_and_64.3c target=atomic_and.3c
569 link path=usr/share/man/man3c/atomic_and_64_nv.3c target=atomic_and.3c
570 link path=usr/share/man/man3c/atomic_and_8.3c target=atomic_and.3c
571 link path=usr/share/man/man3c/atomic_and_8_nv.3c target=atomic_and.3c
572 link path=usr/share/man/man3c/atomic_and_uchar.3c target=atomic_and.3c
573 link path=usr/share/man/man3c/atomic_and_uchar_nv.3c target=atomic_and.3c
574 link path=usr/share/man/man3c/atomic_and_uint.3c target=atomic_and.3c
575 link path=usr/share/man/man3c/atomic_and_uint_nv.3c target=atomic_and.3c
576 link path=usr/share/man/man3c/atomic_and_ulong.3c target=atomic_and.3c
577 link path=usr/share/man/man3c/atomic_and_ulong_nv.3c target=atomic_and.3c
578 link path=usr/share/man/man3c/atomic_and_ushort.3c target=atomic_and.3c
579 link path=usr/share/man/man3c/atomic_and_ushort_nv.3c target=atomic_and.3c
580 link path=usr/share/man/man3c/atomic_cas_16.3c target=atomic_cas.3c
581 link path=usr/share/man/man3c/atomic_cas_32.3c target=atomic_cas.3c
582 link path=usr/share/man/man3c/atomic_cas_64.3c target=atomic_cas.3c
583 link path=usr/share/man/man3c/atomic_cas_8.3c target=atomic_cas.3c
584 link path=usr/share/man/man3c/atomic_cas_ptr.3c target=atomic_cas.3c
585 link path=usr/share/man/man3c/atomic_cas_uchar.3c target=atomic_cas.3c
586 link path=usr/share/man/man3c/atomic_cas_uint.3c target=atomic_cas.3c

```

```

587 link path=usr/share/man/man3c/atomic_cas_ulong.3c target=atomic_cas.3c
588 link path=usr/share/man/man3c/atomic_cas_ushort.3c target=atomic_cas.3c
589 link path=usr/share/man/man3c/atomic_clear_long_excl.3c target=atomic_bits.3c
590 link path=usr/share/man/man3c/atomic_dec_16.3c target=atomic_dec.3c
591 link path=usr/share/man/man3c/atomic_dec_16_nv.3c target=atomic_dec.3c
592 link path=usr/share/man/man3c/atomic_dec_32.3c target=atomic_dec.3c
593 link path=usr/share/man/man3c/atomic_dec_32_nv.3c target=atomic_dec.3c
594 link path=usr/share/man/man3c/atomic_dec_64.3c target=atomic_dec.3c
595 link path=usr/share/man/man3c/atomic_dec_64_nv.3c target=atomic_dec.3c
596 link path=usr/share/man/man3c/atomic_dec_8.3c target=atomic_dec.3c
597 link path=usr/share/man/man3c/atomic_dec_8_nv.3c target=atomic_dec.3c
598 link path=usr/share/man/man3c/atomic_dec_ptr.3c target=atomic_dec.3c
599 link path=usr/share/man/man3c/atomic_dec_ptr_nv.3c target=atomic_dec.3c
600 link path=usr/share/man/man3c/atomic_dec_uchar.3c target=atomic_dec.3c
601 link path=usr/share/man/man3c/atomic_dec_uchar_nv.3c target=atomic_dec.3c
602 link path=usr/share/man/man3c/atomic_dec_uint.3c target=atomic_dec.3c
603 link path=usr/share/man/man3c/atomic_dec_uint_nv.3c target=atomic_dec.3c
604 link path=usr/share/man/man3c/atomic_dec_ulong.3c target=atomic_dec.3c
605 link path=usr/share/man/man3c/atomic_dec_ulong_nv.3c target=atomic_dec.3c
606 link path=usr/share/man/man3c/atomic_dec_ushort.3c target=atomic_dec.3c
607 link path=usr/share/man/man3c/atomic_dec_ushort_nv.3c target=atomic_dec.3c
608 link path=usr/share/man/man3c/atomic_inc_16.3c target=atomic_inc.3c
609 link path=usr/share/man/man3c/atomic_inc_16_nv.3c target=atomic_inc.3c
610 link path=usr/share/man/man3c/atomic_inc_32.3c target=atomic_inc.3c
611 link path=usr/share/man/man3c/atomic_inc_32_nv.3c target=atomic_inc.3c
612 link path=usr/share/man/man3c/atomic_inc_64.3c target=atomic_inc.3c
613 link path=usr/share/man/man3c/atomic_inc_64_nv.3c target=atomic_inc.3c
614 link path=usr/share/man/man3c/atomic_inc_8.3c target=atomic_inc.3c
615 link path=usr/share/man/man3c/atomic_inc_8_nv.3c target=atomic_inc.3c
616 link path=usr/share/man/man3c/atomic_inc_ptr.3c target=atomic_inc.3c
617 link path=usr/share/man/man3c/atomic_inc_ptr_nv.3c target=atomic_inc.3c
618 link path=usr/share/man/man3c/atomic_inc_uchar.3c target=atomic_inc.3c
619 link path=usr/share/man/man3c/atomic_inc_uchar_nv.3c target=atomic_inc.3c
620 link path=usr/share/man/man3c/atomic_inc_uint.3c target=atomic_inc.3c
621 link path=usr/share/man/man3c/atomic_inc_uint_nv.3c target=atomic_inc.3c
622 link path=usr/share/man/man3c/atomic_inc_ulong.3c target=atomic_inc.3c
623 link path=usr/share/man/man3c/atomic_inc_ulong_nv.3c target=atomic_inc.3c
624 link path=usr/share/man/man3c/atomic_inc_ushort.3c target=atomic_inc.3c
625 link path=usr/share/man/man3c/atomic_inc_ushort_nv.3c target=atomic_inc.3c
626 link path=usr/share/man/man3c/atomic_or_16.3c target=atomic_or.3c
627 link path=usr/share/man/man3c/atomic_or_16_nv.3c target=atomic_or.3c
628 link path=usr/share/man/man3c/atomic_or_32.3c target=atomic_or.3c
629 link path=usr/share/man/man3c/atomic_or_32_nv.3c target=atomic_or.3c
630 link path=usr/share/man/man3c/atomic_or_64.3c target=atomic_or.3c
631 link path=usr/share/man/man3c/atomic_or_64_nv.3c target=atomic_or.3c
632 link path=usr/share/man/man3c/atomic_or_8.3c target=atomic_or.3c
633 link path=usr/share/man/man3c/atomic_or_8_nv.3c target=atomic_or.3c
634 link path=usr/share/man/man3c/atomic_or_ptr.3c target=atomic_or.3c
635 link path=usr/share/man/man3c/atomic_or_ptr_nv.3c target=atomic_or.3c
636 link path=usr/share/man/man3c/atomic_or_uint.3c target=atomic_or.3c
637 link path=usr/share/man/man3c/atomic_or_uint_nv.3c target=atomic_or.3c
638 link path=usr/share/man/man3c/atomic_or_ulong.3c target=atomic_or.3c
639 link path=usr/share/man/man3c/atomic_or_ulong_nv.3c target=atomic_or.3c
640 link path=usr/share/man/man3c/atomic_or_ushort.3c target=atomic_or.3c
641 link path=usr/share/man/man3c/atomic_or_ushort_nv.3c target=atomic_or.3c
642 link path=usr/share/man/man3c/atomic_set_long_excl.3c target=atomic_bits.3c
643 link path=usr/share/man/man3c/atomic_swap_16.3c target=atomic_swap.3c
644 link path=usr/share/man/man3c/atomic_swap_32.3c target=atomic_swap.3c
645 link path=usr/share/man/man3c/atomic_swap_64.3c target=atomic_swap.3c
646 link path=usr/share/man/man3c/atomic_swap_8.3c target=atomic_swap.3c
647 link path=usr/share/man/man3c/atomic_swap_ptr.3c target=atomic_swap.3c
648 link path=usr/share/man/man3c/atomic_swap_uchar.3c target=atomic_swap.3c
649 link path=usr/share/man/man3c/atomic_swap_uint.3c target=atomic_swap.3c
650 link path=usr/share/man/man3c/atomic_swap_ulong.3c target=atomic_swap.3c
651 link path=usr/share/man/man3c/atomic_swap_ushort.3c target=atomic_swap.3c
652 link path=usr/share/man/man3c/backtrace.3c target=walkcontext.3c

```

```

653 link path=usr/share/man/man3c/backtrace_symbols.3c target=walkcontext.3c
654 link path=usr/share/man/man3c/backtrace_symbols_fd.3c target=walkcontext.3c
655 link path=usr/share/man/man3c/bcmp.3c target=bstring.3c
656 link path=usr/share/man/man3c/bcopy.3c target=bstring.3c
657 link path=usr/share/man/man3c/bind_textdomain_codeset.3c target=gettext.3c
658 link path=usr/share/man/man3c/bind_textdomain.3c target=gettext.3c
659 link path=usr/share/man/man3c/btowc.1.3c target=btowc.3c
660 link path=usr/share/man/man3c/bzero.3c target=bstring.3c
661 link path=usr/share/man/man3c/calloc.3c target=malloc.3c
662 link path=usr/share/man/man3c/catclose.3c target=catopen.3c
663 link path=usr/share/man/man3c/cfgetospeed.3c target=cfgetispeed.3c
664 link path=usr/share/man/man3c/cfsetospeed.3c target=cfsetispeed.3c
665 link path=usr/share/man/man3c/cftime.3c target=strftime.3c
666 link path=usr/share/man/man3c/clearerr.3c target=ferror.3c
667 link path=usr/share/man/man3c/clock_getres.3c target=clock_settime.3c
668 link path=usr/share/man/man3c/clock_gettime.3c target=clock_settime.3c
669 link path=usr/share/man/man3c/closelog.3c target=syslog.3c
670 link path=usr/share/man/man3c/cond_broadcast.3c target=cond_init.3c
671 link path=usr/share/man/man3c/cond_destroy.3c target=cond_init.3c
672 link path=usr/share/man/man3c/cond_reltimedwait.3c target=cond_init.3c
673 link path=usr/share/man/man3c/cond_signal.3c target=cond_init.3c
674 link path=usr/share/man/man3c/cond_timedwait.3c target=cond_init.3c
675 link path=usr/share/man/man3c/cond_wait.3c target=cond_init.3c
676 link path=usr/share/man/man3c/csetcol.3c target=cset.3c
677 link path=usr/share/man/man3c/csetlen.3c target=cset.3c
678 link path=usr/share/man/man3c/csetno.3c target=cset.3c
679 link path=usr/share/man/man3c/ctermid_r.3c target=ctermid.3c
680 link path=usr/share/man/man3c/ctime_r.3c target=ctime.3c
681 link path=usr/share/man/man3c/dbm_clearerr.3c target=ndbm.3c
682 link path=usr/share/man/man3c/dbm_close.3c target=ndbm.3c
683 link path=usr/share/man/man3c/dbm_delete.3c target=ndbm.3c
684 link path=usr/share/man/man3c/dbm_error.3c target=ndbm.3c
685 link path=usr/share/man/man3c/dbm_fetch.3c target=ndbm.3c
686 link path=usr/share/man/man3c/dbm_firstkey.3c target=ndbm.3c
687 link path=usr/share/man/man3c/dbm_nextkey.3c target=ndbm.3c
688 link path=usr/share/man/man3c/dbm_open.3c target=ndbm.3c
689 link path=usr/share/man/man3c/dbm_store.3c target=ndbm.3c
690 link path=usr/share/man/man3c/dcgettext.3c target=gettext.3c
691 link path=usr/share/man/man3c/dcngettext.3c target=gettext.3c
692 link path=usr/share/man/man3c/decimal_to_double.3c \
693 target=decimal_to_floating.3c
694 link path=usr/share/man/man3c/decimal_to_extended.3c \
695 target=decimal_to_floating.3c
696 link path=usr/share/man/man3c/decimal_to_quadruple.3c \
697 target=decimal_to_floating.3c
698 link path=usr/share/man/man3c/decimal_to_single.3c \
699 target=decimal_to_floating.3c
700 link path=usr/share/man/man3c/dgettext.3c target=gettext.3c
701 link path=usr/share/man/man3c/dladdr1.3c target=dladdr.3c
702 link path=usr/share/man/man3c/dlclose.3c target=dlopen.3c
703 link path=usr/share/man/man3c/dngettext.3c target=gettext.3c
704 link path=usr/share/man/man3c/door_setparam.3c target=door_getparam.3c
705 link path=usr/share/man/man3c/door_unbind.3c target=door_bind.3c
706 link path=usr/share/man/man3c/double_to_decimal.3c \
707 target=floating_to_decimal.3c
708 link path=usr/share/man/man3c/dup3.3c target=dup2.3c
709 link path=usr/share/man/man3c/duplocale.3c target=newlocale.3c
710 link path=usr/share/man/man3c/edata.3c target=end.3c
711 link path=usr/share/man/man3c/endlrent.3c target=getgrnam.3c
712 link path=usr/share/man/man3c/endnetgrent.3c target=getnetgrent.3c
713 link path=usr/share/man/man3c/endlw.3c target=getpwnam.3c
714 link path=usr/share/man/man3c/endlw.3c target=getpwnam.3c
715 link path=usr/share/man/man3c/endusershell.3c target=getusershell.3c
716 link path=usr/share/man/man3c/endutent.3c target=getutent.3c
717 link path=usr/share/man/man3c/endutxent.3c target=getutxent.3c
718 link path=usr/share/man/man3c/erand48.3c target=drand48.3c

```

```

719 link path=usr/share/man/man3c/errno.3c target=perror.3c
720 link path=usr/share/man/man3c/errx.3c target=err.3c
721 link path=usr/share/man/man3c/etext.3c target=end.3c
722 link path=usr/share/man/man3c/euclid.3c target=euclid.3c
723 link path=usr/share/man/man3c/euclid.3c target=euclid.3c
724 link path=usr/share/man/man3c/extended_to_decimal.3c \
725 target=floating_to_decimal.3c
726 link path=usr/share/man/man3c/fconvert.3c target=econvert.3c
727 link path=usr/share/man/man3c/fcvt.3c target=ecvt.3c
728 link path=usr/share/man/man3c/fdopendir.3c target=opendir.3c
729 link path=usr/share/man/man3c/fdwalk.3c target=closedir.3c
730 link path=usr/share/man/man3c/feof.3c target=feof.3c
731 link path=usr/share/man/man3c/fgetgrent.3c target=getgrnam.3c
732 link path=usr/share/man/man3c/fgetgrent_r.3c target=getgrnam.3c
733 link path=usr/share/man/man3c/fgetpwent.3c target=getpwnam.3c
734 link path=usr/share/man/man3c/fgetpwent_r.3c target=getpwnam.3c
735 link path=usr/share/man/man3c/fgets.3c target=getc.3c
736 link path=usr/share/man/man3c/fgetspent.3c target=getspent.3c
737 link path=usr/share/man/man3c/fgetspent_r.3c target=getspent.3c
738 link path=usr/share/man/man3c/fgetwc.1.3c target=fgetwc.3c
739 link path=usr/share/man/man3c/fgetws.3c target=getws.3c
740 link path=usr/share/man/man3c/file_to_decimal.3c target=string_to_decimal.3c
741 link path=usr/share/man/man3c/fileno.3c target=error.3c
742 link path=usr/share/man/man3c/finite.3c target=isnan.3c
743 link path=usr/share/man/man3c/fpclass.3c target=isnan.3c
744 link path=usr/share/man/man3c/fpgetmask.3c target=fpgetround.3c
745 link path=usr/share/man/man3c/fpgetsticky.3c target=fpgetround.3c
746 link path=usr/share/man/man3c/fprintf.3c target=printf.3c
747 link path=usr/share/man/man3c/fpsetmask.3c target=fpgetround.3c
748 link path=usr/share/man/man3c/fpsetround.3c target=fpgetround.3c
749 link path=usr/share/man/man3c/fpsetsticky.3c target=fpgetround.3c
750 link path=usr/share/man/man3c/fputs.3c target=puts.3c
751 link path=usr/share/man/man3c/free.3c target=malloc.3c
752 link path=usr/share/man/man3c/freelocale.3c target=newlocale.3c
753 link path=usr/share/man/man3c/fscanf.3c target=scanf.3c
754 link path=usr/share/man/man3c/fseek.3c target=fseek.3c
755 link path=usr/share/man/man3c/fsetat.3c target=fgetat.3c
756 link path=usr/share/man/man3c/ftello.3c target=ftell.3c
757 link path=usr/share/man/man3c/ftruncate.3c target=truncate.3c
758 link path=usr/share/man/man3c/ftrylockfile.3c target=flockfile.3c
759 link path=usr/share/man/man3c/func_to_decimal.3c target=string_to_decimal.3c
760 link path=usr/share/man/man3c/funlockfile.3c target=flockfile.3c
761 link path=usr/share/man/man3c/gconvert.3c target=econvert.3c
762 link path=usr/share/man/man3c/gcvt.3c target=ecvt.3c
763 link path=usr/share/man/man3c/getattr.3c target=fgetat.3c
764 link path=usr/share/man/man3c/getc.3c target=fgetc.3c
765 link path=usr/share/man/man3c/getc_unlocked.3c target=fgetc.3c
766 link path=usr/share/man/man3c/getchar.3c target=fgetc.3c
767 link path=usr/share/man/man3c/getchar_unlocked.3c target=fgetc.3c
768 link path=usr/share/man/man3c/getdelim.3c target=getline.3c
769 link path=usr/share/man/man3c/getextmntent.3c target=getmntent.3c
770 link path=usr/share/man/man3c/getgrent.3c target=getgrnam.3c
771 link path=usr/share/man/man3c/getgrent_r.3c target=getgrnam.3c
772 link path=usr/share/man/man3c/getgrgid.3c target=getgrnam.3c
773 link path=usr/share/man/man3c/getgrgid_r.3c target=getgrnam.3c
774 link path=usr/share/man/man3c/getgrnam.3c target=getgrnam.3c
775 link path=usr/share/man/man3c/gethomedir.3c target=getpuid.3c
776 link path=usr/share/man/man3c/gethrtime.3c target=gethrtime.3c
777 link path=usr/share/man/man3c/getlogin_r.3c target=getlogin.3c
778 link path=usr/share/man/man3c/getmntany.3c target=getmntent.3c
779 link path=usr/share/man/man3c/getnetgrent_r.3c target=getnetgrent.3c
780 link path=usr/share/man/man3c/getpassphrase.3c target=getpass.3c
781 link path=usr/share/man/man3c/getp.3c target=getpwnam.3c
782 link path=usr/share/man/man3c/getp.3c target=getpwnam.3c
783 link path=usr/share/man/man3c/getpwnam_r.3c target=getpwnam.3c
784 link path=usr/share/man/man3c/getpwuid.3c target=getpwnam.3c

```

```

785 link path=usr/share/man/man3c/getpwuid_r.3c target=getpwnam.3c
786 link path=usr/share/man/man3c/getspent.3c target=getspnam.3c
787 link path=usr/share/man/man3c/getspent_r.3c target=getspnam.3c
788 link path=usr/share/man/man3c/getspnam_r.3c target=getspnam.3c
789 link path=usr/share/man/man3c/getutid.3c target=getutent.3c
790 link path=usr/share/man/man3c/getutline.3c target=getutent.3c
791 link path=usr/share/man/man3c/getutmp.3c target=getutxent.3c
792 link path=usr/share/man/man3c/getutmpx.3c target=getutxent.3c
793 link path=usr/share/man/man3c/getutxid.3c target=getutxent.3c
794 link path=usr/share/man/man3c/getutxline.3c target=getutxent.3c
795 link path=usr/share/man/man3c/getvfsany.3c target=getvfsent.3c
796 link path=usr/share/man/man3c/getvfsfile.3c target=getvfssent.3c
797 link path=usr/share/man/man3c/getvfsspec.3c target=getvfssent.3c
798 link path=usr/share/man/man3c/getw.3c target=fgetc.3c
799 link path=usr/share/man/man3c/getwc_1.3c target=getwc.3c
800 link path=usr/share/man/man3c/getwchar_1.3c target=getwchar.3c
801 link path=usr/share/man/man3c/getzoneidbyname.3c target=getzoneid.3c
802 link path=usr/share/man/man3c/getzonenamebyid.3c target=getzoneid.3c
803 link path=usr/share/man/man3c/globfree.3c target=glob.3c
804 link path=usr/share/man/man3c/gmtime.3c target=ctime.3c
805 link path=usr/share/man/man3c/gmtime_r.3c target=ctime.3c
806 link path=usr/share/man/man3c/gsignal.3c target=ssignal.3c
807 link path=usr/share/man/man3c/hasmntopt.3c target=getmntent.3c
808 link path=usr/share/man/man3c/hcreate.3c target=hsearch.3c
809 link path=usr/share/man/man3c/hdestroy.3c target=hsearch.3c
810 link path=usr/share/man/man3c/initstate.3c target=random.3c
811 link path=usr/share/man/man3c/innetgr.3c target=getnetgrent.3c
812 link path=usr/share/man/man3c/isalnum.3c target=ctype.3c
813 link path=usr/share/man/man3c/isalnum_1.3c target=ctype.3c
814 link path=usr/share/man/man3c/isalpha.3c target=ctype.3c
815 link path=usr/share/man/man3c/isalpha_1.3c target=ctype.3c
816 link path=usr/share/man/man3c/isascii.3c target=ctype.3c
817 link path=usr/share/man/man3c/isblank.3c target=ctype.3c
818 link path=usr/share/man/man3c/isblank_1.3c target=ctype.3c
819 link path=usr/share/man/man3c/iscntrl.3c target=ctype.3c
820 link path=usr/share/man/man3c/iscntrl_1.3c target=ctype.3c
821 link path=usr/share/man/man3c/isdigit.3c target=ctype.3c
822 link path=usr/share/man/man3c/isdigit_1.3c target=ctype.3c
823 link path=usr/share/man/man3c/isenglish.3c target=iswalpha.3c
824 link path=usr/share/man/man3c/isgraph.3c target=ctype.3c
825 link path=usr/share/man/man3c/isgraph_1.3c target=ctype.3c
826 link path=usr/share/man/man3c/isideogram.3c target=iswalpha.3c
827 link path=usr/share/man/man3c/islower.3c target=ctype.3c
828 link path=usr/share/man/man3c/islower_1.3c target=ctype.3c
829 link path=usr/share/man/man3c/isnanf.3c target=isnan.3c
830 link path=usr/share/man/man3c/isnumber.3c target=iswalpha.3c
831 link path=usr/share/man/man3c/isphonogram.3c target=iswalpha.3c
832 link path=usr/share/man/man3c/isprint.3c target=ctype.3c
833 link path=usr/share/man/man3c/isprint_1.3c target=ctype.3c
834 link path=usr/share/man/man3c/ispunct.3c target=ctype.3c
835 link path=usr/share/man/man3c/ispunct_1.3c target=ctype.3c
836 link path=usr/share/man/man3c/isspace.3c target=ctype.3c
837 link path=usr/share/man/man3c/isspace_1.3c target=ctype.3c
838 link path=usr/share/man/man3c/isspecial.3c target=iswalpha.3c
839 link path=usr/share/man/man3c/isupper.3c target=ctype.3c
840 link path=usr/share/man/man3c/isupper_1.3c target=ctype.3c
841 link path=usr/share/man/man3c/iswalnum.3c target=iswalpha.3c
842 link path=usr/share/man/man3c/iswalnum_1.3c target=iswalpha.3c
843 link path=usr/share/man/man3c/iswalnum_1.3c target=iswalpha.3c
844 link path=usr/share/man/man3c/iswascii.3c target=iswalpha.3c
845 link path=usr/share/man/man3c/iswblank.3c target=iswalpha.3c
846 link path=usr/share/man/man3c/iswblank_1.3c target=iswalpha.3c
847 link path=usr/share/man/man3c/iswcntrl.3c target=iswalpha.3c
848 link path=usr/share/man/man3c/iswcntrl_1.3c target=iswalpha.3c
849 link path=usr/share/man/man3c/iswctype_1.3c target=iswctype.3c
850 link path=usr/share/man/man3c/iswdigit.3c target=iswalpha.3c

```

```

851 link path=usr/share/man/man3c/iswdigit_1.3c target=iswalpha.3c
852 link path=usr/share/man/man3c/iswgraph.3c target=iswalpha.3c
853 link path=usr/share/man/man3c/iswgraph_1.3c target=iswalpha.3c
854 link path=usr/share/man/man3c/iswhexnumber.3c target=iswalpha.3c
855 link path=usr/share/man/man3c/iswhexnumber_1.3c target=iswalpha.3c
856 link path=usr/share/man/man3c/iswideogram.3c target=iswalpha.3c
857 link path=usr/share/man/man3c/iswideogram_1.3c target=iswalpha.3c
858 link path=usr/share/man/man3c/iswlower.3c target=iswalpha.3c
859 link path=usr/share/man/man3c/iswlower_1.3c target=iswalpha.3c
860 link path=usr/share/man/man3c/iswnumber.3c target=iswalpha.3c
861 link path=usr/share/man/man3c/iswnumber_1.3c target=iswalpha.3c
862 link path=usr/share/man/man3c/iswphonogram.3c target=iswalpha.3c
863 link path=usr/share/man/man3c/iswphonogram_1.3c target=iswalpha.3c
864 link path=usr/share/man/man3c/iswprint.3c target=iswalpha.3c
865 link path=usr/share/man/man3c/iswprint_1.3c target=iswalpha.3c
866 link path=usr/share/man/man3c/iswpunct.3c target=iswalpha.3c
867 link path=usr/share/man/man3c/iswpunct_1.3c target=iswalpha.3c
868 link path=usr/share/man/man3c/iswspace.3c target=iswalpha.3c
869 link path=usr/share/man/man3c/iswspace_1.3c target=iswalpha.3c
870 link path=usr/share/man/man3c/iswspecial.3c target=iswalpha.3c
871 link path=usr/share/man/man3c/iswspecial_1.3c target=iswalpha.3c
872 link path=usr/share/man/man3c/iswupper.3c target=iswalpha.3c
873 link path=usr/share/man/man3c/iswupper_1.3c target=iswalpha.3c
874 link path=usr/share/man/man3c/iswxdigit.3c target=iswalpha.3c
875 link path=usr/share/man/man3c/iswxdigit_1.3c target=iswalpha.3c
876 link path=usr/share/man/man3c/isxdigit.3c target=ctype.3c
877 link path=usr/share/man/man3c/isxdigit_1.3c target=ctype.3c
878 link path=usr/share/man/man3c/jrand48.3c target=drand48.3c
879 link path=usr/share/man/man3c/l64a.3c target=a64l.3c
880 link path=usr/share/man/man3c/labs.3c target=abs.3c
881 link path=usr/share/man/man3c/lcong48.3c target=drand48.3c
882 link path=usr/share/man/man3c/ldiv.3c target=div.3c
883 link path=usr/share/man/man3c/lfind.3c target=lsearch.3c
884 link path=usr/share/man/man3c/llabs.3c target=abs.3c
885 link path=usr/share/man/man3c/lldiv.3c target=div.3c
886 link path=usr/share/man/man3c/lltostr.3c target=strtol.3c
887 link path=usr/share/man/man3c/localtime.3c target=ctime.3c
888 link path=usr/share/man/man3c/localtime_r.3c target=ctime.3c
889 link path=usr/share/man/man3c/longjmp.3c target=setjmp.3c
890 link path=usr/share/man/man3c/lrand48.3c target=drand48.3c
891 link path=usr/share/man/man3c/major.3c target=makedev.3c
892 link path=usr/share/man/man3c/mblen_1.3c target=mblen.3c
893 link path=usr/share/man/man3c/mbrlen_1.3c target=mbrlen.3c
894 link path=usr/share/man/man3c/mbrtowc_1.3c target=mbrtowc.3c
895 link path=usr/share/man/man3c/mbsinit_1.3c target=mbsinit.3c
896 link path=usr/share/man/man3c/mbsrtowcs.3c target=mbsrtowcs.3c
897 link path=usr/share/man/man3c/mbsrtowcs_1.3c target=mbsrtowcs.3c
898 link path=usr/share/man/man3c/mbsrtowcs_1.3c target=mbsrtowcs.3c
899 link path=usr/share/man/man3c/mbstowcs.3c target=mbsrtowcs.3c
900 link path=usr/share/man/man3c/mbstowcs_1.3c target=mbsrtowcs.3c
901 link path=usr/share/man/man3c/mbtowc_1.3c target=mbtowc.3c
902 link path=usr/share/man/man3c/memalign.3c target=malloc.3c
903 link path=usr/share/man/man3c/membar_consumer.3c target=membar_ops.3c
904 link path=usr/share/man/man3c/membar_enter.3c target=membar_ops.3c
905 link path=usr/share/man/man3c/membar_exit.3c target=membar_ops.3c
906 link path=usr/share/man/man3c/membar_producer.3c target=membar_ops.3c
907 link path=usr/share/man/man3c/memccpy.3c target=memory.3c
908 link path=usr/share/man/man3c/memchr.3c target=memory.3c
909 link path=usr/share/man/man3c/memcmp.3c target=memory.3c
910 link path=usr/share/man/man3c/memcpy.3c target=memory.3c
911 link path=usr/share/man/man3c/memmove.3c target=memory.3c
912 link path=usr/share/man/man3c/memset.3c target=memory.3c
913 link path=usr/share/man/man3c/minor.3c target=makedev.3c
914 link path=usr/share/man/man3c/mkdstemp.3c target=mkstemp.3c
915 link path=usr/share/man/man3c/mkostemp.3c target=mkstemp.3c
916 link path=usr/share/man/man3c/mkostemps.3c target=mkstemp.3c

```



```

917 link path=usr/share/man/man3c/mkstemp.3c target=mkstemp.3c
918 link path=usr/share/man/man3c/mq_reltimedreceive_np.3c target=mq_receive.3c
919 link path=usr/share/man/man3c/mq_reltimedsend_np.3c target=mq_send.3c
920 link path=usr/share/man/man3c/mq_timedreceive.3c target=mq_receive.3c
921 link path=usr/share/man/man3c/mq_timedsend.3c target=mq_send.3c
922 link path=usr/share/man/man3c/mrand48.3c target=drand48.3c
923 link path=usr/share/man/man3c/munlock.3c target=mlock.3c
924 link path=usr/share/man/man3c/munlockall.3c target=mlockall.3c
925 link path=usr/share/man/man3c/mutex_consistent.3c target=mutex_init.3c
926 link path=usr/share/man/man3c/mutex_destroy.3c target=mutex_init.3c
927 link path=usr/share/man/man3c/mutex_lock.3c target=mutex_init.3c
928 link path=usr/share/man/man3c/mutex_trylock.3c target=mutex_init.3c
929 link path=usr/share/man/man3c/mutex_unlock.3c target=mutex_init.3c
930 link path=usr/share/man/man3c/nftw.3c target=ftw.3c
931 link path=usr/share/man/man3c/ngettext.3c target=gettext.3c
932 link path=usr/share/man/man3c/nl_langinfo.1.3c target=nl_langinfo.3c
933 link path=usr/share/man/man3c/nrand48.3c target=drand48.3c
934 link path=usr/share/man/man3c/openlog.3c target=syslog.3c
935 link path=usr/share/man/man3c/pclose.3c target=popen.3c
936 link path=usr/share/man/man3c/port_dissociate.3c target=port_associate.3c
937 link path=usr/share/man/man3c/port_getn.3c target=port_get.3c
938 link path=usr/share/man/man3c/port_sendn.3c target=port_send.3c
939 link path=usr/share/man/man3c/posix_spawn_file_actions_addopen.3c \
940 target=posix_spawn_file_actions_addclose.3c
941 link path=usr/share/man/man3c/posix_spawn_file_actions_init.3c \
942 target=posix_spawn_file_actions_destroy.3c
943 link path=usr/share/man/man3c/posix_spawnattr_init.3c \
944 target=posix_spawnattr_destroy.3c
945 link path=usr/share/man/man3c/posix_spawnattr_setflags.3c \
946 target=posix_spawnattr_getflags.3c
947 link path=usr/share/man/man3c/posix_spawnattr_setpgroup.3c \
948 target=posix_spawnattr_getpgroup.3c
949 link path=usr/share/man/man3c/posix_spawnattr_setschedparam.3c \
950 target=posix_spawnattr_getschedparam.3c
951 link path=usr/share/man/man3c/posix_spawnattr_setschedpolicy.3c \
952 target=posix_spawnattr_getschedpolicy.3c
953 link path=usr/share/man/man3c/posix_spawnattr_setsigdefault.3c \
954 target=posix_spawnattr_getsigdefault.3c
955 link path=usr/share/man/man3c/posix_spawnattr_setsignore_np.3c \
956 target=posix_spawnattr_getsignore_np.3c
957 link path=usr/share/man/man3c/posix_spawnattr_setsigmask.3c \
958 target=posix_spawnattr_getsigmask.3c
959 link path=usr/share/man/man3c/posix_spawnnp.3c target=posix_spawn.3c
960 link path=usr/share/man/man3c/printstack.3c target=walkcontext.3c
961 link path=usr/share/man/man3c/priv_allocset.3c target=priv_addset.3c
962 link path=usr/share/man/man3c/priv_basisset.3c target=priv_addset.3c
963 link path=usr/share/man/man3c/priv_copysset.3c target=priv_addset.3c
964 link path=usr/share/man/man3c/priv_delsset.3c target=priv_addset.3c
965 link path=usr/share/man/man3c/priv_emptyset.3c target=priv_addset.3c
966 link path=usr/share/man/man3c/priv_fillset.3c target=priv_addset.3c
967 link path=usr/share/man/man3c/priv_freerset.3c target=priv_addset.3c
968 link path=usr/share/man/man3c/priv_getbyname.3c target=priv_str_to_set.3c
969 link path=usr/share/man/man3c/priv_getbynum.3c target=priv_str_to_set.3c
970 link path=usr/share/man/man3c/priv_getsetbyname.3c target=priv_str_to_set.3c
971 link path=usr/share/man/man3c/priv_getsetbynum.3c target=priv_str_to_set.3c
972 link path=usr/share/man/man3c/priv_gettext.3c target=priv_str_to_set.3c
973 link path=usr/share/man/man3c/priv_ineffect.3c target=priv_set.3c
974 link path=usr/share/man/man3c/priv_intersect.3c target=priv_addset.3c
975 link path=usr/share/man/man3c/priv_inverse.3c target=priv_addset.3c
976 link path=usr/share/man/man3c/priv_ismemptyset.3c target=priv_addset.3c
977 link path=usr/share/man/man3c/priv_isequalset.3c target=priv_addset.3c
978 link path=usr/share/man/man3c/priv_isfullset.3c target=priv_addset.3c
979 link path=usr/share/man/man3c/priv_ismember.3c target=priv_addset.3c
980 link path=usr/share/man/man3c/priv_issubset.3c target=priv_addset.3c
981 link path=usr/share/man/man3c/priv_set_to_str.3c target=priv_str_to_set.3c
982 link path=usr/share/man/man3c/priv_union.3c target=priv_addset.3c

```

```

983 link path=usr/share/man/man3c/pselect.3c target=select.3c
984 link path=usr/share/man/man3c/psiginfo.3c target=psignal.3c
985 link path=usr/share/man/man3c/pthread_attr_destroy.3c \
986 target=pthread_attr_init.3c
987 link path=usr/share/man/man3c/pthread_attr_setdetachstate.3c \
988 target=pthread_attr_getdetachstate.3c
989 link path=usr/share/man/man3c/pthread_attr_setguardsize.3c \
990 target=pthread_attr_getguardsize.3c
991 link path=usr/share/man/man3c/pthread_attr_setinheritsched.3c \
992 target=pthread_attr_getinheritsched.3c
993 link path=usr/share/man/man3c/pthread_attr_setschedparam.3c \
994 target=pthread_attr_getschedparam.3c
995 link path=usr/share/man/man3c/pthread_attr_setschedpolicy.3c \
996 target=pthread_attr_getschedpolicy.3c
997 link path=usr/share/man/man3c/pthread_attr_setscope.3c \
998 target=pthread_attr_getscope.3c
999 link path=usr/share/man/man3c/pthread_attr_setstack.3c \
1000 target=pthread_attr_getstack.3c
1001 link path=usr/share/man/man3c/pthread_attr_setstackaddr.3c \
1002 target=pthread_attr_getstackaddr.3c
1003 link path=usr/share/man/man3c/pthread_attr_setstacksize.3c \
1004 target=pthread_attr_getstacksize.3c
1005 link path=usr/share/man/man3c/pthread_barrier_init.3c \
1006 target=pthread_barrier_destroy.3c
1007 link path=usr/share/man/man3c/pthread_barrierattr_init.3c \
1008 target=pthread_barrierattr_destroy.3c
1009 link path=usr/share/man/man3c/pthread_barrierattr_setpshared.3c \
1010 target=pthread_barrierattr_getpshared.3c
1011 link path=usr/share/man/man3c/pthread_cond_broadcast.3c \
1012 target=pthread_cond_signal.3c
1013 link path=usr/share/man/man3c/pthread_cond_destroy.3c \
1014 target=pthread_cond_init.3c
1015 link path=usr/share/man/man3c/pthread_cond_reltimedwait_np.3c \
1016 target=pthread_cond_wait.3c
1017 link path=usr/share/man/man3c/pthread_cond_timedwait.3c \
1018 target=pthread_cond_wait.3c
1019 link path=usr/share/man/man3c/pthread_condattr_destroy.3c \
1020 target=pthread_condattr_init.3c
1021 link path=usr/share/man/man3c/pthread_condattr_setclock.3c \
1022 target=pthread_condattr_getclock.3c
1023 link path=usr/share/man/man3c/pthread_condattr_setpshared.3c \
1024 target=pthread_condattr_getpshared.3c
1025 link path=usr/share/man/man3c/pthread_key_create_once_np.3c \
1026 target=pthread_key_create.3c
1027 link path=usr/share/man/man3c/pthread_mutex_destroy.3c \
1028 target=pthread_mutex_init.3c
1029 link path=usr/share/man/man3c/pthread_mutex_reltimedlock_np.3c \
1030 target=pthread_mutex_timedlock.3c
1031 link path=usr/share/man/man3c/pthread_mutex_setprioceiling.3c \
1032 target=pthread_mutex_getprioceiling.3c
1033 link path=usr/share/man/man3c/pthread_mutex_trylock.3c \
1034 target=pthread_mutex_lock.3c
1035 link path=usr/share/man/man3c/pthread_mutex_unlock.3c \
1036 target=pthread_mutex_lock.3c
1037 link path=usr/share/man/man3c/pthread_mutexattr_destroy.3c \
1038 target=pthread_mutexattr_init.3c
1039 link path=usr/share/man/man3c/pthread_mutexattr_setprioceiling.3c \
1040 target=pthread_mutexattr_getprioceiling.3c
1041 link path=usr/share/man/man3c/pthread_mutexattr_setprotocol.3c \
1042 target=pthread_mutexattr_getprotocol.3c
1043 link path=usr/share/man/man3c/pthread_mutexattr_setpshared.3c \
1044 target=pthread_mutexattr_getpshared.3c
1045 link path=usr/share/man/man3c/pthread_mutexattr_settype.3c \
1046 target=pthread_mutexattr_gettype.3c
1047 link path=usr/share/man/man3c/pthread_rwlock_destroy.3c \
1048 target=pthread_rwlock_init.3c

```

```

1049 link path=usr/share/man/man3c/pthread_rwlock_reltimedrdlock_np.3c \
1050 target=pthread_rwlock_timedrdlock.3c
1051 link path=usr/share/man/man3c/pthread_rwlock_reltimedwrllock_np.3c \
1052 target=pthread_rwlock_timedwrllock.3c
1053 link path=usr/share/man/man3c/pthread_rwlock_tryrdlock.3c \
1054 target=pthread_rwlock_rdlock.3c
1055 link path=usr/share/man/man3c/pthread_rwlock_trywrllock.3c \
1056 target=pthread_rwlock_wrlock.3c
1057 link path=usr/share/man/man3c/pthread_rwlockattr_destroy.3c \
1058 target=pthread_rwlockattr_init.3c
1059 link path=usr/share/man/man3c/pthread_rwlockattr_setpshared.3c \
1060 target=pthread_rwlockattr_getpshared.3c
1061 link path=usr/share/man/man3c/pthread_setconcurrency.3c \
1062 target=pthread_getconcurrency.3c
1063 link path=usr/share/man/man3c/pthread_setschedparam.3c \
1064 target=pthread_getschedparam.3c
1065 link path=usr/share/man/man3c/pthread_setspecific.3c \
1066 target=pthread_getspecific.3c
1067 link path=usr/share/man/man3c/pthread_spin_init.3c \
1068 target=pthread_spin_destroy.3c
1069 link path=usr/share/man/man3c/pthread_spin_trylock.3c \
1070 target=pthread_spin_lock.3c
1071 link path=usr/share/man/man3c/putc.3c target=fputc.3c
1072 link path=usr/share/man/man3c/putc_unlocked.3c target=fputc.3c
1073 link path=usr/share/man/man3c/putchar.3c target=fputc.3c
1074 link path=usr/share/man/man3c/putchar_unlocked.3c target=fputc.3c
1075 link path=usr/share/man/man3c/putmntent.3c target=getmntent.3c
1076 link path=usr/share/man/man3c/pututline.3c target=getutent.3c
1077 link path=usr/share/man/man3c/pututxline.3c target=getutxent.3c
1078 link path=usr/share/man/man3c/putw.3c target=fputc.3c
1079 link path=usr/share/man/man3c/putwc.3c target=fputwc.3c
1080 link path=usr/share/man/man3c/putwchar.3c target=fputwc.3c
1081 link path=usr/share/man/man3c/qeconvert.3c target=econvert.3c
1082 link path=usr/share/man/man3c/qfconvert.3c target=econvert.3c
1083 link path=usr/share/man/man3c/qgconvert.3c target=econvert.3c
1084 link path=usr/share/man/man3c/quadruple_to_decimal.3c \
1085 target=floating_to_decimal.3c
1086 link path=usr/share/man/man3c/rand_r.3c target=rand.3c
1087 link path=usr/share/man/man3c/rctlblk_get_enforced_value.3c \
1088 target=rctlblk_set_value.3c
1089 link path=usr/share/man/man3c/rctlblk_get_firing_time.3c \
1090 target=rctlblk_set_value.3c
1091 link path=usr/share/man/man3c/rctlblk_get_global_action.3c \
1092 target=rctlblk_set_value.3c
1093 link path=usr/share/man/man3c/rctlblk_get_global_flags.3c \
1094 target=rctlblk_set_value.3c
1095 link path=usr/share/man/man3c/rctlblk_get_local_action.3c \
1096 target=rctlblk_set_value.3c
1097 link path=usr/share/man/man3c/rctlblk_get_local_flags.3c \
1098 target=rctlblk_set_value.3c
1099 link path=usr/share/man/man3c/rctlblk_get_privilege.3c \
1100 target=rctlblk_set_value.3c
1101 link path=usr/share/man/man3c/rctlblk_get_recipient_pid.3c \
1102 target=rctlblk_set_value.3c
1103 link path=usr/share/man/man3c/rctlblk_get_value.3c target=rctlblk_set_value.3c
1104 link path=usr/share/man/man3c/rctlblk_set_local_action.3c \
1105 target=rctlblk_set_value.3c
1106 link path=usr/share/man/man3c/rctlblk_set_local_flags.3c \
1107 target=rctlblk_set_value.3c
1108 link path=usr/share/man/man3c/rctlblk_set_privilege.3c \
1109 target=rctlblk_set_value.3c
1110 link path=usr/share/man/man3c/rctlblk_set_recipient_pid.3c \
1111 target=rctlblk_set_value.3c
1112 link path=usr/share/man/man3c/rctlblk_size.3c target=rctlblk_set_value.3c
1113 link path=usr/share/man/man3c/re_exec.3c target=re_comp.3c
1114 link path=usr/share/man/man3c/readdir_r.3c target=readdir.3c

```

```

1115 link path=usr/share/man/man3c/realloc.3c target=malloc.3c
1116 link path=usr/share/man/man3c/regerror.3c target=regcomp.3c
1117 link path=usr/share/man/man3c/regex.3c target=regcomp.3c
1118 link path=usr/share/man/man3c/regexec.3c target=regcomp.3c
1119 link path=usr/share/man/man3c/regfree.3c target=regcomp.3c
1120 link path=usr/share/man/man3c/remque.3c target=insque.3c
1121 link path=usr/share/man/man3c/resetmnttab.3c target=getmntent.3c
1122 link path=usr/share/man/man3c/rindex.3c target=index.3c
1123 link path=usr/share/man/man3c/rw_rdlock.3c target=rwlock.3c
1124 link path=usr/share/man/man3c/rw_tryrdlock.3c target=rwlock.3c
1125 link path=usr/share/man/man3c/rw_trywrllock.3c target=rwlock.3c
1126 link path=usr/share/man/man3c/rw_unlock.3c target=rwlock.3c
1127 link path=usr/share/man/man3c/rw_wrllock.3c target=rwlock.3c
1128 link path=usr/share/man/man3c/rwlock_destroy.3c target=rwlock.3c
1129 link path=usr/share/man/man3c/rwlock_init.3c target=rwlock.3c
1130 link path=usr/share/man/man3c/sched_get_priority_min.3c \
1131 target=sched_get_priority_max.3c
1132 link path=usr/share/man/man3c/schedctl_exit.3c target=schedctl_init.3c
1133 link path=usr/share/man/man3c/schedctl_lookup.3c target=schedctl_init.3c
1134 link path=usr/share/man/man3c/schedctl_start.3c target=schedctl_init.3c
1135 link path=usr/share/man/man3c/schedctl_stop.3c target=schedctl_init.3c
1136 link path=usr/share/man/man3c/seconvert.3c target=econvert.3c
1137 link path=usr/share/man/man3c/seed48.3c target=drand48.3c
1138 link path=usr/share/man/man3c/sem_reltimedwait_np.3c target=sem_timedwait.3c
1139 link path=usr/share/man/man3c/sem_trywait.3c target=sem_wait.3c
1140 link path=usr/share/man/man3c/sema_destroy.3c target=semaphore.3c
1141 link path=usr/share/man/man3c/sema_init.3c target=semaphore.3c
1142 link path=usr/share/man/man3c/sema_post.3c target=semaphore.3c
1143 link path=usr/share/man/man3c/sema_trywait.3c target=semaphore.3c
1144 link path=usr/share/man/man3c/sema_wait.3c target=semaphore.3c
1145 link path=usr/share/man/man3c/setattrat.3c target=fgetat.3c
1146 link path=usr/share/man/man3c/setgrent.3c target=getgnam.3c
1147 link path=usr/share/man/man3c/sethostname.3c target=gethostname.3c
1148 link path=usr/share/man/man3c/setlinebuf.3c target=setbuffer.3c
1149 link path=usr/share/man/man3c/setlogmask.3c target=syslog.3c
1150 link path=usr/share/man/man3c/setnetgrent.3c target=getnetgrent.3c
1151 link path=usr/share/man/man3c/setpriority.3c target=getpriority.3c
1152 link path=usr/share/man/man3c/setpwent.3c target=getpwnam.3c
1153 link path=usr/share/man/man3c/setsptent.3c target=getspnam.3c
1154 link path=usr/share/man/man3c/setstate.3c target=random.3c
1155 link path=usr/share/man/man3c/settimeofday.3c target=gettimeofday.3c
1156 link path=usr/share/man/man3c/setusershell.3c target=getusershell.3c
1157 link path=usr/share/man/man3c/setutent.3c target=getutent.3c
1158 link path=usr/share/man/man3c/setutxent.3c target=getutxent.3c
1159 link path=usr/share/man/man3c/setvbuf.3c target=setbuf.3c
1160 link path=usr/share/man/man3c/sfconvert.3c target=econvert.3c
1161 link path=usr/share/man/man3c/sgconvert.3c target=econvert.3c
1162 link path=usr/share/man/man3c/sig2str.3c target=str2sig.3c
1163 link path=usr/share/man/man3c/sigaddset.3c target=sigsetops.3c
1164 link path=usr/share/man/man3c/sigdelset.3c target=sigsetops.3c
1165 link path=usr/share/man/man3c/sigemptyset.3c target=sigsetops.3c
1166 link path=usr/share/man/man3c/sigfillset.3c target=sigsetops.3c
1167 link path=usr/share/man/man3c/sighold.3c target=signal.3c
1168 link path=usr/share/man/man3c/sigignore.3c target=signal.3c
1169 link path=usr/share/man/man3c/sigismember.3c target=sigsetops.3c
1170 link path=usr/share/man/man3c/siglongjmp.3c target=setjmp.3c
1171 link path=usr/share/man/man3c/sigpause.3c target=signal.3c
1172 link path=usr/share/man/man3c/sigrelse.3c target=signal.3c
1173 link path=usr/share/man/man3c/sigset.3c target=signal.3c
1174 link path=usr/share/man/man3c/sigsetjmp.3c target=setjmp.3c
1175 link path=usr/share/man/man3c/sigtimedwait.3c target=sigwaitinfo.3c
1176 link path=usr/share/man/man3c/single_to_decimal.3c \
1177 target=floating_to_decimal.3c
1178 link path=usr/share/man/man3c/snprintf.3c target=printf.3c
1179 link path=usr/share/man/man3c/sprintf.3c target=printf.3c
1180 link path=usr/share/man/man3c/srand.3c target=rand.3c

```

```

1181 link path=usr/share/man/man3c/srand48.3c target=drand48.3c
1182 link path=usr/share/man/man3c/srandom.3c target=random.3c
1183 link path=usr/share/man/man3c/sscanf.3c target=scanf.3c
1184 link path=usr/share/man/man3c/stderr.3c target=stdio.3c
1185 link path=usr/share/man/man3c/stdin.3c target=stdio.3c
1186 link path=usr/share/man/man3c/stdout.3c target=stdio.3c
1187 link path=usr/share/man/man3c/strcasecmp.3c target=string.3c
1188 link path=usr/share/man/man3c/strcasecmp.1.3c target=string.3c
1189 link path=usr/share/man/man3c/strcat.3c target=string.3c
1190 link path=usr/share/man/man3c/strchr.3c target=string.3c
1191 link path=usr/share/man/man3c/strcmp.3c target=string.3c
1192 link path=usr/share/man/man3c/strcoll.1.3c target=string.3c
1193 link path=usr/share/man/man3c/strcpy.3c target=string.3c
1194 link path=usr/share/man/man3c/strcsn.3c target=string.3c
1195 link path=usr/share/man/man3c/strdup.3c target=string.3c
1196 link path=usr/share/man/man3c/strfmon.1.3c target=string.3c
1197 link path=usr/share/man/man3c/strftime.1.3c target=string.3c
1198 link path=usr/share/man/man3c/strerror_r.3c target=string.3c
1199 link path=usr/share/man/man3c/strlcat.3c target=string.3c
1200 link path=usr/share/man/man3c/strncpy.3c target=string.3c
1201 link path=usr/share/man/man3c/strlen.3c target=string.3c
1202 link path=usr/share/man/man3c/strncasecmp.3c target=string.3c
1203 link path=usr/share/man/man3c/strncasecmp.1.3c target=string.3c
1204 link path=usr/share/man/man3c/strncat.3c target=string.3c
1205 link path=usr/share/man/man3c/strncmp.3c target=string.3c
1206 link path=usr/share/man/man3c/strncpy.3c target=string.3c
1207 link path=usr/share/man/man3c/strnlen.3c target=string.3c
1208 link path=usr/share/man/man3c/strpbrk.3c target=string.3c
1209 link path=usr/share/man/man3c/strptime.1.3c target=string.3c
1210 link path=usr/share/man/man3c/strrchr.3c target=string.3c
1211 link path=usr/share/man/man3c/strsep.3c target=string.3c
1212 link path=usr/share/man/man3c/strspn.3c target=string.3c
1213 link path=usr/share/man/man3c/strstr.3c target=string.3c
1214 link path=usr/share/man/man3c/strtof.3c target=string.3c
1215 link path=usr/share/man/man3c/strtok.3c target=string.3c
1216 link path=usr/share/man/man3c/strtok_r.3c target=string.3c
1217 link path=usr/share/man/man3c/strtol.3c target=string.3c
1218 link path=usr/share/man/man3c/strtoll.3c target=string.3c
1219 link path=usr/share/man/man3c/strtoul.3c target=string.3c
1220 link path=usr/share/man/man3c/strtoumax.3c target=string.3c
1221 link path=usr/share/man/man3c/strxfrm.1.3c target=string.3c
1222 link path=usr/share/man/man3c/swapcontext.3c target=string.3c
1223 link path=usr/share/man/man3c/swprintf.3c target=string.3c
1224 link path=usr/share/man/man3c/swscanf.3c target=string.3c
1225 link path=usr/share/man/man3c/tdelete.3c target=string.3c
1226 link path=usr/share/man/man3c/tempnam.3c target=string.3c
1227 link path=usr/share/man/man3c/textdomain.3c target=string.3c
1228 link path=usr/share/man/man3c/tfind.3c target=string.3c
1229 link path=usr/share/man/man3c/thr_continue.3c target=string.3c
1230 link path=usr/share/man/man3c/thr_getspecific.3c target=string.3c
1231 link path=usr/share/man/man3c/thr_keycreate_once.3c target=string.3c
1232 link path=usr/share/man/man3c/thr_setconcurrency.3c \
1233 target=string.3c
1234 link path=usr/share/man/man3c/thr_setprio.3c target=string.3c
1235 link path=usr/share/man/man3c/thr_setspecific.3c target=string.3c
1236 link path=usr/share/man/man3c/timer_getoverrun.3c target=string.3c
1237 link path=usr/share/man/man3c/timer_gettime.3c target=string.3c
1238 link path=usr/share/man/man3c/timerclear.3c target=string.3c
1239 link path=usr/share/man/man3c/timercmp.3c target=string.3c
1240 link path=usr/share/man/man3c/timerisset.3c target=string.3c
1241 link path=usr/share/man/man3c/timersub.3c target=string.3c
1242 link path=usr/share/man/man3c/tmpnam_r.3c target=string.3c
1243 link path=usr/share/man/man3c/tolower.1.3c target=string.3c
1244 link path=usr/share/man/man3c/toupper.1.3c target=string.3c
1245 link path=usr/share/man/man3c/towctrans.3c target=string.3c
1246 link path=usr/share/man/man3c/towctrans.1.3c target=string.3c

```

```

1247 link path=usr/share/man/man3c/towlower.1.3c target=string.3c
1248 link path=usr/share/man/man3c/toupper.1.3c target=string.3c
1249 link path=usr/share/man/man3c/ttyname_r.3c target=string.3c
1250 link path=usr/share/man/man3c/twalk.3c target=string.3c
1251 link path=usr/share/man/man3c/tzset.3c target=string.3c
1252 link path=usr/share/man/man3c/uconv_ul6tou8.3c target=string.3c
1253 link path=usr/share/man/man3c/uconv_u32tou16.3c target=string.3c
1254 link path=usr/share/man/man3c/uconv_u32tou8.3c target=string.3c
1255 link path=usr/share/man/man3c/uconv_u8tou16.3c target=string.3c
1256 link path=usr/share/man/man3c/uconv_u8tou32.3c target=string.3c
1257 link path=usr/share/man/man3c/ucrd_free.3c target=string.3c
1258 link path=usr/share/man/man3c/ucrd_getegid.3c target=string.3c
1259 link path=usr/share/man/man3c/ucrd_geteuid.3c target=string.3c
1260 link path=usr/share/man/man3c/ucrd_getgroups.3c target=string.3c
1261 link path=usr/share/man/man3c/ucrd_getlabel.3c target=string.3c
1262 link path=usr/share/man/man3c/ucrd_getpflags.3c target=string.3c
1263 link path=usr/share/man/man3c/ucrd_getpid.3c target=string.3c
1264 link path=usr/share/man/man3c/ucrd_getprivset.3c target=string.3c
1265 link path=usr/share/man/man3c/ucrd_getprojid.3c target=string.3c
1266 link path=usr/share/man/man3c/ucrd_getruid.3c target=string.3c
1267 link path=usr/share/man/man3c/ucrd_getuid.3c target=string.3c
1268 link path=usr/share/man/man3c/ucrd_getsuid.3c target=string.3c
1269 link path=usr/share/man/man3c/ucrd_getzuid.3c target=string.3c
1270 link path=usr/share/man/man3c/ucrd_getzoneid.3c target=string.3c
1271 link path=usr/share/man/man3c/ucrd_size.3c target=string.3c
1272 link path=usr/share/man/man3c/ulckpwwdf.3c target=string.3c
1273 link path=usr/share/man/man3c/ulltostr.3c target=string.3c
1274 link path=usr/share/man/man3c/unordered.3c target=string.3c
1275 link path=usr/share/man/man3c/updwtmp.3c target=string.3c
1276 link path=usr/share/man/man3c/updwtmpx.3c target=string.3c
1277 link path=usr/share/man/man3c/utmpname.3c target=string.3c
1278 link path=usr/share/man/man3c/utmpxname.3c target=string.3c
1279 link path=usr/share/man/man3c/valloc.3c target=string.3c
1280 link path=usr/share/man/man3c/vasprintf.3c target=string.3c
1281 link path=usr/share/man/man3c/verr.3c target=string.3c
1282 link path=usr/share/man/man3c/verrx.3c target=string.3c
1283 link path=usr/share/man/man3c/vfprintf.3c target=string.3c
1284 link path=usr/share/man/man3c/vfscanf.3c target=string.3c
1285 link path=usr/share/man/man3c/vfwscanf.3c target=string.3c
1286 link path=usr/share/man/man3c/vscanf.3c target=string.3c
1287 link path=usr/share/man/man3c/vsnprintf.3c target=string.3c
1288 link path=usr/share/man/man3c/vsprintf.3c target=string.3c
1289 link path=usr/share/man/man3c/vsscanf.3c target=string.3c
1290 link path=usr/share/man/man3c/vswprintf.3c target=string.3c
1291 link path=usr/share/man/man3c/vswscanf.3c target=string.3c
1292 link path=usr/share/man/man3c/vwarn.3c target=string.3c
1293 link path=usr/share/man/man3c/vwarnx.3c target=string.3c
1294 link path=usr/share/man/man3c/vwprintf.3c target=string.3c
1295 link path=usr/share/man/man3c/vwscanf.3c target=string.3c
1296 link path=usr/share/man/man3c/wait4.3c target=string.3c
1297 link path=usr/share/man/man3c/warn.3c target=string.3c
1298 link path=usr/share/man/man3c/warnx.3c target=string.3c
1299 link path=usr/share/man/man3c/watof.3c target=string.3c
1300 link path=usr/share/man/man3c/watoi.3c target=string.3c
1301 link path=usr/share/man/man3c/watol.3c target=string.3c
1302 link path=usr/share/man/man3c/watoll.3c target=string.3c
1303 link path=usr/share/man/man3c/wcrtomb.1.3c target=string.3c
1304 link path=usr/share/man/man3c/wcscat.3c target=string.3c
1305 link path=usr/share/man/man3c/wcschr.3c target=string.3c
1306 link path=usr/share/man/man3c/wcscmp.3c target=string.3c
1307 link path=usr/share/man/man3c/wcscoll.1.3c target=string.3c
1308 link path=usr/share/man/man3c/wcscpy.3c target=string.3c
1309 link path=usr/share/man/man3c/wcscspn.3c target=string.3c
1310 link path=usr/share/man/man3c/wcsetno.3c target=string.3c
1311 link path=usr/share/man/man3c/wcslen.3c target=string.3c
1312 link path=usr/share/man/man3c/wcsncaat.3c target=string.3c

```

```
1313 link path=usr/share/man/man3c/wcsncmp.3c target=wcstring.3c
1314 link path=usr/share/man/man3c/wcsncpy.3c target=wcstring.3c
1315 link path=usr/share/man/man3c/wcsprbrk.3c target=wcstring.3c
1316 link path=usr/share/man/man3c/wcsrchr.3c target=wcstring.3c
1317 link path=usr/share/man/man3c/wcsspn.3c target=wcstring.3c
1318 link path=usr/share/man/man3c/wcstof.3c target=wcstod.3c
1319 link path=usr/share/man/man3c/wcstok.3c target=wcstring.3c
1320 link path=usr/share/man/man3c/wcstold.3c target=wcstod.3c
1321 link path=usr/share/man/man3c/wcstoll.3c target=wcstol.3c
1322 link path=usr/share/man/man3c/wcstoull.3c target=wcstoul.3c
1323 link path=usr/share/man/man3c/wcstoumax.3c target=wcstoimax.3c
1324 link path=usr/share/man/man3c/wcswcs.3c target=wcstring.3c
1325 link path=usr/share/man/man3c/wcswidth.1.3c target=wcswidth.3c
1326 link path=usr/share/man/man3c/wctob.1.3c target=wctob.3c
1327 link path=usr/share/man/man3c/wctomb.1.3c target=wctomb.3c
1328 link path=usr/share/man/man3c/wctrans.1.3c target=wctrans.3c
1329 link path=usr/share/man/man3c/wctype.1.3c target=wctype.3c
1330 link path=usr/share/man/man3c/wcwidth.1.3c target=wcwidth.3c
1331 link path=usr/share/man/man3c/windex.3c target=wcstring.3c
1332 link path=usr/share/man/man3c/wordfree.3c target=wordexp.3c
1333 link path=usr/share/man/man3c/wprintf.3c target=fwprintf.3c
1334 link path=usr/share/man/man3c/wrindex.3c target=wcstring.3c
1335 link path=usr/share/man/man3c/wscanf.3c target=fwscanf.3c
1336 link path=usr/share/man/man3c/wscasecmp.3c target=wstring.3c
1337 link path=usr/share/man/man3c/wscat.3c target=wcstring.3c
1338 link path=usr/share/man/man3c/wschr.3c target=wcstring.3c
1339 link path=usr/share/man/man3c/wscmp.3c target=wcstring.3c
1340 link path=usr/share/man/man3c/wscol.3c target=wstring.3c
1341 link path=usr/share/man/man3c/wscoll.3c target=wcscoll.3c
1342 link path=usr/share/man/man3c/wscopy.3c target=wcstring.3c
1343 link path=usr/share/man/man3c/wscspn.3c target=wcstring.3c
1344 link path=usr/share/man/man3c/wsdup.3c target=wstring.3c
1345 link path=usr/share/man/man3c/wslen.3c target=wcstring.3c
1346 link path=usr/share/man/man3c/wsncasecmp.3c target=wstring.3c
1347 link path=usr/share/man/man3c/wsncat.3c target=wcstring.3c
1348 link path=usr/share/man/man3c/wsncmp.3c target=wcstring.3c
1349 link path=usr/share/man/man3c/wsncpy.3c target=wcstring.3c
1350 link path=usr/share/man/man3c/wspbrk.3c target=wcstring.3c
1351 link path=usr/share/man/man3c/wsrchr.3c target=wcstring.3c
1352 link path=usr/share/man/man3c/wsspn.3c target=wcstring.3c
1353 link path=usr/share/man/man3c/wstod.3c target=wcstod.3c
1354 link path=usr/share/man/man3c/wstok.3c target=wcstring.3c
1355 link path=usr/share/man/man3c/wstol.3c target=wcstol.3c
1356 link path=usr/share/man/man3c/wstostr.3c target=strows.3c
1357 link path=usr/share/man/man3c/wsxfm.3c target=wcsxfm.3c
```

new/usr/src/pkg/manifests/system-test-libctest.mf

1

\*\*\*\*\*

1423 Fri Jul 11 11:43:13 2014

new/usr/src/pkg/manifests/system-test-libctest.mf

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014, OmniTI Computer Consulting, Inc. All rights reserved.
15 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
16 #
17 #
18 set name=pkg.fmri value=pkg:/system/test/libctest@$(PKGVERS)
19 set name=pkg.description value="C library Unit Tests"
20 set name=pkg.summary value="C Library Unit Test Suite"
21 set name=info.classification \
22     value=org.opensolaris.category.2008:Development/System
23 set name=variant.arch value=$(ARCH)
24 dir path=opt/libc-tests
25 dir path=opt/libc-tests/bin
26 dir path=opt/libc-tests/runfiles
27 dir path=opt/libc-tests/tests
28 file path=opt/libc-tests/README mode=0444
29 file path=opt/libc-tests/bin/libctest mode=0555
30 file path=opt/libc-tests/runfiles/default.run mode=0444
31 file path=opt/libc-tests/tests/newlocale_test mode=0555
32 license lic_CDDL license=lic_CDDL
33 depend fmri=locale/de type=require
34 depend fmri=locale/en type=require
35 depend fmri=locale/ja type=require
36 depend fmri=locale/ru type=require
37 depend fmri=system/test/testrunner type=require
```

new/usr/src/test/Makefile

1

```
*****
624 Fri Jul 11 11:43:13 2014
new/usr/src/test/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #
16 #
17 .PARALLEL: $(SUBDIRS)
18 #
19 SUBDIRS = libc-tests os-tests test-runner util-tests zfs-tests
20 SUBDIRS = os-tests test-runner util-tests zfs-tests
21 include Makefile.com
```

new/usr/src/test/libc-tests/Makefile

1

```
*****
606 Fri Jul 11 11:43:13 2014
new/usr/src/test/libc-tests/Makefile
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****
```

```
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #
16 #
17 .PARALLEL: $(SUBDIRS)
18 #
19 SUBDIRS = cmd runfiles tests doc
20 #
21 include $(SRC)/test/Makefile.com
```

new/usr/src/test/libc-tests/cmd/Makefile

1

```
*****  
      847 Fri Jul 11 11:43:13 2014  
new/usr/src/test/libc-tests/cmd/Makefile  
2964 need POSIX 2008 locale object support  
Reviewed by: Robert Mustacchi <rm@joyent.com>  
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>  
Approved by: TBD  
*****
```

```
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 2014 Garrett D'Amore <garrett@damore.org>  
14 # Copyright (c) 2012 by Delphix. All rights reserved.  
15 #  
  
17 include $(SRC)/Makefile.master  
18 include $(SRC)/test/Makefile.com  
  
20 ROOTOPTPKG = $(ROOT)/opt/libc-tests  
21 ROOTBIN = $(ROOTOPTPKG)/bin  
  
23 PROGS = libctest  
  
25 CMDS = $(PROGS:%=$(ROOTBIN)/%)  
26 $(CMDS) := FILEMODE = 0555  
  
28 all lint clean clobber:  
  
30 install: $(CMDS)  
  
32 $(CMDS): $(ROOTBIN)  
  
34 $(ROOTBIN):  
35     $(INS.dir)  
  
37 $(ROOTBIN)/%: %.ksh  
38     $(INS.rename)
```



new/usr/src/test/libc-tests/cmd/libctest.ksh

1

\*\*\*\*\*

```
1480 Fri Jul 11 11:43:13 2014
new/usr/src/test/libc-tests/cmd/libctest.ksh
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
```

\*\*\*\*\*

```
1 #!/usr/bin/ksh
3 #
4 # This file and its contents are supplied under the terms of the
5 # Common Development and Distribution License ("CDDL"), version 1.0.
6 # You may only use this file in accordance with the terms of version
7 # 1.0 of the CDDL.
8 #
9 # A full copy of the text of the CDDL should have accompanied this
10 # source. A copy of the CDDL is also available via the Internet at
11 # http://www.illumos.org/license/CDDL.
12 #
14 #
15 # Copyright (c) 2012 by Delphix. All rights reserved.
16 # Copyright 2014, OmniTI Computer Consulting, Inc. All rights reserved.
17 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
18 #
20 export MY_TESTS="/opt/libc-tests"
21 runner="/opt/test-runner/bin/run"
23 function fail
24 {
25     echo $1
26     exit ${2:-1}
27 }
29 function find_runfile
30 {
31     typeset distro=
32     if [[ -d /opt/delphix && -h /etc/delphix/version ]]; then
33         distro=delphix
34     elif [[ 0 -ne $(grep -c OpenIndiana /etc/release 2>/dev/null) ]]; then
35         distro=openindiana
36     elif [[ 0 -ne $(grep -c OmniOS /etc/release 2>/dev/null) ]]; then
37         distro=omnios
38     fi
39
40     if [[ ! -f $MY_TESTS/runfiles/$distro.run ]] && \
41         [[ -f $MY_TESTS/runfiles/default.run ]]; then
42         distro=default
43     fi
44
45     [[ -n $distro ]] && echo $MY_TESTS/runfiles/$distro.run
46 }
48 while getopts c: do
49     case $c in
50         'c')
51             runfile=$OPTARG
52             [[ -f $runfile ]] || fail "Cannot read file: $runfile"
53             ;;
54         esac
55     done
56     shift $((OPTIND - 1))
58 [[ -z $runfile ]] && runfile=$(find_runfile)
```

new/usr/src/test/libc-tests/cmd/libctest.ksh

2

```
59 [[ -z $runfile ]] && fail "Couldn't determine distro"
61 $runner -c $runfile
63 exit $?
```

new/usr/src/test/libc-tests/doc/Makefile

1

\*\*\*\*\*

800 Fri Jul 11 11:43:14 2014

new/usr/src/test/libc-tests/doc/Makefile

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 2014 Garrett D'Amore <garrett@damore.org>
14 # Copyright (c) 2012 by Delphix. All rights reserved.
15 #
```

```
17 include $(SRC)/Makefile.master
```

```
19 README = README
```

```
21 ROOTOPTPKG = $(ROOT)/opt/libc-tests
```

```
23 FILES = $(README:%=$(ROOTOPTPKG)/%)
```

```
24 $(FILES) := FILEMODE = 0444
```

```
26 all: $(README)
```

```
28 install: $(ROOTOPTPKG) $(FILES)
```

```
30 clean lint clobber:
```

```
32 $(ROOTOPTPKG):
```

```
33     $(INS.dir)
```

```
35 $(ROOTOPTPKG)/%: %
```

```
36     $(INS.file)
```

```

*****
2351 Fri Jul 11 11:43:14 2014
new/usr/src/test/libc-tests/doc/README
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
Reviewed by: Gordon Ross <gordon.ross@nexenta.com>
Approved by: TBD
*****

```

```

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 #
11 #
12 #
13 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
14 # Copyright (c) 2012 by Delphix. All rights reserved.
15 #
16 #
17 libc Unit Test Suite README
18 #
19 1. What this Unit Test Suite tests
20 2. Building and installing this Unit Test Suite
21 3. Running this Unit Test Suite
22 4. Test results
23 #
24 -----
25 #
26 1. What this Unit Test Suite tests
27 #
28 This Unit Test Suite is for testing various libc interfaces.
29 #
30 2. Building and installing this Unit Test Suite
31 #
32 This Test Suite runs under the testrunner framework (which can be installed
33 as pkg:/system/test/testrunner). To build both this Unit Test Suite and the
34 testrunner without running a full nightly:
35 #
36     build_machine$ bldenv [-d] <your_env_file>
37     build_machine$ cd $SRC/test
38     build_machine$ dmake install
39     build_machine$ cd $SRC/pkg
40     build_machine$ dmake install
41 #
42 Then set the publisher on the test machine to point to your repository and
43 install the Utils Unit Test Suite.
44 #
45     test_machine# pkg install pkg:/system/test/libctest
46 #
47 Note, the framework will be installed automatically, as this test suite
48 depends on it.
49 #
50 Additionally some text locales are required, specifically, en_US.UTF-8,
51 de_DE.UTF-8, ja_JP.UTF-8, and ru_RU.UTF-8. Again, these are listed as
52 dependencies and will be automatically installed.
53 #
54 3. Running this Unit Test Suite
55 #
56 The pre-requisites for running the this Unit Test Suite are:
57 - Any user may perform these tests.
58 - The en_US.UTF-8, ja_JP.UTF-8, de_DE.UTF-8, and ru_RU.UTF-8 locales

```

```

59     must be installed.
60 #
61 Once the pre-requisites are satisfied, simply run the ostest script:
62 #
63     test_machine$ /opt/util-tests/bin/libctest
64 #
65 4. Test results
66 #
67 While the Unit Test Suite is running, one informational line is printed at
68 the end of each test, and a results summary is printed at the end of the run.
69 The results summary includes the location of the complete logs, which is of the
70 form /var/tmp/test_results/<ISO 8601 date>.

```

new/usr/src/test/libc-tests/runfiles/Makefile

1

\*\*\*\*\*

909 Fri Jul 11 11:43:14 2014

new/usr/src/test/libc-tests/runfiles/Makefile

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014, OmniTI Computer Consulting, Inc. All rights reserved.
15 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
16 #
17 #
18 include $(SRC)/Makefile.master
19 #
20 SRCS = default.run
21 #
22 ROOTOPTPKG = $(ROOT)/opt/libc-tests
23 RUNFILES = $(ROOTOPTPKG)/runfiles
24 #
25 CMDS = $(SRCS:%=$(RUNFILES)/%)
26 $(CMDS) := FILEMODE = 0444
27 #
28 all: $(SRCS)
29 #
30 install: $(CMDS)
31 #
32 clean lint clobber:
33 #
34 $(CMDS): $(RUNFILES) $(SRCS)
35 #
36 $(RUNFILES):
37     $(INS.dir)
38 #
39 $(RUNFILES)/%: %
40     $(INS.file)
```

new/usr/src/test/libc-tests/runfiles/default.run

1

\*\*\*\*\*

656 Fri Jul 11 11:43:14 2014

new/usr/src/test/libc-tests/runfiles/default.run

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #
```

```
17 [DEFAULT]
18 pre =
19 verbose = False
20 quiet = False
21 timeout = 60
22 post =
23 outputdir = /var/tmp/test_results

25 [/opt/libc-tests/tests/newlocale_test]
```

new/usr/src/test/libc-tests/tests/Makefile

1

\*\*\*\*\*

570 Fri Jul 11 11:43:14 2014

new/usr/src/test/libc-tests/tests/Makefile

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #
16 #
17 SUBDIRS = newlocale
18 #
19 include $(SRC)/test/Makefile.com
```

new/usr/src/test/libc-tests/tests/newlocale/Makefile

1

\*\*\*\*\*

1088 Fri Jul 11 11:43:14 2014

new/usr/src/test/libc-tests/tests/newlocale/Makefile

2964 need POSIX 2008 locale object support

Reviewed by: Robert Mustacchi <rm@joyent.com>

Reviewed by: Gordon Ross <gordon.ross@nexenta.com>

Approved by: TBD

\*\*\*\*\*

```
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 #
11 #
12 #
13 # Copyright (c) 2012 by Delphix. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #
16 #
17 include $(SRC)/cmd/Makefile.cmd
18 include $(SRC)/test/Makefile.com
19 #
20 PROG = newlocale_test
21 OBJS = $(PROG:%=%.o)
22 SRCS = $(OBJS:%.o=%.c)
23 #
24 C99MODE = -xc99=%all
25 #
26 ROOTOPTPKG = $(ROOT)/opt/libc-tests
27 TESTDIR = $(ROOTOPTPKG)/tests
28 #
29 CMDS = $(PROG:%=$(TESTDIR)/%)
30 $(CMDS) := FILEMODE = 0555
31 #
32 all: $(PROG)
33 #
34 $(PROG): $(OBJS)
35     $(LINK.c) $(OBJS) -o $@ $(LDLIBS)
36     $(POST_PROCESS)
37 #
38 %.o: ../%.c
39     $(COMPILE.c) $<
40 #
41 install: all $(CMDS)
42 #
43 lint: lint_SRCS
44 #
45 clobber: clean
46     -$(RM) $(PROG)
47 #
48 clean:
49     -$(RM) $(OBJS)
50 #
51 $(CMDS): $(TESTDIR) $(PROG)
52 #
53 $(TESTDIR):
54     $(INS.dir)
55 #
56 $(TESTDIR)/%: %
57     $(INS.file)
```





```

125     /* we sleep a random bit to mix it up */
126     (void) usleep(rand() % 10);

128     (void) uselocale(loc);
129     day = nl_langinfo(DAY_1);
130     if (strcmp(day, l->day1) != 0) {
131         test_failed(tname, "uselocale data mismatch (%s != %s)",
132                 day, l->day1);
133     }

135     lc = localeconv();
136     if (strcmp(lc->currency_symbol, l->cursym) != 0) {
137         test_failed(tname, "uselocal cursym %s != %s",
138                 lc->currency_symbol, l->cursym);
139     }
140     if (debug)
141         (void) printf("CSYM: %s\n", lc->currency_symbol);

143     /* we sleep a random bit to mix it up */
144     (void) usleep(rand() % 10);

146     if (uselocale(cloc) != loc) {
147         test_failed(tname, "revert old locale mismatch");
148     }
149     freelocale(loc);
150     if (uselocale(LC_GLOBAL_LOCALE) != cloc) {
151         test_failed(tname, "revert GLOBAL_LOCALE mismatch");
152     }
153 }
154 return (NULL);
155 }

158 void
159 testlocale(void)
160 {
161     char          *tname = "newlocale/uselocale";
162     pthread_t     tid[NUMTHR];

164     test_start(tname, "running %d threads %d iterations\n", NUMTHR, NUMITR);

166     for (int i = 0; i < NUMTHR; i++) {
167         (void) pthread_create(&tid[i], NULL, testlocale_thr, tname);
168     }

170     for (int i = 0; i < NUMTHR; i++) {
171         (void) pthread_join(tid[i], NULL);
172     }
173     test_passed(tname);
174 }

176 int
177 main(int argc, char **argv)
178 {
179     int optc;

181     while ((optc = getopt(argc, argv, "d")) != EOF) {
182         switch (optc) {
183             case 'd':
184                 debug++;
185                 break;
186             default:
187                 (void) fprintf(stderr, "Usage: %s [-d]\n", argv[0]);
188                 exit(1);
189         }
190     }

```

```

192     testlocale();
194     exit(0);
195 }

```



```

122 *
123 * cc -Xa (default)                0                undefined
124 * cc -Xt (transitional)           0                undefined
125 * cc -Xc (strictly conforming)    1                undefined
126 * cc -Xs (K&R C)                  undefined       undefined
127 *
128 * gcc (default)                    1                undefined
129 * gcc -ansi, -std={c89, c99,...}   1                defined
130 * gcc -traditional (K&R)           undefined       undefined
131 *
132 * The default compilation modes for Sun C compilers versus GNU C compilers
133 * results in a differing value for __STDC__ which results in a more
134 * restricted namespace when using Sun compilers. To allow both GNU and Sun
135 * interpretations to peacefully co-exist, we use the following Sun
136 * implementation __STRICT_STDC__ macro:
137 */

139 #if (__STDC__ - 0 == 1 && !defined(__GNUC__)) || \
140     (defined(__GNUC__) && defined(__STRICT_ANSI__))
141 #define __STRICT_STDC
142 #else
143 #undef __STRICT_STDC
144 #endif

146 /*
147 * Compiler complies with ISO/IEC 9899:1999
148 */

150 #if __STDC_VERSION__ - 0 >= 199901L
151 #define __STDC_C99
152 #endif

154 /*
155 * Use strict symbol visibility.
156 */
157 #if (defined(__STRICT_STDC) || defined(__XOPEN_OR_POSIX)) && \
158     !defined(__EXTENSIONS__)
159 #define __STRICT_SYMBOLS
160 #endif

162 /*
163 * Large file interfaces:
164 *
165 *     _LARGEFILE_SOURCE
166 *         1                large file-related additions to POSIX
167 *                         interfaces requested (fseeko, etc.)
168 *     _LARGEFILE64_SOURCE
169 *         1                transitional large-file-related interfaces
170 *                         requested (seek64, stat64, etc.)
171 *
172 * The corresponding announcement macros are respectively:
173 *     _LFS_LARGEFILE
174 *     _LFS64_LARGEFILE
175 * (These are set in <unistd.h>.)
176 *
177 * Requesting _LARGEFILE64_SOURCE implies requesting _LARGEFILE_SOURCE as
178 * well.
179 *
180 * The large file interfaces are made visible regardless of the initial values
181 * of the feature test macros under certain circumstances:
182 * - If no explicit standards-conforming environment is requested (neither
183 *   of _POSIX_SOURCE nor _XOPEN_SOURCE is defined and the value of
184 *   __STDC__ does not imply standards conformance).
185 * - Extended system interfaces are explicitly requested (__EXTENSIONS__
186 *   is defined).
187 * - Access to in-kernel interfaces is requested (_KERNEL or _KMEMUSER is

```

```

188 *     defined). (Note that this dependency is an artifact of the current
189 *     kernel implementation and may change in future releases.)
190 */
191 #if (!defined(__STRICT_STDC) && !defined(__XOPEN_OR_POSIX)) || \
192     defined(_KERNEL) || defined(_KMEMUSER) || \
193     defined(__EXTENSIONS__)
194 #undef _LARGEFILE64_SOURCE
195 #define _LARGEFILE64_SOURCE 1
196 #endif
197 #if _LARGEFILE64_SOURCE - 0 == 1
198 #undef _LARGEFILE_SOURCE
199 #define _LARGEFILE_SOURCE 1
200 #endif

202 /*
203 * Large file compilation environment control:
204 *
205 * The setting of _FILE_OFFSET_BITS controls the size of various file-related
206 * types and governs the mapping between file-related source function symbol
207 * names and the corresponding binary entry points.
208 *
209 * In the 32-bit environment, the default value is 32; if not set, set it to
210 * the default here, to simplify tests in other headers.
211 *
212 * In the 64-bit compilation environment, the only value allowed is 64.
213 */
214 #if defined(_LP64)
215 #ifndef _FILE_OFFSET_BITS
216 #define _FILE_OFFSET_BITS 64
217 #endif
218 #if _FILE_OFFSET_BITS - 0 != 64
219 #error "invalid _FILE_OFFSET_BITS value specified"
220 #endif
221 #else /* _LP64 */
222 #ifndef _FILE_OFFSET_BITS
223 #define _FILE_OFFSET_BITS 32
224 #endif
225 #if _FILE_OFFSET_BITS - 0 != 32 && _FILE_OFFSET_BITS - 0 != 64
226 #error "invalid _FILE_OFFSET_BITS value specified"
227 #endif
228 #endif /* _LP64 */

230 /*
231 * Use of _XOPEN_SOURCE
232 *
233 * The following X/Open specifications are supported:
234 *
235 * X/Open Portability Guide, Issue 3 (XPG3)
236 * X/Open CAE Specification, Issue 4 (XPG4)
237 * X/Open CAE Specification, Issue 4, Version 2 (XPG4v2)
238 * X/Open CAE Specification, Issue 5 (XPG5)
239 * Open Group Technical Standard, Issue 6 (XPG6), also referred to as
240 * IEEE Std. 1003.1-2001 and ISO/IEC 9945:2002.
241 * Open Group Technical Standard, Issue 7 (XPG7), also referred to as
242 * IEEE Std. 1003.1-2008 and ISO/IEC 9945:2009.
243 *
244 * XPG4v2 is also referred to as UNIX 95 (SUS or SUSv1).
245 * XPG5 is also referred to as UNIX 98 or the Single Unix Specification,
246 * Version 2 (SUSv2)
247 * XPG6 is the result of a merge of the X/Open and POSIX specifications
248 * and as such is also referred to as IEEE Std. 1003.1-2001 in
249 * addition to UNIX 03 and SUSv3.
250 * XPG7 is also referred to as UNIX 08 and SUSv4.
251 *
252 * When writing a conforming X/Open application, as per the specification
253 * requirements, the appropriate feature test macros must be defined at

```

```

254 * compile time. These are as follows. For more info, see standards(5).
255 *
256 * Feature Test Macro                               Specification
257 * -----
258 * _XOPEN_SOURCE                                   XPG3
259 * _XOPEN_SOURCE && _XOPEN_VERSION = 4            XPG4
260 * _XOPEN_SOURCE && _XOPEN_SOURCE_EXTENDED = 1    XPG4v2
261 * _XOPEN_SOURCE = 500                             XPG5
262 * _XOPEN_SOURCE = 600 (or POSIX_C_SOURCE=200112L) XPG6
263 * _XOPEN_SOURCE = 700 (or POSIX_C_SOURCE=200809L) XPG7
264 *
265 * In order to simplify the guards within the headers, the following
266 * implementation private test macros have been created. Applications
267 * must NOT use these private test macros as unexpected results will
268 * occur.
269 *
270 * Note that in general, the use of these private macros is cumulative.
271 * For example, the use of _XPG3 with no other restrictions on the X/Open
272 * namespace will make the symbols visible for XPG3 through XPG6
273 * compilation environments. The use of _XPG4_2 with no other X/Open
274 * namespace restrictions indicates that the symbols were introduced in
275 * XPG4v2 and are therefore visible for XPG4v2 through XPG6 compilation
276 * environments, but not for XPG3 or XPG4 compilation environments.
277 *
278 * _XPG3      X/Open Portability Guide, Issue 3 (XPG3)
279 * _XPG4      X/Open CAE Specification, Issue 4 (XPG4)
280 * _XPG4_2    X/Open CAE Specification, Issue 4, Version 2 (XPG4v2/UNIX 95/SUS)
281 * _XPG5      X/Open CAE Specification, Issue 5 (XPG5/UNIX 98/SUSv2)
282 * _XPG6      Open Group Technical Standard, Issue 6 (XPG6/UNIX 03/SUSv3)
283 * _XPG7      Open Group Technical Standard, Issue 7 (XPG7/UNIX 08/SUSv4)
284 */

286 /* X/Open Portability Guide, Issue 3 */
287 #if defined(_XOPEN_SOURCE) && (_XOPEN_SOURCE - 0 < 500) && \
288     (_XOPEN_VERSION - 0 < 4) && !defined(_XOPEN_SOURCE_EXTENDED)
289 #define _XPG3
290 /* X/Open CAE Specification, Issue 4 */
291 #elif defined(_XOPEN_SOURCE) && _XOPEN_VERSION - 0 == 4)
292 #define _XPG4
293 #define _XPG3
294 /* X/Open CAE Specification, Issue 4, Version 2 */
295 #elif defined(_XOPEN_SOURCE) && _XOPEN_SOURCE_EXTENDED - 0 == 1)
296 #define _XPG4_2
297 #define _XPG4
298 #define _XPG3
299 /* X/Open CAE Specification, Issue 5 */
300 #elif (_XOPEN_SOURCE - 0 == 500)
301 #define _XPG5
302 #define _XPG4_2
303 #define _XPG4
304 #define _XPG3
305 #undef _POSIX_C_SOURCE
306 #define _POSIX_C_SOURCE 199506L
307 /* Open Group Technical Standard, Issue 6 */
308 #elif (_XOPEN_SOURCE - 0 == 600) || (_POSIX_C_SOURCE - 0 == 200112L)
309 #define _XPG6
310 #define _XPG5
311 #define _XPG4_2
312 #define _XPG4
313 #define _XPG3
314 #undef _POSIX_C_SOURCE
315 #define _POSIX_C_SOURCE 200112L
316 #undef _XOPEN_SOURCE
317 #define _XOPEN_SOURCE 600
319 /* Open Group Technical Standard, Issue 7 */

```

```

320 #elif (_XOPEN_SOURCE - 0 == 700) || (_POSIX_C_SOURCE - 0 == 200809L)
321 #define _XPG7
322 #define _XPG6
323 #define _XPG5
324 #define _XPG4_2
325 #define _XPG4
326 #define _XPG3
327 #undef _POSIX_C_SOURCE
328 #define _POSIX_C_SOURCE 200809L
329 #undef _XOPEN_SOURCE
330 #define _XOPEN_SOURCE 700
331 #endif

333 /*
334 * _XOPEN_VERSION is defined by the X/Open specifications and is not
335 * normally defined by the application, except in the case of an XPG4
336 * application. On the implementation side, _XOPEN_VERSION defined with
337 * the value of 3 indicates an XPG3 application, _XOPEN_VERSION defined
338 * with the value of 4 indicates an XPG4 or XPG4v2 (UNIX 95) application.
339 * _XOPEN_VERSION defined with a value of 500 indicates an XPG5 (UNIX 98)
340 * application and with a value of 600 indicates an XPG6 (UNIX 03)
341 * application and with a value of 700 indicates an XPG7 (UNIX 08).
342 * The appropriate version is determined by the use of the
343 * application. The appropriate version is determined by the use of the
344 * feature test macros described earlier. The value of _XOPEN_VERSION
345 * defaults to 3 otherwise indicating support for XPG3 applications.
346 */
347 #ifndef _XOPEN_VERSION
348 #if defined(_XPG7)
349 #define _XOPEN_VERSION 700
350 #elif defined(_XPG6)
351 #define _XOPEN_VERSION 600
352 #elif defined(_XPG5)
353 #define _XOPEN_VERSION 500
354 #elif defined(_XPG4_2)
355 #define _XOPEN_VERSION 4
356 #else
357 #define _XOPEN_VERSION 3
358 #endif
359 #endif

360 /*
361 * ANSI C and ISO 9899:1990 say the type long long doesn't exist in strictly
362 * conforming environments. ISO 9899:1999 says it does.
363 *
364 * The presence of _LONGLONG_TYPE says "long long exists" which is therefore
365 * defined in all but strictly conforming environments that disallow it.
366 */
367 #if !defined(_STDC_C99) && defined(_STRICT_STDC) && !defined(__GNUC__)
368 /*
369 * Resist attempts to force the definition of long long in this case.
370 */
371 #if defined(_LONGLONG_TYPE)
372 #error "No long long in strictly conforming ANSI C & 1990 ISO C environments"
373 #endif
374 #else
375 #if !defined(_LONGLONG_TYPE)
376 #define _LONGLONG_TYPE
377 #endif
378 #endif

380 /*
381 * It is invalid to compile an XPG3, XPG4, XPG4v2, or XPG5 application
382 * using c99. The same is true for POSIX.1-1990, POSIX.2-1992, POSIX.1b,
383 * and POSIX.1c applications. Likewise, it is invalid to compile an XPG6

```

```
384 * or a POSIX.1-2001 application with anything other than a c99 or later
385 * compiler. Therefore, we force an error in both cases.
386 */
387 #if defined(_STDC_C99) && (defined(__XOPEN_OR_POSIX) && !defined(__XPG6))
388 #error "Compiler or options invalid for pre-UNIX 03 X/Open applications \
389 and pre-2001 POSIX applications"
390 #elif !defined(_STDC_C99) && \
391 (defined(__XOPEN_OR_POSIX) && defined(__XPG6))
392 #error "Compiler or options invalid; UNIX 03 and POSIX.1-2001 applications \
393 require the use of c99"
394 #endif

396 /*
397 * The following macro defines a value for the ISO C99 restrict
398 * keyword so that _RESTRICT_KYWD resolves to "restrict" if
399 * an ISO C99 compiler is used and "" (null string) if any other
400 * compiler is used. This allows for the use of single prototype
401 * declarations regardless of compiler version.
402 */
403 #if (defined(_STDC_) && defined(_STDC_C99)) && !defined(__cplusplus)
404 #define _RESTRICT_KYWD restrict
405 #else
406 #define _RESTRICT_KYWD
407 #endif

409 /*
410 * The following macro indicates header support for the ANSI C++
411 * standard. The ISO/IEC designation for this is ISO/IEC FDIS 14882.
412 */
413 #define _ISO_CPP_14882_1998

415 /*
416 * The following macro indicates header support for the C99 standard,
417 * ISO/IEC 9899:1999, Programming Languages - C.
418 */
419 #define _ISO_C_9899_1999

421 /*
422 * The following macro indicates header support for DTrace. The value is an
423 * integer that corresponds to the major version number for DTrace.
424 */
425 #define _DTRACE_VERSION 1

427 #ifdef __cplusplus
428 }
  unchanged portion omitted
```