

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

1

2047 Tue May 20 07:58:43 2014
new/usr/src/cmd/localedef/messages.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 Tue May 20 07:58:43 2014

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

2964 need POSIX 2008 locale object support

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

```
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 Tue May 20 07:58:44 2014
new/usr/src/cmd/localedef/numeric.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 }
```

_____unchanged_portion_omitted_____

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

1

4496 Tue May 20 07:58:44 2014

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

2964 need POSIX 2008 locale object support

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

```
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_____

new/usr/src/cmd/mdb/common/kmdb/mapfile_skel

1

```
*****
1734 Tue May 20 07:58:44 2014
new/usr/src/cmd/mdb/common/kmdb/mapfile_skel
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */

26 /*
27  * This file is used to limit the symbols that are to be exported from the
28  * debugger. This ensures that dmods follow the module API.
29  *
30  * There is a special rule for generating the mapfile. If the symbol
31  * is not a function then the symbol, in the mapfile, must have the
32  * the token "variable" as the third symbol on the line - see the
33  * entries for __ctype and errno below.
34 */

36 $mapfile_version 2

38 SYMBOL_SCOPE {
39     global:

41 /* END PROLOGUE */
42 /* BEGIN EPILOGUE */
43 /*
44  * Secret additions to the module API
45 */

47 /* Implementation detail of the ctype macros */
48 __ctype; /* variable */
47 /* There should be only one - ours */
48 errno; /* variable */
49 isprint;
50 isalnum;
51 isalpha;
52 isgraph;
53 iscntrl;
54 isdigit;
55 isxdigit;
56 isupper;
57 islower;
58 ispunct;
```

new/usr/src/cmd/mdb/common/kmdb/mapfile_skel

2

```
59 isspace;

61 mdb_tgt_aread;
62 mdb_dis_create;
63 mdb_dis_destroy;

65 local:
66 *;
67 };
_____unchanged_portion_omitted_____
```



```
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,
43      _U,      _U,      _U,      _U,      _U,      _U,      _U,      _U,
44      _U,      _U,      _U,      _U,      _U,      _U,      _U,      _U,
45      _U,      _U,      _P,      _P,      _P,      _P,      _P,      _P,
46      _P,      _L|_X,   _L|_X,   _L|_X,   _L|_X,   _L|_X,   _L|_X,   _L,
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  };
```

new/usr/src/cmd/sgs/rtld/common/external.c

1

19562 Tue May 20 07:58:44 2014

new/usr/src/cmd/sgs/rtld/common/external.c

2964 need POSIX 2008 locale object support

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

unchanged_portion_omitted

```
672 /*
673  * ASCII versions of ctype character classification functions. This avoids
674  * pulling in the entire locale framework that is in libc.
675  */
```

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

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

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

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

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

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

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

new/usr/src/head/Makefile

1

11016 Tue May 20 07:58:44 2014

new/usr/src/head/Makefile

2964 need POSIX 2008 locale object support

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

```
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 \
```

new/usr/src/head/Makefile

2

```
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      ieee_fp.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     mdsn_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 \
125     nss_dbdefs.h \
126     nss_netdir.h \
```

new/usr/src/head/Makefile

```

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      siginfo.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   \
191      ucred.h      \
192      ulimit.h     \

```

3

new/usr/src/head/Makefile

```

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      wchar_iso.h  \
231      wctype_c99.h \
232      wctype_iso.h \
233
234      ARPAHDRS = \
235      ftp.h      \
236      inet.h     \
237      nameser.h  \
238      telnet.h   \
239      tftp.h     \
240      nameser_compat.h \
241
242      AUDIOHDRS = \
243      au.h       \
244
245      UUIDHDRS = \
246      uuid.h     \
247
248      # rpcsvc headers which are just headers (not derived from a .x file)
249      RPCSVC_SRC_HDRS = \
250      bootparam.h \
251      daemon_utils.h \
252      dbm.h        \
253      nis_db.h     \
254      nislib.h     \
255      svc_dg_priv.h \
256      yp_prot.h    \
257      ypclnt.h     \
258      yppasswd.h   \

```

4

new/usr/src/head/Makefile

```

257         ypupd.h      \
258         rpc_sztypes.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     mhd.h mdiox.h meta_basic.h metad.h metamed.h metamhd.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 mhd.h := RPCGENFLAGS += -M
290 mdiox.h := RPCGENFLAGS += -M
291 metamhd.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     $(UUIDHDRS:%=$(ROOT)/usr/include/uuid/%) \
320     $(RPCSVCHDRS:%=$(ROOT)/usr/include/rpcsvc/%) \
321     $(RPCSVCPROTS:%=$(ROOT)/usr/include/rpcsvc/%) \
322     $(LVMPCHDRS:%=$(ROOT)/usr/include/%) \

```

5

new/usr/src/head/Makefile

```

323     $(PROTOHDRS:%=$(ROOT)/usr/include/protocols/%)

325 DIRS= iso arpa audio rpcsvc protocols security uuid kerberosv5
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 kerberosv5/%.check:      kerberosv5/%.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=kerberosv5/%.check) \
368     $(ISOHDRS:%.h=iso/%.check) \
369     $(ARPAHDRS:%.h=arpa/%.check) \
370     $(AUDIOHDRS:%.h=audio/%.check) \
371     $(UUIDHDRS:%.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)

388 $(ROOT)/usr/include/protocols/%: protocols/%

```

6

new/usr/src/head/Makefile

7

```

389      $(INS.file)

391 $(ROOT)/usr/include/rpcsvc/%: rpcsvc/%
392     $(INS.file)

394 $(ROOT)/usr/include/kerberosv5/%: kerberosv5/%
395     $(INS.file)

397 $(ROOT)/usr/include/arpa/%: arpa/%
398     $(INS.file)

400 $(ROOT)/usr/include/audio/%: audio/%
401     $(INS.file)

403 $(ROOT)/usr/include/iso/%: iso/%
404     $(INS.file)

406 $(ROOT)/usr/include/uuid/%: uuid/%
407     $(INS.file)

409 $(ROOT)/usr/include/%: %
410     $(INS.file)

412 .KEEP_STATE:

414 .PARALLEL:      $(ROOTHDRS) $(CHECKHDRS)

416 install_h:      $(ROOTDIRS) .WAIT $(ROOTHDRS) $(SYMHDRASSERT) $(SYMHDRERRNO) \
417                 $(SYMHDRFLOAT) $(SYMHDRISO646)

419 check: $(CHECKHDRS)

421 clean clobber:
422     $(RM) $(LVMRPGHDRS);
423     cd rpcsvc ; $(RM) $(RPCSVC_GEN_HDRS)

425 $(ROOTDIRS):
426     $(INS.dir)

428 $(SYMHDRASSERT):
429     -$(RM) $@; $(SYMLINK) ../assert.h $@

431 $(SYMHDRERRNO):
432     -$(RM) $@; $(SYMLINK) ../errno.h $@

434 $(SYMHDRFLOAT):
435     -$(RM) $@; $(SYMLINK) ../float.h $@

437 $(SYMHDRISO646):
438     -$(RM) $@; $(SYMLINK) ../iso646.h $@

440 rpcsvc/%.h:      rpcsvc/%.x
441     $(RPCGEN) $(RPCGENFLAGS) $< -o $@

443 rpcsvc/nis.h:    rpcsvc/nis.x
444     $(RPCGEN) $(RPCGENFLAGS) rpcsvc/nis.x | \
445     $(SED) -e '/EDIT_START/,,$$ d' > $@

447 meta_basic.h:    ../uts/common/sys/lvm/meta_basic.x
448     $(RPCGEN) $(RPCGENFLAGS) ../uts/common/sys/lvm/meta_basic.x | \
449     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
450     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
451     { print $0 } \
452     ' > $@

454 metad.h:         metad.x

```

new/usr/src/head/Makefile

8

```

455     $(RPCGEN) $(RPCGENFLAGS) metad.x | \
456     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
457     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
458     { print $0 } \
459     ' > $@

461 mhd_x.h:    ../uts/common/sys/lvm/mhd_x.x
462     $(RPCGEN) $(RPCGENFLAGS) ../uts/common/sys/lvm/mhd_x.x | \
463     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
464     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
465     { print $0 } \
466     ' > $@

468 mdiox.h:    ../uts/common/sys/lvm/mdiox.x
469     $(RPCGEN) $(RPCGENFLAGS) ../uts/common/sys/lvm/mdiox.x | \
470     nawk '{sub(/sys\/lvm\/md_mhd_x/, "mhd_x"); print $$0}' | \
471     nawk '{sub(/sys\/lvm\/md_basic/, "meta_basic"); print $$0}' | \
472     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
473     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
474     { print $0 } \
475     ' > $@

477 metamed.h:    ../uts/common/sys/lvm/metamed.x
478     $(RPCGEN) $(RPCGENFLAGS) ../uts/common/sys/lvm/metamed.x | \
479     nawk '{sub(/sys\/lvm\/md_basic/, "meta_basic"); print $$0}' | \
480     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
481     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
482     { print $0 } \
483     ' > $@

485 metamhd.h:    metamhd.x
486     $(RPCGEN) $(RPCGENFLAGS) metamhd.x | \
487     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
488     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
489     { print $0 } \
490     ' > $@

492 metacl.h:    metacl.x
493     $(RPCGEN) $(RPCGENFLAGS) metacl.x | \
494     awk '/<synch.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t" _
495     /<thread.h>/ { print "#ifdef _REENTRANT"; print $$0; print "#endif\t/* _
496     { print $0 } \
497     ' > $@

```

new/usr/src/head/ctype.h

1

```
*****
3390 Tue May 20 07:58:44 2014
new/usr/src/head/ctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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;
56 using std::toupper;
57 #if __cplusplus >= 201103L
```

new/usr/src/head/ctype.h

2

```
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_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__) */
```

new/usr/src/head/ctype.h

3

```
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 Tue May 20 07:58:44 2014
new/usr/src/head/iso/ctype_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

48 #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 */
57 #define _N      0x00000004      /* Numeral (digit) */
58 #define _S      0x00000008      /* Spacing character */
```

new/usr/src/head/iso/ctype_iso.h

2

```
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
87 #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) */

177 #else /* defined(__STDC__) */

181 extern unsigned char _ctype[];

189 #if !defined(__lint)

191 extern int isalpha();
192 extern int isupper();
193 extern int islower();
194 extern int isdigit();
195 extern int isxdigit();
196 extern int isalnum();
197 extern int isspace();
198 extern int ispunct();
199 extern int isprint();
200 extern int isgraph();
201 extern int iscntrl();
202 extern int isblank();
203 #define isalpha(c)      ((__ctype + 1)[(int)(c)] & (_U | _L))
204 #define isupper(c)      ((__ctype + 1)[(int)(c)] & _U)
205 #define islower(c)      ((__ctype + 1)[(int)(c)] & _L)
206 #define isdigit(c)      ((__ctype + 1)[(int)(c)] & _N)
207 #define isxdigit(c)     ((__ctype + 1)[(int)(c)] & _X)
208 #define isalnum(c)      ((__ctype + 1)[(int)(c)] & (_U | _L | _N))
209 #define isspace(c)      ((__ctype + 1)[(int)(c)] & _S)
210 #define ispunct(c)      ((__ctype + 1)[(int)(c)] & _P)
211 #define isprint(c)      ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N | _B))
212 #define isgraph(c)      ((__ctype + 1)[(int)(c)] & (_P | _U | _L | _N))
213 #define iscntrl(c)      ((__ctype + 1)[(int)(c)] & _C)

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

216 #endif /* defined(__STDC__) */

218 #ifdef __cplusplus
219 }
220 #endif

```

unchanged portion omitted

new/usr/src/head/iso/stdlib_iso.h

1

```
*****
6072 Tue May 20 07:58:44 2014
new/usr/src/head/iso/stdlib_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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]
57 #else
58 unsigned char __mb_cur_max();
```

new/usr/src/head/iso/stdlib_iso.h

2

```
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

```
*****
4735 Tue May 20 07:58:45 2014
new/usr/src/head/iso/wctype_iso.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 {
57 #endif
```

1

new/usr/src/head/iso/wctype_iso.h

```
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 iswalphabetic(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 #else
112 extern int iswalnum();
113 extern int iswalphabetic();
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 */
```

2

```
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 Tue May 20 07:58:45 2014
new/usr/src/head/langinfo.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */

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

new/usr/src/head/langinfo.h

2

```
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

```
*****
2982 Tue May 20 07:58:45 2014
new/usr/src/head/locale.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

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

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

58 #ifdef __cplusplus
```

new/usr/src/head/locale.h

2

```
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 locale_t    duplocale(locale_t);
89 void        freelocale(locale_t);
90 locale_t    newlocale(int, const char *, locale_t);
91 locale_t    uselocale(locale_t);
92 #else /* __STDC__ */
93 locale_t    duplocale();
94 void        freelocale();
95 locale_t    newlocale();
96 locale_t    uselocale();
97 #endif /* __STDC__ */

99 #define LC_GLOBAL_LOCALE    __global_locale
100 extern locale_t    __global_locale;

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

104 #ifdef __cplusplus
105 }

    unchanged_portion_omitted_
```

new/usr/src/head/monetary.h

1

```
*****
1626 Tue May 20 07:58:45 2014
new/usr/src/head/monetary.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

new/usr/src/head/monetary.h

2

```
60 #ifdef __cplusplus
61 }
_____unchanged_portion_omitted_____
```

new/usr/src/head/string.h

1

```
*****
7182 Tue May 20 07:58:45 2014
new/usr/src/head/string.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
25  */

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

30 #ifndef _STRING_H
31 #define _STRING_H

33 #include <iso/string_iso.h>

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::strncmp;
56 using std::strncpy;
57 using std::strpbrk;
58 using std::strrchr;
59 using std::strspn;
60 using std::strstr;
```

new/usr/src/head/string.h

2

```
61 using std::strtok;
62 using std::strxfrm;
63 #endif

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

69 #if defined(__STDC__)

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

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

84 #if defined(__EXTENSIONS__) || !defined(_STRICT_STDC) || \
85     defined(_XOPEN_OR_POSIX)
86 extern void *memccpy(void *_RESTRICT_KYWD, const void *_RESTRICT_KYWD,
87     int, size_t);
88 #endif

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

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 strnlen(const char *, size_t);
101 extern char *strsignal(int);

103 #ifndef _LOCALE_T
104 #define _LOCALE_T
105 typedef struct locale_t;
106 #endif

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);

114 #endif /* defined(_STRICT_SYMBOLS) || defined(_XPG7) */

116 #if !defined(_STRICT_SYMBOLS)

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 uucopy(const void *_RESTRICT_KYWD, void *_RESTRICT_KYWD, size_t);
122 extern int uucopystr(const void *_RESTRICT_KYWD, void *_RESTRICT_KYWD, size_t);
123 extern int ffs(int);
124 extern int ffs_l(long);
125 extern int ffsll(long long);
126 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__)
143 /*
144  * gcc provides this inlining facility but Studio C does not.
145  * We should use it exclusively once Studio C also provides it.
146  */
147 #else
148 extern void *__builtin_alloca(size_t);
149 #endif
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 = strcpy((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     { \
180         __strdupa_str = (char *) (s), \
181         strcpy((char *) __builtin_alloca(strlen(__strdupa_str) + 1), \
182             __strdupa_str) \
183     }

183 #define strndupa(s, n) \
184     { \
185         __strdupa_str = (char *) (s), \
186         __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 *memccpy();
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 int strcasecmp_l();
218 extern int strncasecmp_l();
219 extern char *stpcpy();
220 extern char *stpncpy();
221 extern char *strndup();
222 extern size_t strnlen();
223 extern char *strsignal();
224 #endif

226 #if !defined(_STRICT_SYMBOLS)
227 #if defined(__EXTENSIONS__) || \
228     (!defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX))
229 extern int uucopy();
230 extern int uucopystr();
231 extern int ffs();
232 extern int flsl();
233 extern int flsll();
234 extern int fls();
235 extern int flsl();
236 extern int flsll();
237 extern char *strcasestr();
238 extern char *strcasestr_l();
239 extern char *strnstr();
240 extern size_t strlcpy();
241 extern size_t strlcat();
242 extern char *strsep();
243 extern char *strchrnul();
244 #endif /* _STRICT_SYMBOLS */
245 #endif /* defined(__EXTENSIONS__) ... */

247 #if defined(__EXTENSIONS__) || !defined(__XOPEN_OR_POSIX) || defined(_XPG4_2)
248 extern char *strdup();
249 #endif

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

```

new/usr/src/head/string.h

5

```
246 extern size_t strcoll_1();
247 extern size_t strxfrm_1();
248 #endif

250 #endif /* __STDC__ */

252 #ifdef __cplusplus
253 }
_____unchanged_portion_omitted_
```

new/usr/src/head/strings.h

1

```
*****
2905 Tue May 20 07:58:45 2014
new/usr/src/head/strings.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

32 #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
57  * specifically defined for each of these headers.  As a result,
58  * inclusion of <string.h> would result in declarations not allowed
```

new/usr/src/head/strings.h

2

```
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 Tue May 20 07:58:45 2014
new/usr/src/head/time.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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;
59 using std::localtime;
60 using std::mktime;
```

new/usr/src/head/time.h

2

```
61 using std::time;
62 using std::strptime;
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 strptime __strptime_dontzero
96 #else /* __PRAGMA_REDEFINE_EXTNAME */
97 extern char *_strptime_dontzero(const char *, const char *, struct tm *);
98 #define strptime __strptime_dontzero
99 #endif /* __PRAGMA_REDEFINE_EXTNAME */
100 #endif /* _STRPTIME_DONTZERO */

102 extern char *strptime(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 };
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
```

```
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 Tue May 20 07:58:45 2014
new/usr/src/head/wchar.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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::fgetws;
48 using std::fputwc;
49 using std::fputws;
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::wcscat;
59 using std::wcschr;
60 using std::wcscmp;
```

new/usr/src/head/wchar.h

2

```
61 using std::wcscoll;
62 using std::wcscpy;
63 using std::wcscspn;
64 using std::wcslen;
65 using std::wcsncat;
66 using std::wcsncmp;
67 using std::wcsncpy;
68 using std::wcpbrk;
69 using std::wcsrchr;
70 using std::wcssp;
71 using std::wcxfrm;
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::mbsrtowcs;
84 using std::swprintf;
85 using std::swscanf;
86 using std::vfwprintf;
87 using std::vwprintf;
88 using std::vswprintf;
89 using std::wctomb;
90 using std::wcsrtombs;
91 using std::wcsstr;
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 towlower(wint_t);
142 extern wint_t towupper(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::towlower;
163 using std::towupper;
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_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 *wcpcpy(wchar_t *_RESTRICT_KYWD, const wchar_t *_RESTRICT_KYWD);
176 extern wchar_t *wcpncpy(wchar_t *_RESTRICT_KYWD, const wchar_t *_RESTRICT_KYWD,
177     size_t);
178 extern size_t wcsxfrm_l(wchar_t *_RESTRICT_KYWD, const wchar_t *_RESTRICT_KYWD,
179     size_t, locale_t);
180 extern int wcsxfrm_l(const wchar_t *, const wchar_t *, locale_t);
181 extern wchar_t *wcsdup(const wchar_t *);
182 extern int wcscasecmp(const wchar_t *, const wchar_t *);
183 extern int wcscasecmp_l(const wchar_t *, const wchar_t *, locale_t);
184 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 *_RESTRICT_KYWD, const char **_RESTRICT_KYWD,
191     size_t, size_t, mbstate_t *_RESTRICT_KYWD);
180 #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 towlower();
211 extern wint_t towupper();
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 *wcpcpy();
225 extern wchar_t *wcpncpy();
226 extern size_t wcsxfrm_l();
227 extern int wcsxfrm_l();
228 extern int wcscasecmp();
229 extern int wcscasecmp_l();
230 extern int wcsncasecmp();
231 extern int wcsncasecmp_l();
232 extern size_t mbsnrtowcs();
230 #endif

232 #endif /* __STDC__ */

234 #ifdef __cplusplus
235 }
    unchanged_portion_omitted

```

new/usr/src/head/wctype.h

1

```
*****
5250 Tue May 20 07:58:45 2014
new/usr/src/head/wctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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) || \
38     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;
55 using std::iswprint;
56 using std::iswpunct;
```

new/usr/src/head/wctype.h

2

```
57 using std::iswspace;
58 using std::iswupper;
59 using std::iswxdigit;
60 using std::towlower;
61 using std::towupper;
62 using std::wctrans;
63 using std::towctrans;
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);
```

new/usr/src/head/wctype.h

3

```
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 iswspecial(wint_t);
122 #else
120 #else /* __STDC__ */
124 extern unsigned _iswctype();
125 extern wchar_t _trwctype();
122 /* is* also become functions */
123 extern int isphonogram();
124 extern int isideogram();
125 extern int isenglish();
126 extern int isnumber();
127 extern int isspecial();
128 /* From BSD/MacOS */
129 extern int iswideogram();
130 extern int iswphonogram();
131 extern int iswnumber();
132 extern int iswspecial();
133 #endif
```

```
135 #define iscodeset0(c) isascii(c)
136 #define iscodeset1(c) (((c) & WCHAR_CSMASK) == WCHAR_CS1)
137 #define iscodeset2(c) (((c) & WCHAR_CSMASK) == WCHAR_CS2)
138 #define iscodeset3(c) (((c) & WCHAR_CSMASK) == WCHAR_CS3)
```

```
140 #endif /* !defined(_STRICT_SYMBOLS)... */
139 #endif /* !defined(__XOPEN_OR_POSIX)... */
```

```
143 /* XPG7 extended locale support */
144 #if defined(_XPG7) || !defined(_STRICT_SYMBOLS)
```

```
146 #ifndef _LOCALE_T
147 #define _LOCALE_T
148 typedef struct locale *locale_t;
149 #endif
```

```
151 #if defined(__STDC__)
152 extern wint_t tolower_l(wint_t, locale_t);
153 extern wint_t toupper_l(wint_t, locale_t);
154 extern wint_t towctrans_l(wint_t, wctrans_t, locale_t);
155 extern int iswctype_l(wint_t, wctype_t, locale_t);
156 extern int iswalnum_l(wint_t, locale_t);
157 extern int iswalphal_l(wint_t, locale_t);
158 extern int iswcntrl_l(wint_t, locale_t);
159 extern int iswdigit_l(wint_t, locale_t);
160 extern int iswgraph_l(wint_t, locale_t);
161 extern int iswlower_l(wint_t, locale_t);
162 extern int iswprint_l(wint_t, locale_t);
163 extern int iswpunct_l(wint_t, locale_t);
164 extern int iswspace_l(wint_t, locale_t);
165 extern int iswupper_l(wint_t, locale_t);
166 extern int iswxdigit_l(wint_t, locale_t);
167 extern wctrans_t wctrans_l(const char *, locale_t);
168 extern wctype_t wctype_l(const char *, locale_t);
169 #else /* __STDC__ */
170 extern wint_t tolower_l();
171 extern wint_t toupper_l();
172 extern wint_t towctrans_l();
173 extern int iswctype_l();
```

new/usr/src/head/wctype.h

4

```
174 extern int iswalnum_l();
175 extern int iswalphal_l();
176 extern int iswcntrl_l();
177 extern int iswdigit_l();
178 extern int iswgraph_l();
179 extern int iswlower_l();
180 extern int iswprint_l();
181 extern int iswpunct_l();
182 extern int iswspace_l();
183 extern int iswupper_l();
184 extern int iswxdigit_l();
185 extern wctrans_t wctrans_l();
186 extern wctype_t wctype_l();
187 #endif /* __STDC__ */
188 #endif /* defined(_XPG7) || !defined(_STRICT_SYMBOLS) */
```

```
190 #ifdef __cplusplus
191 }
    unchanged_portion_omitted
```

```

*****
3568 Tue May 20 07:58:45 2014
new/usr/src/head/xlocale.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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_t;
49 #endif

51 int mbsinit_l(const mbstate_t *, locale_t);

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

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

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

```

```

62 int wwidth_l(wchar_t, locale_t);

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

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

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

91 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 wint_t fgetwc_l(FILE *, locale_t);
103 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 Tue May 20 07:58:46 2014
 new/usr/src/lib/libc/amd64/Makefile
 2964 need POSIX 2008 locale object support
 Reviewed by: Robert Mustacchi <rm@joyent.com>

```

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 CRTOBSJ= \
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 XATTROBJS= \
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     makectxt.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 \
125     setjmp.o \
126     signfolst.o \

```

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

```
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      fac1.o             \
186      fchdir.o           \
187      fchroot.o          \
188      fdsync.o           \
189      fpathconf.o        \
190      fstatfs.o          \
191      fstatvfs.o         \
192      getcpuid.o         \
```

3

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

```
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        \
257      sysfs.o            \
258      sysinfo.o          \
```

4

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

```

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 SYSOBJ$= \
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     \
323      pack_float.o       \
324      sigfpe.o           \

```

5

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

```

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      getdtablesize.o    \
388      getenv.o           \
389      getexecname.o      \
390      getgrnam.o         \

```

6

```

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      lckpwdf.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      tcseendbreak.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      ferror.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     wcswcs.o \
666     wmemchr.o \
667     wmemcmp.o \
668     wmemcpy.o \
669     wmemmove.o \
670     wmemset.o \
673     wscasecmp.o \
671     wscat.o \
672     wschr.o \
673     wscmp.o \
674     wscpy.o \
675     wscspn.o \
676     wsdup.o \
677     wslen.o \
681     wsncasecmp.o \
678     wsncat.o \
679     wsncmp.o \
680     wsncpy.o \
681     wspbrk.o \
682     wprintf.o \
683     wsrchr.o \
684     wsscanf.o \
685     wsspn.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     ldpart.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     regexec.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     strptime.o \
758     strxfrm.o \
759     table.o \
760     timelocal.o \
761     tolower.o \
762     towlower.o \
763     ungetwc.o \
764     utf8.o \
765     wcrtoomb.o \
766     wcscasecmp.o \
767     wcscoll.o \
768     wcsftime.o \
769     wcsnrtombs.o \
770     wcsrtombs.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 \

```

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

```

781 AI00BJS= \
782     aio.o \
783     aio_alloc.o \
784     posix_aio.o

786 RT0BJS= \
787     clock_timer.o \
788     mqueue.o \
789     pos4obj.o \
790     sched.o \
791     sem.o \
792     shm.o \
793     sigev_thread.o

795 TP00LOBJS= \
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=

```

13

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

```

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     execle.o \
858     execv.o \
859     fcntl.o \
860     getpagesizes.o \
861     getpeerucrd.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

```

14

```

914 MOSTOBSJS=      \
915     $(STRETS)      \
916     $(CRTOBSJS)    \
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     $(TPOOLOBJS)   \
942     $(THREADSOBJS) \
943     $(THREADSMACHOBJS) \
944     $(THREADSASMOBJS) \
945     $(UNICODEOBJS) \
946     $(UNWINDMACHOBJS) \
947     $(UNWINDASMOBJS) \
948     $(COMSYSOBJS)  \
949     $(SYSOBJS)     \
950     $(COMSYSOBJS64) \
951     $(SYSOBJS64)   \
952     $(VALUES)

954 TRACEOBJS=      \
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) $(MOSTOBSJS) $(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= $(TRACEOBJS:%=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 dtrace 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) $(MOSTOBSJS:%=$(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 \

```

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

```

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/i18n/%.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      $(TPOOLOBJS:.o=$(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/siginfolt.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       \

```

17

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

```

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_LONGLONG

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/lib-libc

```

18

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

19

```
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
```

new/usr/src/lib/libc/i386/Makefile.com

1

24030 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/i386/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 CRTOBSJS=      \
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 \
```

new/usr/src/lib/libc/i386/Makefile.com

2

```
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 XATTROBJS=       \
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           \
```

new/usr/src/lib/libc/i386/Makefile.com

```
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 COMSYSOBJS64=
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 SYSOBJS64=

164 COMSYSOBJS=
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      \
```

3

new/usr/src/lib/libc/i386/Makefile.com

```
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      faccl.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_utrap.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            \
```

4

```

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      ssignal.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      strtointmax.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      ferror.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       \
701      strcasecmp.o  \
702      strcasestr.o  \
703      strncasecmp.o \
704      strtows.o     \
705      wcsnlen.o     \
706      wcsstr.o      \
707      wcstoimax.o   \
708      wcstol.o      \
709      wcstoul.o     \
710      wcsvcs.o      \
711      wmemchr.o     \
712      wmemcmp.o     \
713      wmemcpy.o     \
714      wmemmove.o    \
715      wmemset.o     \
716      wscasecmp.o  \
717      wscat.o       \
718      wschr.o       \
719      wscmp.o       \
720      wscpy.o       \

```

```

717      wscspn.o      \
718      wsdup.o       \
719      wslen.o       \
720      wscasecmp.o   \
721      wscat.o       \
722      wscmp.o       \
723      wscpy.o       \
724      wspbrk.o      \
725      wsprintf.o    \
726      wsrchr.o      \
727      wscanf.o      \
728      wsspn.o       \
729      wstod.o       \
730      wstok.o       \
731      wstol.o       \
732      wstoll.o      \
733      wxfrm.o       \
734      gettext.o     \
735      gettext_gnu.o \
736      gettext_real.o \
737      gettext_util.o \
738      isdigit.o     \
739      plural_parser.o \
740      wdresolve.o   \
741      _ctype.o      \
742      isascii.o     \
743      toascii.o     \
744  PORTI18N_COND=    \
745      wcstol_longlong.o \
746      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      towlower.o         \
805      ungetwc.o          \
806      utf8.o             \
807      wcrntomb.o         \
808      wcscasecmp.o        \
809      wcscoll.o          \
810      wcsftime.o         \
811      wcsnrtombs.o       \
812      wcsrtombs.o        \
813      wcswidth.o         \
814      wcostombs.o        \
815      wcsxfrm.o          \
816      wctob.o            \
817      wctomb.o           \
818      wctrans.o          \
819      wctype.o           \
820      wcwidth.o          \
821      wscoll.o           \
822
823      AI00BJS=           \
824      aio.o              \
825      aio_alloc.o        \
826      posix_aio.o        \
827
828      RT0BJS=            \
829      clock_timer.o      \
830      mqueue.o           \
831      pos4obj.o          \
832      sched.o            \
833      sem.o              \
834      shm.o              \
835      sigev_thread.o     \
836
837      TP00LOBJS=         \
838      thread_pool.o      \
839
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              \
863
864      THREADSMACHOBJS=    \
865      machdep.o          \
866
867      THREADSASMOBJS=     \
868      asm_subr.o         \
869
870      UNICODE0BJS=       \
871      u8_textprep.o      \
872      uconv.o            \
873
874      UNWINDMACHOBJS=     \
875      unwind.o           \
876
877      UNWINDASMOBJS=      \
878      unwind_frame.o     \
879
880      # objects that implement the transitional large file API
881      PORTSYS64=          \
882      lockf64.o          \
883      stat64.o           \
884
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      exacctsys.o        \
894      execl.o            \
895      execle.o           \
896      execv.o            \
897      fcntl.o            \
898      getpagesizes.o     \
899      getpeerucrd.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 MOSTOBSJ=      \
950      $(STRETS)    \
951      $(CRTOBSJ)   \
952      $(DYNOBJS)   \
953      $(FPOBSJ)    \
954      $(FPASMOBSJ) \
955      $(ATOMICOBSJ) \
956      $(XATTOBSJ)  \
957      $(COMOBSJ)   \
958      $(DTRACEOBSJ) \
959      $(GENOBSJ)   \
960      $(PORTFP)    \
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      $(AIOOBSJ)   \
977      $(RTOBSJ)    \
978      $(TPOOLOBSJ) \

```

```

979      $(THREADSOBSJ) \
980      $(THREADSMACHOBSJ) \
981      $(THREADSASMOBSJ) \
982      $(UNICODEOBSJ) \
983      $(UNWINDMACHOBSJ) \
984      $(UNWINDASMOBSJ) \
985      $(COMSYSOBSJ) \
986      $(SYSOBSJ) \
987      $(COMSYSOBSJ64) \
988      $(SYSOBSJ64) \
989      $(VALUES)

991 TRACEOBSJ=      \
992      plockstat.o

994 # NOTE: libc.so.1 must be linked with the minimal crt1.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 crt1.o and
1002 # crtn.o modules is safe and avoids the problems described above.
1003 OBJECTS= $(CRTI) $(MOSTOBSJ) $(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 +=      -znnow

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      $(COMOBJS:%.o=$(SRC)/common/util/%.c) \
1116      $(DTRACEOBJS:%.o=$(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      $(AIOOBJS:%.o=$(LIBCDIR)/port/aio/%.c) \
1126      $(RTOBJS:%.o=$(LIBCDIR)/port/rt/%.c) \
1127      $(TPOOLOBJS:%.o=$(LIBCDIR)/port/tpool/%.c) \
1128      $(THREADSOBJS:%.o=$(LIBCDIR)/port/threads/%.c) \
1129      $(THREADSMACHOBJS:%.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1130      $(UNICODEOBJS:%.o=$(SRC)/common/unicode/%.c) \
1131      $(UNWINDMACHOBJS:%.o=$(LIBCDIR)/port/unwind/%.c) \
1132      $(FPOBJS:%.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1133      $(LIBCBASE)/gen/ecvt.c \
1134      $(LIBCBASE)/gen/makectxt.c \
1135      $(LIBCBASE)/gen/siginfofst.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 \

```

new/usr/src/lib/libc/i386/Makefile.com

```
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 $(COMSYSOBSJ64:%=pics/%) := \
1194     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1196 $(SYSOBSJ64:%=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_LONGLONG

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 -erroff=E_SUPPRESSION_DIRECTIVE_UNUSED

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

1235 $(LINTLIB):= SRCS=$(LIBCDIR)/port/lib-lc
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
```

19

new/usr/src/lib/libc/i386/Makefile.com

```
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_0)

1266 ASSYMDP_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 $(ASSYMDP_OBJS:%=pics/%) := CPPFLAGS += -I.

1276 $(ASSYMDP_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/errlst $(LIBCDIR)/port/gen/
1295     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlst.awk < errlst

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

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

20

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

1

```
*****
3927 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/_ctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */
```

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

2

```
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     0x20000000U      /* 0 width character */
77 #define _CTYPE_SW1     0x40000000U      /* 1 width character */
78 #define _CTYPE_SW2     0x80000000U      /* 2 width character */
79 #define _CTYPE_SW3     0xc0000000U      /* 3 width character */
80 #define _CTYPE_SWM     0xe0000000U      /* 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_ */
```

new/usr/src/lib/libc/port/locale/big5.c

1

```
*****
4970 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/big5.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 <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 #include "lctype.h"

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_wrtomb(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;
```

new/usr/src/lib/libc/port/locale/big5.c

2

```
60 } _BIG5State;

62 void
63 _BIG5_init(struct lc_ctype *lct)
64 {
65     lct->lc_mbrtowc = _BIG5_mbrtowc;
66     lct->lc_wrtomb = _BIG5_wrtomb;
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;

59     __mbrtowc = _BIG5_mbrtowc;
60     __wrtomb = _BIG5_wrtomb;
61     __mbsinit = _BIG5_mbsinit;
62     __CurrentRuneLocale = rl;
63     __ctype[520] = 2;
64     charset_is_ascii = 0;
65     return (0);
72 }

    unchanged_portion_omitted

150 static size_t
151 _BIG5_wrtomb(char * _RESTRICT_KYWD s, wchar_t wc, mbstate_t * _RESTRICT_KYWD ps)
152 {
153     _BIG5State *bs;

155     bs = (_BIG5State *)ps;

157     if (bs->ch != 0) {
158         errno = EINVAL;
159         return ((size_t)-1);
160     }

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 }

174 static size_t
175 _BIG5_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, _BIG5_mbrtowc));
179 }

181 static size_t
182 _BIG5_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, _BIG5_wrtomb));
186 }

    unchanged_portion_omitted
```

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

1

```
*****
2072 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/btowc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 btowc(int c)
39 {
40     static const mbstate_t initial = { 0 };
41     mbstate_t mbs = initial;
42     char cc;
43     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)
53     if (__mbrtowc(&wc, &cc, 1, &mbs) > 1)
54         return (WEOF);
55     return (wc);

57 wint_t
58 btowc(int c)
```

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

2

```
59 {
60     return (btowc_l(c, uselocale(NULL)));
61 }
_____unchanged_portion_omitted_____
```

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

1

```
*****
14344 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/collate.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

1 /*
2  * Copyright 2014 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
2  * 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 <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 */
```

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

2

```
59 struct lc_collate lc_collate_posix = {
60     .lc_is_posix = 1,
61 };
56 static collate_subst_t      *subst_table[COLL_WEIGHTS_MAX];
57 static collate_char_t        *char_pri_table;
58 static collate_large_t       *large_pri_table;
59 static collate_chain_t        *chain_pri_table;
60 static char                   *cache = NULL;
61 static size_t                 cachesz;
62 static char                   collate_encoding[ENCODING_LEN + 1];

63 struct locdata __posix_collate_locdata = {
64     .l_lname = "C",
65     .l_refcnt = (uint32_t)-1,
66     .l_data = { &lc_collate_posix }
67 };
64 /* Exposed externally to other parts of libc. */
65 collate_info_t                *_collate_info;
66 int _collate_load_error = 1;

70 struct locdata *
71 __lc_collate_load(const char *locname)
69 int
70 __collate_load_tables(const char *encoding)
72 {
73     int i, chains, z;
74     char buf[PATH_MAX];
75     char *TMP;
76     char *map;
77     collate_info_t *info;
78     struct stat sbuf;
79     int fd;
80     struct locdata *ldata;
81     struct lc_collate *lcc;

80     /* 'encoding' must be already checked. */
81     if (strcmp(encoding, "C") == 0 || strcmp(encoding, "POSIX") == 0) {
82         _collate_load_error = 1;
83         return (_LDP_CACHE);
84     }

83     /*
87      * If the locale name is the same as our cache, use the cache.
88      */
89     if (cache && (strcmp(encoding, collate_encoding, ENCODING_LEN) == 0)) {
90         _collate_load_error = 0;
91         return (_LDP_CACHE);
92     }

94     /*
84      * Slurp the locale file into the cache.
85      */

87     (void) snprintf(buf, sizeof (buf), "%s/%s/LC_COLLATE/LCL_DATA",
88                     _PathLocale, locname);
99     _PathLocale, encoding);

90     if ((fd = open(buf, O_RDONLY)) < 0) {
91         errno = EINVAL;
92         return (NULL);
93     }
101     if ((fd = open(buf, O_RDONLY)) < 0)
102         return (_LDP_ERROR);
94     if (fstat(fd, &sbuf) < 0) {
95         (void) close(fd);
```

```

96         errno = EINVAL;
97         return (NULL);
105        return (_LDP_ERROR);
98    }
99    if (sbuf.st_size < (COLLATE_STR_LEN + sizeof (info))) {
100        (void) close(fd);
101        errno = EINVAL;
102        return (NULL);
103        return (_LDP_ERROR);
104    }
105    map = mmap(NULL, sbuf.st_size, PROT_READ, MAP_PRIVATE, fd, 0);
106    (void) close(fd);
107    if ((TMP = map) == NULL) {
108        errno = EINVAL;
109        return (NULL);
110        return (_LDP_ERROR);
111    }
112    if (strncmp(TMP, COLLATE_VERSION, COLLATE_STR_LEN) != 0) {
113        (void) munmap(map, sbuf.st_size);
114        errno = EINVAL;
115        return (NULL);
116        return (_LDP_ERROR);
117    }
118    TMP += COLLATE_STR_LEN;
119
120    info = (void *)TMP;
121    TMP += sizeof (*info);
122
123    if ((info->directive_count < 1) ||
124        (info->directive_count >= COLL_WEIGHTS_MAX) ||
125        ((chains = info->chain_count) < 0)) {
126        (void) munmap(map, sbuf.st_size);
127        errno = EINVAL;
128        return (NULL);
129        return (_LDP_ERROR);
130    }
131
132    i = (sizeof (collate_char_t) * (UCHAR_MAX + 1)) +
133        (sizeof (collate_chain_t) * chains) +
134        (sizeof (collate_large_t) * info->large_count);
135    for (z = 0; z < info->directive_count; z++) {
136        for (z = 0; z < (info->directive_count); z++) {
137            i += sizeof (collate_subst_t) * info->subst_count[z];
138        }
139    }
140    if (i != (sbuf.st_size - (TMP - map))) {
141        (void) munmap(map, sbuf.st_size);
142        errno = EINVAL;
143        return (NULL);
144        return (_LDP_ERROR);
145    }
146
147    if ((ldata = __locdata_alloc(locname, sizeof (*lcc))) == NULL) {
148        (void) munmap(map, sbuf.st_size);
149        return (NULL);
150    }
151    lcc = ldata->l_data[0];
152    ldata->l_map = map;
153    ldata->l_map_len = sbuf.st_size;
154
155    lcc->lc_info = info;
156    lcc->lc_directive_count = info->directive_count;
157    lcc->lc_large_count = info->large_count;
158
159    for (z = 0; z < COLL_WEIGHTS_MAX; z++) {

```

```

155        lcc->lc_directive[z] = info->directive[z];
156        lcc->lc_subst_count[z] = info->subst_count[z];
157        lcc->lc_pri_count[z] = info->pri_count[z];
158        lcc->lc_undef_pri[z] = info->undef_pri[z];
159    }
160
161    lcc->lc_char_table = (void *)TMP;
162    char_pri_table = (void *)TMP;
163    TMP += sizeof (collate_char_t) * (UCHAR_MAX + 1);
164
165    for (z = 0; z < lcc->lc_directive_count; z++) {
166        int count;
167        if ((count = lcc->lc_subst_count[z]) > 0) {
168            lcc->lc_subst_table[z] = (void *)TMP;
169            TMP += count * sizeof (collate_subst_t);
170            for (z = 0; z < info->directive_count; z++) {
171                if (info->subst_count[z] > 0) {
172                    subst_table[z] = (void *)TMP;
173                    TMP += info->subst_count[z] * sizeof (collate_subst_t);
174                } else {
175                    lcc->lc_subst_table[z] = NULL;
176                    subst_table[z] = NULL;
177                }
178            }
179        }
180
181        if (chains > 0) {
182            lcc->lc_chain_table = (void *)TMP;
183            chain_pri_table = (void *)TMP;
184            TMP += chains * sizeof (collate_chain_t);
185        } else {
186            lcc->lc_chain_table = NULL;
187            lcc->lc_chain_count = chains;
188            if (lcc->lc_large_count > 0) {
189                lcc->lc_large_table = (void *)TMP;
190                chain_pri_table = NULL;
191                if (info->large_count > 0) {
192                    large_pri_table = (void *)TMP;
193                } else {
194                    lcc->lc_large_table = NULL;
195                    large_pri_table = NULL;
196                }
197            }
198
199            return (ldata);
200            (void) strncpy(collate_encoding, encoding, ENCODING_LEN);
201            _collate_info = info;
202
203            if (cache)
204                (void) munmap(cache, cachesz);
205
206            cache = map;
207            cachesz = sbuf.st_size;
208            _collate_load_error = 0;
209
210            return (_LDP_LOADED);
211        }
212    }
213
214    static const int32_t *
215    substsearch(const struct lc_collate *lcc, const wchar_t key, int pass)
216    static int32_t *
217    substsearch(const wchar_t key, int pass)
218    {
219        const collate_subst_t *p;
220        int n = lcc->lc_subst_count[pass];
221        collate_subst_t *p;
222        int n = _collate_info->subst_count[pass];
223
224        if (n == 0)

```

```

195         return (NULL);

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

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

203     p = lcc->lc_subst_table[pass] + (key & ~COLLATE_SUBST_PRIORITY);
198     p = subst_table[pass] + (key & ~COLLATE_SUBST_PRIORITY);
204     assert(p->key == key);
205     return (p->pri);
206 }

208 /*
209  * Note: for performance reasons, we have expanded bsearch here. This avoids
210  * function call overhead with each comparison.
211  */

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

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

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

229     while (low <= high) {
230         next = (low + high) / 2;
231         p = tab + next;
232         compar = *key - *p->str;
233         if (compar == 0) {
234             l = wcsnlen(p->str, COLLATE_STR_LEN);
235             compar = wcsncmp(key, p->str, l);
236             if (compar == 0) {
237                 *len = l;
238                 return (p);
239             }
240         }
241         if (compar > 0)
242             low = next + 1;
243         else
244             high = next - 1;
245     }
246     return (NULL);
247 }

249 static collate_large_t *
250 largesearch(const struct lc_collate *lcc, const wchar_t key)
245 largesearch(const wchar_t key)
251 {
252     int low = 0;
253     int high = lcc->lc_info->large_count - 1;

```

```

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

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

261     while (low <= high) {
262         next = (low + high) / 2;
263         p = tab + next;
264         compar = key - p->val;
265         if (compar == 0)
266             return (p);
267         if (compar > 0)
268             low = next + 1;
269         else
270             high = next - 1;
271     }
272     return (NULL);
273 }

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

284     /*
285      * If this is the "last" pass for the UNDEFINED, then
286      * we just return the priority itself.
287      */
288     if (which >= lcc->lc_directive_count) {
283     if (which >= info->directive_count) {
289         *pri = *t;
290         *len = 1;
291         *state = NULL;
292         return;
293     }

295     /*
296      * If we have remaining substitution data from a previous
297      * call, consume it first.
298      */
299     if ((sptr = *state) != NULL) {
300         *pri = *sptr;
301         sptr++;
302         *state = *sptr ? sptr : NULL;
303         *len = 0;
304         return;
305     }

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

310     /*
311      * Check for composites such as diphthongs that collate as a
312      * single element (aka chains or collating-elements).

```

```

313     */
314     if (((p2 = chainsearch(lcc, t, &l)) != NULL) &&
315         if (((p2 = chainsearch(t, &l)) != NULL) &&
316             ((p = p2->pri[which]) >= 0)) {
317         *len = 1;
318         *pri = p;
319
320     } else if (*t <= UCHAR_MAX) {
321
322         /*
323          * Character is a small (8-bit) character.
324          * We just look these up directly for speed.
325          */
326         *pri = lcc->lc_char_table[*t].pri[which];
327         *pri = char_pri_table[*t].pri[which];
328
329     } else if ((lcc->lc_info->large_count > 0) &&
330               ((match = largesearch(lcc, *t)) != NULL)) {
331     } else if ((info->large_count > 0) &&
332               ((match = largesearch(*t)) != NULL)) {
333
334         /*
335          * Character was found in the extended table.
336          */
337         *pri = match->pri.pri[which];
338
339     } else {
340         /*
341          * Character lacks a specific definition.
342          */
343         if (lcc->lc_directive[which] & DIRECTIVE_UNDEFINED) {
344             if (info->directive[which] & DIRECTIVE_UNDEFINED) {
345                 /* Mask off sign bit to prevent ordering confusion. */
346                 *pri = (*t & COLLATE_MAX_PRIORITY);
347             } else {
348                 *pri = lcc->lc_undef_pri[which];
349                 *pri = info->undef_pri[which];
350             }
351         }
352         /* No substitutions for undefined characters! */
353         return;
354     }
355
356     /*
357     * Try substituting (expanding) the character. We are
358     * currently doing this *after* the chain compression. I
359     * think it should not matter, but this way might be slightly
360     * faster.
361     *
362     * We do this after the priority search, as this will help us
363     * to identify a single key value. In order for this to work,
364     * its important that the priority assigned to a given element
365     * to be substituted be unique for that level. The localedef
366     * code ensures this for us.
367     */
368     if ((sptr = substsearch(lcc, *pri, which)) != NULL) {
369     if ((sptr = substsearch(*pri, which)) != NULL) {
370         if ((*pri = *sptr) != 0) {
371             sptr++;
372             *state = *sptr ? sptr : NULL;
373         }
374     }
375 }
376 */

```

```

372 * This is the meaty part of wcsxfrm & strxfrm. Note that it does
373 * NOT NULL terminate. That is left to the caller.
374 */
375 size_t
376 _collate_wxfrm(const struct lc_collate *lcc, const wchar_t *src, wchar_t *xf,
377               size_t room)
378 _collate_wxfrm(const wchar_t *src, wchar_t *xf, size_t room)
379 {
380     int pri;
381     int len;
382     const wchar_t *t;
383     wchar_t *tr = NULL;
384     int direc;
385     int pass;
386     const int32_t *state;
387     int32_t *state;
388     size_t want = 0;
389     size_t need = 0;
390     int ndir = lcc->lc_directive_count;
391
392     assert(src);
393
394     for (pass = 0; pass <= ndir; pass++) {
395     for (pass = 0; pass <= _collate_info->directive_count; pass++) {
396
397         state = NULL;
398
399         if (pass != 0) {
400             /* insert level separator from the previous pass */
401             if (room) {
402                 *xf++ = 1;
403                 room--;
404             }
405             want++;
406         }
407
408         /* special pass for undefined */
409         if (pass == ndir) {
410             if (pass == _collate_info->directive_count) {
411                 direc = DIRECTIVE_FORWARD | DIRECTIVE_UNDEFINED;
412             } else {
413                 direc = lcc->lc_directive[pass];
414                 direc = _collate_info->directive[pass];
415             }
416         }
417
418         t = src;
419
420         if (direc & DIRECTIVE_BACKWARD) {
421             wchar_t *bp, *fp, c;
422             if (tr)
423                 free(tr);
424             if ((tr = wcsdup(t)) == NULL) {
425                 errno = ENOMEM;
426                 goto fail;
427             }
428             bp = tr;
429             fp = tr + wcslen(tr) - 1;
430             while (bp < fp) {
431                 c = *bp;
432                 *bp++ = *fp;
433                 *fp-- = c;
434             }
435             t = (const wchar_t *)tr;
436         }
437
438         if (direc & DIRECTIVE_POSITION) {

```

```

433         while (*t || state) {
434             _collate_lookup(lcc, t, &len, &pri, pass,
435                             &state);
436             _collate_lookup(t, &len, &pri, pass, &state);
437             t += len;
438             if (pri <= 0) {
439                 if (pri < 0) {
440                     errno = EINVAL;
441                     goto fail;
442                 }
443                 pri = COLLATE_MAX_PRIORITY;
444             }
445             if (room) {
446                 *xf++ = pri;
447                 room--;
448             }
449             want++;
450             need = want;
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     }
473 end:
474     if (tr)
475         free(tr);
476     return (need);
477
478 fail:
479     if (tr)
480         free(tr);
481     return ((size_t)(-1));
482 }
483
484 /*
485  * In the non-POSIX case, we transform each character into a string of
486  * characters representing the character's priority. Since char is usually
487  * signed, we are limited by 7 bits per byte. To avoid zero, we need to add
488  * XFRM_OFFSET, so we can't use a full 7 bits. For simplicity, we choose 6
489  * bits per byte.
490  *
491  * It turns out that we sometimes have real priorities that are
492  * 31-bits wide. (But: be careful using priorities where the high
493  * order bit is set -- i.e. the priority is negative. The sort order
494  * may be surprising!)
495  *
496  * TODO: This would be a good area to optimize somewhat. It turns out

```

```

497  * that real priorities *except for the last UNDEFINED pass* are generally
498  * very small. We need the localedef code to precalculate the max
499  * priority for us, and ideally also give us a mask, and then we could
500  * severely limit what we expand to.
501  */
502 #define XFRM_BYTES        6
503 #define XFRM_OFFSET       ('0') /* make all printable characters */
504 #define XFRM_SHIFT        6
505 #define XFRM_MASK         ((1 << XFRM_SHIFT) - 1)
506 #define XFRM_SEP          ('.') /* chosen to be less than XFRM_OFFSET */
507
508 static int
509 xfrm(locale_t loc, 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)loc->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 {
529     _collate_sxfrm(const wchar_t *src, char *xf, size_t room)
530     {
531         int pri;
532         int len;
533         const wchar_t *t;
534         wchar_t *tr = NULL;
535         int direc;
536         int pass;
537         const int32_t *state;
538         int32_t *state;
539         size_t want = 0;
540         size_t need = 0;
541         int b;
542         uint8_t buf[XFRM_BYTES];
543         const struct lc_collate *lcc = loc->collate;
544         int ndir = lcc->lc_directive_count;
545
546         assert(src);
547
548         for (pass = 0; pass <= ndir; pass++) {
549             for (pass = 0; pass <= _collate_info->directive_count; pass++) {
550
551                 state = NULL;
552
553                 if (pass != 0) {
554                     /* insert level separator from the previous pass */
555                     if (room) {
556                         *xf++ = XFRM_SEP;
557                         room--;
558                     }
559                     want++;
560                 }
561
562                 /* special pass for undefined */

```

```

558     if (pass == ndir) {
559         if (pass == _collate_info->directive_count) {
560             direc = DIRECTIVE_FORWARD | DIRECTIVE_UNDEFINED;
561         } else {
562             direc = lcc->lc_directive[pass];
563             direc = _collate_info->directive[pass];
564         }

565     t = src;

566     if (direc & DIRECTIVE_BACKWARD) {
567         wchar_t *bp, *fp, c;
568         if (tr)
569             free(tr);
570         if ((tr = wcsdup(t)) == NULL) {
571             errno = ENOMEM;
572             goto fail;
573         }
574         bp = tr;
575         fp = tr + wcslen(tr) - 1;
576         while (bp < fp) {
577             c = *bp;
578             *bp++ = *fp;
579             *fp-- = c;
580         }
581         t = (const wchar_t *)tr;
582     }

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

```

```

619             goto fail;
620         }
621         continue;
622     }

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         need = want;
635     }
636     }
637 }
638 }
639 }

640 end:
641 if (tr)
642     free(tr);
643 return (need);
644 }

645 fail:
646 if (tr)
647     free(tr);
648 return ((size_t)(-1));
649 }
650 }

```

unchanged_portion_omitted

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

1

```
*****
4322 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/collate.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 #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
59  * collate_chain_pri_t  chains[*];              // 0 or more chains
60  * collate_large_pri_t  large[*];              // extended char priorities

```

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

2

```
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_wxfrm(const wchar_t *, wchar_t *, size_t);
103     size_t      _collate_sxfrm(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_wxfrm(const struct lc_collate *, const wchar_t *,
120     wchar_t *, size_t);
121 size_t      _collate_sxfrm(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 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/collcmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 {
40     wchar_t s1[2], s2[2];
41     static wchar_t s1[2], s2[2];
42     s1[0] = c1;
43     s1[1] = 0;
44     s2[0] = c2;
45     s2[1] = 0;
46     return (wcscoll_l(s1, s2, loc));
47     return (wcscoll(s1, s2));
48 }
49
50 _unchanged_portion_omitted_
```

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

1

```
*****
11749 Tue May 20 07:58:46 2014
new/usr/src/lib/libc/port/locale/euc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */

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"

47 static size_t _EUC_mbrtowc_impl(wchar_t *_RESTRICT_KYWD,
48 const char *_RESTRICT_KYWD,
49 size_t, mbstate_t *_RESTRICT_KYWD, uint8_t, uint8_t, uint8_t, uint8_t);
50 static size_t _EUC_wrtomb_impl(char *_RESTRICT_KYWD, wchar_t,
51 mbstate_t *_RESTRICT_KYWD, uint8_t, uint8_t, uint8_t, uint8_t);

53 static size_t _EUC_CN_mbrtowc(wchar_t *_RESTRICT_KYWD,
54 const char *_RESTRICT_KYWD,
55 size_t, mbstate_t *_RESTRICT_KYWD);
56 static size_t _EUC_JP_mbrtowc(wchar_t *_RESTRICT_KYWD,
57 const char *_RESTRICT_KYWD,
58 size_t, mbstate_t *_RESTRICT_KYWD);
59 static size_t _EUC_KR_mbrtowc(wchar_t *_RESTRICT_KYWD,
```

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

2

```
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);

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);

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);

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);

101 static int _EUC_mbsinit(const mbstate_t *);

103 typedef struct {
104 wchar_t ch;
105 int set;
106 int want;
107 } _EucState;

109 int
110 static int
111 _EUC_mbsinit(const mbstate_t *ps)
112 {
113 return (ps == NULL || ((const _EucState *)ps)->want == 0);
114 }

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 *rl)
124 {
125 lct->lc_mbrtowc = _EUC_CN_mbrtowc;
```

```

123     lct->lc_wctomb = _EUC_CN_wctomb;
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     __wctomb = _EUC_CN_wctomb;
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 * _RESTRICT_KYWD dst,
141     const char ** _RESTRICT_KYWD src,
142     size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
143 {
144     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_CN_mbrtowc));
145 }

147 static size_t
148 _EUC_CN_wctomb(char * _RESTRICT_KYWD s, wchar_t wc,
149     mbstate_t * _RESTRICT_KYWD ps)
150 {
151     return (_EUC_wctomb_impl(s, wc, ps, SS2, 4, 0, 0));
152 }

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

161 /*
162  * EUC-KR uses only CS0 and CS1.
163  */
164 void
165 _EUC_KR_init(struct lc_ctype *lct)
166 {
167     int
168     _EUC_KR_init(_RuneLocale *rl)
169 {
170     lct->lc_mbrtowc = _EUC_KR_mbrtowc;
171     lct->lc_wctomb = _EUC_KR_wctomb;
172     lct->lc_mbsnrtowcs = _EUC_KR_mbsnrtowcs;
173     lct->lc_wcsnrtombs = _EUC_KR_wcsnrtombs;
174     lct->lc_mbsinit = _EUC_mbsinit;
175     __mbrtowc = _EUC_KR_mbrtowc;
176     __wctomb = _EUC_KR_wctomb;
177     __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 * _RESTRICT_KYWD dst,
186     const char ** _RESTRICT_KYWD src,
187     size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
188 {
189     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_KR_mbrtowc));
190 }

192 static size_t
193 _EUC_KR_wctomb(char * _RESTRICT_KYWD s, wchar_t wc,
194     mbstate_t * _RESTRICT_KYWD ps)
195 {
196     return (_EUC_wctomb_impl(s, wc, ps, 0, 0, 0, 0));
197 }

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

206 /*
207  * EUC-JP uses CS0, CS1, CS2, and CS3.
208  */
209 void
210 _EUC_JP_init(struct lc_ctype *lct)
211 {
212     int
213     _EUC_JP_init(_RuneLocale *rl)
214 {
215     lct->lc_mbrtowc = _EUC_JP_mbrtowc;
216     lct->lc_wctomb = _EUC_JP_wctomb;
217     lct->lc_mbsnrtowcs = _EUC_JP_mbsnrtowcs;
218     lct->lc_wcsnrtombs = _EUC_JP_wcsnrtombs;
219     lct->lc_mbsinit = _EUC_mbsinit;
220     __mbrtowc = _EUC_JP_mbrtowc;
221     __wctomb = _EUC_JP_wctomb;
222     __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 * _RESTRICT_KYWD dst,
231     const char ** _RESTRICT_KYWD src,
232     size_t nms, size_t len, mbstate_t * _RESTRICT_KYWD ps)
233 {
234     return (__mbsnrtowcs_std(dst, src, nms, len, ps, _EUC_JP_mbrtowc));
235 }

237 static size_t
238 _EUC_JP_wctomb(char * _RESTRICT_KYWD s, wchar_t wc,
239     mbstate_t * _RESTRICT_KYWD ps)
240 {
241     return (_EUC_wctomb_impl(s, wc, ps, SS2, 2, SS3, 3));
242 }

244 static size_t

```

```

245 _EUC_JP_wcsnrtombs(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 (__wcsnrtombs_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_wrtomb = _EUC_TW_wrtomb;
259     lct->lc_mbsnrtowcs = _EUC_TW_mbsnrtowcs;
260     lct->lc_wcsnrtombs = _EUC_TW_wcsnrtombs;
261     lct->lc_mbsinit = _EUC_mbsinit;
262     __mbrtowc = _EUC_TW_mbrtowc;
263     __wrtomb = _EUC_TW_wrtomb;
264     __mbsinit = _EUC_mbsinit;

263     lct->lc_max_mblen = 4;
264     lct->lc_is_ascii = 0;
190     __CurrentRuneLocale = rl;

192     __ctype[520] = 4;
193     charset_is_ascii = 0;
194     return (0);
265 }
    unchanged_portion_omitted

274 static size_t
275 _EUC_TW_mbsnrtowcs(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 (__mbsnrtowcs_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_wcsnrtombs(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 (__wcsnrtombs_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

```

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

1

```
*****
3763 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/fgetwc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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) {
```

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

2

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

86 /*
87  * If we got here it means we got truncated in a character, or
88  * the character did not decode properly. Note that in the case
89  * of a botched decoding, we don't UNGETC the bad bytes. Should
90  * we?
91  */
92 fp->_flag |= _IOERR;
93 errno = EILSEQ;
94 return (WEOF);
95 }

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     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 {
149     __getwc_xpg5(FILE *fp)
150 {
151     wint_t      r;
152     rmutex_t    *l;
153     FLOCKFILE(l, fp);
154     if (GET_NO_MODE(fp))
155         _setorientation(fp, _WC_MODE);
156     r = _fgetwc_unlocked_l(fp, loc);
157     FUNLOCKFILE(l);

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

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

1

```
*****
8483 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/fnmatch.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */

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 */

39 /*
40  * Function fnmatch() as specified in POSIX 1003.2-1992, section B.6.
41  * Compares a filename or pathname to a pattern.
42 */

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 */

55 #include "lint.h"
56 #include <fnmatch.h>
57 #include <limits.h>
58 #include <string.h>
59 #include <wchar.h>
60 #include <xlocale.h>
```

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

2

```
61 #include <wctype.h>
62 #include "localeimpl.h"

63 #include "collate.h"

65 #define EOS      '\\0'

67 #define RANGE_MATCH      1
68 #define RANGE_NOMATCH    0
69 #define RANGE_ERROR      (-1)

71 static int rangematch(const char *, wchar_t, int, char **, mbstate_t *,
72                      locale_t);
73 static int rangematch(const char *, wchar_t, int, char **, mbstate_t *);
74 static int fnmatch1(const char *, const char *, const char *, int, mbstate_t,
75                    mbstate_t, locale_t);
76 static int fnmatch1(const char *, const char *, const char *, int, mbstate_t,
77                    mbstate_t, locale_t);

76 int
77 fnmatch(pattern, string, flags)
78     const char *pattern, *string;
79     int flags;
80 {
81     locale_t loc = uselocale(NULL);
82     static const mbstate_t initial = { 0 };

84     return (fnmatch1(pattern, string, string, flags, initial, initial,
85                      loc));
86     return (fnmatch1(pattern, string, string, flags, initial, initial));
87 }

88 static int
89 fnmatch1(const char *pattern, const char *string, const char *stringstart,
90          int flags, mbstate_t patmb, mbstate_t strmb, locale_t loc)
91 {
92     int flags, mbstate_t patmb, mbstate_t strmb;
93     char *newp;
94     char c;
95     wchar_t pc, sc;
96     size_t pflen, sflen;

97     for (;;) {
98         pflen = mbrtowc_l(&pc, pattern, MB_LEN_MAX, &patmb, loc);
99         pflen = mbrtowc(&pc, pattern, MB_LEN_MAX, &patmb);
100         if (pflen == (size_t)-1 || pflen == (size_t)-2)
101             return (FNM_NOMATCH);
102         pattern += pflen;
103         sflen = mbrtowc_l(&sc, string, MB_LEN_MAX, &strmb, loc);
104         sflen = mbrtowc(&sc, string, MB_LEN_MAX, &strmb);
105         if (sflen == (size_t)-1 || sflen == (size_t)-2) {
106             sc = (unsigned char)*string;
107             sflen = 1;
108             (void) memset(&strmb, 0, sizeof (strmb));
109         }
110         switch (pc) {
111             case EOS:
112                 /*
113                  * Removed FNM_LEADING_DIR, as it is not present
114                  * on Solaris.
115                  */
116                 return (sc == EOS ? 0 : FNM_NOMATCH);
117             case '?':
118                 if (sc == EOS)
119                     return (FNM_NOMATCH);
120                 if (sc == '/' && (flags & FNM_PATHNAME))
121                     return (FNM_NOMATCH);
122             default:
123                 if (pc != sc)
124                     return (FNM_NOMATCH);
125         }
126         string += sflen;
127     }
128     return (0);
129 }
```

```

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, strmb, loc))
154                 return (0);
155             sclen = mbrtowc_l(&sc, string, MB_LEN_MAX,
156                             &strmb, loc);
157             sclen = mbrtowc(&sc, string, MB_LEN_MAX,
158                             &strmb);
159             if (sclen == (size_t)-1 ||
160                 sclen == (size_t)-2) {
161                 sc = (unsigned char)*string;
162                 sclen = 1;
163                 (void) memset(&strmb, 0,
164                             sizeof (strmb));
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)) {
183             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             pcrlen = mbrtowc_l(&pc, pattern, MB_LEN_MAX,
194                             &patmbs, loc);
195             pcrlen = mbrtowc(&pc, pattern, MB_LEN_MAX,
196                             &patmbs);
197             if (pcrlen == (size_t)-1 || pcrlen == (size_t)-2)
198                 return (FNM_NOMATCH);
199             if (pcrlen == 0)
200                 pc = '\\';
201             pattern += pcrlen;
202             /* FALLTHROUGH */
203         }
204     default:
205     norm:
206         if (pc == sc)
207             string += sclen;
208
209         else if ((flags & FNM_IGNORECASE) &&
210                 (tolower_l(pc, loc) == tolower_l(sc, loc)))
211             string += sclen;
212         else
213             return (FNM_NOMATCH);
214
215         break;
216     }
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 {
223     const char *pattern;
224     wchar_t test;
225     int flags;
226     char **newp;
227     mbstate_t *patmbs;
228
229     {
230         int negate, ok;
231         wchar_t c, c2;
232         size_t pcrlen;
233         const char *origpat;
234
235         /*
236          * A bracket expression starting with an unquoted circumflex
237          * character produces unspecified results (IEEE 1003.2-1992,
238          * 3.13.2). This implementation treats it like '!', for
239          * consistency with the regular expression syntax.
240          * J.T. Conklin (conklin@gai.kaleida.com)
241          */
242         if ((negate = (*pattern == '!') || *pattern == '^') != 0)
243             ++pattern;
244
245         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);
259             pflen = mbrtowc(&c, pattern, MB_LEN_MAX, patmbs);
260             if (pflen == (size_t)-1 || pflen == (size_t)-2)
261                 return (RANGE_NOMATCH);
262             pattern += pflen;
263         }
264         if (flags & FNM_IGNORECASE)
265             c = towlower_l(c, loc);
266             c = towlower(c);

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

280             if (flags & FNM_IGNORECASE)
281                 c2 = towlower_l(c2, loc);
282                 c2 = towlower(c2);

283             if (loc->collate->lc_is_posix ?
284                 if (_collate_load_error ?
285                     c <= test && test <= c2 :
286                     _collate_range_cmp(c, test, loc) <= 0 &&
287                     _collate_range_cmp(test, c2, loc) <= 0)
288                     _collate_range_cmp(c, test) <= 0 &&
289                     _collate_range_cmp(test, c2) <= 0)
290                 ok = 1;
291             } else if (c == test)
292                 ok = 1;
293         }
294     }
295     *newp = (char *)pattern;
296     return (ok == negate ? RANGE_NOMATCH : RANGE_MATCH);
297 }

```

unchanged portion omitted

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

1

```
*****
3811 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/fputwc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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.
```

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

2

```
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     }

93     FLOCKFILE(mx, fp);
94     /*
95      * This is used for XPG 5 semantics, which requires the stream
96      * orientation to be changed when the function is called.
97      */
98     if (orient && GET_NO_MODE(fp)) {
99         _setorientation(fp, _WC_MODE);
100     }
101     for (i = 0; i < len; i++) {
102         if (PUTC((unsigned char)buf[i], fp) == EOF) {
103             FUNLOCKFILE(mx);
104             return (WEOF);
105         }
106     }
107     FUNLOCKFILE(mx);
108     return ((wint_t)wc);
109 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libc/port/locale/gb18030.c

1

```
*****
6217 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/gb18030.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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"
43 #include "lctype.h"

45 static size_t _GB18030_mbrtowc(wchar_t *_RESTRIC_KYWD,
46                               const char *_RESTRIC_KYWD,
47                               size_t, mbstate_t *_RESTRIC_KYWD);
48 static int _GB18030_mbsinit(const mbstate_t *);
49 static size_t _GB18030_wrtomb(char *_RESTRIC_KYWD, wchar_t,
50                               mbstate_t *_RESTRIC_KYWD);
51 static size_t _GB18030_mbsnrtowcs(wchar_t *_RESTRIC_KYWD,
52                                   const char **_RESTRIC_KYWD, size_t, size_t,
53                                   mbstate_t *_RESTRIC_KYWD);
54 static size_t _GB18030_wcsnrtombs(char *_RESTRIC_KYWD,
55                                   const wchar_t **_RESTRIC_KYWD, size_t, size_t,
56                                   mbstate_t *_RESTRIC_KYWD);

59 typedef struct {
```

new/usr/src/lib/libc/port/locale/gb18030.c

2

```
60     int count;
61     uchar_t bytes[4];
62 } _GB18030State;

64 void
65 _GB18030_init(struct lc_ctype *lct)
66 {
67     _GB18030_init(_RuneLocale *rl)
68 {
69     lct->lc_mbrtowc = _GB18030_mbrtowc;
70     lct->lc_wrtomb = _GB18030_wrtomb;
71     lct->lc_mbsinit = _GB18030_mbsinit;
72     lct->lc_mbsnrtowcs = _GB18030_mbsnrtowcs;
73     lct->lc_wcsnrtombs = _GB18030_wcsnrtombs;
74     lct->lc_max_mblen = 4;
75     lct->lc_is_ascii = 0;
76     __mbrtowc = _GB18030_mbrtowc;
77     __wrtomb = _GB18030_wrtomb;
78     __mbsinit = _GB18030_mbsinit;
79     __CurrentRuneLocale = rl;
80     __ctype[520] = 4;
81     charset_is_ascii = 0;
82
83     return (0);
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 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
201 }
202 }
203 }
204 }
205 }
```

```
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 }
_unchanged_portion_omitted_
```

new/usr/src/lib/libc/port/locale/gb2312.c

1

```
*****
4907 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/gb2312.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

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

59 void
60 _GB2312_init(struct lc_ctype *lct)
```

new/usr/src/lib/libc/port/locale/gb2312.c

2

```
50 int
51 _GB2312_init(_RuneLocale *rl)
61 {

63     lct->lc_mbrtowc = _GB2312_mbrtowc;
64     lct->lc_wcrtomb = _GB2312_wcrtomb;
65     lct->lc_mbsinit = _GB2312_mbsinit;
66     lct->lc_mbsnrtowcs = _GB2312_mbsnrtowcs;
67     lct->lc_wcsnrtombs = _GB2312_wcsnrtombs;
68     lct->lc_max_mblen = 2;
69     lct->lc_is_ascii = 0;
54     _CurrentRuneLocale = rl;
55     _mbrtowc = _GB2312_mbrtowc;
56     _wcrtomb = _GB2312_wcrtomb;
57     _mbsinit = _GB2312_mbsinit;
58     _ctype[520] = 2;
59     charset_is_ascii = 0;
60     return (0);
70 }

unchanged_portion_omitted

143 static size_t
144 _GB2312_wcrtomb(char *_RESTRIC_KYWD s, wchar_t wc,
145                mbstate_t *_RESTRIC_KYWD ps)
146 {
147     _GB2312State *gs;

149     gs = (_GB2312State *)ps;

151     if (gs->count != 0) {
152         errno = EINVAL;
153         return ((size_t)-1);
154     }

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 *_RESTRIC_KYWD dst,
170                    const char **_RESTRIC_KYWD src, size_t nms, size_t len,
171                    mbstate_t *_RESTRIC_KYWD ps)
172 {
173     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _GB2312_mbrtowc));
174 }

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

unchanged_portion_omitted
```

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

1

```
*****
4945 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/gbk.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 <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 #include "lctype.h"

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_wctomb(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;
```

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

2

```
60 } _GBKState;

62 void
63 _GBK_init(struct lc_ctype *lct)
64 {
65     lct->lc_mbrtowc = _GBK_mbrtowc;
66     lct->lc_wctomb = _GBK_wctomb;
67     lct->lc_mbsinit = _GBK_mbsinit;
68     lct->lc_mbsnrtowcs = _GBK_mbsnrtowcs;
69     lct->lc_wcsnrtombs = _GBK_wcsnrtombs;
70     lct->lc_max_mblen = 2;
71     lct->lc_is_ascii = 0;

59     __mbrtowc = _GBK_mbrtowc;
60     __wctomb = _GBK_wctomb;
61     __mbsinit = _GBK_mbsinit;
62     __CurrentRuneLocale = rl;
63     __ctype[520] = 2;
64     charset_is_ascii = 0;
65     return (0);
72 }

unchanged_portion_omitted

150 static size_t
151 _GBK_wctomb(char *_RESTRICT_KYWD s, wchar_t wc, mbstate_t *_RESTRICT_KYWD ps)
152 {
153     _GBKState *gs;

155     gs = (_GBKState *)ps;

157     if (gs->ch != 0) {
158         errno = EINVAL;
159         return ((size_t)-1);
160     }

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 }

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 }

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_wctomb));
186 }

unchanged_portion_omitted
```

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

1

```
*****
2351 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/isdigit.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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)); \
59 } \
60
```

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

2

```
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)
```

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

1

```
*****
6314 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/iswctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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  *
55  * We aliased (per Solaris) iswideogram, iswspecial, iswspecial to the
56  * equivalent values without "w". The Solaris specific function isenglish()
57  * is here, but does not get an isw* equivalent.
58  *
59  * Note that various code assumes that "numbers" (iswdigit, iswxdigit)
60  * only return true for characters in the portable set. While the assumption
```

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

2

```
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 \
130 #undef iswalnum \
131 int \
132 iswalnum(wint_t wc) \
133 { \
134     return (__istype(wc, _CTYPE_A|_CTYPE_D)); \
135 }

```

```

131 /* kill off any macros */
132 #undef iswalnum
133 #undef iswalpha
134 int
135 iswalpha(wint_t wc)
136 {
137     return (__istype(wc, _CTYPE_A));
138 }

```

```

134 #undef iswblank
135 int
136 iswblank(wint_t wc)
137 {
138     return (__istype(wc, _CTYPE_B));
139 }

```

```

136 DEFN_ISWTYPE(alnum, _CTYPE_A|_CTYPE_D)
137 DEFN_ISWTYPE(alpha, _CTYPE_A)
138 DEFN_ISWTYPE(blank, _CTYPE_B)
139 DEFN_ISWTYPE(ctrnl, _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)
152 #undef iswcntrl
153 int
154 iswcntrl(wint_t wc)
155 {
156     return (__istype(wc, _CTYPE_C));
157 }

```

```

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
157 #undef iswgraph
158 int
159 iswgraph(wint_t wc)
160 {
161     return (__istype(wc, _CTYPE_G));
162 }

```

```

158 #undef iswideogram
159 #pragma weak iswideogram = iswideogram
160 int
161 iswideogram(wint_t wc)
162 {
163     return (__istype(wc, _CTYPE_I));
164 }

```

```

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
163 int
164 isphonogram(wint_t wc)
165 {
166     return (__istype(wc, _CTYPE_Q));
167 }

```

```

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 }

```

```

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));

```

```

232     return (__istype(wc, 0xffffffffU));
204 }
_____unchanged_portion_omitted_

```

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

1

```
*****
1891 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/lctype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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_wcrtomb)(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);
59 size_t __wcrtomb_ascii(char *_RESTRICT_KYWD, wchar_t,
60                        mbstate_t *_RESTRICT_KYWD);
```

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

2

```
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

```
*****
3762 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/ldpart.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 "ldpart.h"
42 #include "setlocale.h"
44 static int split_lines(char *, const char *);
46 int
47 __part_load_locale(const char *name,
48 __part_load_locale(const char *name, int *using_locale,
49 char **locale_buf, const char *category_filename,
50 int locale_buf_size_max, int locale_buf_size_min,
51 const char **dst_localebuf)
52 {
53     int saverr, fd, i, num_lines;
54     char *lbuf, *p;
55     const char *plim;
56     char filename[PATH_MAX];
57     struct stat st;
58     size_t namesize, bufsize;
59     /* 'name' must be already checked. */
```

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

2

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

```
110     for (i = num_lines; i < locale_buf_size_max; i++)
111         dst_localebuf[i] = NULL;
127     *using_locale = 1;

113     return (_LDP_LOADED);

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

124     return (_LDP_ERROR);
125 }
_____unchanged_portion_omitted_____
```

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

1

```
*****
1665 Tue May 20 07:58:47 2014
new/usr/src/lib/libc/port/locale/ldpart.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 *,
36 int __part_load_locale(const char *, int *, char **, const char *,
int, int, const char **);

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

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

1

```
*****
2668 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lmessages.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 *))
37 #define LCMESSAGES_SIZE_FULL (sizeof (struct lc_messages_T) / sizeof (char *))
38 #define LCMESSAGES_SIZE_MIN \
39     (offsetof(struct lc_messages, yesstr) / sizeof (char *))
40 #define LCMESSAGES_SIZE_MIN \
41     (offsetof(struct lc_messages_T, yesstr) / sizeof (char *))

40 static char empty[] = "";

42 struct lc_messages lc_messages_posix = {
43     static const struct lc_messages_T _C_messages_locale = {
44         "[yY]", /* yesexpr */
45         "[nN]", /* noexpr */
46         "yes", /* yesstr */
47         "no", /* nostr */
48     };
49 };

49 struct locdata __posix_messages_locdata = {
50     .l_name = "C",
51     .l_refcnt = (uint32_t)-1,
52     .l_data = { &lc_messages_posix }
53 };

46 static struct lc_messages_T _messages_locale;
47 static int _messages_using_locale;
48 static char *_messages_locale_buf;
```

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

2

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

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

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

70     if (ret != _LDP_LOADED) {
71         __locdata_release(ldata);
72         errno = EINVAL;
73         return (NULL);
74     }
75     ret = __part_load_locale(name, &_messages_using_locale,
76         &_messages_locale_buf, "LC_MESSAGES",
77         LCMESSAGES_SIZE_FULL, LCMESSAGES_SIZE_MIN,
78         (const char **)&_messages_locale);
79     if (ret == _LDP_LOADED) {
80         if (_messages_locale.yesstr == NULL)
81             _messages_locale.yesstr = empty;
82         if (_messages_locale.nostr == NULL)
83             _messages_locale.nostr = empty;
84     }
85     return (ldata);
86 }

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

81     return (ldata);
82 }

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);
73 }

unchanged_portion_omitted
```

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

1

```
*****
1620 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lmessages.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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_ */
```

```

*****
5262 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lmonetary.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 */
28
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 "ldpart.h"
37 #include "lmonetary.h"
38 #include "localeimpl.h"
39
40 extern int __mlocale_changed;
41 extern const char *__fix_locale_grouping_str(const char *);
42
43 #define LCMONETARY_SIZE_FULL (sizeof (struct lc_monetary) / sizeof (char *))
44 #define LCMONETARY_SIZE_FULL (sizeof (struct lc_monetary_T) / sizeof (char *))
45 #define LCMONETARY_SIZE_MIN \
46     (offsetof(struct lc_monetary, int_p_cs_precedes) / sizeof (char *))
47 #define LCMONETARY_SIZE_MIN \
48     (offsetof(struct lc_monetary_T, int_p_cs_precedes) / sizeof (char *))
49
50 static char empty[] = "";
51 static char numempty[] = { CHAR_MAX, '\0' };
52
53 struct lc_monetary lc_monetary_posix = {
54 static const struct lc_monetary_T _C_monetary_locale = {
55     empty, /* int_curr_symbol */
56     empty, /* currency_symbol */
57     empty, /* mon_decimal_point */
58     empty, /* mon_thousands_sep */
59     numempty, /* mon_grouping */
60     empty, /* positive_sign */
61     empty, /* negative_sign */

```

```

57     numempty, /* int_frac_digits */
58     numempty, /* frac_digits */
59     numempty, /* p_cs_precedes */
60     numempty, /* p_sep_by_space */
61     numempty, /* n_cs_precedes */
62     numempty, /* n_sep_by_space */
63     numempty, /* p_sign_posn */
64     numempty, /* n_sign_posn */
65     numempty, /* int_p_cs_precedes */
66     numempty, /* int_n_cs_precedes */
67     numempty, /* int_p_sep_by_space */
68     numempty, /* int_n_sep_by_space */
69     numempty, /* int_p_sign_posn */
70     numempty, /* int_n_sign_posn */
71     empty, /* crncystr */
66     numempty, /* int_n_sign_posn */
72 };
73
74 struct locdata __posix_monetary_locdata = {
75     .l_lname = "C",
76     .l_refcnt = (uint32_t)-1,
77     .l_data = { &lc_monetary_posix }
78 };
79
80 static struct lc_monetary_T _monetary_locale;
81 static int _monetary_using_locale;
82 static char *_monetary_locale_buf;
83
84 static char
85 cnv(const char *str)
86 {
87     int i = strtol(str, NULL, 10);
88
89     if (i == -1)
90         i = CHAR_MAX;
91     return ((char)i);
92 }
93
94 struct locdata *
95 __lc_monetary_load(const char *name)
96 {
97     int ret;
98     int clen;
99     struct lc_monetary *lmon;
100     struct locdata *ldata;
101
102     if ((ldata = __locdata_alloc(name, sizeof (*lmon))) == NULL) {
103         return (NULL);
104     }
105     lmon = ldata->l_data[0];
106     ret = __part_load_locale(name, &_monetary_using_locale,
107         &_monetary_locale_buf, "LC_MONETARY",
108         LCMONETARY_SIZE_FULL, LCMONETARY_SIZE_MIN,
109         (const char **)&_monetary_locale);
110     if (ret != _LDP_ERROR)
111         __mlocale_changed = 1;
112     if (ret == _LDP_LOADED) {
113         _monetary_locale.mon_grouping =
114             __fix_locale_grouping_str(_monetary_locale.mon_grouping);
115
116         ret = __part_load_locale(name, (char **)&ldata->l_data[1],
117             "LC_MONETARY", LCMONETARY_SIZE_FULL, LCMONETARY_SIZE_MIN,
118             (const char **)&lmon);
119
120         if (ret != _LDP_LOADED) {

```

```

108     __locdata_release(ldata);
109     errno = EINVAL;
110     return (NULL);
111 }

113 /* special storage for currency string */
114 clen = strlen(lmon->currency_symbol) + 2;
115 ldata->l_data[2] = malloc(clen);
116 lmon->crncystr = ldata->l_data[2];
117 lmon->crncystr[0] = '\0';

119 lmon->mon_grouping = __fix_locale_grouping_str(lmon->mon_grouping);

121 #define M_ASSIGN_CHAR(NAME) \
122     (((char *)lmon->NAME)[0] = cnv(lmon->NAME))
99     (((char *)_monetary_locale.NAME)[0] = \
100         cnv(_monetary_locale.NAME))

124 M_ASSIGN_CHAR(int_frac_digits);
125 M_ASSIGN_CHAR(frac_digits);
126 M_ASSIGN_CHAR(p_cs_precedes);
127 M_ASSIGN_CHAR(p_sep_by_space);
128 M_ASSIGN_CHAR(n_cs_precedes);
129 M_ASSIGN_CHAR(n_sep_by_space);
130 M_ASSIGN_CHAR(p_sign_posn);
131 M_ASSIGN_CHAR(n_sign_posn);

133 /*
134  * The six additional C99 international monetary formatting
135  * parameters default to the national parameters when
136  * reading FreeBSD LC_MONETARY data files.
137  */
138 #define M_ASSIGN_ICHAR(NAME) \
139     if (lmon->int_##NAME == NULL) \
140         lmon->int_##NAME = lmon->NAME; \
141         if (_monetary_locale.int_##NAME == NULL) \
142             _monetary_locale.int_##NAME = \
143                 _monetary_locale.NAME; \
144     else \
145         M_ASSIGN_CHAR(int_##NAME);

144 M_ASSIGN_ICHAR(p_cs_precedes);
145 M_ASSIGN_ICHAR(n_cs_precedes);
146 M_ASSIGN_ICHAR(p_sep_by_space);
147 M_ASSIGN_ICHAR(n_sep_by_space);
148 M_ASSIGN_ICHAR(p_sign_posn);
149 M_ASSIGN_ICHAR(n_sign_posn);

151 /*
152  * Now calculate the currency string (CRNCYSTR) for nl_langinfo.
153  * This is a legacy SUSv2 interface.
154  */
155 if ((lmon->p_cs_precedes[0] == lmon->n_cs_precedes[0]) &&
156     (lmon->currency_symbol[0] != '\0')) {
157     char sign = '\0';
158     switch (lmon->p_cs_precedes[0]) {
159     case 0:
160         sign = '-';
161         break;
162     case 1:
163         sign = '+';
164         break;
165     case CHAR_MAX:
166         /*
167          * Substitute currency string for radix character.
168          * To the best of my knowledge, no locale uses this.

```

```

169     */
170     if (strcmp(lmon->mon_decimal_point,
171         lmon->currency_symbol) == 0)
172         sign = '.';
173     break;
174 }
175 (void) snprintf(lmon->crncystr, clen, "%c%s", sign,
176     lmon->currency_symbol);
177 }
178 return (ret);
179 }

179 return (ldata);
180 struct lc_monetary_T *
181 __get_current_monetary_locale(void)
182 {
183     return (_monetary_using_locale ? &_monetary_locale :
184         (struct lc_monetary_T *)&_C_monetary_locale);
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 }

```

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

1

2235 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lmonetary.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 *);

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

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

1

```
*****
2806 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lnumeric.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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"

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

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

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

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

47 struct locdata __posix_numeric_locdata = {
48     .l_lname = "C",
49     .l_refcnt = (uint32_t)-1,
50     .l_data = { &lc_numeric_posix }
51 };

48 static struct lc_numeric_T __numeric_locale;
49 static int __numeric_using_locale;
50 static char *__numeric_locale_buf;
```

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

2

```
55 /*
56  * Return the locale's numeric locdata structure, places a hold on it,
57  * which the caller must release.
58  */
59 struct locdata *
60 __lc_numeric_load(const char *name)
61 int
62 __numeric_load_locale(const char *name)
63 {
64     struct locdata *ldata;
65     struct lc_numeric *lnum;
66     const struct lc_numeric_T *leg = &__c_numeric_locale;

67     int ret;

68     if ((ldata = __locdata_alloc(name, sizeof (*lnum))) == NULL) {
69         errno = EINVAL;
70         return (NULL);
71     }
72     lnum = ldata->l_data[0];
73     ret = __part_load_locale(name, &__numeric_using_locale,
74         &__numeric_locale_buf, "LC_NUMERIC", LCNUMERIC_SIZE, LCNUMERIC_SIZE,
75         (const char **) &__numeric_locale);
76     if (ret == _LDP_ERROR)
77         return (_LDP_ERROR);

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

80     if (ret != _LDP_LOADED) {
81         __locdata_release(ldata);
82         return (NULL);
83     }
84     __nlocale_changed = 1;
85     if (ret == _LDP_LOADED) {
86         /* Can't be empty according to C99 */
87         if (*__numeric_locale.decimal_point == '\0')
88             __numeric_locale.decimal_point =
89                 __c_numeric_locale.decimal_point;
90         __numeric_locale.grouping =
91             __fix_locale_grouping_str(__numeric_locale.grouping);
92         leg = (const struct lc_numeric_T *) &__numeric_locale;
93     }
94     /* This is Solaris legacy, required for ABI compatability */
95     __numeric[0] = *leg->decimal_point;
96     __numeric[1] = *leg->grouping;
97     return (ret);
98 }

99 /* Can't be empty according to C99 */
100 if (*lnum->decimal_point == '\0')
101     lnum->decimal_point = lc_numeric_posix.decimal_point;
102 lnum->grouping = __fix_locale_grouping_str(lnum->grouping);

103 return (ldata);
104 struct lc_numeric_T *
105 __get_current_numeric_locale(void)
106 {
107     return (__numeric_using_locale ? &__numeric_locale :
108         (struct lc_numeric_T *) &__c_numeric_locale);
109 }

__unchanged_portion_omitted
```

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

1

```
*****
1611 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/lnumeric.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/localeconv.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 */
32
33 #ifndef _LCONV_C99
34 #define _LCONV_C99 /* so we get all the extensions */
35 #endif
36
37 #include "lint.h"
38 #include <locale.h>
39 #include "lmonetary.h"
40 #include "lnumeric.h"
41 #include "localeimpl.h"
42
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  */
57 int __mlocale_changed = 1;
58 int __nlocale_changed = 1;
59
60 /*

```

```

54  * Return the current locale conversion.
55  */
56 struct lconv *
57 localeconv(void)
58 {
59     struct lconv *lconv;
60     locale_t loc;
61     struct lc_monetary *mptr;
62     struct lc_numeric *nptr;
63     static struct lconv ret;
64
65     loc = uselocale(NULL);
66     lconv = &loc->lconv;
67     if (__mlocale_changed) {
68         /* LC_MONETARY part */
69         struct lc_monetary_T *mptr;
70
71         if (loc->loaded[LC_MONETARY] == 0) {
72             mptr = loc->locdata[LC_MONETARY]->l_data[0];
73             #define M_ASSIGN_STR(NAME) (ret.NAME = (char *)mptr->NAME)
74             #define M_ASSIGN_CHAR(NAME) (ret.NAME = mptr->NAME[0])
75
76             #define M_ASSIGN_STR(NAME) (lconv->NAME = (char *)mptr->NAME)
77             #define M_ASSIGN_CHAR(NAME) (lconv->NAME = mptr->NAME[0])
78
79             mptr = __get_current_monetary_locale();
80             M_ASSIGN_STR(int_curr_symbol);
81             M_ASSIGN_STR(currency_symbol);
82             M_ASSIGN_STR(mon_decimal_point);
83             M_ASSIGN_STR(mon_thousands_sep);
84             M_ASSIGN_STR(mon_grouping);
85             M_ASSIGN_STR(positive_sign);
86             M_ASSIGN_STR(negative_sign);
87             M_ASSIGN_CHAR(int_frac_digits);
88             M_ASSIGN_CHAR(frac_digits);
89             M_ASSIGN_CHAR(p_cs_precedes);
90             M_ASSIGN_CHAR(p_sep_by_space);
91             M_ASSIGN_CHAR(n_cs_precedes);
92             M_ASSIGN_CHAR(n_sep_by_space);
93             M_ASSIGN_CHAR(p_sign_posn);
94             M_ASSIGN_CHAR(n_sign_posn);
95             M_ASSIGN_CHAR(int_p_cs_precedes);
96             M_ASSIGN_CHAR(int_n_cs_precedes);
97             M_ASSIGN_CHAR(int_p_sep_by_space);
98             M_ASSIGN_CHAR(int_n_sep_by_space);
99             M_ASSIGN_CHAR(int_p_sign_posn);
100            M_ASSIGN_CHAR(int_n_sign_posn);
101            loc->loaded[LC_MONETARY] = 1;
102            __mlocale_changed = 0;
103        }
104
105        if (loc->loaded[LC_NUMERIC] == 0) {
106            nptr = loc->locdata[LC_NUMERIC]->l_data[0];
107            if (__nlocale_changed) {
108                /* LC_NUMERIC part */
109                struct lc_numeric_T *nptr;
110
111                #define N_ASSIGN_STR(NAME) (lconv->NAME = (char *)nptr->NAME)
112                #define N_ASSIGN_STR(NAME) (ret.NAME = (char *)nptr->NAME)
113
114                nptr = __get_current_numeric_locale();
115                N_ASSIGN_STR(decimal_point);
116                N_ASSIGN_STR(thousands_sep);
117                N_ASSIGN_STR(grouping);
118                loc->loaded[LC_NUMERIC] = 1;
119                __nlocale_changed = 0;
120            }
121        }
122    }
123    return &ret;
124 }

```

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

3

```
105     }  
107     return (lconv);  
106     return (&ret);  
108 }  
_____unchanged_portion_omitted_____
```

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

1

```
*****
11565 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/localeimpl.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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
59  * be unable to benefit from uselocale(), btw.) The legacy routines are
60  * reimplemented as wrappers that use the appropriate locale object by
```

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

2

```
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 */

74 /*
75  * NB: Each of the structures listed herein should have the refcnt
76  * set to -1, to ensure that posix locale information is never ever freed,
77  * even when changing the global locale.
78 */

80 typedef struct locdata (*loadfn_t)(const char *);

82 static const loadfn_t loaders[LC_ALL] = {
83     __lc_ctype_load,
84     __lc_numeric_load,
85     __lc_time_load,
86     __lc_collate_load,
87     __lc_monetary_load,
88     __lc_messages_load,
89 };

91 extern struct lc_monetary lc_monetary_posix;
92 extern struct lc_numeric lc_numeric_posix;
93 extern struct lc_messages lc_messages_posix;
94 extern struct lc_time lc_time_posix;
95 extern struct lc_ctype lc_ctype_posix;
96 extern struct lc_collate lc_collate_posix;

98 static struct locale posix_locale = {
99     /* locdata */
100     .locdata = {
101         __posix_ctype_locdata,
102         __posix_numeric_locdata,
103         __posix_time_locdata,
104         __posix_collate_locdata,
105         __posix_monetary_locdata,
106         __posix_messages_locdata,
107     },
108     .ctype = &lc_ctype_posix,
109     .numeric = &lc_numeric_posix,
110     .collate = &lc_collate_posix,
111     .monetary = &lc_monetary_posix,
112     .messages = &lc_messages_posix,
113     .time = &lc_time_posix,
114     .runelocale = &DefaultRuneLocale,
115 };

117 locale_t __global_locale = &posix_locale;

119 /*
120  * Category names for getenv() Note that this was modified
121  * for Solaris. See <iso/locale_iso.h>.
122 */
123 #define NUM_CATS 7
124 static char *categories[7] = {
125     "LC_CTYPE",
126     "LC_NUMERIC",
```

```

127     "LC_TIME",
128     "LC_COLLATE",
129     "LC_MONETARY",
130     "LC_MESSAGES",
131     "LC_ALL",
132 };

134 /*
135  * Prototypes.
136  */
137 static const char *get_locale_env(int);
138 static struct locdata *locdata_get(int, const char *);
139 static struct locdata *locdata_get_cache(int, const char *);
140 static void locdata_set_cache(int, struct locdata *);

142 /*
143  * Some utility routines.
144  */
145 struct locdata *
146 __locdata_hold(struct locdata *ld)
147 {
148     if (ld != NULL && ld->l_refcnt != (uint32_t)-1)
149         atomic_inc_32(&ld->l_refcnt);
150     return (ld);
151 }

153 void
154 __locdata_release(struct locdata *ld)
155 {
156     if (ld->l_refcnt == (uint32_t)-1)
157         return;

159     if (atomic_dec_32_nv(&ld->l_refcnt) == 0) {
160         for (int i = 0; i < NLOCDATA; i++)
161             free(ld->l_data[i]);
162         if (ld->l_map && ld->l_map_len) {
163             (void) munmap(ld->l_map, ld->l_map_len);
164         }
165         free(ld);
166     }
167 }

169 struct locdata *
170 __locdata_alloc(const char *name, size_t memsz)
171 {
172     struct locdata *ldata;

174     if ((ldata = calloc(1, sizeof (*ldata))) == NULL) {
175         return (NULL);
176     }
177     if ((ldata->l_data[0] = calloc(1, memsz)) == NULL) {
178         free(ldata);
179         errno = ENOMEM;
180         return (NULL);
181     }
182     (void) strncpy(ldata->l_name, name, sizeof (ldata->l_name));
183     ldata->l_refcnt = 1;

185     return (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 mutex_t cache_lock = DEFAULTMUTEX;

197 /*
198  * Returns the cached data if the locale name is the same. If not,
199  * returns NULL (cache miss). The locdata is returned with a hold on
200  * it, taken on behalf of the caller. The caller should drop the hold
201  * when it is finished.
202  */
203 static struct locdata *
204 locdata_get_cache(int category, const char *locname)
205 {
206     struct locdata *loc;

208     if (category < 0 || category >= LC_ALL)
209         return (NULL);

211     lmutex_lock(&cache_lock);
212     if ((loc = cache_data[category]) != NULL) {
213         if (strcmp(locname, loc->l_name) == 0) {
214             loc = __locdata_hold(loc);
215         } else {
216             loc = NULL;
217         }
218     }
219     lmutex_unlock(&cache_lock);
220     return (loc);
221 }

223 /*
224  * Set the cache for the category to specific content. An additional hold
225  * is taken for the data while it is in the cache, so the caller may drop
226  * its own hold once this is complete. Also, releases the hold on any
227  * previously cached data.
228  */
229 static void
230 locdata_set_cache(int category, struct locdata *loc)
231 {
232     struct locdata *old;

234     if (category < 0 || category >= LC_ALL)
235         return;

237     lmutex_lock(&cache_lock);
238     old = cache_data[category];
239     cache_data[category] = __locdata_hold(loc);
240     lmutex_unlock(&cache_lock);

242     /* drop our reference on the old data */
243     if (old)
244         __locdata_release(old);
245 }

247 /*
248  * Routine to get the locdata for a given category and locale.
249  * This includes retrieving it from cache, retrieving it from
250  * a file, etc.
251  */
252 static struct locdata *
253 locdata_get(int category, const char *locname)
254 {
255     struct locdata *ldata;
256     char scratch[ENCODING_LEN + 1];
257     char *slash;
258     int cnt;

```

```

259     int len;

261     if (locname == NULL || *locname == 0) {
262         locname = get_locale_env(category);
263     }

265     /*
266      * Extract the locale name for the category if it is a composite
267      * locale.
268      */
269     if ((slash = strchr(locname, '/')) != NULL) {
270         for (cnt = category; cnt && slash != NULL; cnt--) {
271             locname = slash + 1;
272             slash = strchr(locname, '/');
273         }
274         if (slash) {
275             len = slash - locname;
276             if (len >= sizeof (scratch)) {
277                 len = sizeof (scratch);
278             }
279         } else {
280             len = sizeof (scratch);
281         }
282         (void) strncpy(scratch, locname, len);
283         locname = scratch;
284     }

286     if ((strcmp(locname, "C") == 0) || (strcmp(locname, "POSIX") == 0))
287         return (__locdata_hold(posix_locale.locdata[category]));

289     ldata = locdata_get_cache(category, locname);
290     if (ldata != NULL)
291         return (ldata);

293     /* Otherwise load it */
294     ldata = (*loaders[category])(locname);
295     if (ldata != NULL) {
296         locdata_set_cache(category, ldata);
297     }
298     return (ldata);
299 }

301 /* tsd destructor */
302 static void
303 freelocptr(void *arg)
304 {
305     locale_t *locptr = arg;
306     if (*locptr != NULL)
307         freelocale(*locptr);
308 }

310 static const char *
311 get_locale_env(int category)
312 {
313     const char *env;

315     /* 1. check LC_ALL. */
316     env = getenv(categories[LC_ALL]);

318     /* 2. check LC_* */
319     if (env == NULL || *env == '\0')
320         env = getenv(categories[category]);

322     /* 3. check LANG */
323     if (env == NULL || *env == '\0')
324         env = getenv("LANG");

```

```

326     /* 4. if none is set, fall to "C" */
327     if (env == NULL || *env == '\0')
328         env = "C";

330     return (env);
331 }

334 /*
335  * This routine is exposed via the MB_CUR_MAX macro. Note that legacy
336  * code will continue to use _ctype[520], but we prefer this function as
337  * it is the only way to get thread-specific information.
338  */
339 unsigned char
340 __mb_cur_max_l(locale_t loc)
341 {
342     return (loc->ctype->lc_max_mblen);
343 }

345 unsigned char
346 __mb_cur_max(void)
347 {
348     return (__mb_cur_max_l(uselocale(NULL)));
349 }

351 /*
352  * Public interfaces.
353  */

355 locale_t
356 duplocale(locale_t src)
357 {
358     locale_t      loc;
359     int            i;

361     loc = calloc(1, sizeof (*loc));
362     if (loc == NULL) {
363         return (NULL);
364     }
365     for (i = 0; i < LC_ALL; i++) {
366         loc->locdata[i] = __locdata_hold(src->locdata[i]);
367         loc->loaded[i] = 0;
368     }
369     loc->collate = loc->locdata[LC_COLLATE]->l_data[0];
370     loc->ctype = loc->locdata[LC_CTYPE]->l_data[0];
371     loc->runelocale = loc->locdata[LC_CTYPE]->l_data[1];
372     loc->messages = loc->locdata[LC_MESSAGES]->l_data[0];
373     loc->monetary = loc->locdata[LC_MONETARY]->l_data[0];
374     loc->numeric = loc->locdata[LC_NUMERIC]->l_data[0];
375     loc->time = loc->locdata[LC_TIME]->l_data[0];
376     return (loc);
377 }

379 void
380 freelocale(locale_t loc)
381 {
382     int i;
383     for (i = 0; i < LC_ALL; i++)
384         __locdata_release(loc->locdata[i]);
385     if (loc != &posix_locale)
386         free(loc);
387 }

389 locale_t
390 newlocale(int catmask, const char *locname, locale_t base)

```

```

391 {
392     locale_t loc;
393     int i, e;

395     if (catmask & ~(LC_ALL_MASK)) {
396         errno = EINVAL;
397         return (NULL);
398     }
399     loc = duplocale(base != NULL ? base : __global_locale);
400     if (loc == NULL) {
401         return (NULL);
402     }

404     for (i = 0; i < LC_ALL; i++) {
405         struct locdata *ldata;
406         loc->loaded[i] = 0;
407         if (((1 << i) & catmask) == 0) {
408             /* Default to base locale if not overriding */
409             continue;
410         }
411         ldata = locdata_get(i, locname);
412         if (ldata == NULL) {
413             e = errno;
414             freelocale(loc);
415             errno = e;
416             return (NULL);
417         }
418         __locdata_release(loc->locdata[i]);
419         loc->locdata[i] = ldata;
420     }
421     if (base && base != __global_locale) {
422         freelocale(base);
423     }
424     loc->collate = loc->locdata[LC_COLLATE]->l_data[0];
425     loc->ctype = loc->locdata[LC_CTYPE]->l_data[0];
426     loc->runelocale = loc->locdata[LC_CTYPE]->l_data[1];
427     loc->messages = loc->locdata[LC_MESSAGES]->l_data[0];
428     loc->monetary = loc->locdata[LC_MONETARY]->l_data[0];
429     loc->numeric = loc->locdata[LC_NUMERIC]->l_data[0];
430     loc->time = loc->locdata[LC_TIME]->l_data[0];
431     return (loc);
432 }

434 locale_t
435 uselocale(locale_t loc)
436 {
437     locale_t lastloc = __global_locale;
438     locale_t *locptr;

440     locptr = tsdalloc(_T_SETLOCALE, sizeof (locale_t), freelocptr);
441     /* Should never occur */
442     if (locptr == NULL) {
443         errno = EINVAL;
444         return (NULL);
445     }

447     if (*locptr != NULL)
448         lastloc = *locptr;

450     /* Argument loc is NULL if we are just querying. */
451     if (loc != NULL) {
452         /*
453          * Set it to LC_GLOBAL_LOCAL to return to using
454          * the global locale (setlocale).
455          */
456         if (loc == __global_locale) {

```

```

457         *locptr = NULL;
458     } else {
459         /* No validation of the provided locale at present */
460         *locptr = loc;
461     }
462 }

464 /*
465  * The caller is responsible for freeing, of course it would be
466  * gross error to call freelocale() on a locale object that is still
467  * in use.
468  */
469     return (lastloc);
470 }

```

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

1

```
*****
3324 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/localeimpl.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 /*
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     uint32_t l_refcnt;
54 };

57 struct locale {
58     struct locdata *locdata[LC_ALL];
59
60     /*
```

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

2

```
61     * Convenience pointers.
62     */
63     const struct lc_ctype *ctype;
64     const struct lc_collate *collate;
65     const struct lc_messages *messages;
66     const struct lc_monetary *monetary;
67     const struct lc_numeric *numeric;
68     const struct lc_time *time;
69     const _RuneLocale *runelocale;

71     /*
72     * The loaded value is used for localeconv. In particular, when
73     * when we change the value of one of the above categories, we will
74     * also need to update the lconv structure. The loaded bit indicates
75     * that the lconv structure is "current" for that category. It's
76     * sort of an "inverse dirty" bit.
77     */
78     int loaded[LC_ALL];
79     struct lconv lconv;
80 };

83 struct locdata * __locdata_alloc(const char *, size_t);
84 struct locdata * __locdata_hold(struct locdata *);
85 void __locdata_release(struct locdata *);
86 struct locdata * __locdata_get_cache(int, const char *);
87 void __locdata_set_cache(int, struct locdata *);

89 struct locdata * __lc_numeric_load(const char *name);
90 struct locdata * __lc_monetary_load(const char *name);
91 struct locdata * __lc_messages_load(const char *name);
92 struct locdata * __lc_time_load(const char *name);
93 struct locdata * __lc_ctype_load(const char *name);
94 struct locdata * __lc_collate_load(const char *name);

96 extern struct locdata __posix_numeric_locdata;
97 extern struct locdata __posix_monetary_locdata;
98 extern struct locdata __posix_messages_locdata;
99 extern struct locdata __posix_time_locdata;
100 extern struct locdata __posix_ctype_locdata;
101 extern struct locdata __posix_collate_locdata;

103 #endif /* _LOCALEIMPL_H_ */
```

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

1

2066 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/mblen.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 <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     static const mbstate_t initial = { 0 };
42     static mbstate_t mbs;
43     size_t rval;

44     if (s == NULL) {
45         /* No support for state dependent encodings. */
46         mbs = initial;
47         return (0);
48     }
49     rval = mbrtowc_l(NULL, s, n, &mbs, loc);
45     rval = __mbrtowc(NULL, s, n, &mbs);
50     if (rval == (size_t)-1 || rval == (size_t)-2)
51         return (-1);
52     return ((int)rval);
53 }

55 int
56 mblen(const char *s, size_t n)
57 {
58     return (mblen_l(s, n, uselocale(NULL)));
```

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

2

```
59 }
    _____
    unchanged_portion_omitted_
```

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

1

```
*****
2592 Tue May 20 07:58:48 2014
new/usr/src/lib/libc/port/locale/mblocal.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 _MSKanjii_init(struct lc_ctype *);
49 int _none_init(_RuneLocale *);
50 int _UTF8_init(_RuneLocale *);
51 int _EUC_CN_init(_RuneLocale *);
52 int _EUC_JP_init(_RuneLocale *);
53 int _EUC_KR_init(_RuneLocale *);
54 int _EUC_TW_init(_RuneLocale *);
55 int _GB18030_init(_RuneLocale *);
56 int _GB2312_init(_RuneLocale *);
57 int _GBK_init(_RuneLocale *);
58 int _BIG5_init(_RuneLocale *);
59 int _MSKanjii_init(_RuneLocale *);
```

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

2

```
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,
52 const char *_RESTRICT_KYWD, size_t, mbstate_t *_RESTRICT_KYWD);
53 typedef size_t (*wrtomb_pfn_t)(char *_RESTRICT_KYWD, wchar_t,
54 extern int (*__mbsinit)(const mbstate_t *);
55 extern size_t (*__mbsnrtowcs)(wchar_t *_RESTRICT_KYWD,
56 const char **_RESTRICT_KYWD, size_t, size_t, mbstate_t *_RESTRICT_KYWD);

57 extern size_t (*__wrtomb)(char *_RESTRICT_KYWD, wchar_t,
58 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, wrtomb_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 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mbrlen.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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_1(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_1(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_1(s, n, ps, uselocale(NULL)));
39     return (__mbrtowc(NULL, s, n, ps));
50 }
__unchanged_portion_omitted__
```

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

1

1999 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mbrtowc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 {
39     mbrtowc(wchar_t * _RESTRICT_KYWD pwc, const char * _RESTRICT_KYWD s,
40           size_t n, mbstate_t * _RESTRICT_KYWD ps)
41 {
42     static mbstate_t mbs;
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 _unchanged_portion_omitted_
```

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

1

727 Tue May 20 07:58:49 2014

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

2964 need POSIX 2008 locale object support

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

```
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 /* Copyright 2013 Garrett D'Amore <garrett@damore.org>
36 */
37
38 #include "lint.h"
39 #include <locale.h>
40 #include "localeimpl.h"
41 #include "lctype.h"
42 #include <wchar.h>
43 #include "mblocal.h"
44
45 int
46 mbsinit_l(const mbstate_t *s, locale_t loc)
47 mbsinit(const mbstate_t *ps)
48 {
49     return (loc->ctype->lc_mbsinit(s));
50 }
51
52 int
53 mbsinit(const mbstate_t *s)
54 {
55     return (mbsinit_l(s, uselocale(NULL)));
56     return (__mbsinit(ps));
57 }
58
59 _unchanged_portion_omitted_
```

2

```

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;

66     s = *src;
67     nchr = 0;

69     if (dst == NULL) {
70         for (;;) {
71             if ((nb = pmbrtowc(&wc, s, nms, ps)) == (size_t)-1)
60             if ((nb = __mbrtowc(&wc, s, nms, ps)) == (size_t)-1)
72                 /* Invalid sequence - mbrtowc() sets errno. */
73                 return ((size_t)-1);
74             else if (nb == 0 || nb == (size_t)-2)
75                 return (nchr);
76             s += nb;
77             nms -= nb;
78             nchr++;
79         }
80         /*NOTREACHED*/
81     }

83     while (len-- > 0) {
84         if ((nb = pmbrtowc(dst, s, nms, ps)) == (size_t)-1) {
73         if ((nb = __mbrtowc(dst, s, nms, ps)) == (size_t)-1) {
85             *src = s;
86             return ((size_t)-1);
87         } else if (nb == (size_t)-2) {
88             *src = s + nms;
89             return (nchr);
90         } else if (nb == 0) {
91             *src = NULL;
92             return (nchr);
93         }
94         s += nb;
95         nms -= nb;
96         nchr++;
97         dst++;
98     }
99     *src = s;
100     return (nchr);
101 }

```

unchanged portion omitted

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

1

2043 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mbsrtowcs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 <locale.h>
31 #include <errno.h>
32 #include <limits.h>
33 #include <stdlib.h>
34 #include <wchar.h>
35 #include "mblocal.h"
36 #include "localeimpl.h"
37 #include "lctype.h"

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)));
50     static mbstate_t mbs;
51
52     if (ps == NULL)
53         ps = &mbs;
54     return (__mbsnrtowcs(dst, src, ULONG_MAX, len, ps));
55 }
```

_____unchanged_portion_omitted_____

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

1

```
*****
2062 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mbstowcs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 }
56 return (__mbsnrtowcs(pwcs, &sp, ULONG_MAX, n, &mbs));
57 }

__unchanged_portion_omitted_
```

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

1

2141 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mbtowc.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 <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     static const mbstate_t initial = { 0 };
42     static mbstate_t mbs;
43     size_t rval;

44     if (s == NULL) {
45         /* No support for state dependent encodings. */
46         mbs = initial;
47         return (0);
48     }
49     rval = mbrtowc_l(pwc, s, n, &mbs, loc);
45     rval = __mbrtowc(pwc, s, n, &mbs);
50     if (rval == (size_t)-1 || rval == (size_t)-2)
51         return (-1);
52     return ((int)rval);
53 }

55 int
56 mbtowc(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s, size_t n)
57 {
58     return (mbtowc_l(pwc, s, n, uselocale(NULL)));
```

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

2

```
59 }
    _____
    unchanged_portion_omitted_
```

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

1

```
*****
5036 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/mskanji.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 */

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"

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_wcrtomb(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);
```

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

2

```
60 typedef struct {
61     wchar_t ch;
62 } _MSKanjiState;

64 void
65 _MSKanji_init(struct lc_ctype *lct)
66 {
67     int
68     _MSKanji_init(_RuneLocale *rl)
69 {
70     lct->lc_mbrtowc = _MSKanji_mbrtowc;
71     lct->lc_wcrtomb = _MSKanji_wcrtomb;
72     lct->lc_mbsnrtowcs = _MSKanji_mbsnrtowcs;
73     lct->lc_wcsnrtombs = _MSKanji_wcsnrtombs;
74     lct->lc_mbsinit = _MSKanji_mbsinit;
75     lct->lc_max_mblen = 2;
76     lct->lc_is_ascii = 0;
77     _mbrtowc = _MSKanji_mbrtowc;
78     _wcrtomb = _MSKanji_wcrtomb;
79     _mbsinit = _MSKanji_mbsinit;
80     _CurrentRuneLocale = rl;
81     _ctype[520] = 2;
82     charset_is_ascii = 0;
83     return (0);
84 }

unchanged_portion_omitted

142 static size_t
143 _MSKanji_wcrtomb(char * _RESTRICT_KYWD s, wchar_t wc,
144 mbstate_t * _RESTRICT_KYWD ps)
145 {
146     _MSKanjiState *ms;
147     int len, i;

149     ms = (_MSKanjiState *)ps;

151     if (ms->ch != 0) {
152         errno = EINVAL;
153         return ((size_t)-1);
154     }

156     if (s == NULL)
157         /* Reset to initial shift state (no-op) */
158         return (1);
159     len = (wc > 0x100) ? 2 : 1;
160     for (i = len; i-- > 0; )
161         *s++ = wc >> (i << 3);
162     return (len);
163 }

165 static size_t
166 _MSKanji_mbsnrtowcs(wchar_t * _RESTRICT_KYWD dst,
167 const char ** _RESTRICT_KYWD src, size_t nms,
168 size_t len, mbstate_t * _RESTRICT_KYWD ps)
169 {
170     return (_mbsnrtowcs_std(dst, src, nms, len, ps, _MSKanji_mbrtowc));
171 }

173 static size_t
174 _MSKanji_wcsnrtombs(char * _RESTRICT_KYWD dst,
175 const wchar_t ** _RESTRICT_KYWD src, size_t nwc,
176 size_t len, mbstate_t * _RESTRICT_KYWD ps)
177 {
178     return (_wcsnrtombs_std(dst, src, nwc, len, ps, _MSKanji_wcrtomb));
179 }

unchanged_portion_omitted
```

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

1

```
*****
3268 Tue May 20 07:58:49 2014
new/usr/src/lib/libc/port/locale/nextwctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 = &CurrentRuneLocale->__runetype_ext;
49     _RuneEntry *base, *re;
50     int noinc;

51     rl = loc->runelocale;
52     rr = &rl->__runetype_ext;

53     noinc = 0;
54     if (wc < _CACHED_RUNES) {
55         wc++;
56         while (wc < _CACHED_RUNES) {
```

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

2

```
57         if (rl->__runetype[wc] & wct)
58             if (_CurrentRuneLocale->__runetype[wc] & wct)
59                 return (wc);
60         wc++;
61     }
62     wc--;
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 }

    unchanged_portion_omitted
```

```

*****
5181 Tue May 20 07:58:49 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>
*****

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 nl_langinfo(nl_item item)
49 {
50     char *ret, *s, *cs;
51     struct locdata *ldata;
52     const struct lc_monetary *lmon = loc->monetary;
53     const struct lc_numeric *lnum = loc->numeric;
54     const struct lc_messages *lmsgs = loc->messages;
55     const struct lc_time *ltime = loc->time;

```

```

53     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_1name : 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)];
118            ret = (char *)TIME->month[_REL(MON_1)];
119            break;

```

```

109     case ABMON_1: case ABMON_2: case ABMON_3: case ABMON_4:
110     case ABMON_5: case ABMON_6: case ABMON_7: case ABMON_8:
111     case ABMON_9: case ABMON_10: case ABMON_11: case ABMON_12:
112         ret = (char *)ltime->mon[_REL(ABMON_1)];
106         ret = (char *)TIME->mon[_REL(ABMON_1)];
113         break;
114     case ERA:
115         /* XXX: need to be implemented */
116         ret = "";
117         break;
118     case ERA_D_FMT:
119         /* XXX: need to be implemented */
120         ret = "";
121         break;
122     case ERA_D_T_FMT:
123         /* XXX: need to be implemented */
124         ret = "";
125         break;
126     case ERA_T_FMT:
127         /* XXX: need to be implemented */
128         ret = "";
129         break;
130     case ALT_DIGITS:
131         /* XXX: need to be implemented */
132         ret = "";
133         break;
134     case RADIXCHAR:
135         ret = (char *)lnum->decimal_point;
129         ret = (char *)NUMERIC->decimal_point;
136         break;
137     case THOUSEP:
138         ret = (char *)lnum->thousands_sep;
132         ret = (char *)NUMERIC->thousands_sep;
139         break;
140     case YESEXPR:
141         ret = (char *)lmsgs->yesexpr;
135         ret = (char *)MESSAGES->yesexpr;
142         break;
143     case NOEXPR:
144         ret = (char *)lmsgs->noexpr;
138         ret = (char *)MESSAGES->noexpr;
145         break;
146     /*
147     * YESSTR and NOSTR items were removed from Issue 7. But
148     * older applications might still need them. Their use is
149     * discouraged.
150     * YESSTR and NOSTR items marked with LEGACY are available, but not
151     * recommended by SUSv2 to be used in portable applications since
152     * they're subject to remove in future specification editions.
153     */
151     case YESSTR: /* LEGACY */
152         ret = (char *)lmsgs->yesstr;
146         ret = (char *)MESSAGES->yesstr;
153         break;
154     case NOSTR: /* LEGACY */
155         ret = (char *)lmsgs->nostr;
149         ret = (char *)MESSAGES->nostr;
156         break;
157     /*
158     * SUSv2 special formatted currency string
159     */
160     case CRNCYSTR:
161         ret = lmon->crncystr;
162         break;
155         ret = "";
156         cs = (char *)MONETARY->currency_symbol;

```

```

157     if (*cs != '\0') {
158         char pos = localeconv()->p_cs_precedes;

160         if (pos == localeconv()->n_cs_precedes) {
161             char psn = '\0';

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;

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;
164     case _DATE_FMT: /* Solaris specific extension */
165         ret = (char *)ltime->date_fmt;
186         ret = (char *)TIME->date_fmt;
166         break;
167     /*
168     * Note that FreeBSD also had a private D_MD_ORDER, but that appears
169     * to have been specific to FreeBSD, so we have not included it here.
170     */
171     default:
172         ret = "";
173     }
174     return (ret);
175 }

177 char *
178 nl_langinfo(nl_item item)
179 {
180     return (nl_langinfo_l(item, uselocale(NULL)));
181 }

```

unchanged_portion_omitted

new/usr/src/lib/libc/port/locale/none.c

1

```
*****
4427 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/none.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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,
57 const wchar_t *_RESTRICT_KYWD,
58 size_t, size_t, mbstate_t *_RESTRICT_KYWD);
```

new/usr/src/lib/libc/port/locale/none.c

2

```
48 /* setup defaults */

50 void
51 _none_init(struct lc_ctype *lct)
52 int
53 _none_init(_RuneLocale *rl)
54 {
55     lct->lc_is_ascii = 1;
56     lct->lc_mbrtowc = __mbrtowc_ascii;
57     lct->lc_mbsinit = __mbsinit_ascii;
58     lct->lc_mbsnrtowcs = __mbsnrtowcs_ascii;
59     lct->lc_wcrtomb = __wcrtomb_ascii;
60     lct->lc_wcsnrtombs = __wcsnrtombs_ascii;
61     lct->lc_max_mblen = 1;
62     charset_is_ascii = 1;

63     __mbrtowc = _none_mbrtowc;
64     __mbsinit = _none_mbsinit;
65     __mbsnrtowcs = _none_mbsnrtowcs;
66     __wcrtomb = _none_wcrtomb;
67     __wcsnrtombs = _none_wcsnrtombs;
68     _CurrentRuneLocale = rl;
69     return (0);
70 }

72 int
73 __mbsinit_ascii(const mbstate_t *_unused)
74 static int
75 _none_mbsinit(const mbstate_t *_unused)
76 {
77     _NOTE(ARGUNUSED(unused));

78     /*
79      * Encoding is not state dependent - we are always in the
80      * initial state.
81      */
82     return (1);
83 }

85 size_t
86 __mbrtowc_ascii(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s,
87 static size_t
88 _none_mbrtowc(wchar_t *_RESTRICT_KYWD pwc, const char *_RESTRICT_KYWD s,
89 size_t n, mbstate_t *_RESTRICT_KYWD unused)
90 {
91     _NOTE(ARGUNUSED(unused));

92     if (s == NULL)
93         /* Reset to initial shift state (no-op) */
94         return (0);
95     if (n == 0)
96         /* Incomplete multibyte sequence */
97         return ((size_t)-2);
98     if (pwc != NULL)
99         *pwc = (unsigned char)*s;
100     return (*s == '\\0' ? 0 : 1);
101 }

103 size_t
104 __wcrtomb_ascii(char *_RESTRICT_KYWD s, wchar_t wc,
105 static size_t
106 _none_wcrtomb(char *_RESTRICT_KYWD s, wchar_t wc,
107 mbstate_t *_RESTRICT_KYWD unused)
108 {
109     _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 (__wcrctomb)(char *_RESTRICT_KYWD, wchar_t, mbstate_t *_RESTRICT_KYWD) =
196    __none_wcrctomb;

198 size_t (__wcsnrtombs)(char *_RESTRICT_KYWD, const wchar_t **_RESTRICT_KYWD,
199    size_t, size_t, mbstate_t *_RESTRICT_KYWD) = __none_wcsnrtombs;

```

```

*****
41502 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/regcomp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 */
36
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>
48
49 #include "runetype.h"
50 #include "collate.h"
51
52 #include "utils.h"
53 #include "regex2.h"
54
55 #include "cname.h"
56 #include "mblocal.h"
57
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 #ifndef unchanged_portion_omitted
77
78 /*
79  * p_b_term - parse one term of a bracketed character list
80  */
81 static void
82 p_b_term(struct parse *p, cset *cs)
83 {
84     char c;
85     wint_t start, finish;
86     wint_t i;
87     locale_t loc = uselocale(NULL);
88
89     /* classify what we've got */
90     switch ((MORE()) ? PEEK() : '\0') {
91     case '[':
92         c = (MORE2()) ? PEEK2() : '\0';
93         break;
94     case '-':
95         SETERROR(REG_ERANGE);
96         return; /* NOTE RETURN */
97     default:
98         c = '\0';
99         break;
100    }
101
102    switch (c) {
103    case '.': /* character class */
104        NEXT2();
105        (void) REQUIRE(MORE(), REG_EBRACK);
106        c = PEEK();
107        (void) REQUIRE(c != '-' && c != ']', REG_ECTYPE);
108        p_b_cclass(p, cs);
109        (void) REQUIRE(MORE(), REG_EBRACK);
110        (void) REQUIRE(EATTWO(':', ']', REG_ECTYPE);
111        break;
112    case '=': /* equivalence class */
113        NEXT2();
114        (void) REQUIRE(MORE(), REG_EBRACK);
115        c = PEEK();
116        (void) REQUIRE(c != '-' && c != ']', REG_ECOLLATE);
117        p_b_ecclass(p, cs);
118        (void) REQUIRE(MORE(), REG_EBRACK);
119        (void) REQUIRE(EATTWO('=', ']', REG_ECOLLATE);
120        break;
121    default: /* symbol, ordinary character, or range */
122        start = p_b_symbol(p);
123        if (SEE('-', ']' && MORE2() && PEEK2() != ']') {
124            /* range */
125            NEXT();
126            if (EAT('-'))
127                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                 finish <= 0, REG_ERANGE);
786                 for (i = 0; i <= UCHAR_MAX; i++) {
787                     if (_collate_range_cmp(start, i, loc)
788                         <= 0 &&
789                         _collate_range_cmp(i, finish, loc)
790                         <= 0)
791                         if (_collate_range_cmp(start, i) <= 0 &&
792                             _collate_range_cmp(i, finish) <= 0)
793                             CHadd(p, cs, i);
794             }
795         }
796     }
797 }

```

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);

1376     /* avoid making error situations worse */
1377     if (p->error != 0)
1378         return;

1380     /*
1381      * It's not generally safe to do a "char" substring search on
1382      * multibyte character strings, but it's safe for at least
1383      * UTF-8 (see RFC 3629).
1384      */
1385     if (MB_CUR_MAX > 1 &&

```

```

1386     strcmp(loc->runeLocale->__encoding, "UTF-8") != 0)
1387     strcmp(_CurrentRuneLocale->__encoding, "UTF-8") != 0)
1388         return;

1389     /* find the longest OCHAR sequence in strip */
1390     newlen = 0;
1391     offset = 0;
1392     g->moffset = 0;
1393     scan = g->strip + 1;
1394     do {
1395         s = *scan++;
1396         switch (OP(s)) {
1397             case OCHAR: /* sequence member */
1398                 if (newlen == 0) { /* new sequence */
1399                     (void) memset(&mbs, 0, sizeof (mbs));
1400                     newstart = scan - 1;
1401                 }
1402                 clen = wctomb(buf, OPND(s), &mbs);
1403                 if (clen == (size_t)-1)
1404                     goto toohard;
1405                 newlen += clen;
1406                 break;
1407             /* things that don't break one */
1408             case OLPAREN:
1409             case ORPAREN:
1410                 break;
1411             /* things that must be skipped */
1412             case OQUEST_:
1413             case OCH_:
1414                 offset = altoffset(scan, offset);
1415                 scan--;
1416                 do {
1417                     scan += OPND(s);
1418                     s = *scan;
1419                     /* assert() interferes w debug printouts */
1420                     if (OP(s) != O_QUEST && OP(s) != O_CH &&
1421                         OP(s) != OOR2) {
1422                         g->iflags |= BAD;
1423                         return;
1424                     }
1425                 } while (OP(s) != O_QUEST && OP(s) != O_CH);
1426             /* FALLTHROUGH */
1427             /* things that break a sequence */
1428             case OBOW:
1429             case OEOW:
1430             case OBOL:
1431             case OEOL:
1432             case O_QUEST:
1433             case O_CH:
1434             case OEND:
1435                 if (newlen > g->mlen) { /* ends one */
1436                     start = newstart;
1437                     g->mlen = newlen;
1438                     if (offset > -1) {
1439                         g->moffset += offset;
1440                         offset = newlen;
1441                     } else
1442                         g->moffset = offset;
1443                 } else {
1444                     if (offset > -1)
1445                         offset += newlen;
1446                 }
1447                 newlen = 0;
1448                 break;
1449             case OANY:
1450                 if (newlen > g->mlen) { /* ends one */
1451                     start = newstart;
1452                     g->mlen = 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->mmlen) { /* ends one */
1467         start = newstart;
1468         g->mmlen = 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->mmlen) { /* ends one */
1491         start = newstart;
1492         g->mmlen = 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->mmlen == 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->mmlen + 1);
1511 if (g->must == NULL) { /* argh; just forget it */
1512     g->mmlen = 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->mmlen) {
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->mmlen);
1527     *cp++ = '\0'; /* just on general principles */
1528 }

```

unchanged_portion_omitted

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

1

```
*****
6830 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/rune.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 "runetype.h"
48 #include "runefile.h"
46 _RuneLocale *_Read_RuneMagi(FILE *);
50 _RuneLocale *
51 _Read_RuneMagi(const char *fname)
52 _Read_RuneMagi(FILE *fp)
53 {
54     char *fdata, *data;
55     void *lastp;
56     _FileRuneLocale *frl;
57     _RuneLocale *rl;
58     _FileRuneEntry *frr;
```

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

2

```
58     _RuneEntry *rr;
59     struct stat sb;
60     int x, saverr;
61     void *variable;
62     _FileRuneEntry *runetype_ext_ranges;
63     _FileRuneEntry *maplower_ext_ranges;
64     _FileRuneEntry *mapupper_ext_ranges;
65     int runetype_ext_len = 0;
66     int fd;
68     if ((fd = open(fname, O_RDONLY)) < 0) {
69         errno = EINVAL;
70         if (fstat(fileno(fp), &sb) < 0)
71             return (NULL);
73     if (fstat(fd, &sb) < 0) {
74         (void) close(fd);
75         if ((size_t)sb.st_size < sizeof (_FileRuneLocale)) {
76             errno = EINVAL;
77             return (NULL);
79     if ((size_t)sb.st_size < sizeof (_FileRuneLocale)) {
80         (void) close(fd);
81         errno = EINVAL;
82         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);
83     }
86     fdata = mmap(NULL, sb.st_size, PROT_READ, MAP_PRIVATE, fd, 0);
87     (void) close(fd);
88     if (fdata == NULL) {
89         errno = EINVAL;
90         if (fread(fdata, sb.st_size, 1, fp) != 1) {
91             saverr = errno;
92             free(fdata);
93             errno = saverr;
94             return (NULL);
96     frl = (_FileRuneLocale *) (void *) fdata;
97     lastp = fdata + sb.st_size;
98     variable = frl + 1;
99     if (memcmp(frl->magic, _FILE_RUNE_MAGIC_1, sizeof (frl->magic))) {
100         goto invalid;
101         free(fdata);
102         errno = EINVAL;
103         return (NULL);
104     }
102     runetype_ext_ranges = (_FileRuneEntry *) variable;
103     variable = runetype_ext_ranges + frl->runetype_ext_nranges;
104     if (variable > lastp) {
105         goto invalid;
```

```

106         free(fdata);
107         errno = EINVAL;
108         return (NULL);
109     }
110
111     maplower_ext_ranges = (_FileRuneEntry *)variable;
112     variable = maplower_ext_ranges + frl->maplower_ext_nranges;
113     if (variable > lastp) {
114         goto invalid;
115         free(fdata);
116         errno = EINVAL;
117         return (NULL);
118     }
119
120     mapupper_ext_ranges = (_FileRuneEntry *)variable;
121     variable = mapupper_ext_ranges + frl->mapupper_ext_nranges;
122     if (variable > lastp) {
123         goto invalid;
124         free(fdata);
125         errno = EINVAL;
126         return (NULL);
127     }
128
129     frr = runetype_ext_ranges;
130     for (x = 0; x < frl->runetype_ext_nranges; ++x) {
131         uint32_t *types;
132
133         if (frr[x].map == 0) {
134             int len = frr[x].max - frr[x].min + 1;
135             types = variable;
136             variable = types + len;
137             runetype_ext_len += len;
138             if (variable > lastp) {
139                 goto invalid;
140                 free(fdata);
141                 errno = EINVAL;
142                 return (NULL);
143             }
144         }
145     }
146
147     if ((char *)variable + frl->variable_len > (char *)lastp) {
148         goto invalid;
149         free(fdata);
150         errno = EINVAL;
151         return (NULL);
152     }
153
154     /*
155     * Convert from disk format to host format.
156     */
157     data = malloc(sizeof (_RuneLocale) +
158         (frl->runetype_ext_nranges + frl->maplower_ext_nranges +
159         frl->mapupper_ext_nranges) * sizeof (_RuneEntry) +
160         runetype_ext_len * sizeof (*rr->__types) +
161         frl->variable_len);
162     if (data == NULL) {
163         saverr = errno;
164         (void) munmap(fdata, sb.st_size);
165         free(fdata);
166         errno = saverr;
167         return (NULL);
168     }
169
170     rl = (_RuneLocale *) (void *)data;
171     rl->__variable = rl + 1;

```

```

157     (void) memcpy(rl->__magic, _RUNE_MAGIC_1, sizeof (rl->__magic));
158     (void) memcpy(rl->__encoding, frl->encoding, sizeof (rl->__encoding));
159
160     rl->__variable_len = frl->variable_len;
161     rl->__runetype_ext.__nranges = frl->runetype_ext_nranges;
162     rl->__maplower_ext.__nranges = frl->maplower_ext_nranges;
163     rl->__mapupper_ext.__nranges = frl->mapupper_ext_nranges;
164
165     for (x = 0; x < _CACHED_RUNES; ++x) {
166         rl->__runetype[x] = frl->runetype[x];
167         rl->__maplower[x] = frl->maplower[x];
168         rl->__mapupper[x] = frl->mapupper[x];
169     }
170
171     rl->__runetype_ext.__ranges = (_RuneEntry *)rl->__variable;
172     rl->__variable = rl->__runetype_ext.__ranges +
173         rl->__runetype_ext.__nranges;
174
175     rl->__maplower_ext.__ranges = (_RuneEntry *)rl->__variable;
176     rl->__variable = rl->__maplower_ext.__ranges +
177         rl->__maplower_ext.__nranges;
178
179     rl->__mapupper_ext.__ranges = (_RuneEntry *)rl->__variable;
180     rl->__variable = rl->__mapupper_ext.__ranges +
181         rl->__mapupper_ext.__nranges;
182
183     variable = mapupper_ext_ranges + frl->mapupper_ext_nranges;
184     frr = runetype_ext_ranges;
185     rr = rl->__runetype_ext.__ranges;
186     for (x = 0; x < rl->__runetype_ext.__nranges; ++x) {
187         uint32_t *types;
188
189         rr[x].__min = frr[x].min;
190         rr[x].__max = frr[x].max;
191         rr[x].__map = frr[x].map;
192         if (rr[x].__map == 0) {
193             int len = rr[x].__max - rr[x].__min + 1;
194             types = variable;
195             variable = types + len;
196             rr[x].__types = rl->__variable;
197             rl->__variable = rr[x].__types + len;
198             while (len-- > 0)
199                 rr[x].__types[len] = types[len];
200         } else
201             rr[x].__types = NULL;
202     }
203
204     frr = maplower_ext_ranges;
205     rr = rl->__maplower_ext.__ranges;
206     for (x = 0; x < rl->__maplower_ext.__nranges; ++x) {
207         rr[x].__min = frr[x].min;
208         rr[x].__max = frr[x].max;
209         rr[x].__map = frr[x].map;
210     }
211
212     frr = mapupper_ext_ranges;
213     rr = rl->__mapupper_ext.__ranges;
214     for (x = 0; x < rl->__mapupper_ext.__nranges; ++x) {
215         rr[x].__min = frr[x].min;
216         rr[x].__max = frr[x].max;
217         rr[x].__map = frr[x].map;
218     }
219
220     (void) memcpy(rl->__variable, variable, rl->__variable_len);
221     (void) munmap(fdata, sb.st_size);

```

```
232     free(fdata);
223
224     /*
225     * Go out and zero pointers that should be zero.
226     */
227     if (!r1->__variable_len)
228         r1->__variable = NULL;
229
230     if (!r1->__runetype_ext.__nranges)
231         r1->__runetype_ext.__ranges = NULL;
232
233     if (!r1->__maplower_ext.__nranges)
234         r1->__maplower_ext.__ranges = NULL;
235
236     if (!r1->__mapupper_ext.__nranges)
237         r1->__mapupper_ext.__ranges = NULL;
238
239     return (r1);
240 invalid:
241     (void) munmap(fdata, sb.st_size);
242     errno = EINVAL;
243     return (NULL);
244 }
```

unchanged_portion_omitted

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

1

```
*****
2456 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/runetype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 __runetype(__ct_rune_t c)
43 {
44     size_t lim;
45     const _RuneRange *rr = &rl->__runetype_ext;
46     const _RuneEntry *base, *re;
47     _RuneRange *rr = &CurrentRuneLocale->__runetype_ext;
48     _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)
56                 return (re->__types[c - re->__min]);
57             else
```

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

2

```
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_____
```

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

1

```
*****
5940 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/runetype.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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.
```

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

2

```
61 */

63 #ifndef _RUNETYPE_H_
64 #define _RUNETYPE_H_

66 #define _CACHED_RUNES (1 << 8) /* Must be a power of 2 */
67 #define _CRMASK (~(_CACHED_RUNES - 1))

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  */

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

95 typedef int __rune_t; /* rune_t (see above) */

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

112 typedef struct {
113     char __magic[8]; /* Magic saying what version we are */
114     char __encoding[32]; /* ASCII name of encoding */
115     char __encoding[32]; /* ASCII name of this encoding */

116     unsigned int __runetype[_CACHED_RUNES];
117     __rune_t __maplower[_CACHED_RUNES];
118     __rune_t __mapupper[_CACHED_RUNES];

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 */
```

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

3

```
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_ */
```

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

1

```
*****
5354 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/setlocale.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 "collate.h"
47 #include "lnumeric.h" /* for struct lc_numeric */
48 #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() */
49 #include "setlocale.h"
51 #include "ldpart.h"
52 #include "timelocal.h" /* for __time_load_locale() */
50 #include "../i18n/_loc_path.h"
51 #include "localeimpl.h"
52 #include "../i18n/_locale.h"
```

54 /*

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

2

```
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 /*
55 * Path to locale storage directory. See ../i18n/_loc_path.h
56 */
57 char *_PathLocale = _DFLT_LOC_PATH;

59 static char *current_locale(locale_t, int);
60 static void install_legacy(locale_t, int);
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 static char *currentlocale(void);
96 static char *loadlocale(int);
97 static const char *__get_locale_env(int);

62 char *
63 setlocale(int category, const char *locname)
100 setlocale(int category, const char *locale)
64 {
65     locale_t oldloc;
66     locale_t newloc;
67     int mask;
102     int i, j, saverr;
103     const char *env, *r;

69     if (category < 0 || category > LC_ALL) {
105     if (category < 0 || category >= NUM_CATS) {
70         errno = EINVAL;
71         return (NULL);
72     }

74     if (locname == NULL)
75         return (current_locale(__global_locale, category));
110     if (locale == NULL)
111         return (category != LC_ALL ?
112             current_categories[category] : currentlocale());
```

```

77     if ((oldloc = duplocale(__global_locale)) == NULL)
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]);

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;

80     mask = (category == LC_ALL ? LC_ALL_MASK : (1 << category));
161     buf = alloca(strlen(locale) + 1);
162     (void) strcpy(buf, locale);

82     newloc = newlocale(mask, locname, oldloc);
83     if (newloc == NULL) {
84         freelocale(oldloc);
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) {
170                 /*
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.
176     */
177     errno = EINVAL;
185     return (NULL);
186 }
180     (void) strcpy(new_categories[i], r,
181                 ENCODING_LEN);
182     r = strtok_r(NULL, "/", &save);
183 }
184 if (r != NULL) {
185     /*
186     * Too many components - we had left over
187     * data in the LC_ALL. It is malformed.
188     */
189     errno = EINVAL;
190     return (NULL);
191 }
192 }
193 }

88     oldloc = __global_locale;
89     __global_locale = newloc;
195     if (category != LC_ALL)
196         return (loadlocale(category));

91     install_legacy(newloc, mask);

93     freelocale(oldloc);
94     return (current_locale(newloc, category));
198     for (i = 0; i < NUM_CATS; ++i) {
199         (void) strcpy(saved_categories[i], current_categories[i]);
200         if (i == LC_ALL)
201             continue;
202         if (loadlocale(i) == NULL) {
203             saverr = errno;
204             for (j = 0; j < i; ++j) {
205                 (void) strcpy(new_categories[j],
206                             saved_categories[j]);
207                 if (i == LC_ALL)
208                     continue;
209                 if (loadlocale(j) == NULL) {
210                     (void) strcpy(new_categories[j], "C");
211                     (void) loadlocale(j);
212                 }
213             }
214             errno = saverr;
215             return (NULL);
216         }
217     }
218     return (currentlocale());
95 }

97 static char *
98 current_locale(locale_t loc, int cat)
222 currentlocale(void)
99 {
224     int i;
100     int composite = 0;
101     static char buf[LC_ALL * (ENCODING_LEN + 1 + 1)];

103     if (cat != LC_ALL) {
104         return (loc->locdata[cat]->l_1name);
105     }

107     /* Look to see if any category is different */

```

```

108     for (int i = 1; i < LC_ALL; ++i) {
109         if (strcmp(loc->locdata[0]->l_name,
110             loc->locdata[i]->l_name) != 0) {
228     for (i = 1; i < NUM_CATS; ++i) {
229         if (i == LC_ALL)
230             continue;
231         if (strcmp(current_categories[0], current_categories[i])) {
111             composite = 1;
112             break;
113         }
114     }
116     if (composite) {
117         /*
118          * Note ordering of these follows the numeric order,
119          * if the order is changed, then setlocale() will need
120          * to be changed as well.
121          */
122         (void) snprintf(buf, sizeof (buf),
243         (void) snprintf(current_locale_string,
244             sizeof (current_locale_string),
123             "%s/%s/%s/%s/%s/%s",
124             loc->locdata[LC_CTYPE]->l_name,
125             loc->locdata[LC_NUMERIC]->l_name,
126             loc->locdata[LC_TIME]->l_name,
127             loc->locdata[LC_COLLATE]->l_name,
128             loc->locdata[LC_MONETARY]->l_name,
129             loc->locdata[LC_MESSAGES]->l_name);
130         return (buf);
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));
131     }
133     return (loc->locdata[0]->l_name);
256     return (current_locale_string);
134 }

136 static void
137 install_legacy(locale_t loc, int mask)
259 static char *
260 loadlocale(int category)
138 {
139     /*
140      * Update the legacy fixed variables that may be baked into
141      * legacy programs. This is really unfortunate, but we can't
142      * solve for them otherwise. Note that such legacy programs
143      * are only going to see the global locale settings, and cannot
144      * benefit from uselocale().
145      */
146     if (mask & LC_NUMERIC_MASK) {
147         struct lc_numeric *lnum;
148         lnum = loc->locdata[LC_NUMERIC]->l_data[0];
149         _numeric[0] = *lnum->decimal_point;
150         _numeric[1] = *lnum->thousands_sep;
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);
151     }

153     if (mask & LC_CTYPE_MASK) {
154         struct lc_ctype *lct;
155         lct = loc->locdata[LC_CTYPE]->l_data[0];
156         for (int i = 0; i < _CACHED_RUNES; i++) {
157             /* ctype can only encode the lower 8 bits. */
158             _ctype[i+1] = lct->lc_ctype_mask[i] & 0xff;
159             _ctype_mask[i] = lct->lc_ctype_mask[i];
273     switch (category) {
274     case LC_CTYPE:
275         func = __wrap_setrunelocale;
276         break;
277     case LC_COLLATE:
278         func = __collate_load_tables;
279         break;
280     case LC_TIME:
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;
292     default:
293         errno = EINVAL;
294         return (NULL);
160     }

162     /* The bottom half is the toupper/lower array */
163     for (int i = 0; i < _CACHED_RUNES; i++) {
164         int u, l;
165         _ctype[258 + i] = i;
166         u = lct->lc_trans_upper[i];
167         l = lct->lc_trans_lower[i];
168         if (u && u != i)
169             _ctype[258+i] = u;
170         if (l && l != i)
171             _ctype[258+i] = l;
297     if (strcmp(new, old) == 0)
298         return (old);

173     /* Don't forget these annoyances either! */
174     _trans_upper[i] = u;
175     _trans_lower[i] = l;
300     if (func(new) != _LDP_ERROR) {
301         (void) strcpy(old, new);
302         return (old);
176     }

178     /* Maximum mblen, cswidth, weird legacy */
179     _ctype[520] = lct->lc_max_mblen;
180     }
305     return (NULL);
306 }

308 static const char *
309 __get_locale_env(int category)

```

_unchanged_portion_omitted_

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

1

```
*****
1608 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/setlocale.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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_ */
```

```

*****
4466 Tue May 20 07:58:50 2014
new/usr/src/lib/libc/port/locale/setrunelocale.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 */
34
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
53 extern _RuneLocale *_Read_RuneMagi(const char *);
54 extern _RuneLocale *_Read_RuneMagi(FILE *);
55 extern unsigned char __ctype_C[];
56
57 struct lc_ctype lc_ctype_posix = {
58     .lc_mbrtowc = __mbrtowc_ascii,
59     .lc_mbsinit = __mbsinit_ascii,
60     .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 };
66
67 static int
68     __setrunelocale(const char *);
69
70 struct locdata __posix_ctype_locdata = {
71     .l_lname = "C",
72     .l_refcnt = (uint32_t)-1,
73     .l_data = { &lc_ctype_posix, &_DefaultRuneLocale }
74 };
75
76 static int
77     __setrunelocale(const char *encoding)
78 {
79     FILE *fp;
80     char name[PATH_MAX];
81     _RuneLocale *rl;
82     int saverr, ret;
83     size_t (*old_mbrtowc)(wchar_t *_RESTRIC_KYWD,
84         const char *_RESTRIC_KYWD, size_t, mbstate_t *_RESTRIC_KYWD);
85     size_t (*old_wcrtomb)(char *_RESTRIC_KYWD, wchar_t,
86         mbstate_t *_RESTRIC_KYWD);
87     int (*old_mbsinit)(const mbstate_t *);
88     size_t (*old_mbsnrtowcs)(wchar_t *_RESTRIC_KYWD,
89         const char **_RESTRIC_KYWD, size_t, size_t,
90         mbstate_t *_RESTRIC_KYWD);
91     size_t (*old_wcsnrtombs)(char *_RESTRIC_KYWD,
92         const wchar_t **_RESTRIC_KYWD, size_t, size_t,
93         mbstate_t *_RESTRIC_KYWD);
94     static char ctype_encoding[ENCODING_LEN + 1];
95     static _RuneLocale *CachedRuneLocale;
96     static size_t (*Cached_mbrtowc)(wchar_t *_RESTRIC_KYWD,
97         const char *_RESTRIC_KYWD, size_t, mbstate_t *_RESTRIC_KYWD);
98     static size_t (*Cached_wcrtomb)(char *_RESTRIC_KYWD, wchar_t,
99         mbstate_t *_RESTRIC_KYWD);
100     static int (*Cached_mbsinit)(const mbstate_t *);
101     static size_t (*Cached_mbsnrtowcs)(wchar_t *_RESTRIC_KYWD,
102         const char **_RESTRIC_KYWD, size_t, size_t,
103         mbstate_t *_RESTRIC_KYWD);
104     static size_t (*Cached_wcsnrtombs)(char *_RESTRIC_KYWD,
105         const wchar_t **_RESTRIC_KYWD, size_t, size_t,
106         mbstate_t *_RESTRIC_KYWD);
107
108     74 /*
109     75  * Table of initializers for encodings. When you add a new encoding type,
110     76  * this table should be updated.
111     77  */
112     78  * The "C" and "POSIX" locale are always here.
113     79
114     80 static struct {
115     81     const char *e_name;
116     82     void (*e_init)(struct lc_ctype *);
117     83 } encodings[] = {
118     84     { "NONE", _none_init },
119     85     { "UTF-8", _UTF8_init },
120     86     { "EUC-CN", _EUC_CN_init },
121     87     { "EUC-JP", _EUC_JP_init },
122     88     { "EUC-KR", _EUC_KR_init },
123     89     { "EUC-TW", _EUC_TW_init },
124     90     { "GB18030", _GB18030_init },
125     91     { "GB2312", _GB2312_init },

```

```

90     { "GBK",      _GBK_init },
91     { "BIG5",     _BIG5_init },
92     { "MSKanji",  _MSKanji_init },
93     { NULL,       NULL };
94 };
95 if (strcmp(encoding, "C") == 0 || strcmp(encoding, "POSIX") == 0) {
96     int i;
97
98     (void) memcpy(__ctype, __ctype_C, SZ_TOTAL);
99
100 struct locdata *
101 _lc_ctype_load(const char *name)
102 {
103     struct locdata *ldata;
104     struct lc_ctype *lct;
105     _RuneLocale *rl;
106     int i;
107     char path[PATH_MAX];
108     for (i = 0; i < _CACHED_RUNES; i++) {
109         __ctype_mask[i] = _DefaultRuneLocale.__runetype[i];
110         __trans_upper[i] = _DefaultRuneLocale.__mapupper[i];
111         __trans_lower[i] = _DefaultRuneLocale.__maplower[i];
112     }
113
114     if ((ldata = _locdata_alloc(name, sizeof (*lct))) == NULL)
115         return (NULL);
116     lct = ldata->l_data[0];
117     (void) _none_init(&_DefaultRuneLocale);
118     return (0);
119 }
120
121 /*
122  * If the locale name is the same as our cache, use the cache.
123  */
124 if (CachedRuneLocale != NULL &&
125     strcmp(encoding, ctype_encoding) == 0) {
126     _CurrentRuneLocale = CachedRuneLocale;
127     __mbrtowc = Cached__mbrtowc;
128     __mbsinit = Cached__mbsinit;
129     __mbsnrtowcs = Cached__mbsnrtowcs;
130     __wctomb = Cached__wctomb;
131     __wcsnrtombs = Cached__wcsnrtombs;
132     return (0);
133 }
134
135 /*
136  * Slurp the locale file into the cache.
137  */
138
139 (void) snprintf(path, sizeof (path), "%s/%s/LC_CTYPE/LCL_DATA",
140     _PathLocale, name);
141 (void) snprintf(name, sizeof (name), "%s/%s/LC_CTYPE/LCL_DATA",
142     _PathLocale, encoding);
143
144 if ((rl = _Read_RuneMagi(path)) == NULL) {
145     _locdata_release(ldata);
146     errno = EINVAL;
147     return (NULL);
148 }
149 if ((fp = fopen(name, "r")) == NULL)
150     return (errno == 0 ? ENOENT : errno);
151
152 if ((rl = _Read_RuneMagi(fp)) == NULL) {
153     saverr = (errno == 0 ? EINVAL : errno);
154     (void) fclose(fp);
155     return (saverr);
156 }

```

```

121 ldata->l_data[1] = rl;
122 (void) fclose(fp);
123
124 lct->lc_mbrotowc = NULL;
125 lct->lc_mbsinit = NULL;
126 lct->lc_mbsnrtowcs = NULL;
127 lct->lc_wctomb = NULL;
128 lct->lc_wcsnrtombs = NULL;
129 lct->lc_ctype_mask = rl->__runetype;
130 lct->lc_trans_upper = rl->__mapupper;
131 lct->lc_trans_lower = rl->__maplower;
132 old__mbrotowc = __mbrotowc;
133 old__mbsinit = __mbsinit;
134 old__mbsnrtowcs = __mbsnrtowcs;
135 old__wctomb = __wctomb;
136 old__wcsnrtombs = __wcsnrtombs;
137
138 /* set up the function pointers */
139 for (i = 0; encodings[i].e_name != NULL; i++) {
140     int l = strlen(encodings[i].e_name);
141     if ((strcmp(rl->__encoding, encodings[i].e_name, l) == 0) &&
142         (rl->__encoding[l] == '\0' || rl->__encoding[l] == '@')) {
143         encodings[i].e_init(lct);
144         break;
145     }
146     __mbrotowc = NULL;
147     __mbsinit = NULL;
148     __mbsnrtowcs = __mbsnrtowcs_std;
149     __wctomb = NULL;
150     __wcsnrtombs = __wcsnrtombs_std;
151
152     if (strcmp(rl->__encoding, "NONE") == 0)
153         ret = _none_init(rl);
154     else if (strcmp(rl->__encoding, "UTF-8") == 0)
155         ret = _UTF8_init(rl);
156     else if (strcmp(rl->__encoding, "EUC-CN") == 0)
157         ret = _EUC_CN_init(rl);
158     else if (strcmp(rl->__encoding, "EUC-JP") == 0)
159         ret = _EUC_JP_init(rl);
160     else if (strcmp(rl->__encoding, "EUC-KR") == 0)
161         ret = _EUC_KR_init(rl);
162     else if (strcmp(rl->__encoding, "EUC-TW") == 0)
163         ret = _EUC_TW_init(rl);
164     else if (strcmp(rl->__encoding, "GB18030") == 0)
165         ret = _GB18030_init(rl);
166     else if (strcmp(rl->__encoding, "GB2312") == 0)
167         ret = _GB2312_init(rl);
168     else if (strcmp(rl->__encoding, "GBK") == 0)
169         ret = _GBK_init(rl);
170     else if (strcmp(rl->__encoding, "BIG5") == 0)
171         ret = _BIG5_init(rl);
172     else if (strcmp(rl->__encoding, "MSKanji") == 0)
173         ret = _MSKanji_init(rl);
174     else
175         ret = EINVAL;
176
177 if (ret == 0) {
178     if (CachedRuneLocale != NULL) {
179         free(CachedRuneLocale);
180     }
181     CachedRuneLocale = _CurrentRuneLocale;
182     Cached__mbrotowc = __mbrotowc;
183     Cached__mbsinit = __mbsinit;
184     Cached__mbsnrtowcs = __mbsnrtowcs;
185     Cached__wctomb = __wctomb;
186     Cached__wcsnrtombs = __wcsnrtombs;
187     (void) strcpy(ctype_encoding, encoding);

```

```

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 vaueus 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] = r1->__runetype[i] & 0xff;
210          __ctype_mask[i] = r1->__runetype[i];
211      }
212      if (encodings[i].e_name == NULL) {
213          __locdata_release(ldata);
214          errno = EINVAL;
215          return (NULL);
216      }
217
218      /* The bottom half is the toupper/lower array */
219      for (int i = 0; i < _CACHEd_RUNES; i++) {
220          __ctype[258 + i] = i;
221          if (r1->__mapupper[i] && r1->__mapupper[i] != i)
222              __ctype[258+i] = r1->__mapupper[i];
223          if (r1->__maplower[i] && r1->__maplower[i] != i)
224              __ctype[258+i] = r1->__maplower[i];
225
226          /* Don't forget these annoyances either! */
227          __trans_upper[i] = r1->__mapupper[i];
228          __trans_lower[i] = r1->__maplower[i];
229      }
230
231      /*
232       * Note that we expect the init code will have populated
233       * the CSWIDTH array (__ctype[514-520]) properly.
234       */
235      } else {
236          __mbrtowc = old__mbrtowc;
237          __mbsinit = old__mbsinit;
238          __mbsnrtowcs = old__mbsnrtowcs;
239          __wcrto mb = old__wcrto mb;
240          __wcsnrtombs = old__wcsnrtombs;
241          free(r1);
242      }
243
244      return (ldata);
245      return (ret);
246 }
247
248 int
249 __wrap_setrunelocale(const char *locale)
250 {

```

```

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_____

```

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

1

```
*****
2119 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strcasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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      */
59     if (lct->lc_is_ascii)
60     if (charset_is_ascii)
```

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

2

```
57         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
```

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

1

```
*****
2170 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strcasestr.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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];
```

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

2

```
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_____
```

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

1

```
*****
3408 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strcoll.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 };
```

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

2

```
54     if (_collate_load_error)
55         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 }
```

unchanged_portion_omitted

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

1

```
*****
16934 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strfmon.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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) { \
```

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

2

```
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     groups++; \
98 }

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 {
109     strfmon(char *_RESTRICT_KYWD s, size_t maxsize, \
110         const char *_RESTRICT_KYWD format, ...)
111     {
112         va_list ap;
113         char *dst; /* output destination pointer */
114         const char *fmt; /* current format poistion pointer */
115         struct lconv *lc; /* pointer to lconv structure */
116         char *asciivalue; /* formatted double pointer */
117
118         int flags; /* formatting options */
119         int pad_char; /* padding character */
120         int pad_size; /* pad size */
121         int width; /* field width */
122         int left_prec; /* left precision */
123         int right_prec; /* right precision */
124         double value; /* just value */
125         char space_char = ' '; /* space after currency */
126
127         char cs_precedes; /* values from struct lc_monetary */
128         char cs_follows; /* values gathered from struct lconv */
129         char sep_by_space;
130         char sign_posn;
131         const char *signstr;
132         const char *currency_symbol;
133         char *signstr;
134         char *currency_symbol;
135
136         char *tmppptr; /* temporary vars */
137         int sverrno;
138         const struct lc_monetary *lmon; /* monetary structure */
139         const struct lc_numeric *lnum; /* numeric structure */
140
141         lmon = loc->monetary;
142         lnum = loc->numeric;
143
144         if (dst >= s + maxsize)
145             goto e2big_error;
146         *dst++ = CH;
147     }
}
```

```

126 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);
241 if (flags & USE_INTL_CURRENCY) {
242     currency_symbol = lmon->int_curr_symbol;
243     /* by definition three letters followed by a space */
244     currency_symbol = strdup(lc->int_curr_symbol);
245     if (currency_symbol != NULL)
246         space_char = currency_symbol[3];
247     space_char = *(currency_symbol+3);
241 } else
242     currency_symbol = lmon->currency_symbol;
243     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 }

```

```

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     if (asciivalue != NULL)
269         free(asciivalue);
270     asciivalue = format_grouped_double(lmon, lnum, value, &flags,
271         asciivalue = __format_grouped_double(value, &flags,
272             left_prec, right_prec, pad_char);
273     if (asciivalue == NULL)
274         goto end_error; /* errno already set */
275         /* to ENOMEM by malloc() */
276
277     /* set some variables for later use */
278     setup_vars(lmon, flags, &cs_precedes, &sep_by_space,
279         &sign_posn, &signstr);
280     __setup_vars(flags, &cs_precedes, &sep_by_space, &sign_posn,
281         &signstr);
282
283     /*
284     * Description of some LC_MONETARY's values:
285     *
286     * p_cs_precedes & n_cs_precedes
287     *
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     *
294     * = 0 - no space separates $currency_symbol
295     *       from the value for a monetary quantity with a
296     *       non-negative value
297     * = 1 - space separates the symbol from the value
298     * = 2 - space separates the symbol and the sign string,
299     *       if adjacent.
300     *
301     * p_sign_posn & n_sign_posn
302     *
303     * = 0 - parentheses enclose the quantity and the
304     *       $currency_symbol
305     * = 1 - the sign string precedes the quantity and the
306     *       $currency_symbol
307     * = 2 - the sign string succeeds the quantity and the
308     *       $currency_symbol
309     * = 3 - the sign string precedes the $currency_symbol
310     * = 4 - the sign string succeeds the $currency_symbol
311     */
312
313     tmpptr = dst;
314     while (pad_size-- > 0)
315         PRINT(' ');
316     if (sign_posn == 0 && (flags & IS_NEGATIVE))
317         PRINT('(');

```

```

318     if (cs_precedes == 1) {
319         if (sign_posn == 1 || sign_posn == 3) {
320             PRINTS(signstr);
321             if (sep_by_space == 2)
322                 PRINT(' '); /* XXX: ? */
323         }
324     }
325     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 (sign_posn == 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 - tmpptr < width) {
374         if (flags & LEFT_JUSTIFY) {
375             while (dst - tmpptr < width)
376                 PRINT(' ');
377         } else {
378             pad_size = dst - tmpptr;
379             (void) memmove(tmpptr + width - pad_size, tmpptr,
380                 pad_size);
381             (void) memset(tmpptr, ' ', width - pad_size);
382             dst += width - pad_size;

```

```

383     }
384 }
385
387     PRINT('\0');
388     va_end(ap);
389     free(asciivalue);
390     free(currency_symbol);
391     return (dst - s - 1); /* size of put data except trailing '\0' */
392
393 e2big_error:
394     errno = E2BIG;
395     goto end_error;
396
397 format_error:
398     errno = EINVAL;
399
400 end_error:
401     sverrno = errno;
402     if (asciivalue != NULL)
403         free(asciivalue);
404     if (currency_symbol != NULL)
405         free(currency_symbol);
406     errno = sverrno;
407     va_end(ap);
408     return (-1);
409 }
410
411 ssize_t
412 strfmon(char *_RESTRICT_KYWD s, size_t maxsize,
413         const char *_RESTRICT_KYWD format, ...)
414 static void
415 __setup_vars(int flags, char *cs_precedes, char *sep_by_space,
416             char *sign_posn, char **signstr)
417 {
418     va_list ap;
419     ssize_t ret;
420
421     va_start(ap, format);
422     ret = strfmon_impl(s, maxsize, uselocale(NULL), format, ap);
423     va_end(ap);
424     return (ret);
425 }
426
427 struct lconv *lc = localeconv();
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

```

```

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 }
478
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 }
488
489 static int
490 calc_left_pad(const struct lc_monetary *lmon, int flags, const char *cur_symb)
491 __calc_left_pad(int flags, char *cur_symb)
492 {
493     char cs_precedes, sep_by_space, sign_posn;
494     const char *signstr;
495
496     cs_precedes = sep_by_space = sign_posn = *signstr;
497     int left_chars = 0;
498
499     setup_vars(lmon, flags, &cs_precedes, &sep_by_space, &sign_posn,
500               &signstr);
501     __setup_vars(flags, &cs_precedes, &sep_by_space, &sign_posn, &signstr);
502
503     if (cs_precedes != 0) {
504         left_chars += strlen(cur_symb);
505         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 {
530     char    *rslt;
531     char    *avalue;
532     int     avalue_size;
533     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;
544     struct lconv *lc = localeconv();
545     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) snprintf(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);
606         return (NULL);
607     }

```

```

598     bufend = rslt + bufsize - 1;    /* reserve space for trailing '\0' */

600     /* skip spaces at beginning */
601     /* skip spaces at beggining */
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         }
642     } else {
643         bufend -= avalue_size;
644         (void) memcpy(bufend, avalue+padded, avalue_size);
645         if (right_prec == 0)
646             padded--;    /* decrease assumed $decimal_point */
647     }

648     /* do padding with pad_char */
649     if (padded > 0) {
650         bufend -= padded;
651         (void) memset(bufend, pad_char, padded);
652     }

653
654     bufsize = bufsize - (bufend - rslt) + 1;
655     (void) memmove(rslt, bufend, bufsize);
656     free(avalue);
657     return (rslt);
658 }

```

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

1

```
*****
13844 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strftime.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

1 /*
2  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
3  * Copyright 2010 Nexenta Systems, Inc. All rights reserved.
4  * Copyright (c) 1989 The Regents of the University of California.
5  * All rights reserved.
6  *
7  * Redistribution and use in source and binary forms are permitted
8  * provided that the above copyright notice and this paragraph are
9  * duplicated in all such forms and that any documentation,
10 * advertising materials, and other materials related to such
11 * distribution and use acknowledge that the software was developed
12 * by the University of California, Berkeley. The name of the
13 * University may not be used to endorse or promote products derived
14 * from this software without specific prior written permission.
15 * THIS SOFTWARE IS PROVIDED 'AS IS' AND WITHOUT ANY EXPRESS OR
16 * IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
17 * WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
18 */

20 #include "lint.h"
21 #include "tzfile.h"
22 #include <fcntl.h>
23 #include <sys/stat.h>
24 #include <string.h>
25 #include <stdio.h>
26 #include <locale.h>
27 #include "timelocal.h"
28 #include "localeimpl.h"

30 static char *add(const char *, char *, const char *);
31 static char *_conv(int, const char *, char *, const char *);
32 static char *fmt(locale_t, const char *, const struct tm *, char *,
33     const char * const);
34 static char *fmt(const char *, const struct tm *, char *, const char * const);
35 static char *_yconv(int, int, int, int, char *, const char *);

36 extern char *tzname[];

38 #define IN_NONE 0
39 #define IN_SOME 1
40 #define IN_THIS 2
41 #define IN_ALL 3

43 #define PAD_DEFAULT 0
44 #define PAD_LESS 1
45 #define PAD_SPACE 2
46 #define PAD_ZERO 3

48 static const char *fmt_padding[][4] = {
49     /* DEFAULT, LESS, SPACE, ZERO */
50     #define PAD_FMT_MONTHDAY 0
51     #define PAD_FMT_HMS 0
52     #define PAD_FMT_CENTURY 0
53     #define PAD_FMT_SHORTYEAR 0
54     #define PAD_FMT_MONTH 0
55     #define PAD_FMT_WEEKOFYEAR 0
56     #define PAD_FMT_DAYOFMONTH 0
57     { "%02d", "%d", "%2d", "%02d" },
58     #define PAD_FMT_SDAYOFMONTH 1
59     #define PAD_FMT_SHMS 1
```

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

2

```
60     { "%2d", "%d", "%2d", "%02d" },
61 #define PAD_FMT_DAYOFYEAR 2
62     { "%03d", "%d", "%3d", "%03d" },
63 #define PAD_FMT_YEAR 3
64     { "%04d", "%d", "%4d", "%04d" }
65 };

68 size_t
69 strftime_l(char * _RESTRICT_KYWD s, size_t maxsize,
70     const char * _RESTRICT_KYWD format, const struct tm * _RESTRICT_KYWD t,
71     locale_t loc)
72 strftime(char * _RESTRICT_KYWD s, size_t maxsize,
73     const char * _RESTRICT_KYWD format, const struct tm * _RESTRICT_KYWD t)
74 {
75     char *p;
76
77     tzset();
78     p = _fmt(loc, ((format == NULL) ? "%c" : format), t, s, s + maxsize);
79     p = _fmt(((format == NULL) ? "%c" : format), t, s, s + maxsize);
80     if (p == s + maxsize)
81         return (0);
82     *p = '\0';
83     return (p - s);
84 }

83 size_t
84 strftime(char * _RESTRICT_KYWD s, size_t maxsize,
85     const char * _RESTRICT_KYWD format, const struct tm * _RESTRICT_KYWD t)
86 {
87     return (strftime_l(s, maxsize, format, t, uselocale(NULL)));
88 }

90 static char *
91 _fmt(locale_t loc, const char *format, const struct tm *t, char *pt,
92     const char * const ptlim)
93 _fmt(const char *format, const struct tm *t, char *pt, const char * const ptlim)
94 {
95     int Ealternative, Oalternative, PadIndex;
96     const struct lc_time *tptr = loc->time;
97     struct lc_time_T *tptr = __get_current_time_locale();

98 #define PADDING(x)    fmt_padding[x][PadIndex]

99     for (; *format; ++format) {
100         if (*format == '%') {
101             Ealternative = 0;
102             Oalternative = 0;
103             PadIndex = PAD_DEFAULT;
104 label:
105             switch (*++format) {
106             case '\\':
107                 --format;
108                 break;
109             case 'A':
110                 pt = _add((t->tm_wday < 0 ||
111                     t->tm_wday >= DAYSPERWEEK) ?
112                     "?" : tptr->weekday[t->tm_wday],
113                     pt, ptlim);
114                 continue;
115             case 'a':
116                 pt = _add((t->tm_wday < 0 ||
117                     t->tm_wday >= DAYSPERWEEK) ?
118                     "?" : tptr->wday[t->tm_wday],
119                     pt, ptlim);
120                 continue;
```

```

121         case 'B':
122             pt = _add((t->tm_mon < 0 ||
123                 t->tm_mon >= MONSPERYEAR) ?
124                 "?" : (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                 "?" : 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, t->tm_mon, t, pt, ptlim);
147             pt = _fmt(t->tm_mon, 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 %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':
185             pt = _conv(t->tm_hour, PADDING(PAD_FMT_HMS),
186                 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                 t->tm_hour : t->tm_hour - 12, 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, t->tm_mon, t, pt, ptlim);
244             pt = _fmt(t->tm_mon, t, pt, ptlim);
245             continue;
246         case 'S':
247             pt = _conv(t->tm_sec, PADDING(PAD_FMT_HMS),
248                 pt, ptlim);
249             continue;

```

```

249         case 's':
250         {
251             struct tm tm;
252             char *buf;
253
254             tm = *t;
255             (void) asprintf(&buf, "%ld", mktime(&tm));
256             pt = _add(buf, pt, ptlim);
257             continue;
258         }
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_WEEKOFEAR),
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':
285             /* ISO 8601 week number */
286         case 'G':
287             /* ISO 8601 year (four digits) */
288         case 'g':
289             /* ISO 8601 year (two digits) */
290
291         /*
292          * From Arnold Robbins' strftime version 3.0: "the week number of the
293          * year (the first Monday as the first day of week 1) as a decimal number
294          * (01-53)."
295          * (ado, 1993-05-24)
296          *
297          * From "http://www.ft.uni-erlangen.de/~mskuhn/iso-time.html" by Markus Kuhn:
298          * "Week 01 of a year is per definition the first week which has the
299          * Thursday in this year, which is equivalent to the week which contains
300          * the fourth day of January. In other words, the first week of a new year
301          * is the week which has the majority of its days in the new year. Week 01
302          * might also contain days from the previous year and the week before week
303          * 01 of a year is the last week (52 or 53) of the previous year even if
304          * it contains days from the new year. A week starts with Monday (day 1)
305          * and ends with Sunday (day 7). For example, the first week of the year
306          * 1997 lasts from 1996-12-30 to 1997-01-05..."
307          * (ado, 1996-01-02)
308          */
309         {
310             int year;
311             int base;
312             int yday;
313             int wday;
314             int w;
315
316             year = t->tm_year;
317             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;
319
320                 len = isleap_sum(year, base) ?
321                     DAYSPERLYEAR : DAYSPERNYEAR;
322                 /*
323                  * What yday (-3 ... 3) does
324                  * the ISO year begin on?
325                  */
326                 bot = ((yday + 11 - wday) %
327                       DAYSPERWEEK) - 3;
328                 /*
329                  * What yday does the NEXT
330                  * ISO year begin on?
331                  */
332                 top = bot - (len % DAYSPERWEEK);
333                 if (top < -3)
334                     top += DAYSPERWEEK;
335                 top += len;
336                 if (yday >= top) {
337                     ++base;
338                     w = 1;
339                     break;
340                 }
341                 if (yday >= bot) {
342                     w = 1 + ((yday - bot) /
343                             DAYSPERWEEK);
344                     break;
345                 }
346                 --base;
347                 yday += isleap_sum(year, base) ?
348                     DAYSPERLYEAR : DAYSPERNYEAR;
349             }
350 #ifdef XPG4_1994_04_09
351             if ((w == 52 && t->tm_mon == TM_JANUARY) ||
352                 (w == 1 && t->tm_mon == TM_DECEMBER))
353                 w = 53;
354 #endif /* defined XPG4_1994_04_09 */
355             if (*format == 'V')
356                 pt = _conv(w,
357                           PADDING(PAD_FMT_WEEKOFEAR),
358                           pt, ptlim);
359             else if (*format == 'g') {
360                 pt = _yconv(year, base, 0, 1,
361                             pt, ptlim);
362             } else
363                 pt = _yconv(year, base, 1, 1,
364                             pt, ptlim);
365         }
366         continue;
367     case 'v':
368         /*
369          * From Arnold Robbins' strftime version 3.0:
370          * "date as dd-bbb-YYYY"
371          * (ado, 1993-05-24)
372          */
373         pt = _fmt(loc, "%e-%b-%Y", t, pt, ptlim);
374         pt = _fmt("%e-%b-%Y", t, pt, ptlim);
375         continue;
376     case 'W':
377         pt = _conv((t->tm_yday + DAYSPERWEEK -
378                   (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         /*
434          * C99 says that %Z must be replaced by the
435          * empty string if the time zone is not
436          * determinable, so output nothing if the
437          * appropriate variables are not available.
438          */
439         if (t->tm_isdst == 0)
440             diff = -timezone;
441         else
442             diff = -altzone;
443         if (diff < 0) {
444             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 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strncasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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     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_strncasecmp() function.
58      */
59     if (lct->lc_is_ascii)
60     if (charset_is_ascii)
```

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

2

```
57         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_____
```

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

1

```
*****
11596 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strptime.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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;
```

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

2

```
59     const char *ptr;
60     int i, len, recurse = 0;
61     int Ealternative, Oalternative;
62     const struct lc_time *tpr = loc->time;
58     struct lc_time_T *tptr = __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, tpr->date_fmt, tm, flagsp, loc);
96                 buf = __strptime(buf, tpr->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);

248     tm->tm_hour = i;

250     break;

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     }

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     }

277     return (NULL);

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);

293     tm->tm_wday = i;
294     buf += len;
295     break;

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);

317         break;

319 case 'w':
320     if (!isdigit(*buf))
321         return (NULL);

323     i = *buf - '0';
324     if (i > 6)
325         return (NULL);

327     tm->tm_wday = i;

329     break;

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      * The %e specifier is explicitly documented as not
341      * being zero-padded but there is no harm in allowing
342      * such padding.
343      *
344      * XXX The %e specifier may gobble one too many
345      * digits if used incorrectly.
346      */
347     if (!isdigit(*buf))
348         return (NULL);

350     len = 2;
351     for (i = 0; len && isdigit(*buf); buf++) {
352         i *= 10;
353         i += *buf - '0';
354         len--;
355     }
356     if (i > 31)
357         return (NULL);

359     tm->tm_mday = i;

361     break;

363 case 'B':
364 case 'b':
365 case 'h':
366     for (i = 0; i < asizeof(tptr->month); i++) {
367         len = strlen(tptr->month[i]);
368         if (strncasecmp(buf, tptr->month[i], len) == 0)
369             break;
370     }
371     /*
372      * 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

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

1

```
*****
2747 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/strxfrm.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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)
42 strxfrm(char *_RESTRICT_KYWD xf, const char *_RESTRICT_KYWD src, size_t dlen)
43 {
44     size_t slen;
45     size_t xlen;
46     wchar_t *wcs = NULL;

47     if (!*src) {
48         if (dlen > 0)
49             *xf = '\0';
50         return (0);
51     }

52     /*
53      * The conversion from multibyte to wide character strings is
54      * strictly reducing (one byte of an mbs cannot expand to more
55      * than one wide character.)
56      */
57     slen = strlen(src);
```

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

2

```
60     if (loc->collate->lc_is_posix)
61         if (_collate_load_error)
62             goto error;

63     if ((wcs = malloc((slen + 1) * sizeof (wchar_t))) == NULL)
64         goto error;

65     if (mbstowcs_l(wcs, src, slen + 1, loc) == (size_t)-1)
66         if (mbstowcs(wcs, src, slen + 1) == (size_t)-1)
67             goto error;

68     if ((xlen = _collate_sxfrm(wcs, xf, dlen, loc)) == (size_t)-1)
69         if ((xlen = _collate_sxfrm(wcs, xf, dlen)) == (size_t)-1)
70             goto error;

71     if (wcs)
72         free(wcs);

73     if (dlen > xlen) {
74         xf[xlen] = 0;
75     } else if (dlen) {
76         xf[dlen-1] = 0;
77     }

78     return (xlen);

79 error:
80     /* errno should be set to ENOMEM if malloc failed */
81     if (wcs)
82         free(wcs);
83     (void) strcpy(xf, src, dlen);

84     return (slen);
85 }

86 size_t
87 strxfrm(char *_RESTRICT_KYWD xf, const char *_RESTRICT_KYWD src, size_t dlen)
88 {
89     return (strxfrm_l(xf, src, dlen, uselocale(NULL)));
90 }

91 _____
92 unchanged_portion_omitted_
```

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

1

```
*****
9592 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/table.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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,
```

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

2

```
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 },
191 /* 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 = &_amp;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;
```

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

1

```
*****
3197 Tue May 20 07:58:51 2014
new/usr/src/lib/libc/port/locale/timelocal.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 = {
41     {
42         "Jan", "Feb", "Mar", "Apr", "May", "Jun",
43         "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"
44     }, {
45         "January", "February", "March", "April", "May", "June",
46         "July", "August", "September", "October", "November", "December"
47     }, {
48         "Sun", "Mon", "Tue", "Wed",
49         "Thu", "Fri", "Sat"
50     }, {
51         "Sunday", "Monday", "Tuesday", "Wednesday",
52         "Thursday", "Friday", "Saturday"
53     },
54     /* x_fmt */
```

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

2

```
55     "%H:%M:%S",
56
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",
65
66     /*
67      * c_fmt
68      */
69     "%a %b %e %H:%M:%S %Y",
70
71     /* am */
72     "AM",
73
74     /* pm */
75     "PM",
76
77     /* date_fmt */
78     "%a %b %e %H:%M:%S %Z %Y",
79
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_refcnt = (uint32_t)-1,
89     .l_data = { &lc_time_posix }
90 };

92 struct locdata *
93 __lc_time_load(const char *name)
94 {
95     struct lc_time_T *
96     __get_current_time_locale(void)
97     {
98         struct locdata *ldata;
99         struct lc_time *ltime;
100         int ret;
101         return (_time_using_locale ? &_time_locale :
102             (struct lc_time_T *)&_C_time_locale);
103     }

104     if ((ldata = __locdata_alloc(name, sizeof (*ltime))) == NULL) {
105         errno = EINVAL;
106         return (NULL);
107     }
108     ltime = ldata->l_data[0];

109     ret = __part_load_locale(name, (char **)&ldata->l_data[1],
110         "LC_TIME", LCTIME_SIZE, (const char **)&ltime);

111     if (ret != _LDP_LOADED) {
112         __locdata_release(ldata);
113         errno = EINVAL;
114         return (NULL);
115     }

116     return (ldata);
117 }

118 int
```

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

3

```
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));
115 }
unchanged_portion_omitted
```

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

1

```
*****
1978 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/timelocal.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/tolower.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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]);
42 }
```

```
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]);
49 }
```

_____unchanged_portion_omitted_____

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

1

```
*****
3709 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/towlower.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 *rrr;
55     _RuneEntry *base, *re;

56     if (c < 0 || c == EOF)
57         return (c);
```

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

2

```
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 }

103 #undef toupper
104 wint_t
105 toupper(wint_t wc)
106 {
107     return (iswascii(wc) ? __trans_upper[wc] :
108         (wc < 0 || wc >= _CACHED_RUNES) ?
109         change_case_ext(uselocale(NULL), wc, 0) :
110         uselocale(NULL)->runeLocale->__mapupper[wc]);
111     return ((wc < 0 || wc >= _CACHED_RUNES) ?
112         __change_case_ext(wc, 0) :
113         _CurrentRuneLocale->__mapupper[wc]);
114 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libc/port/locale/utf8.c

1

```
*****
10018 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/utf8.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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"
37 #include "lctype.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
```

new/usr/src/lib/libc/port/locale/utf8.c

2

```
57 _UTF8_init(_RuneLocale *rl)
59 {
60     lct->lc_mbrtowc = _UTF8_mbrtowc;
61     lct->lc_wcrtomb = _UTF8_wcrtomb;
62     lct->lc_mbsinit = _UTF8_mbsinit;
63     lct->lc_mbsnrtowcs = _UTF8_mbsnrtowcs;
64     lct->lc_wcsnrtombs = _UTF8_wcsnrtombs;
65     lct->lc_is_ascii = 0;
66     lct->lc_max_mblen = 4;
67     __mbrtowc = _UTF8_mbrtowc;
68     __wcrtomb = _UTF8_wcrtomb;
69     __mbsinit = _UTF8_mbsinit;
70     __mbsnrtowcs = _UTF8_mbsnrtowcs;
71     __wcsnrtombs = _UTF8_wcsnrtombs;
72     _CurrentRuneLocale = rl;

66     charset_is_ascii = 0;

68     /*
69      * In theory up to 6 bytes can be used for the encoding,
70      * but only encodings with more than 4 bytes are illegal.
71      */
72     __ctype[520] = 4;
73     /*
74      * Note that the other CSWIDTH members are nonsensical for this
75      * this coding. They only are valid with EUC codings.
76      */

78     return (0);
67 }

unchanged_portion_omitted
```

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

1

```
*****
1962 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcrtomb.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

```
*****
2009 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcscasecmp.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 (c) 1992, 2010, Oracle and/or its affiliates. All rights reserved.
24  * Copyright 2013 Garrett D'Amore <garrett@damore.org>
25  */

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  */

33 #include "lint.h"
34 #include <stdlib.h>
35 #include <wchar.h>
36 #include <locale.h>
37 #include "libc.h"

39 /* Legacy Sun interfaces */
40 #pragma weak wcscasecmp = wcscasecmp
41 #pragma weak wcsncasecmp = wcsncasecmp

43 int
44 wcscasecmp_l(const wchar_t *s1, const wchar_t *s2, locale_t loc)
45 {
46     if (s1 == s2)
47         return (0);

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 }

58 int
59 wcscasecmp(const wchar_t *s1, const wchar_t *s2)
60 {
```

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

2

```
61     return (wcscasecmp_l(s1, s2,uselocale(NULL)));
62 }

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);

70     while ((tolower_l(*s1, loc) == tolower_l(*s2, loc)) && n-- ) {
71         if (*s1 == 0)
72             return (0);
73         s1++;
74         s2++;
75     }
76     return (tolower_l(*s1, loc) - tolower_l(*s2, loc));
77 }

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 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcscoll.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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 */
28
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"
37
38 int
39 wcscoll_l(const wchar_t *ws1, const wchar_t *ws2, locale_t loc)
40 wcscoll(const wchar_t *ws1, const wchar_t *ws2)
41 {
42     int len1, len2, pri1, pri2, ret;
43     wchar_t *tr1 = NULL, *tr2 = NULL;
44     int direc, pass;
45     const struct lc_collate *lcc = loc->collate;
46
47     if (lcc->lc_is_posix)
48         collate_info_t *info = _collate_info;
49
50     if (_collate_load_error)
51         /*
52          * Locale has no special collating order or could not be
53          * loaded, do a fast binary comparison.
54          */
55         return (wcscmp(ws1, ws2));
56
57     ret = 0;
58
59     /*
60      * 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
70             const int32_t *st1 = NULL;
71             const int32_t *st2 = NULL;
72             int32_t *st1 = NULL;
73             int32_t *st2 = NULL;
74             const wchar_t *w1 = ws1;
75             const wchar_t *w2 = ws2;
76
77             /* special pass for UNDEFINED */
78             if (pass == lcc->lc_directive_count) {
79                 if (pass == info->directive_count) {
80                     direc = DIRECTIVE_FORWARD | DIRECTIVE_UNDEFINED;
81                 } else {
82                     direc = lcc->lc_directive[pass];
83                     direc = info->directive[pass];
84                 }
85
86                 if (direc & DIRECTIVE_BACKWARD) {
87                     wchar_t *bp, *fp, c;
88                     if ((tr1 = wcsdup(w1)) == NULL)
89                         goto fail;
90                     bp = tr1;
91                     fp = tr1 + wcslen(tr1) - 1;
92                     while (bp < fp) {
93                         c = *bp;
94                         *bp++ = *fp;
95                         *fp-- = c;
96                     }
97                     if ((tr2 = wcsdup(w2)) == NULL)
98                         goto fail;
99                     bp = tr2;
100                     fp = tr2 + wcslen(tr2) - 1;
101                     while (bp < fp) {
102                         c = *bp;
103                         *bp++ = *fp;
104                         *fp-- = c;
105                     }
106                     w1 = tr1;
107                     w2 = tr2;
108                 }
109
110                 if (direc & DIRECTIVE_POSITION) {
111                     while ((*w1 || st1) && (*w2 || st2)) {
112                         pri1 = pri2 = 0;
113                         _collate_lookup(lcc, w1, &len1, &pri1, pass,
114                                         &st1);
115                         _collate_lookup(w1, &len1, &pri1, pass, &st1);
116                         if (pri1 <= 0) {
117                             if (pri1 < 0) {
118                                 errno = EINVAL;
119                                 goto fail;
120                             }
121                             pri1 = COLLATE_MAX_PRIORITY;
122                         }
123                     }
124                 }
125             }
126         }
127     }

```

```

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 (pri1 != pri2) {
128         ret = pri1 - pri2;
129         goto end;
130     }
131     w1 += len1;
132     w2 += len2;
133 } else {
134     while ((*w1 || st1) && (*w2 || st2)) {
135         pri1 = pri2 = 0;
136         while (*w1) {
137             _collate_lookup(lcc, w1, &len1,
138                             _collate_lookup(w1, &len1,
139                                             &pri1, pass, &st1);
140             if (pri1 > 0)
141                 break;
142             if (pri1 < 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 (!pri1 || !pri2)
161             break;
162         if (pri1 != pri2) {
163             ret = pri1 - 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;

```

```

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 int
194 wcscoll(const wchar_t *ws1, const wchar_t *ws2)
195 {
196     return (wcscoll_l(ws1, ws2,uselocale(NULL)));
197 }
198
199 unchanged_portion_omitted

```

```

*****
3551 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcsnrtoombs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 {
43     static mbstate_t mbs;
44
45     if (ps == NULL)
46         ps = &mbs;
47     return (loc->ctype->lc_wcsnrtoombs(dst, src, nwc, len, ps));
48 }

50 size_t
51 wcsnrtoombs(char * _RESTRICT_KYWD dst, const wchar_t ** _RESTRICT_KYWD src,
52             __wcsnrtoombs_std(char * _RESTRICT_KYWD dst, const wchar_t ** _RESTRICT_KYWD src,
53                               size_t nwc, size_t len, mbstate_t * _RESTRICT_KYWD ps)
54 {
55     return (wcsnrtoombs_l(dst, src, nwc, len, ps, uselocale(NULL)));
56 }

```

```

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                  wcrtoomb_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;
67
68     s = *src;
69     nbytes = 0;
70
71     if (dst == NULL) {
72         while (nwc-- > 0) {
73             if ((nb = pwcrtomb(buf, *s, ps)) == (size_t)-1)
74                 if ((nb = __wcrtoomb(buf, *s, ps)) == (size_t)-1)
75                     /* Invalid character - wcrtoomb() 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
85     while (len > 0 && nwc-- > 0) {
86         if (len > (size_t)MB_CUR_MAX) {
87             /* Enough space to translate in-place. */
88             if ((nb = pwcrtomb(dst, *s, ps)) == (size_t)-1) {
89                 if ((nb = __wcrtoomb(dst, *s, ps)) == (size_t)-1) {
90                     *src = s;
91                     return ((size_t)-1);
92                 }
93             } else {
94                 /*
95                  * May not be enough space; use temp. buffer.
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 = __wcrtoomb(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;
121             nbytes += nb;
122         }
123     }

```

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

3

```
120         *src = s;  
121         return (nbytes);  
122     }  
_____ unchanged_portion_omitted
```

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

1

2007 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcsrtombs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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_wcsnrtombs(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 (__wcsnrtombs(dst, src, ULONG_MAX, len, ps));
54 }
```

_____unchanged_portion_omitted_____

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

1

```
*****
2054 Tue May 20 07:58:52 2014
new/usr/src/lib/libc/port/locale/wcstombs.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****
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));
55 }

_____unchanged_portion_omitted_____
```

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

1

```
*****
2530 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wcswidth.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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     return (len);
59 }
```

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

2

```
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 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wcsxfrm.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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)
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;
55     }

```

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

2

```
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) wcscpy(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 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wctob.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 {
40     static const mbstate_t initial = { 0 };
41     mbstate_t mbs = initial;
42     char buf[MB_LEN_MAX];

44     if (c == WEOF || wctomb_l(buf, c, &mbs, loc) != 1)
45         if (c == WEOF || __wctomb(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_
```

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

1

2018 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wctomb.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 <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 wctomb(char *s, wchar_t wchar)
39 {
40     static const mbstate_t initial = { 0 };
41     static mbstate_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 = wctomb_l(s, wchar, &mbs, loc)) == (size_t)-1)
49     if ((rval = __wctomb(s, wchar, &mbs)) == (size_t)-1)
50         return (-1);
51     return ((int)rval);
52 }

53 int
54 wctomb(char *s, wchar_t wchar)
55 {
56     return (wctomb_l(s, wchar, uselocale(NULL)));
57 }

unchanged_portion_omitted
```

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

1

```
*****
2642 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wctrans.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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         break;
58     }
59 }
```

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

2

```
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 {
74     struct {
75         const char    *name;
76         wctrans_t     trans;
77     } ccls[] = {
78         { "tolower",    _WCT_TOLOWER },
79         { "toupper",    _WCT_TOUPPER },
80         { NULL,         _WCT_ERROR },      /* Default */
81     };
82     int i;
83     _NOTE(ARGUNUSED(loc));

85     i = 0;
86     while (ccls[i].name != NULL && strcmp(ccls[i].name, charclass) != 0)
87         i++;

89     if (ccls[i].trans == _WCT_ERROR)
90         errno = EINVAL;
91     return (ccls[i].trans);
92 }

94 wctrans_t
95 wctrans(const char *charclass)
96 {
97     return (wctrans_l(charclass, uselocale(NULL)));
98 }

unchanged_portion_omitted
```

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

1

```
*****
2693 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wctype.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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)
45 {
46     static const struct {
47         const char *name;
48         wctype_t mask;
49     } props[] = {
50         { "alnum", _CTYPE_A|_CTYPE_D },
51         { "alpha", _CTYPE_A },
52         { "blank", _CTYPE_B },
53         { "cntrl", _CTYPE_C },
54         { "digit", _CTYPE_D },
55         { "graph", _CTYPE_G },
56         { "lower", _CTYPE_L },
57         { "print", _CTYPE_R },
58         { "punct", _CTYPE_P },
59         { "space", _CTYPE_S },
60         { "upper", _CTYPE_U },
```

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

2

```
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 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/locale/wcwidth.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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;

58     if (wc == 0)
59         return (0);
```

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

2

```
61     x = ((wc < 0 || wc >= _CACHED_RUNES) ? __runetype(rl, wc) :
62          rl->__runetype[wc] & (_CTYPE_SWM|_CTYPE_R));
63     x = ((wc < 0 || wc >= _CACHED_RUNES) ? __runetype(wc) :
64          _CurrentRuneLocale->__runetype[wc] & (_CTYPE_SWM|_CTYPE_R));

65     if ((x & _CTYPE_SWM) != 0)
66         return ((x & _CTYPE_SWM) >> _CTYPE_SWS);
67     return ((x & _CTYPE_R) != 0 ? 1 : -1);

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
```

new/usr/src/lib/libc/port/mapfile-vers

1

```

*****
56255 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/port/mapfile-vers
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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

```

new/usr/src/lib/libc/port/mapfile-vers

2

```

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         __mb_cur_max;
99         btowc_l;
100         duplocale;
101         fgetwc_l;
102         freelocale;
103         getwc_l;
104         isalnum_l;
105         isalpha_l;
106         isblank_l;
107         iscntrl_l;
108         isdigit_l;
109         isgraph_l;
110         islower_l;
111         isprint_l;
112         ispunct_l;
113         isspace_l;
114         isupper_l;
115         iswdeogram;
116         iswdeogram_l;
117         iswnumber;
118         iswnumber_l;
119         iswhexnumber;
120         iswhexnumber_l;
121         iswphonogram;
122         iswphonogram_l;
123         iswspecial;
124         iswspecial_l;
125         iswalnum_l;

```

```
126     iswalpha_l;
127     iswblank_l;
128     iswcntrl_l;
129     iswctype_l;
130     iswdigit_l;
131     iswgraph_l;
132     iswlower_l;
133     iswprint_l;
134     iswpunct_l;
135     iswspace_l;
136     iswupper_l;
137     mblen_l;
138     mbrlen_l;
139     mbsinit_l;
140     mbsnrtowcs;
141     mbsnrtowcs_l;
142     mbsrtowcs_l;
143     mbstowcs_l;
144     mbtowc_l;
145     newlocale;
146     nl_langinfo_l;
147     strcasecmp_l;
148     strcasestr_l;
149     strcoll_l;
150     strfmon_l;
151     strftime_l;
152     strncasecmp_l;
153     strptime_l;
154     strxfrm_l;
155     tolower_l;
156     toupper_l;
157     towlower_l;
158     towupper_l;
159     towctrans_l;
160    uselocale;
161     wctomb_l;
162     wcscasecmp_l;
163     wcscoll_l;
164     wcsncasecmp_l;
165     wcsrtombs_l;
166     wcstombs_l;
167     wcswidth_l;
168     wcsxfrm_l;
169     wctob_l;
170     wctomb_l;
171     wctrans_l;
172     wctype_l;
173     wcwidth_l;
174 } ILLUMOS_0.7;

176 SYMBOL_VERSION ILLUMOS_0.7 {      # Illumos additions
177     protected:
178         _glob_ext;
179         _globfree_ext;
180 } ILLUMOS_0.6;
unchanged_portion_omitted
```

new/usr/src/lib/libc/sparc/Makefile.com

1

25827 Tue May 20 07:58:53 2014
new/usr/src/lib/libc/sparc/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 LIBC_DIR= $(SRC)/lib/libc
32 LIB_PIC= lib_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 $(LIBC_DIR)/capabilities/sun4u/sparc/pics/symcap.o \
42 $(LIBC_DIR)/capabilities/sun4u-opl/sparc/pics/symcap.o \
43 $(LIBC_DIR)/capabilities/sun4u-us3-hwcap1/sparc/pics/symcap.o \
44 $(LIBC_DIR)/capabilities/sun4u-us3-hwcap2/sparc/pics/symcap.o \
45 $(LIBC_DIR)/capabilities/sun4v-hwcap1/sparc/pics/symcap.o \
46 $(LIBC_DIR)/capabilities/sun4v-hwcap2/sparc/pics/symcap.o
```

48 # local objects

```
49 STRETS= \
50 stret1.o \
51 stret2.o \
52 stret4.o
```

```
54 CRTOBS= \
55 _ftou.o \
56 cerror.o \
57 cerror64.o \
58 hwmuldiv.o
```

```
60 DYNOBJS= \
```

new/usr/src/lib/libc/sparc/Makefile.com

2

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      errlist.o         \
146      getcctxt.o        \
147      ladd.o            \
148      lmul.o            \
149      lock.o            \
150      lshiftl.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_recvmmsg.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      gttty.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      priocntlset.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      sigprocmask.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  SYSOBSJS=            \
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      ashldi3.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      getsptent.o     \
472      getsptent_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      lckpwwd.o       \
497      lconstants.o    \
498      ldivide.o       \
499      lexp10.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      mkdtemp.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      ssignal.o         \
577      stack.o           \
578      stpcpy.o          \
579      stpncpy.o         \
580      str2sig.o         \
581      strcase_charmap.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      strtimax.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 \
661 \
662 PORTSTDIO_C89= \
663     vscanf_c89.o \
664     wscanf_c89.o \
665 \
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 \
671 \
672 PORTSTDIO_W= \
673     doscan_w.o \
674 \
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     wscanf.o \
730     wscanf.o \
731 \
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     wchr.o \
748     wcmp.o \
749     wcpy.o \
750     wscspn.o \
751     wsdup.o \
752     wslen.o \
753     wncasecmp.o \
754     wncat.o \
755     wncmp.o \
756     wncpy.o \
757     wspbrk.o \
758     wprintf.o \
759     wsrchr.o \
760     wscanf.o \
761     wsspn.o \
762     wstod.o \
763     wstok.o \
764     wstol.o \
765     wstoll.o \
766     wsxfrm.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 \
783 \
784 PORTI18N_COND= \
785     wcstol_longlong.o \
786     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     ldpart.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     mbsnrtowcs.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     regex.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     strptime.o \
835     strxfrm.o \
836     table.o \
837     timelocal.o \
838     tolower.o \
839     towlower.o \
840     ungetwc.o \
841     utf8.o \
842     wcrtoomb.o \
843     wcscasecmp.o \
844     wcscoll.o \
845     wcsftime.o \
846     wcsnrtombs.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 \
857 \
858     AI00BJS= \
859     aio.o \
860     aio_alloc.o \
861     posix_aio.o \
862 \
863     RT0BJS= \
864     clock_timer.o \
865     mqueue.o \
866     pos4obj.o \
867     sched.o \
868     sem.o \
869     shm.o \
870     sigev_thread.o \
871 \
872     TP00LOBJS= \
873     thread_pool.o \
874 \
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     synch.o \
893     tdb_agent.o \
894     thr.o \
895     thread_interface.o \
896     tls.o \
897     tsd.o \
898 \
899     THREADSMACHOBJS= \
900     machdep.o \
901 \
902     THREADSASMOBJS= \
903     asm_subr.o \
904 \
905     UNICODEOBJS= \
906     u8_textprep.o \
907     uconv.o \
908 \
909     UNWINDMACHOBJS= \
910     unwind.o \
911 \
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     exaccts.o \
929     execl.o \
930     execle.o \
931     execv.o \
932     fcntl.o \
933     getpagesizes.o \
934     getpeerucrd.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     pgrpsys.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_errno.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 MOSTOBSJ= \
987     $(STRETS) \
988     $(CRTOBSJ) \
989     $(DYNOBJS) \
990     $(FPOBJS) \
991     $(FPASMOBJS) \
992     $(ATOMICOBJS) \
993     $(XATTROBJS) \
994     $(COMOBJS) \
995     $(DTRACEOBJS) \
996     $(GENOBJS) \
997     $(PRFOBJS) \
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     $(AI0OBJS) \
1015     $(RT0BJS) \
1016     $(TP00LOBJS) \
1017     $(THREADSOBJS) \
1018     $(THREADSMACH0BJS) \
1019     $(THREADSASMOBJS) \
1020     $(UNICODEOBJS) \
1021     $(UNWINDMACH0BJS) \
1022     $(UNWINDASMOBJS) \
1023     $(COMSYS0BJS) \
1024     $(SYS0BJS) \
1025     $(COMSYS0BJS64) \
1026     $(SYS0BJS64) \
1027     $(VALUES)

1029 TRACE0BJS= \
1030     plockstat.o

1032 # NOTE: libc.so.1 must be linked with the minimal crt0.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 crt0.o and
1040 # crtn.o modules is safe and avoids the problems described above.
1041 OBJECTS= $(CRTI) $(MOSTOBSJ) $(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= $(TRACEOBSJ:=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) $(MOSTOBSJ:=$(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 CLOBBERVERFILES += $(LIB_PIC)

1142 # list of C source for lint
1143 SRCS= \
1144 $(ATOMICOBJS:=.o=$(SRC)/common/atomic/%.c) \
1145 $(XATROBJS:=.o=$(SRC)/common/xattr/%.c) \
1146 $(COMOBJS:=.o=$(SRC)/common/util/%.c) \
1147 $(DTRACEOBSJ:=.o=$(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 $(TPOOLOBJS:=.o=$(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_clrptr.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/siginfofst.c \
1172 $(LIBCBASE)/gen/siglongjmp.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     getctx.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     swapctx.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 = -x04 -dalign
1254 pics/__quad.o := sparc_COPTFLAG = -x04 -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 = -x04

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-1c
1305 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1306 $(LINTLIB) := LINTFLAGS=-nvx

1308 # object files that depend on inline template

```

new/usr/src/lib/libc/sparc/Makefile.com

21

```
1309 $(TIL:%=pics/%): $(LIBCBASE)/threads/sparc.il
1310 $(IL:%=pics/%): $(LIBCBASE)/fp/base.il
1311 $(QIL:%=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>&1
1321     $(MCS) -d -n .SUNW_dof $@ > /dev/null 2>&1
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
```

new/usr/src/lib/libc/sparc/Makefile.com

22

```
1376 pics/errlst.o: $(LIBCDIR)/port/gen/errlst.c
1378 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c
```

new/usr/src/lib/libc/sparcv9/Makefile.com

1

24490 Tue May 20 07:58:54 2014
new/usr/src/lib/libc/sparcv9/Makefile.com
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>

```
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 CRTOBS=      \
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=      \
```

new/usr/src/lib/libc/sparcv9/Makefile.com

2

```
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 $(__GNUC)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 GENOBSJ=          \
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      getcctxt.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 COMSYSOBSJ64=

```

```

168 SYSOBSJ64=

```

```

170 COMSYSOBSJ=      \
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_recvmmsg.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      facld.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      gttty.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      getdtablesize.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      lckpwwdf.o     \
459      lconstants.o   \
460      ldivide.o       \
461      lexp10.o        \
462      lfind.o         \
463      lfnt.o          \
464      lfnt_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_charmap.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      strtointmax.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           \
597
598      PORTPRINT_W=      \
599      doprnt_w.o        \
600
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         \
614
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=
618
619      PORTSTDIO_W=      \
620      doscan_w.o        \
621
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      ferror.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        \
682      strcasecmp.o   \
683      strcasestr.o   \
684      strncasecmp.o   \
683      strtows.o     \
684      wcsnlen.o      \
685      wcstoimax.o    \
686      wcstol.o       \
687      wcstoul.o      \
688      wcswcs.o       \
691      wscasecmp.o    \
689      wscat.o        \
690      wscr.o         \
691      wscmp.o        \
692      wscpy.o        \
693      wscspn.o       \
694      wsdup.o        \
695      wslen.o        \
699      wsncasecmp.o   \
696      wsnecat.o      \
697      wsncmp.o       \
698      wsncpy.o       \
699      wspbrk.o       \
700      wprintf.o      \
701      wsrchr.o       \
702      wsscanf.o      \
703      wsspn.o        \
704      wstod.o        \
705      wstok.o        \
706      wstol.o        \
707      wstoll.o       \
708      wsxfrm.o       \
709      wmemchr.o      \
710      wmemcmp.o      \
711      wmemcpy.o      \
712      wmemmove.o     \
713      wmemset.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       \
727      isdigit.o       \
723      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      ldpair.o        \
749      lmessages.o     \
750      lnumeric.o      \
751      lmonetary.o     \
752      localeconv.o    \
753      localeimpl.o    \
754      mbtowc.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      regex.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      wcrtoomb.o          \
790      wscasecmp.o         \
791      wscoll.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      AI00BJS=           \
806      aio.o               \
807      aio_alloc.o         \
808      posix_aio.o         \
810      RT0BJS=            \
811      clock_timer.o       \
812      mqueue.o            \
813      pos4obj.o           \
814      sched.o             \
815      sem.o               \
816      shm.o               \
817      sigev_thread.o      \
819      TP00LOBJS=         \
820      thread_pool.o       \
822      THREADS0BJS=       \
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      THREADSMACH0BJS=   \

```

```

847      machdep.o          \
849      THREADSASMOBJS=    \
850      asm_subr.o          \
852      UNICODE0BJS=       \
853      u8_textprep.o       \
854      uconv.o             \
856      UNWINDMACH0BJS=    \
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      exacetsys.o         \
875      execl.o             \
876      execle.o            \
877      execv.o             \
878      fcntl.o             \
879      getpagesizes.o      \
880      getpeerucrd.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 crt1.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 crt1.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=tls_rtldinfo
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 += -znw

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 $(TPOOLOBJS:%.o=$(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/siginfolst.c \
1106 $(LIBCBASE)/gen/siglongjmp.c \
1107 $(LIBCBASE)/gen/swapctxt.c

1109 # conditional assignments
1110 $(DYNLIB) := CRTI = crt1.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_qtou.o \
1176 _Q_sqrt.o \

```

```

1177     _Q_stoq.o           \
1178     _Q_sub.o            \
1179     _Qp_qtox.o          \
1180     _Qp_qtox.o
1182 $(QIL:%=pics%) := CFLAGS64 += $(LIBCDIR)/$(MACH)/fp/___quad.il
1183 pics/_Qp%.o := CFLAGS64 += -I$(LIBCDIR)/$(MACH)/fp
1184 pics/_Q%.o := sparcv9_COPTFLAG = -x04 -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_LONGLONG

1216 # Files which need extra optimization
1217 pics/getenv.o := sparcv9_COPTFLAG = -x04

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

```

new/usr/src/lib/libumem/common/stub_stand.c

1

```
*****
2452 Tue May 20 07:58:54 2014
new/usr/src/lib/libumem/common/stub_stand.c
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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':
```

new/usr/src/lib/libumem/common/stub_stand.c

2

```
160     case '\v':
161         return (1);
162     }
163     return (0);
164 }
    unchanged_portion_omitted_
```

8360 Tue May 20 07:58:54 2014
 new/usr/src/man/man3c/ctype.3c
 2964 need POSIX 2008 locale object support
 Reviewed by: Robert Mustacchi <rm@joyent.com>

```

1  \" te
2  \" Copyright 2013 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 \"Sep 18, 2013\"
12 .TH CTYPE 3C \"Jan 28, 2005\"
13 .SH NAME
14 ctype, isalpha, isalnum, isascii, isblank, iscntrl, isdigit, islower, isprint,
15 isspace, isupper, ispunct, isgraph, isxdigit \- character handling
16 .SH SYNOPSIS
17 .LP
18 #include <ctype.h>

20 \fBint\fR \fBisalpha\fR(\fBint\fR \fIc\fR);
21 .fi

23 .LP
24 .nf
25 \fBint\fR \fBisalnum\fR(\fBint\fR \fIc\fR);
26 .fi

28 .LP
29 .nf
30 \fBint\fR \fBisascii\fR(\fBint\fR \fIc\fR);
31 .fi

33 .LP
34 .nf
35 \fBint\fR \fBisblank\fR(\fBint\fR \fIc\fR);
36 .fi

38 .LP
39 .nf
40 \fBint\fR \fBiscntrl\fR(\fBint\fR \fIc\fR);
41 .fi

43 .LP
44 .nf
45 \fBint\fR \fBisdigit\fR(\fBint\fR \fIc\fR);
46 .fi

48 .LP
49 .nf
50 \fBint\fR \fBisgraph\fR(\fBint\fR \fIc\fR);
51 .fi

53 .LP
54 .nf
55 \fBint\fR \fBislower\fR(\fBint\fR \fIc\fR);
56 .fi

58 .LP
59 .nf

```

```

60 \fBint\fR \fBisprint\fR(\fBint\fR \fIc\fR);
61 .fi

63 .LP
64 .nf
65 \fBint\fR \fBispunct\fR(\fBint\fR \fIc\fR);
66 .fi

68 .LP
69 .nf
70 \fBint\fR \fBisspace\fR(\fBint\fR \fIc\fR);
71 .fi

73 .LP
74 .nf
75 \fBint\fR \fBisupper\fR(\fBint\fR \fIc\fR);
76 .fi

78 .LP
79 .nf
80 \fBint\fR \fBisxdigit\fR(\fBint\fR \fIc\fR);
81 .fi

83 .SH DESCRIPTION
84 .sp
85 .LP
86 These macros classify character-coded integer values. Each is a predicate
87 returning non-zero for true, \fB0\fR for false. The behavior of these macros,
88 except \fBisascii()\fR, is affected by the current locale (see
89 \fBsetlocale\fR(3C)). To modify the behavior, change the \fBLC_TYPE\fR category
90 in \fBsetlocale()\fR, that is, \fBsetlocale(\fR\FBLC_CTYPE\fR,
91 \fBnewlocale\fR). In the \"C\" locale, or in a locale where character type
92 information is not defined, characters are classified according to the rules of
93 the \fBUS-ASCII\fR 7-bit coded character set.
94 .sp
95 .LP
96 The \fBisascii()\fR macro is defined on all integer values. The rest are
97 defined only where the argument is an \fBint\fR, the value of which is
98 representable as an \fBunsigned char\fR, or \fBEOF\fR, which is defined by the
99 <\fBstdio.h\fR> header and represents end-of-file.
100 .sp
101 .LP
102 Functions exist for all the macros defined below. To get the function form, the
103 macro name must be undefined (for example, \fB#undef isdigit\fR).
103 .sp
104 .LP
105 For macros described with \fBDefault\fR and \fBStandard conforming\fR versions,
106 standard-conforming behavior is provided for standard-conforming applications
107 (see \fBstandards\fR(5)) and for applications that define
108 \fB_XPG4_CHAR_CLASS_\fR before including <\fBctype.h\fR>.
109 .SS \"Default\"
110 .sp
111 .ne 2
112 .na
113 \fB\fBisalpha()\fR\fR
114 .ad
115 .RS 13n
116 Tests for any character for which \fBisupper()\fR or \fBislower()\fR is true.
117 .RE

119 .SS \"Standard conforming\"
120 .sp
121 .ne 2
122 .na
123 \fB\fBisalpha()\fR\fR
124 .ad

```

```

110 .RS 13n
111 Tests for any character for which \fBisupper()\fR or \fBislower()\fR is true,
112 or any character that is one of the current locale-defined set of characters
113 for which none of \fBisctrl()\fR, \fBisdigit()\fR, \fBispunct()\fR, or
114 \fBisspace()\fR is true. In "C" locale, \fBisalpha()\fR returns true only for
115 the characters for which \fBisupper()\fR or \fBislower()\fR is true.
116 .RE

118 .sp
119 .ne 2
120 .na
121 \fB\fBisalnum()\fR\fR
122 .ad
123 .RS 13n
124 Tests for any character for which \fBisalpha()\fR or \fBisdigit()\fR is true
125 (letter or digit).
126 .RE

128 .sp
129 .ne 2
130 .na
131 \fB\fBisascii()\fR\fR
132 .ad
133 .RS 13n
134 Tests for any ASCII character, code between \fB0\fR and \fB0177\fR inclusive.
135 .RE

137 .sp
138 .ne 2
139 .na
140 \fB\fBisblank()\fR\fR
141 .ad
142 .RS 13n
143 Tests whether \fIc\fR is a character of class blank in the current locale. This
144 macro/function is not available to applications conforming to standards prior
145 to SUSv3. See \fBstandards\fR(5)
146 .RE

148 .sp
149 .ne 2
150 .na
151 \fB\fBisctrl()\fR\fR
152 .ad
153 .RS 13n
154 Tests for any "control character" as defined by the character set.
155 .RE

157 .sp
158 .ne 2
159 .na
160 \fB\fBisdigit()\fR\fR
161 .ad
162 .RS 13n
163 Tests for any decimal-digit character.
164 .RE

181 .SS "Default"
166 .sp
167 .ne 2
168 .na
169 \fB\fBisgraph()\fR\fR
170 .ad
171 .RS 13n
188 Tests for any character for which \fBispunct()\fR, \fBisupper()\fR,
189 \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
172 Tests for any character for which \fBisalnum()\fR and \fBispunct()\fR are true,
173 or any character in the current locale-defined "graph" class which is neither a
174 space ("\|") nor a character for which \fBisctrl()\fR is true.
175 .RE

177 .sp
178 .ne 2
179 .na
180 \fB\fBislower()\fR\fR
181 .ad
182 .RS 13n
183 Tests for any character that is a lower-case letter or is one of the current
184 locale-defined set of characters for which none of \fBisctrl()\fR,
185 \fBisdigit()\fR, \fBispunct()\fR, \fBisspace()\fR, or \fBisupper()\fR is true.
186 In the "C" locale, \fBislower()\fR returns true only for the characters defined
187 as lower-case \fBASCII\fR characters.
188 .RE

217 .SS "Default"
190 .sp
191 .ne 2
192 .na
193 \fB\fBisprint()\fR\fR
194 .ad
195 .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
196 Tests for any character for which \fBisctrl()\fR is false, and
197 \fBisalnum()\fR, \fBisgraph()\fR, \fBispunct()\fR, the space character ("\|"),
198 and the characters in the current locale-defined "print" class are true.
199 .RE

201 .sp
202 .ne 2
203 .na
204 \fB\fBispunct()\fR\fR
205 .ad
206 .RS 13n
207 Tests for any printing character which is neither a space ("\|") nor a
208 character for which \fBisalnum()\fR or \fBisctrl()\fR is true.
209 .RE

211 .sp
212 .ne 2
213 .na
214 \fB\fBisspace()\fR\fR
215 .ad
216 .RS 13n
217 Tests for any space, tab, carriage-return, newline, vertical-tab or form-feed

```

```
218 (standard white-space characters) or for one of the current locale-defined set
219 of characters for which \fBisalnum()\fR is false. In the "C" locale,
220 \fBisspace()\fR returns true only for the standard white-space characters.
221 .RE

223 .sp
224 .ne 2
225 .na
226 \fB\fBisupper()\fR\fR
227 .ad
228 .RS 13n
229 Tests for any character that is an upper-case letter or is one of the current
230 locale-defined set of characters for which none of \fBisctrl()\fR,
231 \fBisdigit()\fR, \fBispunct()\fR, \fBisspace()\fR, or \fBislower()\fR is true.
232 In the "C" locale, \fBisupper()\fR returns true only for the characters defined
233 as upper-case \fBASCII\fR characters.
234 .RE

275 .SS "Default"
276 .sp
277 .ne 2
278 .na
279 \fB\fBisxdigit()\fR\fR
280 .ad
281 .RS 14n
282 Tests for any hexadecimal-digit character (\fB[0\(\mi9)\fR, \fB[A\(\miF)\fR, or
283 \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
294 \fB[a\(\mif)\fR or the current locale-defined sets of characters representing
295 the hexadecimal digits \fB10\fR to \fB15\fR inclusive). In the "C" locale, only
296 .sp
297 .in +2
298 .nf
299 0 1 2 3 4 5 6 7 8 9 A B C D E F a b c d e f
300 .fi
301 .in -2

352 are included.
353 .RE

355 .SH RETURN VALUES
356 .sp
357 .LP
358 If the argument to any of the character handling macros is not in the domain of
359 the function, the result is undefined. Otherwise, the macro or function returns
360 non-zero if the classification is \fBTRUE\fR and \fB0\fR if the classification
361 is \fBFALSE\fR.
362 .SH USAGE
363 .sp
364 .LP
365 These macros or functions can be used safely in multithreaded applications, as
366 long as \fBsetlocale\fR(3C) is not being called to change the locale.
367 .SH ATTRIBUTES
368 .sp
369 .LP
370 See \fBattributes\fR(5) for descriptions of the following attributes:
371 .sp
```

```
273 .sp
274 .TS
275 box;
276 l l
277 l l .
278 ATTRIBUTE TYPE ATTRIBUTE VALUE
279 CSI Enabled
280 Interface Stability Standard
281 MT-Level MT-Safe with exceptions
282 .TE

284 .SH SEE ALSO
285 .sp
286 .LP
287 \fBsetlocale\fR(3C), \fBstdio\fR(3C), \fBascii\fR(5), \fBenviron\fR(5),
288 \fBstandards\fR(5)
```

new/usr/src/pkg/manifests/system-header.mf

1

```
*****
89943 Tue May 20 07:58:54 2014
new/usr/src/pkg/manifests/system-header.mf
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 (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 2013 Garrett D'Amore <garrett@damore.org>
27 #

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
58 dir path=usr/include/nfs
59 dir path=usr/include/protocols
60 dir path=usr/include/rpc
```

new/usr/src/pkg/manifests/system-header.mf

2

```
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,Serverblade1 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/appttrace.h
197 file path=usr/include/appttrace_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/cpvar.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/dldefs.h
258 file path=usr/include/ast/dt.h
```

```
259 file path=usr/include/ast/endian.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/lc.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/ast/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/deflt.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/ieeeep.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_ftable.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/imp.h
435 file path=usr/include/imp_admin.h
436 file path=usr/include/imp_mpathd.h
437 file path=usr/include/imp_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/libdladm.h
479 file path=usr/include/libdlbridge.h
480 file path=usr/include/libdlib.h
481 file path=usr/include/libdlink.h
482 file path=usr/include/libdmpi.h
483 file path=usr/include/libdlvlan.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/libgrubmgmt.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/libnwam.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/ndbm.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_sysm.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_seq.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/tcpi.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/pfamt.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/pkgtrans.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/regex.h
644 file path=usr/include/regex.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/rpcsec_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/libsmpt.h
742 file path=usr/include/scsi/libsmpt_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/sharet.h
766 file path=usr/include/siginfo.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/aioCb.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/condvar.h
842 file path=usr/include/sys/condvar_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/dnlc.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/ecpps.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/pvfs_ki.h
1033 file path=usr/include/sys/fs/sdev_impl.h
1034 file path=usr/include/sys/fs/snode.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/ftrace.h
1065 file path=usr/include/sys/fx.h
1066 file path=usr/include/sys/fxpriocntl.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/pci.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/ibnexus/ibnexus_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/llc1.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/mmapiobj.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/mode.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/nbmlock.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/pattr.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/prsystm.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/rsmndi.h
1352 file path=usr/include/sys/rsm/rsmapi.h
1353 file path=usr/include/sys/rsm/rsmapi_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/stddef.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/semaphore.h
1415 file path=usr/include/sys/sendfile.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/sobject.h
1435 $(sparc_ONLY)file path=usr/include/sys/socal_cq_defs.h
1436 $(sparc_ONLY)file path=usr/include/sys/socalio.h
1437 $(sparc_ONLY)file path=usr/include/sys/socalmap.h
1438 $(sparc_ONLY)file path=usr/include/sys/socalreg.h
1439 $(sparc_ONLY)file path=usr/include/sys/socalvar.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/sysevent.h
1484 file path=usr/include/sys/sysevent/ap_driver.h
1485 file path=usr/include/sys/sysevent/dev.h
1486 file path=usr/include/sys/sysevent/domain.h
1487 file path=usr/include/sys/sysevent/dr.h
1488 file path=usr/include/sys/sysevent/env.h
1489 file path=usr/include/sys/sysevent/eventdefs.h
1490 file path=usr/include/sys/sysevent/imp.h
1491 file path=usr/include/sys/sysevent/pwrctl.h
1492 file path=usr/include/sys/sysevent/svm.h
1493 file path=usr/include/sys/sysevent/vrrp.h
1494 file path=usr/include/sys/sysevent_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/sysmacros.h
1499 file path=usr/include/sys/systeminfo.h
1500 file path=usr/include/sys/systm.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/teliocctl.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/tspriocntl.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/uio.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/sysexits.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/ucrd.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/memnode.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/i86xpv/include/sys/balloon.h
1721 $(i386_ONLY)file path=usr/platform/i86xpv/include/sys/machprivregs.h
1722 $(i386_ONLY)file path=usr/platform/i86xpv/include/sys/xen_mmu.h
1723 $(i386_ONLY)file path=usr/platform/i86xpv/include/sys/xpv_impl.h
1724 $(i386_ONLY)file path=usr/platform/i86xpv/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/async.h
1727 $(sparc_ONLY)file path=usr/platform/sun4u/include/sys/cheetahregs.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/cvc.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/eeeprom.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/enviro.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/trap.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/zsmach.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/sun4u/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/eeeprom.h
1811 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/fcode.h
1812 $(sparc_ONLY)file path=usr/platform/sun4v/include/sys/hsv.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/niagaraargs.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/siginfo.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/utsname.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/wordep.h.3head
1937 file path=usr/share/man/man4/note.4
1938 file path=usr/share/man/man5/prof.5
1939 file path=usr/share/man/man7i/cdio.7i
1940 file path=usr/share/man/man7i/dkio.7i
1941 file path=usr/share/man/man7i/fbio.7i
1942 file path=usr/share/man/man7i/fdio.7i
1943 file path=usr/share/man/man7i/hdio.7i
1944 file path=usr/share/man/man7i/iec61883.7i
1945 file path=usr/share/man/man7i/mhd.7i
1946 file path=usr/share/man/man7i/mtio.7i
1947 file path=usr/share/man/man7i/prnio.7i
1948 file path=usr/share/man/man7i/quotactl.7i
1949 file path=usr/share/man/man7i/sesio.7i
1950 file path=usr/share/man/man7i/sockio.7i
1951 file path=usr/share/man/man7i/streamio.7i
1952 file path=usr/share/man/man7i/termio.7i
1953 file path=usr/share/man/man7i/termiox.7i
1954 file path=usr/share/man/man7i/uscsi.7i
1955 file path=usr/share/man/man7i/visual_io.7i
1956 file path=usr/share/man/man7i/vt.7i
1957 file path=usr/xpg4/include/curses.h
1958 file path=usr/xpg4/include/term.h
1959 file path=usr/xpg4/include/unctrl.h
1960 legacy pkg=SUNWhea \
1961 desc="SunOS C/C++ header files for general development of software" \
1962 name="SunOS Header Files"
1963 license cr_Sun license=cr_Sun
1964 license lic_CDDL license=lic_CDDL
1965 license license_in_headers license=license_in_headers
1966 license usr/src/lib/pkcs11/include/THIRDPARTYLICENSE \
1967 license=usr/src/lib/pkcs11/include/THIRDPARTYLICENSE
1968 link path=usr/include/iso/assert_iso.h target=../assert.h
1969 link path=usr/include/iso/errno_iso.h target=../errno.h
1970 link path=usr/include/iso/float_iso.h target=../float.h
1971 link path=usr/include/iso/iso646_iso.h target=../iso646.h
1972 $(sparc_ONLY)link path=usr/platform/SUNW,A70/include target=../sun4u/include

```

```
1973 $(sparc_ONLY)link path=usr/platform/SUNW,Netra-T12/include \
1974     target=../sun4u/include
1975 $(sparc_ONLY)link path=usr/platform/SUNW,Netra-T4/include \
1976     target=../sun4u/include
1977 $(sparc_ONLY)link path=usr/platform/SUNW,SPARC-Enterprise/include \
1978     target=../sun4u/include
1979 $(sparc_ONLY)link path=usr/platform/SUNW,Serverblade1/include \
1980     target=../sun4u/include
1981 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-100/include \
1982     target=../sun4u/include
1983 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-1000/include \
1984     target=../sun4u/include
1985 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-1500/include \
1986     target=../sun4u/include
1987 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Blade-2500/include \
1988     target=../sun4u/include
1989 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-15000/include \
1990     target=../sun4u/include
1991 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-280R/include \
1992     target=../sun4u/include
1993 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-480R/include \
1994     target=../sun4u/include
1995 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-880/include \
1996     target=../sun4u/include
1997 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V215/include \
1998     target=../sun4u/include
1999 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V240/include \
2000     target=../sun4u/include
2001 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V250/include \
2002     target=../sun4u/include
2003 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V440/include \
2004     target=../sun4u/include
2005 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V445/include \
2006     target=../sun4u/include
2007 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V490/include \
2008     target=../sun4u/include
2009 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire-V890/include \
2010     target=../sun4u/include
2011 $(sparc_ONLY)link path=usr/platform/SUNW,Sun-Fire/include \
2012     target=../sun4u/include
2013 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-2/include \
2014     target=../sun4u/include
2015 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-250/include \
2016     target=../sun4u/include
2017 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-4/include \
2018     target=../sun4u/include
2019 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-Enterprise-10000/include \
2020     target=../sun4u/include
2021 $(sparc_ONLY)link path=usr/platform/SUNW,Ultra-Enterprise/include \
2022     target=../sun4u/include
2023 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-40/include \
2024     target=../sun4u/include
2025 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-IIe-NetraCT-60/include \
2026     target=../sun4u/include
2027 $(sparc_ONLY)link path=usr/platform/SUNW,UltraSPARC-III-Netract/include \
2028     target=../sun4u/include
2029 link path=usr/share/man/man3head/acct.3head target=acct.h.3head
2030 link path=usr/share/man/man3head/aio.3head target=aio.h.3head
2031 link path=usr/share/man/man3head/ar.3head target=ar.h.3head
2032 link path=usr/share/man/man3head/archives.3head target=archives.h.3head
2033 link path=usr/share/man/man3head/assert.3head target=assert.h.3head
2034 link path=usr/share/man/man3head/complex.3head target=complex.h.3head
2035 link path=usr/share/man/man3head/cpio.3head target=cpio.h.3head
2036 link path=usr/share/man/man3head/dirent.3head target=dirent.h.3head
2037 link path=usr/share/man/man3head/errno.3head target=errno.h.3head
2038 link path=usr/share/man/man3head/fcntl.3head target=fcntl.h.3head
```

```
2039 link path=usr/share/man/man3head/fenv.3head target=fenv.h.3head
2040 link path=usr/share/man/man3head/float.3head target=float.h.3head
2041 link path=usr/share/man/man3head/floatpoint.3head \
2042     target=floatpoint.h.3head
2043 link path=usr/share/man/man3head/fmtmsg.3head target=fmtmsg.h.3head
2044 link path=usr/share/man/man3head/fnmatch.3head target=fnmatch.h.3head
2045 link path=usr/share/man/man3head/ftw.3head target=ftw.h.3head
2046 link path=usr/share/man/man3head/glob.3head target=glob.h.3head
2047 link path=usr/share/man/man3head/grp.3head target=grp.h.3head
2048 link path=usr/share/man/man3head/iconv.3head target=iconv.h.3head
2049 link path=usr/share/man/man3head/if.3head target=if.h.3head
2050 link path=usr/share/man/man3head/in.3head target=in.h.3head
2051 link path=usr/share/man/man3head/inet.3head target=inet.h.3head
2052 link path=usr/share/man/man3head/inttypes.3head target=inttypes.h.3head
2053 link path=usr/share/man/man3head/ipc.3head target=ipc.h.3head
2054 link path=usr/share/man/man3head/iso646.3head target=iso646.h.3head
2055 link path=usr/share/man/man3head/langinfo.3head target=langinfo.h.3head
2056 link path=usr/share/man/man3head/libgen.3head target=libgen.h.3head
2057 link path=usr/share/man/man3head/libintl.3head target=libintl.h.3head
2058 link path=usr/share/man/man3head/limits.3head target=limits.h.3head
2059 link path=usr/share/man/man3head/locale.3head target=locale.h.3head
2060 link path=usr/share/man/man3head/math.3head target=math.h.3head
2061 link path=usr/share/man/man3head/mman.3head target=mman.h.3head
2062 link path=usr/share/man/man3head/monetary.3head target=monetary.h.3head
2063 link path=usr/share/man/man3head/mqueue.3head target=mqueue.h.3head
2064 link path=usr/share/man/man3head/msg.3head target=msg.h.3head
2065 link path=usr/share/man/man3head/ndbm.3head target=ndbm.h.3head
2066 link path=usr/share/man/man3head/netdb.3head target=netdb.h.3head
2067 link path=usr/share/man/man3head/nl_types.3head target=nl_types.h.3head
2068 link path=usr/share/man/man3head/poll.3head target=poll.h.3head
2069 link path=usr/share/man/man3head/pthread.3head target=pthread.h.3head
2070 link path=usr/share/man/man3head/pwd.3head target=pwd.h.3head
2071 link path=usr/share/man/man3head/regex.3head target=regex.h.3head
2072 link path=usr/share/man/man3head/resource.3head target=resource.h.3head
2073 link path=usr/share/man/man3head/sched.3head target=sched.h.3head
2074 link path=usr/share/man/man3head/search.3head target=search.h.3head
2075 link path=usr/share/man/man3head/select.3head target=select.h.3head
2076 link path=usr/share/man/man3head/sem.3head target=sem.h.3head
2077 link path=usr/share/man/man3head/semaphore.3head target=semaphore.h.3head
2078 link path=usr/share/man/man3head/setjmp.3head target=setjmp.h.3head
2079 link path=usr/share/man/man3head/shm.3head target=shm.h.3head
2080 link path=usr/share/man/man3head/signinfo.3head target=signinfo.h.3head
2081 link path=usr/share/man/man3head/signal.3head target=signal.h.3head
2082 link path=usr/share/man/man3head/socket.3head target=socket.h.3head
2083 link path=usr/share/man/man3head/spawn.3head target=spawn.h.3head
2084 link path=usr/share/man/man3head/stat.3head target=stat.h.3head
2085 link path=usr/share/man/man3head/statvfs.3head target=statvfs.h.3head
2086 link path=usr/share/man/man3head/stdbool.3head target=stdbool.h.3head
2087 link path=usr/share/man/man3head/stddef.3head target=stddef.h.3head
2088 link path=usr/share/man/man3head/stdint.3head target=stdint.h.3head
2089 link path=usr/share/man/man3head/stdio.3head target=stdio.h.3head
2090 link path=usr/share/man/man3head/stdlib.3head target=stdlib.h.3head
2091 link path=usr/share/man/man3head/string.3head target=string.h.3head
2092 link path=usr/share/man/man3head/strings.3head target=strings.h.3head
2093 link path=usr/share/man/man3head/stropts.3head target=stropts.h.3head
2094 link path=usr/share/man/man3head/syslog.3head target=syslog.h.3head
2095 link path=usr/share/man/man3head/tar.3head target=tar.h.3head
2096 link path=usr/share/man/man3head/tcp.3head target=tcp.h.3head
2097 link path=usr/share/man/man3head/termios.3head target=termios.h.3head
2098 link path=usr/share/man/man3head/tgmth.3head target=tgmth.h.3head
2099 link path=usr/share/man/man3head/time.3head target=time.h.3head
2100 link path=usr/share/man/man3head/timelib.3head target=timelib.h.3head
2101 link path=usr/share/man/man3head/times.3head target=times.h.3head
2102 link path=usr/share/man/man3head/types.3head target=types.h.3head
2103 link path=usr/share/man/man3head/types32.3head target=types32.h.3head
2104 link path=usr/share/man/man3head/ucontext.3head target=ucontext.h.3head
```

```
2105 link path=usr/share/man/man3head/uio.3head target=uio.h.3head
2106 link path=usr/share/man/man3head/ulimit.3head target=ulimit.h.3head
2107 link path=usr/share/man/man3head/un.3head target=un.h.3head
2108 link path=usr/share/man/man3head/unistd.3head target=unistd.h.3head
2109 link path=usr/share/man/man3head/utime.3head target=utime.h.3head
2110 link path=usr/share/man/man3head/utmpx.3head target=utmpx.h.3head
2111 link path=usr/share/man/man3head/utsname.3head target=utsname.h.3head
2112 link path=usr/share/man/man3head/values.3head target=values.h.3head
2113 link path=usr/share/man/man3head/wait.3head target=wait.h.3head
2114 link path=usr/share/man/man3head/wchar.3head target=wchar.h.3head
2115 link path=usr/share/man/man3head/wctype.3head target=wctype.h.3head
2116 link path=usr/share/man/man3head/wordexp.3head target=wordexp.h.3head
2117 $(i386_ONLY)link path=usr/share/src/uts/i86pc/sys \
2118     target=../../../../platform/i86pc/include/sys
2119 $(i386_ONLY)link path=usr/share/src/uts/i86pc/vm \
2120     target=../../../../platform/i86pc/include/vm
2121 $(i386_ONLY)link path=usr/share/src/uts/i86xpv/sys \
2122     target=../../../../platform/i86xpv/include/sys
2123 $(i386_ONLY)link path=usr/share/src/uts/i86xpv/vm \
2124     target=../../../../platform/i86xpv/include/vm
2125 $(sparc_ONLY)link path=usr/share/src/uts/sun4u/sys \
2126     target=../../../../platform/sun4u/include/sys
2127 $(sparc_ONLY)link path=usr/share/src/uts/sun4u/vm \
2128     target=../../../../platform/sun4u/include/vm
2129 $(sparc_ONLY)link path=usr/share/src/uts/sun4v/sys \
2130     target=../../../../platform/sun4v/include/sys
2131 $(sparc_ONLY)link path=usr/share/src/uts/sun4v/vm \
2132     target=../../../../platform/sun4v/include/vm
```

new/usr/src/uts/common/sys/feature_tests.h

1

```
*****
15951 Tue May 20 07:58:54 2014
new/usr/src/uts/common/sys/feature_tests.h
2964 need POSIX 2008 locale object support
Reviewed by: Robert Mustacchi <rm@joyent.com>
*****

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 2006 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 _SYS_FEATURE_TESTS_H
31 #define _SYS_FEATURE_TESTS_H

33 #include <sys/ccompile.h>
34 #include <sys/isa_defs.h>

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

40 /*
41  * Values of _POSIX_C_SOURCE
42  *
43  *      undefined    not a POSIX compilation
44  *      1             POSIX.1-1990 compilation
45  *      2             POSIX.2-1992 compilation
46  *      199309L       POSIX.1b-1993 compilation (Real Time)
47  *      199506L       POSIX.1c-1995 compilation (POSIX Threads)
48  *      200112L       POSIX.1-2001 compilation (Austin Group Revision)
49  *      200809L       POSIX.1-2008 compilation
50 */
51 #if defined(_POSIX_SOURCE) && !defined(_POSIX_C_SOURCE)
52 #define _POSIX_C_SOURCE 1
53 #endif

55 /*
56  * The feature test macros __XOPEN_OR_POSIX, __STRICT_STDC, __STRICT_SYMBOLS,
57  * and __STDC_C99 are Sun implementation specific macros created in order to
58  * compress common standards specified feature test macros for easier reading.
59  * The feature test macros __XOPEN_OR_POSIX, __STRICT_STDC, and __STDC_C99
60  * are Sun implementation specific macros created in order to compress
```

new/usr/src/uts/common/sys/feature_tests.h

2

```
54 * common standards specified feature test macros for easier reading.
59 * These macros should not be used by the application developer as
60 * unexpected results may occur. Instead, the user should reference
61 * standards(5) for correct usage of the standards feature test macros.
62 *
63 * __XOPEN_OR_POSIX    Used in cases where a symbol is defined by both
64 *                     X/Open or POSIX or in the negative, when neither
65 *                     X/Open or POSIX defines a symbol.
66 *
67 * __STRICT_STDC       __STDC__ is specified by the C Standards and defined
68 *                     by the compiler. For Sun compilers the value of
69 *                     __STDC__ is either 1, 0, or not defined based on the
70 *                     compilation mode (see cc(1)). When the value of
71 *                     __STDC__ is 1 and in the absence of any other feature
72 *                     test macros, the namespace available to the application
73 *                     is limited to only those symbols defined by the C
74 *                     Standard. __STRICT_STDC provides a more readable means
75 *                     of identifying symbols defined by the standard, or in
76 *                     the negative, symbols that are extensions to the C
77 *                     Standard. See additional comments for GNU C differences.
78 *
79 * __STDC_C99          __STDC_VERSION__ is specified by the C standards and
80 *                     defined by the compiler and indicates the version of
81 *                     the C standard. A value of 199901L indicates a
82 *                     compiler that complies with ISO/IEC 9899:1999, other-
83 *                     wise known as the C99 standard.
84 *
85 * __STRICT_SYMBOLS    Used in cases where symbol visibility is restricted
86 *                     by the standards, and the user has not explicitly
87 *                     relaxed the strictness via __EXTENSIONS__.
88 */

90 #if defined(_XOPEN_SOURCE) || defined(_POSIX_C_SOURCE)
91 #define __XOPEN_OR_POSIX
92 #endif

94 /*
95  * ISO/IEC 9899:1990 and it's revision, ISO/IEC 9899:1999 specify the
96  * following predefined macro name:
97  *
98  * __STDC__           The integer constant 1, intended to indicate a conforming
99  *                     implementation.
100 *
101 * Furthermore, a strictly conforming program shall use only those features
102 * of the language and library specified in these standards. A conforming
103 * implementation shall accept any strictly conforming program.
104 *
105 * Based on these requirements, Sun's C compiler defines __STDC__ to 1 for
106 * strictly conforming environments and __STDC__ to 0 for environments that
107 * use ANSI C semantics but allow extensions to the C standard. For non-ANSI
108 * C semantics, Sun's C compiler does not define __STDC__.
109 *
110 * The GNU C project interpretation is that __STDC__ should always be defined
111 * to 1 for compilation modes that accept ANSI C syntax regardless of whether
112 * or not extensions to the C standard are used. Violations of conforming
113 * behavior are conditionally flagged as warnings via the use of the
114 * -pedantic option. In addition to defining __STDC__ to 1, the GNU C
115 * compiler also defines __STRICT_ANSI__ as a means of specifying strictly
116 * conforming environments using the -ansi or -std=<standard> options.
117 *
118 * In the absence of any other compiler options, Sun and GNU set the value
119 * of __STDC__ as follows when using the following options:
120 *
121 *                               Value of __STDC__  __STRICT_ANSI__
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 *
169 *     _LARGEFILE64_SOURCE
170 *         1          transitional large-file-related interfaces
171 *                   requested (seek64, stat64, etc.)
172 *
173 * The corresponding announcement macros are respectively:
174 *     _LFS_LARGEFILE
175 *     _LFS64_LARGEFILE
176 * (These are set in <unistd.h>.)
177 *
178 * Requesting _LARGEFILE64_SOURCE implies requesting _LARGEFILE_SOURCE as
179 * well.
180 *
181 * The large file interfaces are made visible regardless of the initial values
182 * of the feature test macros under certain circumstances:
183 * - If no explicit standards-conforming environment is requested (neither
184 *   of _POSIX_SOURCE nor _XOPEN_SOURCE is defined and the value of
185 *   __STDC__ does not imply standards conformance).
186 * - Extended system interfaces are explicitly requested (__EXTENSIONS__
187 *   is defined).
188 * - Access to in-kernel interfaces is requested (_KERNEL or _KMEMUSER is
189 *   defined). (Note that this dependency is an artifact of the current
190 *   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
257 * -----
258 * _XOPEN_SOURCE
259 * _XOPEN_SOURCE && _XOPEN_VERSION = 4
260 * _XOPEN_SOURCE && _XOPEN_SOURCE_EXTENDED = 1
261 * _XOPEN_SOURCE = 500
262 * _XOPEN_SOURCE = 600 (or POSIX_C_SOURCE=200112L)
263 * _XOPEN_SOURCE = 700 (or POSIX_C_SOURCE=200809L)
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

```

```

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 }
429 unchanged_portion_omitted
```