

```
*****
32718 Fri Sep 27 14:13:12 2013
new/usr/src/cmd/getconf/getconf.c
4146 getconf(1) does not support XATTR_ENABLED, XATTR_EXISTS, SATTR_ENABLED, SAT
Reviewed by: Cedric Blancher <cedric.blancher@gmail.com>
*****
unchanged_portion_omitted_
89 /*
90 * Some entries in this structure have both leading _ and non leading _
91 * versions for compatibility with various versions of the specifications.
92 * It is not expected that both forms will be required for new entries.
93 */
94
95 /* POSIX.2-1992 table 2-17 */
96 _POSIX2_BC_BASE_MAX, "POSIX2_BC_BASE_MAX", SELFCONF, NOFLAGS,
97 _POSIX2_BC_BASE_MAX, "_POSIX2_BC_BASE_MAX", SELFCONF, NOFLAGS,
98 _POSIX2_BC_DIM_MAX, "POSIX2_BC_DIM_MAX", SELFCONF, NOFLAGS,
99 _POSIX2_BC_DIM_MAX, "_POSIX2_BC_DIM_MAX", SELFCONF, NOFLAGS,
100 _POSIX2_BC_SCALE_MAX, "POSIX2_BC_SCALE_MAX", SELFCONF, NOFLAGS,
101 _POSIX2_BC_SCALE_MAX, "_POSIX2_BC_SCALE_MAX", SELFCONF, NOFLAGS,
102 _POSIX2_BC_STRING_MAX, "POSIX2_BC_STRING_MAX", SELFCONF, NOFLAGS,
103 _POSIX2_BC_STRING_MAX, "_POSIX2_BC_STRING_MAX", SELFCONF, NOFLAGS,
104 _POSIX2_COLL_WEIGHTS_MAX, "POSIX2_COLL_WEIGHTS_MAX", SELFCONF, NOFLAGS,
105 _POSIX2_COLL_WEIGHTS_MAX, "_POSIX2_COLL_WEIGHTS_MAX", SELFCONF, NOFLAGS,
106 _POSIX2_EXPR_NEST_MAX, "POSIX2_EXPR_NEST_MAX", SELFCONF, NOFLAGS,
107 _POSIX2_EXPR_NEST_MAX, "_POSIX2_EXPR_NEST_MAX", SELFCONF, NOFLAGS,
108 _POSIX2_LINE_MAX, "POSIX2_LINE_MAX", SELFCONF, NOFLAGS,
109 _POSIX2_LINE_MAX, "_POSIX2_LINE_MAX", SELFCONF, NOFLAGS,
110 _POSIX2_RE_DUP_MAX, "POSIX2_RE_DUP_MAX", SELFCONF, NOFLAGS,
111 _POSIX2_RE_DUP_MAX, "_POSIX2_RE_DUP_MAX", SELFCONF, NOFLAGS,
112 /* See limits.h section below for _POSIX_RE_DUP_MAX */
113
114 /* UNIX 98 */
115 _POSIX_AIO_LISTIO_MAX, "_POSIX_AIO_LISTIO_MAX", SELFCONF, NOFLAGS,
116 _POSIX_AIO_MAX, "_POSIX_AIO_MAX", SELFCONF, NOFLAGS,
117 _POSIX_DELAYTIMER_MAX, "_POSIX_DELAYTIMER_MAX", SELFCONF, NOFLAGS,
118 _POSIX_LOGIN_NAME_MAX, "_POSIX_LOGIN_NAME_MAX", SELFCONF, NOFLAGS,
119 _POSIX_MQ_OPEN_MAX, "_POSIX_MQ_OPEN_MAX", SELFCONF, NOFLAGS,
120 _POSIX_MQ_PRIO_MAX, "_POSIX_MQ_PRIO_MAX", SELFCONF, NOFLAGS,
121 _POSIX_RTSIG_MAX, "_POSIX_RTSIG_MAX", SELFCONF, NOFLAGS,
122 _POSIX_SEM_NSEMS_MAX, "_POSIX_SEM_NSEMS_MAX", SELFCONF, NOFLAGS,
123 _POSIX_SEM_VALUE_MAX, "_POSIX_SEM_VALUE_MAX", SELFCONF, NOFLAGS,
124 _POSIX_SIGQUEUE_MAX, "_POSIX_SIGQUEUE_MAX", SELFCONF, NOFLAGS,
125 _POSIX_TTY_NAME_MAX, "_POSIX_TTY_NAME_MAX", SELFCONF, NOFLAGS,
126
127 /* UNIX 03 */
128 _POSIX_HOST_NAME_MAX, "_POSIX_HOST_NAME_MAX", SELFCONF, NOFLAGS,
129 _POSIX_RE_DUP_MAX, "_POSIX_RE_DUP_MAX", SELFCONF, NOFLAGS,
130 _POSIX_CLOCKRES_MIN, "_POSIX_CLOCKRES_MIN", SELFCONF, NOFLAGS,
131 _POSIX_SYMLINK_MAX, "_POSIX_SYMLINK_MAX", SELFCONF, NOFLAGS,
132 _POSIX_SYMLOOP_MAX, "_POSIX_SYMLOOP_MAX", SELFCONF, NOFLAGS,
133 CHARCLASS_NAME_MAX, "_POSIX2_CHARCLASS_NAME_MAX", SELFCONF, NOFLAGS,
134
135 /* POSIX.2-1992 table 2-17 */
136 _SC_2_VERSION, "POSIX2_VERSION", SYSCONF, NOFLAGS,
137 _SC_2_VERSION, "_POSIX2_VERSION", SYSCONF, NOFLAGS,
138
139 /* POSIX.2-1992 table 2-18 */
140 _SC_BC_BASE_MAX, "BC_BASE_MAX", SYSCONF, NOFLAGS,
141 _SC_BC_DIM_MAX, "BC_DIM_MAX", SYSCONF, NOFLAGS,
142 _SC_BC_SCALE_MAX, "BC_SCALE_MAX", SYSCONF, NOFLAGS,
143 _SC_BC_STRING_MAX, "BC_STRING_MAX", SYSCONF, NOFLAGS,
144 _SC_COLL_WEIGHTS_MAX, "COLL_WEIGHTS_MAX", SYSCONF, NOFLAGS,
145 _SC_EXPR_NEST_MAX, "EXPR_NEST_MAX", SYSCONF, NOFLAGS,
146 _SC_LINE_MAX, "LINE_MAX", SYSCONF, NOFLAGS,
147 _SC_RE_DUP_MAX, "RE_DUP_MAX", SYSCONF, NOFLAGS,
```

```
149 /* POSIX.2-1992 table 2-19 */
150 _SC_2_C_BIND, "POSIX2_C_BIND", SYSCONF, NOFLAGS,
151 _SC_2_C_BIND, "_POSIX2_C_BIND", SYSCONF, NOFLAGS,
152 _SC_2_C_DEV, "POSIX2_C_DEV", SYSCONF, NOFLAGS,
153 _SC_2_C_DEV, "_POSIX2_C_DEV", SYSCONF, NOFLAGS,
154 _SC_2_CHAR_TERM, "POSIX2_CHAR_TERM", SYSCONF, NOFLAGS,
155 _SC_2_CHAR_TERM, "_POSIX2_CHAR_TERM", SYSCONF, NOFLAGS,
156 _SC_2_FORT_DEV, "POSIX2_FORT_DEV", SYSCONF, NOFLAGS,
157 _SC_2_FORT_DEV, "_POSIX2_FORT_DEV", SYSCONF, NOFLAGS,
158 _SC_2_FORT_RUN, "POSIX2_FORT_RUN", SYSCONF, NOFLAGS,
159 _SC_2_FORT_RUN, "_POSIX2_FORT_RUN", SYSCONF, NOFLAGS,
160 _SC_2_LOCALEDEF, "POSIX2_LOCALEDEF", SYSCONF, NOFLAGS,
161 _SC_2_LOCALEDEF, "_POSIX2_LOCALEDEF", SYSCONF, NOFLAGS,
162 _SC_2_SW_DEV, "POSIX2_SW_DEV", SYSCONF, NOFLAGS,
163 _SC_2_SW_DEV, "_POSIX2_SW_DEV", SYSCONF, NOFLAGS,
164 _SC_2_UPE, "POSIX2_UPE", SYSCONF, NOFLAGS,
165 _SC_2_UPE, "_POSIX2_UPE", SYSCONF, NOFLAGS,
166
167 /* POSIX.1-1990 table 2-3 */
168 _POSIX_ARG_MAX, "_POSIX_ARG_MAX", SELFCONF, NOFLAGS,
169 _POSIX_CHILD_MAX, "_POSIX_CHILD_MAX", SELFCONF, NOFLAGS,
170 _POSIX_LINK_MAX, "_POSIX_LINK_MAX", SELFCONF, NOFLAGS,
171 _POSIX_MAX_CANON, "_POSIX_MAX_CANON", SELFCONF, NOFLAGS,
172 _POSIX_MAX_INPUT, "_POSIX_MAX_INPUT", SELFCONF, NOFLAGS,
173 _POSIX_NAME_MAX, "_POSIX_NAME_MAX", SELFCONF, NOFLAGS,
174 _POSIX_NGROUPS_MAX, "_POSIX_NGROUPS_MAX", SELFCONF, NOFLAGS,
175 _POSIX_OPEN_MAX, "_POSIX_OPEN_MAX", SELFCONF, NOFLAGS,
176 _POSIX_PATH_MAX, "_POSIX_PATH_MAX", SELFCONF, NOFLAGS,
177 _POSIX_PIPE_BUF, "_POSIX_PIPE_BUF", SELFCONF, NOFLAGS,
178 _POSIX_SSIZE_MAX, "_POSIX_SSIZE_MAX", SELFCONF, NOFLAGS,
179 _POSIX_STREAM_MAX, "_POSIX_STREAM_MAX", SELFCONF, NOFLAGS,
180 _POSIX_TIMER_MAX, "_POSIX_TIMER_MAX", SELFCONF, NOFLAGS,
181 _POSIX_THREAD_DESTRUCTOR_ITERATIONS, "_POSIX_THREAD_DESTRUCTOR_ITERATIONS", SELFCONF, NOFLAGS,
182 _POSIX_THREAD_KEYS_MAX, "_POSIX_THREAD_KEYS_MAX", SELFCONF, NOFLAGS,
183 _POSIX_THREAD_THREADS_MAX, "_POSIX_THREAD_THREADS_MAX", SELFCONF, NOFLAGS,
184 _POSIX_TZNAME_MAX, "_POSIX_TZNAME_MAX", SELFCONF, NOFLAGS,
185
186 /* POSIX.1-1990 table 4-2 */
187 _SC_ARG_MAX, "ARG_MAX", SYSCONF, NOFLAGS,
188 _SC_CHILD_MAX, "CHILD_MAX", SYSCONF, NOFLAGS,
189 _SC_CLK_TCK, "CLK_TCK", SYSCONF, NOFLAGS,
190 _SC_JOB_CONTROL, "_POSIX_JOB_CONTROL", SYSCONF, NOFLAGS,
191 _SC_NGROUPS_MAX, "NGROUPS_MAX", SYSCONF, NOFLAGS,
192 _SC_OPEN_MAX, "OPEN_MAX", SYSCONF, NOFLAGS,
193 _SC_SAVED_IDS, "_POSIX_SAVED_IDS", SYSCONF, NOFLAGS,
194 _SC_STREAM_MAX, "STREAM_MAX", SYSCONF, NOFLAGS,
195 _SC_TZNAME_MAX, "TZNAME_MAX", SYSCONF, NOFLAGS,
196 _SC_VERSION, "_POSIX_VERSION", SYSCONF, NOFLAGS,
197
198 /* pathconf() variables */
199
200 /* pathconf() variables */
201
202 /* POSIX.1-1990 table 5-2 */
203
204
205 _PC_LINK_MAX, "LINK_MAX", PATHCONF, NOFLAGS,
206 _PC_MAX_CANON, "MAX_CANON", PATHCONF, NOFLAGS,
207 _PC_MAX_INPUT, "MAX_INPUT", PATHCONF, NOFLAGS,
208 _PC_NAME_MAX, "NAME_MAX", PATHCONF, NOFLAGS,
209 _PC_PATH_MAX, "PATH_MAX", PATHCONF, NOFLAGS,
210 _PC_PIPE_BUF, "PIPE_BUF", PATHCONF, NOFLAGS,
211 _PC_CHOWN_RESTRICTED, "_POSIX_CHOWN_RESTRICTED", PATHCONF, NOFLAGS,
212 _PC_NO_TRUNC, "_POSIX_NO_TRUNC", PATHCONF, NOFLAGS,
213 _PC_VDISABLE, "_POSIX_VDISABLE", PATHCONF, NOFLAGS,
```

```

215      /* UNIX 98 additions */
217      _PC_ASYNC_IO,          "_POSIX_ASYNC_IO",      PATHCONF, NOFLAGS,
218      _PC_PRIO_IO,           "_POSIX_PRIO_IO",      PATHCONF, NOFLAGS,
219      _PC_SYNC_IO,           "_POSIX_SYNC_IO",      PATHCONF, NOFLAGS,
221
222      /* Large File Summit name (UNIX 98) */
223
223      _PC_FILESIZEBITS,      "FILESIZEBITS",        PATHCONF, NOFLAGS,
224      _PC_XATTR_ENABLED,     "XATTR_ENABLED",       PATHCONF, NOFLAGS,
225      _PC_XATTR_EXISTS,      "XATTR_EXISTS",        PATHCONF, NOFLAGS,
226 #endif /* ! codereview */
228
228      /* UNIX 03 Additions */
229
230      /* Advisory Information (ADV) */
231      _PC_ALLOC_SIZE_MIN,    "POSIX_ALLOC_SIZE_MIN", PATHCONF, NOFLAGS,
232      _PC_REC_INCR_XFER_SIZE, "POSIX_REC_INCR_XFER_SIZE", PATHCONF, NOFLAGS,
233      _PC_REC_MAX_XFER_SIZE, "POSIX_REC_MAX_XFER_SIZE", PATHCONF, NOFLAGS,
234      _PC_REC_MIN_XFER_SIZE, "POSIX_REC_MIN_XFER_SIZE", PATHCONF, NOFLAGS,
235      _PC_REC_XFER_ALIGN,    "POSIX_REC_XFER_ALIGN",   PATHCONF, NOFLAGS,
236
237      _PC_2_SYMLINKS,        "POSIX2_SYMLINKS",      PATHCONF, NOFLAGS,
238      _PC_SYMLINK_MAX,       "SYMLINK_MAX",        PATHCONF, NOFLAGS,
239
240      _PC_ACL_ENABLED,       "ACL_ENABLED",         PATHCONF, NOFLAGS,
241      _PC_MIN_HOLE_SIZE,     "MIN_HOLE_SIZE",       PATHCONF, NOFLAGS,
242      _PC_SATTR_ENABLED,     "SATTR_ENABLED",       PATHCONF, NOFLAGS,
243      _PC_SATTR_EXISTS,      "SATTR_EXISTS",        PATHCONF, NOFLAGS,
244
245 #endif /* ! codereview */
246
246      /* UNIX 08 additions */
247
248      _PC_TIMESTAMP_RESOLUTION, "_POSIX_TIMESTAMP_RESOLUTION",
249      PATHCONF, NOFLAGS,
250
251      /* confstr() variables */
252
253      /* POSIX.2-1992 description of getconf utility */
254      _CS_PATH,               "PATH",                CONFSTR, NOFLAGS,
255
256      /* POSIX.2-1992 table B-18 */
257      _CS_PATH,               "CS_PATH",              CONFSTR, NOFLAGS,
258      _CS_PATH,               "_CS_PATH",             CONFSTR, NOFLAGS,
259
260      /* command names for large file configuration information */
261      /* large file compilation environment configuration */
262
263      _CS_LFS_CFLAGS,          "LFS_CFLAGS",          CONFSTR, NOFLAGS,
264      _CS_LFS_LDFLAGS,         "LFS_LDFLAGS",         CONFSTR, NOFLAGS,
265      _CS_LFS_LIBS,            "LFS_LIBS",             CONFSTR, NOFLAGS,
266      _CS_LFS_LINTFLAGS,       "LFS_LINTFLAGS",        CONFSTR, NOFLAGS,
267
268      /* transitional large file interface configuration */
269
270      _CS_LFS64_CFLAGS,        "LFS64_CFLAGS",        CONFSTR, NOFLAGS,
271      _CS_LFS64_LDFLAGS,       "LFS64_LDFLAGS",       CONFSTR, NOFLAGS,
272      _CS_LFS64_LIBS,          "LFS64_LIBS",           CONFSTR, NOFLAGS,
273      _CS_LFS64_LINTFLAGS,     "LFS64_LINTFLAGS",      CONFSTR, NOFLAGS,
274
275      /*
276      *  UNIX 98:
277      *  confstr additions, these support
278      *  programming specifications/environments.
279

```

```

280      * Unfortunately there is some uncertainty with
281      * the standard at this point about the constant
282      * name, thus we need to recognize BOTH with and
283      * without the initial underscore.
284      */
285      _CS_XBS5_ILP32_OFF32_CFLAGS,    "XBS5_ILP32_OFF32_CFLAGS",
286      CONFSTR, NOFLAGS,
287      _CS_XBS5_ILP32_OFF32_CFLAGS,    "_XBS5_ILP32_OFF32_CFLAGS",
288      CONFSTR, NOFLAGS,
289
290      _CS_XBS5_ILP32_OFF32_LDFLAGS,   "XBS5_ILP32_OFF32_LDFLAGS",
291      CONFSTR, NOFLAGS,
292      _CS_XBS5_ILP32_OFF32_LDFLAGS,   "_XBS5_ILP32_OFF32_LDFLAGS",
293      CONFSTR, NOFLAGS,
294
295      _CS_XBS5_ILP32_OFF32_LIBS,     "XBS5_ILP32_OFF32_LIBS",
296      CONFSTR, NOFLAGS,
297      _CS_XBS5_ILP32_OFF32_LIBS,     "_XBS5_ILP32_OFF32_LIBS",
298      CONFSTR, NOFLAGS,
299
300      _CS_XBS5_ILP32_OFF32_LINTFLAGS, "XBS5_ILP32_OFF32_LINTFLAGS",
301      CONFSTR, NOFLAGS,
302      _CS_XBS5_ILP32_OFF32_LINTFLAGS, "_XBS5_ILP32_OFF32_LINTFLAGS",
303      CONFSTR, NOFLAGS,
304
305      _CS_XBS5_ILP32_OFBIG_CFLAGS,   "XBS5_ILP32_OFBIG_CFLAGS",
306      CONFSTR, NOFLAGS,
307      _CS_XBS5_ILP32_OFBIG_CFLAGS,   "_XBS5_ILP32_OFBIG_CFLAGS",
308      CONFSTR, NOFLAGS,
309
310      _CS_XBS5_ILP32_OFBIG_LDFLAGS,  "XBS5_ILP32_OFBIG_LDFLAGS",
311      CONFSTR, NOFLAGS,
312      _CS_XBS5_ILP32_OFBIG_LDFLAGS,  "_XBS5_ILP32_OFBIG_LDFLAGS",
313      CONFSTR, NOFLAGS,
314
315      _CS_XBS5_ILP32_OFBIG_LIBS,    "XBS5_ILP32_OFBIG_LIBS",
316      CONFSTR, NOFLAGS,
317      _CS_XBS5_ILP32_OFBIG_LIBS,    "_XBS5_ILP32_OFBIG_LIBS",
318      CONFSTR, NOFLAGS,
319
320      _CS_XBS5_ILP32_OFBIG_LINTFLAGS, "XBS5_ILP32_OFBIG_LINTFLAGS",
321      CONFSTR, NOFLAGS,
322      _CS_XBS5_ILP32_OFBIG_LINTFLAGS, "_XBS5_ILP32_OFBIG_LINTFLAGS",
323      CONFSTR, NOFLAGS,
324
325      _CS_XBS5_LP64_OFF64_CFLAGS,   "XBS5_LP64_OFF64_CFLAGS",
326      CONFSTR, NOFLAGS,
327      _CS_XBS5_LP64_OFF64_CFLAGS,   "_XBS5_LP64_OFF64_CFLAGS",
328      CONFSTR, NOFLAGS,
329
330      _CS_XBS5_LP64_OFF64_LDFLAGS,  "XBS5_LP64_OFF64_LDFLAGS",
331      CONFSTR, NOFLAGS,
332      _CS_XBS5_LP64_OFF64_LDFLAGS,  "_XBS5_LP64_OFF64_LDFLAGS",
333      CONFSTR, NOFLAGS,
334
335      _CS_XBS5_LP64_OFF64_LIBS,    "XBS5_LP64_OFF64_LIBS",
336      CONFSTR, NOFLAGS,
337      _CS_XBS5_LP64_OFF64_LIBS,    "_XBS5_LP64_OFF64_LIBS",
338      CONFSTR, NOFLAGS,
339
340      _CS_XBS5_LP64_OFF64_LINTFLAGS, "XBS5_LP64_OFF64_LINTFLAGS",
341      CONFSTR, NOFLAGS,
342      _CS_XBS5_LP64_OFF64_LINTFLAGS, "_XBS5_LP64_OFF64_LINTFLAGS",
343      CONFSTR, NOFLAGS,
344
345      _CS_XBS5_LPBIG_OFBIG_CFLAGS,  "XBS5_LPBIG_OFBIG_CFLAGS",

```

```

346     CONFSTR,           NOFLAGS,
347     _CS_XBS5_LPBIG_OFFBIG_CFLAGS,   "_XBS5_LPBIG_OFFBIG_CFLAGS",
348     CONFSTR,           NOFLAGS,
349
350     _CS_XBS5_LPBIG_OFFBIG_LDFLAGS, "XBS5_LPBIG_OFFBIG_LDFLAGS",
351     CONFSTR,           NOFLAGS,
352     _CS_XBS5_LPBIG_OFFBIG_LDFLAGS, "_XBS5_LPBIG_OFFBIG_LDFLAGS",
353     CONFSTR,           NOFLAGS,
354
355     _CS_XBS5_LPBIG_OFFBIG_LIBS,   "XBS5_LPBIG_OFFBIG_LIBS",
356     CONFSTR,           NOFLAGS,
357     _CS_XBS5_LPBIG_OFFBIG_LIBS,   "_XBS5_LPBIG_OFFBIG_LIBS",
358     CONFSTR,           NOFLAGS,
359
360     _CS_XBS5_LPBIG_OFFBIG_LINTFLAGS, "XBS5_LPBIG_OFFBIG_LINTFLAGS",
361     CONFSTR,           NOFLAGS,
362     _CS_XBS5_LPBIG_OFFBIG_LINTFLAGS, "_XBS5_LPBIG_OFFBIG_LINTFLAGS",
363     CONFSTR,           NOFLAGS,
364
365     /*
366      * UNIX 03 confstr() additions for compilation environments
367      */
368     _CS_POSIX_V6_ILP32_OFF32_CFLAGS, "POSIX_V6_ILP32_OFF32_CFLAGS",
369     CONFSTR,           NOFLAGS,
370     _CS_POSIX_V6_ILP32_OFF32_LDFLAGS, "POSIX_V6_ILP32_OFF32_LDFLAGS",
371     CONFSTR,           NOFLAGS,
372     _CS_POSIX_V6_ILP32_OFF32_LIBS,  "POSIX_V6_ILP32_OFF32_LIBS",
373     CONFSTR,           NOFLAGS,
374     _CS_POSIX_V6_ILP32_OFFBIG_CFLAGS, "POSIX_V6_ILP32_OFFBIG_CFLAGS",
375     CONFSTR,           NOFLAGS,
376     _CS_POSIX_V6_ILP32_OFFBIG_LDFLAGS, "POSIX_V6_ILP32_OFFBIG_LDFLAGS",
377     CONFSTR,           NOFLAGS,
378     _CS_POSIX_V6_ILP32_OFFBIG_LIBS,  "POSIX_V6_ILP32_OFFBIG_LIBS",
379     CONFSTR,           NOFLAGS,
380     _CS_POSIX_V6_LP64_OFF64_CFLAGS,  "POSIX_V6_LP64_OFF64_CFLAGS",
381     CONFSTR,           NOFLAGS,
382     _CS_POSIX_V6_LP64_OFF64_LDFLAGS, "POSIX_V6_LP64_OFF64_LDFLAGS",
383     CONFSTR,           NOFLAGS,
384     _CS_POSIX_V6_LP64_OFF64_LIBS,   "POSIX_V6_LP64_OFF64_LIBS",
385     CONFSTR,           NOFLAGS,
386     _CS_POSIX_V6_LPBIG_OFFBIG_CFLAGS, "POSIX_V6_LPBIG_OFFBIG_CFLAGS",
387     CONFSTR,           NOFLAGS,
388     _CS_POSIX_V6_LPBIG_OFFBIG_LDFLAGS, "POSIX_V6_LPBIG_OFFBIG_LDFLAGS",
389     CONFSTR,           NOFLAGS,
390     _CS_POSIX_V6_LPBIG_OFFBIG_LIBS,  "POSIX_V6_LPBIG_OFFBIG_LIBS",
391     CONFSTR,           NOFLAGS,
392     _CS_POSIX_V6_WIDTH_RESTRICTED_ENVS, "POSIX_V6_WIDTH_RESTRICTED_ENVS",
393     CONFSTR,           NOFLAGS,
394     /* End of UNIX 03 compilation environments */
395
396 #ifdef _SC_2_C_VERSION          /* POSIX.2 table B-4 */
397     _SC_2_C_VERSION,        "_POSIX2_C_VERSION",    SYSCONF,    NOFLAGS,
398 #else
399     _POSIX2_C_VERSION,      "_POSIX2_C_VERSION",    SELFCONF,   NOFLAGS,
400 #endif
401
402 /*
403  * XPG4 support BEGINS
404 */
405
406 #ifndef CHARCLASS_NAME_MAX
407 #define CHARCLASS_NAME_MAX      -1
408 #endif
409 #ifndef CHAR_BIT
410 #define CHAR_BIT                -1
411 #endif

```

```

412 #ifndef CHAR_MAX
413 #define CHAR_MAX             -1
414 #endif
415 #ifndef CHAR_MIN
416 #define CHAR_MIN             -1
417 #endif
418 #ifndef INT_MAX
419 #define INT_MAX              -1
420 #endif
421 #ifndef INT_MIN
422 #define INT_MIN              -1
423 #endif
424 #ifndef LONG_BIT
425 #define LONG_BIT              -1
426 #endif
427 #ifndef LONG_MAX
428 #define LONG_MAX              -1
429 #endif
430 #ifndef LONG_MIN
431 #define LONG_MIN              -1
432 #endif
433 #ifndef MB_LEN_MAX
434 #define MB_LEN_MAX             -1
435 #endif
436 #ifndef NL_NMAX
437 #define NL_NMAX               -1
438 #endif
439 #ifndef NL_ARGMAX
440 #define NL_ARGMAX              -1
441 #endif
442 #ifndef NL_LANGMAX
443 #define NL_LANGMAX             -1
444 #endif
445 #ifndef NL_MSGMAX
446 #define NL_MSGMAX              -1
447 #endif
448 #ifndef NL_SETMAX
449 #define NL_SETMAX              -1
450 #endif
451 #ifndef NL_TEXTMAX
452 #define NL_TEXTMAX              -1
453 #endif
454 #ifndef NZERO
455 #define NZERO                 -1
456 #endif
457 #ifndef SCHAR_MAX
458 #define SCHAR_MAX              -1
459 #endif
460 #ifndef SCHAR_MIN
461 #define SCHAR_MIN              -1
462 #endif
463 #ifndef SHRT_MAX
464 #define SHRT_MAX               -1
465 #endif
466 #ifndef SHRT_MIN
467 #define SHRT_MIN              -1
468 #endif
469 #ifndef TMP_MAX
470 #define TMP_MAX                -1
471 #endif
472 #ifndef WORD_BIT
473 #define WORD_BIT               -1
474 #endif
475
476 #ifndef _XOPEN_XPG2
477 #endif

```

```

478 #define _XOPEN_XPG2 -1
479 #endif
480 #ifndef _XOPEN_XPG3
481 #define _XOPEN_XPG3 -1
482 #endif
483 #ifndef _XOPEN_XPG4
484 #define _XOPEN_XPG4 -1
485 #endif

487 /*
488 * the following are values that we should find in <limits.h>
489 * so we use SELFCONF here.
490 */
491 */

493 /* POSIX.1-1990 */
494 CHARCLASS_NAME_MAX, "CHARCLASS_NAME_MAX", SELFCONF, NOFLAGS,
495 SSIZE_MAX, "SSIZE_MAX", SELFCONF, NOFLAGS,
496

497 /* POSIX.1-1988 */
498 CHAR_BIT, "CHAR_BIT", SELFCONF, NOFLAGS,
499 CHAR_MAX, "CHAR_MAX", SELFCONF, NOFLAGS,
500 CHAR_MIN, "CHAR_MIN", SELFCONF, NOFLAGS,
501 INT_MAX, "INT_MAX", SELFCONF, NOFLAGS,
502 INT_MIN, "INT_MIN", SELFCONF, NOFLAGS,
503 LONG_BIT, "LONG_BIT", SELFCONF, NOFLAGS,
504 LONG_MAX, "LONG_MAX", SELFCONF, NOFLAGS,
505 LONG_MIN, "LONG_MIN", SELFCONF, NOFLAGS,
506 MB_LEN_MAX, "MB_LEN_MAX", SELFCONF, NOFLAGS,
507 NL_NMAX, "NL_NMAX", SELFCONF, NOFLAGS,
508 NL_ARGMAX, "NL_ARGMAX", SELFCONF, NOFLAGS,
509 NL_LANGMAX, "NL_LANGMAX", SELFCONF, NOFLAGS,
510 NL_MSGMAX, "NL_MSGMAX", SELFCONF, NOFLAGS,
511 NL_SETMAX, "NL_SETMAX", SELFCONF, NOFLAGS,
512 NL_TEXTMAX, "NL_TEXTMAX", SELFCONF, NOFLAGS,
513 NZERO, "NZERO", SELFCONF, NOFLAGS,
514 SCHAR_MAX, "SCHAR_MAX", SELFCONF, NOFLAGS,
515 SCHAR_MIN, "SCHAR_MIN", SELFCONF, NOFLAGS,
516 SHRT_MAX, "SHRT_MAX", SELFCONF, NOFLAGS,
517 SHRT_MIN, "SHRT_MIN", SELFCONF, NOFLAGS,
518 TMP_MAX, "TMP_MAX", SELFCONF, NOFLAGS,
519

520 /*
521 * for the unsigned maximums, we cannot rely on the value -1
522 * to indicate "undefined".
523 */
524 #ifndef UCHAR_MAX
525 0, "UCHAR_MAX", SELFCONF, UNDEFINED,
526 #else
527 (long)UCHAR_MAX, "UCHAR_MAX", SELFCONF, UNSIGNED_VALUE,
528 #endif /* UCHAR_MAX */
529 #ifndef UINT_MAX
530 0, "UINT_MAX", SELFCONF, UNDEFINED,
531 #else
532 (long)UINT_MAX, "UINT_MAX", SELFCONF, UNSIGNED_VALUE,
533 #endif /* UINT_MAX */
534 #ifndef ULONG_MAX
535 0, "ULONG_MAX", SELFCONF, UNDEFINED,
536 #else
537 (long)ULONG_MAX, "ULONG_MAX", SELFCONF, UNSIGNED_VALUE,
538 #endif /* ULONG_MAX */
539 #ifndef USHRT_MAX
540 0, "USHRT_MAX", SELFCONF, UNDEFINED,
541 #else
542 (long)USHRT_MAX, "USHRT_MAX", SELFCONF, UNSIGNED_VALUE,
543 #endif /* USHRT_MAX */

```

```

544 #ifndef SIZE_MAX
545 0, "SIZE_MAX", SELFCONF, UNDEFINED,
546 #else
547 (long)SIZE_MAX, "SIZE_MAX", SELFCONF, UNSIGNED_VALUE,
548 #endif /* SIZE_MAX */

550 WORD_BIT, "WORD_BIT", SELFCONF, NOFLAGS,
552 /* XPG4 Version 2 */
553 _XOPEN_IOV_MAX, "_XOPEN_IOV_MAX", SELFCONF, NOFLAGS,
555 /* UNIX 03 */
556 _XOPEN_NAME_MAX, "_XOPEN_NAME_MAX", SELFCONF, NOFLAGS,
557 _XOPEN_PATH_MAX, "_XOPEN_PATH_MAX", SELFCONF, NOFLAGS,
559 /*
560 * The following are defined via sysconf(). These are considered
561 * an extension to sysconf().
562 * The -0 is needed if _XOPEN_XPG? is defined to be an empty string.
563 */
564 _XOPEN_XPG2-0, "_XOPEN_XPG2", SELFCONF, NOFLAGS,
565 _XOPEN_XPG3-0, "_XOPEN_XPG3", SELFCONF, NOFLAGS,
566 _XOPEN_XPG4-0, "_XOPEN_XPG4", SELFCONF, NOFLAGS,
568 /*
569 * The following should be provided by sysconf() (e.g use SYSCONF),
570 * so we look for the appropriate _SC_* symbol in <unistd.h>.
571 * If it is not defined, then we use SELFCONF with the value of -1.
572 */
573 _SC_XOPEN_VERSION, "_XOPEN_VERSION", SYSCONF, NOFLAGS,
574 _SC_XOPEN_XCU_VERSION, "_XOPEN_XCU_VERSION", SYSCONF, NOFLAGS,
575 _SC_XOPEN_CRYPT, "_XOPEN_CRYPT", SYSCONF, NOFLAGS,
576 _SC_XOPEN_ENH_I18N, "_XOPEN_ENH_I18N", SYSCONF, NOFLAGS,
577 _SC_XOPEN_SHM, "_XOPEN_SHM", SYSCONF, NOFLAGS,
579 /*
580 * XPG4 support ends
581 */
583 /* XPG4, Version 2 support */
585 _SC_ATEXIT_MAX, "ATEXIT_MAX", SYSCONF, NOFLAGS,
586 _SC_PAGESIZE, "PAGESIZE", SYSCONF, NOFLAGS,
587 _SC_PAGE_SIZE, "PAGE_SIZE", SYSCONF, NOFLAGS,
588 _SC_XOPEN_UNIX, "_XOPEN_UNIX", SYSCONF, NOFLAGS,
589 _SC_IOV_MAX, "IOV_MAX", SYSCONF, NOFLAGS,
591 /*
592 * UNIX 98 - XPG5 additions
593 */
594 _SC_XOPEN_LEGACY, "_XOPEN_LEGACY", SYSCONF, NOFLAGS,
595 _SC_PASS_MAX, "PASS_MAX", SYSCONF, NOFLAGS,
596 _SC_AIO_LISTIO_MAX, "AIO_LISTIO_MAX", SYSCONF, NOFLAGS,
597 _SC_AIO_MAX, "AIO_MAX", SYSCONF, NOFLAGS,
598 _SC_AIO_PRIO_DELTA_MAX, "AIO_PRIO_DELTA_MAX", SYSCONF, NOFLAGS,
599 _SC_DELAYTIMER_MAX, "DELAYTIMER_MAX", SYSCONF, NOFLAGS,
600 _SC_MQ_OPEN_MAX, "MQ_OPEN_MAX", SYSCONF, NOFLAGS,
601 _SC_MQ_PRIO_MAX, "MQ_PRIO_MAX", SYSCONF, NOFLAGS,
602 _SC_RTSIG_MAX, "RTSIG_MAX", SYSCONF, NOFLAGS,
603 _SC_SEM_NSEMS_MAX, "SEM_NSEMS_MAX", SYSCONF, NOFLAGS,
604 _SC_SEM_VALUE_MAX, "SEM_VALUE_MAX", SYSCONF, NOFLAGS,
605 _SC_SIGQUEUE_MAX, "SIGQUEUE_MAX", SYSCONF, NOFLAGS,
606 _SC_TIMER_MAX, "TIMER_MAX", SYSCONF, NOFLAGS,
608 _SC_ASYNCHRONOUS_IO, "_POSIX_ASYNCHRONOUS_IO", SYSCONF, NOFLAGS,
609 _SC_FSYNC, "_POSIX_FSYNC", SYSCONF, NOFLAGS,

```

```

610     _SC_MAPPED_FILES,      "_POSIX_MAPPED_FILES",    SYSCONF,      NOFLAGS,
611     _SC_MEMLOCK,          "_POSIX_MEMLOCK",        SYSCONF,      NOFLAGS,
612     _SC_MEMLOCK_RANGE,   "_POSIX_MEMLOCK_RANGE",  SYSCONF,      NOFLAGS,
613     _SC_MEMORY_PROTECTION, "_POSIX_MEMORY_PROTECTION", SYSCONF,      NOFLAGS,
614     _SC_MESSAGE_PASSING,  "_POSIX_MESSAGE_PASSING", SYSCONF,      NOFLAGS,
615     _SC_PRIORITIZED_IO,   "_POSIX_PRIORITIZED_IO",  SYSCONF,      NOFLAGS,
616     _SC_PRIORITY_SCHEDULING, "_POSIX_PRIORITY_SCHEDULING", SYSCONF,      NOFLAGS,
617     _SC_REALTIME_SIGNALS,  "_POSIX_REALTIME_SIGNALS", SYSCONF,      NOFLAGS,
618     _SC_SEMAPHORES,        "_POSIX_SEMAPHORES",       SYSCONF,      NOFLAGS,
619
620     _SC_SHARED_MEMORY_OBJECTS, "_POSIX_SHARED_MEMORY_OBJECTS", SYSCONF,      NOFLAGS,
621
622     _SC_SYNCHRONIZED_IO,   "_POSIX_SYNCHRONIZED_IO",  SYSCONF,      NOFLAGS,
623     _SC_TIMERS,            "_POSIX_TIMERS",           SYSCONF,      NOFLAGS,
624
625     _SC_LOGIN_NAME_MAX,    "LOGIN_NAME_MAX",          SYSCONF,      NOFLAGS,
626
627     _SC_THREAD_DESTRUCTOR_ITERATIONS, "PTHREAD_DESTRUCTOR_ITERATIONS", SYSCONF,      NOFLAGS,
628
629     _SC_THREAD_KEYS_MAX,   "PTHREAD_KEYS_MAX",        SYSCONF,      NOFLAGS,
630     _SC_THREAD_STACK_MIN,  "PTHREAD_STACK_MIN",       SYSCONF,      NOFLAGS,
631     _SC_THREAD_THREADS_MAX, "PTHREAD_THREADS_MAX",     SYSCONF,      NOFLAGS,
632
633     _SC_THREADS,           "_POSIX_THREADS",          SYSCONF,      NOFLAGS,
634
635     _SC_THREAD_ATTR_STACKADDR, "POSIX_THREAD_ATTR_STACKADDR", SYSCONF,      NOFLAGS,
636     _SC_THREAD_ATTR_STACKADDR, "XOPEN_THREAD_ATTR_STACKADDR", NOFLAGS,
637     _SC_THREAD_ATTR_STACKADDR, "_POSIX_THREAD_ATTR_STACKADDR", SYSCONF,      NOFLAGS,
638
639     _SC_THREAD_ATTR_STACKSIZE, "POSIX_THREAD_ATTR_STACKSIZE", SYSCONF,      NOFLAGS,
640     _SC_THREAD_ATTR_STACKSIZE, "XOPEN_THREAD_ATTR_STACKSIZE", NOFLAGS,
641     _SC_THREAD_ATTR_STACKSIZE, "_POSIX_THREAD_ATTR_STACKSIZE", SYSCONF,      NOFLAGS,
642
643     _SC_THREAD_PRIORITY_SCHEDULING, "POSIX_THREAD_PRIORITY_SCHEDULING", SYSCONF,      NOFLAGS,
644     _SC_THREAD_PRIORITY_SCHEDULING, "XOPEN_THREAD_PRIORITY_SCHEDULING", NOFLAGS,
645     _SC_THREAD_PRIORITY_SCHEDULING, "_POSIX_THREAD_PRIORITY_SCHEDULING", SYSCONF,      NOFLAGS,
646
647     _SC_THREAD_PRIO_INHERIT, "POSIX_THREAD_PRIO_INHERIT", SYSCONF,      NOFLAGS,
648
649     _SC_THREAD_PRIO_INHERIT, "XOPEN_THREAD_PRIO_INHERIT", NOFLAGS,
650     _SC_THREAD_PRIO_INHERIT, "_POSIX_THREAD_PRIO_INHERIT", SYSCONF,      NOFLAGS,
651
652     _SC_THREAD_PRIO_PROTECT, "POSIX_THREAD_PRIO_PROTECT", SYSCONF,      NOFLAGS,
653
654     _SC_THREAD_PRIO_PROTECT, "XOPEN_THREAD_PRIO_PROTECT", NOFLAGS,
655     _SC_THREAD_PRIO_PROTECT, "_POSIX_THREAD_PRIO_PROTECT", SYSCONF,      NOFLAGS,
656
657     _SC_THREAD_PROCESS_SHARED, "POSIX_THREAD_PROCESS_SHARED", SYSCONF,      NOFLAGS,
658
659     _SC_THREAD_PROCESS_SHARED, "XOPEN_THREAD_PROCESS_SHARED", NOFLAGS,
660     _SC_THREAD_PROCESS_SHARED, "_POSIX_THREAD_PROCESS_SHARED", SYSCONF,      NOFLAGS,
661
662     _SC_THREAD_SAFE_FUNCTIONS, "POSIX_THREAD_SAFE_FUNCTIONS", SYSCONF,      NOFLAGS,
663
664     _SC_THREAD_SAFE_FUNCTIONS, "XOPEN_THREAD_SAFE_FUNCTIONS", NOFLAGS,
665     _SC_THREAD_SAFE_FUNCTIONS, "_POSIX_THREAD_SAFE_FUNCTIONS", SYSCONF,      NOFLAGS,
666
667     _SC_TTY_NAME_MAX,        "TTY_NAME_MAX",           SYSCONF,      NOFLAGS,
668     _SC_XOPEN_REALTIME,     "_XOPEN_REALTIME",        SYSCONF,      NOFLAGS,
669     _SC_XOPEN_REALTIME_THREADS, "XOPEN_REALTIME_THREADS", SYSCONF,      NOFLAGS,
670
671     _SC_XOPEN_REALTIME_THREADS, "XOPEN_REALTIME_THREADS", SYSCONF,      NOFLAGS,
672
673     _SC_XOPEN_REALTIME_THREADS, "XOPEN_REALTIME_THREADS", SYSCONF,      NOFLAGS,
674
675     _SC_XOPEN_REALTIME_THREADS, "XOPEN_REALTIME_THREADS", SYSCONF,      NOFLAGS,
676
677     /* Due to ambiguity in the standard we need
678      * to recognize both the underscored name
679      * and the non-underscored version in the
680      * following variables.
681      */
682     _SC_XBS5_ILP32_OFF32,   "XBS5_ILP32_OFF32",       SYSCONF,      NOFLAGS,
683     _SC_XBS5_ILP32_OFF32,   "_XBS5_ILP32_OFF32",      SYSCONF,      NOFLAGS,
684
685     _SC_XBS5_ILP32_OFFBIG,  "XBS5_ILP32_OFFBIG",       SYSCONF,      NOFLAGS,
686     _SC_XBS5_ILP32_OFFBIG,  "_XBS5_ILP32_OFFBIG",      SYSCONF,      NOFLAGS,
687
688     _SC_XBS5_LP64_OFF64,   "XBS5_LP64_OFF64",        SYSCONF,      NOFLAGS,
689     _SC_XBS5_LP64_OFF64,   "_XBS5_LP64_OFF64",       SYSCONF,      NOFLAGS,
690
691     _SC_XBS5_LPBIG_OFFBIG, "XBS5_LPBIG_OFFBIG",       SYSCONF,      NOFLAGS,
692     _SC_XBS5_LPBIG_OFFBIG, "_XBS5_LPBIG_OFFBIG",      SYSCONF,      NOFLAGS,
693
694     /*
695      * UNIX 03 additions for sysconf() names, and names that are not in interfaces
696      */
697     _SC_REGEXP,              "_POSIX_REGEXP",          SYSCONF,      NOFLAGS,
698     _SC_SHELL,               "_POSIX_SHELL",          SYSCONF,      NOFLAGS,
699     _SC_HOST_NAME_MAX,      "HOST_NAME_MAX",         SYSCONF,      NOFLAGS,
700     _SC_READER_WRITER_LOCKS, "POSIX_READER_WRITER_LOCKS", SYSCONF,      NOFLAGS,
701     _SC_IPV6,                "_POSIX_IPV6",          SYSCONF,      NOFLAGS,
702     _SC_RAW_SOCKETS,        "_POSIX_RAW_SOCKETS",    SYSCONF,      NOFLAGS,
703     _SC_XOPEN_STREAMS,      "_XOPEN_STREAMS",       SYSCONF,      NOFLAGS,
704     _SC_SYMLOOP_MAX,        "SYMLOOP_MAX",          SYSCONF,      NOFLAGS,
705     _SC_2_PBS,               "_POSIX2_PBS",          SYSCONF,      NOFLAGS,
706     _SC_2_PBS_ACCOUNTING,  "POSIX2_PBS_ACCOUNTING", SYSCONF,      NOFLAGS,
707     _SC_2_PBS_CHECKPOINT,  "POSIX2_PBS_CHECKPOINT",  SYSCONF,      NOFLAGS,
708     _SC_2_PBS_LOCATE,       "_POSIX2_PBS_LOCATE",    SYSCONF,      NOFLAGS,
709     _SC_2_PBS_MESSAGE,      "_POSIX2_PBS_MESSAGE",   SYSCONF,      NOFLAGS,
710     _SC_2_PBS_TRACK,        "_POSIX2_PBS_TRACK",     SYSCONF,      NOFLAGS,
711
712     _SC_ADVISORY_INFO,      "_POSIX_ADVISORY_INFO",  SYSCONF,      NOFLAGS,
713     _SC_BARRIERS,           "_POSIX_BARRIERS",        SYSCONF,      NOFLAGS,
714     _SC_CLOCK_SELECTION,   "_POSIX_CLOCK_SELECTION", SYSCONF,      NOFLAGS,
715     _SC_CPUTIME,             "_POSIX_CPUTIME",        SYSCONF,      NOFLAGS,
716     _SC_MONOTONIC_CLOCK,   "_POSIX_MONOTONIC_CLOCK", SYSCONF,      NOFLAGS,
717     _SC_SPAWN,               "_POSIX_SPAWN",          SYSCONF,      NOFLAGS,
718     _SC_SPIN_LOCKS,         "_POSIX_SPIN_LOCKS",     SYSCONF,      NOFLAGS,
719     _SC_SPORADIC_SERVER,   "_POSIX_SPORADIC_SERVER", SYSCONF,      NOFLAGS,
720     _SC_SS_REPL_MAX,        "_POSIX_SS_REPL_MAX",    SYSCONF,      NOFLAGS,
721     _SC_THREAD_CPUTIME,     "_POSIX_THREAD_CPUTIME",  SYSCONF,      NOFLAGS,
722     _SC_THREAD_SPORADIC_SERVER, "POSIX_THREAD_SPORADIC_SERVER", SYSCONF,      NOFLAGS,
723
724     _SC_TIMEOUTS,            "_POSIX_TIMEOUTS",       SYSCONF,      NOFLAGS,
725     _SC_TRACE,               "_POSIX_TRACE",          SYSCONF,      NOFLAGS,
726     _SC_TRACE_EVENT_FILTER, "POSIX_TRACE_EVENT_FILTER", SYSCONF,      NOFLAGS,
727     _SC_TRACE_EVENT_NAME_MAX, "POSIX_TRACE_EVENT_NAME_MAX", SYSCONF,      NOFLAGS,
728
729     _SC_TRACE_INHERIT,      "_POSIX_TRACE_INHERIT",  SYSCONF,      NOFLAGS,
730     _SC_TRACE_LOG,           "_POSIX_TRACE_LOG",       SYSCONF,      NOFLAGS,
731     _SC_TRACE_NAME_MAX,     "POSIX_TRACE_NAME_MAX",  SYSCONF,      NOFLAGS,
732     _SC_TRACE_SYS_MAX,      "_POSIX_TRACE_SYS_MAX",  SYSCONF,      NOFLAGS,
733     _SC_TRACE_USER_EVENT_MAX, "POSIX_TRACE_USER_EVENT_MAX", SYSCONF,      NOFLAGS,
734
735     _SC_TYPED_MEMORY_OBJECTS, "POSIX_TYPED_MEMORY_OBJECTS", SYSCONF,      NOFLAGS,
736
737     /* Due to an ambiguity in the specification, we need to recognize both
738      * the underscored POSIX name and the non-underscored name. In addition,
739      * we need to permit the use of the _V6_* environment names specified
740
741

```

```

742     * in unistd.h.
743     */
744     _SC_V6_ILP32_OFF32,      "_V6_ILP32_OFF32",      SYSCONF,      NOFLAGS,
745     _SC_V6_ILP32_OFF32,      "_POSIX_V6_ILP32_OFF32",  SYSCONF,      NOFLAGS,
746     _SC_V6_ILP32_OFF32,      "POSIX_V6_ILP32_OFF32",   SYSCONF,      NOFLAGS,
747
748     _SC_V6_ILP32_OFFBIG,    "_V6_ILP32_OFFBIG",     SYSCONF,      NOFLAGS,
749     _SC_V6_ILP32_OFFBIG,    "_POSIX_V6_ILP32_OFFBIG", SYSCONF,      NOFLAGS,
750     _SC_V6_ILP32_OFFBIG,    "POSIX_V6_ILP32_OFFBIG",  SYSCONF,      NOFLAGS,
751
752     _SC_V6_LP64_OFF64,      "_V6_LP64_OFF64",      SYSCONF,      NOFLAGS,
753     _SC_V6_LP64_OFF64,      "_POSIX_V6_LP64_OFF64",  SYSCONF,      NOFLAGS,
754     _SC_V6_LP64_OFF64,      "POSIX_V6_LP64_OFF64",   SYSCONF,      NOFLAGS,
755
756     _SC_V6_LPBIG_OFFBIG,    "_V6_LPBIG_OFFBIG",    SYSCONF,      NOFLAGS,
757     _SC_V6_LPBIG_OFFBIG,    "_POSIX_V6_LPBIG_OFFBIG", SYSCONF,      NOFLAGS,
758     _SC_V6_LPBIG_OFFBIG,    "POSIX_V6_LPBIG_OFFBIG",  SYSCONF,      NOFLAGS,
759 */
760 /* System V Interface Definition
761 */
762     _SC_NPROCESSORS_CONF,   "NPROCESSORS_CONF",    SYSCONF,      NOFLAGS,
763     _SC_NPROCESSORS_MAX,    "NPROCESSORS_MAX",     SYSCONF,      NOFLAGS,
764     _SC_NPROCESSORS_ONLN,   "NPROCESSORS_ONLN",   SYSCONF,      NOFLAGS,
765     0,                      NULL,                  0,          0
766 };

767 /*
768 * Print usage message.
769 */
770 static int
771 usage()
772 {
773     (void) fputs(gettext(
774         "usage:\`tgetconf [-v specification] system_var\n"
775         "\`tgetconf [-v specification] path_var pathname\n"
776         "\`tgetconf -a\`n"), stderr);
777     return (2);
778 }

779 static int
780 namecmp(const void *a, const void *b)
781 {
782     return (strcoll((const struct sctab *)a)->name,
783             ((const struct sctab *)b)->name));
784 }

785 static int
786 getconf(struct sctab *scp, int argc, char *name, char *file)
787 {
788     register size_t len;
789     register char *buffer;
790     long value;
791
792     switch (scp->type) {
793     case SELFCONF:
794         if (argc > 2)
795             return (usage());
796         if (scp->flag & UNDEFINED ||
797             (!(scp->flag&UNSIGNED_VALUE) && scp->value == -1)) {
798             /* DO NOT TRANSLATE */
799             (void) printf("undefined\n");
800         } else
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
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
```

```

874         /* Probably problems with the pathname itself */
875         if (aflag) {
876             (void) printf("%s\n", file);
877         } else {
878             (void) printf("%s", file);
879         }
880         return (1);
881     }
882     (void) printf("%ld\n", value);
883     break;
884 }
885 return (0);
886 }

889 int
890 main(int argc, char **argv)
891 {
892     register struct sctab *scp;
893     int c;
894     int exstat = 0;

895     (void) setlocale(LC_ALL, "");
896 #if !defined(TEXT_DOMAIN) /* Should be defined by cc -D */
897 #define TEXT_DOMAIN "SYS_TEST"
898 #endif
900     (void) textdomain(TEXT_DOMAIN);

902     while ((c = getopt(argc, argv, "av:")) != -1)
903         switch (c) {
904         case 'a':
905             aflag = 1;
906             break;
907         case 'v':
908             /*
909              * Unix98 adds the -v option to allow
910              * programming 'specifications' to be
911              * indicated, for present purposes
912              * the specification isn't really
913              * doing anything of value, so for
914              * the moment getopt just processes the
915              * option value and argv[1] is adjusted.
916              *
917              * At some later date we might want to
918              * do specification processing at this
919              * point.
920              */
921             break;
922         case '?':
923             return (usage());
924         }
925     argc -= optind-1;
926     argv += optind-1;

927     if ((aflag && argc >= 2) || (!aflag && argc < 2))
928         return (usage());
929     if (aflag) {
930
931 #define TabStop      8
932 #define RightColumn 32
933 #define DefPathName "."
934
935         /*
936          * sort the table by the "name" field
937          * so we print it in sorted order
938         */

```

```

940     qsort(&sctab[0], (sizeof (sctab)/sizeof (struct sctab))-1,
941           sizeof (struct sctab), namecmp);
942
943     /*
944      * print all the known symbols and their values
945      */
946     for (scp = &sctab[0]; scp->name != NULL; ++scp) {
947         int stat;
948
949         /*
950          * create 2 columns:
951          *   variable name in the left column,
952          *   value in the right column.
953          * The right column starts at a tab stop.
954         */
955         (void) printf("%s:\t", scp->name);
956
957         c = strlen(scp->name) + 1;
958         c = (c+TabStop) / TabStop, c *= TabStop;
959         for (; c < RightColumn; c += TabStop)
960             (void) putchar('t');
961
962         /*
963          * for pathconf() related variables that require
964          * a pathname, use "."
965         */
966         stat = getconf(scp, scp->type == PATHCONF ? 3 : 2,
967                      scp->name, DefPathName);
968
969         exstat |= stat;
970
971         /*
972          * if stat != 0, then an error message was already
973          * printed in getconf(),
974          * so don't need to print one here
975         */
976     } else {
977
978         /*
979          * find the name of the specified variable (argv[1])
980          * and print its value.
981         */
982         for (scp = &sctab[0]; scp->name != NULL; ++scp)
983             if (strcmp(argv[1], scp->name) == 0) {
984                 exstat = getconf(scp, argc, argv[1], argv[2]);
985                 break;
986             }
987
988         /*
989          * if at last entry in table, then the user specified
990          * variable is invalid
991         */
992         if (scp->name == NULL) {
993             errno = EINVAL;
994             (void) fprintf(stderr, gettext(INVAL_ARG), argv[1]);
995             return (1);
996         }
997     }
998
999     return (exstat);
1000 }

```

```
*****
17542 Fri Sep 27 14:13:12 2013
new/usr/src/man/man1/getconf.1
4146 getconf(1) does not support XATTR_ENABLED, XATTR_EXISTS, SATTR_ENABLED, SAT
Reviewed by: Cedric Blancher <cedric.blancher@gmail.com>
*****
1 '\\" te
2 .\" Copyright (c) 1992, X/Open Company Limited All Rights Reserved
3 .\" Portions Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved
4 .\" Portions Copyright (c) 1982-2007 AT&T Knowledge Ventures
5 .\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
6 .\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
7 .\" are reprinted and reproduced in electronic form in the Sun OS Reference Manu
8 .\" and Electronics Engineers, Inc and The Open Group. In the event of any discr
9 .\" This notice shall appear on any product containing this material.
10 .\" The contents of this file are subject to the terms of the Common Development
11 .\" See the License for the specific language governing permissions and limitat
12 .\" the fields enclosed by brackets "[]" replaced with your own identifying info
13 .TH GETCONF 1 "Sep 27, 2013"
13 .TH GETCONF 1 "Jan 16, 2013"
14 .SH NAME
15 getconf \- get configuration values
16 .SH SYNOPSIS
17 .SS "/usr/bin/getconf, /usr/xpg4/bin/getconf, /usr/xpg6/bin/getconf"
18 .LP
19 .nf
20 \fB/usr/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIisystem_var\fR
21 .fi
23 .LP
24 .nf
25 \fB/usr/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIpPath_var\fR \fIpPathname
26 .fi
28 .LP
29 .nf
30 \fB/usr/bin/getconf\fR \fB-a\fR
31 .fi
33 .LP
34 .nf
35 \fB/usr/xpg4/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIisystem_var\fR
36 .fi
38 .LP
39 .nf
40 \fB/usr/xpg4/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIpPath_var\fR \fIpAt
41 .fi
43 .LP
44 .nf
45 \fB/usr/xpg4/bin/getconf\fR \fB-a\fR
46 .fi
48 .LP
49 .nf
50 \fB/usr/xpg6/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIisystem_var\fR
51 .fi
53 .LP
54 .nf
55 \fB/usr/xpg6/bin/getconf\fR [\fB-v\fR \fIspecification\fR] \fIpPath_var\fR \fIpAt
56 .fi
58 .LP
59 .nf
```

```
60 \fB/usr/xpg6/bin/getconf\fR \fB-a\fR
61 .fi
63 .SS "ksh93"
64 .LP
65 .nf
66 \fBgetconf\fR [\fB-abdlpqrtw\fR] [\fB-c\fR RE] [\fB-n\fR RE] [\fB-s\fR RE] [\fB-
67 .fi
69 .SH DESCRIPTION
70 .SS "/usr/bin/getconf, /usr/xpg4/bin/getconf, /usr/xpg6/bin/getconf"
71 .sp
72 .LP
73 In the first synopsis form, the \fBgetconf\fR utility writes to the standard
74 output the value of the variable specified by \fIisystem_var\fR, in accordance
75 with \fIspecification\fR if the \fB-v\fR option is used.
76 .sp
77 .LP
78 In the second synopsis form, \fBgetconf\fR writes to the standard output the
79 value of the variable specified by \fIpPath_var\fR for the path specified by
80 \fIpPathname\fR, in accordance with \fIspecification\fR if the \fB-v\fR option
81 is used.
82 .sp
83 .LP
84 In the third synopsis form, \fBconfig\fR writes to the standard output the
85 names of the current system configuration variables.
86 .sp
87 .LP
88 The value of each configuration variable is determined as if it were obtained
89 by calling the function from which it is defined to be available. The value
90 reflects conditions in the current operating environment.
91 .SS "ksh93"
92 .sp
93 .LP
94 The \fBgetconf\fR built-in in \fBksh93\fR is associated with the \fB/bin\fR and
95 \fB/usr/bin\fR paths. It is invoked when \fBgetconf\fR is executed without a
96 pathname prefix and the pathname search finds a \fB/bin/getconf\fR or
97 \fB/usr/bin/getconf\fR executable.
98 .sp
99 .LP
100 \fBgetconf\fR displays the system configuration value for \fIname\fR. If
101 \fIname\fR is a filesystem specific variable, the value is determined relative
102 to path, or, the current directory, if path is omitted. If \fIvalue\fR is
103 specified, \fBgetconf\fR attempts to change the process local value to
104 \fIvalue\fR. \fB-\fR can be used in place of path when it is not relevant. Only
105 writable variables can be set. \fBreadonly\fR variables cannot be changed.
106 .sp
107 .LP
108 The current value for \fIname\fR is written to the standard output. If
109 \fIname\fR is valid but undefined, \fBundefined\fR is written to the standard
110 output. If \fIname\fR is invalid or an error occurs in determining its value, a
111 diagnostic is written to standard error and \fBgetconf\fR exits with a
112 \fBnon-zero\fR exit status.
113 .sp
114 .LP
115 More than one variable can be set or queried by providing the \fIname path
116 value\fR 3-tuple for each variable, specifying \fB-\fR for value when querying.
117 .sp
118 .LP
119 If no operands are specified, all known variables are written in
120 \fIname\fR\fB-\fR\fIvalue\fR form to the standard output, one per line. Only
121 one of \fB--call\fR, \fB--name\fR or \fB--standard\fR can be specified.
122 .SH OPTIONS
123 .SS "/usr/bin/getconf, /usr/xpg4/bin/getconf, /usr/xpg6/bin/getconf"
124 .sp
125 .LP
```

```

126 The following options are supported by \fB/usr/bin/getconf\fR,
127 \fB/usr/xpg4/bin/getconf\fR, and \fB/usr/xpg6/bin/getconf\fR
128 .sp
129 .ne 2
130 .na
131 \fB\fB-a\fR\fR
132 .ad
133 .RS 19n
134 Writes the names of the current system configuration variables to the standard
135 output.
136 .RE

138 .sp
139 .ne 2
140 .na
141 \fB\fB-v\fR\fIspecification\fR\fR
142 .ad
143 .RS 19n
144 Gives the specification which governs the selection of values for configuration
145 variables.
146 .RE

148 .SS "ksh93"
149 .sp
150 .LP
151 The following options are supported by the \fBgetconf\fR built-in command in
152 \fBksh93\fR:
153 .sp
154 .ne 2
155 .na
156 \fB\fB-a\fR\fR
157 .ad
158 .br
159 .na
160 \fB\fB--all\fR\fR
161 .ad
162 .RS 24n
163 Call the native \fBgetconf -a\fR.
164 .RE

166 .sp
167 .ne 2
168 .na
169 \fB\fB-b\fR\fR
170 .ad
171 .br
172 .na
173 \fB\fB--base\fR\fR
174 .ad
175 .RS 24n
176 List base variable name without \fBcall\fR and standard prefixes.
177 .RE

179 .sp
180 .ne 2
181 .na
182 \fB\fB-c\fR\fR
183 .ad
184 .br
185 .na
186 \fB\fB--call=\fR\fIRE\fR\fR
187 .ad
188 .RS 24n
189 Display variables with \fBcall\fR prefix that matches \fIRE\fR. The \fBcall\fR
190 prefixes are:
191 .RS +4

```

```

192 .TP
193 .ie t \((bu
194 .el o
195 \fBCS\fR
196 .sp
197 \fBconfstr\fR(3C),
198 .RE
199 .RS +4
200 .TP
201 .ie t \((bu
202 .el o
203 \fBPC\fR
204 .sp
205 \fBpathconf\fR(2)
206 .RE
207 .RS +4
208 .TP
209 .ie t \((bu
210 .el o
211 \fBSC\fR
212 .sp
213 \fBsysconf\fR(3C)
214 .RE
215 .RS +4
216 .TP
217 .ie t \((bu
218 .el o
219 \fBSI\fR
220 .sp
221 \fBsysinfo\fR(2)
222 .RE
223 .RS +4
224 .TP
225 .ie t \((bu
226 .el o
227 \fBX\fR
228 .sp
229 Constant value
230 .RE
231 .RE

233 .sp
234 .ne 2
235 .na
236 \fB\fB-d\fR\fR
237 .ad
238 .br
239 .na
240 \fB\fB--defined\fR\fR
241 .ad
242 .RS 24n
243 Only display defined values when no operands are specified.
244 .RE

246 .sp
247 .ne 2
248 .na
249 \fB\fB-l\fR\fR
250 .ad
251 .br
252 .na
253 \fB\fB--lowercase\fR\fR
254 .ad
255 .RS 24n
256 List variable names in lower case.
257 .RE

```

```

259 .sp
260 .ne 2
261 .na
262 \fB\fB-n\fR\fR
263 .ad
264 .br
265 .na
266 \fB\fB--name=\fR\fIRE\fR\fR
267 .ad
268 .RS 24n
269 Display variables with names that match \fIRE\fR.
270 .RE

272 .sp
273 .ne 2
274 .na
275 \fB\fB-p\fR\fR
276 .ad
277 .br
278 .na
279 \fB\fB--portable\fR\fR
280 .ad
281 .RS 24n
282 Display the named writable variables and values in a form that can be directly
283 executed by \fBksh93\fR(1) to set the values. If \fIname\fR is omitted, lists
284 all writable variables.
285 .RE

287 .sp
288 .ne 2
289 .na
290 \fB\fB-q\fR\fR
291 .ad
292 .br
293 .na
294 \fB\fB--quote\fR\fR
295 .ad
296 .RS 24n
297 Quote values (\fB"..." \fR).
298 .RE

300 .sp
301 .ne 2
302 .na
303 \fB\fB-r\fR\fR
304 .ad
305 .br
306 .na
307 \fB\fB--readonly\fR\fR
308 .ad
309 .RS 24n
310 Display the named \fBreadonly\fR variables in \fIname\fR\fB=\fR\fIvalue\fR
311 form. If \fIname\fR is omitted, lists all \fBreadonly\fR variables.
312 .RE

314 .sp
315 .ne 2
316 .na
317 \fB\fB-s\fR\fR
318 .ad
319 .br
320 .na
321 \fB\fB--standard=\fR\fIRE\fR\fR
322 .ad
323 .RS 24n

```

```

324 Display variables with standard prefix that matches \fIRE\fR. Use the
325 \fB--table\fR option to view all standard prefixes, including local additions.
326 .sp
327 The standard prefixes available on all systems are:
328 .sp
329 .in +2
330 .nf

332 AES
333 AST
334 C
335 GNU
336 POSIX
337 SVID
338 XBS5
339 XOPEN
340 XPG
341 .fi
342 .in -2
343 .sp

345 .RE

347 .sp
348 .ne 2
349 .na
350 \fB\fB-t\fR\fR
351 .ad
352 .br
353 .na
354 \fB\fB--table\fR\fR
355 .ad
356 .RS 24n
357 Display the internal table that contains the name, standard, standard section,
358 and system call symbol prefix for each variable.
359 .RE

361 .sp
362 .ne 2
363 .na
364 \fB\fB-w\fR\fR
365 .ad
366 .br
367 .na
368 \fB\fB--writable\fR\fR
369 .ad
370 .RS 24n
371 Display the named writable variables in \fIname\fR\fB=\fR\fIvalue\fR form. If
372 \fIname\fR is omitted, lists all writable variables.
373 .RE

375 .sp
376 .ne 2
377 .na
378 \fB\fB-v\fR\fR
379 .ad
380 .br
381 .na
382 \fB\fB--specification=name\fR\fR
383 .ad
384 .RS 24n
385 Call the native \fBgetconf -v\fR \fIname\fR.
386 .RE

388 .SH OPERANDS
389 .SS "/usr/bin/getconf, /usr/xpg4/bin/getconf, /usr/xpg6/bin/getconf"

```

```

390 .sp
391 .LP
392 The following operands are supported by \fB/usr/bin/getconf\fR,
393 \fB/usr/xpg4/bin/getconf\fR, and \fB/usr/xpg6/bin/getconf\fR:
394 .sp
395 .ne 2
396 .na
397 \fB\fIpath_var\fR\fR
398 .ad
399 .RS 12n
400 A name of a configuration variable whose value is available from the
401 \fBpathconf\fR(2) function. All of the values in the following table are
402 supported:
403 .RE

405 .sp
407 .sp
408 .TS
409 l 1 1
410 l 1 1 .
411 ACL_ENABLED NAME_MAX XATTR_ENABLED
412 LINK_MAX PATH_MAX XATTR_EXISTS
413 MAX_CANON PIPE_BUF _POSIX_CHOWN_RESTRICTED
414 MAX_INPUT SATTR_ENABLED _POSIX_NO_TRUNC
415 MIN_HOLE_SIZE SATTR_EXISTS _POSIX_VDISABLE
416 NAME_MAX _POSIX_CHOWN_RESTRICTED
417 PATH_MAX _POSIX_NO_TRUNC
418 PIPE_BUF _POSIX_VDISABLE
419 .TE

420 .sp
421 .ne 2
422 .na
423 \fB\fIpathname\fR\fR
424 .ad
425 .RS 14n
426 A path name for which the variable specified by \fIpath_var\fR is to be
427 determined.
428 .RE

429 .sp
430 .ne 2
431 .na
432 \fB\fIfilesystem_var\fR\fR
433 .ad
434 .RS 14n
435 A name of a configuration variable whose value is available from
436 \fBconfstr\fR(3C) or \fBsysconf\fR(3C). All of the values in the following
437 table are supported:
438 .RE

439 .sp
440 .TS
441 ARG_MAX BC_BASE_MAX
442 BC_DIM_MAX BC_SCALE_MAX
443 BC_STRING_MAX CHAR_BIT
444 CHARCLASS_NAME_MAX CHAR_MAX
445 CHAR_MIN CHILD_MAX
446 CLK_TCK COLL_WEIGHTS_MAX
447 CS_PATH EXPR_NEST_MAX
448 HOST_NAME_MAX INT_MAX

```

```

453 INT_MIN LFS64_CFLAGS
454 LFS64_LDFLAGS LFS64_LIBS
455 LFS64_LINTFLAGS LFS_CFLAGS
456 LFS_LDFLAGS LFS_LIBS
457 LFS_LINTFLAGS LINE_MAX
458 LONG_BIT LONG_MAX
459 LONG_MIN MB_LEN_MAX
460 NGROUPS_MAX NL_ARGMAX
461 NL_LANGMAX NL_MSGMAX
462 NL_NMAX NL_SETMAX
463 NL_TEXTMAX NPROCESSORS_CONF
464 NPROCESSORS_MAX NPROCESSORS_ONLN
465 NZERO OPEN_MAX
466 POSIX2_BC_BASE_MAX POSIX2_BC_DIM_MAX
467 POSIX2_BC_SCALE_MAX POSIX2_BC_STRING_MAX
468 POSIX2_C_BIND POSIX2_C_DEV
469 POSIX2_CHAR_TERM POSIX2_COLL_WEIGHTS_MAX
470 POSIX2_C_VERSION POSIX2_EXPR_NEST_MAX
471 POSIX2_FORT_DEV POSIX2_FORT_RUN
472 POSIX2_LINE_MAX POSIX2_LOCALEDEF
473 POSIX2_RE_DUP_MAX POSIX2_SW_DEV
474 POSIX2_SYMLINKS POSIX2_UPE
475 POSIX2_VERSION POSIX_ALLOC_SIZE_MIN
476 POSIX_REC_INCR_XFER_SIZE POSIX_REC_MAX_XFER_SIZE
477 POSIX_REC_MIN_XFER_SIZE POSIX_REC_XFER_ALIGN
478 POSIX_V6_ILP32_OFF32 POSIX_V6_ILP32_OFF32_CFLAGS
479 POSIX_V6_ILP32_OFF32_LDFLAGS POSIX_V6_ILP32_OFF32_LIBS
480 POSIX_V6_ILP32_OFFBIG POSIX_V6_ILP32_OFFBIG_CFLAGS
481 POSIX_V6_ILP32_OFFBIG_LDFLAGS POSIX_V6_ILP32_OFFBIG_LIBS
482 POSIX_V6_LP64_OFFSET POSIX_V6_LP64_OFFSET_CFLAGS
483 POSIX_V6_LP64_OFFSET_LDFLAGS POSIX_V6_LP64_OFFSET_LIBS
484 POSIX_V6_LPBIG_OFFSET POSIX_V6_LPBIG_OFFSET_CFLAGS
485 POSIX_V6_LPBIG_OFFSET_LDFLAGS POSIX_V6_LPBIG_OFFSET_LIBS
486 POSIX_V6_WIDTH_RESTRICTED_ENVS SYMLINK_MAX
487 SYMLOOP_MAX _POSIX2_BC_BASE_MAX
488 _POSIX2_BC_DIM_MAX _POSIX2_BC_SCALE_MAX
489 _POSIX2_BC_STRING_MAX _POSIX2_CHARCLASS_NAME_MAX
490 _POSIX2_CHAR_TERM _POSIX2_COLL_WEIGHTS_MAX
491 _POSIX2_C_BIND _POSIX2_C_DEV
492 _POSIX2_C_VERSION _POSIX2_EXPR_NEST_MAX
493 _POSIX2_FORT_DEV _POSIX2_FORT_RUN
494 _POSIX2_LINE_MAX _POSIX2_LOCALEDEF
495 _POSIX2_PBS _POSIX2_PBS_ACCOUNTING
496 _POSIX2_PBS_CHECKPOINT _POSIX2_PBS_LOCATE
497 _POSIX2_PBS_MESSAGE _POSIX2_PBS_TRACK
498 _POSIX2_RE_DUP_MAX _POSIX2_SW_DEV
499 _POSIX2_UPE _POSIX2_VERSION
500 _POSIX_ADVISORY_INFO _POSIX_AIO_LISTIO_MAX
501 _POSIX_AIO_MAX _POSIX_ARG_MAX
502 _POSIX_ASYNC_IO _POSIX_BARRIERS
503 _POSIX_CHILD_MAX _POSIX_CLOCKRES_MIN
504 _POSIX_CLOCK_SELECTION _POSIX_CPUTIME
505 _POSIX_DELAYTIMER_MAX _POSIX_HOST_NAME_MAX
506 _POSIX_IPV6 _POSIX_JOB_CONTROL
507 _POSIX_LINK_MAX _POSIX_LOGIN_NAME_MAX
508 _POSIX_MAX_CANON _POSIX_MAX_INPUT
509 _POSIX_MONOTONIC_CLOCK _POSIX_MQ_OPEN_MAX
510 _POSIX_MQ_PRIO_MAX _POSIX_NAME_MAX
511 _POSIX_NGROUPS_MAX _POSIX_OPEN_MAX
512 _POSIX_PATH_MAX _POSIX_PIPE_BUF
513 _POSIX_PRIO_IO _POSIX_RAW_SOCKETS
514 _POSIX_READER_WRITER_LOCKS _POSIX_REGEXP
515 _POSIX_RE_DUP_MAX _POSIX_RTSIG_MAX
516 _POSIX_SAVED_IDS _POSIX_SEM_NSEMS_MAX
517 _POSIX_SEM_VALUE_MAX _POSIX_SHELL
518 _POSIX_SIGQUEUE_MAX _POSIX_SPAWN

```

```

519 _POSIX_SPIN_LOCKS      _POSIX_SPORADIC_SERVER
520 _POSIX_SSIZE_MAX       _POSIX_SS_REPL_MAX
521 _POSIX_STREAM_MAX      _POSIX_SYMLINK_MAX
522 _POSIX_SYMLOOP_MAX     _POSIX_SYNC_IO
523 _POSIX_THREAD_ATTR_STACKADDR _POSIX_THREAD_ATTR_STACKSIZE
524 _POSIX_THREAD_CPUTIME   _POSIX_THREAD_DESTRUCTOR_ITERATIONS
525 _POSIX_THREAD_KEYS_MAX  _POSIX_THREAD_PRIORITY_SCHEDULING
526 _POSIX_THREAD_PRIO_INHERIT _POSIX_THREAD_PRIO_PROTECT
527 _POSIX_THREAD_PROCESS_SHARED _POSIX_THREAD_SAFE_FUNCTIONS
528 _POSIX_THREAD_SPORADIC_SERVER _POSIX_THREAD_THREADS_MAX
529 _POSIX_TIMEOUTS _POSIX_TIMER_MAX
530 _POSIX_TRA_POSIX_TIMER_MAXCE _POSIX_TIMESTAMP_RESOLUTION
531 _POSIX_TRACE_EVENT_FILTER _POSIX_TRACE_EVENT_NAME_MAX
532 _POSIX_TRACE_INHERIT    _POSIX_TRACE_LOG
533 _POSIX_TRACE_NAME_MAX   _POSIX_TRACE_SYS_MAX
534 _POSIX_TRACE_USER_EVENT_MAX _POSIX_TTY_NAME_MAX
535 _POSIX_TYPED_MEMORY_OBJECTS _POSIX_TZNAME_MAX
536 _POSIX_VERSION _POSIX_V6_ILP32_OFF32
537 _POSIX_V6_ILP32_OFFBIG _POSIX_V6_LP64_OFF64
538 _POSIX_V6_LPBIG_OFFBIG _V6_ILP32_OFF32
539 _V6_ILP32_OFFBIG        _V6_LP64_OFF64
540 _V6_LPBIG_OFFBIG        RE_DUP_MAX
541 SCHAR_MAX              SCHAR_MIN
542 SHRT_MAX               SHRT_MIN
543 SSIZE_MAX              STREAM_MAX
544 TMP_MAX                TZNAME_MAX
545 UCHAR_MAX              UINT_MAX
546 ULONG_MAX              USHRT_MAX
547 WORD_BIT               XBS5_ILP32_OFF32
548 XBS5_ILP32_OFF32_CFLAGS XBS5_ILP32_OFF32_LDFLAGS
549 XBS5_ILP32_OFF32_LIBS  XBS5_ILP32_OFF32_LINTFLAGS
550 XBS5_ILP32_OFFBIG      XBS5_ILP32_OFFBIG_CFLAGS
551 XBS5_ILP32_OFFBIG_LDFLAGS XBS5_ILP32_OFFBIG_LIBS
552 XBS5_ILP32_OFFBIG_LINTFLAGS XBS5_LP64_OFF64
553 XBS5_LP64_OFF64_CFLAGS XBS5_LP64_OFF64_LDFLAGS
554 XBS5_LP64_OFF64_LIBS   XBS5_LP64_OFF64_LINTFLAGS
555 XBS5_LPBIG_OFFBIG     XBS5_LPBIG_OFFBIG_CFLAGS
556 XBS5_LPBIG_OFFBIG_LDFLAGS XBS5_LPBIG_OFFBIG_LIBS
557 XBS5_LPBIG_OFFBIG_LINTFLAGS _XOPEN_CRYPT
558 _XOPEN_ENH_I18N _XOPEN_IOV_MAX
559 _XOPEN_LEGACY    _XOPEN_NAME_MAX
560 _XOPEN_PATH_MAX   _XOPEN_SHM
561 _XOPEN_STREAMS   _XOPEN_VERSION
562 _XOPEN_XCU_VERSION _XOPEN_XPG2
563 _XOPEN_XPG3      _XOPEN_XPG4
564 .TE

566 .sp
567 .LP
568 The symbol \fBPATH\fR also is recognized, yielding the same value as the
569 \fBconfstr()\fR name value \fBCS_PATH\fR.
570 .SH USAGE
571 .sp
572 .LP
573 See \fBlargefile\fR(5) for the description of the behavior of
574 \fB/usr/bin/getconf\fR when encountering files greater than or equal to 2 Gbyte
575 (231 bytes).
576 .SH EXAMPLES
577 .LP
578 \fBExample 1 \fRWriting the Value of a Variable
579 .sp
580 .LP
581 This example illustrates the value of \fB{NGROUPS_MAX}\fR

583 .sp
584 .in +2

```

```

585 .nf
586 example% \fBgetconf NGROUPS_MAX\fR
587 .fi
588 .in -2
589 .sp

591 .LP
592 \fBExample 2 \fRWriting the Value of a Variable for a Specific Directory
593 .sp
594 .LP
595 This example illustrates the value of \fBNAME_MAX\fR for a specific directory:

597 .sp
598 .in +2
599 .nf
600 example% \fBgetconf NAME_MAX /usr\fR
601 .fi
602 .in -2
603 .sp

605 .LP
606 \fBExample 3 \fRDealing with Unspecified Results
607 .sp
608 .LP
609 This example shows how to deal more carefully with results that might be
610 unspecified:

612 .sp
613 .in +2
614 .nf
615 \fBif value=$(getconf PATH_MAX /usr); then
616 if [ "$value" = "undefined" ]; then
617 echo PATH_MAX in /usr is infinite.
618 else
619 echo PATH_MAX in /usr is $value.
620 fi
621 else
622 echo Error in getconf.
623 fi\fR
624 .fi
625 .in -2

627 .sp
628 .LP
629 For example:

631 .sp
632 .in +2
633 .nf
634 \fBsysconf(_SC_POSIX_C_BIND);\fR
635 .fi
636 .in -2

638 .sp
639 .LP
640 and

642 .sp
643 .in +2
644 .nf
645 \fBsystem("getconf POSIX2_C_BIND");\fR
646 .fi
647 .in -2

649 .sp
650 .LP

```

651 in a C program could give different answers. The `\fBsysconf\fR` call supplies a  
 652 value that corresponds to the conditions when the program was either compiled  
 653 or executed, depending on the implementation. The `\fBsystem\fR` call to  
 654 `\fBgetconf\fR` always supplies a value corresponding to conditions when the  
 655 program is executed.

657 .SH ENVIRONMENT VARIABLES

658 .sp  
 659 .LP  
 660 See `\fBenviron\fR(5)` for descriptions of the following environment variables  
 661 that affect the execution of `\fBgetconf\fR`: `\fBLANG\fR`, `\fBLC_ALL\fR`,  
 662 `\fBLC_CTYPE\fR`, `\fBLC_MESSAGES\fR`, and `\fBNLSPATH\fR`.  
 663 .sp  
 664 .LP  
 665 The following environment variable is supported by the `\fBksh93\fR` built-in  
 666 `\fBgetconf\fR` command:  
 667 .sp  
 668 .ne 2  
 669 .na  
 670 `\fB\fB_AST_FEATURES\fR\fR`  
 671 .ad  
 672 .RS 17n  
 673 Local writable values that are different from the default are stored in the  
 674 `\fB_AST_FEATURES\fR` environment variable. The `\fB_AST_FEATURES\fR` value is a  
 675 space-separated list of `\fIname path value\fR` 3-tuples, where `\fIname\fR` is the  
 676 system configuration name, `\fIpath\fR` is the corresponding path, `\fB-\fR` if no  
 677 path is applicable, and `\fIvalue\fR` is the system configuration value.  
 678 .RE

680 .SH EXIT STATUS

681 .sp  
 682 .LP  
 683 The following exit values are returned:  
 684 .sp  
 685 .ne 2  
 686 .na  
 687 `\fB\fB0\fR\fR`  
 688 .ad  
 689 .RS 6n  
 690 The specified variable is valid and information about its current state was  
 691 written successfully.  
 692 .RE

694 .sp  
 695 .ne 2  
 696 .na  
 697 `\fB\fB>0\fR\fR`  
 698 .ad  
 699 .RS 6n  
 700 An error occurred.  
 701 .RE

703 .SH ATTRIBUTES

704 .sp  
 705 .LP  
 706 See `\fBattributes\fR(5)` for descriptions of the following attributes:  
 707 .SS "/usr/bin/getconf, /usr/xpg4/bin/getconf, /usr/xpg6/bin/getconf"  
 708 .sp  
 710 .sp  
 711 .TS  
 712 box;  
 713 c | c  
 714 l | l .  
 715 ATTRIBUTE TYPE ATTRIBUTE VALUE  
 716 \_

717 Interface Stability Committed  
 718 \_  
 719 Standard See `\fBstandards\fR(5)`.  
 720 .TE  
 722 .SS "ksh93"  
 723 .sp  
 725 .sp  
 726 .TS  
 727 box;  
 728 c | c  
 729 l | l .  
 730 ATTRIBUTE TYPE ATTRIBUTE VALUE  
 731 \_  
 732 Interface Stability See below.  
 733 .TE  
 735 .sp  
 736 .LP  
 737 The `\fBksh93\fR` built-in binding to `\fB/bin\fR` and `\fB/usr/bin\fR` is Volatile.  
 738 The built-in command-line interface is Committed.  
 739 .SH SEE ALSO  
 740 .sp  
 741 .LP  
 742 `\fBksh93\fR(1)`, `\fBsh\fR(1)`, `\fBpathconf\fR(2)`, `\fBsysinfo\fR(2)`,  
 743 `\fBconfstr\fR(3C)`, `\fBsysconf\fR(3C)`, `\fBattributes\fR(5)`, `\fBenviron\fR(5)`,  
 744 `\fBlargefile\fR(5)`, `\fBstandards\fR(5)`