```
**********************************************************
   15716 Sat Nov  8 22:21:53 2014
new/usr/src/man/man5/standards.5
5058 standards(5) Duplicate line in a table
**********************************************************
'\" te
.\" Copyright (c) 2007, Sun Microsystems, Inc.  All Rights Reserved.
.\" The contents of this file are subject to the terms of the Common Development
.\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
.\" When distributing Covered Code, include this CDDL HEADER in each file and in
.TH STANDARDS 5 "Nov 08, 2014"
.TH STANDARDS 5 "Jan 14, 2004"
.SH NAME
standards, ANSI, C, C++, ISO, POSIX, POSIX.1, POSIX.2, SUS, SUSv2, SUSv3, SVID,
SVID3, XNS, XNS4, XNS5, XPG, XPG3, XPG4, XPG4v2 \- standards and specifications
supported by Solaris
.SH DESCRIPTION
.sp
.LP
Solaris 10 supports IEEE Std 1003.1 and IEEE Std 1003.2, commonly known as
POSIX.1 and POSIX.2, respectively. The following table lists each version of
these standards with a brief description and the SunOS or Solaris release that
first conformed to it.
.sp

.sp
.TS
c c c
l l l .
POSIX Standard  Description     Release
_
POSIX.1-1988    system interfaces and headers   SunOS 4.1
_
POSIX.1-1990    POSIX.1-1988 update     Solaris 2.0
_
POSIX.1b-1993   realtime extensions     Solaris 2.4
_
POSIX.1c-1996   threads extensions      Solaris 2.6
_
POSIX.2-1992    shell and utilities     Solaris 2.5
_
POSIX.2a-1992   interactive shell and utilities Solaris 2.5
_
POSIX.1-2001    T{
POSIX.1-1990, POSIX.1b-1993, POSIX.1c-1996, POSIX.2-1992, and POSIX.2a-1992 upda
T}      Solaris 10
.TE

.sp
.LP
Solaris 10 also  supports the X/Open Common Applications Environment (CAE)
Portability Guide Issue 3 (XPG3) and Issue 4 (XPG4); Single UNIX Specification
(SUS, also known as XPG4v2); Single UNIX Specification, Version 2 (SUSv2); and
Single UNIX Specification, Version 3 (SUSv3). Both XPG4 and SUS include
Networking Services Issue 4 (XNS4). SUSv2 includes Networking Services Issue 5
(XNS5).
.sp
.LP
The following table lists each X/Open specification with a brief description
and the SunOS or Solaris release that first conformed to it.
.sp

.sp
.TS
c c c
c c c .
```



```
X/Open CAE
_
 Specification  Description     Release
_
XPG3    T{
superset of POSIX.1-1988 containing utilities from SVID3
T}      SunOS 4.1
_
XPG4    T{
superset of POSIX.1-1990, POSIX.2-1992, and POSIX.2a-1992 containing extensions
T}      Solaris 2.4
_
SUS (XPG4v2)    T{
superset of XPG4 containing historical BSD interfaces widely used by common appl
T}      Solaris 2.6
_
XNS4    sockets and XTI interfaces      Solaris 2.6
_
SUSv2   T{
superset of SUS extended to support POSIX.1b-1993, POSIX.1c-1996, and ISO/IEC 98
T}      Solaris 7
_
XNS5    T{
superset and LP64-clean derivative of XNS4.
T}      Solaris 7
_
SUSv3   same as POSIX.1-2001    Solaris 10
.TE

.sp
.LP
The XNS4 specification is safe for use only in ILP32 (32-bit) environments and
should not be used for LP64 (64-bit) application environments. Use XNS5 or
SUSv3, which have LP64-clean interfaces that are portable across ILP32 and LP64
environments. Solaris releases 7 through 10 support both the ILP32 and LP64
environments.
.sp
.LP
Solaris releases 7 through 10 have been branded to conform to The Open Group's
UNIX 98 Product Standard. Solaris 10 has been branded to conform to The Open
Group's UNIX 03 Product Standard.
.sp
.LP
Solaris releases 2.0 through 10 support the interfaces specified by the System
V Interface Definition, Third Edition, Volumes 1 through 4 (SVID3).  Note,
however, that since the developers of this specification (UNIX Systems
Laboratories) are no longer in business and since this specification defers to
POSIX and X/Open CAE specifications, there is some disagreement about what is
currently required for conformance to this specification.
.sp
.LP
When \fBSun Studio C Compiler 5.6\fR is installed, Solaris releases 2.0 through
10 support the ANSI X3.159-1989 Programming Language - C and ISO/IEC 9899:1990
Programming Language - C (C) interfaces.
.sp
.LP
When \fBSun Studio C Compiler 5.6\fR is installed, Solaris releases 7 through
10 support ISO/IEC 9899:1990 Amendment 1:1995: C Integrity.
.sp
.LP
When \fBSun Studio C Compiler 5.6\fR is installed, Solaris 10 supports ISO/IEC
9899:1999 Programming Languages - C.
.sp
.LP
When \fBSun Studio C++ Compiler 5.6\fR is installed, Solaris releases 2.5.1
through 10 support ISO/IEC 14882:1998 Programming Languages - C++.  Unsupported
```

```
127 features of that standard are described in the compiler README file.
128 .SS "Utilities"
129 .sp
130 .LP
131 If the behavior required by POSIX.2, POSIX.2a, XPG4, SUS, or SUSv2 conflicts
132 with historical Solaris utility behavior, the original Solaris version of the
133 utility is unchanged; a new version that is standard-conforming has been
134 provided in \fB/usr/xpg4/bin\fR. If the behavior required by POSIX.1-2001 or
135 SUSv3 conflicts with historical Solaris utility behavior, a new version that is
136 standard-conforming has been provided in \fB/usr/xpg4/bin\fR or in
137 \fB/usr/xpg6/bin\fR. If the behavior required by POSIX.1-2001 or SUSv3
138 conflicts with POSIX.2, POSIX.2a, SUS, or SUSv2, a new version that is SUSv3
139 standard-conforming has been provided in \fB/usr/xpg6/bin\fR.
140 .sp
141 .LP
142 An application that wants to use standard-conforming utilitues must set the
143 \fBPATH\fR (\fBsh\fR(1) or \fBksh\fR(1)) or \fBpath\fR (\fBcsh\fR(1))
144 environment variable to specify the directories listed below in the order
145 specified to get the appropriate utilities:
146 .sp
147 .ne 2
148 .na
149 \fBSVID3, XPG3\fR
150 .ad
151 .sp .6
152 .RS 4n
153 .RS +4
154 .TP
155 1.
156 \fB/usr/ccs/bin\fR
157 .RE
158 .RS +4
159 .TP
160 2.
161 \fB/usr/bin\fR
162 .RE
163 .RS +4
164 .TP
165 3.
166 directory containing binaries for your compiler
167 .RE
168 .RS +4
169 .TP
170 4.
171 other directories containing binaries needed by the application
172 .RE
173 .RE

175 .sp
176 .ne 2
177 .na
178 \fBPOSIX.2, POSIX.2a, SUS, SUSv2, XPG4\fR
179 .ad
180 .sp .6
181 .RS 4n
182 .RS +4
183 .TP
184 1.
185 \fB/usr/xpg4/bin\fR
186 .RE
187 .RS +4
188 .TP
189 2.
190 \fB/usr/ccs/bin\fR
191 .RE
192 .RS +4
```



```
193 .TP
194 3.
195 \fB/usr/bin\fR
196 .RE
197 .RS +4
198 .TP
199 4.
200 directory containing binaries for your compiler
201 .RE
202 .RS +4
203 .TP
204 5.
205 other directories containing binaries needed by the application
206 .RE
207 .RE

209 .sp
210 .ne 2
211 .na
212 \fBPOSIX.1-2001, SUSv3\fR
213 .ad
214 .sp .6
215 .RS 4n
216 .RS +4
217 .TP
218 1.
219 \fB/usr/xpg6/bin\fR
220 .RE
221 .RS +4
222 .TP
223 2.
224 \fB/usr/xpg4/bin\fR
225 .RE
226 .RS +4
227 .TP
228 3.
229 \fB/usr/ccs/bin\fR
230 .RE
231 .RS +4
232 .TP
233 4.
234 \fB/usr/bin\fR
235 .RE
236 .RS +4
237 .TP
238 5.
239 directory containing binaries for your compiler
240 .RE
241 .RS +4
242 .TP
243 6.
244 other directories containing binaries needed by the application
245 .RE
246 .RE

248 .SS "Feature Test Macros"
249 .sp
250 .LP
251 Feature test macros are used by applications to indicate additional sets of
252 features that are desired beyond those specified by the C standard. If an
253 application uses only those interfaces and headers defined by a particular
254 standard (such as POSIX or X/Open CAE),  then it need only define the
255 appropriate feature test macro specified by that standard. If the application
256 is using interfaces and headers not defined by that standard, then in addition
257 to defining the appropriate standard feature test macro, it must also define
258 \fB__EXTENSIONS__\fR. Defining \fB__EXTENSIONS__\fR provides the application
```

```
259 with access to all interfaces and headers not in conflict with the specified
260 standard. The application must define \fB__EXTENSIONS__\fR either on the
261 compile command line or within the application source files.
262 .SS "1989 ANSI C, 1990 ISO C, 1999 ISO C"
263 .sp
264 .LP
265 No feature test macros need to be defined to indicate that an application is a
266 conforming C application.
267 .SS "ANSI/ISO C++"
268 .sp
269 .LP
270 ANSI/ISO C++ does not define any feature test macros. If the standard C++
271 announcement macro \fB__cplusplus\fR is predefined to value 199711 or greater,
272 the compiler operates in a standard-conforming mode, indicating C++ standards
273 conformance. The value 199711 indicates conformance to ISO/IEC 14882:1998, as
274 required by that standard.  (As noted above, conformance to the standard is
275 incomplete.)  A standard-conforming mode is not available with compilers prior
276 to Sun WorkShop C++ 5.0.
277 .sp
278 .LP
279 C++ bindings are not defined for POSIX or X/Open CAE, so specifying feature
280 test macros such as \fB_POSIX_SOURCE\fR, \fB_POSIX_C_SOURCE\fR, and
281 \fB_XOPEN_SOURCE\fR can result in compilation errors due to conflicting
282 requirements of standard C++ and those specifications.
283 .SS "POSIX"
284 .sp
285 .LP
286 Applications that are intended to be conforming POSIX.1 applications must
287 define the feature test macros specified by the standard before including any
288 headers.  For the standards listed below, applications must define the feature
289 test macros listed.  Application writers must check the corresponding standards
290 for other macros that can be queried to determine if desired options are
291 supported by the implementation.
292 .sp

294 .sp
295 .TS
296 c c
297 l l .
298 \fBPOSIX Standard\fR    \fBFeature Test Macros\fR
299 _
300 POSIX.1-1990    \fB_POSIX_SOURCE\fR
301 _
302 T{
303 POSIX.1-1990 and POSIX.2-1992  C-Language Bindings Option
304 T}      \fB_POSIX_SOURCE\fR and \fB_POSIX_C_SOURCE=2\fR
305 POSIX.1b-1993   \fB_POSIX_C_SOURCE=199309L\fR
306 _
307 POSIX.1c-1996   \fB_POSIX_C_SOURCE=199506L\fR
308 _
309 POSIX.1-2001    \fB_POSIX_C_SOURCE=200112L\fR
310 .TE

312 .SS "SVID3"
313 .sp
314 .LP
315 The SVID3 specification does not specify any feature test macros to indicate
316 that an application is written to meet SVID3 requirements.  The SVID3
317 specification was written before the C standard was completed.
318 .SS "X/Open CAE"
319 .sp
320 .LP
321 To build or compile an application that conforms to one of the X/Open CAE
322 specifications, use the following guidelines. Applications need not set the
323 POSIX feature test macros if they require both CAE and POSIX functionality.
324 .sp
```



```
325 .ne 2
326 .na
327 \fBXPG3\fR
328 .ad
329 .RS 16n
330 The application must define \fB_XOPEN_SOURCE\fR. If \fB_XOPEN_SOURCE\fR is
331 defined with a value, the value must be less than 500.
332 .RE

334 .sp
335 .ne 2
336 .na
337 \fBXPG4\fR
338 .ad
339 .RS 16n
340 The application must define \fB_XOPEN_SOURCE\fR and set \fB_XOPEN_VERSION=4\fR.
341 If \fB_XOPEN_SOURCE\fR is defined with a value, the value must be less than
342 500.
343 .RE

345 .sp
346 .ne 2
347 .na
348 \fBSUS (XPG4v2)\fR
349 .ad
350 .RS 16n
351 The application must define \fB_XOPEN_SOURCE\fR and set
352 \fB_XOPEN_SOURCE_EXTENDED=1\fR. If \fB_XOPEN_SOURCE\fR is defined with a value,
353 the value must be less than 500.
354 .RE

356 .sp
357 .ne 2
358 .na
359 \fBSUSv2\fR
360 .ad
361 .RS 16n
362 The application must define \fB_XOPEN_SOURCE=500\fR.
363 .RE

365 .sp
366 .ne 2
367 .na
368 \fBSUSv3\fR
369 .ad
370 .RS 16n
371 The application must define \fB_XOPEN_SOURCE=600\fR.
372 .RE

374 .SS "Compilation"
375 .sp
376 .LP
377 A POSIX.1 (1988-1996)-, XPG4-, SUS-, or SUSv2-conforming implementation must
378 include an ANSI X3.159-1989 (ANSI C Language) standard-conforming compilation
379 system and the \fBcc\fR and \fBc89\fR utilities. A POSIX.1-2001- or
380 SUSv3-conforming implementation must include an ISO/IEC 99899:1999 (1999 ISO C
381 Language) standard-conforming compilation system and the \fBc99\fR utility.
382 Solaris 10 was tested with the \fBcc\fR, \fBc89\fR, and \fBc99\fR utilities and
383 the compilation environment provided by \fBSun Studio C Compiler 5.6\fR.
384 .sp
385 .LP
386 When \fBcc\fR is used to link applications, \fB/usr/lib/values-xpg4.o\fR must
387 be specified on any link/load command line, unless the application is
388 POSIX.1-2001- or SUSv3-conforming, in which case \fB/usr/lib/values-xpg6.o\fR
389 must be specified on any link/load compile line. The preferred way to build
390 applications, however, is described in the table below.
```

```
.sp
.LP
An XNS4- or XNS5-conforming application must include \fB-l\fR \fBXNS\fR on any
link/load command line in addition to defining the feature test macros
specified for SUS or SUSv2, respectively.
.sp
.LP
If the compiler suppports the \fBredefine_extname\fR pragma feature (the \fBSun
Studio C Compiler 5.6\fR compilers define the macro
\fB__PRAGMA_REDEFINE_EXTNAME\fR to indicate that it supports this feature),
then the standard headers use \fB#pragma\fR \fBredefine_extname\fR directives
to properly map function names onto library entry point names. This mapping
provides full support for ISO C, POSIX, and X/Open namespace reservations.
.sp
.LP
If this pragma feature is not supported by the compiler, the headers use the
\fB#define\fR directive to map internal function names onto appropriate library
entry point names. In this instance, applications should avoid using the
explicit 64-bit file offset symbols listed on the \fBlf64\fR(5) manual page,
since these names are used by the implementation to name the alternative entry
points.
.sp
.LP
When using \fBSun Studio C Compiler 5.6\fR compilers, applications conforming
to the specifications listed above should be compiled using the utilities and
flags indicated in the following table:
.sp
.in +2
.nf
Specification            Compiler/Flags         Feature Test Macros
_____________________________________________________________________
1989 ANSI C and 1990 ISO C    c89                none
_____________________________________________________________________
1999 ISO C                    c99                none
_____________________________________________________________________
SVID3                         cc -Xt -xc99=none  none
_____________________________________________________________________
POSIX.1-1990                  c89                _POSIX_SOURCE
_____________________________________________________________________
POSIX.1-1990 and POSIX.2-1992 c89                _POSIX_SOURCE  and
  C-Language Bindings Option                     POSIX_C_SOURCE=2
_____________________________________________________________________
POSIX.1b-1993                 c89                _POSIX_C_SOURCE=199309L
_____________________________________________________________________
POSIX.1c-1996                 c89                _POSIX_C_SOURCE=199506L
_____________________________________________________________________
POSIX.1-2001                  c99                _POSIX_C_SOURCE=200112L
_____________________________________________________________________
POSIX.1c-1996                 c89                _POSIX_C_SOURCE=199506L
_____________________________________________________________________
CAE XPG3                      cc -Xa -xc99=none  _XOPEN_SOURCE
_____________________________________________________________________
CAE XPG4                      c89                _XOPEN_SOURCE and
                                                 _XOPEN_VERSION=4
_____________________________________________________________________
SUS (CAE XPG4v2)              c89                _XOPEN_SOURCE and
  (includes XNS4)                                 _XOPEN_SOURCE_EXTENDED=1
_____________________________________________________________________
SUSv2 (includes XNS5)         c89                _XOPEN_SOURCE=500
_____________________________________________________________________
SUSv3                         c99                _XOPEN_SOURCE=600
.fi
.in -2
.sp

.sp
```



```
.LP
For platforms supporting the LP64 (64-bit) programming environment,
SUSv2-conforming LP64 applications using XNS5 library calls should be built
with command lines of the form:
.sp
.in +2
.nf
c89 $(getconf XBS5_LP64_OFF64_CFLAGS) -D_XOPEN_SOURCE=500 \e
    $(getconf XBS5_LP64_OFF64_LDFLAGS) foo.c -o foo \e
    $(getconf XBS5_LP64_OFF64_LIBS) -lxnet
.fi
.in -2

.sp
.LP
Similar SUSv3-conforming LP64 applications should be built with command lines
of the form:
.sp
.in +2
.nf
c99 $(getconf POSIX_V6_LP64_OFF64_CFLAGS) -D_XOPEN_SOURCE=600 \e
    $(getconf POSIX_V6_LP64_OFF64_LDFLAGS) foo.c -o foo \e
    $(getconf POSIX_V6_LP64_OFF64_LIBS) -lxnet
.fi
.in -2

.SS "SUSv3"
.sp
.ne 2
.na
\fB\fBc99\fR\fR
.ad
.RS 28n
\fB_XOPEN_SOURCE=600\fR
.RE

.SH SEE ALSO
.sp
.LP
\fBcsh\fR(1), \fBksh\fR(1), \fBsh\fR(1), \fBexec\fR(2), \fBsysconf\fR(3C),
\fBsystem\fR(3C), \fBenviron\fR(5), \fBlf64\fR(5)
```