

new/usr/src/Makefile.master

1

```
*****
36595 Fri Dec 7 21:21:44 2018
new/usr/src/Makefile.master
10051 cw(1) shouldn't have to translate -g
*****
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) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright (c) 2012 by Delphix. All rights reserved.
25 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
26 # Copyright 2015, OmniTI Computer Consulting, Inc. All rights reserved.
27 # Copyright 2015 Gary Mills
28 # Copyright 2015 Igor Kozhukhov <ikozhukhov@gmail.com>
29 # Copyright 2016 Toomas Soome <tsoome@me.com>
30 # Copyright 2018 OmniOS Community Edition (OmniOSce) Association.
31 #
33 #
34 # Makefile.master, global definitions for system source
35 #
36 ROOT= /proto
38 #
39 # Adjunct root, containing an additional proto area to be used for headers
40 # and libraries.
41 #
42 ADJUNCT_PROTO=
44 #
45 # Adjunct for building things that run on the build machine.
46 #
47 NATIVE_ADJUNCT= /usr
49 #
50 # RELEASE_BUILD should be cleared for final release builds.
51 # NOT_RELEASE_BUILD is exactly what the name implies.
52 #
53 # __GNUC toggles the building of ON components using gcc and related tools.
54 # Normally set to '#', set it to '' to do gcc build.
55 #
56 # The declaration POUND_SIGN is always '#'. This is needed to get around the
57 # make feature that '#' is always a comment delimiter, even when escaped or
58 # quoted. We use this macro expansion method to get POUND_SIGN rather than
59 # always breaking out a shell because the general case can cause a noticeable
60 # slowdown in build times when so many Makefiles include Makefile.master.
61 #
```

new/usr/src/Makefile.master

2

```
62 # While the majority of users are expected to override the setting below
63 # with an env file (via nightly or bldenv), if you aren't building that way
64 # (ie, you're using "ws" or some other bootstrapping method) then you need
65 # this definition in order to avoid the subshell invocation mentioned above.
66 #
68 PRE_POUND= pre\#
69 POUND_SIGN= $(PRE_POUND:pre\%=%)
71 NOT_RELEASE_BUILD=
72 RELEASE_BUILD= $(POUND_SIGN)
73 $(RELEASE_BUILD)NOT_RELEASE_BUILD= $(POUND_SIGN)
74 PATCH_BUILD= $(POUND_SIGN)
76 # SPARC_BLD is '#' for an Intel build.
77 # INTEL_BLD is '#' for a Sparc build.
78 SPARC_BLD_1= $(MACH:i386=$(POUND_SIGN))
79 SPARC_BLD= $(SPARC_BLD_1:sparc=)
80 INTEL_BLD_1= $(MACH:sparc=$(POUND_SIGN))
81 INTEL_BLD= $(INTEL_BLD_1:i386=)
83 # The variables below control the compilers used during the build.
84 # There are a number of permutations.
85 #
86 # __GNUC and __SUNC control (and indicate) the primary compiler. Whichever
87 # one is not POUND_SIGN is the primary, with the other as the shadow. They
88 # may also be used to control entirely compiler-specific Makefile assignments.
89 # __GNUC and GCC are the default.
90 #
91 # __GNUC64 indicates that the 64bit build should use the GNU C compiler.
92 # There is no Sun C analogue.
93 #
94 # The following version-specific options are operative regardless of which
95 # compiler is primary, and control the versions of the given compilers to be
96 # used. They also allow compiler-version specific Makefile fragments.
97 #
99 __SUNC= $(POUND_SIGN)
100 $(__SUNC)__GNUC= $(POUND_SIGN)
101 __GNUC64= $(__GNUC)
103 # Allow build-time "configuration" to enable or disable some things.
104 # The default is POUND_SIGN, meaning "not enabled". If the environment
105 # passes in an override like ENABLE_SMB_PRINTING= (empty) that will
106 # uncomment things in the lower Makefiles to enable the feature.
107 ENABLE_SMB_PRINTING= $(POUND_SIGN)
109 # CLOSED is the root of the tree that contains source which isn't released
110 # as open source
111 CLOSED= $(SRC)/../closed
113 # BUILD_TOOLS is the root of all tools including compilers.
114 # ONBLD_TOOLS is the root of all the tools that are part of SUNWonbld.
116 BUILD_TOOLS= /ws/onnv-tools
117 ONBLD_TOOLS= $(BUILD_TOOLS)/onbld
119 # define runtime JAVA_HOME, primarily for cmd/pools/poold
120 JAVA_HOME= /usr/java
121 # define buildtime JAVA_ROOT
122 JAVA_ROOT= /usr/java
123 # Build uses java7 by default. Pass one the variables below set to empty
124 # string in the environment to override.
125 BLD_JAVA_6= $(POUND_SIGN)
126 BLD_JAVA_8= $(POUND_SIGN)
```

new/usr/src/Makefile.master

3

```

128 GNUC_ROOT=      /opt/gcc/4.4.4
129 GCCLIBDIR=      $(GNUC_ROOT)/lib
130 GCCLIBDIR64=    $(GNUC_ROOT)/lib/$(MACH64)

132 DOCBOOK_XSL_ROOT=      /usr/share/sgml/docbook/xsl-stylesheets

134 RPCGEN=         /usr/bin/rpcgen
135 STABS=          $(ONBLD_TOOLS)/bin/$(MACH)/stabs
136 ELFXTRACT=      $(ONBLD_TOOLS)/bin/$(MACH)/elfextract
137 MBH_PATCH=     $(ONBLD_TOOLS)/bin/$(MACH)/mbh_patch
138 BTXILD=         $(ONBLD_TOOLS)/bin/$(MACH)/btxild
139 VTFONTCVT=     $(ONBLD_TOOLS)/bin/$(MACH)/vtfontcvt
140 # echo(1) and true(1) are specified without absolute paths, so that the shell
141 # spawned by make(1) may use the built-in versions. This is minimally
142 # problematic, as the shell spawned by make(1) is known and under control, the
143 # only risk being if the shell falls back to $PATH.
144 #
145 # We specifically want an echo(1) that does interpolation of escape sequences,
146 # which ksh93, /bin/sh, and bash will all provide.
147 ECHO=           echo
148 TRUE=           true
149 INS=            $(ONBLD_TOOLS)/bin/$(MACH)/install
150 SYMLINK=       /usr/bin/ln -s
151 LN=            /usr/bin/ln
152 MKDIR=         /usr/bin/mkdir
153 CHMOD=         /usr/bin/chmod
154 MV=            /usr/bin/mv -f
155 RM=            /usr/bin/rm -f
156 CUT=           /usr/bin/cut
157 NM=            /usr/ccs/bin/nm
158 DIFF=         /usr/bin/diff
159 GREP=          /usr/bin/grep
160 EGREP=         /usr/bin/egrep
161 ELFWRAP=       /usr/bin/elfwrap
162 KSH93=        /usr/bin/ksh93
163 SED=           /usr/bin/sed
164 AWK=           /usr/bin/nawk
165 CP=            /usr/bin/cp -f
166 MCS=           /usr/ccs/bin/mcs
167 CAT=           /usr/bin/cat
168 ELFDUMP=       /usr/ccs/bin/elfdump
169 M4=            /usr/bin/m4
170 STRIP=         /usr/ccs/bin/strip
171 LEX=           /usr/ccs/bin/lex
172 FLEX=          /usr/bin/flex
173 YACC=          /usr/ccs/bin/yacc
174 CPP=           /usr/lib/cpp
175 ANSI_CPP=     $(GNUC_ROOT)/bin/cpp
176 JAVAC=         $(JAVA_ROOT)/bin/javac
177 JAVAHA=       $(JAVA_ROOT)/bin/java
178 JAVADOC=      $(JAVA_ROOT)/bin/javadoc
179 RMIC=          $(JAVA_ROOT)/bin/rmic
180 JAR=           $(JAVA_ROOT)/bin/jar
181 CTFCONVERT=   $(ONBLD_TOOLS)/bin/$(MACH)/ctfconvert
182 CTFMERGE=     $(ONBLD_TOOLS)/bin/$(MACH)/ctfmerge
183 CTFSTABS=     $(ONBLD_TOOLS)/bin/$(MACH)/ctfstabs
184 CTFSTRIP=     $(ONBLD_TOOLS)/bin/$(MACH)/ctfstrip
185 NDRGEN=       $(ONBLD_TOOLS)/bin/$(MACH)/ndrgen
186 GENOFFSETS=  $(ONBLD_TOOLS)/bin/genoffsets
187 XREF=         $(ONBLD_TOOLS)/bin/xref
188 FIND=         /usr/bin/find
189 PERL=         /usr/bin/perl
190 PERL_VERSION= 5.10.0
191 PERL_PKGVERS= -510
192 PERL_ARCH =   i86pc-solaris-64int
193 $(SPARC_BLD)PERL_ARCH = sun4-solaris-64int

```

new/usr/src/Makefile.master

4

```

194 PYTHON_VERSION= 2.7
195 PYTHON_PKGVERS= -27
196 PYTHON_SUFFIX=
197 PYTHON=        /usr/bin/python$(PYTHON_VERSION)
198 PYTHON3_VERSION= 3.5
199 PYTHON3_PKGVERS= -35
200 PYTHON3_SUFFIX= m
201 PYTHON3=       /usr/bin/python$(PYTHON3_VERSION)
202 SORT=          /usr/bin/sort
203 TOUCH=         /usr/bin/touch
204 WC=            /usr/bin/wc
205 XARGS=         /usr/bin/xargs
206 ELFEDIT=       /usr/bin/elfedit
207 DTRACE=        /usr/sbin/dtrace -xnolib
208 UNIQ=          /usr/bin/uniq
209 TAR=           /usr/bin/tar
210 ASTBINDIR=     /usr/ast/bin
211 MSGCC=         $(ASTBINDIR)/msgcc
212 MSGFMT=        /usr/bin/msgfmt -s
213 LCDEF=         $(ONBLD_TOOLS)/bin/$(MACH)/localedef
214 TIC=           $(ONBLD_TOOLS)/bin/$(MACH)/tic
215 ZIC=           $(ONBLD_TOOLS)/bin/$(MACH)/zic
216 OPENSSL=       /usr/bin/openssl

218 FILEMODE=     644
219 DIRMODE=      755

221 # Declare that nothing should be built in parallel.
222 # Individual Makefiles can use the .PARALLEL target to declare otherwise.
223 .NO_PARALLEL:

225 # For stylistic checks
226 #
227 # Note that the X and C checks are not used at this time and may need
228 # modification when they are actually used.
229 #
230 CSTYLE=       $(ONBLD_TOOLS)/bin/cstyle
231 CSTYLE_TAIL=
232 HDRCHK=       $(ONBLD_TOOLS)/bin/hdrchk
233 HDRCHK_TAIL=
234 JSTYLE=       $(ONBLD_TOOLS)/bin/jstyle

236 DOT_H_CHECK=  \
237     @$(ECHO) "checking $<"; $(CSTYLE) $< $(CSTYLE_TAIL); \
238     $(HDRCHK) $< $(HDRCHK_TAIL)

240 DOT_X_CHECK=  \
241     @$(ECHO) "checking $<"; $(RPCGEN) -C -h $< | $(CSTYLE) $(CSTYLE_TAIL); \
242     $(RPCGEN) -C -h $< | $(HDRCHK) $< $(HDRCHK_TAIL)

244 DOT_C_CHECK=  \
245     @$(ECHO) "checking $<"; $(CSTYLE) $< $(CSTYLE_TAIL)

247 MANIFEST_CHECK= \
248     @$(ECHO) "checking $<"; \
249     SVCCFG_DTD=$(SRC)/cmd/svc/dtd/service_bundle.dtd.1 \
250     SVCCFG_REPOSITORY=$(SRC)/cmd/svc/seed/global.db \
251     SVCCFG_CONFIGD_PATH=$(SRC)/cmd/svc/configd/svc.configd-native \
252     $(SRC)/cmd/svc/svccfg/svccfg-native validate $<

254 INS.file=     $(RM) $@; $(INS) -s -m $(FILEMODE) -f $@D $<
255 INS.dir=      $(INS) -s -d -m $(DIRMODE) $@
256 # installs and renames at once
257 #
258 INS.rename=   $(INS.file); $(MV) $@D/$(<F) $@

```

```

260 # install a link
261 INSLINKTARGET= $<
262 INS.link= $(RM) $@; $(LN) $(INSLINKTARGET) $@
263 INS.symlink= $(RM) $@; $(SYMLINK) $(INSLINKTARGET) $@

265 #
266 # Python bakes the mtime of the .py file into the compiled .pyc and
267 # rebuilds if the baked-in mtime != the mtime of the source file
268 # (rather than only if it's less than), thus when installing python
269 # files we must make certain to not adjust the mtime of the source
270 # (.py) file.
271 #
272 INS.pyfile= $(RM) $@; $(SED) -e "1s:^\#!@PYTHON@:\#!$(PYTHON):" < $< > $@; $

274 # MACH must be set in the shell environment per uname -p on the build host
275 # More specific architecture variables should be set in lower makefiles.
276 #
277 # MACH64 is derived from MACH, and BUILD64 is set to '#' for
278 # architectures on which we do not build 64-bit versions.
279 # (There are no such architectures at the moment.)
280 #
281 # Set BUILD64=# in the environment to disable 64-bit amd64
282 # builds on i386 machines.

284 MACH64_1= $(MACH:sparc=sparcv9)
285 MACH64= $(MACH64_1:i386=amd64)

287 MACH32_1= $(MACH:sparc=sparcv7)
288 MACH32= $(MACH32_1:i386=i86)

290 sparc_BUILD64=
291 i386_BUILD64=
292 BUILD64= $($MACH)_BUILD64)

294 #
295 # C compiler mode. Future compilers may change the default on us,
296 # so force extended ANSI mode globally. Lower level makefiles can
297 # override this by setting CCMODE.
298 #
299 CCMODE= -Xa
300 CCMODE64= -Xa

302 #
303 # C compiler verbose mode. This is so we can enable it globally,
304 # but turn it off in the lower level makefiles of things we cannot
305 # (or aren't going to) fix.
306 #
307 CCVERBOSE= -v

309 # set this to the secret flag "-Wc,-Qiselect-v9abiwarn=1" to get warnings
310 # from the compiler about places the -xarch=v9 may differ from -xarch=v9c.
311 V9ABIWARN=

313 # set this to the secret flag "-Wc,-Qiselect-regsym=0" to disable register
314 # symbols (used to detect conflicts between objects that use global registers)
315 # we disable this now for safety, and because genunix doesn't link with
316 # this feature (the v9 default) enabled.
317 #
318 # REGSYM is separate since the C++ driver syntax is different.
319 CCREGSYM= -Wc,-Qiselect-regsym=0
320 CCCREGSYM= -Qoption cg -Qiselect-regsym=0

322 # Prevent the removal of static symbols by the SPARC code generator (cg).
323 # The x86 code generator (ube) does not remove such symbols and as such
324 # using this workaround is not applicable for x86.
325 #

```

```

326 CCSTATICSYM= -Wc,-Qassembler-ounrefsym=0
327 #
328 # generate 32-bit addresses in the v9 kernel. Saves memory.
329 CCABS32= -Wc,-xcode=abs32
330 #
331 # generate v9 code which tolerates callers using the v7 ABI, for the sake of
332 # system calls.
333 CC32BITCALLERS= -_gcc=-massume-32bit-callers

335 # GCC, especially, is increasingly beginning to auto-inline functions and
336 # sadly does so separately not under the general -fno-inline-functions
337 # Additionally, we wish to prevent optimisations which cause GCC to clone
338 # functions -- in particular, these may cause unhelpful symbols to be
339 # emitted instead of function names
340 CCNOAUTOINLINE= \
341     -_gcc=-fno-inline-small-functions \
342     -_gcc=-fno-inline-functions-called-once \
343     -_gcc=-fno-ipa-cp \
344     -_gcc7=-fno-ipa-icf \
345     -_gcc8=-fno-ipa-icf \
346     -_gcc7=-fno-clone-functions \
347     -_gcc8=-fno-clone-functions

349 # GCC may put functions in different named sub-sections of .text based on
350 # their presumed calling frequency. At least in the kernel, where we actually
351 # deliver relocatable objects, we don't want this to happen.
352 #
353 # Since at present we don't benefit from this even in userland, we disable it gl
354 # but the application of this may move into usr/src/uts/ in future.
355 CCNOREORDER= \
356     -_gcc7=-fno-reorder-functions \
357     -_gcc8=-fno-reorder-functions

359 # One optimization the compiler might perform is to turn this:
360 #     #pragma weak foo
361 #     extern int foo;
362 #     if (&foo)
363 #         foo = 5;
364 # into
365 #     foo = 5;
366 # Since we do some of this (foo might be referenced in common kernel code
367 # but provided only for some cpu modules or platforms), we disable this
368 # optimization.
369 #
370 sparc_CCUNBOUND = -Wd,-xsafe=unboundsym
371 i386_CCUNBOUND =
372 CCUNBOUND = $($MACH)_CCUNBOUND)

374 #
375 # compiler '-xarch' flag. This is here to centralize it and make it
376 # overridable for testing.
377 sparc_XARCH= -m32
378 sparcv9_XARCH= -m64
379 i386_XARCH= -m32
380 amd64_XARCH= -m64 -Ui386 -U__i386

382 # assembler '-xarch' flag. Different from compiler '-xarch' flag.
383 sparc_AS_XARCH= -xarch=v8plus
384 sparcv9_AS_XARCH= -xarch=v9
385 i386_AS_XARCH=
386 amd64_AS_XARCH= -xarch=amd64 -P -Ui386 -U__i386

388 #
389 # These flags define what we need to be 'standalone' i.e. -not- part
390 # of the rather more cosy userland environment. This basically means
391 # the kernel.

```

```

392 #
393 # XX64 future versions of gcc will make -mcmmodel=kernel imply -mno-red-zone
394 #
395 sparc_STAND_FLAGS=      -_gcc=-ffreestanding
396 sparcv9_STAND_FLAGS=    -_gcc=-ffreestanding
397 # Disabling MMX also disables 3DNow, disabling SSE also disables all later
398 # additions to SSE (SSE2, AVX ,etc.)
399 NO_SIMD=                -_gcc=-mno-mmx -_gcc=-mno-sse
400 i386_STAND_FLAGS=       -_gcc=-ffreestanding $(NO_SIMD)
401 amd64_STAND_FLAGS=      -xmodel=kernel $(NO_SIMD)

403 SAVEARGS=              -Wu,-save_args
404 amd64_STAND_FLAGS      += $(SAVEARGS)

406 STAND_FLAGS_32 = $($(MACH)_STAND_FLAGS)
407 STAND_FLAGS_64 = $($(MACH64)_STAND_FLAGS)

409 #
410 # disable the incremental linker
411 ILDOFF=                 -xildoff
412 #
413 XFFLAG=                 -xF=%all
414 XESS=                   -xs
415 XSTRCONST=             -xstrconst

417 #
418 # turn warnings into errors (C)
419 CERRWARN = -errtags=yes -errwarn=%all
420 CERRWARN += -erroff=E_EMPTY_TRANSLATION_UNIT
421 CERRWARN += -erroff=E_STATEMENT_NOT_REACHED

423 CERRWARN += -_gcc=-Wno-missing-braces
424 CERRWARN += -_gcc=-Wno-sign-compare
425 CERRWARN += -_gcc=-Wno-unknown-pragmas
426 CERRWARN += -_gcc=-Wno-unused-parameter
427 CERRWARN += -_gcc=-Wno-missing-field-initializers

429 # Unfortunately, this option can misfire very easily and unfixably.
430 CERRWARN += -_gcc=-Wno-array-bounds

432 # DEBUG v. -nd make for frequent unused variables, empty conditions, etc. in
433 # -nd builds
434 $(RELEASE_BUILD)CERRWARN += -_gcc=-Wno-unused
435 $(RELEASE_BUILD)CERRWARN += -_gcc=-Wno-empty-body

437 #
438 # turn warnings into errors (C++)
439 CCERRWARN=              -xwe

441 # C standard. Keep Studio flags until we get rid of lint.
442 CSTD_GNU89=             -xc99=%none
443 CSTD_GNU99=             -xc99=%all
444 CSTD=                   $(CSTD_GNU89)
445 C99LMODE=              $(CSTD:-xc99%=-Xc99%)

447 # In most places, assignments to these macros should be appended with +=
448 # (CPPFLAGS.first allows values to be prepended to CPPFLAGS).
449 sparc_CFLAGS=          $(sparc_XARCH) $(CCSTATICSYM)
450 sparcv9_CFLAGS=        $(sparcv9_XARCH) -dalign $(CCVERBOSE) $(V9ABIWARN) $(CCREGSYM) \
451                        $(CCSTATICSYM)
452 i386_CFLAGS=           $(i386_XARCH)
453 amd64_CFLAGS=          $(amd64_XARCH)

455 sparc_ASFLAGS=         $(sparc_AS_XARCH)
456 sparcv9_ASFLAGS=       $(sparcv9_AS_XARCH)
457 i386_ASFLAGS=          $(i386_AS_XARCH)

```

```

458 amd64_ASFLAGS=        $(amd64_AS_XARCH)

460 #
461 sparc_COPTFLAG=        -xO3
462 sparcv9_COPTFLAG=     -xO3
463 i386_COPTFLAG=        -O
464 amd64_COPTFLAG=       -xO3

466 COPTFLAG=             $($(MACH)_COPTFLAG)
467 COPTFLAG64=           $($(MACH64)_COPTFLAG)

469 # When -g is used, the compiler globalizes static objects
470 # (gives them a unique prefix). Disable that.
471 CNOGLOBAL= -W0,-noglobal

473 # Direct the Sun Studio compiler to use a static globalization prefix based on t
474 # name of the module rather than something unique. Otherwise, objects
475 # will not build deterministically, as subsequent compilations of identical
476 # source will yeild objects that always look different.
477 #
478 # In the same spirit, this will also remove the date from the N_OPT stab.
479 CGLOBALSTATIC= -W0,-xglobalstatic

481 # Sometimes we want all symbols and types in debugging information even
482 # if they aren't used.
483 CALLSYMS=              -W0,-xdbggen=no%usedonly

485 #
486 # Default debug format for Sun Studio 11 is dwarf, so force it to
487 # generate stabs.
488 #
489 DEBUGFORMAT= -_cc=-xdebugformat=stabs -_gcc=-gdwarf-2

491 #
492 # Ask the compiler to include debugging information
493 #
494 CCGDEBUG= -g $(DEBUGFORMAT)
489 DEBUGFORMAT= -xdebugFormat=stabs

496 #
497 # Flags used to build in debug mode for ctf generation. Bugs in the Devpro
498 # compilers currently prevent us from building with cc-emitted DWARF.
499 #
500 CTF_FLAGS_sparc = $(CCGDEBUG) -Wc,-Qiselect-T1 $(CSTD) $(CNOGLOBAL)
501 CTF_FLAGS_i386 = $(CCGDEBUG) $(CSTD) $(CNOGLOBAL)
495 CTF_FLAGS_sparc = -g -Wc,-Qiselect-T1 $(CSTD) $(CNOGLOBAL) $(CDWARFSTR)
496 CTF_FLAGS_i386 = -g $(CSTD) $(CNOGLOBAL) $(CDWARFSTR)

503 CTF_FLAGS_sparcv9      = $(CTF_FLAGS_sparc)
504 CTF_FLAGS_amd64        = $(CTF_FLAGS_i386)

506 # Sun Studio produces broken userland code when saving arguments.
507 $(__GNUC)CTF_FLAGS_amd64 += $(SAVEARGS)

509 CTF_FLAGS_32 = $(CTF_FLAGS_$(MACH))
510 CTF_FLAGS_64 = $(CTF_FLAGS_$(MACH64))
504 CTF_FLAGS_32 = $(CTF_FLAGS_$(MACH)) $(DEBUGFORMAT)
505 CTF_FLAGS_64 = $(CTF_FLAGS_$(MACH64)) $(DEBUGFORMAT)
511 CTF_FLAGS = $(CTF_FLAGS_32)

513 #
514 # Flags used with genoffsets
515 #
516 GOFLAGS = $(CALLSYMS)
511 GOFLAGS = $(CALLSYMS) $(CDWARFSTR)

```

```

518 OFFSETS_CREATE = $(GENOFFSETS) -s $(CTFSTABS) -r $(CTFCONVERT) \
519     $(CW) --noecho $(CW_CC_COMPILERS) -- $(GOFLAGS) $(CFLAGS) $(CPPFLAGS)

521 OFFSETS_CREATE64 = $(GENOFFSETS) -s $(CTFSTABS) -r $(CTFCONVERT) \
522     $(CW) --noecho $(CW_CC_COMPILERS) -- $(GOFLAGS) $(CFLAGS64) $(CPPFLAGS)

524 #
525 # tradeoff time for space (smaller is better)
526 #
527 sparc_SPACEFLAG      = -xspace -W0,-Lt
528 sparcv9_SPACEFLAG   = -xspace -W0,-Lt
529 i386_SPACEFLAG       = -xspace
530 amd64_SPACEFLAG      =

532 SPACEFLAG           = $(MACH)_SPACEFLAG
533 SPACEFLAG64         = $(MACH64)_SPACEFLAG

535 #
536 # The Sun Studio 11 compiler has changed the behaviour of integer
537 # wrap arounds and so a flag is needed to use the legacy behaviour
538 # (without this flag panics/hangs could be exposed within the source).
539 #
540 sparc_IROPTFLAG      = -W2,-xwrap_int
541 sparcv9_IROPTFLAG    = -W2,-xwrap_int
542 i386_IROPTFLAG       =
543 amd64_IROPTFLAG      =

545 IROPTFLAG           = $(MACH)_IROPTFLAG
546 IROPTFLAG64         = $(MACH64)_IROPTFLAG

548 sparc_XREGSFLAG      = -xregs=no%appl
549 sparcv9_XREGSFLAG    = -xregs=no%appl
550 i386_XREGSFLAG       =
551 amd64_XREGSFLAG      =

553 XREGSFLAG           = $(MACH)_XREGSFLAG
554 XREGSFLAG64         = $(MACH64)_XREGSFLAG

556 # dmake SOURCEDEBUG=yes ... enables source-level debugging information, and
557 # avoids stripping it.
558 SOURCEDEBUG          = $(POUND_SIGN)
559 SRCDBGBLD            = $(SOURCEDEBUG:yes=)

561 #
562 # These variables are intended ONLY for use by developers to safely pass extra
563 # flags to the compilers without unintentionally overriding Makefile-set
564 # flags. They should NEVER be set to any value in a Makefile.
565 #
566 # They come last in the associated FLAGS variable such that they can
567 # explicitly override things if necessary, there are gaps in this, but it's
568 # the best we can manage.
569 #
570 CUSERFLAGS           =
571 CUSERFLAGS64         = $(CUSERFLAGS)
572 CCUSERFLAGS          =
573 CCUSERFLAGS64        = $(CCUSERFLAGS)

575 CSOURCEDEBUGFLAGS   =
576 CCSOURCEDEBUGFLAGS  =
577 $(SRCDBGBLD)CSOURCEDEBUGFLAGS = $(CCGDEBUG) -xs
578 $(SRCDBGBLD)CCSOURCEDEBUGFLAGS = $(CCGDEBUG) -xs
579 $(SRCDBGBLD)CSOURCEDEBUGFLAGS = -g -xs
580 $(SRCDBGBLD)CCSOURCEDEBUGFLAGS = -g -xs

580 CFLAGS=              $(COPTFLAG) $(MACH)_CFLAGS $(SPACEFLAG) $(CCMODE) \
581     $(ILDOFF) $(CERRWARN) $(CSTD) $(CCUNBOUND) $(IROPTFLAG) \

```

```

582     $(CGLOBALSTATIC) $(CCNOAUTOINLINE) $(CCNOREORDER) \
583     $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS)
584 CFLAGS64=            $(COPTFLAG64) $(MACH64)_CFLAGS $(SPACEFLAG64) $(CCMODE64) \
585     $(ILDOFF) $(CERRWARN) $(CSTD) $(CCUNBOUND) $(IROPTFLAG64) \
586     $(CGLOBALSTATIC) $(CCNOAUTOINLINE) $(CCNOREORDER) \
587     $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS64)
588 #
589 # Flags that are used to build parts of the code that are subsequently
590 # run on the build machine (also known as the NATIVE_BUILD).
591 #
592 NATIVE_CFLAGS=       $(COPTFLAG) $(NATIVE_MACH)_CFLAGS $(CCMODE) \
593     $(ILDOFF) $(CERRWARN) $(CSTD) $(NATIVE_MACH)_CCUNBOUND) \
594     $(IROPTFLAG) $(CGLOBALSTATIC) $(CCNOAUTOINLINE) \
595     $(CCNOREORDER) $(CSOURCEDEBUGFLAGS) $(CUSERFLAGS)

597 DTEXTDOM=-DTEXT_DOMAIN="\$(TEXT_DOMAIN)" # For messaging.
598 DTS_ERRNO=-D_TS_ERRNO
599 CPPFLAGS.first= # Please keep empty. Only lower makefiles should set this.
600 CPPFLAGS.master=$(DTEXTDOM) $(DTS_ERRNO) \
601     $(ENVCPPFLAGS1) $(ENVCPPFLAGS2) $(ENVCPPFLAGS3) $(ENVCPPFLAGS4) \
602     $(ADJUNCT_PROTO:=-I%/usr/include)
603 CPPFLAGS.native=$(ENVCPPFLAGS1) $(ENVCPPFLAGS2) $(ENVCPPFLAGS3) \
604     $(ENVCPPFLAGS4) -I$(NATIVE_ADJUNCT)/include
605 CPPFLAGS=            $(CPPFLAGS.first) $(CPPFLAGS.master)
606 AS_CPPFLAGS=         $(CPPFLAGS.first) $(CPPFLAGS.master)
607 JAVAFLAGS=          -source 1.6 -target 1.6 -Xlint:deprecation,-options

609 #
610 # For source message catalogue
611 #
612 .SUFFIXES: $(SUFFIXES) .i .po
613 MSGROOT= $(ROOT)/catalog
614 MSGDOMAIN= $(MSGROOT)/$(TEXT_DOMAIN)
615 MSGDOMAINPOFILE = $(MSGDOMAIN)/$(POFILE)
616 DCMMSGDOMAIN= $(MSGROOT)/LC_TIME/$(TEXT_DOMAIN)
617 DCMMSGDOMAINPOFILE = $(DCMSGDOMAIN)/$(DCFILE.dc:.dc=.po)

619 CLOBBERFILES += $(POFILE) $(POFILES)
620 COMPILE.cpp= $(CC) -E -C $(CFLAGS) $(CPPFLAGS)
621 XGETTEXT= /usr/bin/xgettext
622 XGETTEXTFLAGS= -c TRANSLATION_NOTE
623 GNUXGETTEXT= /usr/gnu/bin/xgettext
624 GNUXGETTEXTFLAGS= --add-comments=TRANSLATION_NOTE --keyword=_ \
625     --strict --no-location --omit-header
626 BUILD.po= $(XGETTEXT) $(XGETTEXTFLAGS) -d $(<F) $(<i ;\
627     $(RM)    @ ;\
628     $(SED)  "/"^domain/d" < $(<F).po > $@ ;\
629     $(RM)  $(<F).po $(<i

631 #
632 # This is overwritten by local Makefile when PROG is a list.
633 #
634 POFILE= $(PROG).po

636 sparc_CCFLAGS=       -cg92 -compat=4 \
637     -Qoption ccfe -messages=no%anachronism \
638     $(CCERRWARN)
639 sparcv9_CCFLAGS=    $(sparcv9_XARCH) -dalign -compat=5 \
640     -Qoption ccfe -messages=no%anachronism \
641     -Qoption ccfe -features=no%conststrings \
642     $(CCREGSYM) \
643     $(CCERRWARN)
644 i386_CCFLAGS=        -compat=4 \
645     -Qoption ccfe -messages=no%anachronism \
646     -Qoption ccfe -features=no%conststrings \
647     $(CCERRWARN)

```

```

648 amd64_CCFLAGS=      $(amd64_XARCH) -compat=5 \
649                   -Qoption ccfe -messages=no%anachronism \
650                   -Qoption ccfe -features=no%conststrings \
651                   $(CCERRWARN)

653 sparc_CCOPTFLAG=    -O
654 sparcv9_CCOPTFLAG=  -O
655 i386_CCOPTFLAG=     -O
656 amd64_CCOPTFLAG=    -O

658 CCOPTFLAG=          $(MACH)_CCOPTFLAG
659 CCOPTFLAG64=        $(MACH64)_CCOPTFLAG
660 CCFLAGS=            $(CCOPTFLAG) $(MACH)_CCFLAGS $(CCSOURCEDEBUGFLAGS) \
661                   $(CCUSERFLAGS)
662 CCFLAGS64=          $(CCOPTFLAG64) $(MACH64)_CCFLAGS $(CCSOURCEDEBUGFLAGS) \
663                   $(CCUSERFLAGS64)

665 #
666 #
667 #
668 ELFWRAP_FLAGS =
669 ELFWRAP_FLAGS64 =   -64

671 #
672 # Various mapfiles that are used throughout the build, and delivered to
673 # /usr/lib/ld.
674 #
675 MAPFILE.NED_i386 =   $(SRC)/common/mapfiles/common/map.noexdata
676 MAPFILE.NED_sparc =
677 MAPFILE.NED =        $(MAPFILE.NED_$(MACH))
678 MAPFILE.PGA =        $(SRC)/common/mapfiles/common/map.pagealign
679 MAPFILE.NES =        $(SRC)/common/mapfiles/common/map.noexstk
680 MAPFILE.FLT =        $(SRC)/common/mapfiles/common/map.filter
681 MAPFILE.LEX =        $(SRC)/common/mapfiles/common/map.lex.yy

683 #
684 # Generated mapfiles that are compiler specific, and used throughout the
685 # build. These mapfiles are not delivered in /usr/lib/ld.
686 #
687 MAPFILE.NGB_sparc=   $(SRC)/common/mapfiles/gen/sparc_cc_map.noexglobs
688 $(__GNUC64)MAPFILE.NGB_sparc= \
689                   $(SRC)/common/mapfiles/gen/sparc_gcc_map.noexglobs
690 MAPFILE.NGB_sparcv9= $(SRC)/common/mapfiles/gen/sparcv9_cc_map.noexglobs
691 $(__GNUC64)MAPFILE.NGB_sparcv9= \
692                   $(SRC)/common/mapfiles/gen/sparcv9_gcc_map.noexglobs
693 MAPFILE.NGB_i386=    $(SRC)/common/mapfiles/gen/i386_cc_map.noexglobs
694 $(__GNUC64)MAPFILE.NGB_i386= \
695                   $(SRC)/common/mapfiles/gen/i386_gcc_map.noexglobs
696 MAPFILE.NGB_amd64=   $(SRC)/common/mapfiles/gen/amd64_cc_map.noexglobs
697 $(__GNUC64)MAPFILE.NGB_amd64= \
698                   $(SRC)/common/mapfiles/gen/amd64_gcc_map.noexglobs
699 MAPFILE.NGB =        $(MAPFILE.NGB_$(MACH))

701 #
702 # A generic interface mapfile name, used by various dynamic objects to define
703 # the interfaces and interposers the object must export.
704 #
705 MAPFILE.INT =        mapfile-intf

707 #
708 # LDLIBS32 and LDLIBS64 can be set in the environment to override the following
709 # assignments.
710 #
711 # These environment settings make sure that no libraries are searched outside
712 # of the local workspace proto area:
713 #     LDLIBS32=-YP,$ROOT/lib:$ROOT/usr/lib

```

```

714 #     LDLIBS64=-YP,$ROOT/lib:$MACH64:$ROOT/usr/lib:$MACH64
715 #
716 LDLIBS32 =          $(ENVLDLIBS1) $(ENVLDLIBS2) $(ENVLDLIBS3)
717 LDLIBS32 +=         $(ADJUNCT_PROTO:%=-L%/usr/lib -L%/lib)
718 LDLIBS.cmd =        $(LDLIBS32)
719 LDLIBS.lib =         $(LDLIBS32)

721 LDLIBS64 =          $(ENVLDLIBS1:%=%/$(MACH64)) \
722                   $(ENVLDLIBS2:%=%/$(MACH64)) \
723                   $(ENVLDLIBS3:%=%/$(MACH64))
724 LDLIBS64 +=         $(ADJUNCT_PROTO:%=-L%/usr/lib/$(MACH64) -L%/lib/$(MACH64))

726 #
727 # Define compilation macros.
728 #
729 COMPILE.c =          $(CC) $(CFLAGS) $(CPPFLAGS) -c
730 COMPILE64.c =        $(CC) $(CFLAGS64) $(CPPFLAGS) -c
731 COMPILE.cc =         $(CCC) $(CCFLAGS) $(CPPFLAGS) -c
732 COMPILE64.cc =       $(CCC) $(CCFLAGS64) $(CPPFLAGS) -c
733 COMPILE.s =          $(AS) $(ASFLAGS) $(AS_CPPFLAGS)
734 COMPILE64.s =        $(AS) $(ASFLAGS) $(MACH64)_AS_XARCH $(AS_CPPFLAGS)
735 COMPILE.d =          $(DTRACE) -G -32
736 COMPILE64.d =        $(DTRACE) -G -64
737 COMPILE.b =          $(ELFWRAP) $(ELFWRAP_FLAGS$(CLASS))
738 COMPILE64.b =        $(ELFWRAP) $(ELFWRAP_FLAGS$(CLASS))

740 CLASSPATH=
741 COMPILE.java =       $(JAVAC) $(JAVAFLAGS) -classpath $(CLASSPATH)

743 #
744 # Link time macros
745 #
746 CCNEEDED              = -lC
747 CCEXTNEEDED           = -lCrun -lCstd
748 $(__GNUC)CCNEEDED     = -L$(GCCLIBDIR) -lstc++ -lgcc_s
749 $(__GNUC)CCEXTNEEDED = $(CCNEEDED)

751 LINK.c =              $(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS)
752 LINK64.c =            $(CC) $(CFLAGS64) $(CPPFLAGS) $(LDFLAGS)
753 NORUNPATH=            -norunpath -nolib
754 LINK.cc =             $(CCC) $(CCFLAGS) $(CPPFLAGS) $(NORUNPATH) \
755                   $(LDFLAGS) $(CCNEEDED)
756 LINK64.cc =           $(CCC) $(CCFLAGS64) $(CPPFLAGS) $(NORUNPATH) \
757                   $(LDFLAGS) $(CCNEEDED)

759 #
760 # lint macros
761 #
762 # Note that the undefine of __PRAGMA_REDEFINE_EXTNAME can be removed once
763 # ON is built with a version of lint that has the fix for 4484186.
764 #
765 ALWAYS_LINT_DEFS =    -errtags=yes -s
766 ALWAYS_LINT_DEFS +=  -erroff=E_PTRDIFF_OVERFLOW
767 ALWAYS_LINT_DEFS +=  -erroff=E_ASSIGN_NARROW_CONV
768 ALWAYS_LINT_DEFS +=  -U__PRAGMA_REDEFINE_EXTNAME
769 ALWAYS_LINT_DEFS +=  $(C99LMODE)
770 ALWAYS_LINT_DEFS +=  -errsecurity=$(SECLEVEL)
771 ALWAYS_LINT_DEFS +=  -erroff=E_SEC_CREAT_WITHOUT_EXCL
772 ALWAYS_LINT_DEFS +=  -erroff=E_SEC_FORBIDDEN_WARN_CREAT
773 # XX64 -- really only needed for amd64 lint
774 ALWAYS_LINT_DEFS +=  -erroff=E_ASSIGN_INT_TO_SMALL_INT
775 ALWAYS_LINT_DEFS +=  -erroff=E_CAST_INT_CONST_TO_SMALL_INT
776 ALWAYS_LINT_DEFS +=  -erroff=E_CAST_INT_TO_SMALL_INT
777 ALWAYS_LINT_DEFS +=  -erroff=E_CAST_TO_PTR_FROM_INT
778 ALWAYS_LINT_DEFS +=  -erroff=E_COMP_INT_WITH_LARGE_INT
779 ALWAYS_LINT_DEFS +=  -erroff=E_INTEGRAL_CONST_EXP_EXPECTED

```

```

780 ALWAYS_LINT_DEFS += -erroff=E_PASS_INT_TO_SMALL_INT
781 ALWAYS_LINT_DEFS += -erroff=E_PTR_CONV_LOSES_BITS

783 # This forces lint to pick up note.h and sys/note.h from Devpro rather than
784 # from the proto area. The note.h that ON delivers would disable NOTE().
785 ONLY_LINT_DEFS = -I$(SPRO_VROOT)/prod/include/lint

787 SECLEVEL= core
788 LINT.c= $(LINT) $(ONLY_LINT_DEFS) $(LINTFLAGS) $(CPPFLAGS) \
789 $(ALWAYS_LINT_DEFS)
790 LINT64.c= $(LINT) $(ONLY_LINT_DEFS) $(LINTFLAGS64) $(CPPFLAGS) \
791 $(ALWAYS_LINT_DEFS)
792 LINT.s= $(LINT.c)

794 # For some future builds, NATIVE_MACH and MACH might be different.
795 # Therefore, NATIVE_MACH needs to be redefined in the
796 # environment as 'uname -p' to override this macro.
797 #
798 # For now at least, we cross-compile amd64 on i386 machines.
799 NATIVE_MACH= $(MACH:amd64=i386)

801 # Define native compilation macros
802 #

804 # Base directory where compilers are loaded.
805 # Defined here so it can be overridden by developer.
806 #
807 SPRO_ROOT= $(BUILD_TOOLS)/SUNwspro
808 SPRO_VROOT= $(SPRO_ROOT)/SS12
809 GNU_ROOT= /usr

811 $(__GNUC)PRIMARY_CC= gcc4,$(GNU_ROOT)/bin/gcc.gnu
812 $(__SUNC)PRIMARY_CC= studio12,$(SPRO_VROOT)/bin/cc,sun
813 $(__GNUC)PRIMARY_CCC= gcc4,$(GNU_ROOT)/bin/g++.gnu
814 $(__SUNC)PRIMARY_CCC= studio12,$(SPRO_VROOT)/bin/CC,sun

816 CW_CC_COMPILERS= $(PRIMARY_CC:%--primary %) $(SHADOW_CCS:%--shadow %)
817 CW_CCC_COMPILERS= $(PRIMARY_CCC:%--primary %) $(SHADOW_CCCS:%--shadow %)

820 # Till SS12ul formally becomes the NV CBE, LINT is hard
821 # coded to be picked up from the $SPRO_ROOT/sunstudio12.1/
822 # location. Impacted variables are sparc_LINT, sparcv9_LINT,
823 # i386_LINT, amd64_LINT.
824 # Reset them when SS12ul is rolled out.
825 #

827 # Specify platform compiler versions for languages
828 # that we use (currently only c and c++).
829 #
830 CW= $(ONBLD_TOOLS)/bin/$(MACH)/cw

832 BUILD_CC= $(CW) $(CW_CC_COMPILERS) --
833 BUILD_CCC= $(CW) -C $(CW_CCC_COMPILERS) --
834 BUILD_CPP= /usr/ccs/lib/cpp
835 BUILD_LD= /usr/ccs/bin/ld
836 BUILD_LINT= $(SPRO_ROOT)/sunstudio12.1/bin/lint

838 $(MACH)_CC= $(BUILD_CC)
839 $(MACH)_CCC= $(BUILD_CCC)
840 $(MACH)_CPP= $(BUILD_CPP)
841 $(MACH)_LD= $(BUILD_LD)
842 $(MACH)_LINT= $(BUILD_LINT)
843 $(MACH64)_CC= $(BUILD_CC)
844 $(MACH64)_CCC= $(BUILD_CCC)
845 $(MACH64)_CPP= $(BUILD_CPP)

```

```

846 $(MACH64)_LD= $(BUILD_LD)
847 $(MACH64)_LINT= $(BUILD_LINT)

849 sparc_AS= /usr/ccs/bin/as -xregsym=no
850 sparcv9_AS= $(MACH)_AS

852 i386_AS= /usr/ccs/bin/as
853 $(__GNUC)i386_AS= $(ONBLD_TOOLS)/bin/$(MACH)/aw
854 amd64_AS= $(ONBLD_TOOLS)/bin/$(MACH)/aw

856 NATIVECC= $(MACH)_CC
857 NATIVECCC= $(MACH)_CCC
858 NATIVECPP= $(MACH)_CPP
859 NATIVEAS= $(MACH)_AS
860 NATIVELD= $(MACH)_LD
861 NATIVELINT= $(MACH)_LINT

863 #
864 # Makefile.master.64 overrides these settings
865 #
866 CC= $(NATIVECC)
867 CCC= $(NATIVECCC)
868 CPP= $(NATIVECPP)
869 AS= $(NATIVEAS)
870 LD= $(NATIVELD)
871 LINT= $(NATIVELINT)

873 # Pass -Y flag to cpp (method of which is release-dependent)
874 CCYFLAG= -Y I,

876 BDIRECT= -Bdirect
877 BDYNAMIC= -Bdynamic
878 BLOCAL= -Blocal
879 BNODIRECT= -Bnodirect
880 BREDUCE= -Breduce
881 BSTATIC= -Bstatic

883 ZDEFS= -zdefs
884 ZDIRECT= -zdirect
885 ZIGNORE= -zignore
886 ZINITFIRST= -zinitfirst
887 ZINTERPOSE= -zinterpose
888 ZLAZYLOAD= -zlazyload
889 ZLOADFLTR= -zloadfltr
890 ZMULDEFS= -zmuldefs
891 ZNODEFAULTLIB= -znodefaultlib
892 ZNODEFS= -znodefs
893 ZNODELETE= -znodelete
894 ZNODLOPEN= -znodlopen
895 ZNODUMP= -znodump
896 ZNOLAZYLOAD= -znolazyload
897 ZNOLDYNSYM= -znolddynsym
898 ZNORELOC= -znoreloc
899 ZNOVERSION= -znoversion
900 ZRECORD= -zrecord
901 ZREDLOCSYM= -zredlocsymb
902 ZTEXT= -ztext
903 ZVERBOSE= -zverbose

905 GSHARED= -G
906 CCMT= -mt

908 # Handle different PIC models on different ISAs
909 # (May be overridden by lower-level Makefiles)

911 sparc_C_PICFLAGS = -fpic

```

```

912 sparcv9_C_PICFLAGS = -fpic
913 i386_C_PICFLAGS = -fpic
914 amd64_C_PICFLAGS = -fpic
915 C_PICFLAGS = $(MACH)_C_PICFLAGS
916 C_PICFLAGS64 = $(MACH64)_C_PICFLAGS

918 sparc_C_BIGPICFLAGS = -fPIC
919 sparcv9_C_BIGPICFLAGS = -fPIC
920 i386_C_BIGPICFLAGS = -fPIC
921 amd64_C_BIGPICFLAGS = -fPIC
922 C_BIGPICFLAGS = $(MACH)_C_BIGPICFLAGS
923 C_BIGPICFLAGS64 = $(MACH64)_C_BIGPICFLAGS

925 # CC requires there to be no space between '-K' and 'pic' or 'PIC'.
926 # and does not support -f
927 sparc_CC_PICFLAGS = -_cc=-Kpic -_gcc=-fpic
928 sparcv9_CC_PICFLAGS = -_cc=-KPIC -_gcc=-fPIC
929 i386_CC_PICFLAGS = -_cc=-Kpic -_gcc=-fpic
930 amd64_CC_PICFLAGS = -_cc=-Kpic -_gcc=-fpic
931 CC_PICFLAGS = $(MACH)_CC_PICFLAGS
932 CC_PICFLAGS64 = $(MACH64)_CC_PICFLAGS

934 AS_PICFLAGS= -K pic
935 AS_BIGPICFLAGS= -K PIC

937 #
938 # Default label for CTF sections
939 #
940 CTFCVTFLAGS= -i -L VERSION

942 #
943 # Override to pass module-specific flags to ctfmerge. Currently used only by
944 # krtld to turn on fuzzy matching, and source-level debugging to inhibit
945 # stripping.
946 #
947 CTFMRGFLAGS=

949 CTFCONVERT_O = $(CTFCONVERT) $(CTFCVTFLAGS) $@

951 # Rules (normally from make.rules) and macros which are used for post
952 # processing files. Normally, these do stripping of the comment section
953 # automatically.
954 # RELEASE_CM: Should be edited to reflect the release.
955 # POST_PROCESS_O: Post-processing for '.o' files.
956 # POST_PROCESS_A: Post-processing for '.a' files (currently null).
957 # POST_PROCESS_SO: Post-processing for '.so' files.
958 # POST_PROCESS: Post-processing for executable files (no suffix).
959 # Note that these macros are not completely generalized as they are to be
960 # used with the file name to be processed following.
961 #
962 # It is left as an exercise to Release Engineering to embellish the generation
963 # of the release comment string.
964 #
965 # If this is a standard development build:
966 # compress the comment section (mcs -c)
967 # add the standard comment (mcs -a $(RELEASE_CM))
968 # add the development specific comment (mcs -a $(DEV_CM))
969 #
970 # If this is an installation build:
971 # delete the comment section (mcs -d)
972 # add the standard comment (mcs -a $(RELEASE_CM))
973 # add the development specific comment (mcs -a $(DEV_CM))
974 #
975 # If this is an release build:
976 # delete the comment section (mcs -d)
977 # add the standard comment (mcs -a $(RELEASE_CM))

```

```

978 #
979 # The following list of macros are used in the definition of RELEASE_CM
980 # which is used to label all binaries in the build:
981 #
982 # RELEASE Specific release of the build, eg: 5.2
983 # RELEASE_MAJOR Major version number part of $(RELEASE)
984 # RELEASE_MINOR Minor version number part of $(RELEASE)
985 # VERSION Version of the build (alpha, beta, Generic)
986 # PATCHID If this is a patch this value should contain
987 # the patchid value (eg: "Generic 100832-01"), otherwise
988 # it will be set to $(VERSION)
989 # RELEASE_DATE Date of the Release Build
990 # PATCH_DATE Date the patch was created, if this is blank it
991 # will default to the RELEASE_DATE
992 #
993 RELEASE_MAJOR= 5
994 RELEASE_MINOR= 11
995 RELEASE= $(RELEASE_MAJOR).$(RELEASE_MINOR)
996 VERSION= SunOS Development
997 PATCHID= $(VERSION)
998 RELEASE_DATE= release date not set
999 PATCH_DATE= $(RELEASE_DATE)
1000 RELEASE_CM= "@$(POUND_SIGN)SunOS $(RELEASE) $(PATCHID) $(PATCH_DATE)"
1001 DEV_CM= "@$(POUND_SIGN)SunOS Internal Development: non-nightly build"

1003 PROCESS_COMMENT= @?${MCS} -d -a $(RELEASE_CM) -a $(DEV_CM)
1004 $(RELEASE_BUILD)PROCESS_COMMENT= @?${MCS} -d -a $(RELEASE_CM)

1006 STRIP_STABS= $(STRIP) -x $@
1007 $(SRCDBGBLD)STRIP_STABS= :

1009 POST_PROCESS_O=
1010 POST_PROCESS_A=
1011 POST_PROCESS_SO= $(PROCESS_COMMENT) $@ ; $(STRIP_STABS) ; \
1012 $(ELFSIGN_OBJECT)
1013 POST_PROCESS= $(PROCESS_COMMENT) $@ ; $(STRIP_STABS) ; \
1014 $(ELFSIGN_OBJECT)

1016 #
1017 # chk4ubin is a tool that inspects a module for a symbol table
1018 # ELF section size which can trigger an OBP bug on older platforms.
1019 # This problem affects only specific sun4u bootable modules.
1020 #
1021 CHK4UBIN= $(ONBLD_TOOLS)/bin/$(MACH)/chk4ubin
1022 CHK4UBINFLAGS=
1023 CHK4UBINARY= $(CHK4UBIN) $(CHK4UBINFLAGS) $@

1025 #
1026 # PKGARCHIVE specifies the default location where packages should be
1027 # placed if built.
1028 #
1029 $(RELEASE_BUILD)PKGARCHIVESUFFIX= -nd
1030 PKGARCHIVE=$(SRC)/../../packages/$(MACH)/nightly$(PKGARCHIVESUFFIX)

1032 #
1033 # The repositories will be created with these publisher settings. To
1034 # update an image to the resulting repositories, this must match the
1035 # publisher name provided to "pkg set-publisher."
1036 #
1037 PKGPUBLISHER_REDIST= on-nightly
1038 PKGPUBLISHER_NONREDIST= on-extra

1040 # Default build rules which perform comment section post-processing.
1041 #
1042 .c:
1043 $(LINK.c) -o $@ $< $(LDLIBS)

```



```

1044      $(POST_PROCESS)
1045 .c.o:  $(COMPILE.c) $(OUTPUT_OPTION) $< $(CTFCONVERT_HOOK)
1046      $(POST_PROCESS_O)
1047
1048 .c.a:  $(COMPILE.c) -o $% $<
1049      $(PROCESS_COMMENT) $%
1050      $(AR) $(ARFLAGS) $@ $%
1051      $(RM) $%
1052
1053 .s.o:  $(COMPILE.s) -o $@ $<
1054      $(POST_PROCESS_O)
1055
1056 .s.a:  $(COMPILE.s) -o $% $<
1057      $(PROCESS_COMMENT) $%
1058      $(AR) $(ARFLAGS) $@ $%
1059      $(RM) $%
1060
1061 .cc:   $(LINK.cc) -o $@ $< $(LDLIBS)
1062      $(POST_PROCESS)
1063
1064 .cc.o: $(COMPILE.cc) $(OUTPUT_OPTION) $<
1065      $(POST_PROCESS_O)
1066
1067 .cc.a: $(COMPILE.cc) -o $% $<
1068      $(AR) $(ARFLAGS) $@ $%
1069      $(PROCESS_COMMENT) $%
1070      $(RM) $%
1071
1072 .y:    $(YACC.y) $<
1073      $(LINK.c) -o $@ y.tab.c $(LDLIBS)
1074      $(POST_PROCESS)
1075      $(RM) y.tab.c
1076
1077 .y.o:  $(YACC.y) $<
1078      $(COMPILE.c) -o $@ y.tab.c $(CTFCONVERT_HOOK)
1079      $(POST_PROCESS_O)
1080      $(RM) y.tab.c
1081
1082 .l:    $(RM) $*.c
1083      $(LEX.l) $< > $*.c
1084      $(LINK.c) -o $@ $*.c -ll $(LDLIBS)
1085      $(POST_PROCESS)
1086      $(RM) $*.c
1087
1088 .l.o:  $(RM) $*.c
1089      $(LEX.l) $< > $*.c
1090      $(COMPILE.c) -o $@ $*.c $(CTFCONVERT_HOOK)
1091      $(POST_PROCESS_O)
1092      $(RM) $*.c
1093
1095 .bin.o: $(COMPILE.b) -o $@ $<
1096      $(POST_PROCESS_O)
1097
1099 .java.class:
1100      $(COMPILE.java) $<
1101
1102 # Bourne and Korn shell script message catalog build rules.
1103 # We extract all gettext strings with sed(1) (being careful to permit
1104 # multiple gettext strings on the same line), weed out the dups, and
1105 # build the catalogue with awk(1).
1106
1107 .sh.po .ksh.po:
1108      $(SED) -n -e ":a" \
1109      -e "h" \

```

```

1110      -e "s/.*gettext *\(\("[^"]*\)"\).*\/\1/p" \
1111      -e "x" \
1112      -e "s/\(.*\)gettext *\("[^"]*\)"\(\.*\)\/\1\2/" \
1113      -e "t a" \
1114      $< | sort -u | $(AWK) '{ print "msgid\t" $$0 "\nmsgstr" }' > $@
1115
1116 #
1117 # Python and Perl executable and message catalog build rules.
1118 #
1119 .SUFFIXES: .pl .pm .py .pyc
1120
1121 .pl:   $(RM) $@;
1122      $(SED) -e "s@TEXT_DOMAIN@\$(TEXT_DOMAIN)\@" $< > $@;
1123      $(CHMOD) +x $@
1124
1125
1126 .py:   $(RM) $@; $(SED) -e "1s:^\#!@PYTHON@:\#!$(PYTHON):" < $< > $@; $(CHMOD)
1127
1128
1129 .py.pyc:
1130      $(RM) $@
1131      $(PYTHON) -mpy_compile $<
1132      @[ $(<)c = $@ ] || $(MV) $(<)c $@
1133
1134 .py.po:
1135      $(GNUXGETTEXT) $(GNUXGETFLAGS) -d $(<F:%.py=) $< ;
1136
1137 .pl.po .pm.po:
1138      $(XGETTEXT) $(XGETFLAGS) -d $(<F) $< ;
1139      $(RM) $@ ;
1140      $(SED) "/^domain/d" < $(<F).po > $@ ;
1141      $(RM) $(<F).po
1142
1143 #
1144 # When using xgettext, we want messages to go to the default domain,
1145 # rather than the specified one. This special version of the
1146 # COMPILE.cpp macro effectively prevents expansion of TEXT_DOMAIN,
1147 # causing xgettext to put all messages into the default domain.
1148 #
1149 CPPFORPO=$(COMPILE.cpp:\ "$(TEXT_DOMAIN)\ "=TEXT_DOMAIN)
1150
1151 .c.i:  $(CPPFORPO) $< > $@
1152
1153
1154 .h.i:  $(CPPFORPO) $< > $@
1155
1156
1157 .y.i:  $(YACC) -d $<
1158      $(CPPFORPO) y.tab.c > $@
1159      $(RM) y.tab.c
1160
1161
1162 .l.i:  $(LEX) $<
1163      $(CPPFORPO) lex.yy.c > $@
1164      $(RM) lex.yy.c
1165
1166
1167 .c.po: $(CPPFORPO) $< > $<.i
1168      $(BUILD.po)
1169
1170
1171 .cc.po: $(CPPFORPO) $< > $<.i
1172      $(BUILD.po)
1173
1174
1175 .y.po:

```

new/usr/src/Makefile.master

19

```
1176      $(YACC) -d $<
1177      $(CPPFORPO) y.tab.c > $<.i
1178      $(BUILD.po)
1179      $(RM) y.tab.c

1181 .l.po:
1182      $(LEX) $<
1183      $(CPPFORPO) lex.yy.c > $<.i
1184      $(BUILD.po)
1185      $(RM) lex.yy.c

1187 #
1188 # Rules to perform stylistic checks
1189 #
1190 .SUFFIXES: .x .xml .check .xmlchk

1192 .h.check:
1193      $(DOT_H_CHECK)

1195 .x.check:
1196      $(DOT_X_CHECK)

1198 .xml.xmlchk:
1199      $(MANIFEST_CHECK)

1201 #
1202 # Include rules to render automated sccs get rules "safe".
1203 #
1204 include $(SRC)/Makefile.noget
```

new/usr/src/cmd/backup/Makefile.targ

1

1490 Fri Dec 7 21:21:45 2018

new/usr/src/cmd/backup/Makefile.targ

10051 cw(1) shouldn't have to translate -g

```
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 # ident "%Z%M% %I% %E% SMI"
24 #
25 # Copyright 2004 Sun Microsystems, Inc. All rights reserved.
26 # Use is subject to license terms.
27 #
28 # cmd/backup/Makefile.targ
29 #
30 # support for debugging
31 debug:=          DUMPLIB= libdump_g.a
32 debug:=          CFLAGS= $(CCGDEBUG) $(XESS) -DDEBUG -DFDEBUG -DTDEBUG ${SBFLAGS}
33 debug:=          CFLAGS= -g $(XESS) -DDEBUG -DFDEBUG -DTDEBUG ${SBFLAGS}
34 CFLAGS +=        $(XSTRCONST)
35 DEBUGDIR=        .debug
36 DEBUGOBS=        $(OBJECTS:%=$(DEBUGDIR)/%)
37 DEBUGPRODUCTS=  $(UFSDEBUGPRODUCT)
38 UFSDEBUGPRODUCT= $(PRODUCT:%=%_g)
39 UFSDEBUGPROG=    $(PROG:%=%_g)
40 debug:           $(UFSDEBUGPROG)
41 $(UFSDEBUGPROG): $(DEBUGDIR) $$$(DEBUGOBS) $$$(LIBDUMP)
42                 $(LINK.c) -o $@ $(DEBUGOBS) $(LDLIBS)
43                 $(POST_PROCESS)
44 $(DEBUGDIR)/%.o: %.c
45                 $(COMPILE.c) -o $@ $<
46                 $(POST_PROCESS_O)
47
48 $(DEBUGDIR):
49     -@mkdir -p $@
50
51 include $(SRC)/cmd/Makefile.targ
```

new/usr/src/cmd/backup/dump/Makefile

1

```
*****
3972 Fri Dec 7 21:21:45 2018
new/usr/src/cmd/backup/dump/Makefile
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
3 # Use is subject to license terms.
4 #
5 # cmd/backup/dump/Makefile
6 #
7 # Copyright (c) 1980 Regents of the University of California.
8 # All rights reserved. The Berkeley software License Agreement
9 # specifies the terms and conditions for redistribution.
10 #
11 #      dump.h          header file
12 #      dumpfstab.c    reads fstab/vfstab, mtab/mnttab
13 #      dumpitime.c    reads /etc/dumpdates
14 #      dumpmain.c     driver
15 #      dumponline.c   online dump support
16 #      dumpoptr.c     operator interface
17 #      dumptime.c     handles the mag tape and opening/closing
18 #      dumptraverse.c traverses the file system
19 #      lftw.c          fixed version of ftw(3)
20 #      partial.c      partial filesystem dump support
21 #      unctime.c      undo ctime
22 #
23 # CPPFLAGS:
24 #      DEBUG          use local directory to find ddate and dumpdates
25 #      PARTIAL       enable partial filesystem dump support
26 #      TDEBUG        trace out the process forking
27 #
28 #      CAUTION: FSTYPE must be defined before including ../Makefile.backup,
29 #              otherwise UFSROOTPKGUSRLIBFSTYPE doesn't get expanded
30 #              properly and the magic doesn't happen.
31 #
32 FSTYPE=      ufs
33 #
34 include ../Makefile.backup
35 #
36 #CFLAGS +=    -DTDEBUG
37 UFSDUMP=     ufsdump
38 #
39 PROG=        $(UFSDUMP)
40 PRODUCT=     $(PROG)
41 ROLLDIR=     ../../fs.d/ufs/roll_log
42 ROLLOBJS=    $(ROLLDIR)/$(MACH)/roll_log.o
43 ROLLSRCS=    $(ROLLDIR)/roll_log.c
44 #
45 OBJECTS=     dumponline.o dumpfstab.o dumpitime.o dumpmain.o \
46             dumpoptr.o dumptime.o dumptraverse.o unctime.o \
47             partial.o lftw.o
48 OBJECTS+=    $(OBJECTS) roll_log.o
49 SRCS+=       $(OBJECTS:.o=.c)
50 #
51 POFILES=     $(OBJECTS:.o=.po) ../lib/libdump.po
52 POFILE=      ufsdump.po
53 ROLLPOFILES= roll_log.po
54 #
55 # XXX This is only needed for the check: target. It would be nice to
56 # XXX automatically generate the list when needed.
57 HEADERS=     ../../../head/protocols/dumprestore.h \
58             ../../fs.d/ufs/roll_log/roll_log.h ../include/byteorder.h \
59             ../include/memutils.h \
60             ../include/rmt.h dump.h dumpusg.h
```

new/usr/src/cmd/backup/dump/Makefile

2

```
62 CLOBBERFILES += $(PRODUCT) $(DEBUGPRODUCTS) dumpdates
63 #
64 LOCAL=       .
65 GENERAL=     ../include
66 GLOBAL=      ../../../head
67 CPPFLAGS=    -I$(LOCAL) -I$(GENERAL) -I$(GLOBAL) -DPARTIAL \
68             -I$(ROLLDIR) $(CPPFLAGS.master) \
69             -D_LARGEFILE64_SOURCE=1
70 CERRWARN +=  -_gcc=-Wno-implicit-function-declaration
71 CERRWARN +=  -_gcc=-Wno-parentheses
72 CERRWARN +=  -_gcc=-Wno-uninitialized
73 CERRWARN +=  -_gcc=-Wno-clobbered
74 CERRWARN +=  -_gcc=-Wno-extra
75 LIBDUMP=     ../lib/libdump.a
76 LINTLIBDUMP= ../lib/lldump.ldump.ln -lkstat
77 LDLIBS +=    $(BSTATIC) -L../lib -ldump $(BDYNAMIC) \
78             -lsocket -lnsl -ladm -lm -lkstat
79 #
80 UFSROOTDUMPDATES= $(UFSROOTETC)/dumpdates
81 UFSROOTLINK=     $(UFSROOTUSRSBIN)/$(PROG)
82 LINKVALUE=       ../lib/fs/$(FSTYPE)/$(PROG)
83 LIBDIR=          $(UFSROOTUSRRLIBFS)/$(FSTYPE)
84 $(UFSROOTDUMPDATES) := FILEMODE= 0664
85 $(LIBDIR) :=          DIRMODE= 0755
86 #
87 FILEMODE=        04555
88 #
89 .KEEP_STATE:
90 #
91 all:              $(PRODUCT)
92 #
93 $(PROG):          $(OBJECTS) $$$(LIBDUMP)
94                  $(LINK.c) -o $@ $(OBJECTS) $(LDLIBS)
95                  $(POST_PROCESS)
96 #
97 $(LIBDUMP): FRC
98              $(DO_LIBDIR)
99 #
100 $(POFILE):       $(POFILES)
101                 $(RM) $@; cat $(POFILES) > $@
102 #
103 FRC:
104 #
105 install:         all $(LIBDIR) $(UFSROOTPKGUSRLIBFSTYPE) dumpdates \
106                 $(UFSROOTDUMPDATES) $(UFSROOTLINK)
107 #
108 dumpdates:
109                 cp /dev/null dumpdates
110 #
111 #XXX Should actually note return value from close(2), particularly regarding
112 #XXX tape descriptors. Some drives don't actually force anything to media
113 #XXX except when they are writing a file mark....
114 #
115 # grep is looking for non-space/tab
116 lint:            $(SRCS) $(ROLLSRCS) $$$(LINTLIBDUMP)
117                 $(LINT.c) $(SRCS) $(ROLLSRCS) $(LINTLIBDUMP) 2>&1 \
118                 | sed -f lint.sed | grep '^[^ ]'
119 #
120 $(LINTLIBDUMP): FRC
121                 cd ../lib; pwd; $(MAKE) lint
122                 pwd
123 #
124 check:          FRC
125                 $(CSTYLE) $(CSTYLEFLAGS) $(SRCS) $(ROLLSRCS) $(HEADERS)
126                 $(HDRCHK) $(HDRCHKFLAGS) $(HEADERS)
```

```
128 clean:
129     $(RM) $(OBJECTS) $(ROLLOBJS) $(DEBUGOBS) *.ln

131 $(LIBDIR):
132     $(INS.dir)

134 $(UFSROOTLINK):
135     -$(RM) $@; $(SYMLINK) $(LINKVALUE) $(UFSROOTLINK)

137 roll_log.o:
138     cd $(ROLLDIR) ; pwd ; $(MAKE)
139     cp $(ROLLOBJS) .
140     pwd

142 .debug/roll_log.o:
143     cd $(ROLLDIR) ; pwd ; $(MAKE) CFLAGS=-g
144     cp $(ROLLOBJS) .debug
145     pwd

142 $(ROLLPOFILES): $(ROLLSRCS)
143     cd $(ROLLDIR) ; pwd ; $(MAKE) `echo $(ROLLPOFILES) | sed -e 's@$(ROLLDIR)
144     cp $(ROLLDIR)/*.po .
145     pwd

147 ../lib/libdump.po:
148     cd ../lib ; pwd ; $(MAKE) libdump.po

150 include ../Makefile.targ
```

```

*****
3330 Fri Dec 7 21:21:46 2018
new/usr/src/cmd/backup/lib/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 LIBRARY=      libdump.a

28 # Has to be before include of Makefile.backup
29 # This should be POFILE=libdump.po, but that causes make to
30 # fall over due to some seriously weird interactions in the
31 # various indirectly-included makefiles. So, since this works
32 # and is otherwise harmless, we fake it.
33 PROG=          libdump

35 # Include library definitions, then backup definitions, as in general
36 # we want the flags and such from our tree.

38 include ../../lib/Makefile.lib
39 include ../Makefile.backup

41 SRCDIR =      .

43 # Specifically request the construction of a static library.
44 # This library is not installed in the proto area.
45 LIBS=         $(LIBRARY)

47 HDRS=        ../include/byteorder.h \
48              ../include/memutils.h ../include/myrcmd.h \
49              ../../head/protocols/dumpstore.h \
50              ../include/rmt.h

52 YFILE=        getdate.y
53 YSRC=         getdate.c

55 CLOBBERFILES += $(LIBS) $(GLIB)

57 LOBJS=        rmtlib.o myrcmd.o \
58              $(YSRC:%.c=%.o) \
59              byteorder.o memutils.o $(RPC_CLNT:%.c=%.o) $(RPC_XDR:%.c=%.o)

61 OBJECTS=      $(LOBJS)

```

```

62 POFILES=      $(OBJECTS:.o=.po)
63 GENERAL=      ../include
64 GLOBAL=       ../../../../head
65 CPPFLAGS=     -I$(GENERAL) -I$(GLOBAL) $(CPPFLAGS.master)
66 CERRWARN +=   -_gcc=-Wno-implicit-function-declaration
67 CERRWARN +=   -_gcc=-Wno-parentheses
68 CERRWARN +=   -_gcc=-Wno-unused-label
69 CERRWARN +=   -_gcc=-Wno-unused-variable
70 LINTOUT=      lint.out
71 CLEANFILES=   $(OBJECTS) $(LINTOUT) $(LINTLIB) $(DEBUGS) *.ln \
72              $(YSRC) $(LIBRARY)
73 LINTFLAGS +=  -y

75 # support for -g library
76 GLIB=         libdump_g.a
77 DEBUGS=      $(OBJECTS:%=.debug%)
78 $(GLIB):=    AROBJS = $(DEBUGS)
79 $(GLIB):=    DIR = .debug
80 $(GLIB):=    CFLAGS= $(CCGDEBUG) $(XESS) -DDEBUG -DYDEBUG ${SBFLAGS}
80 $(GLIB):=    CFLAGS= -g $(XESS) -DDEBUG -DYDEBUG ${SBFLAGS}

82 .KEEP_STATE:

84 all:         $(LIBS)

86 debug:      $(LIBS) $(GLIB)

88 .debug:     -@mkdir -p $@
89

91 .debug/%.o: %.c
92             $(COMPILE.c) -o $@ $<
93             $(POST_PROCESS_O)

95 $(GLIB): .debug $$$(DEBUGS)
96           $(BUILD.AR)
97           $(POST_PROCESS_A)

99 $(OBJECTS): $(HDRS)

101 install:   all

103 $(POFILE): $(POFILES)
104            $(RM) $@; cat $(POFILES) > $@

106 # rpcgen produces unused local variables that we can't easily suppress.
107 # It is also stupid about 32/64 bit integers. Since we don't support
108 # the RPC subsystem any more, just ignore complaints about it all.
109 # We have no control over yaccpar, and it has lots of 32/64 complaints.
110 # Assumes lint run with -s argument
111 lint:       lint.out
112            sed -f lint.sed lint.out

114 lint.out:  $(LINTLIB)

116 check:     $(HDRS)
117            $(CSTYLE) $(CSTYLEFLAGS) `echo $(SRCS) | sed -e s/getdate.c/' $(HDRS)
118            $(HDRCHK) $(HDRCHKFLAGS) $(HDRS)

120 # include library targets
121 include ../../lib/Makefile.targ

123 _msg:      $(POFILE)

```

new/usr/src/cmd/cmd-inet/sbin/dhcpagent/Makefile

1

```
*****
2001 Fri Dec 7 21:21:46 2018
new/usr/src/cmd/cmd-inet/sbin/dhcpagent/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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 2007 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 # Copyright (c) 2016, Chris Fraire <cfraire@me.com>.
25 #
26
27 PROG = dhcpagent
28 ROOTFS_PROG = $(PROG)
29 DEFAULTFILES = dhcpagent.dfl
30
31 OBJS = adopt.o agent.o async.o bound.o class_id.o defaults.o inform.o \
32        init_reboot.o interface.o ipc_action.o packet.o release.o renew.o \
33        request.o script_handler.o select.o states.o util.o
34
35 include ../../../../Makefile.cmd
36
37 SRCS = $(OBJS:%.o=%.c)
38 POFILES = $(OBJS:%.o=%.po)
39 XGETFLAGS += -a -x dhcpagent.xcl
40
41 CERRWARN += -_gcc=-Wno-switch
42 CERRWARN += -_gcc=-Wno-parentheses
43
44 #
45 # to compile a debug version, do a 'make COPTFLAG="-g -XO0"'
46 #
47
48 CPPFLAGS += -D_XOPEN_SOURCE=500 -D__EXTENSIONS__
49 LDLIBS += -lnet -lnvpair -ldhcpagent -ldhcputil -linetutil -ldevinfo \
46        -ldlpi -lresolv -lsocket -lipadm
50
51 # Disable warnings that affect all XPG applications.
52 LINTFLAGS += -erroff=E_INCONS_ARG_DECL2 -erroff=E_INCONS_VAL_TYPE_DECL2
53
54 .KEEP_STATE:
55
56 all: $(ROOTFS_PROG)
57
58 install: all $(ROOTSBINPROG) $(ROOTETCDEFAULTFILES)
59
60 $(PROG): $(OBJS)
```

new/usr/src/cmd/cmd-inet/sbin/dhcpagent/Makefile

2

```
58 $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
59 $(POST_PROCESS)
60
61 $(POFILE): $(POFILES)
62 $(RM) $@; $(CAT) $(POFILES) > $@; $(RM) $(POFILES)
63
64 clean:
65 $(RM) $(OBJS)
66
67 lint: lint_SRCS
68
69 include ../../../../Makefile.targ
```

new/usr/src/cmd/file/Makefile

1

2534 Fri Dec 7 21:21:46 2018

new/usr/src/cmd/file/Makefile

10051 cw(1) shouldn't have to translate -g

```
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 2007 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 PROG= file
27 XPG4PROG= file
28 MAGIC= magic

30 ELFCAP= $(SRC)/common/elfcap
31 SGSRTCID= $(SRC)/common/sgrtcid

33 LOBJS= file.o elf_read32.o elf_read64.o magicutils.o
34 OBJJS= $(LOBJS).elfcap.o
35 XPG4OBJJS= $(OBJJS:%.o=xpg4%.o)
36 SRCS= file.c elf_read.c magicutils.c $(ELFCAP)/elfcap.c

38 include ../Makefile.cmd

40 CSTD= $(CSTD_GNU99)
41 C99LMODE= -Xc99=%all

43 CERRWARN += -_gcc=-Wno-uninitialized
44 CERRWARN += -_gcc=-Wno-type-limits

46 POFILE= file_all.po
47 POFILES= $(SRCS:%.c=%.po)

49 # The debug binary can be built using the flags
50 # SOURCEDEBUG=yes CGLOBALSTATIC=
49 # The debug binary can be built using the flag
50 # -D COPTFLAG=-g CGLOBALSTATIC=
51 # This will avoid the multiple symbols definition error
52 # for static global variables in elf_read32.o and elf_read64.o

54 LDLIBS += -lelf
55 CPPFLAGS += -I$(ELFCAP) -I$(SGSRTCID)
56 $(XPG4) := CFLAGS += -DXPG4

58 ROOTETCMAGIC= $(MAGIC:%=$(ROOTETC)/%)
```

new/usr/src/cmd/file/Makefile

2

```
60 $(ROOTETCMAGIC) := FILEMODE = $(LIBFILEMODE)

62 .PARALLEL: $(OBJJS) $(XPG4OBJJS) $(POFILES)

64 .KEEP_STATE:

66 all: $(PROG) $(XPG4) $(MAGIC)

68 $(PROG) : $(OBJJS)
69 $(LINK.c) $(OBJJS) -o $@ $(LDLIBS)
70 $(POST_PROCESS)

72 $(XPG4) : $(XPG4OBJJS)
73 $(LINK.c) $(XPG4OBJJS) -o $@ $(LDLIBS)
74 $(POST_PROCESS)

76 %.o: %.c
77 $(COMPILE.c) -o $@ $<

79 %32.o: %.c
80 $(COMPILE.c) -o $@ $<

82 %64.o: %.c
83 $(COMPILE.c) -D_ELF64 -o $@ $<

85 xpg4%.o: %.c
86 $(COMPILE.c) -o $@ $<

88 xpg4%32.o: %.c
89 $(COMPILE.c) -o $@ $<

91 xpg4%64.o: %.c
92 $(COMPILE.c) -D_ELF64 -o $@ $<

94 elfcap.o: $(ELFCAP)/elfcap.c
95 $(COMPILE.c) -o $@ $(ELFCAP)/elfcap.c

97 xpg4_elfcap.o: $(ELFCAP)/elfcap.c
98 $(COMPILE.c) -o $@ $(ELFCAP)/elfcap.c

100 $(POFILE): $(POFILES)
101 $(RM) $@
102 cat $(POFILES) > $@

104 install: all $(ROOTPROG) $(ROOTXPG4PROG) $(ROOTETCMAGIC)

106 clean:
107 $(RM) $(OBJJS) $(XPG4OBJJS)

109 lint: lint_SRCS

111 include ../Makefile.targ
```


new/usr/src/cmd/fs.d/pcfs/fsck/Makefile

1

```
*****
1803 Fri Dec 7 21:21:47 2018
new/usr/src/cmd/fs.d/pcfs/fsck/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 FSTYPE=      pcfs
27 LIBPROG=     fsck
28 ATTKM=      $(LIBPROG)

30 include      ../../Makefile.fstype

32 FSCKOBS=     fsck.o bpb.o clusters.o fat.o dir.o pcfs_common.o
33 FSCKSRCS=    $(FSCKOBS:%.o=%.c)

35 #
36 # Error injection module for debugging purposes
37 #
38 #DEBUGOBS=   inject.o
39 #DEBUGSRCS=  $(DEBUGOBS:%.o=%.c)

41 OBS=        $(FSCKOBS) $(DEBUGOBS)
42 SRCS=       $(FSCKSRCS) $(DEBUGSRCS)
43 LSRCS=      $(FSCKSRCS) $(DEBUGSRCS)

45 # for messaging catalog
46 #
47 POFILE=     fsck.po

49 catalog:    $(POFILE)

51 #
52 # Uncomment the following two lines to build for dbx debugging
53 #COPTFLAG = -g
54 #COPTFLAG64 = -g
55 #
56 CPPFLAGS += -D_LARGEFILE64_SOURCE

53 CERRWARN += -_gcc=-Wno-parentheses
54 CERRWARN += -_gcc=-Wno-unused-variable
55 CERRWARN += -_gcc=-Wno-uninitialized
```

new/usr/src/cmd/fs.d/pcfs/fsck/Makefile

2

```
57 #
58 # We need to tell lint not to complain about the routines
59 # we are grabbing from the kernel as being undefined.
60 #
61 LINTFLAGS += -u

63 $(LIBPROG):  $(OBS)
64              $(LINK.c) -o $@ $(OBS) $(LDLIBS)
65              $(POST_PROCESS)

67 lint_LSRCS:
68             $(LINT.c) $(LSRCS) $(LDLIBS)

70 lint:       lint_LSRCS

72 clean:
73            $(RM) $(FSCKOBS) $(DEBUGOBS)
```

new/usr/src/cmd/fs.d/smbclnt/mount/Makefile

1

1501 Fri Dec 7 21:21:47 2018

new/usr/src/cmd/fs.d/smbclnt/mount/Makefile

10051 cw(1) shouldn't have to translate -g

```
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 #
26 #
27 # cmd/fs.d/smbclnt/mount/Makefile
28 #
29 #
30 FSTYPE=      smbfs
31 LIBPROG=     mount
32 ROOTFS_PROG= $(LIBPROG)
33 #
34 include      ../../Makefile.fstype
35 #
36 OBJS=       $(LIBPROG).o
37 SRCS=       $(LIBPROG).c
38 POFILE=     $(LIBPROG).po
39 CLOBBERFILES += $(LIBPROG)
40 #
41 $(ROOTLIBFSTYPE)/mount := FILEMODE= 04555
42 #
43 CFLAGS += $(CVERBOSE)
44 CERRWARN += _gcc=-Wno-parentheses
45 CERRWARN += _gcc=-Wno-unused-label
46 CSTD= $(CSTD_GNU99)
47 #
48 LDLIBS += -lsmbfs
49 #
50 CPPFLAGS += -I$(SRC)/uts/common -I$(SRC)/lib/libmbfs
51 #
52 # uncomment these for dbx debugging
53 #COPTFLAG = -g
54 #CTF_FLAGS =
55 #CTFCONVERT_O=
56 #CTFMERGE_LIB=
57 #
58 .KEEP_STATE:
59 #
60 all:        $(ROOTFS_PROG)
```

new/usr/src/cmd/fs.d/smbclnt/mount/Makefile

2

```
56 POFILE=     mount.po
57 #
58 catalog:    $(POFILE)
59 #
60 install:    $(ROOTLIBFSTYPEPROG)
61 #
62 lint:      lint_SRCS
63 #
64 clean:
65          $(RM) $(OBJS) $(POFILE)
```

new/usr/src/cmd/fs.d/smbclnt/smbiod/Makefile

1

```
*****
1435 Fri Dec 7 21:21:48 2018
new/usr/src/cmd/fs.d/smbclnt/smbiod/Makefile
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 2009, 2010, Oracle and/or its affiliates. All rights reserved.
24 #
25 #
26 #
27 # cmd/fs.d/smbclnt/smbiod/Makefile
28 #
29 #
30 FSTYPE=      smbfs
31 TYPEPROG=    smbiod
32 #
33 include      ../../Makefile.fstype
34 #
35 OBJS=        $(TYPEPROG).o
36 SRCS=        $(TYPEPROG).c
37 POFILE=      $(TYPEPROG).po
38 #
39 CFLAGS +=    $(CCVERBOSE)
40 CSTD=        $(CSTD_GNU99)
41 #
42 # This is a multi-thread program but Nevada
43 # no longer needs -lthread
44 LDLIBS +=    -lsmbfs -lumem
45 #
46 CPPFLAGS +=  -I$(SRC)/lib/libmbfs \
47              -I$(SRC)/uts/common/smbclnt -I$(SRC)/uts/common
48 #
49 # Debugging
50 ${NOT_RELEASE_BUILD} CPPFLAGS += -DDEBUG
51 #
52 # uncomment these for dbx debugging
53 #COPTFLAG = -g
54 #CTF_FLAGS =
55 #CTFCONVERT_O=
56 #CTFMERGE_LIB=
57 #
58 all:         $(TYPEPROG)
59 #
60 catalog:     $(POFILE)
```

new/usr/src/cmd/fs.d/smbclnt/smbiod/Makefile

2

```
56 lint:      lint_SRCS
57 #
58 clean:
59             $(RM) $(OBJS) $(POFILE)
60 #
61 .KEEP_STATE:
```

new/usr/src/cmd/fs.d/smbclnt/smbutil/Makefile

1

```
*****
2069 Fri Dec 7 21:21:48 2018
new/usr/src/cmd/fs.d/smbclnt/smbutil/Makefile
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #
26 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
27 #
28 #
29 #
30 # cmd/fs.d/smbclnt/smbutil/Makefile
31 #
32 #
33 PROG=          smbutil
34 #
35 OBJS=  smbutil.o discon.o info.o login.o lookup.o print.o status.o view.o \
36        shares_rap.o shares_rpc.o srsvsvcl_clnt.o srsvsvcl_ndr.o
37 #
38 SRCS=  $(OBJS:%.o=%.c)
39 #
40 include        $(SRC)/cmd/Makefile.cmd
41 #
42 POFILE= smbutil_all.po
43 POFILES= $(OBJS:%.o=%.po)
44 #
45 CLEANFILES +=  srsvsvcl_ndr.c
46 CLOBBERFILES+= $(POFILE) $(POFILES)
47 #
48 CSTD=  $(CSTD_GNU99)
49 #
50 LDLIBS += -lmlrpc -lsmbfs -lnsl
51 #
52 CPPFLAGS += -I$(SRC)/lib/libmbfs \
53            -I$(SRC)/uts/common/smbclnt -I$(SRC)/uts/common
54 #
55 # Debugging
56 ${NOT_RELEASE_BUILD} CPPFLAGS += -DDEBUG
57 #
58 # uncomment these for dbx debugging
59 #COPTFLAG = -g
60 #CTF_FLAGS =
61 #CTFCONVERT_O=
```

new/usr/src/cmd/fs.d/smbclnt/smbutil/Makefile

2

```
62 #CTFMERGE_LIB=
63 #
64 # disable some of the less important lint
65 LINTFLAGS      += -erroff=E_FUNC_RET_ALWAYS_IGNORE2
66 LINTFLAGS      += -erroff=E_FUNC_RET_MAYBE_IGNORED2
67 #
68 all:           $(PROG)
69 #
70 $(PROG):       $(OBJS)
71                $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
72                $(POST_PROCESS)
73 #
74 install:       all $(ROOTPROG)
75 #
76 $(ROOTPROG):  $(PROG)
77                $(INS.file) $(PROG)
78 #
79 catalog:      $(POFILE)
80 #
81 $(POFILE):    $(POFILES)
82                $(RM) $@
83                $(CAT) $(POFILES) > $@
84 #
85 lint:         lint_SRCS
86 #
87 clean:
88                $(RM) $(OBJS) $(CLEANFILES)
89 #
90 include ../../Makefile.targ
91 #
92 srsvsvcl_ndr.c : srsvsvcl.ndl
93                  $(NDRGEN) -Y $(ANSI_CPP) $(CPPFLAGS) srsvsvcl.ndl
94 #
95 .KEEP_STATE:
```

new/usr/src/cmd/geniconvtbl/Makefile.com

1

3976 Fri Dec 7 21:21:49 2018

new/usr/src/cmd/geniconvtbl/Makefile.com

10051 cw(1) shouldn't have to translate -g

```
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 2007 Sun Microsystems, Inc. All rights reserved.
22 # Use is subject to license terms.
23 #
```

25 \$(NOT_NATIVE)NATIVE_BUILD = \$(POUND_SIGN)

27 ITM = geniconvtbl.so
28 PROG = geniconvtbl

30 SRCSH1 = iconv_tm.h hash.h
31 SRCCH1 = itmcomp.h itm_util.h matype.h
32 SRCSC1 = itmcomp.c assemble.c disassemble.c itm_util.c
33 SRCY1 = itm_comp.y
34 SRCL1 = itm_comp.l
35 SRCI1 = geniconvtbl.c

38 YTABC = y.tab.c
39 YTABH = y.tab.h
40 LEXYY = lex.yy.c
41 YOUT = y.output
42 MAPFILE = ../mapfile

46 SRCSH = \$(SRCSH1:%h=./%.h)
47 SRCCH = \$(SRCCH1:%h=./%.h)
48 SRCSC = \$(SRCSC1:%c=./%.c)
49 SRCI = \$(SRCI1:%c=./%.c)
50 SRCY = \$(SRCY1:%y=./%.y)
51 SRCL = \$(SRCL1:%l=./%.l)

53 SRCYC = \$(SRCY:%y=%.c)
54 SRCLC = \$(SRCL:%l=%.c)

56 SRC = \$(SRCSC) \$(YTABC) \$(LEXYY)
57 HDRS = \$(SRCCH1) \$(ERNOSTRH)

61 SED = sed

new/usr/src/cmd/geniconvtbl/Makefile.com

2

62 LEXSED = ../lex.sed
63 YACCSED = ../yacc.sed

67 # include ../../lib/Makefile.lib
68 include ../../Makefile.cmd

71 ROOTDIRS32= \$(ROOTLIB)/iconv
72 ROOTDIRS64= \$(ROOTLIB)/iconv/\$(MACH64)
73 ROOTITM32 = \$(ROOTDIRS32)/\$(ITM)
74 ROOTITM64 = \$(ROOTDIRS64)/\$(ITM)

76 # defined for some useful targets like clean,
76 #
77 # definition for some useful target like clean,
77 OBJ = \$(SRCSC1:%c=%.o) \$(YTABC:%c=%.o) \$(LEXYY:%c=%.o)

79 CHECKHDRS = \$(HDRS%.h=%.check)

81 CLOBBERFILES= \$(ITM) \$(SRCYC)
82 CLEANFILES = \$(OBJ) \$(YTABC) \$(YTABH) \$(LEXYY) \$(YOUT) \
83 \$(POFILES) \$(POFILE)

85 CPPFLAGS += -I. -I..
86 CERRWARN += _gcc=-Wno-uninitialized
87 CERRWARN += _gcc=-Wno-unused-label
88 CERRWARN += _gcc=-Wno-switch
89 CERRWARN += _gcc=-Wno-unused-variable
90 CERRWARN += _gcc=-Wno-implicit-function-declaration
91 YFLAGS += -d -v
92 CFLAGS += -D_FILE_OFFSET_BITS=64

94 \$(ITM) := CFLAGS += \$(GSHARED) \$(C_PICFLAGS) \$(ZTEXT) -h \$@
95 \$(ITM) := CPPFLAGS += -D_REENTRANT
96 \$(ITM) := sparc_CFLAGS += -xregs=no%appl
97 \$(ITM) := sparcv9_CFLAGS += -xregs=no%appl

99 LDLIBS += -lgen

101 MY_NATIVE_CPPFLAGS = -D_FILE_OFFSET_BITS=64 -I. -I..
102 MY_NATIVE_LDFLAGS = \$(MAPFILE.NES:%=-M%) \$(MAPFILE.PGA:%=-M%) \$(MAPFILE.NED:%=-M%)
103 MY_NATIVE_LDLIBS = -lgen

105 #
106 # Message catalog
107 #
108 POFILES= \$(SRCSC1:%c=%.po) \$(SRCI1:%c=%.po) \
109 \$(SRCY1:%y=%.po) \$(SRCL1:%l=%.po)

111 POFILE= geniconvtbl_.po

117 .KEEP_STATE:

119 .PARALLEL: \$(ITM) \$(OBJ)

121 \$(PROG): \$(OBJ)
122 \$(LINK.c) \$(OBJ) -o \$@ \$(LDLIBS)
123 \$(POST_PROCESS)

125 \$(ITM): \$(SRCI)

new/usr/src/cmd/geniconvtbl/Makefile.com

3

```
126 $(CC) $(CFLAGS) $(CPPFLAGS) -M$(MAPFILE) -o $@ $(SRCI) $(LDLIBS)
127 $(POST_PROCESS_SO)

129 $(YTABC) $(YTABH): $(SRCY)
130 $(YACC) $(YFLAGS) $(SRCY)
131 @ $(MV) $(YTABC) $(YTABC)~
132 @ $(SED) -f $(YACSED) $(YTABC)~ > $(YTABC)
133 @ $(RM) $(YTABC)~

135 $(LEXY): $(SRCL) $(YTABH)
136 $(LEX) -t $(SRCL) | $(SED) -f $(LEXSED) > $(LEXY)

139 $(POFILE): .WAIT $(POFILES)
140 $(RM) $@
141 $(CAT) $(POFILES) >$@

143 $(POFILES): $(SRCSC) $(SRCI) $(SRCY) $(SRCL)

145 %.po: ../%.c
146 $(COMPILE.cpp) $< > $<.i
147 $(BUILD.po)

150 lint : lint_SRCS1 lint_SRCS2

153 lint_SRCS1: $(SRCS)
154 $(LINT.c) $(SRCS) $(LDLIBS)

156 lint_SRCS2: $(SRCI)
157 $(LINT.c) $(SRCI) $(LDLIBS)

161 hdrchk: $(HDRCHECKS)

163 cstyle: $(SRCS)
164 $(DOT_C_CHECK)

166 clean:
167 $(RM) $(CLEANFILES)

170 debug:
171 $(MAKE) all COPTFLAG='' COPTFLAG64='' CFLAGS='-g -DDEBUG'

169 %.o: %.c
170 $(COMPILE.c) $<

172 %.o: ../%.c
173 $(COMPILE.c) $<

177 # install rule
183 #
178 $(ROOTDIRS32)/%: $(ROOTDIRS32) %
179 -$(INS.file)

181 $(ROOTDIRS64)/%: $(ROOTDIRS64) %
182 -$(INS.file)

184 $(ROOTDIRS32): $(ROOTLIB)
185 -$(INS.dir)
```

new/usr/src/cmd/geniconvtbl/Makefile.com

4

```
187 $(ROOTDIRS64): $(ROOTDIRS32)
188 -$(INS.dir)

190 $(ROOTLIB) $(ROOTBIN):
191 -$(INS.dir)

193 include ../../Makefile.targ
```

new/usr/src/cmd/hal/addons/cpufreq/Makefile

1

```
*****
1784 Fri Dec 7 21:21:49 2018
new/usr/src/cmd/hal/addons/cpufreq/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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

21 #
22 #
23 # Copyright 2008 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #

27 PROG =          hald-addon-cpufreq
28 OBJS =          addon-cpufreq.o logger.o adt_data.o
29 SRCS =          addon-cpufreq.c ../../hald/logger.c ../../utils/adt_data.c

31 include ../../Makefile.cmd
32 include ../../Makefile.hal

34 ROOTCMDDIR =   $(ROOTLIB_HAL)

36 LDLIBS +=      -lc -ldbus-1 -lhal -lglib-2.0 -ldbus-glib-1 -lpolkit -lbsm

38 CPPFLAGS +=    $(HAL_DBUS_CPPFLAGS) $(HAL_GLIB_CPPFLAGS) $(HAL_CONFIG_CPPFLAGS)
39 CPPFLAGS +=    -I$(ROOT)/usr/include/hal -I../../hald
40 CPPFLAGS +=    -I$(ROOT)/usr/include/libpolkit
41 CFLAGS +=      $(CCGDEBBUG)
42 CPPFLAGS +=    -I$(ROOT)/usr/include/libpolkit -g
43 CSTD =          $(CSTD_GNU99)

44 .KEEP_STATE:

46 all:$(PROG)

48 logger.o: ../../hald/logger.c
49          $(COMPILE.c) -o $@ ../../hald/logger.c
50          $(POST_PROCESS_O)

52 adt_data.o: ../../utils/adt_data.c
53          $(COMPILE.c) -o $@ ../../utils/adt_data.c
54          $(POST_PROCESS_O)

56 $(PROG): $(OBJS)
57          $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
58          $(POST_PROCESS)

60 install: all $(ROOTCMD)
```

new/usr/src/cmd/hal/addons/cpufreq/Makefile

2

```
62 clean:
63          $(RM) $(OBJS)

65 FRC:

67 include ../../Makefile.targ
```

new/usr/src/cmd/idmap/idmap/Makefile

1

```
*****
2058 Fri Dec 7 21:21:49 2018
new/usr/src/cmd/idmap/idmap/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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) 2007, 2010, Oracle and/or its affiliates. All rights reserved.
23 #

25 PROG =          idmap
26 CLIENTOBSJS =   idmap.o idmap_engine.o namemaps.o
27 # idmap_clnt.o
28 CLIENTSRCS =    $(CLIENTOBSJS:%.o=%.c)
29 POFILES =       $(CLIENTOBSJS:.o=.po)
30 OBJS =          $(CLIENTOBSJS)
31 SRCS =          $(CLIENTSRCS)
32 IDMAP_PROT_DIR = $(SRC)/head/rpcsvc

34 include ../../Makefile.cmd

36 CERRWARN +=     -_gcc=-Wno-parentheses
37 CERRWARN +=     -_gcc=-Wno-switch
38 CERRWARN +=     -_gcc=-Wno-unused-label
39 CERRWARN +=     -_gcc=-Wno-char-subscripts
40 CERRWARN +=     -_gcc=-Wno-unused-function
41 CERRWARN +=     -_gcc=-Wno-uninitialized
42 CERRWARN +=     -_gcc=-Wno-address

44 POFILE =        $(PROG)_all.po
45 LDLIBS +=       -lidmap -ladutils -lslldap -lldap
46 FILEMODE =      0555

48 INCS += -I. \
49 -I../../lib/libidmap/common \
50 -I../../lib/libadutils/common \
51 -I../../lib/libslldap/common \
52 -I$(IDMAP_PROT_DIR)

54 CFLAGS += $(CCVERBOSE)
55 $(OBJS) :=      CPPFLAGS += $(INCS) -D_REENTRANT
56 $(POFILE) :=    CPPFLAGS += $(INCS)

58 lint_SRCS :=    CPPFLAGS += $(INCS)

60 .KEEP_STATE:
```

new/usr/src/cmd/idmap/idmap/Makefile

2

```
62 all: $(PROG)

64 $(PROG): $(OBJS)
65 $(LINK.c) $(CCGDEBUG) -o $@ $(OBJS) $(LDLIBS)
65 $(LINK.c) -g -o $@ $(OBJS) $(LDLIBS)
66 $(POST_PROCESS)

68 $(POFILE): $(POFILES)
69 $(RM) $@
70 cat $(POFILES) > $@

72 install: all $(ROOTUSRSBINPROG)

74 clean:
75 $(RM) $(OBJS)

77 lint: lint_SRCS

79 LINTFLAGS += -erroff=E_NAME_DEF_NOT_USED2

81 include ../../Makefile.targ
```


new/usr/src/cmd/idmap/nltest/Makefile

1

```
*****
1598 Fri Dec 7 21:21:50 2018
new/usr/src/cmd/idmap/nltest/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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) 2007, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright 2014 Nexenta Systems, Inc. All rights reserved.
24 #

26 PROG = nltest
27 CLIENTOBS = nltest.o

29 CLIENTSRCS = $(CLIENTOBS:%.o=%.c)
30 POFILES = $(CLIENTOBS:.o=.po)
31 OBJS = $(CLIENTOBS)
32 SRCS = $(CLIENTSRCS)

34 include ../../Makefile.cmd

36 POFILE = $(PROG)_all.po
37 LDLIBS += -lads -luuid
38 FILEMODE = 0555

40 INCS += -I.

42 CFLAGS += $(CCVERBOSE)
43 $(OBJS) := CPPFLAGS += $(INCS) -D_REENTRANT
44 $(POFILE) := CPPFLAGS += $(INCS)

46 lint_SRCS := CPPFLAGS += $(INCS)

48 .KEEP_STATE:

50 all: $(PROG)

52 $(PROG): $(OBJS)
53 $(LINK.c) $(CCGDEBUG) -o $@ $(OBJS) $(LDLIBS)
53 $(LINK.c) -g -o $@ $(OBJS) $(LDLIBS)
54 $(POST_PROCESS)

56 $(POFILE): $(POFILES)
57 $(RM) $@
58 cat $(POFILES) > $@

60 install: all $(ROOTUSRSBINPROG)
```

new/usr/src/cmd/idmap/nltest/Makefile

2

```
62 clean:
63 $(RM) $(OBJJS)

65 lint: lint_SRCS

67 include ../../Makefile.targ
```

new/usr/src/cmd/idmap/test-getdc/Makefile

1

```
*****
1645 Fri Dec 7 21:21:50 2018
new/usr/src/cmd/idmap/test-getdc/Makefile
10051 cw(1) shouldn't have to translate -g
*****
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) 2007, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright 2014 Nexenta Systems, Inc. All rights reserved.
24 #

26 PROG =          test-getdc
27 OBJS =          getdc_main.o

29 SRCS = $(OBJS:%.o=%.c)
30 POFILES =       $(OBJS:%.o=%.po)

32 include ../../Makefile.cmd

34 POFILE =        $(PROG)_all.po

36 LDLIBS +=       -ladutils -lnsl -lumem
37 LDFLAGS +=      -R'$$ORIGIN/../lib'

39 INCS += -I. -I../../lib/libadutils/common

41 LINTFLAGS += -xerroff=E_NAME_DEF_NOT_USED2

43 CFLAGS += $(CCVERBOSE)
44 $(OBJS) := CPPFLAGS += $(INCS) -D_REENTRANT
45 $(POFILE) := CPPFLAGS += $(INCS)

47 lint_SRCS := CPPFLAGS += $(INCS)

49 .KEEP_STATE:

51 all: $(PROG)

53 $(PROG): $(OBJS) FRC
54 $(LINK.c) $(CCGDEBUG) -o $@ $(OBJS) $(LDLIBS)
54 $(LINK.c) -g -o $@ $(OBJS) $(LDLIBS)
55 $(POST_PROCESS)

57 $(POFILE): $(POFILES)
58 $(RM) $@
59 cat $(POFILES) > $@
```

new/usr/src/cmd/idmap/test-getdc/Makefile

2

```
61 install: all $(ROOTPROG)

63 clean:
64 $(RM) $(OBJS)

66 lint: lint_SRCS

68 include ../../Makefile.targ

70 FRC:
```

```

*****
2680 Fri Dec 7 21:21:51 2018
new/usr/src/cmd/mdb/sun4u/Makefile.kmdb
10051 cw(1) shouldn't have to translate -g
*****
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 PROMSRCS += \
27     prom_2path.c \
28     prom_devname.c \
29     prom_devtype.c \
30     prom_enter.c \
31     prom_env.c \
32     prom_exit.c \
33     prom_fb.c \
34     prom_getchar.c \
35     prom_init.c \
36     prom_inpath.c \
37     prom_interp.c \
38     prom_io.c \
39     prom_mmu.c \
40     prom_node.c \
41     prom_outpath.c \
42     prom_panic.c \
43     prom_phandle.c \
44     prom_printf.c \
45     prom_prop.c \
46     prom_putchar.c \
47     prom_sparc.c \
48     prom_stdin.c \
49     prom_stdout.c \
50     prom_string.c

52 KMDBSRCS += \
53     kaif.c \
54     kaif_activate.c \
55     kaif_start_isadep.c \
56     kmdb_dpi_isadep.c \
57     kmdb_fault_isadep.c \
58     kmdb_kdi_isadep.c \
59     kmdb_makecontext.c \
60     kmdb_promif_isadep.c \
61     kvm_isadep.c

```

```

63 KMDBML += \
64     kmdb_asmutil.s

66 KCTLSRCS += \
67     kctl_isadep.c

69 KCTLML += \
70     kctl_asm.s

72 SACPPFLAGS = -D__sparc

74 STANDBLIBS += \
75     ../../../../sparc/v9/libstandctf/libstandctf.so \
76     $(SRC)/lib/libumem/$(MACHDIR)/libstandumem.so \
77     ../../../../sparc/v9/libstand/libstand.a

79 KMDBLIBS = $(STANDBLIBS) ../../../../sparc/v9/mdb_ks/kmod/mdb_ks

81 MAPFILE_SOURCES = \
82     $(MAPFILE_SOURCES_COMMON) \
83     ../../../../sparc/kmdb/kmdb_dpi_isadep.h \
84     $(MAPFILE_SOURCES_$(MACH))

86 %.fth: $(SRC)/cmd/mdb/sparc/kmdb/%.fdbg
87     grep '^#' <$(F).c
88     grep -v '^#' <$(F).tmp
89     $(CC) $(CTFSTABSCFLAGS) $(CPPFLAGS) $(CCGDEBUG) -c -o $(F).o $(F).c
89     $(CC) $(CTFSTABSCFLAGS) $(CPPFLAGS) -g -c -o $(F).o $(F).c
90     $(CTFCONVERT) $(CTFCVTFLAGS) $(F).o
91     $(CTFSTABS) -t forth -i $(F).tmp -o $@ $(F).o
92     $(RM) $(F).c $(F).tmp $(F).o

94 %.o: ../../../../psm/promif/ieeel275/common/%.c
95     $(COMPILE.c) $(F)
96     $(CTFCONVERT_O)

98 %.o: ../../../../psm/promif/ieeel275/sun4/%.c
99     $(COMPILE.c) $(F)
100    $(CTFCONVERT_O)

102 %.o: ../../../../psm/promif/ieeel275/sun4u/%.c
103    $(COMPILE.c) $(F)
104    $(CTFCONVERT_O)

106 %.ln: ../../../../psm/promif/ieeel275/common/%.c
107    $(LINT.c) -c $(F)

109 %.ln: ../../../../psm/promif/ieeel275/sun4/%.c
110    $(LINT.c) -c $(F)

112 %.ln: ../../../../psm/promif/ieeel275/sun4u/%.c
113    $(LINT.c) -c $(F)

```

new/usr/src/cmd/mdb/sun4v/Makefile.kmdb

1

```
*****
2971 Fri Dec 7 21:21:51 2018
new/usr/src/cmd/mdb/sun4v/Makefile.kmdb
10051 cw(1) shouldn't have to translate -g
*****
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 PROMSRCS += \
27     prom_2path.c \
28     prom_devname.c \
29     prom_devtype.c \
30     prom_enter.c \
31     prom_env.c \
32     prom_exit.c \
33     prom_fb.c \
34     prom_getchar.c \
35     prom_init.c \
36     prom_inpath.c \
37     prom_interp.c \
38     prom_io.c \
39     prom_mmu.c \
40     prom_node.c \
41     prom_outpath.c \
42     prom_panic.c \
43     prom_phandle.c \
44     prom_printf.c \
45     prom_prop.c \
46     prom_putchar.c \
47     prom_reboot.c \
48     prom_sparc.c \
49     prom_stdin.c \
50     prom_stdout.c \
51     prom_string.c \
52     promif_emul.c \
53     promif_interp.c \
54     promif_io.c \
55     promif_mon.c \
56     promif_node.c \
57     promif_prop.c \
58     promif_reboot.c \
59     promif_stree.c

61 KMDBSRCS += \
```

new/usr/src/cmd/mdb/sun4v/Makefile.kmdb

2

```
62     kaif.c \
63     kaif_activate.c \
64     kaif_start_isadep.c \
65     kmdb_dpi_isadep.c \
66     kmdb_fault_isadep.c \
67     kmdb_kdi_isadep.c \
68     kmdb_makecontext.c \
69     kmdb_promif_isadep.c \
70     kvm_isadep.c

72 KMDBML += \
73     kmdb_asmutil.s

75 KCTLSRCS += \
76     kctl_isadep.c

78 KCTLML += \
79     kctl_asm.s

81 SACPPFLAGS = -D__sparc

83 STANDLIBS += \
84     ../../../../sparc/v9/libstandctf/libstandctf.so \
85     $(SRC)/lib/libumem/$(MACHDIR)/libstandumem.so \
86     ../../../../sparc/v9/libstand/libstand.a

88 KMDBLIBS = $(STANDLIBS) ../../../../sparc/v9/mdb_ks/kmod/mdb_ks

90 MAPFILE_SOURCES = \
91     $(MAPFILE_SOURCES_COMMON) \
92     ../../../../sparc/kmdb/kmdb_dpi_isadep.h \
93     $(MAPFILE_SOURCES_$(MACH))

95 %.fth: $(SRC)/cmd/mdb/sparc/kmdb/%.fdbg
96     grep '^#' <$(F).c
97     grep -v '^#' <$(F).tmp
98     $(CC) $(CTFSTABSCFLAGS) $(CPPFLAGS) $(CCGDEBUG) -c -o $(<F).o $(<F).c
98     $(CC) $(CTFSTABSCFLAGS) $(CPPFLAGS) -g -c -o $(<F).o $(<F).c
99     $(CTFCONVERT) $(CTFCVTFLAGS) $(<F).o
100     $(CTFSTABS) -t forth -i $(<F).tmp -o $@ $(<F).o
101     $(RM) $(<F).c $(<F).tmp $(<F).o

103 %.o: ../../../../psm/promif/ieee1275/common/%.c
104     $(COMPILE.c) $<
105     $(CTFCONVERT_O)

107 %.o: ../../../../psm/promif/ieee1275/sun4/%.c
108     $(COMPILE.c) $<
109     $(CTFCONVERT_O)

111 %.o: ../../../../psm/promif/ieee1275/sun4u/%.c
112     $(COMPILE.c) $<
113     $(CTFCONVERT_O)

115 %.o: ../../../../uts/sun4v/promif/%.c
116     $(COMPILE.c) $<
117     $(CTFCONVERT_O)

119 %.ln: ../../../../psm/promif/ieee1275/common/%.c
120     $(LINT.c) -c $<

122 %.ln: ../../../../psm/promif/ieee1275/sun4/%.c
123     $(LINT.c) -c $<

125 %.ln: ../../../../psm/promif/ieee1275/sun4u/%.c
126     $(LINT.c) -c $<
```

new/usr/src/cmd/mdb/sun4v/Makefile.kmdb

3

```
128 %.ln: ../../../../uts/sun4v/promif/%.c
129     $(LINT.c) -c $<
```

new/usr/src/cmd/print/printmgr/com/sun/admin/pm/server/Makefile

1

2970 Fri Dec 7 21:21:52 2018

new/usr/src/cmd/print/printmgr/com/sun/admin/pm/server/Makefile

10051 cw(1) shouldn't have to translate -g

```
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 #
26 #
27 # Makefile for Java Print Manager server
28 #
29 #
30 LIBRARY = libpMgr.a
31 VERS = .1
32 #
33 JNIHDR_FILES = com_sun_admin_pm_server_DoPrinterNS.h
34 #
35 OBJECTS = DoPrinterNS.o NS.o
36 #
37 CLASSFILES = Printer.class \
38 Host.class \
39 SysCommand.class \
40 NameService.class \
41 PrinterUtil.class \
42 DoPrinterView.class \
43 DoPrinterAdd.class \
44 DoPrinterDelete.class \
45 DoPrinterUtil.class \
46 DoPrinterMod.class \
47 DoPrinterNS.class \
48 Valid.class \
49 Test.class \
50 Debug.class \
51 PrinterDebug.class \
52 pmException.class \
53 pmAuthException.class \
54 pmAuthHostException.class \
55 pmCmdFailedException.class \
56 pmInternalErrorException.class \
57 pmHostNotPingableException.class \
58 pmNSNotConfiguredException.class \
59 pmMisc.class
60 #
61 JNICALSSFILES = DoPrinterNS.class
```

new/usr/src/cmd/print/printmgr/com/sun/admin/pm/server/Makefile

2

```
63 include $(SRC)/lib/Makefile.lib
64 #
65 SRCDIR = .
66 #
67 # There should be a mapfile here
68 MAPFILES =
69 #
70 CLASSPATH= $(SRC)/cmd/print/printmgr
71 #
72 JAVAFILES = $(CLASSFILES:.class=.java)
73 #
74 ROOTDIRS = $(ROOT)/usr/sadm/admin \
75 $(ROOT)/usr/sadm/admin/printmgr \
76 $(ROOT)/usr/sadm/admin/printmgr/lib
77 #
78 # override ROOTLIBDIR and ROOTLINKS
79 ROOTLIBDIR = $(ROOT)/usr/sadm/admin/printmgr/lib
80 ROOTLIBS= $(LIBS:%=$(ROOTLIBDIR)/%)
81 #
82 #
83 # Following variables define where to find header files
84 CPPFLAGS += -I$(JAVA_ROOT)/include -I$(JAVA_ROOT)/include/solaris -I.
85 #
86 CERRWARN += -gcc=-Wno-unused-variable
87 CERRWARN += -gcc=-Wno-uninitialized
88 #
89 LDLIBS += -lprint -lnsl -lsocket -lc
90 #
91 CLEANFILES= *.class $(LINTLIB) $(LINTOUT)
92 CLOBBERFILES= $(JNIHDR_FILES) $(LIBLINKS)
93 #
94 all: $(CLASSFILES) $(JNIHDR_FILES) $(DYNLIB)
95 #
96 install: all $(ROOTDIRS) $(ROOTLINKS)
97 #
98 #
99 # Build jni header file
100 # Use $@ instead of the "unreliable" $*
101 #
102 $(JNIHDR_FILES): $(JNICALSSFILES)
103 $(JAVA) -jni -classpath $(CLASSPATH) \
104 `echo $@ | sed 's/.$$$/' | tr _ .`
105 #
106 #
107 # Build standalone programs for testing
108 #
109 NS: NS.c
110 $(CC) NS.c $(CCGDEBUG) $(ILDOFF) -o NS -DMAIN=1 $(ENVLDLIBS1) $(ENVLIBS2)
111 $(CC) NS.c -g $(ILDOFF) -o NS -DMAIN=1 $(ENVLDLIBS1) $(ENVLIBS2) \
112 -lprint -lnsl -lsocket
113 lint: $(JNIHDR_FILES) lintcheck
114 #
115 cstyle:
116 cstyle $(SRCS)
117 #
118 jstyle:
119 jstyle $(JAVAFILES)
120 #
121 _msg:
122 #
123 $(ROOTDIRS):
124 $(INS.dir)
125 #
126 include $(SRC)/lib/Makefile.targ
```

```

*****
3601 Fri Dec 7 21:21:52 2018
new/usr/src/cmd/sort/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 #

27 #
28 # Debugging targets
29 # sort provides a number of debugging options to simplify failure analysis.
30 #
31 # debug          provides symbol information and sets DEBUG; includes
32 #                convert, invoke
33 # stats          builds binaries with statistics reporting enabled
34 # convert        builds convert binaries (converts input to collation
35 #                vectors)
36 # invoke         builds invoke binaries (allows inspection of options
37 #                parser outcome)
38 #
39 # Debugging #defines
40 # DEBUG          activate assertions; allow wider range of memory
41 #                settings (-S)
42 # DEBUG_FORCE_WIDE force all i/o through wide streams
43 # DEBUG_DISALLOW_MMAPP force all i/o through stdio or wide streams
44 # DEBUG_NO_CACHE_TEMP do not cache last sorted portion in memory; write all
45 #                data to temporary files before merge
46 #

48 PROG = sort
49 XPG4PROG = sort

51 BASE_OBJS = \
52   check.o \
53   fields.o \
54   initialize.o \
55   internal.o \
56   merge.o \
57   options.o \
58   streams.o \
59   streams_array.o \
60   streams_mmap.o \
61   streams_stdio.o \

```

```

62   streams_wide.o \
63   utility.o
64 OBJS = main.o $(BASE_OBJS)
65 INVOKE_OBJS = invoke.o $(BASE_OBJS)
66 CONVERT_OBJS = convert.o $(BASE_OBJS)
67 STATS_OBJS = main.o statistics.o $(BASE_OBJS)

69 XPG4OBJS = $(OBJS:%.o=xpg4_%.o)
70 SRCS = $(OBJS:%.o=../common/%.c)
71 LNTS = $(OBJS:%.o=%ln)
72 CLEANFILES = $(OBJS) $(XPG4OBJS) $(LNTS)

74 include ../../Makefile.cmd

76 SED =          sed
77 DCFILE =       $(PROG).dc

79 SPACEFLAG =
80 SPACEFLAG64 =

82 CFLAGS +=      $(CCVERBOSE) $(SORT_DEBUG)
83 CFLAGS64 +=    $(CCVERBOSE) $(SORT_DEBUG)
84 CPPFLAGS +=    -D_FILE_OFFSET_BITS=64
85 LINTFLAGS +=   -U_FILE_OFFSET_BITS

87 CERRWARN +=    -_gcc=-Wno-parentheses
88 CERRWARN +=    -_gcc=-Wno-uninitialized
89 CERRWARN +=    -_gcc=-Wno-unused-function

91 $(XPG4) :=     CFLAGS += -DXPG4

93 debug :=       SORT_DEBUG = $(CCGDEBUG) -DDEBUG $(ILDOFF)
94 debug :=       SORT_DEBUG = -g -DDEBUG $(ILDOFF)
95 debug :=       COPTFLAG =
96 debug :=       COPTFLAG64 =
97 stats :=       SORT_DEBUG = $(CCGDEBUG) -DSTATS -DDEBUG $(ILDOFF)
98 stats :=       SORT_DEBUG = -g -DSTATS -DDEBUG $(ILDOFF)
99 stats :=       COPTFLAG =
100 stats :=      COPTFLAG64 =

100 .KEEP_STATE :

102 .PARALLEL : $(OBJS) $(XPG4OBJS) $(LNTS)

104 all : $(PROG) $(XPG4)

106 debug : $(PROG) convert invoke

108 lint : $(LNTS)
109        $(LINT.c) $(LINTFLAGS) $(LNTS) $(LDLIBS)

111 clean :
112        $(RM) $(CLEANFILES)

114 include ../../Makefile.targ

116 # rules for $(PROG) and $(XPG4)

118 $(PROG) : $(OBJS)
119        $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
120        $(POST_PROCESS)

122 $(XPG4) : $(XPG4OBJS)
123        $(LINK.c) -o $@ $(XPG4OBJS) $(LDLIBS)
124        $(POST_PROCESS)

```

new/usr/src/cmd/sort/Makefile.com

3

```
126 invoke: $(INVOKE_OBJS)
127     $(LINK.c) -o $@ $(INVOKE_OBJS) $(LDLIBS)

129 convert: $(CONVERT_OBJS)
130     $(LINK.c) -o $@ $(CONVERT_OBJS) $(LDLIBS)

132 stats: $(STATS_OBJS)
133     $(LINK.c) -o $@ $(STATS_OBJS) $(LDLIBS)

135 %.o : ../common/%.c
136     $(COMPILE.c) -o $@ $<
137     $(POST_PROCESS_O)

139 xpg4_%.o : ../common/%.c
140     $(COMPILE.c) -o $@ $<
141     $(POST_PROCESS_O)

143 %.o : ../common/%.h types.h

145 xpg4_%.o : ../common/%.h types.h

147 %.ln: ../common/%.c
148     $(LINT.c) $(LINTFLAGS) -c $<
```


new/usr/src/cmd/stmfproxy/stmfproxy/Makefile

1

2161 Fri Dec 7 21:21:53 2018

new/usr/src/cmd/stmfproxy/stmfproxy/Makefile

10051 cw(1) shouldn't have to translate -g

```
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 #
26 # cmd/stmfproxy/stmfproxy/Makefile
27 #

29 PROG = svc-stmfproxy

31 COMMONBASE = ../../common

33 include ../../Makefile.cmd

35 DEMOFILES = \
36     stmfproxy.xml

38 DEMOBINFILES = \
39     $(PROG)

41 ROOTDEMODIR = $(ROOT)/usr/demo/comstar
42 ROOTDEMOBINDIR = $(ROOTDEMODIR)/bin
43 ROOTDEMOFILES = $(DEMOFILES:%=$(ROOTDEMODIR)/%)
44 ROOTDEMOBINFILES = $(DEMOBINFILES:%=$(ROOTDEMOBINDIR)/%)

46 PRODUCT= $(PROG)
47 SRCS= $(OBJS:%.o=%.c)
48 OBJS= stmfproxy.o
49 LLOBJS= $(OBJS:%.o=%.ll)
50 #POFILES= $(OBJS:%.o=%.po)
51 #POFILE= stmfproxy.po

53 $(ROOTDEMODIR)/stmfproxy.xml := FILEMODE = 0444

56 CCVERBOSE =
57 LDLIBS += -lnsl -lstmf -lstmfproxy
58 CPPFLAGS += -I. -I$(COMMONBASE)/cmdparse

60 # Uncomment the following to help with debugging
61 #CFLAGS += -g
```

new/usr/src/cmd/stmfproxy/stmfproxy/Makefile

2

```
62 #i386_COPTFLAG=
63 #i386_CCOPTFLAG=

60 .KEEP_STATE:

62 $(PROG): $(OBJS)
63     $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
64     $(POST_PROCESS)

66 LINTFLAGS += -erroff=E_FUNC_HAS_NO_RETURN_STMT
67 LINTFLAGS += -erroff=E_BAD_PTR_CAST_ALIGN

69 .PARALLEL: $(OBJS)

71 all: $(PROG)

73 clean:
74     $(RM) $(PROG) $(OBJS) $(LLOBJS)

76 lint: lint_SRCS

78 #$(POFILE): $(POFILES)
79 # $(RM) $@
80 # cat $(POFILES) > $@

82 install: all $(ROOTDEMOFILES) $(ROOTDEMOBINFILES)

84 $(ROOTDEMODIR) $(ROOTDEMOBINDIR):
85     $(INS.dir)

87 $(ROOTDEMODIR)/% $(ROOTDEMOBINDIR)/% : %
88     $(INS.file)

90 $(ROOTDEMOFILES): $(ROOTDEMODIR)

92 $(ROOTDEMOBINFILES): $(ROOTDEMOBINDIR)

94 check: $(CHKMANIFEST)
95     $(CSTYLE) -pPc $(SRCS:%=%)

98 include ../../Makefile.targ
```

```

*****
4356 Fri Dec 7 21:21:53 2018
new/usr/src/cmd/svc/svccfg/Makefile
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 2004, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright 2017 Joyent, Inc.
25 #
26 #
27 MYPROG =      svccfg
28 PROG =       $(MYPROG)
29 #
30 SRCS =       svccfg_main.c \
31              svccfg_engine.c \
32              svccfg_internal.c \
33              svccfg_libscf.c \
34              svccfg_tmpl.c \
35              svccfg_xml.c \
36              svccfg_help.c
37 #
38 LNTS =       $(SRCS:%.c=%.ln) \
39              manifest_find.ln \
40              manifest_hash.ln
41 #
42 MYOBSJS =    $(SRCS:%.c=%.o) \
43              svccfg_grammar.o \
44              svccfg_lex.o \
45              manifest_find.o \
46              manifest_hash.o \
47              notify_params.o
48 OBJS =       $(MYOBSJS)
49 #
50 POFILES =   $(SRCS:%.c=%.po) \
51              svccfg_grammar.po \
52              svccfg_lex.po \
53              ../common/manifest_find.po \
54              ../common/manifest_hash.po
55 #
56 include ../../Makefile.cmd
57 include ../../Makefile.ctf
58 #
59 POFILE =     $(PROG)_all.po
60 #
61 NATIVE_BUILD=$(POUND_SIGN)

```

```

62 $(NATIVE_BUILD)NOT_NATIVE=$(POUND_SIGN)
63 #
64 $(NATIVE_BUILD)PROG = $(MYPROG:%=%-native)
65 $(NATIVE_BUILD)OBSJS = $(MYOBSJS:%.o=%-native.o)
66 #
67 # svccfg has a name clash with main() and libl.so.1. However, svccfg must
68 # still export a number of "yy*" (libl) interfaces. Reduce all other symbols
69 # to local scope.
70 MAPFILES +=  $(MAPFILE.LEX) $(MAPFILE.NGB)
71 MAPOPTS =    $(MAPFILES:%=-M%)
72 #
73 MYCPPFLAGS = -I ../common -I$(ADJUNCT_PROTO)/usr/include/libxml2
74 CPPFLAGS += $(MYCPPFLAGS)
75 LDFLAGS +=  $(MAPOPTS)
76 #
77 CERRWARN +=  -_gcc=-Wno-unused-label
78 CERRWARN +=  -_gcc=-Wno-implicit-function-declaration
79 CERRWARN +=  -_gcc=-Wno-switch
80 CERRWARN +=  -_gcc=-Wno-uninitialized
81 CERRWARN +=  -_gcc=-Wno-unused-variable
82 CERRWARN +=  -_gcc=-Wno-parentheses
83 #
84 LFLAGS =    -t
85 YFLAGS =    -d
86 #
87 CLOBBERFILES += svccfg_lex.c svccfg_grammar.c svccfg_grammar.h \
88                $(MYPROG:%=%-native)
89 #
90 SVCCFG_EXTRA_LIBS = -lxml2 -lscf -ll -luutil -lumem -lmd5 -lnvpair
91 $(NOT_NATIVE)SVCCFG_EXTRA_LIBS += -ltecla
92 #
93 LIBSCF      = $(SRC)/lib/libscf
94 LIBTECLA    = $(SRC)/lib/libtecla      # just for the header
95 LIBUUTIL    = $(SRC)/lib/libuutil
96 #
97 debug := COPTFLAG = -g
98 #
99 #
100 #
101 #
102 $(NATIVE_BUILD)CC = $(NATIVECC)
103 $(NATIVE_BUILD)LD = $(NATIVELD)
104 $(NATIVE_BUILD)CFLAGS = $(NATIVE_CFLAGS)
105 $(NATIVE_BUILD)CPPFLAGS = \
106     -DNATIVE_BUILD \
107     $(MYCPPFLAGS) \
108     -I$(LIBSCF)/inc \
109     -I$(LIBTECLA) \
110     -I$(LIBUUTIL)/common
111 $(NATIVE_BUILD)LDFLAGS =
112 $(NATIVE_BUILD)LDLIBS = \
113     -L$(LIBUUTIL)/native -R $(LIBUUTIL)/native \
114     -L$(LIBSCF)/native -R $(LIBSCF)/native \
115     -L$(ADJUNCT_PROTO)/usr/lib -R$(ADJUNCT_PROTO)/usr/lib \
116     $(SVCCFG_EXTRA_LIBS)
117 #
118 svccfg_lex.o svccfg_grammar.o := CCVERBOSE =
119 #
120 svccfg_help.po := XGETFLAGS = -a
121 #
122 .KEEP_STATE:
123 .PARALLEL: $(OBJ) $(LNTS)
124 #
125 all debug: $(PROG)

```

```
127 native: FRC
128     @cd $(LIBUTIL)/native; pwd; $(MAKE) $(MFLAGS) install
129     @cd $(LIBSCF)/native; pwd; $(MAKE) $(MFLAGS) install
130     @NATIVE_BUILD= $(MAKE) $(MFLAGS) all

132 $(PROG): $(OBJS) $(MAPFILES)
133     $(LINK.c) -o $@ $(OBJS) $(LDLIBS)
134     $(POST_PROCESS)

136 $(POFILES): svccfg_grammar.h

138 $(POFILE): $(POFILES)
139     cat $(POFILES) > $(POFILE)

141 install: all $(ROOTUSRBINPROG)

143 svccfg_lex.c: svccfg.l svccfg_grammar.h
144     $(LEX) $(LFLAGS) svccfg.l > $@

146 svccfg_help.o: svccfg_grammar.h
147 svccfg_help-native.o: svccfg_grammar.h

149 svccfg_grammar.h svccfg_grammar.c: svccfg.y
150     $(YACC) $(YFLAGS) svccfg.y
151     @$ (MV) y.tab.h svccfg_grammar.h
152     @$ (MV) y.tab.c svccfg_grammar.c

154 clean: FRC
155     $(RM) $(MYOBJS) $(MYOBJS:%.o=%.native.o) $(LNTS)

157 lint: $(LNTS)
158     $(LINT.c) $(LINTFLAGS) $(LNTS) $(LDLIBS)

160 %-native.o: %.c
161     $(COMPILE.c) -o $@ $<
162     $(POST_PROCESS_O)

164 %-native.o: ../common/%.c
165     $(COMPILE.c) -o $@ $<
166     $(POST_PROCESS_O)

168 %.o: ../common/%.c
169     $(COMPILE.c) $(OUTPUT_OPTION) $<
170     $(POST_PROCESS_O)

172 %.ln: ../common/%.c
173     $(LINT.c) $(OUTPUT_OPTION) -c $<

175 include ../../Makefile.targ

177 FRC:
```

```

*****
2358 Fri Dec 7 21:21:54 2018
new/usr/src/cmd/tnf/prex/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
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 #
24 # Copyright 1989,2003 Sun Microsystems, Inc. All rights reserved.
25 # Use is subject to license terms.
26 #
27 # cmd/tnf/prex/Makefile.com
28 #

30 PROG=      prex

32 OBJS.c=    source.o      \
33            main.o        \
34            util.o        \
35            expr.o        \
36            spec.o        \
37            set.o         \
38            queue.o       \
39            cmd.o         \
40            new.o         \
41            list.o        \
42            fcn.o         \
43            prbk.o        \
44            help.o

46 OBJS.yl=   prexgram.o    \
47            prexlex.o

49 OBJS=      $(OBJS.yl) $(OBJS.c)

51 SRCS=      $(OBJS.c:%.o=./%.c) $(OBJS.yl:%.o=%c)

53 SRCS.yl =  $(OBJS.yl:%.o=%c)
54 CLEANFILES = $(SRCS.yl) y.tab.h

56 include ../../Makefile.cmd

58 POFILE=    prex.po
59 POFILES=   $(OBJS.c:%.o=%po)

61 #YFLAGS=   -d -t -v

```

```

62 YFLAGS=    -d
63 LFLAGS=    -v
64 # FOR normal makefile, uncomment the next line
65 LDLIBS +=  -lgen -ltnfctl -lelf -lc
66 # Uncomment the following line for a debug build
67 # COPTFLAG = -g -DDEBUG

67 CFLAGS +=  $(CCVERBOSE)
68 CERRWARN += -_gcc=-Wno-unused-label
69 CERRWARN += -_gcc=-Wno-unused-variable
70 CERRWARN += -_gcc=-Wno-parentheses
71 CERRWARN += -_gcc=-Wno-uninitialized

73 .KEEP_STATE:

75 .PARALLEL: $(OBJS)

77 all: $(PROG)

79 #OBJS can be built in parallel after all .c (and y.tab.h) are properly built
80 $(PROG): $(SRCS.yl) .WAIT $(OBJS)
81         $(LINK.c) $(OBJS) -o $@ $(LDLIBS)
82         $(POST_PROCESS)

84 #This also builds y.tab.h
85 prexgram.c: ../prexgram.y
86         $(YACC.y) ../prexgram.y
87         mv y.tab.c $@

89 prexlex.c: ../prexlex.l
90         $(RM) $@
91         $(LEX.l) ../prexlex.l > $@

93 #Use %.c in priority to ../%.c for prexgram.c and prexlec.c
94 %.o:      %.c
95           $(COMPILE.c) $<

97 %.o:      ../%.c
98           $(COMPILE.c) $<

101 $(ROOTBIN):
102         $(INS.dir)

104 $(POFILE): $(POFILES)
105         $(RM) $@
106         cat $(POFILES) > $@

108 clean:
109         $(RM) $(OBJS) $(CLEANFILES)

111 lint: $(OBJS)
112         $(LINT.c) $(SRCS)

114 include ../../Makefile.targ

```

new/usr/src/cmd/zlook/Makefile.com

1

1326 Fri Dec 7 21:21:54 2018

new/usr/src/cmd/zlook/Makefile.com

10051 cw(1) shouldn't have to translate -g

```
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 2009 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 PROG= zlook
27 SRCS= ../$(PROG).c

29 include ../../Makefile.cmd

31 CSTD= $(CSTD_GNU99)
32 C99LMODE= -Xc99=%all
33 CFLAGS += $(CCGDEBBUG) $(CCVERBOSE)
34 CFLAGS64 += $(CCGDEBBUG) $(CCVERBOSE)
35 CFLAGS += -g $(CCVERBOSE)
36 CFLAGS64 += -g $(CCVERBOSE)
37 CPPFLAGS += -D_LARGEFILE64_SOURCE=1 -D_REENTRANT $(INCS)

39 .KEEP_STATE:

41 all: $(PROG)

43 $(PROG): $(SRCS)
44     $(LINK.c) -o $(PROG) $(SRCS) $(LDLIBS)
45     $(POST_PROCESS)

47 clean:

49 lint: lint_SRCS

51 include ../../Makefile.targ
```

```

*****
26724 Fri Dec 7 21:21:54 2018
new/usr/src/lib/libc/sparc/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
24 # Copyright 2016 Joyent, Inc.
25 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
26 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
27 # Copyright 2018 Nexenta Systems, Inc.
28 #
29 #
30 LIBCDIR=      $(SRC)/lib/libc
31 LIB_PIC=      libc_pic.a
32 VERS=        .1
33 CPP=         /usr/lib/cpp
34 TARGET_ARCH= sparc
35 #
36 # objects are grouped by source directory
37 #
38 # Symbol capabilities objects.
39 EXTPICS=
40 $(LIBCDIR)/capabilities/sun4u/sparc/pics/symcap.o \
41 $(LIBCDIR)/capabilities/sun4u-opl/sparc/pics/symcap.o \
42 $(LIBCDIR)/capabilities/sun4u-us3-hwcap1/sparc/pics/symcap.o \
43 $(LIBCDIR)/capabilities/sun4u-us3-hwcap2/sparc/pics/symcap.o \
44 $(LIBCDIR)/capabilities/sun4v-hwcap1/sparc/pics/symcap.o \
45 $(LIBCDIR)/capabilities/sun4v-hwcap2/sparc/pics/symcap.o
46 #
47 # local objects
48 STRETS=
49 stret1.o
50 stret2.o
51 stret4.o
52 #
53 CRTOBSJ=
54 _ftou.o
55 cerror.o
56 cerror64.o
57 hwmuldiv.o
58 #
59 DYNOBJS=
60 _rtbootld.o

```

```

62 FPOBJS=
63 _D_cplx_div.o
64 _D_cplx_div_ix.o
65 _D_cplx_div_rx.o
66 _D_cplx_mul.o
67 _F_cplx_div.o
68 _F_cplx_div_ix.o
69 _F_cplx_div_rx.o
70 _F_cplx_mul.o
71 _Q_add.o
72 _Q_cmp.o
73 _Q_cmpe.o
74 _Q_cplx_div.o
75 _Q_cplx_div_ix.o
76 _Q_cplx_div_rx.o
77 _Q_cplx_lr_div.o
78 _Q_cplx_lr_div_ix.o
79 _Q_cplx_lr_div_rx.o
80 _Q_cplx_lr_mul.o
81 _Q_cplx_mul.o
82 _Q_div.o
83 _Q_dtoq.o
84 _Q_fcc.o
85 _Q_itoq.o
86 _Q_lltoq.o
87 _Q_mul.o
88 _Q_neg.o
89 _Q_qtod.o
90 _Q_qtoi.o
91 _Q_qtos.o
92 _Q_qtou.o
93 _Q_scl.o
94 _Q_set_except.o
95 _Q_sqrt.o
96 _Q_stoq.o
97 _Q_sub.o
98 _Q_ulltoq.o
99 _Q_utoq.o
100 __quad_mag.o
101 #
102 FPASMOBJS=
103 _Q_get_rp_rd.o
104 fpgetmask.o
105 fpgetrnd.o
106 fpgetsticky.o
107 fpsetmask.o
108 fpsetrnd.o
109 fpsetsticky.o
110 #
111 $(__GNUCC)FPASMOBJS +=
112 __quad.o
113 #
114 ATOMICOBJS=
115 atomic.o
116 #
117 CHACHAOBJS=
118 chacha.o
119 #
120 XATTROBJS=
121 xattr_common.o
122 #
123 COMOBJS=
124 bcmp.o
125 bcopy.o
126 bzero.o
127 bsearch.o

```

```

128      memccpy.o      \
129      qsort.o        \
130      strtol.o       \
131      strtoul.o      \
132      strtoll.o      \
133      strtoull.o     \

135 DTRACEOBSJ=      \
136      dtrace_data.o \

138 SECFLAGSOBSJ=    \
139      secflags.o    \

141 GENOBSJ=         \
142      _getsp.o       \
143      _xregs_clrptr.o \
144      abs.o           \
145      alloca.o       \
146      arc4random.o   \
147      arc4random_uniform.o \
148      ascii_strcasecmp.o \
149      byteorder.o    \
150      cuexit.o       \
151      ecvt.o         \
152      endian.o       \
153      errlst.o       \
154      gettxt.o       \
155      ladd.o         \
156      lmul.o         \
157      lock.o         \
158      lshifl.o       \
159      lsign.o        \
160      lsub.o         \
161      makectxt.o     \
162      memchr.o       \
163      memcmp.o       \
164      new_list.o     \
165      setjmp.o       \
166      siginfolst.o   \
167      siglongjmp.o  \
168      smt_pause.o    \
169      sparc_data.o   \
170      strchr.o       \
171      strcmp.o       \
172      strlcpy.o      \
173      strncmp.o      \
174      strncpy.o      \
175      strnlen.o      \
176      swapctxt.o     \
177      sync_instruction_memory.o \

179 # sysobjs that contain large-file interfaces
180 COMSYSOBSJ64=     \
181      fstatvfs64.o  \
182      getdents64.o  \
183      getrlimit64.o \
184      lseek64.o     \
185      mmap64.o      \
186      pread64.o     \
187      preadv64.o    \
188      pwrite64.o    \
189      pwritev64.o   \
190      setrlimit64.o \
191      statvfs64.o   \

```

```

193 SYSOBSJ64=

```

```

195 COMSYSOBSJ=      \
196      __clock_timer.o \
197      __getloadavg.o  \
198      __rusagesys.o   \
199      __signotify.o   \
200      __sigrt.o       \
201      __time.o        \
202      _lgrp_home_fast.o \
203      _lgrpsys.o      \
204      _nfssys.o       \
205      _portfs.o       \
206      _pset.o         \
207      _rpcsys.o       \
208      _sigaction.o    \
209      _so_accept.o    \
210      _so_bind.o      \
211      _so_connect.o   \
212      _so_getpeername.o \
213      _so_getsockname.o \
214      _so_getsockopt.o \
215      _so_listen.o    \
216      _so_recv.o      \
217      _so_recvfrom.o  \
218      _so_recvmsg.o   \
219      _so_send.o      \
220      _so_sendmsg.o   \
221      _so_sendto.o    \
222      _so_setsockopt.o \
223      _so_shutdown.o  \
224      _so_socket.o    \
225      _so_socketpair.o \
226      _sockconfig.o   \
227      acct.o         \
228      acl.o          \
229      adjtime.o      \
230      alarm.o        \
231      brk.o          \
232      chdir.o        \
233      chroot.o       \
234      cladm.o        \
235      close.o        \
236      execve.o       \
237      exit.o         \
238      facl.o         \
239      fchdir.o       \
240      fchroot.o      \
241      fdsync.o       \
242      fpathconf.o    \
243      fstatfs.o      \
244      fstatvfs.o     \
245      getcpuid.o     \
246      getdents.o     \
247      getegid.o      \
248      geteuid.o      \
249      getgid.o       \
250      getgroups.o    \
251      gethrtime.o    \
252      getitimer.o    \
253      getmsg.o       \
254      getpid.o       \
255      getpmsg.o      \
256      getppid.o     \
257      getrandom.o    \
258      getrlimit.o    \
259      getuid.o       \

```

```

260      gtty.o                \|
261      install_utrap.o       \|
262      ioctl.o               \|
263      kaio.o                \|
264      kill.o                \|
265      llseek.o              \|
266      lseek.o               \|
267      memcntl.o             \|
268      mincore.o             \|
269      mmap.o                \|
270      mmapobjsys.o         \|
271      modctl.o              \|
272      mount.o               \|
273      mprotect.o           \|
274      munmap.o              \|
275      nice.o                \|
276      ntp_adjtime.o         \|
277      ntp_gettime.o        \|
278      p_online.o            \|
279      pathconf.o            \|
280      pause.o               \|
281      pcsample.o            \|
282      pipe2.o               \|
283      pollsys.o             \|
284      pread.o               \|
285      preadv.o              \|
286      priocntlset.o         \|
287      processor_bind.o     \|
288      processor_info.o     \|
289      profil.o              \|
290      psecflagsset.o        \|
291      putmsg.o              \|
292      putpmsg.o             \|
293      pwrite.o              \|
294      pwritev.o             \|
295      read.o                \|
296      readv.o               \|
297      resolvepath.o         \|
298      seteguid.o            \|
299      setgid.o               \|
300      setgroups.o           \|
301      setitimer.o           \|
302      setreid.o             \|
303      setrlimit.o           \|
304      setuid.o               \|
305      sigaltstk.o           \|
306      sigprocmsk.o          \|
307      sigsendset.o          \|
308      sigsuspend.o          \|
309      statfs.o               \|
310      statvfs.o             \|
311      stty.o                \|
312      sync.o                \|
313      sysconfig.o           \|
314      sysfs.o               \|
315      sysinfo.o             \|
316      syslwp.o              \|
317      times.o               \|
318      ulimit.o              \|
319      umask.o                \|
320      umount2.o             \|
321      utssys.o              \|
322      uucopy.o              \|
323      vhangup.o             \|
324      waitid.o              \|
325      write.o               \|

```

```

326      writev.o              \|
327      yield.o               \|

329  SYSOBJS=                  \|
330      __clock_gettime.o     \|
331      __getcontext.o        \|
332      __lwp_mutex_unlock.o  \|
333      __stack_grow.o        \|
334      __uadmin.o            \|
335      door.o                \|
336      forkx.o               \|
337      forkallx.o            \|
338      gettimeofday.o        \|
339      ptrace.o               \|
340      syscall.o              \|
341      tls_get_addr.o        \|
342      uadmin.o               \|
343      umount.o              \|
344      uname.o                \|
345      vforkx.o              \|

347  # objects under $(LIBCDIR)/port which contain transitional large file interfaces
348  PORTGEN64=                 \|
349      _xftw64.o             \|
350      attropen64.o          \|
351      fts64.o                \|
352      ftw64.o                \|
353      mkstemp64.o           \|
354      nftw64.o              \|
355      tell64.o              \|
356      truncate64.o          \|

358  # objects from source under $(LIBCDIR)/port
359  PORTFP=                     \|
360      __flt_decim.o         \|
361      __flt_rounds.o        \|
362      __tbl_10_b.o          \|
363      __tbl_10_h.o          \|
364      __tbl_10_s.o          \|
365      __tbl_2_b.o           \|
366      __tbl_2_h.o           \|
367      __tbl_2_s.o           \|
368      __tbl_fdq.o           \|
369      __tbl_tens.o          \|
370      __x_power.o           \|
371      _base_sup.o           \|
372      aconvert.o            \|
373      decimal_bin.o         \|
374      double_decim.o        \|
375      econvert.o            \|
376      fconvert.o            \|
377      file_decim.o          \|
378      finite.o              \|
379      fp_data.o             \|
380      func_decim.o          \|
381      gconvert.o            \|
382      hex_bin.o             \|
383      ieee_globals.o        \|
384      pack_float.o          \|
385      sigfpe.o              \|
386      string_decim.o        \|
387      ashldi3.o             \|
388      ashrdi3.o             \|
389      cmpdi2.o              \|
390      divdi3.o              \|
391      floatdidf.o           \|

```



```

392 floatdisf.o \
393 floatundidf.o \
394 floatundisf.o \
395 lshrdi3.o \
396 moddi3.o \
397 muldi3.o \
398 qdivrem.o \
399 ucmpdi2.o \
400 udivdi3.o \
401 umoddi3.o \

403 PORTGEN= \
404 _env_data.o \
405 _ftoll.o \
406 _ftoull.o \
407 _xftw.o \
408 a64l.o \
409 abort.o \
410 addsev.o \
411 ascii_strncasecmp.o \
412 assert.o \
413 atexit.o \
414 atfork.o \
415 atof.o \
416 atoi.o \
417 atol.o \
418 atoll.o \
419 attrat.o \
420 attropen.o \
421 basename.o \
422 calloc.o \
423 catgets.o \
424 catopen.o \
425 cfgetispeed.o \
426 cfgetospeed.o \
427 cfree.o \
428 cfsetispeed.o \
429 cfsetospeed.o \
430 cftime.o \
431 clock.o \
432 closedir.o \
433 closefrom.o \
434 confstr.o \
435 crypt.o \
436 csetlen.o \
437 ctime.o \
438 ctime_r.o \
439 daemon.o \
440 default.o \
441 directio.o \
442 dirname.o \
443 div.o \
444 drand48.o \
445 dup.o \
446 env_data.o \
447 err.o \
448 errno.o \
449 euclen.o \
450 event_port.o \
451 execvp.o \
452 explicit_bzero.o \
453 fattach.o \
454 fdetach.o \
455 fdopendir.o \
456 ffs.o \
457 flock.o \

```

```

458 fls.o \
459 fmtmsg.o \
460 freezer.o \
461 ftime.o \
462 ftok.o \
463 fts.o \
464 ftw.o \
465 gcvt.o \
466 get_nprocs.o \
467 getauxv.o \
468 getcwd.o \
469 getdate_err.o \
470 getdtblsize.o \
471 getentropy.o \
472 getenv.o \
473 getexecname.o \
474 getgrnam.o \
475 getgrnam_r.o \
476 gethostid.o \
477 gethostname.o \
478 gethz.o \
479 getisax.o \
480 getloadavg.o \
481 getlogin.o \
482 getmntent.o \
483 getnetgrent.o \
484 getopt.o \
485 getopt_long.o \
486 getpagesize.o \
487 getpw.o \
488 getpwnam.o \
489 getpwnam_r.o \
490 getrusage.o \
491 getspent.o \
492 getspent_r.o \
493 getsubopt.o \
494 gettxt.o \
495 getusershell.o \
496 getut.o \
497 getutx.o \
498 getvfsent.o \
499 getwd.o \
500 getwidth.o \
501 getxby_door.o \
502 gtxt.o \
503 hsearch.o \
504 iconv.o \
505 imaxabs.o \
506 imaxdiv.o \
507 index.o \
508 initgroups.o \
509 insque.o \
510 isaexec.o \
511 isastream.o \
512 isatty.o \
513 killpg.o \
514 klpdlib.o \
515 l64a.o \
516 lckpwnam.o \
517 lconstants.o \
518 ldivide.o \
519 lexpl0.o \
520 lfind.o \
521 lfnt.o \
522 lfnt_log.o \
523 llabs.o \

```

```

524      lldiv.o          \
525      llog10.o         \
526      lltostr.o       \
527      localtime.o    \
528      lsearch.o       \
529      madvise.o        \
530      malloc.o         \
531      memalign.o      \
532      memmem.o         \
533      memset_s.o      \
534      mkdev.o          \
535      mkdtemp.o        \
536      mkfifo.o         \
537      mkstemp.o        \
538      mktemp.o         \
539      mlock.o          \
540      mlockall.o      \
541      mon.o            \
542      msync.o          \
543      munlock.o        \
544      munlockall.o    \
545      ndbm.o           \
546      nftw.o           \
547      nlspath_checks.o \
548      nsparse.o        \
549      nss_common.o     \
550      nss_dbdefs.o     \
551      nss_deffinder.o \
552      opendir.o        \
553      opt_data.o       \
554      perror.o         \
555      pfmt.o           \
556      pfmt_data.o     \
557      pfmt_print.o    \
558      pipe.o           \
559      plock.o          \
560      poll.o           \
561      posix_fadvise.o  \
562      posix_fallocate.o \
563      posix_madvise.o  \
564      posix_memalign.o \
565      priocntl.o       \
566      priv_str_xlate.o \
567      privlib.o        \
568      psecflags.o      \
569      psiginfo.o       \
570      psignal.o        \
571      pt.o             \
572      putpwent.o       \
573      putspent.o       \
574      raise.o          \
575      rand.o           \
576      random.o         \
577      rctlops.o        \
578      readdir.o        \
579      readdir_r.o      \
580      reallocarray.o   \
581      reallocarray.o   \
582      realpath.o       \
583      reboot.o         \
584      regexpr.o        \
585      remove.o         \
586      rewinddir.o     \
587      rindex.o         \
588      scandir.o        \
589      seekdir.o        \

```

```

590      select.o         \
591      set_constraint_handler_s.o \
592      setlabel.o       \
593      setpriority.o    \
594      settimeofday.o   \
595      sh_locks.o        \
596      sigflag.o         \
597      siglist.o         \
598      sigsend.o         \
599      sigsetops.o      \
600      ssignal.o         \
601      stack.o          \
602      stpcpy.o          \
603      stpncpy.o         \
604      str2sig.o         \
605      strcase_charmap.o \
606      strcat.o          \
607      strchrnul.o       \
608      strcspn.o         \
609      strdup.o          \
610      strerror.o        \
611      strlcat.o         \
612      strncat.o         \
613      strndup.o         \
614      strpbrk.o         \
615      strrchr.o         \
616      strsep.o          \
617      strsignal.o       \
618      strspn.o          \
619      strstr.o          \
620      strtod.o          \
621      strtointmax.o    \
622      strtok.o          \
623      strtok_r.o        \
624      strtonum.o        \
625      strtoumax.o      \
626      swab.o            \
627      swapctl.o         \
628      sysconf.o         \
629      syslog.o          \
630      tcdrain.o         \
631      tcflow.o          \
632      tcflush.o         \
633      tcgetattr.o       \
634      tcgetpgrp.o       \
635      tcgetsid.o        \
636      tcsendbreak.o     \
637      tcsetattr.o       \
638      tcsetpgrp.o       \
639      tell.o            \
640      telldir.o         \
641      tfind.o           \
642      time_data.o       \
643      time_gdata.o      \
644      timespec_get.o    \
645      tls_data.o        \
646      truncate.o        \
647      tsdalloc.o        \
648      tsearch.o         \
649      ttyname.o         \
650      ttyslot.o         \
651      ualarm.o          \
652      ucred.o           \
653      valloc.o          \
654      vlfmt.o           \
655      vpfmt.o           \

```

```

656      waitpid.o      \
657      walkstack.o   \
658      wdata.o        \
659      xgetwidth.o   \
660      xpg4.o         \
661      xpg6.o         \

663 PORTINET=        \
664      inet_lnaof.o   \
665      inet_makeaddr.o \
666      inet_network.o \
667      inet_ntoa.o    \
668      inet_ntop.o    \
669      inet_pton.o    \

671 PORTPRINT_W=     \
672      doprnt_w.o     \

674 PORTPRINT=       \
675      asprintf.o     \
676      doprnt.o       \
677      fprintf.o      \
678      printf.o        \
679      snprintf.o     \
680      sprintf.o       \
681      vfprintf.o     \
682      vprintf.o       \
683      vsnprintf.o    \
684      vsprintf.o     \
685      vwprintf.o     \
686      wprintf.o      \

688 # c89 variants to support 32-bit size of c89 u/intmax_t (32-bit libc only)
689 PORTPRINT_C89=    \
690      vfprintf_c89.o \
691      vprintf_c89.o  \
692      vsnprintf_c89.o \
693      vsprintf_c89.o \
694      vwprintf_c89.o \

696 PORTSTDIO_C89=   \
697      vscanf_c89.o  \
698      vwscanf_c89.o \

700 # portable stdio objects that contain large file interfaces.
701 # Note: fopen64 is a special case, as we build it small.
702 PORTSTDIO64=     \
703      fopen64.o     \
704      fpos64.o      \

706 PORTSTDIO_W=     \
707      doscan_w.o    \

709 PORTSTDIO=       \
710      __extensions.o \
711      __endopen.o    \
712      _filbuf.o      \
713      _findbuf.o     \
714      _flsbuf.o      \
715      _wrtchk.o      \
716      clearerr.o     \
717      ctermid.o      \
718      ctermid_r.o    \
719      cuserid.o       \
720      data.o         \
721      doscan.o       \

```

```

722      fdopen.o       \
723      feof.o         \
724      ferrord.o      \
725      fgetc.o        \
726      fgets.o        \
727      fileno.o       \
728      flockf.o       \
729      flush.o        \
730      fopen.o        \
731      fpos.o         \
732      fputc.o        \
733      fputs.o        \
734      fread.o        \
735      fseek.o        \
736      fseeko.o       \
737      ftell.o        \
738      ftello.o       \
739      fwrite.o       \
740      getc.o         \
741      getchar.o      \
742      getline.o      \
743      getpass.o      \
744      gets.o         \
745      getw.o         \
746      popen.o        \
747      putc.o         \
748      putchar.o     \
749      puts.o         \
750      putw.o         \
751      rewind.o       \
752      scanf.o        \
753      setbuf.o       \
754      setbuffer.o    \
755      setvbuf.o      \
756      system.o       \
757      tempnam.o      \
758      tmpfile.o      \
759      tmpnam_r.o     \
760      ungetc.o       \
761      mse.o          \
762      vscanf.o       \
763      vwscanf.o      \
764      wscanf.o       \

766 PORTI18N=        \
767      getwchar.o     \
768      putwchar.o     \
769      putws.o        \
770      strtows.o      \
771      wcsnlen.o      \
772      wcstoimax.o    \
773      wctol.o        \
774      wcstoul.o      \
775      wcs wcs.o      \
776      wscat.o        \
777      wchr.o         \
778      wscmp.o        \
779      wscopy.o       \
780      wscspn.o       \
781      wsdup.o        \
782      wslen.o        \
783      wscat.o        \
784      wscmp.o        \
785      wscpy.o        \
786      wspbrk.o       \
787      wsprintf.o     \

```

```

788      wsrchr.o      \
789      wsscanf.o    \
790      wssp.o       \
791      wstod.o      \
792      wstok.o      \
793      wstol.o      \
794      wstoll.o     \
795      wsxfrm.o     \
796      wmemchr.o    \
797      wmemcmp.o    \
798      wmemcpy.o    \
799      wmemmove.o   \
800      wmemset.o    \
801      wcsstr.o     \
802      gettext.o    \
803      gettext_real.o \
804      gettext_util.o \
805      gettext_gnu.o \
806      plural_parser.o \
807      wdresolve.o  \
808      _ctype.o     \
809      isascii.o    \
810      toascii.o    \
\
812 PORTI18N_COND= \
813      wcstol_longlong.o \
814      wcstoul_longlong.o \
\
816 PORTLOCALE= \
817      big5.o       \
818      btowc.o      \
819      collate.o    \
820      collcmp.o   \
821      euc.o        \
822      fnmatch.o   \
823      fgetwc.o    \
824      fgetws.o    \
825      fix_grouping.o \
826      fputwc.o    \
827      fputws.o   \
828      fwide.o     \
829      gb18030.o   \
830      gb2312.o    \
831      gbk.o        \
832      getdate.o   \
833      isdigit.o   \
834      iswctype.o  \
835      ldp.o        \
836      lmessages.o \
837      lnumeric.o  \
838      lmonetary.o \
839      localeimpl.o \
840      localeconv.o \
841      mbftowc.o   \
842      mblen.o     \
843      mbrlen.o    \
844      mbrtowc.o   \
845      mbsinit.o   \
846      mbsnrtowcs.o \
847      mbsrtowcs.o \
848      mbstowcs.o  \
849      mbtowc.o    \
850      mskanji.o   \
851      nextwctype.o \
852      nl_langinfo.o \
853      none.o      \

```

```

854      rune.o       \
855      runetype.o   \
856      setlocale.o  \
857      setrunelocale.o \
858      strcasecmp.o \
859      strcasestr.o \
860      strcoll.o    \
861      strfmon.o    \
862      strftime.o   \
863      strncasecmp.o \
864      strptime.o   \
865      strxfrm.o    \
866      table.o      \
867      timelocal.o  \
868      tolower.o    \
869      tolower.o    \
870      ungetwc.o    \
871      utf8.o       \
872      wctomb.o     \
873      wcscasecmp.o \
874      wcscoll.o    \
875      wcsftime.o   \
876      wcsnrtombs.o \
877      wcsrtombs.o \
878      wcstombs.o   \
879      wcswidth.o   \
880      wcsxfrm.o    \
881      wctob.o      \
882      wctomb.o     \
883      wctrans.o    \
884      wctype.o     \
885      wcwidth.o    \
886      wscoll.o     \
\
888 AIOOBS= \
889      aio.o        \
890      aio_alloc.o \
891      posix_aio.o  \
\
893 RTOBJS= \
894      clock_timer.o \
895      mqueue.o      \
896      pos4obj.o     \
897      sched.o       \
898      sem.o         \
899      shm.o         \
900      sigev_thread.o \
\
902 TPOOLOBS= \
903      thread_pool.o \
\
905 THREADSOBS= \
906      alloc.o       \
907      assfail.o     \
908      cancel.o      \
909      cll_thr.o     \
910      door_calls.o  \
911      tmem.o        \
912      pthr_attr.o   \
913      pthr_barrier.o \
914      pthr_cond.o   \
915      pthr_mutex.o  \
916      pthr_rwlock.o \
917      pthread.o     \
918      rwlock.o     \
919      scalls.o     \

```

```

920      sema.o          \
921      sigaction.o     \
922      spawn.o         \
923      synch.o         \
924      tdb_agent.o    \
925      thr.o           \
926      thread_interface.o \
927      tls.o           \
928      tsd.o           \

930 THREADSMACHOBJS= \
931      machdep.o     \

933 THREADSASMOBJS= \
934      asm_subr.o    \

936 UNICODEOBJS= \
937      u8_textprep.o \
938      uconv.o       \

940 UNWINDMACHOBJS= \
941      unwind.o      \

943 UNWINDASMOBJS= \
944      unwind_frame.o \

946 # objects that implement the transitional large file API
947 PORTSYS64= \
948      lockf64.o    \
949      stat64.o     \

951 PORTSYS= \
952      _autofssys.o \
953      access.o     \
954      acctctl.o    \
955      bsd_signal.o \
956      chmod.o      \
957      chown.o      \
958      corectl.o    \
959      epoll.o       \
960      eventfd.o    \
961      exacctsys.o  \
962      execl.o       \
963      execl.e.o    \
964      execv.o       \
965      fcntl.o       \
966      getpagesizes.o \
967      getpeerucred.o \
968      inst_sync.o  \
969      issetugid.o  \
970      label.o       \
971      link.o        \
972      lockf.o       \
973      lwp.o         \
974      lwp_cond.o   \
975      lwp_rwlock.o \
976      lwp_sigmask.o \
977      meminfosys.o \
978      mkdir.o      \
979      mknod.o       \
980      msgsys.o      \
981      nfssys.o      \
982      open.o        \
983      pgrpssys.o   \
984      posix_sigwait.o \
985      ppriv.o       \

```

```

986      psetsys.o    \
987      rctlsys.o    \
988      readlink.o   \
989      rename.o     \
990      sbrk.o        \
991      semsys.o     \
992      set_errno.o  \
993      sharefs.o    \
994      shmsys.o     \
995      sidsys.o     \
996      siginterrupt.o \
997      signal.o     \
998      signalfd.o   \
999      sigpending.o \
1000     sigstack.o   \
1001     stat.o        \
1002     symlink.o     \
1003     tasksys.o     \
1004     time.o         \
1005     time_util.o   \
1006     timerfd.o     \
1007     ucontext.o    \
1008     unlink.o      \
1009     ustat.o       \
1010     utimesys.o   \
1011     zone.o        \

1013 PORTREGEX= \
1014     glob.o     \
1015     regcmp.o   \
1016     regcomp.o  \
1017     regerror.o \
1018     regex.o    \
1019     regexec.o  \
1020     regfree.o  \
1021     wordexp.o  \

1023 PORTREGEX64= \
1024     glob64.o   \

1026 VALUES= values-Xa.o \

1028 MOSTOBJS= \
1029     $(STRETS) \
1030     $(CRTOBJS) \
1031     $(DYNOBJS) \
1032     $(FPOBJS) \
1033     $(FPASMOBJS) \
1034     $(ATOMICOBJS) \
1035     $(CHACHAOBJS) \
1036     $(XATTROBJS) \
1037     $(COMOBJS) \
1038     $(DTRACEOBJS) \
1039     $(GENOBJS) \
1040     $(PRFOBJS) \
1041     $(PORTFFP) \
1042     $(PORTFN) \
1043     $(PORTGEN64) \
1044     $(PORTI18N) \
1045     $(PORTI18N_COND) \
1046     $(PORTINET) \
1047     $(PORTLOCALE) \
1048     $(PORTPRINT) \
1049     $(PORTPRINT_C89) \
1050     $(PORTPRINT_W) \
1051     $(PORTREGEX) \

```

```

1052 $(PORTREGEX64) \
1053 $(PORTSTDIO) \
1054 $(PORTSTDIO64) \
1055 $(PORTSTDIO_C89) \
1056 $(PORTSTDIO_W) \
1057 $(PORTSYS) \
1058 $(PORTSYS64) \
1059 $(AIOBJS) \
1060 $(RTOBJS) \
1061 $(SECFLAGSOBJS) \
1062 $(TPOOLBJS) \
1063 $(THREADSOBJS) \
1064 $(THREADSMACHOBJS) \
1065 $(THREADSASMOBJS) \
1066 $(UNICODEOBJS) \
1067 $(UNWINDMACHOBJS) \
1068 $(UNWINDASMOBJS) \
1069 $(COMSYSOBJS) \
1070 $(SYSOBJS) \
1071 $(COMSYSOBJS64) \
1072 $(SYSOBJS64) \
1073 $(VALUES)

1075 TRACEOBJS= \
1076     plockstat.o

1078 # NOTE: libc.so.1 must be linked with the minimal crt1.o and crtn.o
1079 # modules whose source is provided in the $(SRC)/lib/crt directory.
1080 # This must be done because otherwise the Sun C compiler would insert
1081 # its own versions of these modules and those versions contain code
1082 # to call out to C++ initialization functions. Such C++ initialization
1083 # functions can call back into libc before thread initialization is
1084 # complete and this leads to segmentation violations and other problems.
1085 # Since libc contains no C++ code, linking with the minimal crt1.o and
1086 # crtn.o modules is safe and avoids the problems described above.
1087 OBJECTS= $(CRTI) $(MOSTOBJS) $(CRTN)
1088 CRTSRCS= ../../crt/sparc

1090 # include common library definitions
1091 include $(SRC)/lib/Makefile.lib

1093 # we need to override the default SONAME here because we might
1094 # be building a variant object (still libc.so.1, but different filename)
1095 SONAME = libc.so.1

1097 CFLAGS += $(CCVERBOSE)

1099 # This is necessary to avoid problems with calling _ex_unwind().
1100 # We probably don't want any inlining anyway.
1101 CFLAGS += -xinline=

1103 CERRWARN += -_gcc=-Wno-parentheses
1104 CERRWARN += -_gcc=-Wno-switch
1105 CERRWARN += -_gcc=-Wno-uninitialized
1106 CERRWARN += -_gcc=-Wno-unused-value
1107 CERRWARN += -_gcc=-Wno-unused-label
1108 CERRWARN += -_gcc=-Wno-unused-variable
1109 CERRWARN += -_gcc=-Wno-type-limits
1110 CERRWARN += -_gcc=-Wno-char-subscripts
1111 CERRWARN += -_gcc=-Wno-clobbered
1112 CERRWARN += -_gcc=-Wno-unused-function
1113 CERRWARN += -_gcc=-Wno-address

1115 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
1116 # enables ASSERT() checking in the threads portion of the library.
1117 # This is automatically enabled for DEBUG builds, not for non-debug builds.

```

```

1118 THREAD_DEBUG =
1119 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

1121 # Make string literals read-only to save memory.
1122 CFLAGS += $(XSTRCONST)

1124 ALTPICS= $(TRACEOBJS:%=pics/%)

1126 $(DYNLIB) := BUILD.SO = $(LD) -o $@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) $(EXTPICS)

1128 MAPFILES = $(LIBCDIR)/port/mapfile-vers

1130 CFLAGS += $(EXTN_CFLAGS)
1131 CPPFLAGS= -D_REENTRANT -Dsparc $(EXTN_CPPFLAGS) $(THREAD_DEBUG) \
1132           -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1133 ASFLAGS= $(EXTN_ASFLAGS) $(AS_PICFLAGS) -P -D__STDC__ -D_ASM $(CPPFLAGS)

1135 # As a favor to the dtrace syscall provider, libc still calls the
1136 # old syscall traps that have been obsoleted by the *at() interfaces.
1137 # Delete this to compile libc using only the new *at() system call traps
1138 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1140 # Inform the run-time linker about libc specialized initialization
1141 RTLDINFO = -z rtldinfo=tls_rtldinfo
1142 DYNFLAGS += $(RTLDINFO)

1144 # Force libc's internal references to be resolved immediately upon loading
1145 # in order to avoid critical region problems. Since almost all libc symbols
1146 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1147 DYNFLAGS += -znow

1149 DYNFLAGS += -e __rtboot
1150 DYNFLAGS += $(EXTN_DYNFLAGS)

1152 # Inform the kernel about the initial DTrace area (in case
1153 # libc is being used as the interpreter / runtime linker).
1154 DTRACE_DATA = -zdtrace=dtrace_data
1155 DYNFLAGS += $(DTRACE_DATA)

1157 # DTrace needs an executable data segment.
1158 MAPFILE.NED=

1160 BUILD.s= $(AS) $(ASFLAGS) $< -o $@

1162 # Override this top level flag so the compiler builds in its native
1163 # C99 mode. This has been enabled to support the complex arithmetic
1164 # added to libc.
1165 CSTD= $(CSTD_GNU99)

1167 # libc method of building an archive
1168 # The "$(GREP) -v ' L '" part is necessary only until
1169 # lorder is fixed to ignore thread-local variables.
1170 BUILD.AR= $(RM) $@ ; \
1171           $(AR) q $@ '$(LORDER) $(MOSTOBJS:%=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR

1173 # extra files for the clean target
1174 CLEANFILES= \
1175     $(LIBCDIR)/port/gen/errlst.c \
1176     $(LIBCDIR)/port/gen/new_list.c \
1177     assym.h \
1178     genassym \
1179     $(LIBCBASE)/crt/_rtld.s \
1180     $(LIBCBASE)/crt/_rtbootld.s \
1181     pics/_rtbootld.o \
1182     pics/crt1.o \
1183     pics/crtn.o \

```

```

1184      $(ALTPICS)
1186 CLOBBERFILES += $(LIB_PIC)

1188 # list of C source for lint
1189 SRCS=
1190      $(ATOMICOBJS:%.o=$(SRC)/common/atomic/%.c) \
1191      $(XATTROBJS:%.o=$(SRC)/common/xattr/%.c) \
1192      $(COMOBJS:%.o=$(SRC)/common/util/%.c) \
1193      $(DTRACEOBJS:%.o=$(SRC)/common/dtrace/%.c) \
1194      $(PORTFP:%.o=$(LIBCDIR)/port/fp/%.c) \
1195      $(PORTGEN:%.o=$(LIBCDIR)/port/gen/%.c) \
1196      $(PORTI18N:%.o=$(LIBCDIR)/port/il8n/%.c) \
1197      $(PORTINET:%.o=$(LIBCDIR)/port/inet/%.c) \
1198      $(PORTLOCALE:%.o=$(LIBCDIR)/port/locale/%.c) \
1199      $(PORTPRINT:%.o=$(LIBCDIR)/port/print/%.c) \
1200      $(PORTREGEX:%.o=$(LIBCDIR)/port/regex/%.c) \
1201      $(PORTSTDIO:%.o=$(LIBCDIR)/port/stdio/%.c) \
1202      $(PORTSYS:%.o=$(LIBCDIR)/port/sys/%.c) \
1203      $(AIOBJS:%.o=$(LIBCDIR)/port/aio/%.c) \
1204      $(RTOBJS:%.o=$(LIBCDIR)/port/rt/%.c) \
1205      $(SECFLAGSOBJS:%.o=$(SRC)/common/secflags/%.c) \
1206      $(TPOOLBJS:%.o=$(LIBCDIR)/port/tpool/%.c) \
1207      $(THREADSOBJS:%.o=$(LIBCDIR)/port/threads/%.c) \
1208      $(THREADSMACHOBJS:%.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1209      $(UNICODEOBJS:%.o=$(SRC)/common/unicode/%.c) \
1210      $(UNWINDMACHOBJS:%.o=$(LIBCDIR)/port/unwind/%.c) \
1211      $(FPOBJS:%.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1212      $(LIBCBASE)/crt/_ftou.c \
1213      $(LIBCBASE)/gen/_xregs_clrptr.c \
1214      $(LIBCBASE)/gen/byteorder.c \
1215      $(LIBCBASE)/gen/ecvt.c \
1216      $(LIBCBASE)/gen/endian.c \
1217      $(LIBCBASE)/gen/getctxt.c \
1218      $(LIBCBASE)/gen/lmul.c \
1219      $(LIBCBASE)/gen/makeectxt.c \
1220      $(LIBCBASE)/gen/siginfofst.c \
1221      $(LIBCBASE)/gen/siglongjmp.c \
1222      $(LIBCBASE)/gen/swapctxt.c \
1223      $(LIBCBASE)/sys/ptrace.c \
1224      $(LIBCBASE)/sys/uadmin.c

```

```

1226 # conditional assignments
1227 $(DYNLIB) := CRTI = crti.o
1228 $(DYNLIB) := CRTN = crtn.o

```

```

1230 # Files which need the threads .il inline template

```

```

1231 TIL=
1232      aio.o \
1233      alloc.o \
1234      assfail.o \
1235      atexit.o \
1236      atfork.o \
1237      cancel.o \
1238      door_calls.o \
1239      err.o \
1240      errno.o \
1241      getctxt.o \
1242      lwp.o \
1243      ma.o \
1244      machdep.o \
1245      posix_aio.o \
1246      pthr_attr.o \
1247      pthr_barrier.o \
1248      pthr_cond.o \
1249      pthr_mutex.o \

```

```

1250      pthr_rwlock.o \
1251      pthread.o \
1252      rand.o \
1253      rwalk.o \
1254      scalls.o \
1255      sched.o \
1256      sema.o \
1257      sigaction.o \
1258      sigev_thread.o \
1259      spawn.o \
1260      stack.o \
1261      swapctxt.o \
1262      synch.o \
1263      tdb_agent.o \
1264      thr.o \
1265      thread_interface.o \
1266      thread_pool.o \
1267      tls.o \
1268      tsd.o \
1269      unwind.o

1271 $(TIL:%=pics%) := CFLAGS += $(LIBCBASE)/threads/sparc.il

1273 # special kludge for inlines with 'cas':
1274 pics/rwalk.o pics/synch.o pics/lwp.o pics/door_calls.o := \
1275      sparc_CFLAGS += -_gcc=-Wa,-xarch=v8plus

1277 # Files in port/fp subdirectory that need base.il inline template
1278 IL=
1279      __flt_decim.o \
1280      decimal_bin.o

1282 $(IL:%=pics%) := CFLAGS += $(LIBCBASE)/fp/base.il

1284 # Files in fp subdirectory which need __quad.il inline template
1285 QIL=
1286      __Q_add.o \
1287      __Q_cmp.o \
1288      __Q_cmpe.o \
1289      __Q_div.o \
1290      __Q_dtoq.o \
1291      __Q_fcc.o \
1292      __Q_mul.o \
1293      __Q_qtod.o \
1294      __Q_qtoi.o \
1295      __Q_qtos.o \
1296      __Q_qtou.o \
1297      __Q_sqrt.o \
1298      __Q_stoq.o \
1299      __Q_sub.o

```

```

1301 $(QIL:%=pics%) := CFLAGS += $(LIBCDIR)/$(MACH)/fp/__quad.il
1302 pics/_Q%.o := sparc_COPTFLAG = -xO4 -dalign
1303 pics/__quad%.o := sparc_COPTFLAG = -xO4 -dalign

```

```

1305 # large-file-aware components that should be built large

```

```

1307 $(COMSYSOBS64:%=pics%) := \
1308      CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1310 $(SYSOBJS64:%=pics%) := \
1311      CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1313 $(PORTGEN64:%=pics%) := \
1314      CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

```

```

1316 $(PORTREGEX64:%=pics/%) := \
1317     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1319 $(PORTSTDIO64:%=pics/%) := \
1320     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1322 $(PORTSYS64:%=pics/%) := \
1323     CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1325 $(PORTSTDIO_W:%=pics/%) := \
1326     CPPFLAGS += -D_WIDE

1328 $(PORTPRINT_W:%=pics/%) := \
1329     CPPFLAGS += -D_WIDE

1331 # printf/scanf functions to support c89-sized intmax_t variables
1332 $(PORTPRINT_C89:%=pics/%) := \
1333     CPPFLAGS += -D_C89_INTMAX32

1335 $(PORTSTDIO_C89:%=pics/%) := \
1336     CPPFLAGS += -D_C89_INTMAX32

1338 $(PORTI18N_COND:%=pics/%) := \
1339     CPPFLAGS += -D_WCS_LOGLONG

1341 pics/arc4random.o := CPPFLAGS += -I$(SRC)/common/crypto/chacha

1343 # Files which need extra optimization
1344 pics/getenv.o := sparc_COPTFLAG = -xO4

1346 .KEEP_STATE:

1348 all: $(LIBS) $(LIB_PIC)

1350 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1351 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1352 lint := LINTFLAGS += -mn

1354 lint:
1355     @echo $(LINT.c) ... $(LDLIBS)
1356     @$$(LINT.c) $(SRCS) $(LDLIBS)

1358 $(LINTLIB):= SRCS=$(LIBCDIR)/port/l1ib-lc
1359 $(LINTLIB):= CPPFLAGS += -D_MSE_INT_H
1360 $(LINTLIB):= LINTFLAGS=-nvx

1362 # object files that depend on inline template
1363 $(TIL:%=pics/%) : $(LIBCBASE)/threads/sparc.il
1364 $(IL:%=pics/%) : $(LIBCBASE)/fp/base.il
1365 $(QL:%=pics/%) : $(LIBCDIR)/$(MACH)/fp/__quad.il

1367 # include common libc targets
1368 include $(LIBCDIR)/Makefile.targ

1370 # We need to strip out all CTF and DOF data from the static library
1371 $(LIB_PIC) := DIR = pics
1372 $(LIB_PIC): pics $$ (PICS)
1373     $(BUILD.AR)
1374     $(MCS) -d -n .SUNW_ctf $$@ > /dev/null 2>&1
1375     $(MCS) -d -n .SUNW_dof $$@ > /dev/null 2>&1
1376     $(AR) -ts $$@ > /dev/null
1377     $(POST_PROCESS_A)

1379 # special cases
1380 $(STRETS:%=pics/%) : $(LIBCBASE)/crt/stret.s
1381     $(AS) $(ASFLAGS) -DSTRET$(@F:stret%.o=) $(LIBCBASE)/crt/stret.s -o $$@

```

```

1382     $(POST_PROCESS_O)

1384 $(LIBCBASE)/crt/_rtbootld.s: $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.
1385     $(CC) $(CPPFLAGS) $(CTF_FLAGS) -O -S $(C_PICFLAGS) \
1386     $(LIBCBASE)/crt/_rtld.c -o $(LIBCBASE)/crt/_rtld.s
1387     $(CAT) $(LIBCBASE)/crt/_rtboot.s $(LIBCBASE)/crt/_rtld.s > $$@
1388     $(RM) $(LIBCBASE)/crt/_rtld.s

1390 # partially built from C source
1391 pics/_rtbootld.o: $(LIBCBASE)/crt/_rtbootld.s
1392     $(AS) $(ASFLAGS) $(LIBCBASE)/crt/_rtbootld.s -o $$@
1393     $(CTFCONVERT_O)

1395 ASSYMDEP_OBJS= \
1396     _lwp_mutex_unlock.o \
1397     _stack_grow.o \
1398     asm_subr.o \
1399     setjmp.o \
1400     smt_pause.o \
1401     tls_get_addr.o \
1402     unwind_frame.o \
1403     vforkx.o

1405 $(ASSYMDEP_OBJS:%=pics/%) := CPPFLAGS += -I.

1407 $(ASSYMDEP_OBJS:%=pics/%) : assym.h

1409 # assym.h build rules

1411 assym.h := CFLAGS += $(CCGDEBUG)
1411 assym.h := CFLAGS += -g

1413 GENASSYM_C = $(LIBCDIR)/$(MACH)/genassym.c

1415 genassym: $(GENASSYM_C)
1416     $(NATIVECC) $(NATIVE_CFLAGS) -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc \
1417     $(CPPFLAGS.native) -o $$@ $(GENASSYM_C)

1419 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in

1421 assym.h: $(OFFSETS) genassym
1422     $(OFFSETS_CREATE) <$(OFFSETS) >$$@
1423     ./genassym >>$$@

1425 # derived C source and related explicit dependencies
1426 $(LIBCDIR)/port/gen/errlst.c + \
1427 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/
1428     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist

1430 pics/errlst.o: $(LIBCDIR)/port/gen/errlst.c

1432 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c

```



```

*****
25261 Fri Dec 7 21:21:55 2018
new/usr/src/lib/libc/sparcv9/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright 2016 Gary Mills
24 # Copyright (c) 1989, 2010, Oracle and/or its affiliates. All rights reserved.
25 # Copyright (c) 2015, Joyent, Inc. All rights reserved.
26 # Copyright (c) 2013, OmniTI Computer Consulting, Inc. All rights reserved.
27 # Copyright 2013 Garrett D'Amore <garrett@damore.org>
28 # Copyright 2018 Nexenta Systems, Inc.
29 #
30 #
31 LIBCDIR=      $(SRC)/lib/libc
32 LIB_PIC=      libc_pic.a
33 VERS=        .1
34 CPP=         /usr/lib/cpp
35 TARGET_ARCH= sparc
36 #
37 # objects are grouped by source directory
38 #
39 # Symbol capabilities objects.
40 EXTPICS=      \
41 $(LIBCDIR)/capabilities/sun4u/sparcv9/pics/symcap.o \
42 $(LIBCDIR)/capabilities/sun4u-op1/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
47 #
48 # local objects
49 STRETS=
50 #
51 CRTOBSJ=      \
52 __align_cpy_2.o \
53 __align_cpy_4.o \
54 __align_cpy_8.o \
55 __ftou.o \
56 cerror.o
57 #
58 DYNOBJS=
59 #
60 FPOBJS=       \
61 _D_cplx_div.o \

```

```

62 _D_cplx_div_ix.o \
63 _D_cplx_div_rx.o \
64 _D_cplx_mul.o \
65 _F_cplx_div.o \
66 _F_cplx_div_ix.o \
67 _F_cplx_div_rx.o \
68 _F_cplx_mul.o \
69 _Q_add.o \
70 _Q_cmp.o \
71 _Q_cmpe.o \
72 _Q_cplx_div.o \
73 _Q_cplx_div_ix.o \
74 _Q_cplx_div_rx.o \
75 _Q_cplx_lr_div.o \
76 _Q_cplx_lr_div_ix.o \
77 _Q_cplx_lr_div_rx.o \
78 _Q_cplx_lr_mul.o \
79 _Q_cplx_mul.o \
80 _Q_div.o \
81 _Q_dtoq.o \
82 _Q_fcc.o \
83 _Q_itoq.o \
84 _Q_mul.o \
85 _Q_neg.o \
86 _Q_qtod.o \
87 _Q_qtoi.o \
88 _Q_qtos.o \
89 _Q_qtou.o \
90 _Q_scl.o \
91 _Q_sqrt.o \
92 _Q_stoq.o \
93 _Q_sub.o \
94 _Q_utoq.o
95 #
96 FPOBJS64=    \
97 _Qp_qtox.o \
98 _Qp_qtoux.o \
99 _Qp_xtou.o \
100 _Qp_uxtoq.o \
101 __dtoul.o \
102 __ftoul.o
103 #
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
113 #
114 $(__GNUC)FPASMOBJS += \
115 __quad.o
116 #
117 ATOMICOBJS=  \
118 atomic.o
119 #
120 CHACHAOBJS=  \
121 chacha.o
122 #
123 XATTROBJS=   \
124 xattr_common.o
125 #
126 COMOBJS=     \
127 bcmp.o \

```

```

128      bcopy.o           \
129      bsearch.o        \
130      bzero.o          \
131      memccpy.o        \
132      qsort.o          \
133      strtol.o         \
134      strtoul.o        \
135      strtoll.o        \
136      strtoull.o       \

138 GENOBSJ=            \
139      _getsp.o         \
140      _xregs_clrptr.o \
141      abs.o            \
142      alloca.o         \
143      arc4random.o     \
144      arc4random_uniform.o \
145      ascii_strcasecmp.o \
146      byteorder.o      \
147      cuexit.o         \
148      ecvt.o           \
149      endian.o         \
150      getctxt.o        \
151      lock.o           \
152      makectxt.o       \
153      memchr.o         \
154      memcmp.o         \
155      new_list.o       \
156      setjmp.o         \
157      siginfolst.o     \
158      siglongjmp.o     \
159      smt_pause.o      \
160      sparc_data.o     \
161      strchr.o         \
162      strcmp.o         \
163      strlcpy.o        \
164      strncmp.o        \
165      strncpy.o        \
166      strnlen.o        \
167      swapctxt.o      \
168      sync_instruction_memory.o \

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

174 SYSOBSJS64=

176 COMSYSOBSJS=        \
177      __clock_timer.o \
178      __getloadavg.o  \
179      __rusagesys.o   \
180      __signotify.o   \
181      __sigrt.o       \
182      __time.o         \
183      _lgrp_home_fast.o \
184      _lgrpsys.o       \
185      _nfssys.o        \
186      _portfs.o        \
187      _pset.o          \
188      _rpcsys.o        \
189      _sigaction.o     \
190      _so_accept.o     \
191      _so_bind.o       \
192      _so_connect.o    \
193      _so_getpeername.o \

```

```

194      _so_getsockname.o \
195      _so_getsockopt.o  \
196      _so_listen.o     \
197      _so_recv.o       \
198      _so_recvfrom.o   \
199      _so_recvmsg.o    \
200      _so_send.o       \
201      _so_sendmsg.o    \
202      _so_sendto.o     \
203      _so_setsockopt.o \
204      _so_shutdown.o   \
205      _so_socket.o     \
206      _so_socketpair.o \
207      _sockconfig.o    \
208      acct.o           \
209      acl.o            \
210      adjtime.o        \
211      alarm.o          \
212      brk.o            \
213      chdir.o          \
214      chroot.o         \
215      cladm.o          \
216      close.o          \
217      execve.o         \
218      exit.o           \
219      faccl.o          \
220      fchdir.o         \
221      fchroot.o        \
222      fdsync.o         \
223      fpathconf.o     \
224      fstatfs.o        \
225      fstatvfs.o      \
226      getcpuid.o       \
227      getdents.o       \
228      getegid.o        \
229      geteuid.o        \
230      getgid.o         \
231      getgroups.o      \
232      gethrtime.o     \
233      getitimer.o     \
234      getmsg.o         \
235      getpid.o         \
236      getpmsg.o        \
237      getppid.o        \
238      getrandom.o      \
239      getrlimit.o     \
240      getuid.o         \
241      gtty.o           \
242      install_utrap.o \
243      ioctl.o         \
244      kaio.o          \
245      kill.o          \
246      llseek.o        \
247      lseek.o         \
248      memcntl.o       \
249      mincore.o       \
250      mmap.o          \
251      mmapobjsys.o    \
252      modctl.o        \
253      mount.o         \
254      mprotect.o      \
255      munmap.o        \
256      nice.o          \
257      ntp_adjtime.o   \
258      ntp_gettime.o   \
259      p_online.o      \

```

```

260 pathconf.o \
261 pause.o \
262 pcsample.o \
263 pipe2.o \
264 pollsys.o \
265 pread.o \
266 preadv.o \
267 priocntlset.o \
268 processor_bind.o \
269 processor_info.o \
270 profil.o \
271 psecflagsset.o \
272 putmsg.o \
273 putpmsg.o \
274 pwrite.o \
275 pwritev.o \
276 read.o \
277 readv.o \
278 resolvepath.o \
279 seteguid.o \
280 setgid.o \
281 setgroups.o \
282 setitimer.o \
283 setreid.o \
284 setrlimit.o \
285 setuid.o \
286 sigaltstk.o \
287 sigprocmsk.o \
288 sigsendset.o \
289 sigsuspend.o \
290 statfs.o \
291 statvfs.o \
292 stty.o \
293 sync.o \
294 sysconfig.o \
295 sysfs.o \
296 sysinfo.o \
297 syslwp.o \
298 times.o \
299 ulimit.o \
300 umask.o \
301 umount2.o \
302 utssys.o \
303 uucopy.o \
304 vhangup.o \
305 waitid.o \
306 write.o \
307 writev.o \
308 yield.o \

310 SYSOBSJ= \
311 __clock_gettime.o \
312 __getcontext.o \
313 __uadmin.o \
314 __lwp_mutex_unlock.o \
315 __stack_grow.o \
316 door.o \
317 forkx.o \
318 forkallx.o \
319 gettimeofday.o \
320 sparcv9_utrap_install.o \
321 syscall.o \
322 tls_get_addr.o \
323 uadmin.o \
324 umount.o \
325 uname.o \

```

```

326 vforkx.o \

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

332 # objects from source under $(LIBCDIR)/port
333 PORTFP= \
334 __flt_decim.o \
335 __flt_rounds.o \
336 __tbl_10_b.o \
337 __tbl_10_h.o \
338 __tbl_10_s.o \
339 __tbl_2_b.o \
340 __tbl_2_h.o \
341 __tbl_2_s.o \
342 __tbl_fdq.o \
343 __tbl_tens.o \
344 __x_power.o \
345 __base_sup.o \
346 aconvert.o \
347 decimal_bin.o \
348 double_decim.o \
349 econvert.o \
350 fconvert.o \
351 file_decim.o \
352 finite.o \
353 fp_data.o \
354 func_decim.o \
355 gconvert.o \
356 hex_bin.o \
357 ieee_globals.o \
358 pack_float.o \
359 sigfpe.o \
360 string_decim.o \

362 PORTGEN= \
363 __env_data.o \
364 __xftw.o \
365 a64l.o \
366 abort.o \
367 addsev.o \
368 ascii_strncasecmp.o \
369 assert.o \
370 atexit.o \
371 atfork.o \
372 atof.o \
373 atoi.o \
374 atol.o \
375 atoll.o \
376 attrat.o \
377 attropen.o \
378 basename.o \
379 calloc.o \
380 catgets.o \
381 catopen.o \
382 cfgetispeed.o \
383 cfgetospeed.o \
384 cfree.o \
385 cfsetispeed.o \
386 cfsetospeed.o \
387 cftime.o \
388 clock.o \
389 closedir.o \
390 closefrom.o \
391 confstr.o \

```

```

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

```

```

458      getxby_door.o   \|
459      gtxt.o          \|
460      hsearch.o       \|
461      iconv.o         \|
462      imaxabs.o       \|
463      imaxdiv.o       \|
464      index.o         \|
465      initgroups.o    \|
466      insque.o        \|
467      isaexec.o       \|
468      isastream.o     \|
469      isatty.o        \|
470      killpg.o        \|
471      klpdlib.o       \|
472      l64a.o         \|
473      lckpwdf.o      \|
474      lconstants.o   \|
475      ldivide.o       \|
476      lexpl0.o        \|
477      lfind.o         \|
478      lfmt.o          \|
479      lfmt_log.o      \|
480      lldiv.o         \|
481      llog10.o        \|
482      lltostr.o       \|
483      lmath.o         \|
484      localtime.o    \|
485      lsearch.o       \|
486      madvise.o       \|
487      malloc.o        \|
488      memalign.o      \|
489      memmem.o        \|
490      memset_s.o      \|
491      mkdev.o         \|
492      mkdtemp.o       \|
493      mkfifo.o        \|
494      mkstemp.o       \|
495      mktemp.o        \|
496      mlock.o         \|
497      mlockall.o     \|
498      mon.o           \|
499      msync.o         \|
500      munlock.o       \|
501      munlockall.o   \|
502      ndbm.o          \|
503      nftw.o          \|
504      nlspath_checks.o \|
505      nsparse.o       \|
506      nss_common.o    \|
507      nss_dbdefs.o    \|
508      nss_deffinder.o \|
509      opendir.o       \|
510      opt_data.o      \|
511      perror.o        \|
512      pfmt.o          \|
513      pfmt_data.o     \|
514      pfmt_print.o    \|
515      pipe.o          \|
516      plock.o         \|
517      poll.o          \|
518      posix_fadvise.o \|
519      posix_fallocate.o \|
520      posix_madvise.o \|
521      posix_memalign.o \|
522      priocntl.o      \|
523      priv_str_xlate.o \|

```

```

524     privlib.o           \
525     psecflags.o        \
526     psiginfo.o         \
527     psignal.o          \
528     pt.o               \
529     putpwent.o         \
530     putsptent.o        \
531     raise.o            \
532     rand.o             \
533     random.o           \
534     rctlops.o          \
535     readdir.o          \
536     readdir_r.o        \
537     reallocarray.o     \
538     reallocarray.o     \
539     realpath.o         \
540     reboot.o           \
541     regexpr.o          \
542     remove.o           \
543     rewinddir.o        \
544     rindex.o           \
545     scandir.o          \
546     seekdir.o          \
547     select.o           \
548     set_constraint_handler_s.o \
549     setlabel.o         \
550     setpriority.o       \
551     settimeofday.o     \
552     sh_locks.o         \
553     sigflag.o          \
554     siglist.o          \
555     sigsend.o          \
556     sigsetops.o        \
557     signal.o           \
558     stack.o            \
559     stpcpy.o           \
560     stpncpy.o          \
561     str2sig.o           \
562     strcase_uchar.o    \
563     strcat.o           \
564     strchrnul.o        \
565     strcspn.o          \
566     strdup.o           \
567     strerror.o         \
568     strlcat.o          \
569     strncat.o          \
570     strndup.o          \
571     strpbrk.o          \
572     strrchr.o          \
573     strsep.o           \
574     strsignal.o        \
575     strspn.o           \
576     strstr.o           \
577     strtod.o           \
578     strtointmax.o      \
579     strtok.o           \
580     strtok_r.o         \
581     strtonum.o         \
582     strtoumax.o        \
583     swab.o             \
584     swapctl.o          \
585     sysconf.o          \
586     syslog.o           \
587     tcdrain.o          \
588     tcflow.o           \
589     tcflush.o          \

```

```

590     tcgetattr.o        \
591     tcgetpgrp.o        \
592     tcgetsid.o         \
593     tcsendbreak.o     \
594     tcsetattr.o        \
595     tcsetpgrp.o        \
596     tell.o             \
597     telldir.o          \
598     tfind.o            \
599     time_data.o        \
600     time_gdata.o      \
601     timespec_get.o     \
602     tls_data.o         \
603     truncate.o         \
604     tsdalloc.o         \
605     tsearch.o          \
606     ttyname.o          \
607     ttyslot.o          \
608     ualarm.o           \
609     ucred.o            \
610     valloc.o           \
611     vlfmt.o            \
612     vpfmt.o            \
613     waitpid.o          \
614     walkstack.o        \
615     wdata.o            \
616     xgetwidth.o        \
617     xpg4.o             \
618     xpg6.o             \
620 PORTINET=             \
621     inet_lnaof.o       \
622     inet_makeaddr.o   \
623     inet_network.o    \
624     inet_ntoa.o        \
625     inet_ntop.o        \
626     inet_pton.o        \
628 PORTPRINT_W=         \
629     doprnt_w.o         \
631 PORTPRINT=           \
632     asprintf.o         \
633     doprnt.o           \
634     fprintf.o          \
635     printf.o           \
636     snprintf.o         \
637     sprintf.o          \
638     vfprintf.o         \
639     vprintf.o          \
640     vsnprintf.o        \
641     vsprintf.o         \
642     vwprintf.o         \
643     wprintf.o          \
645 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
646 # This macro should ALWAYS be empty; native APIs are already 'large file'.
647 PORTSTDIO64=
649 PORTSTDIO_W=         \
650     doscan_w.o         \
652 PORTSTDIO=           \
653     __extensions.o     \
654     __endopen.o        \
655     __filbuf.o         \

```

```

656     _findbuf.o      \|
657     _flsbuf.o      \|
658     _wrtchk.o      \|
659     clearerr.o     \|
660     ctermid.o      \|
661     ctermid_r.o    \|
662     cuserid.o      \|
663     data.o         \|
664     doscan.o       \|
665     fdopen.o       \|
666     feof.o         \|
667     ferror.o       \|
668     fgetc.o        \|
669     fgets.o        \|
670     fileno.o       \|
671     flockf.o       \|
672     flush.o        \|
673     fopen.o        \|
674     fpos.o         \|
675     fputc.o        \|
676     fputs.o        \|
677     fread.o        \|
678     fseek.o        \|
679     fseeko.o       \|
680     ftell.o        \|
681     ftello.o       \|
682     fwrite.o       \|
683     getc.o         \|
684     getchar.o      \|
685     getline.o      \|
686     getpass.o      \|
687     gets.o         \|
688     getw.o         \|
689     popen.o        \|
690     putc.o         \|
691     putchar.o     \|
692     puts.o         \|
693     putw.o         \|
694     rewind.o       \|
695     scanf.o        \|
696     setbuf.o       \|
697     setbuffer.o   \|
698     setvbuf.o     \|
699     system.o       \|
700     tmpnam.o       \|
701     tmpfile.o     \|
702     tmpnam_r.o    \|
703     ungetc.o       \|
704     mse.o          \|
705     vscanf.o      \|
706     vwscanf.o     \|
707     wscanf.o      \|

709 PORTI18N=      \|
710     getwchar.o    \|
711     putwchar.o    \|
712     puts.o        \|
713     strtows.o     \|
714     wcsnlen.o     \|
715     wcstoimax.o   \|
716     wcstol.o      \|
717     wcstoul.o     \|
718     wcswcs.o      \|
719     wscat.o       \|
720     wschr.o       \|
721     wscmp.o       \|

```

```

722     wscpy.o        \|
723     wscspn.o       \|
724     wsdup.o        \|
725     wslen.o        \|
726     wsncat.o       \|
727     wsncmp.o       \|
728     wsncpy.o       \|
729     wspbrk.o       \|
730     wsprintf.o     \|
731     wsrchr.o       \|
732     wsscanf.o      \|
733     wsspn.o        \|
734     wstod.o        \|
735     wstok.o        \|
736     wstol.o        \|
737     wstoll.o       \|
738     wsxfrm.o       \|
739     wmemchr.o      \|
740     wmemcmp.o      \|
741     wmemcpy.o      \|
742     wmemmove.o     \|
743     wmemset.o      \|
744     wcstr.o        \|
745     gettext.o      \|
746     gettext_real.o \|
747     gettext_util.o \|
748     gettext_gnu.o  \|
749     plural_parser.o \|
750     wdresolve.o    \|
751     _ctype.o       \|
752     isascii.o      \|
753     toascii.o      \|

755 PORTI18N_COND= \|
756     wcstol_longlong.o \|
757     wcstoul_longlong.o \|

759 PORTLOCALE=     \|
760     big5.o         \|
761     btowc.o        \|
762     collate.o      \|
763     collcmp.o      \|
764     euc.o          \|
765     fnmatch.o      \|
766     fgetwc.o       \|
767     fgetws.o       \|
768     fix_grouping.o \|
769     fputwc.o       \|
770     fputws.o       \|
771     fwide.o        \|
772     gb18030.o      \|
773     gb2312.o       \|
774     gbk.o          \|
775     getdate.o      \|
776     isdigit.o      \|
777     iswctype.o     \|
778     ldpair.o       \|
779     lmessages.o    \|
780     lnumeric.o     \|
781     lmonetary.o    \|
782     localeconv.o   \|
783     localeimpl.o   \|
784     mbftowc.o      \|
785     mblen.o        \|
786     mbrlen.o       \|
787     mbrtowc.o      \|

```

```

788     mbsinit.o           \
789     mbsnrto wcs.o       \
790     mbsrtowcs.o        \
791     mbstowcs.o         \
792     mbtowc.o           \
793     mskanji.o          \
794     nextwctype.o       \
795     nl_langinfo.o      \
796     none.o             \
797     rune.o             \
798     runetype.o         \
799     setlocale.o        \
800     setrunelocale.o   \
801     strcasecmp.o       \
802     strcasestr.o       \
803     strcoll.o          \
804     strfmon.o          \
805     strftime.o         \
806     strncasecmp.o      \
807     strptime.o         \
808     strxfrm.o          \
809     table.o            \
810     timelocal.o        \
811     tolower.o          \
812     towlower.o         \
813     ungetwc.o          \
814     utf8.o             \
815     wcrto mb.o         \
816     wcscasecmp.o      \
817     wcscoll.o          \
818     wcsftime.o         \
819     wcsnrto mbs.o      \
820     wcsrtombs.o        \
821     wcstombs.o         \
822     wcswidth.o         \
823     wcsxfrm.o          \
824     wctob.o            \
825     wctomb.o           \
826     wctrans.o          \
827     wctype.o           \
828     wcwidth.o          \
829     wscoll.o           \
\
831 AIOOBJS=              \
832     aio.o               \
833     aio_alloc.o        \
834     posix_aio.o        \
\
836 RTOBJS=               \
837     clock_timer.o      \
838     mqueue.o           \
839     pos4obj.o          \
840     sched.o            \
841     sem.o              \
842     shm.o              \
843     sigev_thread.o    \
\
845 SECFLAGSOBJS=         \
846     secflags.o         \
\
848 TPOOLOBJS=            \
849     thread_pool.o     \
\
851 THREADSOBJS=          \
852     alloc.o            \
853     assfail.o          \

```

```

854     cli_thr.o           \
855     cancel.o           \
856     door_calls.o       \
857     tmem.o             \
858     pthr_attr.o        \
859     pthr_barrier.o     \
860     pthr_cond.o        \
861     pthr_mutex.o       \
862     pthr_rwlock.o      \
863     pthread.o          \
864     rwlock.o           \
865     scalls.o           \
866     sema.o             \
867     sigaction.o        \
868     spawn.o            \
869     synch.o            \
870     tdb_agent.o        \
871     thr.o              \
872     thread_interface.o \
873     tls.o              \
874     tsd.o              \
\
876 THREADSMACHOBJS=     \
877     machdep.o          \
\
879 THREADSASMOBJS=      \
880     asm_subr.o         \
\
882 UNICODOBJS=          \
883     u8_textprep.o     \
884     uconv.o           \
\
886 UNWINDMACHOBJS=     \
887     unwind.o          \
\
889 UNWINDASMOBJS=      \
890     unwind_frame.o    \
\
892 # Preserved solely to ease maintenance of 32-bit and 64-bit library builds
893 # This macro should ALWAYS be empty; native APIs are already 'large file'.
894 PORTSYS64=
\
896 PORTSYS=              \
897     _autofssys.o      \
898     access.o           \
899     acctctl.o          \
900     bsd_signal.o       \
901     chmod.o            \
902     chown.o            \
903     corectl.o          \
904     eventfd.o          \
905     epoll.o            \
906     exacctsyst.o       \
907     execl.o            \
908     execl.o            \
909     execv.o            \
910     fcntl.o            \
911     getpagesizes.o     \
912     getpeerucred.o     \
913     inst_sync.o        \
914     issetugid.o        \
915     label.o            \
916     link.o             \
917     lockf.o            \
918     lwp.o              \
919     lwp_cond.o         \

```

```

920 lwp_rwlock.o \
921 lwp_sigmask.o \
922 meminfosys.o \
923 mkdir.o \
924 mknod.o \
925 msgsys.o \
926 nfssys.o \
927 open.o \
928 pgrpsys.o \
929 posix_sigwait.o \
930 ppriv.o \
931 psetsys.o \
932 rctlsys.o \
933 readlink.o \
934 rename.o \
935 sbrk.o \
936 semsys.o \
937 set_errno.o \
938 sharefs.o \
939 shmsys.o \
940 sidsys.o \
941 siginterrupt.o \
942 signal.o \
943 signalfd.o \
944 sigpending.o \
945 sigstack.o \
946 stat.o \
947 symlink.o \
948 tasksys.o \
949 time.o \
950 time_util.o \
951 timerfd.o \
952 ucontext.o \
953 unlink.o \
954 ustat.o \
955 utimesys.o \
956 zone.o \

958 PORTREGEX= \
959 glob.o \
960 regcmp.o \
961 regcomp.o \
962 regerror.o \
963 regex.o \
964 regexec.o \
965 regfree.o \
966 wordexp.o \

968 VALUES= values-Xa.o \

970 MOSTOBSJ= \
971 $(STRETS) \
972 $(CRTOBSJ) \
973 $(DYNOBJS) \
974 $(FPOBJS) \
975 $(FPOBJS64) \
976 $(FPASMOBJS) \
977 $(ATOMICOBJS) \
978 $(CHACHAOBJS) \
979 $(XATTROBJS) \
980 $(COMOBJS) \
981 $(GENOBJS) \
982 $(PRFOBJS) \
983 $(PORTFP) \
984 $(PORTGEN) \
985 $(PORTGEN64) \

```

```

986 $(PORTI18N) \
987 $(PORTI18N_COND) \
988 $(PORTINET) \
989 $(PORTLOCALE) \
990 $(PORTPRINT) \
991 $(PORTPRINT_W) \
992 $(PORTREGEX) \
993 $(PORTSTDIO) \
994 $(PORTSTDIO64) \
995 $(PORTSTDIO_W) \
996 $(PORTSYS) \
997 $(PORTSYS64) \
998 $(AIOOBJS) \
999 $(RTOBJS) \
1000 $(SECFLAGSOBJS) \
1001 $(TPOOLBJS) \
1002 $(THREADSOBJS) \
1003 $(THREADSMACHOBJS) \
1004 $(THREADSASMOBJS) \
1005 $(UNICODEOBJS) \
1006 $(UNWINDMACHOBJS) \
1007 $(UNWINDASMOBJS) \
1008 $(COMSYSOBJS) \
1009 $(YSOBSJ) \
1010 $(COMSYSOBJS64) \
1011 $(YSOBSJ64) \
1012 $(VALUES) \

1014 TRACEOBSJ= \
1015 plockstat.o \

1017 # NOTE: libc.so.1 must be linked with the minimal crt.i.o and crtn.o
1018 # modules whose source is provided in the $(SRC)/lib/crt directory.
1019 # This must be done because otherwise the Sun C compiler would insert
1020 # its own versions of these modules and those versions contain code
1021 # to call out to C++ initialization functions. Such C++ initialization
1022 # functions can call back into libc before thread initialization is
1023 # complete and this leads to segmentation violations and other problems.
1024 # Since libc contains no C++ code, linking with the minimal crt.i.o and
1025 # crtn.o modules is safe and avoids the problems described above.
1026 OBJECTS= $(CRTI) $(MOSTOBSJ) $(CRTN)
1027 CRTSRCS= ../../crt/sparc

1029 # include common library definitions
1030 include $(SRC)/lib/Makefile.lib
1031 include $(SRC)/lib/Makefile.lib.64

1033 # we need to override the default SONAME here because we might
1034 # be building a variant object (still libc.so.1, but different filename)
1035 SONAME = libc.so.1

1037 CFLAGS64 += $(CCVERBOSE)

1039 # This is necessary to avoid problems with calling _ex_unwind().
1040 # We probably don't want any inlining anyway.
1041 CFLAGS64 += -xinline=

1043 CERRWARN += -_gcc=-Wno-parentheses
1044 CERRWARN += -_gcc=-Wno-switch
1045 CERRWARN += -_gcc=-Wno-uninitialized
1046 CERRWARN += -_gcc=-Wno-unused-value
1047 CERRWARN += -_gcc=-Wno-unused-label
1048 CERRWARN += -_gcc=-Wno-unused-variable
1049 CERRWARN += -_gcc=-Wno-type-limits
1050 CERRWARN += -_gcc=-Wno-char-subscripts
1051 CERRWARN += -_gcc=-Wno-clobbered

```



```

1052 CERRWARN += _gcc=-Wno-unused-function
1053 CERRWARN += _gcc=-Wno-address

1055 # Setting THREAD_DEBUG = -DTHREAD_DEBUG (make THREAD_DEBUG=-DTHREAD_DEBUG ...)
1056 # enables ASSERT() checking in the threads portion of the library.
1057 # This is automatically enabled for DEBUG builds, not for non-debug builds.
1058 THREAD_DEBUG =
1059 $(NOT_RELEASE_BUILD)THREAD_DEBUG = -DTHREAD_DEBUG

1061 # Make string literals read-only to save memory.
1062 CFLAGS64 += $(XSTRCONST)

1064 ALTPICS= $(TRACEOBS:%=pics/%)

1066 $(DYNLIB) := BUILD.SO = $(LD) -o $@ -G $(DYNFLAGS) $(PICS) $(ALTPICS) $(EXTPICS)

1068 MAPFILES = $(LIBCDIR)/port/mapfile-vers

1070 sparcv9_C_PICFLAGS= $(sparcv9_C_BIGPICFLAGS)
1071 CFLAGS64 += $(EXTN_CFLAGS)
1072 CPPFLAGS= -D_REENTRANT -Dsparc $(EXTN_CPPFLAGS) $(THREAD_DEBUG) \
1073 -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc $(CPPFLAGS.master)
1074 ASFLAGS= $(EXTN_ASFLAGS) $(AS_BIGPICFLAGS) -P -D__STDC__ -D__ASM -D__sparc
1075 $(sparcv9_AS_XARCH)

1077 # As a favor to the dtrace syscall provider, libc still calls the
1078 # old syscall traps that have been obsoleted by the *at() interfaces.
1079 # Delete this to compile libc using only the new *at() system call traps
1080 CPPFLAGS += -D_RETAIN_OLD_SYSCALLS

1082 # Inform the run-time linker about libc specialized initialization
1083 RTLDINFO = -z rtldinfo=tls_rtldinfo
1084 DYNFLAGS += $(RTLDINFO)

1086 # Force libc's internal references to be resolved immediately upon loading
1087 # in order to avoid critical region problems. Since almost all libc symbols
1088 # are marked 'protected' in the mapfiles, this is a minimal set (15 to 20).
1089 DYNFLAGS += -znov

1091 DYNFLAGS += $(EXTN_DYNFLAGS)

1093 BUILD.s= $(AS) $(ASFLAGS) $< -o $@

1095 # Override this top level flag so the compiler builds in its native
1096 # C99 mode. This has been enabled to support the complex arithmetic
1097 # added to libc.
1098 CSTD= $(CSTD_GNU99)

1100 # libc method of building an archive
1101 # The "$(GREP) -v ' L '" part is necessary only until
1102 # lorder is fixed to ignore thread-local variables.
1103 BUILD.AR= $(RM) $@ ; \
1104 $(AR) q $@ '$(LORDER) $(MOSTOBS:%=$(DIR)/%) | $(GREP) -v ' L ' | $(TSOR

1106 # extra files for the clean target
1107 CLEANFILES= \
1108 $(LIBCDIR)/port/gen/errlst.c \
1109 $(LIBCDIR)/port/gen/new_list.c \
1110 assym.h \
1111 genassym \
1112 pics/crti.o \
1113 pics/crtn.o \
1114 $(ALTPICS)

1116 CLOBBERFILES += $(LIB_PIC)

```

```

1118 # list of C source for lint
1119 SRCS= \
1120 $(ATOMICOBJS:%.o=$(SRC)/common/atomic/%.c) \
1121 $(XATTROBJS:%.o=$(SRC)/common/xattr/%.c) \
1122 $(COMOBS:%.o=$(SRC)/common/util/%.c) \
1123 $(PORTFP:%.o=$(LIBCDIR)/port/fp/%.c) \
1124 $(PORTGEN:%.o=$(LIBCDIR)/port/gen/%.c) \
1125 $(PORTI18N:%.o=$(LIBCDIR)/port/i18n/%.c) \
1126 $(PORTINET:%.o=$(LIBCDIR)/port/inet/%.c) \
1127 $(PORTLOCALE:%.o=$(LIBCDIR)/port/locale/%.c) \
1128 $(PORTPRINT:%.o=$(LIBCDIR)/port/print/%.c) \
1129 $(PORTREGEX:%.o=$(LIBCDIR)/port/regex/%.c) \
1130 $(PORTSTDIO:%.o=$(LIBCDIR)/port/stdio/%.c) \
1131 $(PORTSYS:%.o=$(LIBCDIR)/port/sys/%.c) \
1132 $(AIOOBJS:%.o=$(LIBCDIR)/port/aio/%.c) \
1133 $(RTOBJS:%.o=$(LIBCDIR)/port/rt/%.c) \
1134 $(SECFLAGSOBJS:%.o=$(SRC)/common/secflags/%.c) \
1135 $(TPOOLSOBJS:%.o=$(LIBCDIR)/port/tpool/%.c) \
1136 $(THREADSOBJS:%.o=$(LIBCDIR)/port/threads/%.c) \
1137 $(THREADSMACHOBJS:%.o=$(LIBCDIR)/$(MACH)/threads/%.c) \
1138 $(UNICODEOBJS:%.o=$(SRC)/common/unicode/%.c) \
1139 $(UNWINDMACHOBJS:%.o=$(LIBCDIR)/port/unwind/%.c) \
1140 $(FPOBJS:%.o=$(LIBCDIR)/$(MACH)/fp/%.c) \
1141 $(FPOBJS64:%.o=$(LIBCBASE)/fp/%.c) \
1142 $(LIBCBASE)/crt/_ftou.c \
1143 $(LIBCBASE)/gen/_xregs_clrptr.c \
1144 $(LIBCBASE)/gen/byteorder.c \
1145 $(LIBCBASE)/gen/endian.c \
1146 $(LIBCBASE)/gen/ecvt.c \
1147 $(LIBCBASE)/gen/getctxt.c \
1148 $(LIBCBASE)/gen/makectxt.c \
1149 $(LIBCBASE)/gen/signfolst.c \
1150 $(LIBCBASE)/gen/siglongjmp.c \
1151 $(LIBCBASE)/gen/swapctxt.c

1153 # conditional assignments
1154 $(DYNLIB) := CRTI = crt.i.o
1155 $(DYNLIB) := CRTN = crtn.o

1157 # Files which need the threads .il inline template
1158 TIL= \
1159 aio.o \
1160 alloc.o \
1161 assfail.o \
1162 atexit.o \
1163 atfork.o \
1164 cancel.o \
1165 door_calls.o \
1166 err.o \
1167 errno.o \
1168 getctxt.o \
1169 lwp.o \
1170 ma.o \
1171 machdep.o \
1172 posix_aio.o \
1173 pthr_attr.o \
1174 pthr_barrier.o \
1175 pthr_cond.o \
1176 pthr_mutex.o \
1177 pthr_rwlock.o \
1178 pthread.o \
1179 rand.o \
1180 rwlock.o \
1181 scalls.o \
1182 sched.o \
1183 sema.o \

```

```

1184 sigaction.o \
1185 sigev_thread.o \
1186 spawn.o \
1187 stack.o \
1188 swapctxt.o \
1189 synch.o \
1190 tdb_agent.o \
1191 thr.o \
1192 thread_interface.o \
1193 thread_pool.o \
1194 tls.o \
1195 tsd.o \
1196 unwind.o

1198 $(TIL:%=pics/%) := CFLAGS64 += $(LIBCBASE)/threads/sparcv9.il

1200 # Files in fp, port/fp subdirectories that need base.il inline template
1201 IL= \
1202 __flt_decim.o \
1203 decimal_bin.o

1205 $(IL:%=pics/%) := CFLAGS64 += $(LIBCBASE)/fp/base.il

1207 # Files in fp subdirectory which need __quad.il inline template
1208 QIL= \
1209 __Q_add.o \
1210 __Q_cmp.o \
1211 __Q_cmpe.o \
1212 __Q_div.o \
1213 __Q_dtoq.o \
1214 __Q_fcc.o \
1215 __Q_mul.o \
1216 __Q_qtod.o \
1217 __Q_qtoi.o \
1218 __Q_qtos.o \
1219 __Q_qtou.o \
1220 __Q_sqrt.o \
1221 __Q_stoq.o \
1222 __Q_sub.o \
1223 __Qp_qtox.o \
1224 __Qp_qtoux.o

1226 $(QIL:%=pics/%) := CFLAGS64 += $(LIBCDIR)/$(MACH)/fp/__quad.il
1227 pics/_Op%.o := CFLAGS64 += -I$(LIBCDIR)/$(MACH)/fp
1228 pics/_Q%.o := sparcv9_COPTFLAG = -x04 -xchip=ultra

1230 # Files in crt subdirectory which need muldiv64.il inline template
1231 #CIL= mul64.o divrem64.o
1232 #$(CIL:%=pics/%) := CFLAGS += $(LIBCBASE)/crt/mul64.il

1234 # large-file-aware components that should be built large

1236 #$(COMSYSOBS64:%=pics/%) := \
1237 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1239 #$(SYSOBS64:%=pics/%) := \
1240 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1242 #$(PORTGEN64:%=pics/%) := \
1243 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1245 #$(PORTSTDIO64:%=pics/%) := \
1246 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

1248 #$(PORTSYS64:%=pics/%) := \
1249 # CPPFLAGS += -D_LARGEFILE_SOURCE -D_FILE_OFFSET_BITS=64

```

```

1251 $(PORTSTDIO_W:%=pics/%) := \
1252 CPPFLAGS += -D_WIDE

1254 $(PORTPRINT_W:%=pics/%) := \
1255 CPPFLAGS += -D_WIDE

1257 $(PORTI18N_COND:%=pics/%) := \
1258 CPPFLAGS += -D_WCS_LOGLONG

1260 pics/arc4random.o := CPPFLAGS += -I$(SRC)/common/crypto/chacha

1262 # Files which need extra optimization
1263 pics/getenv.o := sparcv9_COPTFLAG = -x04

1265 .KEEP_STATE:

1267 all: $(LIBS) $(LIB_PIC)

1269 lint := CPPFLAGS += -I$(LIBCDIR)/$(MACH)/fp
1270 lint := CPPFLAGS += -D_MSE_INT_H -D_LCONV_C99
1271 lint := LINTFLAGS64 += -mn

1273 lint:
1274 @echo $(LINT.c) ... $(LDLIBS)
1275 @$(LINT.c) $(SRCS) $(LDLIBS)

1277 $(LINTLIB) := SRCS=$(LIBCDIR)/port/l1ib-1c
1278 $(LINTLIB) := CPPFLAGS += -D_MSE_INT_H
1279 $(LINTLIB) := LINTFLAGS64=-nvx -m64

1281 # object files that depend on inline template
1282 $(TIL:%=pics/%) := $(LIBCBASE)/threads/sparcv9.il
1283 $(IL:%=pics/%) := $(LIBCBASE)/fp/base.il
1284 $(QIL:%=pics/%) := $(LIBCDIR)/$(MACH)/fp/__quad.il
1285 #$(CIL:%=pics/%) := $(LIBCBASE)/crt/muldiv64.il

1287 # include common libc targets
1288 include $(LIBCDIR)/Makefile.targ

1290 # We need to strip out all CTF and DOF data from the static library
1291 $(LIB_PIC) := DIR = pics
1292 $(LIB_PIC): pics $$ (PICS)
1293 $(BUILD.AR)
1294 $(MCS) -d -n .SUNW_ctf $$@ > /dev/null 2>&1
1295 $(MCS) -d -n .SUNW_dof $$@ > /dev/null 2>&1
1296 $(AR) -ts $$@ > /dev/null
1297 $(POST_PROCESS_A)

1299 # special cases
1300 #$(STRETS:%=pics/%) := crt/stret.s
1301 # $(AS) $(ASFLAGS) -DSTRET$(F:stret%.o=) crt/stret.s -o $$@
1302 # $(POST_PROCESS_O)

1304 #crt/_rtbootld.s: crt/_rtboot.s crt/_rtld.c
1305 # $(CC) $(CPPFLAGS) -O -S -K pic crt/_rtld.c -o crt/_rtld.s
1306 # $(CAT) crt/_rtboot.s crt/_rtld.s > $$@
1307 # $(RM) crt/_rtld.s

1309 ASSYMDEP_OBJS= \
1310 __lwp_mutex_unlock.o \
1311 __stack_grow.o \
1312 asm_subr.o \
1313 setjmp.o \
1314 smt_pause.o \
1315 tls_get_addr.o

```

```
1316         unwind_frame.o      \  
1317         vforkx.o  
  
1319 $(ASSYMDEP_OBJS:%=pics/%) := CPPFLAGS += -I.  
  
1321 $(ASSYMDEP_OBJS:%=pics/%): assym.h  
  
1323 # assym.h build rules  
  
1325 assym.h := CFLAGS64 += $(CCGDEBUG)  
1325 assym.h := CFLAGS64 += -g  
  
1327 GENASSYM_C = $(LIBCDIR)/$(MACH)/genassym.c  
  
1329 genassym: $(GENASSYM_C)  
1330     $(NATIVECC) $(NATIVE_CFLAGS) -I$(LIBCBASE)/inc -I$(LIBCDIR)/inc \  
1331     $(CPPFLAGS.native) -o $@ $(GENASSYM_C)  
  
1333 OFFSETS = $(LIBCDIR)/$(MACH)/offsets.in  
  
1335 assym.h: $(OFFSETS) genassym  
1336     $(OFFSETS_CREATE) <$(OFFSETS) >$@  
1337     ./genassym >>$@  
  
1339 # derived C source and related explicit dependencies  
1340 $(LIBCDIR)/port/gen/new_list.c: $(LIBCDIR)/port/gen/errlist $(LIBCDIR)/port/gen/  
1341     cd $(LIBCDIR)/port/gen; pwd; $(AWK) -f errlist.awk < errlist  
  
1343 pics/new_list.o: $(LIBCDIR)/port/gen/new_list.c
```

new/usr/src/lib/libexact/Makefile.com

1

```
*****
1616 Fri Dec 7 21:21:55 2018
new/usr/src/lib/libexact/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
```

```
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
```

```
26 LIBRARY =      libexact.a
27 VERS =         .1
28 COMMON_OBJS =  exacct_core.o
29 LIB_OBJS =     exacct_ops.o
30 OBJECTS =     $(COMMON_OBJS) $(LIB_OBJS)
```

```
32 include ../../Makefile.lib
```

```
34 SRCS=          $(COMMON_OBJS:%.o=../../common/exacct/%.c) \
35                $(LIB_OBJS:%.o=../../common/%.c)
```

```
37 LIBS =        $(DYNLIB) $(LINTLIB)
38 LDLIBS +=     -lc
```

```
40 SRCDIR =      ../common
41 $(LINTLIB) := SRCS = $(SRCDIR)/$(LINTSRC)
```

```
43 CFLAGS +=     $(CCVERBOSE)
44 CERRWARN +=   -_gcc=-Wno-uninitialized
45 CPPFLAGS +=   -D_FILE_OFFSET_BITS=64
```

```
47 debug :=     CPPFLAGS += -DLIBEXACT_DEBUG
48 debug :=     COPTFLAG = $(CCGDEBUG)
49 debug :=     COPTFLAG64 = $(CCGDEBUG)
48 debug :=     COPTFLAG = -g
49 debug :=     COPTFLAG64 = -g
```

```
51 .KEEP_STATE:
```

```
53 all debug: $(LIBS)
```

```
55 lint: lintcheck
```

```
57 include ../../Makefile.targ
```

```
59 objs/%.o pics/%.o: ../../common/exacct/%.c
```

new/usr/src/lib/libexact/Makefile.com

2

```
60          $(COMPILE.c) -o $@ $<
61          $(POST_PROCESS_O)
```

new/usr/src/lib/libsmfbs/Makefile.com

1

```
*****
2589 Fri Dec 7 21:21:56 2018
new/usr/src/lib/libsmfbs/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright 2009 Sun Microsystems, Inc. All rights reserved.
24 # Use is subject to license terms.
25 # Copyright 2015 Igor Kozhukhov <ikozhukhov@gmail.com>
26 #
27 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
28 #
29 #
30 #
31 # lib/libsmfbs/Makefile.com
32 #
33 #
34 LIBRARY=      libsmfbs.a
35 VERS=        .1
36 #
37 # leaving out: kiconv.o
38 #
39 OBJ_LIB=\
40     acl_api.o \
41     acl_print.o \
42     charsets.o \
43     cfopt.o \
44     connect.o \
45     crypt.o \
46     ctx.o \
47     derparse.o \
48     file.o \
49     findvc.o \
50     getaddr.o \
51     iod_cl.o \
52     iod_wk.o \
53     keychain.o \
54     krb5ssp.o \
55     mbuf.o \
56     nb.o \
57     nb_name.o \
58     nb_net.o \
59     nb_ssn.o \
60     nbns_rq.o \
61     negprot.o \
```

new/usr/src/lib/libsmfbs/Makefile.com

2

```
62     newvc.o \
63     nls.o \
64     ntlm.o \
65     ntlmssp.o \
66     print.o \
67     rap.o \
68     rcfile.o \
69     rq.o \
70     signing.o \
71     spnego.o \
72     spnegoparse.o \
73     ssnsetup.o \
74     ssp.o \
75     subr.o \
76     ui-sun.o \
77     utf_str.o
78 #
79 OBJ_CMN= smfbs_ntacl.o
80 #
81 OBJECTS= $(OBJ_LIB) $(OBJ_CMN)
82 #
83 include $(SRC)/lib/Makefile.lib
84 #
85 LIBS =      $(DYNLIB) $(LINTLIB)
86 #
87 SRCDIR=     ../smf
88 CMNDIR=     $(SRC)/common/smfclnt
89 #
90 SRCS=       $(OBJ_LIB:%.o=$(SRCDIR)/%.c) \
91             $(OBJ_CMN:%.o=$(CMNDIR)/%.c)
92 #
93 $(LINTLIB) := SRCS = $(SRCDIR)/$(LINTSRC)
94 #
95 CSTD=       $(CSTD_GNU99)
96 #
97 LDLIBS += -lsocket -lnsl -lc -lmd -lpkcs11 -lkrb5 -lsec -lidmap
98 #
99 # normal warnings...
100 CFLAGS += $(CVERBOSE)
101 #
102 CERRWARN += -_gcc=-Wno-uninitialized
103 CERRWARN += -_gcc=-Wno-unused-variable
104 #
105 CPPFLAGS += -D_EXTENSIONS__ -D_REENTRANT -DMIA \
106             -I$(SRCDIR) -I.. -I../netsmb \
107             -I$(SRC)/uts/common \
108             -I$(SRC)/common/smfclnt
109 #
110 # Debugging
111 ${NOT_RELEASE_BUILD} CPPFLAGS += -DDEBUG
112 #
113 # uncomment these for dbx debugging
114 #COPTFLAG = -g
115 #CTF_FLAGS =
116 #CTFCONVERT_O=
117 #CTFMERGE_LIB=
118 #
119 # Filter out the less important lint.
120 # See lgrep.awk
121 LGREP = $(AWK) -f $(SRCDIR)/lgrep.awk
122 LTAIL += 2>&1 | $(LGREP)
123 #
124 all: $(LIBS)
125 #
126 lint: lintcheck_t
```

new/usr/src/lib/libmbfs/Makefile.com

3

```
122 include ../../Makefile.targ

124 lintcheck_t: $$SRCS
125     $(LINT.c) $(LINTCHECKFLAGS) $(SRCS) $(LDLIBS) $(LTAIL)

127 objs/%.o pics/%.o: $(CMNDIR)/%.c
128     $(COMPILE.c) -o $@ $<
129     $(POST_PROCESS_O)

131 .KEEP_STATE:
```

new/usr/src/lib/libtnfctl/Makefile.com

1

```
*****
2369 Fri Dec 7 21:21:56 2018
new/usr/src/lib/libtnfctl/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #

26 LIBRARY=      libtnfctl.a
27 VERS=         .1
28 OBJECTS.c=    open.o      \
29               prb_child.o \
30               prb_shmem.o  \
31               prb_proc.o   \
32               prb_lmap.o   \
33               prb_rtl.d.o  \
34               prb_findexec.o \
35               prb_status.o \
36               util.o       \
37               traverse.o   \
38               sym.o        \
39               elf.o        \
40               continue.o   \
41               checklib.o   \
42               probes.o     \
43               probes_ext.o \
44               close.o      \
45               targmem.o    \
46               comb.o       \
47               kernel_int.o \
48               kernel.o     \
49               internal.o   \
50               status.o

53 OBJECTS.s=    $(MACH)_asm.o

55 OBJECTS=      $(OBJECTS.c) $(OBJECTS.s)

57 include ../../Makefile.lib

59 # We omit $(OBJECTS.s:%.o=%.s) in the next line, because lint no like
60 SRCS= $(OBJECTS.c:%.o=../../%.c)
```

new/usr/src/lib/libtnfctl/Makefile.com

2

```
62 LIBS=        $(DYNLIB)

64 HDRS=        tnfctl.h
65 ROOTHDRDIR=  $(ROOT)/usr/include/tnf
66 ROOTHDRS=    $(HDRS:%=$(ROOTHDRDIR)/%)
67 CHECKHDRS=   $(HDRS:%.h=%.check)
68 $(ROOTHDRS) := FILEMODE = 0644
69 CHECKHDRS =  $(HDRS:%.h=%.check)

71 LDLIBS +=    -lc -lelf

73 # Uncomment the following line for a debug build
74 # COPTFLAG =  -g -DDEBUG $(CCVERBOSE)
73 CPPFLAGS +=  -I$(SRC)/lib/libtnfprobe -D_REENTRANT -I$(SRC)/cmd/sgs/include

75 LINTFLAGS += -y

77 CERRWARN +=  -_gcc=-Wno-uninitialized
78 CERRWARN +=  -_gcc=-Wno-empty-body
79 CERRWARN +=  -_gcc=-Wno-parentheses

81 ASFLAGS +=   -P

83 $(ROOTHDRS) := FILEMODE = 644

85 .KEEP_STATE:

87 all: $(LIBS)

89 install_h: $(ROOTHDRDIR) $(ROOTHDRS)

91 lint:
92     $(LINT.c) $(SRCS)

94 check: $(CHECKHDRS)

96 $(ROOTLIBDIR) $(ROOTHDRDIR):
97     $(INS.dir)

99 $(ROOTHDRDIR)/% : %
100     $(INS.file)

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

104 objs/%.o pics/%.o: ../../%.s
105     $(COMPILE.s) -o $@ $<
106     $(POST_PROCESS_O)

108 objs/%.o pics/%.o: ../../%.c
109     $(COMPILE.c) -o $@ $<
110     $(POST_PROCESS_O)

113 %.o:         ../../%.c
114     $(COMPILE.c) -o $@ $<
115     $(POST_PROCESS_O)

117 include ../../Makefile.targ
```

new/usr/src/lib/libzpool/Makefile.com

1

```
*****
3184 Fri Dec 7 21:21:57 2018
new/usr/src/lib/libzpool/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
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) 2005, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright (c) 2013, 2016 by Delphix. All rights reserved.
24 # Copyright 2017 Joyent, Inc.
25 #
26
27 LIBRARY= libzpool.a
28 VERS= .1
29
30 # include the list of ZFS sources
31 include ../../uts/common/Makefile.files
32 KERNEL_OBJS = kernel.o util.o
33 DTRACE_OBJS = zfs.o
34
35 OBJECTS=$(LUA_OBJS) $(ZFS_COMMON_OBJS) $(ZFS_SHARED_OBJS) $(KERNEL_OBJS)
36
37 # include library definitions
38 include ../../Makefile.lib
39
40 LUA_SRCS= $(LUA_OBJS:%o=../../uts/common/fs/zfs/lua/%.c)
41 ZFS_COMMON_SRCS= $(ZFS_COMMON_OBJS:%o=../../uts/common/fs/zfs/%.c)
42 ZFS_SHARED_SRCS= $(ZFS_SHARED_OBJS:%o=../../uts/common/fs/zfs/%.c)
43 KERNEL_SRCS= $(KERNEL_OBJS:%o=../../common/%.c)
44
45 SRCS=$(LUA_SRCS) $(ZFS_COMMON_SRCS) $(ZFS_SHARED_SRCS) $(KERNEL_SRCS)
46 SRCDIR= ../common
47
48 # There should be a mapfile here
49 MAPFILES =
50
51 LIBS += $(DYNLIB) $(LINTLIB)
52
53 INCS += -I../common
54 INCS += -I../../uts/common/fs/zfs
55 INCS += -I../../uts/common/fs/zfs/lua
56 INCS += -I../../common/zfs
57 INCS += -I../../common
58
59 CLEANFILES += ../common/zfs.h
60 CLEANFILES += $(EXTPICS)
```

new/usr/src/lib/libzpool/Makefile.com

2

```
62 $(LINTLIB) := SRCS= $(SRCDIR)/$(LINTSRC)
63 $(LINTLIB): ../common/zfs.h
64 $(LIBS): ../common/zfs.h
65
66 CSTD= $(CSTD_GNU99)
67 C99LMODE= -Xc99=%all
68
69 CFLAGS += $(CCGDEBUG) $(CCVERBOSE) $(CNOGLOBAL)
70 CFLAGS64 += $(CCGDEBUG) $(CCVERBOSE) $(CNOGLOBAL)
69 CFLAGS += -g $(CCVERBOSE) $(CNOGLOBAL)
70 CFLAGS64 += -g $(CCVERBOSE) $(CNOGLOBAL)
71 LDLIBS += -lcmdutils -lumem -lavl -lnvpair -lz -lc -lsysevent -lmd \
72 -lfakekernel
73 CPPFLAGS.first = -I$(SRC)/lib/libfakekernel/common
74 CPPFLAGS += $(INCS) -DDEBUG -D_FAKE_KERNEL
75
76 LINTFLAGS += -erroff=E_STATIC_UNUSED $(INCS)
77 LINTFLAGS64 += -erroff=E_STATIC_UNUSED $(INCS)
78
79 CERRWARN += -_gcc=-Wno-parentheses
80 CERRWARN += -_gcc=-Wno-switch
81 CERRWARN += -_gcc=-Wno-type-limits
82 CERRWARN += -_gcc=-Wno-unused-variable
83 CERRWARN += -_gcc=-Wno-empty-body
84 CERRWARN += -_gcc=-Wno-unused-function
85 CERRWARN += -_gcc=-Wno-unused-label
86
87 .KEEP_STATE:
88
89 all: $(LIBS)
90
91 lint: $(LINTLIB)
92
93 include ../../Makefile.targ
94
95 EXTPICS= $(DTRACE_OBJS:%=pics%)
96
97 pics/%.o: ../../uts/common/fs/zfs/%.c ../common/zfs.h
98 $(COMPILE.c) -o $@ $<
99 $(POST_PROCESS_O)
100
101 pics/%.o: ../../uts/common/fs/zfs/lua/%.c
102 $(COMPILE.c) -o $@ $<
103 $(POST_PROCESS_O)
104
105 pics/%.o: ../../common/zfs/%.c ../common/zfs.h
106 $(COMPILE.c) -o $@ $<
107 $(POST_PROCESS_O)
108
109 pics/%.o: ../common/%.d $(PICS)
110 $(COMPILE.d) -C -s $< -o $@ $(PICS)
111 $(POST_PROCESS_O)
112
113 ../common/%.h: ../common/%.d
114 $(DTRACE) -xnolib -h -s $< -o $@
```



```

*****
6033 Fri Dec 7 21:21:57 2018
new/usr/src/lib/smbsrv/libfksmbsrv/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
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 2008 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
25 # Copyright 2014 Nexenta Systems, Inc. All rights reserved.
26 #

28 LIBRARY = libfksmbsrv.a
29 VERS = .1

31 OBJS_LOCAL = \
32 fksmb_cred.o \
33 fksmb_dt.o \
34 fksmb_fem.o \
35 fksmb_idmap.o \
36 fksmb_init.o \
37 fksmb_kdoor.o \
38 fksmb_sign_pkcs.o \
39 fake_lookup.o \
40 fake_nblk.o \
41 fake_vfs.o \
42 fake_vnode.o \
43 fake_vop.o \
44 fake_xattr.o \
45 reparse.o \
46 vncache.o

48 # See also: $SRC/uts/common/Makefile.files
49 # NB: Intentionally omitted, compared w/ the above:
50 # smb_cred, smb_fem, smb_idmap, smb_init, smb_kdoor
51 #
52 OBJS_FS_SMBSRV = \
53 smb_acl.o \
54 smb_alloc.o \
55 smb_authenticate.o \
56 smb_close.o \
57 smb_cmnr_rename.o \
58 smb_cmnr_setfile.o \
59 smb_common_open.o \
60 smb_common_transact.o \
61 smb_create.o

```

```

62 smb_delete.o \
63 smb_dfs.o \
64 smb_directory.o \
65 smb_dispatch.o \
66 smb_echo.o \
67 smb_errno.o \
68 smb_find.o \
69 smb_flush.o \
70 smb_fsinfo.o \
71 smb_fsops.o \
72 smb_kshare.o \
73 smb_kutil.o \
74 smb_lock.o \
75 smb_lock_byte_range.o \
76 smb_locking_andx.o \
77 smb_logoff_andx.o \
78 smb_mangle_name.o \
79 smb_mbuf_marshall.o \
80 smb_mbuf_util.o \
81 smb_negotiate.o \
82 smb_net.o \
83 smb_node.o \
84 smb_notify.o \
85 smb_nt_cancel.o \
86 smb_nt_create_andx.o \
87 smb_nt_transact_create.o \
88 smb_nt_transact_ioctl.o \
89 smb_nt_transact_notify_change.o \
90 smb_nt_transact_quota.o \
91 smb_nt_transact_security.o \
92 smb_odir.o \
93 smb_ofile.o \
94 smb_open_andx.o \
95 smb_opipe.o \
96 smb_oplock.o \
97 smb_pathname.o \
98 smb_print.o \
99 smb_process_exit.o \
100 smb_query_fileinfo.o \
101 smb_quota.o \
102 smb_read.o \
103 smb_rename.o \
104 smb_sd.o \
105 smb_seek.o \
106 smb_server.o \
107 smb_session.o \
108 smb_session_setup_andx.o \
109 smb_set_fileinfo.o \
110 smb_signing.o \
111 smb_thread.o \
112 smb_tree.o \
113 smb_trans2_create_directory.o \
114 smb_trans2_dfs.o \
115 smb_trans2_find.o \
116 smb_tree_connect.o \
117 smb_unlock_byte_range.o \
118 smb_user.o \
119 smb_vfs.o \
120 smb_vops.o \
121 smb_vss.o \
122 smb_write.o \
123 \
124 smb2_dispatch.o \
125 smb2_cancel.o \
126 smb2_change_notify.o \
127 smb2_close.o \

```

```

128         smb2_create.o \
129         smb2_echo.o \
130         smb2_flush.o \
131         smb2_ioctl.o \
132         smb2_lock.o \
133         smb2_logoff.o \
134         smb2_negotiate.o \
135         smb2_ofile.o \
136         smb2_oplock.o \
137         smb2_qinfo_file.o \
138         smb2_qinfo_fs.o \
139         smb2_qinfo_sec.o \
140         smb2_qinfo_quota.o \
141         smb2_query_dir.o \
142         smb2_query_info.o \
143         smb2_read.o \
144         smb2_session_setup.o \
145         smb2_set_info.o \
146         smb2_setinfo_file.o \
147         smb2_setinfo_fs.o \
148         smb2_setinfo_quota.o \
149         smb2_setinfo_sec.o \
150         smb2_signing.o \
151         smb2_tree_connect.o \
152         smb2_tree_disconn.o \
153         smb2_write.o

155 # Can't just link with -lsmb because of user vs kernel API
156 # i.e. can't call free with mem from kmem_alloc, which is
157 # what happens if we just link with -lsmb
158 OBJS_CMN_SMBRSRV = \
159         smb_inet.o \
160         smb_match.o \
161         smb_msgbuf.o \
162         smb_native.o \
163         smb_netbios_util.o \
164         smb_oem.o \
165         smb_sid.o \
166         smb_string.o \
167         smb_token.o \
168         smb_token_xdr.o \
169         smb_utf8.o \
170         smb_xdr.o

172 OBJS_MISC = \
173         acl_common.o \
174         pathname.o \
175         refstr.o \
176         smb_status2winerr.o \
177         xattr_common.o

179 OBJECTS = \
180         $(OBJ_LOCAL) \
181         $(OBJ_FS_SMBRSRV) \
182         $(OBJ_CMN_SMBRSRV) \
183         $(OBJ_MISC)

185 include ../../Makefile.lib
186 include ../../Makefile.lib

188 # Force SOURCEDEBUG
189 CSOURCEDEBUGFLAGS = $(CCGDEBUG)
190 CCSOURCEDEBUGFLAGS = $(CCGDEBUG)
189 CSOURCEDEBUGFLAGS = -g
190 CCSOURCEDEBUGFLAGS = -g
191 STRIP_STABS = :

```

```

194 # Note: need our sys includes _before_ ENVCPFLAGS, proto etc.
195 CPPFLAGS.first += -I../../libfakekernel/common
196 CPPFLAGS.first += -I./common

198 INCS += -I$(SRC)/uts/common
199 INCS += -I$(SRC)/common/smbsrv
200 INCS += -I$(SRC)/common

202 LINTCHECKFLAGS += -erroff=E_INCONS_ARG_DECL2
203 LINTCHECKFLAGS += -erroff=E_INCONS_VAL_TYPE_DECL2
204 LINTCHECKFLAGS += -erroff=E_INCONS_VAL_TYPE_USED2

206 LDLIBS += $(MACH_LDLIBS)
207 LDLIBS += -lfakekernel -lidmap -lcmdutils
208 LDLIBS += -lavl -lnvpair -lnsl -lpkcs11 -lreparse -lc

210 CPPFLAGS += $(INCS) -D_REENTRANT -D_FAKE_KERNEL
211 CPPFLAGS += -D_FILE_OFFSET_BITS=64
212 # Always want DEBUG here
213 CPPFLAGS += -DDEBUG

215 CERRWARN += -_gcc=-Wno-parentheses
216 CERRWARN += -_gcc=-Wno-switch

218 SRCS= $(OBJ_LOCAL:%.o=$(SRCDIR)/%.c) \
219        $(OBJ_FS_SMBRSRV:%.o=$(SRC)/uts/common/fs/smbsrv/%.c) \
220        $(OBJ_CMN_SMBRSRV:%.o=$(SRC)/common/smbsrv/%.c)

222 all:

224 pics/%.o: $(SRC)/uts/common/fs/smbsrv/%.c
225           $(COMPILE.c) -o $@ $<
226           $(POST_PROCESS_O)

228 pics/acl_common.o: $(SRC)/common/acl/acl_common.c
229           $(COMPILE.c) -o $@ $(SRC)/common/acl/acl_common.c
230           $(POST_PROCESS_O)

232 pics/smb_status2winerr.o: $(SRC)/common/smbclnt/smb_status2winerr.c
233           $(COMPILE.c) -o $@ $(SRC)/common/smbclnt/smb_status2winerr.c
234           $(POST_PROCESS_O)

236 pics/pathname.o: $(SRC)/uts/common/fs/pathname.c
237           $(COMPILE.c) -o $@ $(SRC)/uts/common/fs/pathname.c
238           $(POST_PROCESS_O)

240 pics/refstr.o: $(SRC)/uts/common/os/refstr.c
241           $(COMPILE.c) -o $@ $(SRC)/uts/common/os/refstr.c
242           $(POST_PROCESS_O)

244 pics/xattr_common.o: $(SRC)/common/xattr/xattr_common.c
245           $(COMPILE.c) -o $@ $(SRC)/common/xattr/xattr_common.c
246           $(POST_PROCESS_O)

248 # Makefile.targ has rule for $(SRC)/common/smbsrv/%.c

250 .KEEP_STATE:

252 include ../../Makefile.targ
253 include ../../Makefile.targ

```

new/usr/src/lib/udapl/libdat/Makefile.com

1

1484 Fri Dec 7 21:21:58 2018

new/usr/src/lib/udapl/libdat/Makefile.com

10051 cw(1) shouldn't have to translate -g

```
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 2006 Sun Microsystems, Inc. All rights reserved.
23 # Use is subject to license terms.
24 #
```

```
26 LIBRARY=      libdat.a
27 VERS=         .1
```

```
29 OBJECTS =     \
30     dat_dictionary.o \
31     dat_init.o \
32     dat_dr.o \
33     dat_osd.o \
34     dat_sr.o \
35     dat_strerror.o \
36     dat_api.o \
37     udat.o \
38     udat_api.o \
39     udat_sr_parser.o
```

```
41 include ../../../../Makefile.lib
```

```
43 LIBS = $(DYNLIB) $(LINTLIB)
44 LDLIBS += -lc
```

```
46 SRCDIR =      ../common
```

```
48 CPPFLAGS +=   -I../include
49 CFLAGS +=     $(CCVERBOSE)
50 LINTFLAGS +=  -DDEBUG
51 LINTFLAGS64 += -DDEBUG
52 $(LINTLIB):=  SRCS = $(SRCDIR)/$(LINTSRC)
```

```
54 CERRWARN +=   -_gcc=-Wno-type-limits
```

```
56 $(NOT_RELEASE_BUILD)CPPFLAGS += -DDEBUG
```

```
57 debug := COPTFLAG = -g
```

```
58 debug := COPTFLAG64 = -g
```

```
58 .KEEP_STATE:
```

new/usr/src/lib/udapl/libdat/Makefile.com

2

```
60 all: $(LIBS)
```

```
62 debug: all
```

```
64 lint: lintcheck
```

```
66 include ../../../../Makefile.targ
```

```

*****
3906 Fri Dec 7 21:21:58 2018
new/usr/src/lib/udapl/udapl_tavor/Makefile.com
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 #
23 # Copyright (c) 2009, 2010, Oracle and/or its affiliates. All rights reserved.
24 #
25 #
26 LIBRARY=      udapl_tavor.a
27 VERS=         .1
28 #
29 LOCOBJS = \
30     dapl_cno_create.o \
31     dapl_cno_free.o \
32     dapl_cno_modify_agent.o \
33     dapl_cno_query.o \
34     dapl_cno_util.o \
35     dapl_cno_wait.o \
36     dapl_cookie.o \
37     dapl_cr_accept.o \
38     dapl_cr_callback.o \
39     dapl_cr_handoff.o \
40     dapl_cr_query.o \
41     dapl_cr_reject.o \
42     dapl_cr_util.o \
43     dapl_debug.o \
44     dapl_ep_connect.o \
45     dapl_ep_create.o \
46     dapl_ep_create_with_srq.o \
47     dapl_ep_disconnect.o \
48     dapl_ep_dup_connect.o \
49     dapl_ep_free.o \
50     dapl_ep_get_status.o \
51     dapl_ep_modify.o \
52     dapl_ep_post_rdma_read.o \
53     dapl_ep_post_rdma_write.o \
54     dapl_ep_post_recv.o \
55     dapl_ep_post_send.o \
56     dapl_ep_query.o \
57     dapl_ep_reset.o \
58     dapl_ep_util.o \
59     dapl_evd_clear_unwaitable.o \
60     dapl_evd_connection_callb.o \
61     dapl_evd_cq_async_error_callb.o \

```

```

62     dapl_evd_create.o \
63     dapl_evd_dequeue.o \
64     dapl_evd_disable.o \
65     dapl_evd_dto_callb.o \
66     dapl_evd_enable.o \
67     dapl_evd_free.o \
68     dapl_evd_modify_cno.o \
69     dapl_evd_post_se.o \
70     dapl_evd_qp_async_error_callb.o \
71     dapl_evd_query.o \
72     dapl_evd_resize.o \
73     dapl_evd_set_unwaitable.o \
74     dapl_evd_un_async_error_callb.o \
75     dapl_evd_util.o \
76     dapl_evd_wait.o \
77     dapl_get_consumer_context.o \
78     dapl_get_handle_type.o \
79     dapl_hash.o \
80     dapl_hca_util.o \
81     dapl_ia_close.o \
82     dapl_ia_open.o \
83     dapl_ia_query.o \
84     dapl_ia_util.o \
85     dapl_init.o \
86     dapl_llist.o \
87     dapl_lmr_create.o \
88     dapl_lmr_free.o \
89     dapl_lmr_query.o \
90     dapl_lmr_sync_rdma.o \
91     dapl_lmr_util.o \
92     dapl_mr_util.o \
93     dapl_name_service.o \
94     dapl_osd.o \
95     dapl_provider.o \
96     dapl_psp_create.o \
97     dapl_psp_create_any.o \
98     dapl_psp_free.o \
99     dapl_psp_query.o \
100    dapl_pz_create.o \
101    dapl_pz_free.o \
102    dapl_pz_query.o \
103    dapl_pz_util.o \
104    dapl_ring_buffer_util.o \
105    dapl_rmr_bind.o \
106    dapl_rmr_create.o \
107    dapl_rmr_free.o \
108    dapl_rmr_query.o \
109    dapl_rmr_util.o \
110    dapl_rsp_create.o \
111    dapl_rsp_free.o \
112    dapl_rsp_query.o \
113    dapl_set_consumer_context.o \
114    dapl_sp_util.o \
115    dapl_srq.o \
116    dapl_srq_util.o
117 #
118 TAVOROBJS = \
119     dapl_tavor_hca.o \
120     dapl_tavor_hw.o \
121     dapl_arbel_hw.o \
122     dapl_hermon_hw.o \
123     dapl_tavor_ibtft_cm.o \
124     dapl_tavor_ibtft_dto.o \
125     dapl_tavor_ibtft_mrsync.o \
126     dapl_tavor_ibtft_qp.o \
127     dapl_tavor_ibtft_util.o \

```

```
128     dapl_tavor_wr.o
130 OBJECTS = $(LOCOBJS) $(TAVOROBJS)
132 include $(SRC)/lib/Makefile.lib
134 LIBS =      $(DYNLIB)
135 LDLIBS +=   -ldevinfo -lsocket -lnsl -ldat -lc -ldladm
137 SRCDIR =    ../common
138 TAVORSRCDIR = ../tavor
140 SRCS = $(LOCOBJS:%.o=$(SRCDIR)/%.c) $(TAVOROBJS:%.o=$(TAVORSRCDIR)/%.c)
142 CPPFLAGS += -I$(SRC)/lib/udapl/udapl_tavor/include
143 CPPFLAGS += -I$(SRC)/lib/udapl/udapl_tavor/tavor
144 CPPFLAGS += -I$(SRC)/uts/common/sys/ib/clients/daplt
145 CPPFLAGS += -I$(SRC)/uts/common
146 CPPFLAGS += -I$(SRC)/uts/common/sys/ib/clients
147 CFLAGS +=   $(CCVERBOSE)
148 LINTFLAGS += -DDAPL_DBG
149 LINTFLAGS64 += -DDAPL_DBG
151 CERRWARN +=  _gcc=-Wno-parentheses
152 CERRWARN +=  _gcc=-Wno-uninitialized
153 CERRWARN +=  _gcc=-Wno-switch
155 $(NOT_RELEASE_BUILD)CPPFLAGS += -DDAPL_DBG
156 debug := COPTFLAG = -g
157 debug := COPTFLAG64 = -g
157 .KEEP_STATE:
159 all: $(LIBS)
161 debug: all
163 lint: lintcheck
165 pics/%.o: $(TAVORSRCDIR)/%.c
166     $(COMPILE.c) -o $@ $<
167     $(POST_PROCESS_O)
169 include $(SRC)/lib/Makefile.targ
```

```

*****
41849 Fri Dec 7 21:21:58 2018
new/usr/src/tools/cw/cw.c
10051 cw(1) shouldn't have to translate -g
*****

2 /*
3 * CDDL HEADER START
4 *
5 * The contents of this file are subject to the terms of the
6 * Common Development and Distribution License (the "License").
7 * You may not use this file except in compliance 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 */

23 /*
24 * Copyright 2018, Richard Lowe.
25 */
26 /*
27 * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
28 * Use is subject to license terms.
29 */

31 /*
32 * Wrapper for the GNU C compiler to make it accept the Sun C compiler
33 * arguments where possible.
34 *
35 * Since the translation is inexact, this is something of a work-in-progress.
36 *
37 */

39 /* If you modify this file, you must increment CW_VERSION */
40 #define CW_VERSION "4.0"
41 #define CW_VERSION "3.0"

42 /*
43 * -# Verbose mode
44 * -### Show compiler commands built by driver, no compilation
45 * -A<name[(tokens)]> Preprocessor predicate assertion
46 * -B[<static|dynamic|>] Specify dynamic or static binding
47 * -C Prevent preprocessor from removing comments
48 * -c Compile only - produce .o files, suppress linking
49 * -cg92 Alias for -xtarget=ss1000
50 * -D<name[(token)]> Associate name with token as if by #define
51 * -d[y|n] dynamic [-dy] or static [-dn] option to linker
52 * -E Compile source through preprocessor only, output to stdout
53 * -erroff=<t> Suppress warnings specified by tags t(%none, %all, <tag list>)
54 * -errtags=<a> Display messages with tags a(no, yes)
55 * -errwarn=<t> Treats warnings specified by tags t(%none, %all, <tag list>)
56 * as errors
57 * -fast Optimize using a selection of options
58 * -fd Report old-style function definitions and declarations
59 * -fnonstd Initialize floating-point hardware to non-standard preferences
60 * -fns[=<yes|no>] Select non-standard floating point mode

```

```

61 * -fprecision=<p> Set FP rounding precision mode p(single, double, extended)
62 * -fround=<r> Select the IEEE rounding mode in effect at startup
63 * -fsimple[=<n>] Select floating-point optimization preferences <n>
64 * -fsingle Use single-precision arithmetic (-Xt and -Xs modes only)
65 * -ftrap=<t> Select floating-point trapping mode in effect at startup
66 * -fstore force floating pt. values to target precision on assignment
67 * -G Build a dynamic shared library
68 * -g Compile for debugging
69 * -H Print path name of each file included during compilation
70 * -h <name> Assign <name> to generated dynamic shared library
71 * -I<dir> Add <dir> to preprocessor #include file search path
72 * -i Passed to linker to ignore any LD_LIBRARY_PATH setting
73 * -keeptmp Keep temporary files created during compilation
74 * -L<dir> Pass to linker to add <dir> to the library search path
75 * -l<name> Link with library lib<name>.a or lib<name>.so
76 * -mc Remove duplicate strings from .comment section of output files
77 * -mr Remove all strings from .comment section of output files
78 * -mr,"string" Remove all strings and append "string" to .comment section
79 * -mt Specify options needed when compiling multi-threaded code
80 * -native Find available processor, generate code accordingly
81 * -nofstore Do not force floating pt. values to target precision
82 * on assignment
83 * -norunpath Do not build in a runtime path for shared libraries
84 * -O Use default optimization level (-xO2 or -xO3. Check man page.)
85 * -o <outputfile> Set name of output file to <outputfile>
86 * -P Compile source through preprocessor only, output to .i file
87 * -p Compile for profiling with prof
88 * -Q[y|n] Emit/don't emit identification info to output file
89 * -R<dir[:dir]> Build runtime search path list into executable
90 * -S Compile and only generate assembly code (.s)
91 * -s Strip symbol table from the executable file
92 * -t Turn off duplicate symbol warnings when linking
93 * -U<name> Delete initial definition of preprocessor symbol <name>
94 * -V Report version number of each compilation phase
95 * -v Do stricter semantic checking
96 * -W<c>,<arg> Pass <arg> to specified component <c> (a,l,m,p,0,2,h,i,u)
97 * -w Suppress compiler warning messages
98 * -Xa Compile assuming ANSI C conformance, allow K & R extensions
99 * (default mode)
100 * -Xs Compile assuming (pre-ANSI) K & R C style code
101 * -Xt Compile assuming K & R conformance, allow ANSI C
102 * -xarch=<a> Specify target architecture instruction set
103 * -xbuiltin[=<b>] When profitable inline, or substitute intrinsic functions
104 * for system functions, b={%all,%none}
105 * -xCC Accept C++ style comments
106 * -xchip=<c> Specify the target processor for use by the optimizer
107 * -xcode=<c> Generate different code for forming addresses
108 * -xcrossfile[=<n>] Enable optimization and inlining across source files,
109 * n={0|1}
110 * -xe Perform only syntax/semantic checking, no code generation
111 * -xF Compile for later mapfile reordering or unused section
112 * elimination
113 * -xhelp=<f> Display on-line help information f(flags, readme, errors)
114 * -xildoff Cancel -xildon
115 * -xildon Enable use of the incremental linker, ild
116 * -xinline[=<a>,...,<a>] Attempt inlining of specified user routines,
117 * <a>={%auto,func,no%func}
118 * -xlibmieee Force IEEE 754 return values for math routines in
119 * exceptional cases
120 * -xlibmil Inline selected libm math routines for optimization
121 * -xlic_lib=sunperf Link in the Sun supplied performance libraries
122 * -xlicinfo Show license server information
123 * -xmaxopt=[off,1,2,3,4,5] maximum optimization level allowed on #pragma opt
124 * -xO<n> Generate optimized code (n={1|2|3|4|5})
125 * -xP Print prototypes for function definitions
126 * -xprofile=<p> Collect data for a profile or use a profile to optimize

```

```

127 * <p>={{collect,use}[:<path>],tcov}
128 * -xregs=<r> Control register allocation
129 * -xs Allow debugging without object (.o) files
130 * -xsb Compile for use with the WorkShop source browser
131 * -xsbfast Generate only WorkShop source browser info, no compilation
132 * -xsfpconst Represent unaffixed floating point constants as single
133 * precision
134 * -xspace Do not do optimizations that increase code size
135 * -xstrconst Place string literals into read-only data segment
136 * -xtarget=<t> Specify target system for optimization
137 * -xtemp=<dir> Set directory for temporary files to <dir>
138 * -xtime Report the execution time for each compilation phase
139 * -xunroll=n Enable unrolling loops n times where possible
140 * -Y<c>,<dir> Specify <dir> for location of component <c> (a,l,m,p,0,h,i,u)
141 * -YA,<dir> Change default directory searched for components
142 * -YI,<dir> Change default directory searched for include files
143 * -YP,<dir> Change default directory for finding libraries files
144 * -YS,<dir> Change default directory for startup object files
145 */

147 /*
148 * Translation table:
149 */
150 /*
151 * -# -v
152 * -### error
153 * -A<name[(tokens)]> pass-thru
154 * -B[<static|dynamic|>] pass-thru (syntax error for anything else)
155 * -C pass-thru
156 * -c pass-thru
157 * -cg92 -m32 -mcpu=v8 -mtune=supersparc (SPARC only)
158 * -D<name[=token]> pass-thru
159 * -dy or -dn -Wl,-dy or -Wl,-dn
160 * -E pass-thru
161 * -erroff=E_EMPTY_TRANSLATION_UNIT ignore
162 * -errtags=%all -Wall
163 * -errwarn=%all -Werror else -Wno-error
164 * -fast error
165 * -fd error
166 * -fnonstd error
167 * -fns[=<yes|no>] error
168 * -fprecision=<p> error
169 * -fround=<r> error
170 * -fsimple[=<n>] error
171 * -fsingle[=<n>] error
172 * -ftrap=<t> error
173 * -fstore error
174 * -G pass-thru
175 * -g pass-thru
176 * -H pass-thru
177 * -h <name> pass-thru
178 * -I<dir> pass-thru
179 * -i pass-thru
180 * -keeptmp -save-temps
181 * -L<dir> pass-thru
182 * -l<name> pass-thru
183 * -mc error
184 * -mr error
185 * -mr,"string" error
186 * -mt -D_REENTRANT
187 * -native error
188 * -nofstore error
189 * -nolib -nodefaultlibs
190 * -norunpath ignore
191 * -O -O1 (Check the man page to be certain)
192 * -o <outputfile> pass-thru

```

```

193 * -P -E -o filename.i (or error)
194 * -p pass-thru
195 * -Q[y|n] error
196 * -R<dir[:dir]> pass-thru
197 * -S pass-thru
198 * -s -Wl,-s
199 * -t -Wl,-t
200 * -U<name> pass-thru
201 * -V --version
202 * -v -Wall
203 * -Wa,<arg> pass-thru
204 * -Wp,<arg> pass-thru except -xc99=<a>
205 * -Wl,<arg> pass-thru
206 * -W{m,0,2,h,i,u} error/ignore
207 * -xmodel=kernel -ffreestanding -mmodel=kernel -mno-red-zone
208 * -Wu,-save_args -msave-args
209 * -w pass-thru
210 * -Xa -std=iso9899:199409 or -ansi
211 * -Xt error
212 * -Xs -traditional -std=c89
213 * -xarch=<a> table
214 * -xbuiltin[=<b>] -fbuiltin (-fno-builtin otherwise)
215 * -xCC ignore
216 * -xchip=<c> table
217 * -xcode=<c> table
218 * -xdebugformat=<format> ignore (always use dwarf-2 for gcc)
219 * -xcrossfile[=<n>] ignore
220 * -xe error
221 * -xF error
222 * -xhelp=<f> error
223 * -xildoff ignore
224 * -xinline ignore
225 * -xlibmieee error
226 * -xlibmil error
227 * -xlic_lib=sunperf error
228 * -xmaxopt=[...] error
229 * -xO<n> -O<n>
230 * -xP error
231 * -xprofile=<p> error
232 * -xregs=<r> table
233 * -xs error
234 * -xsb error
235 * -xsbfast error
236 * -xsfpconst error
237 * -xspace ignore (-not -Os)
238 * -xstrconst ignore
239 * -xtarget=<t> table
240 * -xtemp=<dir> error
241 * -xtime error
242 * -xtransition -Wtransition
243 * -xunroll=n error
244 * -W0,-xdbggen=no%usedonly -fno-eliminate-unused-debug-symbols
245 * -Y<c>,<dir> -fno-eliminate-unused-debug-types
246 * -YA,<dir> error
247 * -YI,<dir> error
248 * -YP,<dir> -nostdinc -I<dir>
249 * -YS,<dir> error
250 * -YS,<dir> error
251 */

253 #include <ctype.h>
254 #include <err.h>
255 #include <errno.h>
256 #include <fcntl.h>
257 #include <getopt.h>

```

```

258 #include <stdio.h>
259 #include <stdlib.h>
260 #include <string.h>
261 #include <unistd.h>

263 #include <sys/param.h>
264 #include <sys/stat.h>
265 #include <sys/types.h>
266 #include <sys/utsname.h>
267 #include <sys/wait.h>

269 #define CW_F_CXX      0x01
270 #define CW_F_SHADOW  0x02
271 #define CW_F_EXEC    0x04
272 #define CW_F_ECHO    0x08
273 #define CW_F_XLATE   0x10
274 #define CW_F_PROG    0x20

276 typedef enum cw_op {
277     CW_O_NONE = 0,
278     CW_O_PREPROCESS,
279     CW_O_COMPILE,
280     CW_O_LINK
281 } cw_op_t;
    unchanged portion omitted

584 static void
585 do_gcc(cw_ictx_t *ctx)
586 {
587     int c;
588     int nolibs = 0;
589     int in_output = 0, seen_o = 0, c_files = 0;
590     cw_op_t op = CW_O_LINK;
591     char *model = NULL;
592     char *nameflag;
593     int mflag = 0;

595     if (ctx->i_flags & CW_F_PROG) {
596         newae(ctx->i_ae, "--version");
597         return;
598     }

600     newae(ctx->i_ae, "-fident");
601     newae(ctx->i_ae, "-finline");
602     newae(ctx->i_ae, "-fno-inline-functions");
603     newae(ctx->i_ae, "-fno-builtin");
604     newae(ctx->i_ae, "-fno-asm");
605     newae(ctx->i_ae, "-fdiagnostics-show-option");
606     newae(ctx->i_ae, "-nodefaultlibs");

608 #if defined(__sparc)
609     /*
610     * The SPARC ldd and std instructions require 8-byte alignment of
611     * their address operand. gcc correctly uses them only when the
612     * ABI requires 8-byte alignment; unfortunately we have a number of
613     * pieces of buggy code that doesn't conform to the ABI. This
614     * flag makes gcc work more like Studio with -xmemalign=4.
615     */
616     newae(ctx->i_ae, "-mno-integer-ldd-std");
617 #endif

619     /*
620     * This is needed because 'u' is defined
621     * under a conditional on 'sun'. Should
622     * probably just remove the conditional,
623     * or make it be dependent on '__sun'.

```

```

624     *
625     * -Dunix is also missing in enhanced ANSI mode
626     */
627     newae(ctx->i_ae, "-D__sun");

629     if (asprintf(&nameflag, "-_%s=", ctx->i_compiler->c_name) == -1)
630         nomem();

632     /*
633     * Walk the argument list, translating as we go ..
634     */
635     while (--ctx->i_oldargc > 0) {
636         char *arg = ++ctx->i_oldargv;
637         size_t arglen = strlen(arg);

639         if (*arg == '-') {
640             arglen--;
641         } else {
642             /*
643             * Discard inline files that gcc doesn't grok
644             */
645             if (!in_output && arglen > 3 &&
646                 strcmp(arg + arglen - 3, ".il") == 0)
647                 continue;

649             if (!in_output && arglen > 2 &&
650                 arg[arglen - 2] == '.' &&
651                 (arg[arglen - 1] == 'S' || arg[arglen - 1] == 's' ||
652                  arg[arglen - 1] == 'c' || arg[arglen - 1] == 'i'))
653                 c_files++;

655             /*
656             * Otherwise, filenames and partial arguments
657             * are passed through for gcc to chew on. However,
658             * output is always discarded for the secondary
659             * compiler.
660             */
661             if ((ctx->i_flags & CW_F_SHADOW) && in_output) {
662                 ctx->i_discard = discard_file_name(arg);
663                 newae(ctx->i_ae, ctx->i_discard);
664             } else {
665                 newae(ctx->i_ae, arg);
666             }
667             in_output = 0;
668             continue;
669         }

671         if (ctx->i_flags & CW_F_CXX) {
672             if (strncmp(arg, "-g++=", 6) == 0) {
673                 newae(ctx->i_ae, strchr(arg, '=') + 1);
674                 continue;
675             }
676             if (strncmp(arg, "-compat=", 8) == 0) {
677                 /* discard -compat=4 and -compat=5 */
678                 continue;
679             }
680             if (strcmp(arg, "-Qoption") == 0) {
681                 /* discard -Qoption and its two arguments */
682                 if (ctx->i_oldargc < 3)
683                     error(arg);
684                 ctx->i_oldargc -= 2;
685                 ctx->i_oldargv += 2;
686                 continue;
687             }
688             if (strcmp(arg, "-xwe") == 0) {
689                 /* turn warnings into errors */

```



```

690         newae(ctx->i_ae, "-Werror");
691         continue;
692     }
693     if (strcmp(arg, "-norunpath") == 0) {
694         /* gcc has no corresponding option */
695         continue;
696     }
697     if (strcmp(arg, "-nolib") == 0) {
698         /* -nodefaultlibs is on by default */
699         nolibc = 1;
700         continue;
701     }
702 #if defined(__sparc)
703     if (strcmp(arg, "-cg92") == 0) {
704         mflag |= xlate_xtb(ctx->i_ae, "v8");
705         xlate(ctx->i_ae, "super", xchip_ttbl);
706         continue;
707     }
708 #endif /* __sparc */
709 }

711     switch ((c = arg[1])) {
712     case '_':
713         if ((strncmp(arg, nameflag, strlen(nameflag)) == 0) ||
714             (strncmp(arg, "-_gcc=", 6) == 0) ||
715             (strncmp(arg, "-_gnu=", 6) == 0)) {
716             newae(ctx->i_ae, strchr(arg, '=') + 1);
717         }
718         break;
719     case '#':
720         if (arglen == 1) {
721             newae(ctx->i_ae, "-v");
722             break;
723         }
724         error(arg);
725         break;
726     case 'f':
727         if ((strcmp(arg, "-fpic") == 0) ||
728             (strcmp(arg, "-fPIC") == 0)) {
729             newae(ctx->i_ae, arg);
730             break;
731         }
732         error(arg);
733         break;
734     case 'g':
735         newae(ctx->i_ae, "-gdwarf-2");
736         break;
737     case 'E':
738         if (arglen == 1) {
739             newae(ctx->i_ae, "-xc");
740             newae(ctx->i_ae, arg);
741             op = CW_O_PREPROCESS;
742             nolibc = 1;
743             break;
744         }
745         error(arg);
746         break;
747     case 'c':
748     case 'S':
749         if (arglen == 1) {
750             op = CW_O_COMPILE;
751             nolibc = 1;
752         }
753         /* FALLTHROUGH */
754     case 'C':
755     case 'H':

```

```

753     case 'p':
754         if (arglen == 1) {
755             newae(ctx->i_ae, arg);
756             break;
757         }
758         error(arg);
759         break;
760     case 'A':
761     case 'g':
762 #endif /* !codereview */
763     case 'h':
764     case 'I':
765     case 'i':
766     case 'L':
767     case 'l':
768     case 'R':
769     case 'U':
770     case 'u':
771     case 'w':
772         newae(ctx->i_ae, arg);
773         break;
774     case 'o':
775         seen_o = 1;
776         if (arglen == 1) {
777             in_output = 1;
778             newae(ctx->i_ae, arg);
779         } else if (ctx->i_flags & CW_F_SHADOW) {
780             newae(ctx->i_ae, "-o");
781             ctx->i_discard = discard_file_name(arg);
782             newae(ctx->i_ae, ctx->i_discard);
783         } else {
784             newae(ctx->i_ae, arg);
785         }
786         break;
787     case 'D':
788         newae(ctx->i_ae, arg);
789         /*
790          * XXX Clearly a hack ... do we need _KADB too?
791          */
792         if (strcmp(arg, "-D_KERNEL") == 0 ||
793             strcmp(arg, "-D_BOOT") == 0)
794             newae(ctx->i_ae, "-ffreestanding");
795         break;
796     case 'd':
797         if (arglen == 2) {
798             if (strcmp(arg, "-dy") == 0) {
799                 newae(ctx->i_ae, "-WL,-dy");
800                 break;
801             }
802             if (strcmp(arg, "-dn") == 0) {
803                 newae(ctx->i_ae, "-WL,-dn");
804                 break;
805             }
806         }
807         if (strcmp(arg, "-dalign") == 0) {
808             /*
809              * -dalign forces alignment in some cases;
810              * gcc does not need any flag to do this.
811              */
812             break;
813         }
814         error(arg);
815         break;
816     case 'e':
817         if (strcmp(arg,
818             "-erroff=E_EMPTY_TRANSLATION_UNIT") == 0) {

```

```

819     /*
820      * Accept but ignore this -- gcc doesn't
821      * seem to complain about empty translation
822      * units
823      */
824     break;
825 }
826 /* XX64 -- ignore all -erroff= options, for now */
827 if (strncmp(arg, "-erroff=", 8) == 0)
828     break;
829 if (strcmp(arg, "-errtags=yes") == 0) {
830     warnings(ctx->i_ae);
831     break;
832 }
833 if (strcmp(arg, "-errwarn=all") == 0) {
834     newae(ctx->i_ae, "-Werror");
835     break;
836 }
837 error(arg);
838 break;
839 case 'G':
840     newae(ctx->i_ae, "-shared");
841     nolibc = 1;
842     break;
843 case 'k':
844     if (strcmp(arg, "-keeptmp") == 0) {
845         newae(ctx->i_ae, "-save-temps");
846         break;
847     }
848     error(arg);
849     break;
850 case 'm':
851     if (strcmp(arg, "-mt") == 0) {
852         newae(ctx->i_ae, "-D_REENTRANT");
853         break;
854     }
855     if (strcmp(arg, "-m64") == 0) {
856         newae(ctx->i_ae, "-m64");
857 #if defined(__x86)
858         newae(ctx->i_ae, "-mtune=opteron");
859 #endif
860         mflag |= M64;
861         break;
862     }
863     if (strcmp(arg, "-m32") == 0) {
864         newae(ctx->i_ae, "-m32");
865         mflag |= M32;
866         break;
867     }
868     error(arg);
869     break;
870 case 'B': /* linker options */
871 case 'M':
872 case 'z':
873     {
874         char *opt;
875         size_t len;
876         char *s;
877
878         if (arglen == 1) {
879             opt = **++ctx->i_oldargv;
880             if (opt == NULL || *opt == '\0')
881                 error(arg);
882             ctx->i_oldargc--;
883         } else {
884             opt = arg + 2;

```

```

885     }
886     len = strlen(opt) + 7;
887     if ((s = malloc(len)) == NULL)
888         nomem();
889     (void) snprintf(s, len, "-Wl,-%c%s", c, opt);
890     newae(ctx->i_ae, s);
891     free(s);
892 }
893 break;
894 case 'O':
895     if (arglen == 1) {
896         newae(ctx->i_ae, "-O");
897         break;
898     }
899     error(arg);
900     break;
901 case 'P':
902     /*
903      * We could do '-E -o filename.i', but that's hard,
904      * and we don't need it for the case that's triggering
905      * this addition. We'll require the user to specify
906      * -o in the Makefile. If they don't they'll find out
907      * in a hurry.
908      */
909     newae(ctx->i_ae, "-E");
910     op = CW_O_PREPROCESS;
911     nolibc = 1;
912     break;
913 case 's':
914     if (arglen == 1) {
915         newae(ctx->i_ae, "-Wl,-s");
916         break;
917     }
918     error(arg);
919     break;
920 case 't':
921     if (arglen == 1) {
922         newae(ctx->i_ae, "-Wl,-t");
923         break;
924     }
925     error(arg);
926     break;
927 case 'V':
928     if (arglen == 1) {
929         ctx->i_flags &= ~CW_F_ECHO;
930         newae(ctx->i_ae, "--version");
931         break;
932     }
933     error(arg);
934     break;
935 case 'v':
936     if (arglen == 1) {
937         warnings(ctx->i_ae);
938         break;
939     }
940     error(arg);
941     break;
942 case 'W':
943     if (strncmp(arg, "-Wp,-xc99", 9) == 0) {
944         /*
945          * gcc's preprocessor will accept c99
946          * regardless, so accept and ignore.
947          */
948         break;
949     }
950     if (strncmp(arg, "-Wa,", 4) == 0 ||

```

```

951     strcmp(arg, "-Wp,", 4) == 0 ||
952     strcmp(arg, "-Wl,", 4) == 0) {
953         newae(ctx->i_ae, arg);
954         break;
955     }
956     if (strcmp(arg, "-W0,-noglobal") == 0 ||
957         strcmp(arg, "-W0,-xglobalstatic") == 0) {
958         /*
959          * gcc doesn't prefix local symbols
960          * in debug mode, so this is not needed.
961          */
962         break;
963     }
964     if (strcmp(arg, "-W0,-Lt") == 0) {
965         /*
966          * Generate tests at the top of loops.
967          * There is no direct gcc equivalent, ignore.
968          */
969         break;
970     }
971     if (strcmp(arg, "-W0,-xdbggen=no%usedonly") == 0) {
972         newae(ctx->i_ae,
973             "-fno-eliminate-unused-debug-symbols");
974         newae(ctx->i_ae,
975             "-fno-eliminate-unused-debug-types");
976         break;
977     }
978     if (strcmp(arg, "-W2,-xwrap_int") == 0) {
979         /*
980          * Use the legacy behaviour (pre-SS11)
981          * for integer wrapping.
982          * gcc does not need this.
983          */
984         break;
985     }
986     if (strcmp(arg, "-Wd,-xsafe=unboundsym") == 0) {
987         /*
988          * Prevents optimizing away checks for
989          * unbound weak symbol addresses. gcc does
990          * not do this, so it's not needed.
991          */
992         break;
993     }
994     if (strcmp(arg, "-Wc,-xcode=", 11) == 0) {
995         xlate(ctx->i_ae, arg + 11, xcode_tbl);
996         break;
997     }
998     if (strcmp(arg, "-Wc,-Qiselect", 13) == 0) {
999         /*
1000          * Prevents insertion of register symbols.
1001          * gcc doesn't do this, so ignore it.
1002          */
1003         break;
1004     }
1005     if (strcmp(arg, "-Wc,-Qassembler-ounrefsym=0") == 0) {
1006         /*
1007          * Prevents optimizing away of static variables.
1008          * gcc does not do this, so it's not needed.
1009          */
1010         break;
1011     }
1012 #if defined(__x86)
1013     if (strcmp(arg, "-Wu,-save_args") == 0) {
1014         newae(ctx->i_ae, "-msave-args");
1015         break;
1016     }

```

```

1017 #endif /* __x86 */
1018     error(arg);
1019     break;
1020     case 'X':
1021         if (strcmp(arg, "-Xa") == 0 ||
1022             strcmp(arg, "-Xt") == 0) {
1023             break;
1024         }
1025         if (strcmp(arg, "-Xs") == 0) {
1026             Xsmode(ctx->i_ae);
1027             break;
1028         }
1029         error(arg);
1030         break;
1031     case 'x':
1032         if (arglen == 1)
1033             error(arg);
1034         switch (arg[2]) {
1035             case 'a':
1036                 if (strcmp(arg, "-xarch=", 7) == 0) {
1037                     mflag |= xlate_xtb(ctx->i_ae, arg + 7);
1038                     break;
1039                 }
1040                 error(arg);
1041                 break;
1042             case 'b':
1043                 if (strcmp(arg, "-xbuiltin=", 10) == 0) {
1044                     if (strcmp(arg + 10, "%all"))
1045                         newae(ctx->i_ae, "-fbuiltin");
1046                     break;
1047                 }
1048                 error(arg);
1049                 break;
1050             case 'C':
1051                 /* Accept C++ style comments -- ignore */
1052                 if (strcmp(arg, "-xCC") == 0)
1053                     break;
1054                 error(arg);
1055                 break;
1056             case 'c':
1057                 if (strcmp(arg, "-xc99=%all", 10) == 0) {
1058                     newae(ctx->i_ae, "-std=gnu99");
1059                     break;
1060                 }
1061                 if (strcmp(arg, "-xc99=%none", 11) == 0) {
1062                     newae(ctx->i_ae, "-std=gnu89");
1063                     break;
1064                 }
1065                 if (strcmp(arg, "-xchip=", 7) == 0) {
1066                     xlate(ctx->i_ae, arg + 7, xchip_tbl);
1067                     break;
1068                 }
1069                 if (strcmp(arg, "-xcode=", 7) == 0) {
1070                     xlate(ctx->i_ae, arg + 7, xcode_tbl);
1071                     break;
1072                 }
1073                 if (strcmp(arg, "-xcrossfile", 11) == 0)
1074                     break;
1075                 error(arg);
1076                 break;
1077             case 'd':
1078                 if (strcmp(arg, "-xdebugformat=", 14) == 0)
1079                     break;
1080                 error(arg);
1081                 break;
1082             case 'F':

```

```

1078     /*
1079     * Compile for mapfile reordering, or unused
1080     * section elimination, syntax can be -xF or
1081     * more complex, like -xF=%all -- ignore.
1082     */
1083     if (strncmp(arg, "-xF", 3) == 0)
1084         break;
1085     error(arg);
1086     break;
1087 case 'i':
1088     if (strncmp(arg, "-xinline", 8) == 0)
1089         /* No inlining; ignore */
1090         break;
1091     if (strncmp(arg, "-xildon") == 0 ||
1092         strncmp(arg, "-xildoff") == 0)
1093         /* No incremental linking; ignore */
1094         break;
1095     error(arg);
1096     break;
1097 #if defined(__x86)
1098 case 'm':
1099     if (strncmp(arg, "-xmodel=kernel") == 0) {
1100         newae(ctx->i_ae, "-ffreestanding");
1101         newae(ctx->i_ae, "-mno-red-zone");
1102         model = "-mcmmodel=kernel";
1103         nolibc = 1;
1104         break;
1105     }
1106     error(arg);
1107     break;
1108 #endif /* __x86 */
1109 case 'O':
1110     if (strncmp(arg, "-xO", 3) == 0) {
1111         size_t len = strlen(arg);
1112         char *s = NULL;
1113         int c = *(arg + 3);
1114         int level;

1116         if (len != 4 || !isdigit(c))
1117             error(arg);

1119         level = atoi(arg + 3);
1120         if (level > 5)
1121             error(arg);
1122         if (level >= 2) {
1123             /*
1124              * For gcc-3.4.x at -O2 we
1125              * need to disable optimizations
1126              * that break ON.
1127              */
1128             optim_disable(ctx->i_ae, level);
1129             /*
1130              * limit -xO3 to -O2 as well.
1131              */
1132             level = 2;
1133         }
1134         if (asprintf(&s, "-O%d", level) == -1)
1135             nomem();
1136         newae(ctx->i_ae, s);
1137         free(s);
1138         break;
1139     }
1140     error(arg);
1141     break;
1142 case 'r':
1143     if (strncmp(arg, "-xregs=", 7) == 0) {

```

```

1144         xlate(ctx->i_ae, arg + 7, xregs_tbl);
1145         break;
1146     }
1147     error(arg);
1148     break;
1149 case 's':
1150     if (strncmp(arg, "-xs") == 0 ||
1151         strncmp(arg, "-xspace") == 0 ||
1152         strncmp(arg, "-xstrconst") == 0)
1153         break;
1154     error(arg);
1155     break;
1156 case 't':
1157     if (strncmp(arg, "-xtarget=", 9) == 0) {
1158         xlate(ctx->i_ae, arg + 9, xtarget_tbl);
1159         break;
1160     }
1161     error(arg);
1162     break;
1163 case 'e':
1164 case 'h':
1165 case 'l':
1166 default:
1167     error(arg);
1168     break;
1169 }
1170 break;
1171 case 'Y':
1172     if (arglen == 1) {
1173         if ((arg = *++ctx->i_oldargv) == NULL ||
1174             *arg == '\0')
1175             error("-Y");
1176         ctx->i_oldargc--;
1177         arglen = strlen(arg + 1);
1178     } else {
1179         arg += 2;
1180     }
1181     /* Just ignore -YS,... for now */
1182     if (strncmp(arg, "S", 2) == 0)
1183         break;
1184     if (strncmp(arg, "l", 2) == 0) {
1185         char *s = strdup(arg);
1186         s[0] = '-';
1187         s[1] = 'B';
1188         newae(ctx->i_ae, s);
1189         free(s);
1190         break;
1191     }
1192     if (strncmp(arg, "I", 2) == 0) {
1193         char *s = strdup(arg);
1194         s[0] = '-';
1195         s[1] = 'I';
1196         newae(ctx->i_ae, "-nostdinc");
1197         newae(ctx->i_ae, s);
1198         free(s);
1199         break;
1200     }
1201     error(arg);
1202     break;
1203 case 'Q':
1204     /*
1205     * We could map -Qy into -Wl,-Qy etc.
1206     */
1207 default:
1208     error(arg);
1209     break;

```

```

1210     }
1211 }
1213 free(nameflag);

1215 /*
1216  * When compiling multiple source files in a single invocation some
1217  * compilers output objects into the current directory with
1218  * predictable and conventional names.
1219  *
1220  * We prevent any attempt to compile multiple files at once so that
1221  * any such objects created by a shadow can't escape into a later
1222  * link-edit.
1223  */
1224 if (c_files > 1 && op != CW_O_PREPROCESS) {
1225     errx(2, "multiple source files are "
1226         "allowed only with -E or -P");
1227 }

1229 /*
1230  * Make sure that we do not have any unintended interactions between
1231  * the xarch options passed in and the version of the Studio compiler
1232  * used.
1233  */
1234 if ((mflag & (SS11|SS12)) == (SS11|SS12)) {
1235     errx(2,
1236         "Conflicting \"-xarch=\" flags (both Studio 11 and 12)\n");
1237 }

1239 switch (mflag) {
1240 case 0:
1241     /* FALLTHROUGH */
1242 case M32:
1243 #if defined(__sparc)
1244     /*
1245      * Only -m32 is defined and so put in the missing xarch
1246      * translation.
1247      */
1248     newae(ctx->i_ae, "-mcpu=v8");
1249     newae(ctx->i_ae, "-mno-v8plus");
1250 #endif
1251     break;
1252 case M64:
1253 #if defined(__sparc)
1254     /*
1255      * Only -m64 is defined and so put in the missing xarch
1256      * translation.
1257      */
1258     newae(ctx->i_ae, "-mcpu=v9");
1259 #endif
1260     break;
1261 case SS12:
1262 #if defined(__sparc)
1263     /* no -m32/-m64 flag used - this is an error for sparc builds */
1264     (void) fprintf(stderr, "No -m32/-m64 flag defined\n");
1265     exit(2);
1266 #endif
1267     break;
1268 case SS11:
1269     /* FALLTHROUGH */
1270 case (SS11|M32):
1271 case (SS11|M64):
1272     break;
1273 case (SS12|M32):
1274 #if defined(__sparc)
1275     /*

```

```

1276     * Need to add in further 32 bit options because with SS12
1277     * the xarch=sparcvis option can be applied to 32 or 64
1278     * bit, and so the translation table (xtbl) cannot handle
1279     * that.
1280     */
1281     newae(ctx->i_ae, "-mv8plus");
1282 #endif
1283     break;
1284 case (SS12|M64):
1285     break;
1286 default:
1287     (void) fprintf(stderr,
1288         "Incompatible -xarch= and/or -m32/-m64 options used.\n");
1289     exit(2);
1290 }

1292 if (ctx->i_flags & CW_F_SHADOW) {
1293     if (op == CW_O_PREPROCESS)
1294         exit(0);
1295     else if (op == CW_O_LINK && c_files == 0)
1296         exit(0);
1297 }

1299 if (model != NULL)
1300     newae(ctx->i_ae, model);
1301 if (!nolibc)
1302     newae(ctx->i_ae, "-lc");
1303 if (!seen_o && (ctx->i_flags & CW_F_SHADOW)) {
1304     ctx->i_discard = discard_file_name(NULL);
1305     newae(ctx->i_ae, "-o");
1306     newae(ctx->i_ae, ctx->i_discard);
1307 }
1308 }
_____unchanged_portion_omitted_____

```

new/usr/src/uts/intel/cxgbe/cxgbe/Makefile

1

```
*****
1756 Fri Dec 7 21:21:59 2018
new/usr/src/uts/intel/cxgbe/cxgbe/Makefile
10051 cw(1) shouldn't have to translate -g
*****
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright (c) 2013 by Chelsio Communications, Inc. All rights reserved.
14 #
15 #
16 #
17 # This makefile drives the production of the Chelsio Terminator 4 10G Ethernet
18 # (CXGBE) driver modules on x86 systems.
19 #
20 #
21 #
22 # Paths to the base of the uts directory trees
23 #
24 UTSBASE = ../../..
25 #
26 #
27 # Define the module and object file sets.
28 #
29 MODULE          = cxgbe
30 OBJECTS          = $(CXGBE_OBJS:%=$(OBJS_DIR)/%)
31 LINTS            = $(CXGBE_OBJS:%.o=$(LINTS_DIR)/%.ln)
32 ROOTMODULE      = $(ROOT_DRV_DIR)/$(MODULE)
33 #
34 #
35 # Include common rules
36 #
37 include $(UTSBASE)/intel/Makefile.intel
38 #
39 #
40 # Define targets
41 #
42 ALL_TARGET = $(BINARY)
43 LINT_TARGET = $(MODULE).lint
44 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)
45 #
46 CFLAGS += -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
47 CFLAGS += -g -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
48 CFLAGS += -I$(UTSBASE)/common/io/cxgbe/t4nex -I$(UTSBASE)/common/io/cxgbe/shared
49 #
50 # Driver depends
51 #
52 LDFLAGS += -dy -N misc/mac -N drv/ip
53 #
54 #
55 # Lint
56 #
57 LINTFLAGS += -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
58 LINTFLAGS += -I$(UTSBASE)/common/io/cxgbe/t4nex -I$(UTSBASE)/common/io/cxgbe/shared
59 LINTFLAGS += -Xc99=%all
```

new/usr/src/uts/intel/cxgbe/cxgbe/Makefile

2

```
61 #
62 # Default build targets.
63 #
64 .KEEP_STATE:
65 #
66 def:          $(DEF_DEPS)
67 #
68 all:         $(ALL_DEPS)
69 #
70 clean:       $(CLEAN_DEPS)
71 #
72 clobber:     $(CLOBBER_DEPS)
73 #
74 lint:        $(LINT_DEPS)
75 #
76 modlintlib:  $(MODLINTLIB_DEPS)
77 #
78 clean.lint:  $(CLEAN_LINT_DEPS)
79 #
80 install:     $(INSTALL_DEPS)
81 #
82 #
83 # Include common targets.
84 #
85 include $(UTSBASE)/intel/Makefile.targ
```

new/usr/src/uts/intel/cxgbe/t4nex/Makefile

1

```
*****
2983 Fri Dec 7 21:21:59 2018
new/usr/src/uts/intel/cxgbe/t4nex/Makefile
10051 cw(1) shouldn't have to translate -g
*****

1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #

12 #
13 # Copyright (c) 2013 by Chelsio Communications, Inc. All rights reserved.
14 #

16 #
17 # This makefile drives the production of the Chelsio Terminator 4 10G Ethernet
18 # (CXGBE) driver nexus modules on x86 systems.
19 #

21 #
22 # Paths to the base of the uts directory trees
23 #
24 UTSBASE = ../../..

26 #
27 # Define the module and object file sets.
28 #
29 MODULE      = t4nex
30 OBJECTS     = $(CXGBE_COM_OBJS:%=$(OBJDIR)/%) \
31             $(CXGBE_FW_OBJS:%=$(OBJDIR)/%) \
32             $(CXGBE_NEX_OBJS:%=$(OBJDIR)/%)
33 LINTS       = $(CXGBE_COM_OBJS:%.o=$(LINTSDIR)/%.ln) \
34             $(CXGBE_FW_OBJS:%.o=$(LINTSDIR)/%.ln) \
35             $(CXGBE_NEX_OBJS:%.o=$(LINTSDIR)/%.ln)
36 ROOTMODULE = $(ROOTDRVDIR)/$(MODULE)

38 #
39 # Include common rules
40 #
41 include $(UTSBASE)/intel/Makefile.intel

43 #
44 # Define targets
45 #
46 ALL_TARGET = $(BINARY)
47 LINT_TARGET = $(MODULE).lint
48 INSTALL_TARGET = $(BINARY) $(ROOTMODULE)

50 CFLAGS += -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
51 CFLAGS += -g -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
52             -I$(UTSBASE)/common/io/cxgbe/t4nex -I$(UTSBASE)/common/io/cxgbe/shared \
53             -I$(UTSBASE)/common/io/cxgbe/firmware

54 #
55 # Driver depends
56 #
57 LDFLAGS += -dy -N misc/mac -N drv/ip

59 #
60 # Lint
```

new/usr/src/uts/intel/cxgbe/t4nex/Makefile

2

```
61 #
62 LINTFLAGS += -I$(UTSBASE)/common/io/cxgbe -I$(UTSBASE)/common/io/cxgbe/common \
63             -I$(UTSBASE)/common/io/cxgbe/t4nex -I$(UTSBASE)/common/io/cxgbe/shared \
64             -I$(UTSBASE)/common/io/cxgbe/firmware
65 LINTFLAGS += -Xc99=%all

67 #
68 # A lot of these come from the fact that there are static inlines and
69 # that this driver leverages various C99 and some GNU C extensions.
70 #
71 LINTTAGS += -erroff=E_STATIC_UNUSED
72 LINTTAGS += -erroff=E_FUNC_ARG_UNUSED
73 LINTTAGS += -erroff=E_ZERO_OR_NEGATIVE_SUBSCRIPT
74 LINTTAGS += -erroff=E_ZERO_SIZED_STRUCT_UNION
75 LINTTAGS += -erroff=E_ANONYMOUS_UNION_DECL
76 LINTTAGS += -erroff=E_CONSTANT_CONDITION

78 #
79 # These have been manually verified at the moment. When updating this
80 # driver, please manually check the following lint warnings. Note some
81 # of these are due to limitations in Sun Studio.
82 #
83 LINTTAGS += -erroff=E_BAD_PTR_CAST_ALIGN
84 LINTTAGS += -erroff=E_ENUM_VAL_OVERFLOWS_INT_MAX
85 LINTTAGS += -erroff=E_FUNC_NO_RET_VAL
86 LINTTAGS += -erroff=E_NOP_IF_STMT
87 LINTTAGS += -erroff=E_FUNC_SET_NOT_USED
88 LINTTAGS += -erroff=E_FUNC_RET_ALWAYS_IGNORED2
89 LINTTAGS += -erroff=E_FUNC_RET_MAYBE_IGNORED2
90 LINTTAGS += -erroff=E_CASE_FALLTHRU
91 LINTTAGS += -erroff=E_ASSIGN_NARROW_CONV

93 #
94 # Default build targets.
95 #
96 .KEEP_STATE:

98 def:          $(DEF_DEPS)

100 all:         $(ALL_DEPS)

102 clean:       $(CLEAN_DEPS)

104 clobber:     $(CLOBBER_DEPS)

106 lint:        $(LINT_DEPS)

108 modlintlib:  $(MODLINTLIB_DEPS)

110 clean.lint:  $(CLEAN_LINT_DEPS)

112 install:     $(INSTALL_DEPS)

114 #
115 # Include common targets.
116 #
117 include $(UTSBASE)/intel/Makefile.targ
```