

new/usr/src/cmd/make/bin/make/common/ar.cc

1

24875 Wed May 20 11:09:46 2015

new/usr/src/cmd/make/bin/make/common/ar.cc

make: remove SCCS ident stuff

```
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)ar.cc 1.28 06/12/12
27 */
28
29 #pragma ident    "@(#)ar.cc      1.28   06/12/12"
30
31
32 /*
33  * ar.c
34  *
35  * Deal with the lib.a(member.o) and lib.a((entry-point)) notations
36  *
37  * Look inside archives for notations a(b) and a((b))
38  * a(b) is file member b in archive a
39  * a((b)) is entry point b in object archive a
40  *
41  * For 6.0, create a make which can understand all archive
42  * formats. This is kind of tricky, and <ar.h> isnt any help.
43  */
44
45 /*
46  * Included files
47  */
48 #include <avo/avo_alloc.h>          /* alloca() */
49 #include <ar.h>
50 #include <errno.h>                  /* errno */
51 #include <fcntl.h>                  /* open() */
52 #include <mk/defs.h>
53 #include <mksh/misc.h>              /* retmem_mb() */
54
55 #if defined(SUN5_0) || defined(HP_UX) || defined(linux)
56 struct ranlib {
57     union {
58         off_t   ran_strx;          /* string table index of */
59         char    *ran_name;         /* symbol defined by */
60     };
61     ran_un;
62     off_t   ran_off;              /* library member at this offset */
63 };
64
65 unchanged_portion_omitted
```

```

*****
4797 Wed May 20 11:09:47 2015
new/usr/src/cmd/make/bin/make/common/default.mk.file
make: remove SCCS ident stuff
*****
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 1993 Sun Microsystems, Inc. All rights reserved.
22 # Use is subject to license terms.
23 #
24 # @(#)default.mk.file 1.4 06/12/12
25 #

26 SUFFIXES = .o .c .c~ .cc .cc~ .s .s~ .S .S~ .ln .f .f~ .F .F~ .l .l~ \
27           .mod .mod~ .sym .def .def~ .p .p~ .r .r~ .y .y~ .h .h~ .sh .sh~ \
28           .cps .cps~
.SUFFIXES: $(SUFFIXES)

30 # OUTPUT_OPTION should be defined to "-o $" when
31 # the default rules are used for non-local files.
32 OUTPUT_OPTION=

34 #      C language section.
35 CC=cc
36 CFLAGS=
37 CPPFLAGS=
38 LINT=lint
39 LINTFLAGS=
40 COMPILE.c=$(CC) $(CFLAGS) $(CPPFLAGS) -target $(TARGET_ARCH:-%=) -c
41 LINK.c=$(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS) -target $(TARGET_ARCH:-%=)
42 LINT.c=$(LINT) $(LINTFLAGS) $(CPPFLAGS) $(TARGET_ARCH)
43 .c:
44     $(LINK.c) -o $@ $< $(LDLIBS)
45 .c.ln:
46     $(LINT.c) $(OUTPUT_OPTION) -i $<
47 .c.o:
48     $(COMPILE.c) $(OUTPUT_OPTION) $<
49 .c.a:
50     $(COMPILE.c) -o $% $<
51     $(AR) $(ARFLAGS) $@ $%
52     $(RM) $%

54 #      C language section. yacc.
55 YACC=yacc
56 YFLAGS=
57 YACC.y=$(YACC) $(YFLAGS)
58 .y:
59     $(YACC.y) $<

```

```

60     $(LINK.c) -o $@ y.tab.c $(LDLIBS)
61     $(RM) y.tab.c
62 .y.c:
63     $(YACC.y) $<
64     mv y.tab.c $@
65 .y.ln:
66     $(YACC.y) $<
67     $(LINT.c) -o $@ -i y.tab.c
68     $(RM) y.tab.c
69 .y.o:
70     $(YACC.y) $<
71     $(COMPILE.c) -o $@ y.tab.c
72     $(RM) y.tab.c

74 #      C language section. lex.
75 LEX=lex
76 LFLAGS=
77 LEX.l=$(LEX) $(LFLAGS) -t
78 .l:
79     $(RM) $*.c
80     $(LEX.l) $< > $*.c
81     $(LINK.c) -o $@ $*.c -ll $(LDLIBS)
82     $(RM) $*.c
83 .l.c :
84     $(RM) $@
85     $(LEX.l) $< > $@
86 .l.ln:
87     $(RM) $*.c
88     $(LEX.l) $< > $*.c
89     $(LINT.c) -o $@ -i $*.c
90     $(RM) $*.c
91 .l.o:
92     $(RM) $*.c
93     $(LEX.l) $< > $*.c
94     $(COMPILE.c) -o $@ $*.c
95     $(RM) $*.c

97 #      C++ language section.
98 CCC=CC
99 CCFLAGS=
100 COMPILE.cc=$(CCC) $(CCFLAGS) $(CPPFLAGS) $(TARGET_ARCH) -c
101 LINK.cc=$(CCC) $(CCFLAGS) $(CPPFLAGS) $(LDFLAGS) $(TARGET_ARCH)
102 .cc:
103     $(LINK.cc) -o $@ $< $(LDLIBS)
104 .cc.o:
105     $(COMPILE.cc) $(OUTPUT_OPTION) $<
106 .cc.a:
107     $(COMPILE.cc) -o $% $<
108     $(AR) $(ARFLAGS) $@ $%
109     $(RM) $%

111 #      FORTRAN section.
112 FC=f77
113 FFLAGS=
114 COMPILE.f=$(FC) $(FFLAGS) $(TARGET_ARCH) -c
115 LINK.f=$(FC) $(FFLAGS) $(LDFLAGS) $(TARGET_ARCH)
116 COMPILE.F=$(FC) $(FFLAGS) $(CPPFLAGS) $(TARGET_ARCH) -c
117 LINK.F=$(FC) $(FFLAGS) $(CPPFLAGS) $(LDFLAGS) $(TARGET_ARCH)
118 .f:
119     $(LINK.f) -o $@ $< $(LDLIBS)
120 .f.o:
121     $(COMPILE.f) $(OUTPUT_OPTION) $<
122 .f.a:
123     $(COMPILE.f) -o $% $<
124     $(AR) $(ARFLAGS) $@ $%
125     $(RM) $%

```

```

126 .F:
127     $(LINK.F) -o $@ $< $(LDLIBS)
128 .F.o:
129     $(COMPILE.F) $(OUTPUT_OPTION) $<
130 .F.a:
131     $(COMPILE.F) -o $$ $<
132     $(AR) $(ARFLAGS) $@ $$
133     $(RM) $$

135 #      FORTRAN section. ratfor.
136 RFLAGS=
137 COMPILE.r=$(FC) $(FFLAGS) $(RFLAGS) $(TARGET_ARCH) -c
138 LINK.r=$(FC) $(FFLAGS) $(RFLAGS) $(LDFLAGS) $(TARGET_ARCH)
139 .r:
140     $(LINK.r) -o $@ $< $(LDLIBS)
141 .r.o:
142     $(COMPILE.r) $(OUTPUT_OPTION) $<
143 .r.a:
144     $(COMPILE.r) -o $$ $<
145     $(AR) $(ARFLAGS) $@ $$
146     $(RM) $$

148 #      Modula-2 section.
149 M2C=m2c
150 M2FLAGS=
151 MODFLAGS=
152 DEFFLAGS=
153 COMPILE.def=$(M2C) $(M2FLAGS) $(DEFFLAGS) $(TARGET_ARCH)
154 COMPILE.mod=$(M2C) $(M2FLAGS) $(MODFLAGS) $(TARGET_ARCH)
155 .def.sym:
156     $(COMPILE.def) -o $@ $<
157 .mod:
158     $(COMPILE.mod) -o $@ -e $@ $<
159 .mod.o:
160     $(COMPILE.mod) -o $@ $<
161 .mod.a:
162     $(COMPILE.mod) -o $$ $<
163     $(AR) $(ARFLAGS) $@ $$
164     $(RM) $$

166 #      Pascal section.
167 PC=pc
168 PFLAGS=
169 COMPILE.p=$(PC) $(PFLAGS) $(CPPFLAGS) $(TARGET_ARCH) -c
170 LINK.p=$(PC) $(PFLAGS) $(CPPFLAGS) $(LDFLAGS) $(TARGET_ARCH)
171 .p:
172     $(LINK.p) -o $@ $< $(LDLIBS)
173 .p.o:
174     $(COMPILE.p) $(OUTPUT_OPTION) $<
175 .p.a:
176     $(COMPILE.p) -o $$ $<
177     $(AR) $(ARFLAGS) $@ $$
178     $(RM) $$

180 #      Assembly section.
181 AS=as
182 ASFLAGS=
183 COMPILE.s=$(AS) $(ASFLAGS) $(TARGET_MACH)
184 COMPILE.S=$(CC) $(ASFLAGS) $(CPPFLAGS) $(TARGET_ARCH) -c
185 .s.o:
186     $(COMPILE.s) -o $@ $<
187 .s.a:
188     $(COMPILE.s) -o $$ $<
189     $(AR) $(ARFLAGS) $@ $$
190     $(RM) $$
191 .S.o:

```

```

192     $(COMPILE.S) -o $@ $<
193 .S.a:
194     $(COMPILE.S) -o $$ $<
195     $(AR) $(ARFLAGS) $@ $$
196     $(RM) $$

198 #      Shell section.
199 .sh:
200     $(RM) $@
201     cat $< > $@
202     chmod +x $@

204 #      NeWS section
205 CPS=cps
206 CPSFLAGS=
207 .cps.h:
208     $(CPS) $(CPSFLAGS) $*.cps

210 #      Miscellaneous section.
211 LD=ld
212 LDFLAGS=
213 LDLIBS=
214 MAKE=make
215 RM=rm -f
216 AR=ar
217 ARFLAGS=rv
218 GET=/usr/sccs/get
219 GFLAGS=

221 markfile.o:      markfile
222     echo "static char _sccsid[] = \"`grep @'(#)' markfile`\";" > markfile.c
223     cc -c markfile.c
224     $(RM) markfile.c

226 SCCSFLAGS=
227 SCCSGETFLAGS=-s
228 .SCCS_GET:
229     sccs $(SCCSFLAGS) get $(SCCSGETFLAGS) $@ -G$@

```

new/usr/src/cmd/make/bin/make/common/depvar.cc

1

3397 Wed May 20 11:09:47 2015

new/usr/src/cmd/make/bin/make/common/depvar.cc

make: remove SCCS ident stuff

```
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 1995 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)depvar.cc 1.14 06/12/12
27 */
28
29 #pragma ident      "@(#)depvar.cc 1.14 06/12/12"
30
31
32 /*
33 * Included files
34 */
35 #include <mk/defs.h>
36 #include <mksh/misc.h>          /* getmem() */
37
38 /*
39 * This file deals with "Dependency Variables".
40 * The "-V var" command line option is used to indicate
41 * that var is a dependency variable. Used in conjunction with
42 * the -P option the user is asking if the named variables affect
43 * the dependencies of the given target.
44 */
45
46 struct _Depvar {
47     Name      name;          /* Name of variable */
48     struct _Depvar *next;    /* Linked list */
49     Boolean    cmdline;      /* Macro defined on the cmdline? */
50 };
51
52 unchanged_portion_omitted
```

new/usr/src/cmd/make/bin/make/common/dist.cc

1

```
*****
15419 Wed May 20 11:09:48 2015
new/usr/src/cmd/make/bin/make/common/dist.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)dist.cc 1.35 06/12/12
27 */
29 #pragma ident      "@(#)dist.cc  1.25  96/03/12"

26 #ifdef DISTRIBUTED
27 /*
28 *      dist.cc
29 *
30 *      Deal with the distributed processing
31 */

33 #include <avo/err.h>
34 #include <avo/find_dir.h>
35 #include <avo/util.h>
36 #include <dm/Avo_AcknowledgeMsg.h>
37 #include <dm/Avo_DoJobMsg.h>
38 #include <dm/Avo_JobResultMsg.h>
39 #include <mk/defs.h>
40 #include <mksh/misc.h>          /* getmem() */
41 #include <rw/pstream.h>
42 #include <rw/queuecol.h>
43 #include <rw/xdrstrea.h>
44 #include <signal.h>
45 #ifdef linux
46 #include <sstream>
47 using namespace std;
48 #else
49 #include <strstream.h>
50 #endif
51 #include <sys/stat.h>          /* stat() */
52 #include <sys/types.h>
53 #include <sys/wait.h>
54 #include <unistd.h>
55 #include <errno.h>
```

new/usr/src/cmd/make/bin/make/common/dist.cc

2

```
57 /*
58  * Defined macros
59  */

61 #define AVO_BLOCK_INTERRUPTS sigfillset(&newset) ; \
62     sigprocmask(SIG_SETMASK, &newset, &oldset)
63
64 #define AVO_UNBLOCK_INTERRUPTS \
65     sigprocmask(SIG_SETMASK, &oldset, &newset)

68 /*
69  * typedefs & structs
70  */

72 /*
73  * Static variables
74  */
75 int          dmake_ifd;
76 FILE*        dmake_ifp;
77 XDR          xdrs_in;

79 int          dmake_ofd;
80 FILE*        dmake_ofp;
81 XDR          xdrs_out;

83 // These instances are required for the RWFactory.
84 static Avo_JobResultMsg dummyJobResultMsg;
85 static Avo_AcknowledgeMsg dummyAcknowledgeMsg;
86 static int firstAcknowledgeReceived = 0;
87
88 int          rxmPid = 0;

90 /*
91  * File table of contents
92  */
93 static void  set_dmake_env_vars(void);

95 /*
96  * void
97  * startup_rxm(void)
98  *
99  * When startup_rxm() is called, read_command_options() and
100 * read_files_and_state() have already been read, so DMake
101 * will now know what options to pass to rxm.
102  *
103  * rxm
104  * [ -k ] [ -n ]
105  * [ -c <dmake_rcfile> ]
106  * [ -g <dmake_group> ]
107  * [ -j <dmake_max_jobs> ]
108  * [ -m <dmake_mode> ]
109  * [ -o <dmake_odir> ]
110  * <read_fd> <write_fd>
111  *
112  * rxm will, among other things, read the rc file.
113  *
114  */
115 void
116 startup_rxm(void)
117 {
118     Name          dmake_name;
119     Name          dmake_value;
120     Avo_err       *find_run_dir_err;
121     int           pipel[2], pipe2[2];
122     Property      prop;
```

```

123     char          *run_dir;
124     char          rxm_command[MAXPATHLEN];
125     int           rxm_debug = 0;

127     int           length;
128     char *        env;

130     firstAcknowledgeReceived = 0;
131     /*
132     * Create two pipes, one for dmake->rxm, and one for rxm->dmake.
133     * pipe1 is dmake->rxm,
134     * pipe2 is rxm->dmake.
135     */
136     if ((pipe(pipe1) < 0) || (pipe(pipe2) < 0)) {
137         fatal(catgets(catd, 1, 245, "pipe() failed: %s"), errmsg(errno))
138     }

140     set_dmake_env_vars();

142     if ((rxmPid = fork()) < 0) { /* error */
143         fatal(catgets(catd, 1, 246, "fork() failed: %s"), errmsg(errno))
144     } else if (rxmPid > 0) { /* parent, dmake */
145         dmake_ofd = pipe1[1]; // write side of pipe
146         if (!(dmake_ofd = fdopen(dmake_ofd, "a"))) {
147             fatal(catgets(catd, 1, 247, "fdopen() failed: %s"), errmsg(errno))
148         }
149         xdrstdio_create(&xdrs_out, dmake_ofd, XDR_ENCODE);

151         dmake_ifd = pipe2[0]; // read side of pipe
152         if (!(dmake_ifd = fdopen(dmake_ifd, "r"))) {
153             fatal(catgets(catd, 1, 248, "fdopen() failed: %s"), errmsg(errno))
154         }
155         xdrstdio_create(&xdrs_in, dmake_ifd, XDR_DECODE);

157         close(pipe1[0]); // read side
158         close(pipe2[1]); // write side
159     } else { /* child, rxm */
160         close(pipe1[1]); // write side
161         close(pipe2[0]); // read side

163         /* Find the run directory of dmake, for rxm. */
164         find_run_dir_err = avo_find_run_dir(&run_dir);
165         if (find_run_dir_err) {
166             delete find_run_dir_err;
167             /* Use the path to find rxm. */
168             (void) sprintf(rxm_command, NOCATGETS("rxm"));
169         } else {
170             /* Use the run dir of dmake for rxm. */
171             (void) sprintf(rxm_command, NOCATGETS("%s/rxm"), run_dir);
172         }

174         if (continue_after_error) {
175             (void) strcat(rxm_command, NOCATGETS(" -k"));
176         }
177         if (do_not_exec_rule) {
178             (void) strcat(rxm_command, NOCATGETS(" -n"));
179         }
180         if (rxm_debug) {
181             (void) strcat(rxm_command, NOCATGETS(" -S"));
182         }
183         if (send_mtool_msgs) {
184             (void) sprintf(&rxm_command[strlen(rxm_command)],
185                 NOCATGETS(" -O %d"),
186                 mtool_msgs_fd);
187         }
188         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_RCFILE"));

```

```

189         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
190         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
191             ((dmake_value = prop->body.macro.value) != NULL)) {
192             (void) sprintf(&rxm_command[strlen(rxm_command)],
193                 NOCATGETS(" -c %s"),
194                 dmake_value->string_mb);
195         } else {
196             length = 2 + strlen(NOCATGETS("DMAKE_RCFILE"));
197             env = getmem(length);
198             (void) sprintf(env,
199                 "%s=",
200                 NOCATGETS("DMAKE_RCFILE"));
201             (void) putenv(env);
202         }
203         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_GROUP"));
204         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
205         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
206             ((dmake_value = prop->body.macro.value) != NULL)) {
207             (void) sprintf(&rxm_command[strlen(rxm_command)],
208                 NOCATGETS(" -g %s"),
209                 dmake_value->string_mb);
210         } else {
211             length = 2 + strlen(NOCATGETS("DMAKE_GROUP"));
212             env = getmem(length);
213             (void) sprintf(env,
214                 "%s=",
215                 NOCATGETS("DMAKE_GROUP"));
216             (void) putenv(env);
217         }
218         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_MAX_JOBS"));
219         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
220         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
221             ((dmake_value = prop->body.macro.value) != NULL)) {
222             (void) sprintf(&rxm_command[strlen(rxm_command)],
223                 NOCATGETS(" -j %s"),
224                 dmake_value->string_mb);
225         } else {
226             length = 2 + strlen(NOCATGETS("DMAKE_MAX_JOBS"));
227             env = getmem(length);
228             (void) sprintf(env,
229                 "%s=",
230                 NOCATGETS("DMAKE_MAX_JOBS"));
231             (void) putenv(env);
232         }
233         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_MODE"));
234         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
235         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
236             ((dmake_value = prop->body.macro.value) != NULL)) {
237             (void) sprintf(&rxm_command[strlen(rxm_command)],
238                 NOCATGETS(" -m %s"),
239                 dmake_value->string_mb);
240         } else {
241             length = 2 + strlen(NOCATGETS("DMAKE_MODE"));
242             env = getmem(length);
243             (void) sprintf(env,
244                 "%s=",
245                 NOCATGETS("DMAKE_MODE"));
246             (void) putenv(env);
247         }
248         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_ODIR"));
249         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
250         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
251             ((dmake_value = prop->body.macro.value) != NULL)) {
252             (void) sprintf(&rxm_command[strlen(rxm_command)],
253                 NOCATGETS(" -o %s"),
254                 dmake_value->string_mb);

```

```
255     } else {
256         length = 2 + strlen(NOCATGETS("DMAKE_ODIR"));
257         env = getmem(length);
258         (void) sprintf(env,
259             "%s=",
260             NOCATGETS("DMAKE_ODIR"));
261         (void) putenv(env);
262     }
263
264     (void) sprintf(&rxm_command[strlen(rxm_command)],
265         NOCATGETS(" %d %d"),
266         pipe1[0], pipe2[1]);
267 #ifdef linux
268     execl(NOCATGETS("/bin/sh"),
269 #else
270     execl(NOCATGETS("/usr/bin/sh"),
271 #endif
272         NOCATGETS("sh"),
273         NOCATGETS("-c"),
274         rxm_command,
275         (char *)NULL);
276     _exit(127);
277 }
278 }
unchanged_portion_omitted
```

new/usr/src/cmd/make/bin/make/common/dmake.cc

1

926 Wed May 20 11:09:48 2015

new/usr/src/cmd/make/bin/make/common/dmake.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)dmake.cc 1.2 06/12/12
27 */
28
29 #pragma ident      "@(#)dmake.cc 1.2 06/12/12"
```



```

*****
105392 Wed May 20 11:09:49 2015
new/usr/src/cmd/make/bin/make/common/doname.cc
make: remove SCCS ident stuff
*****
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 */
25 /*
26 * @(#)doname.cc 1.115 06/12/12
27 */
28
29 #pragma ident      "@(#)doname.cc 1.115 06/12/12"
30
31
32 /*
33  * doname.c
34  *
35  * Figure out which targets are out of date and rebuild them
36  */
37
38 /*
39  * Included files
40  */
41 #include <avo/avo_alloc.h>          /* alloc() */
42 #if defined(TEAMWARE_MAKE_CMN)
43 #include <avo/util.h>              /* avo_get_user(), avo_hostname() */
44 #endif
45
46 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolík */
47 #include <avo/strings.h> /* AVO_STRDUP() */
48 #include <dm/Avo_MToolJobResultMsg.h>
49 #include <dm/Avo_MToolJobStartMsg.h>
50 #include <dm/Avo_MToolRsrcInfoMsg.h>
51 #include <dm/Avo_macro_defs.h> /* AVO_BLOCK_INTERRUPTS & AVO_UNBLOCK_INTER
52 #include <dmthread/Avo_ServerState.h>
53 #include <rw/pstream.h>
54 #include <rw/xdrstrea.h>
55 #endif
56
57 #include <fcntl.h>
58 #include <mk/defs.h>
59 #include <mksh/il8n.h> /* get_char_semantics_value() */
60 #include <mksh/macro.h> /* getvar(), expand_value() */
61 #include <mksh/misc.h> /* getmem() */
62 #include <poll.h>

```

```

58 #ifdef PARALLEL
59 #include <rx/api.h>
60 #endif
61
62 #include <signal.h>
63
64 #ifndef HP_UX
65 #include <stropts.h>
66 #endif
67
68 #include <sys/errno.h>
69 #include <sys/stat.h>
70 #include <sys/types.h>
71 #include <sys/utsname.h> /* uname() */
72 #include <sys/wait.h>
73 #include <unistd.h> /* close() */
74
75 /*
76  * Defined macros
77  */
78 #ifndef PARALLEL
79 #define LOCALHOST "localhost"
80 #endif
81
82 #define MAXRULES 100
83
84 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolík */
85 #define SEND_MTOOL_MSG(cmds) \
86     if (send_mtool_msgs) { \
87         cmds \
88     }
89 #else
90 #define SEND_MTOOL_MSG(cmds)
91 #endif
92
93 // Sleep for .1 seconds between stat()'s
94 const int STAT_RETRY_SLEEP_TIME = 100000;
95
96 /*
97  * typedefs & structs
98  */
99
100 /*
101  * Static variables
102  */
103 static char hostName[MAXNAMELEN] = "";
104 static char userName[MAXNAMELEN] = "";
105
106 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolík */
107 static FILE *mtool_msgs_fp;
108 static XDR xdrs;
109 static int sent_rsrc_info_msg = 0;
110 #endif
111
112 static int second_pass = 0;
113
114 /*
115  * File table of contents
116  */
117 extern Doname doname_check(register Name target, register Boolean do_g
118 extern Doname doname(register Name target, register Boolean do_get, re
119 static Boolean check_dependencies(Doname *result, Property line, Boolea
120 void dynamic_dependencies(Name target);
121 static Doname run_command(register Property line, Boolean print_machin
122 extern Doname execute_serial(Property line);

```

```

123 extern Name      vpath_translation(register Name cmd);
124 extern void      check_state(Name temp_file_name);
125 static void      read_dependency_file(register Name filename);
126 static void      check_read_state_file(void);
127 static void      do_assign(register Name line, register Name target);
128 static void      build_command_strings(Name target, register Property lin
129 static Doname    touch_command(register Property line, register Name targ
130 extern void      update_target(Property line, Doname result);
131 static Doname    sccs_get(register Name target, register Property *comman
132 extern void      read_directory_of_file(register Name file);
133 static void      add_pattern_conditionals(register Name target);
134 extern void      set_locals(register Name target, register Property old_l
135 extern void      reset_locals(register Name target, register Property old
136 extern Boolean   check_auto_dependencies(Name target, int auto_count, Nam
137 static void      delete_query_chain(Chain ch);

139 // From read2.cc
140 extern Name      normalize_name(register wchar_t *name_string, register i

143 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
144     static void      append_job_result_msg(Avo_MToolJobResultMsg *job
145     static int       pollResults(char *outFn, char *errFn, char *host
146     static void      pollResultsAction(char *outFn, char *errFn);
147     static void      rxmGetNextResultsBlock(int fd);
148     static int       us_sleep(unsigned int nusecs);
149     extern "C" void  Avo_PollResultsAction_Sigusr1Handler(int foo);
150 #endif

152 /*
153  * DONE.
154  *
155  * doname_check(target, do_get, implicit, automatic)
156  *
157  * Will call doname() and then inspect the return value
158  *
159  * Return value:
160  *             Indication if the build failed or not
161  *
162  * Parameters:
163  *     target      The target to build
164  *     do_get      Passed thru to doname()
165  *     implicit    Passed thru to doname()
166  *     automatic   Are we building a hidden dependency?
167  *
168  * Global variables used:
169  *     build_failed_seen      Set if -k is on and error occurs
170  *     continue_after_error   Indicates that -k is on
171  *     report_dependencies    No error msg if -P is on
172  */
173 Doname
174 doname_check(register Name target, register Boolean do_get, register Boolean imp
175 {
176     int first_time = 1;
177     (void) fflush(stdout);
178 try_again:
179     switch (doname(target, do_get, implicit, automatic)) {
180     case build_ok:
181         second_pass = 0;
182         return build_ok;
183     case build_running:
184         second_pass = 0;
185         return build_running;
186     case build_failed:
187         if (!continue_after_error) {
188             fatal(catgets(catd, 1, 13, "Target '%s' not remade becau

```

```

189         target->string_mb);
190     }
191     build_failed_seen = true;
192     second_pass = 0;
193     return build_failed;
194 case build_dont_know:
195     /*
196     * If we can't figure out how to build an automatic
197     * (hidden) dependency, we just ignore it.
198     * We later declare the target to be out of date just in
199     * case something changed.
200     * Also, don't complain if just reporting the dependencies
201     * and not building anything.
202     */
203     if (automatic || (report_dependencies_level > 0)) {
204         second_pass = 0;
205         return build_dont_know;
206     }
207     if (first_time) {
208         first_time = 0;
209         second_pass = 1;
210         goto try_again;
211     }
212     second_pass = 0;
213     if (continue_after_error && !svr4) {
214         warning(catgets(catd, 1, 14, "Don't know how to make tar
215             target->string_mb);
216         build_failed_seen = true;
217         return build_failed;
218     }
219     fatal(catgets(catd, 1, 15, "Don't know how to make target '%s'")
220     break;
221 }
222 #ifdef lint
223     return build_failed;
224 #endif
225 }

```

unchanged_portion_omitted

```
*****
```

```
4322 Wed May 20 11:09:49 2015
```

```
new/usr/src/cmd/make/bin/make/common/dosys.cc
```

```
make: remove SCCS ident stuff
```

```
*****
```

```
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)dosys.cc 1.45 06/12/12
27 */
28
29 #pragma ident      "@(#)dosys.cc  1.45  06/12/12"
30
31
32 /*
33  * Included files
34 */
35 #include <fcntl.h>          /* open() */
36 #include <mk/defs.h>
37 #include <mksh/dosys.h>    /* doshell(), doexec() */
38 #include <mksh/misc.h>    /* getmem() */
39 #include <sys/stat.h>      /* open() */
40 #include <unistd.h>        /* getpid() */
41
42 /*
43  * Defined macros
44 */
45
46 /*
47  * typedefs & structs
48 */
49
50 /*
51  * Static variables
52 */
53 static int      filter_file;
54 static char     *filter_file_name;
55
56 /*
```

```
57  * File table of contents
58  */
59 static void     redirect_stderr(void);
60
61 /*
62  * dosys(command, ignore_error, call_make, silent_error, target)
63  *
64  * Check if command string contains meta chars and dispatch to
65  * the proper routine for executing one command line.
66  *
67  * Return value:
68  *
69  * Indicates if the command execution failed
70  *
71  * Parameters:
72  * command      The command to run
73  * ignore_error Should make abort when an error is seen?
74  * call_make    Did command reference $(MAKE) ?
75  * silent_error Should error messages be suppressed for pmake?
76  * target       Target we are building
77  *
78  * Global variables used:
79  * do_not_exec_rule Is -n on?
80  * working_on_targets We started processing real targets
81 */
82 Doname
83 dosys(register Name command, register Boolean ignore_error, register Boolean cal
84 {
85     timestruc_t      before;
86     register int     length = command->hash.length;
87     Wstring          wcb(command);
88     register wchar_t *p = wcb.get_string();
89     register wchar_t *q;
90     Doname           result;
91
92     /* Strip spaces from head of command string */
93     while (iswspace(*p)) {
94         p++, length--;
95     }
96     if (*p == (int) nul_char) {
97         return build_failed;
98     }
99     /* If we are faking it we just return */
100    if (do_not_exec_rule &&
101        working_on_targets &&
102        !call_make &&
103        !always_exec) {
104        return build_ok;
105    }
106    /* no_action_was_taken is used to print special message */
107    no_action_was_taken = false;
108
109    /* Copy string to make it OK to write it. */
110    q = ALLOC_WC(length + 1);
111    (void) wscpy(q, p);
112    /* Write the state file iff this command uses make. */
113    if (call_make && command_changed) {
114        write_state_file(0, false);
115    }
116    make_state->stat.time = file_no_time;
117    (void)exists(make_state);
118    before = make_state->stat.time;
119    /*
120     * Run command directly if it contains no shell meta chars,
121     * else run it using the shell.
122     */
123    if (await(ignore_error,
```

```
123         silent_error,
124         target,
125         wcb.get_string(),
126         command->meta ?
127         doshell(q, ignore_error, redirect_out_err,
128                 stdout_file, stderr_file, 0) :
129         doexec(q, ignore_error, redirect_out_err,
130               stdout_file, stderr_file,
131               vroot_path, 0),
132         send_mtool_msgs,
133 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
134         get_xdrs_ptr(),
135         get_job_msg_id()
136 #else
137         NULL,
138         -1
139 #endif
140         )) {
141     result = build_ok;
142 } else {
143     result = build_failed;
144 }
145 retmem(q);

147 if ((report_dependencies_level == 0) &&
148     call_make) {
149     make_state->stat.time = file_no_time;
150     (void)exists(make_state);
151     if (before == make_state->stat.time) {
152         return result;
153     }
154     makefile_type = reading_statefile;
155     if (read_trace_level > 1) {
156         trace_reader = true;
157     }
158     temp_file_number++;
159     (void) read_simple_file(make_state,
160                             false,
161                             false,
162                             false,
163                             false,
164                             false,
165                             true);
166     trace_reader = false;
167 }
168 return result;
169 }
```

unchanged_portion_omitted

```

*****
18838 Wed May 20 11:09:50 2015
new/usr/src/cmd/make/bin/make/common/files.cc
make: remove SCCS ident stuff
*****
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)files.cc 1.37 06/12/12
27 */
28
29 #pragma ident      "@(#)files.cc  1.37  06/12/12"
30
31
32 /*
33  *
34  * files.c
35  *
36  * Various file related routines:
37  *   Figure out if file exists
38  *   Wildcard resolution for directory reader
39  *   Directory reader
40  *
41  *
42  * Included files
43  *
44  * #if defined(SUN5_0) || defined(HP_UX)
45  * #include <dirent.h>          /* opendir() */
46  * #else
47  * #include <sys/dir.h>        /* opendir() */
48  * #endif
49  * #include <errno.h>          /* errno */
50  * #include <mk/defs.h>
51  * #include <mksh/macro.h>     /* getvar() */
52  * #include <mksh/misc.h>     /* get_prop(), append_prop() */
53  * #include <sys/stat.h>      /* lstat() */
54  *
55  * Defined macros
56  */
57
58 /*
59  * typedefs & structs
60  */

```

```

58 /*
59  * Static variables
60 */
61
62 /*
63  * File table of contents
64 */
65 extern timestruc_t& exists(register Name target);
66 extern void set_target_stat(register Name target, struct stat buf);
67 static timestruc_t& vpath_exists(register Name target);
68 static Name enter_file_name(wchar_t *name_string, wchar_t *library);
69 static Boolean star_match(register char *string, register char *pattern);
70 static Boolean amatch(register wchar_t *string, register wchar_t *patte
71
72 /*
73  * exists(target)
74  *
75  * Figure out the timestamp for one target.
76  *
77  * Return value:
78  *
79  * The time the target was created
80  *
81  * Parameters:
82  *
83  * target The target to check
84  *
85  * Global variables used:
86  *
87  * debug_level Should we trace the stat call?
88  * recursion_level Used for tracing
89  * vpath_defined Was the variable VPATH defined in environment?
90  */
91 timestruc_t& exists(register Name target)
92 {
93     struct stat buf;
94     register int result;
95
96     /* We cache stat information. */
97     if (target->stat.time != file_no_time) {
98         return target->stat.time;
99     }
100
101     /*
102      * If the target is a member, we have to extract the time
103      * from the archive.
104      */
105     if (target->is_member &&
106         (get_prop(target->prop, member_prop) != NULL)) {
107         return read_archive(target);
108     }
109
110     if (debug_level > 1) {
111         (void) printf(NOCATGETS("%*sstat(%s)\n",
112             recursion_level,
113             "",
114             target->string_mb);
115     }
116
117     result = lstat_vroot(target->string_mb, &buf, NULL, VROOT_DEFAULT);
118     if ((result != -1) && ((buf.st_mode & S_IFMT) == S_IFLNK)) {
119         /*
120          * If the file is a symbolic link, we remember that
121          * and then we get the status for the refd file.
122          */
123         target->stat.is_sym_link = true;
124         result = stat_vroot(target->string_mb, &buf, NULL, VROOT_DEFAULT);

```

```
123     } else {
124         target->stat.is_sym_link = false;
125     }
126
127     if (result < 0) {
128         target->stat.time = file_doesnt_exist;
129         target->stat.stat_errno = errno;
130         if ((errno == ENOENT) &&
131             vpath_defined &&
132             /* azv, fixing bug 1262942, VPATH works with a leaf name
133              * but not a directory name.
134              */
135             (target->string_mb[0] != (int) slash_char) ) {
136             /* BID_1214655 */
137             /* azv */
138                 vpath_exists(target);
139                 // return vpath_exists(target);
140         }
141     } else {
142         /* Save all the information we need about the file */
143         target->stat.stat_errno = 0;
144         target->stat.is_file = true;
145         target->stat.mode = buf.st_mode & 0777;
146         target->stat.size = buf.st_size;
147         target->stat.is_dir =
148             BOOLEAN((buf.st_mode & S_IFMT) == S_IFDIR);
149         if (target->stat.is_dir) {
150             target->stat.time = file_is_dir;
151         } else {
152             /* target->stat.time = buf.st_mtime; */
153             /* BID_1129806 */
154             /* vis@nbsp.nsk.su */
155             #if defined(linux)
156                 timestruc_t ttime = { buf.st_mtime, 0 };
157                 target->stat.time = MAX(ttime, file_min_time);
158             #else
159                 target->stat.time = MAX(buf.st_mtim, file_min_time);
160             #endif
161         }
162     }
163     if ((target->colon_splits > 0) &&
164         (get_prop(target->prop, time_prop) == NULL)) {
165         append_prop(target, time_prop)->body.time.time =
166             target->stat.time;
167     }
168     return target->stat.time;
169 }
```

unchanged_portion_omitted

new/usr/src/cmd/make/bin/make/common/globals.cc

1

```
*****
5380 Wed May 20 11:09:50 2015
new/usr/src/cmd/make/bin/make/common/globals.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)globals.cc 1.42 06/12/12
27 */
29 #pragma ident "@(#)globals.cc 1.42 06/12/12"

26 /*
27  *      globals.cc
28  *
29  *      This declares all global variables
30  */

32 /*
33  * Included files
34  */
35 #include <nl_types.h>
36 #include <mk/defs.h>
37 #include <sys/stat.h>

39 /*
40  * Defined macros
41  */

43 /*
44  * typedefs & structs
45  */

47 /*
48  * Global variables used by make only
49  */
50      FILE          *dependency_report_file;

52 /*
53  * Global variables used by make
54  */
55      Boolean      allrules_read=false;
56      Name         posix_name;
```

new/usr/src/cmd/make/bin/make/common/globals.cc

2

```
57      Name         svr4_name;
58      Boolean      sdot_target; /* used to identify s.m(/M)akefile */
59      Boolean      all_parallel; /* TEAMWARE_MAKE_CMN */
60      Boolean      assign_done;
61      int          foo;
62      Boolean      build_failed_seen;
63 #ifdef DISTRIBUTED
64      Boolean      building_serial;
65 #endif
66      Name         built_last_make_run;
67      Name         c_at;
68 #ifdef DISTRIBUTED
69      Boolean      called_make = false;
70 #endif
71      Boolean      cleanup;
72      Boolean      close_report;
73      Boolean      command_changed;
74      Boolean      commands_done;
75      Chain        conditional_targets;
76      Name         conditionals;
77      Boolean      continue_after_error; /* '-k' */
78      Property     current_line;
79      Name         current_make_version;
80      Name         current_target;
81      short        debug_level;
82      Cmd_line     default_rule;
83      Name         default_rule_name;
84      Name         default_target_to_build;
85      Name         dmake_group;
86      Name         dmake_max_jobs;
87      Name         dmake_mode;
88      DMake_mode   dmake_mode_type;
89      Name         dmake_output_mode;
90      DMake_output_mode output_mode = txt1_mode;
91      Name         dmake_odir;
92      Name         dmake_rcfile;
93      Name         done;
94      Name         dot;
95      Name         dot_keep_state;
96      Name         dot_keep_state_file;
97      Name         empty_name;
98 #if defined(HP_UX) || defined(linux)
99      int          exit_status;
100 #endif
101      Boolean      fatal_in_progress;
102      int          file_number;
103 #if 0
104      Boolean      filter_stderr; /* '-X' */
105 #endif
106      Name         force;
107      Name         ignore_name;
108      Boolean      ignore_errors; /* '-i' */
109      Boolean      ignore_errors_all; /* '-i' */
110      Name         init;
111      int          job_msg_id;
112      Boolean      keep_state;
113      Name         make_state;
114 #ifdef TEAMWARE_MAKE_CMN
115      timestruc_t  make_state_before;
116 #endif
117      Dependency   makefiles_used;
118      Name         makeflags;
119 //      Boolean     make_state_locked; // Moved to lib/mksh
120      Name         make_version;
121      char         mbs_buffer2[(MAXPATHLEN * MB_LEN_MAX)];
122      char         *mbs_ptr;
```

```

123     char          *mbs_ptr2;
124     int           mtool_msgs_fd;
125     Boolean      depinfo_already_read = false;
126 #ifdef NSE
127     Name         derived_src;
128     Boolean      nse; /* NSE on */
129     Name         nse_backquote_seen;
130     char         nse_depinfo_lockfile[MAXPATHLEN];
131     Boolean      nse_depinfo_locked;
132     Boolean      nse_did_recursion;
133     Name         nse_shell_var_used;
134     Boolean      nse_watch_vars = false;
135     wchar_t      current_makefile[MAXPATHLEN];
136 #endif
137     Boolean      no_action_was_taken = true; /* true if we've not **
138                                           ** run any command */

140     Boolean      no_parallel = false; /* TEAMWARE_MAKE_CMN */
141 #ifdef SGE_SUPPORT
142     Boolean      grid = false; /* TEAMWARE_MAKE_CMN */
143 #endif
144     Name         no_parallel_name;
145     Name         not_auto;
146     Boolean      only_parallel; /* TEAMWARE_MAKE_CMN */
147     Boolean      parallel; /* TEAMWARE_MAKE_CMN */
148     Name         parallel_name;
149     Name         localhost_name;
150     int          parallel_process_cnt;
151     Percent      percent_list;
152     Dyntarget    dyntarget_list;
153     Name         plus;
154     Name         pmake_machinesfile;
155     Name         precious;
156     Name         primary_makefile;
157     Boolean      quest; /* '-q' */
158     short        read_trace_level;
159     Boolean      reading_dependencies = false;
160     Name         recursive_name;
161     int          recursion_level;
162     short        report_dependencies_level = 0; /* -P */
163     Boolean      report_pwd;
164     Boolean      rewrite_statefile;
165     Running      running_list;
166     char         *sccs_dir_path;
167     Name         sccs_get_name;
168     Name         sccs_get_posix_name;
169     Cmd_line     sccs_get_rule;
170     Cmd_line     sccs_get_org_rule;
171     Cmd_line     sccs_get_posix_rule;
172     Name         get_name;
173     Cmd_line     get_rule;
174     Name         get_posix_name;
175     Cmd_line     get_posix_rule;
176     Boolean      send_mtool_msgs; /* '-K' */
177     Boolean      all_precious;
178     Boolean      silent_all; /* '-s' */
179     Boolean      report_cwd; /* '-w' */
180     Boolean      silent; /* '-s' */
181     Name         silent_name;
182     char         *stderr_file = NULL;
183     char         *stdout_file = NULL;
184 #ifdef SGE_SUPPORT
185     char         script_file[MAXPATHLEN] = "";
186 #endif
187     Boolean      stdout_stderr_same;
188     Dependency   suffixes;

```

```

189     Name         suffixes_name;
190     Name         sunpro_dependencies;
191     Boolean      target_variants;
192     char         *tmpdir = NOCATGETS("/tmp");
193     char         *temp_file_directory = NOCATGETS(".");
194     Name         temp_file_name;
195     short        temp_file_number;
196     time_t       timing_start;
197     wchar_t      *top_level_target;
198     Boolean      touch; /* '-t' */
199     Boolean      trace_reader; /* '-D' */
200     Boolean      build_unconditional; /* '-u' */
201     pathpt       vroot_path = VROOT_DEFAULT;
202     Name         wait_name;
203     wchar_t      wcs_buffer2[MAXPATHLEN];
204     wchar_t      *wcs_ptr;
205     wchar_t      *wcs_ptr2;
206     nl_catd      catd;
207     long int     hostid;

209 /*
210 * File table of contents
211 */

```


new/usr/src/cmd/make/bin/make/common/implicit.cc

1

```
*****
43577 Wed May 20 11:09:51 2015
new/usr/src/cmd/make/bin/make/common/implicit.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)implicit.cc 1.64 06/12/12
27 */
28
29 #pragma ident      "@(#)implicit.cc      1.64      06/12/12"
30
31
32 /*
33  * Included files
34  */
35 #include <mk/defs.h>
36 #include <mksh/macro.h>      /* expand_value() */
37 #include <mksh/misc.h>      /* retmem() */
38
39 /*
40  * Defined macros
41  */
42
43 /*
44  * typedefs & structs
45  */
46
47 /*
48  * Static variables
49  */
50 static wchar_t      WIDE_NULL[1] = {(wchar_t) nul_char};
51
52 /*
53  * File table of contents
54  */
55 extern Doname      find_suffix_rule(Name target, Name target_body, Name tar
56 extern Doname      find_ar_suffix_rule(register Name target, Name true_targ
```

new/usr/src/cmd/make/bin/make/common/implicit.cc

2

```
57 extern Doname      find_double_suffix_rule(register Name target, Property *
58 extern void        build_suffix_list(register Name target_suffix);
59 extern Doname      find_percent_rule(register Name target, Property *comman
60 static void        create_target_group_and_dependencies_list(Name target, P
61 static Boolean     match_found_with_pattern(Name target, Percent pat_rule,
62 static void        construct_string_from_pattern(Percent pat_rule, String p
63 static Boolean     dependency_exists(Name target, Property line);
64 extern Property    maybe_append_prop(Name, Property_id);
65 extern void        add_target_to_chain(Name target, Chain * query);
66
67 /*
68  *      find_suffix_rule(target, target_body, target_suffix, command, rechecking
69  *
70  * Does the lookup for single and double suffix rules.
71  * It calls build_suffix_list() to build the list of possible suffixes
72  * for the given target.
73  * It then scans the list to find the first possible source file that
74  * exists. This is done by concatenating the body of the target name
75  * (target name less target suffix) and the source suffix and checking
76  * if the resulting file exists.
77  *
78  * Return value:
79  *
80  *      Indicates if search failed or not
81  *
82  * Parameters:
83  *      target          The target we need a rule for
84  *      target_body     The target name without the suffix
85  *      target_suffix   The suffix of the target
86  *      command        Pointer to slot to deposit cmd in if found
87  *      rechecking      true if we are rechecking target which depends
88  *                     on conditional macro and keep_state is set
89  *
90  * Global variables used:
91  *      debug_level     Indicates how much tracing to do
92  *      recursion_level Used for tracing
93  */
94 extern int printf (const char *, ...);
95
96 static Boolean actual_doname = false;
97
98 /* /tolik/
99  * fix bug 1247448: Suffix Rules failed when combine with Pattern Matching Rules
100 * When make attempts to apply % rule it didn't look for a single suffix rule bec
101 * if "doname" is called from "find_percent_rule" argument "implicit" is set to
102 * and find_suffix_rule was not called. I've commented the checking of "implicit
103 * in "doname" and make got infinite recursion for SVR4 tilde rules.
104 * Usage of "we_are_in_tilde" is intended to avoid this recursion.
105 */
106
107 static Boolean we_are_in_tilde = false;
108
109 Doname
110 find_suffix_rule(Name target, Name target_body, Name target_suffix, Property *co
111 {
112     static wchar_t      static_string_buf_3M [ 3 * MAXPATHLEN ];
113     Name                 true_target = target;
114     wchar_t              *sourcename = (wchar_t*)static_string_buf_3M;
115     register wchar_t     *put_suffix;
116     register Property    source_suffix;
117     register Name        source;
118     Doname               result;
119     register Property    line;
120     extern Boolean       tilde_rule;
121     Boolean               name_found = true;
122     Boolean               posix_tilde_attempt = true;
```

```

123     int                src_len = MAXPATHLEN + strlen(target_body->strin
125     /*
126     * To avoid infinite recursion
127     */
128     if(we_are_in_tilde) {
129         we_are_in_tilde = false;
130         return(build_dont_know);
131     }
133     /*
134     * If the target is a constructed one for a ":" target,
135     * we need to consider that.
136     */
137     if (target->has_target_prop) {
138         true_target = get_prop(target->prop,
139                               target_prop->body.target.target;
140     }
141     if (debug_level > 1) {
142         (void) printf(NOCATGETS("%*sfind_suffix_rule(%s,%s,%s)\n"),
143                     recursion_level,
144                     "",
145                     true_target->string_mb,
146                     target_body->string_mb,
147                     target_suffix->string_mb);
148     }
149     if (command != NULL) {
150         if ((true_target->suffix_scan_done == true) && (*command == NULL
151             return build_ok;
152         }
153     }
154     true_target->suffix_scan_done = true;
155     /*
156     * Enter all names from the directory where the target lives as
157     * files that makes sense.
158     * This will make finding the synthesized source possible.
159     */
160     read_directory_of_file(target_body);
161     /* Cache the suffixes for this target suffix if not done. */
162     if (!target_suffix->has_read_suffixes) {
163         build_suffix_list(target_suffix);
164     }
165     /* Preload the sourcename vector with the head of the target name. */
166     if (src_len >= sizeof(static_string_buf_3M)) {
167         sourcename = ALLOC_WC(src_len);
168     }
169     (void) mbstowcs(sourcename,
170                   target_body->string_mb,
171                   (int) target_body->hash.length);
172     put_suffix = sourcename + target_body->hash.length;
173     /* Scan the suffix list for the target if one exists. */
174     if (target_suffix->has_suffixes) {
175 posix_attempts:
176         for (source_suffix = get_prop(target_suffix->prop,
177                                     suffix_prop);
178            source_suffix != NULL;
179            source_suffix = get_prop(source_suffix->next,
180                                  suffix_prop)) {
181             /* Build the synthesized source name. */
182             (void) mbstowcs(put_suffix,
183                           source_suffix->body.
184                           suffix.suffix->string_mb,
185                           (int) source_suffix->body.
186                           suffix.suffix->hash.length);
187             put_suffix[source_suffix->body.
188                       suffix.suffix->hash.length] =

```

```

189         (int) nul_char;
190         if (debug_level > 1) {
191             WCSTOMBS(mbs_buffer, sourcename);
192             (void) printf(catgets(catd, 1, 218, "%*sTrying %
193                 recursion_level,
194                 "",
195                 mbs_buffer);
196         }
197         source = getname_fn(sourcename, FIND_LENGTH, false, &nam
198         /*
199         * If the source file is not registered as
200         * a file, this source suffix did not match.
201         */
202         if (vpath_defined && !posix && !svr4) {
203             (void) exists(source);
204         }
205         if (!source->stat.is_file) {
206             if (!(posix|svr4))
207             {
208                 if (!name_found) {
209                     free_name(source);
210                 }
211                 continue;
212             }
214         /* following code will ensure that the corresponding
215         ** tilde rules are executed when corresponding s. fil
216         ** exists in the current directory. Though the curren
217         ** target ends with a ~ character, there wont be any
218         ** any file in the current directory with that suffix
219         ** as it's fictitious. Even if it exists, it'll
220         ** execute all the rules for the ~ target.
221         */
223         if (source->string_mb[source->hash.length - 1] == '~'
224             ( svr4 || posix_tilde_attempt ) )
225         {
226             char *p, *np;
227             char *tmpbuf;
229             tmpbuf = getmem(source->hash.length + 8);
230             /* + 8 to add "s." or "SCCS/s." */
231             memset(tmpbuf, 0, source->hash.length + 8);
232             source->string_mb[source->hash.length - 1] = '\0
233             if (p = (char *) memchr((char *)source->string_mb
234                 {
235                 while(1) {
236                     if (np = (char *) memchr((char *)p+1, '/', sour
237                         p = np;
238                     } else {break;}
239                 }
240             /* copy everything including '/' */
241             strncpy(tmpbuf, source->string_mb, p - source-
242             strcat(tmpbuf, NOCATGETS("s."));
243             strcat(tmpbuf, p+1);
244             retmem((wchar_t *) source->string_mb);
245             source->string_mb = tmpbuf;
246         } else {
247             strcpy(tmpbuf, NOCATGETS("s."));
248             strcat(tmpbuf, source->string_mb);
249             retmem((wchar_t *) source->string_mb);
250             source->string_mb = tmpbuf;
251         }
252     }
253     source->hash.length = strlen(source->string_mb);
254

```



```

387      /*
388      * Determine if this new dependency made the
389      * target out of date.
390      */
391      (*command)->body.line.dependency_time =
392      MAX((*command)->body.line.dependency_time,
393      source->stat.time);
394      Boolean out_of_date;
395      if (target->is_member) {
396          out_of_date = (Boolean) OUT_OF_DATE_SEC(target->
397          (*command)
398      } else {
399          out_of_date = (Boolean) OUT_OF_DATE(target->stat
400          (*command)->
401      }
402      if (build_unconditional || out_of_date) {
403          if (!rechecking) {
404              line->body.line.is_out_of_date = true;
405          }
406          if (debug_level > 0) {
407              (void) printf(catgets(catd, 1, 221, "%*s
408              recursion_level,
409              ",
410              true_target->string_mb,
411              source_suffix->body.suffix
412              target_suffix->string_mb,
413              source->string_mb);
414          }
415      }
416      /*
417      * Add the implicit rule as the target's explicit
418      * rule if none actually given, and register
419      * dependency.
420      * The time checking above really should be
421      * conditional on actual use of implicit rule
422      * as well.
423      */
424      line->body.line.sccs_command = false;
425      if (line->body.line.command_template == NULL) {
426          line->body.line.command_template =
427          source_suffix->body.suffix.command_template;
428      }
429      enter_dependency(line, source, false);
430      line->body.line.target = true_target;
431      /*
432      * Also make sure the rule is built with
433      * $* and $< bound properly.
434      */
435      line->body.line.star = target_body;
436      if (svr4|posix) {
437          char * p;
438          char tstr[256];
439          extern Boolean dollarless_flag;
440          extern Name dollarless_value;
441
442          if (tilde_rule) {
443              MBSTOWCS(wcs_buffer, NOCATGETS(source->string_mb))
444              dollarless_value = GETNAME(wcs_buffer, FIND_LENGTH)
445          }
446          else {
447              dollarless_flag = false;
448          }
449      }
450      line->body.line.less = source;
451      line->body.line.percent = NULL;
452      add_target_to_chain(source, &(line->body.line.query));

```

```

453          if (sourcename != static_string_buf_3M) {
454              retmem(sourcename);
455          }
456          return build_ok;
457      }
458      if (posix && posix_tilde_attempt) {
459          posix_tilde_attempt = false;
460          goto posix_attempts;
461      }
462      if ((command != NULL) &&
463          ((*command) != NULL) &&
464          ((*command)->body.line.star == NULL)) {
465          (*command)->body.line.star = target_body;
466      }
467      }
468      if (sourcename != static_string_buf_3M) {
469          retmem(sourcename);
470      }
471      /* Return here in case no rule matched the target */
472      return build_dont_know;
473 }

```

_____unchanged portion omitted_____

```

*****
5031 Wed May 20 11:09:51 2015
new/usr/src/cmd/make/bin/make/common/macro.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)macro.cc 1.28 06/12/12
27 */
28
29 #pragma ident "@(#)macro.cc 1.28 06/12/12"
30
31
32 /*
33  * Included files
34  */
35 #ifdef DISTRIBUTED
36 #include <avo/strings.h> /* AVO_STRDUP() */
37 #include <dm/Avo_DoJobMsg.h>
38 #endif
39 #include <mk/defs.h>
40 #include <mksh/macro.h> /* getvar(), expand_value() */
41 #include <mksh/misc.h> /* getmem() */
42
43 /*
44  * Defined macros
45  */
46
47 /*
48  * typedefs & structs
49  */
50
51 /*
52  * Static variables
53  */
54
55 /*
56  * File table of contents

```

```

57 */
58
59 void
60 setvar_append(register Name name, register Name value)
61 {
62     register Property macro_apx = get_prop(name->prop, macro_append_pr
63     register Property macro = get_prop(name->prop, macro_prop);
64     int length;
65     String_rec destination;
66     wchar_t buffer[STRING_BUFFER_LENGTH];
67     register Chain chain;
68     Name val = NULL;
69
70     if(macro_apx == NULL) {
71         macro_apx = append_prop(name, macro_append_prop);
72         if(macro != NULL) {
73             macro_apx->body.macro_appendix.value = macro->body.macro
74         }
75     }
76
77     val = macro_apx->body.macro_appendix.value_to_append;
78
79     INIT_STRING_FROM_STACK(destination, buffer);
80     buffer[0] = 0;
81     if (val != NULL) {
82         APPEND_NAME(val,
83                     &destination,
84                     (int) val->hash.length);
85         if (value != NULL) {
86             MBTOWC(wcs_buffer, " ");
87             append_char(wcs_buffer[0], &destination);
88         }
89     }
90     if (value != NULL) {
91         APPEND_NAME(value,
92                     &destination,
93                     (int) value->hash.length);
94     }
95     value = GETNAME(destination.buffer.start, FIND_LENGTH);
96     if (destination.free_after_use) {
97         retmem(destination.buffer.start);
98     }
99     macro_apx->body.macro_appendix.value_to_append = value;
100
101     SETVAR(name, empty_name, true);
102 }

```

unchanged portion omitted

new/usr/src/cmd/make/bin/make/common/main.cc

1

```
*****
103143 Wed May 20 11:09:52 2015
new/usr/src/cmd/make/bin/make/common/main.cc
make: remove SCCS ident stuff
*****
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 */
25 /*
26 * @(#)main.cc 1.158 06/12/12
27 */
28
29 #pragma ident      "@(#)main.cc      1.158  06/12/12"
30
31
32 /*
33  * Included files
34  */
35 #if defined(TEAMWARE_MAKE_CMN)
36 #   include <avo/intl.h>
37 #   include <avo/libcli.h>          /* libcli_init() */
38 #   include <avo/cli_license.h>    /* avo_cli_get_license() */
39 #   include <avo/find_dir.h>       /* avo_find_run_dir() */
40 #   include <avo/version_string.h>
41 #   include <avo/util.h>           /* avo_init() */
42 #ifdef USE_DMS_CCR
43 #   include <avo/usage_tracking.h>
44 #else
45 #   include <avo/cleanup.h>
46 #endif
47 #endif
48
49 #if defined(TEAMWARE_MAKE_CMN)
50 /* This is for dmake only (not for Solaris make).
51  * Include code to check updates (dmake patches)
52  */
53 #ifdef _CHECK_UPDATE_H
54 #include <libAU.h>
55 #endif
56 #endif
```

new/usr/src/cmd/make/bin/make/common/main.cc

2

```
58 #include <bsd/bsd.h>          /* bsd_signal() */
59
60 #ifdef DISTRIBUTED
61 #   include <dm/Avo_AcknowledgeMsg.h>
62 #   include <rw/xdrstrea.h>
63 #   include <dmrc/dmrc.h> /* dmake file processing */
64 #endif
65
66 #include <locale.h>           /* setlocale() */
67 #include <mk/copyright.h>
68 #include <mk/defs.h>
69 #include <mksdmsil8n/mksdmsil8n.h> /* libmksdmsil8n_init() */
70 #include <mksh/macro.h>      /* getvar() */
71 #include <mksh/misc.h>       /* getmem(), setup_char_semantics() */
72
73 #if defined(TEAMWARE_MAKE_CMN)
74 #ifdef USE_DMS_CCR
75 #   include <pubdmsil8n/pubdmsil8n.h> /* libpubdmsil8n_init() */
76 #endif
77 #endif
78
79 #include <pwd.h>              /* getpwnam() */
80 #include <setjmp.h>
81 #include <signal.h>
82 #include <stdlib.h>
83 #include <sys/errno.h>       /* ENOENT */
84 #include <sys/stat.h>        /* fstat() */
85 #include <fcntl.h>           /* open() */
86
87 #ifdef SUN5_0
88 #   include <sys/systeminfo.h> /* sysinfo() */
89 #endif
90
91 #include <sys/types.h>        /* stat() */
92 #include <sys/wait.h>         /* wait() */
93 #include <unistd.h>           /* execv(), unlink(), access() */
94 #include <vroot/report.h>     /* report_dependency(), get_report_file() */
95
96 // From read2.cc
97 extern Name                  normalize_name(register wchar_t *name_string, register i
98
99 // From parallel.cc
100 #if defined(TEAMWARE_MAKE_CMN)
101 #define MAXJOBS_ADJUST_RFE4694000
102
103 #ifdef MAXJOBS_ADJUST_RFE4694000
104 extern void job_adjust_fini();
105 #endif /* MAXJOBS_ADJUST_RFE4694000 */
106 #endif /* TEAMWARE_MAKE_CMN */
107
108 #if defined(linux)
109 #include <ctype.h>
110 #endif
111
112 /*
113  * Defined macros
114  */
115 #define LD_SUPPORT_ENV_VAR      NOCATGETS("SGS_SUPPORT")
116 #define LD_SUPPORT_MAKE_LIB     NOCATGETS("libmakestate.so.1")
117
118 /*
119  * typedefs & structs
120  */
121
122 /*
```

```

123 * Static variables
124 */
125 static char      *argv_zero_string;
126 static Boolean   build_failed_ever_seen;
127 static Boolean   continue_after_error_ever_seen; /* '-k' */
128 static Boolean   dmake_group_specified; /* '-g' */
129 static Boolean   dmake_max_jobs_specified; /* '-j' */
130 static Boolean   dmake_mode_specified; /* '-m' */
131 static Boolean   dmake_add_mode_specified; /* '-x' */
132 static Boolean   dmake_output_mode_specified; /* '-x DMAKE_OUTPUT_MODE */
133 static Boolean   dmake_compat_mode_specified; /* '-x SUN_MAKE_COMPAT_M */
134 static Boolean   dmake_odir_specified; /* '-o' */
135 static Boolean   dmake_rcfile_specified; /* '-c' */
136 static Boolean   env_wins; /* '-e' */
137 static Boolean   ignore_default_mk; /* '-r' */
138 static Boolean   list_all_targets; /* '-T' */
139 static int      mf_argc;
140 static char     **mf_argv;
141 static Dependency_rec not_auto_depen_struct;
142 static Dependency   not_auto_depen = &not_auto_depen_struct;
143 static Boolean   pmake_cap_r_specified; /* '-R' */
144 static Boolean   pmake_machinesfile_specified; /* '-M' */
145 static Boolean   stop_after_error_ever_seen; /* '-S' */
146 static Boolean   trace_status; /* '-p' */

148 #ifdef DMAKE_STATISTICS
149 static Boolean   getname_stat = false;
150 #endif

152 #if defined(TEAMWARE_MAKE_CMN)
153     static time_t   start_time;
154     static int      g_argc;
155     static char     **g_argv;
156 #ifdef USE_DMS_CCR
157     static Avo_usage_tracking *usageTracking = NULL;
158 #else
159     static Avo_cleanup      *cleanup = NULL;
160 #endif
161 #endif

163 /*
164 * File table of contents
165 */
166 #if defined(SUN5_0) || defined(HP_UX) || defined(linux)
167     extern "C" void      cleanup_after_exit(void);
168 #else
169     extern void          cleanup_after_exit(int, ...);
170 #endif

172 #ifdef TEAMWARE_MAKE_CMN
173 extern "C" {
174     extern void          dmake_exit_callback(void);
175     extern void          dmake_message_callback(char *);
176 }

```

unchanged_portion_omitted

new/usr/src/cmd/make/bin/make/common/make.cc

1

925 Wed May 20 11:09:53 2015

new/usr/src/cmd/make/bin/make/common/make.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)make.cc 1.2 06/12/12
27 */
28
29 #pragma ident      "@(#)make.cc 1.2 06/12/12"
```



```

*****
10219 Wed May 20 11:09:54 2015
new/usr/src/cmd/make/bin/make/common/make.rules.file
make: remove SCCS ident stuff
*****
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 2003 Sun Microsystems, Inc. All rights reserved.
22 # Use is subject to license terms.
23 #
24 # @(#)make.rules.file 1.22 06/12/12
25 #

26 SUFFIXES = .o .c .c~ .cc .cc~ .y .y~ .l .l~ .s .s~ .sh .sh~ .S .S~ .i .ln \
27           .h .h~ .f .f~ .for .for~ .F .F~ .f90 .f90~ .ftn .ftn~ .mod .mod~ \
28           .sym .def .def~ .p .p~ .r .r~ .cps .cps~ .C .C~ .Y .Y~ .L .L~ \
29           .java .java~ .class

30 .SUFFIXES: $(SUFFIXES)

32 # OUTPUT_OPTION should be defined to "-o $@" when
33 # the default rules are used for non-local files.
34 OUTPUT_OPTION=

36 #      C language section.
37 CC=cc
38 CFLAGS=
39 CPPFLAGS=
40 LINT=lint
41 LINTFLAGS=
42 COMPILE.c=$(CC) $(CFLAGS) $(CPPFLAGS) -c
43 LINK.c=$(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS)
44 LINT.c=$(LINT) $(LINTFLAGS) $(CPPFLAGS)
45 .c:
46     $(LINK.c) -o $@ $< $(LDLIBS)
47 .c~:
48     $(GET) $(GFLAGS) -p $< > $*.c
49     $(CC) $(CFLAGS) $(LDFLAGS) -o $@ $*.c
50 .c.o:
51     $(COMPILE.c) $(OUTPUT_OPTION) $<
52 .c~.o:
53     $(GET) $(GFLAGS) -p $< > $*.c
54     $(CC) $(CFLAGS) -c $*.c
55 .c.i:
56     $(CC) $(CFLAGS) $(CPPFLAGS) -P $<
57 .c~.i:
58     $(GET) $(GFLAGS) -p $< > $*.c
59     $(CC) $(CFLAGS) $(CPPFLAGS) -P $*.c

```

```

60 .c.ln:
61     $(LINT.c) $(OUTPUT_OPTION) -c $<
62 .c~.ln:
63     $(GET) $(GFLAGS) -p $< > $*.c
64     $(LINT.c) $(OUTPUT_OPTION) -c $*.c
65 .c.a:
66     $(COMPILE.c) -o $% $<
67     $(AR) $(ARFLAGS) $@ $%
68     $(RM) $%
69 .c~.a:
70     $(GET) $(GFLAGS) -p $< > $*.c
71     $(COMPILE.c) -o $% $*.c
72     $(AR) $(ARFLAGS) $@ $%
73     $(RM) $%

75 #      C language section. yacc.
76 YACC=yacc
77 YFLAGS=
78 YACC.y=$(YACC) $(YFLAGS)
79 .y:
80     $(YACC.y) $<
81     $(LINK.c) -o $@ y.tab.c $(LDLIBS)
82     $(RM) y.tab.c
83 .y~:
84     $(GET) $(GFLAGS) -p $< > $*.y
85     $(YACC) $(YFLAGS) $*.y
86     $(COMPILE.c) -o $@ y.tab.c
87     $(RM) y.tab.c

89 .y.c:
90     $(YACC.y) $<
91     mv y.tab.c $@
92 .y~.c:
93     $(GET) $(GFLAGS) -p $< > $*.y
94     $(YACC) $(YFLAGS) $*.y
95     mv y.tab.c $@
96 .y.ln:
97     $(YACC.y) $<
98     $(LINT.c) -o $@ -i y.tab.c
99     $(RM) y.tab.c
100 .y~.ln:
101     $(GET) $(GFLAGS) -p $< > $*.y
102     $(YACC.y) $*.y
103     $(LINT.c) -o $@ -i y.tab.c
104     $(RM) y.tab.c
105 .y.o:
106     $(YACC.y) $<
107     $(COMPILE.c) -o $@ y.tab.c
108     $(RM) y.tab.c
109 .y~.o:
110     $(GET) $(GFLAGS) -p $< > $*.y
111     $(YACC) $(YFLAGS) $*.y
112     $(CC) $(CFLAGS) -c y.tab.c
113     rm -f y.tab.c
114     mv y.tab.o $@

116 #      C language section. lex.
117 LEX=lex
118 LFLAGS=
119 LEX.l=$(LEX) $(LFLAGS) -t
120 .l:
121     $(RM) $*.c
122     $(LEX.l) $< > $*.c
123     $(LINK.c) -o $@ $*.c -ll $(LDLIBS)
124     $(RM) $*.c
125 .l~:

```

```

126 $(GET) $(GFLAGS) -p $< > $*.l
127 $(LEX) $(LFLAGS) $*.l
128 $(CC) $(CFLAGS) -c lex.yy.c
129 rm -f lex.yy.c
130 mv lex.yy.c $@

132 .l.c :
133 $(RM) $@
134 $(LEX.l) $< > $@
135 .l~.c:
136 $(GET) $(GFLAGS) -p $< > $*.l
137 $(LEX) $(LFLAGS) $*.l
138 mv lex.yy.c $@
139 .l.ln:
140 $(RM) $*.c
141 $(LEX.l) $< > $*.c
142 $(LINT.c) -o $@ -i $*.c
143 $(RM) $*.c
144 .l~.ln:
145 $(GET) $(GFLAGS) -p $< > $*.l
146 $(RM) $*.c
147 $(LEX.l) $*.l > $*.c
148 $(LINT.c) -o $@ -i $*.c
149 $(RM) $*.c
150 .l.o:
151 $(RM) $*.c
152 $(LEX.l) $< > $*.c
153 $(COMPILE.c) -o $@ $*.c
154 $(RM) $*.c
155 .l~.o:
156 $(GET) $(GFLAGS) -p $< > $*.l
157 $(LEX) $(LFLAGS) $*.l
158 $(CC) $(CFLAGS) -c lex.yy.c
159 rm -f lex.yy.c
160 mv lex.yy.c $@

162 # C++ language section.
163 CCC=CC
164 CCFLAGS=
165 COMPILE.cc=$(CCC) $(CCFLAGS) $(CPPFLAGS) -c
166 LINK.cc=$(CCC) $(CCFLAGS) $(CPPFLAGS) $(LDFLAGS)
167 COMPILE.C=$(CCC) $(CCFLAGS) $(CPPFLAGS) -c
168 LINK.C=$(CCC) $(CCFLAGS) $(CPPFLAGS) $(LDFLAGS)
169 .cc:
170 $(LINK.cc) -o $@ $< $(LDLIBS)
171 .cc~:
172 $(GET) $(GFLAGS) -p $< > $*.cc
173 $(LINK.cc) -o $@ $*.cc $(LDLIBS)
174 .cc.o:
175 $(COMPILE.cc) $(OUTPUT_OPTION) $<
176 .cc~.o:
177 $(GET) $(GFLAGS) -p $< > $*.cc
178 $(COMPILE.cc) $(OUTPUT_OPTION) $*.cc
179 .cc.i:
180 $(CCC) $(CCFLAGS) $(CPPFLAGS) -P $<
181 .cc~.i:
182 $(GET) $(GFLAGS) -p $< > $*.cc
183 $(CCC) $(CCFLAGS) $(CPPFLAGS) -P $*.cc
184 .cc.a:
185 $(COMPILE.cc) -o $% $<
186 $(AR) $(ARFLAGS) $@ $%
187 $(RM) $%
188 .cc~.a:
189 $(GET) $(GFLAGS) -p $< > $*.cc
190 $(COMPILE.cc) -o $% $*.cc
191 $(AR) $(ARFLAGS) $@ $%

```

```

192 $(RM) $%

194 .C:
195 $(LINK.C) -o $@ $< $(LDLIBS)
196 .C~:
197 $(GET) $(GFLAGS) -p $< > $*.C
198 $(LINK.C) -o $@ $*.C $(LDLIBS)
199 .C.o:
200 $(COMPILE.C) $(OUTPUT_OPTION) $<
201 .C~.o:
202 $(GET) $(GFLAGS) -p $< > $*.C
203 $(COMPILE.C) $(OUTPUT_OPTION) $*.C
204 .C.i:
205 $(CCC) $(CCFLAGS) $(CPPFLAGS) -P $<
206 .C~.i:
207 $(GET) $(GFLAGS) -p $< > $*.C
208 $(CCC) $(CCFLAGS) $(CPPFLAGS) -P $*.C
209 .C.a:
210 $(COMPILE.C) -o $% $<
211 $(AR) $(ARFLAGS) $@ $%
212 $(RM) $%
213 .C~.a:
214 $(GET) $(GFLAGS) -p $< > $*.C
215 $(COMPILE.C) -o $% $*.C
216 $(AR) $(ARFLAGS) $@ $%
217 $(RM) $%

219 # FORTRAN section.
220 FC=f77
221 FFLAGS=
222 COMPILE.f=$(FC) $(FFLAGS) -c
223 LINK.f=$(FC) $(FFLAGS) $(LDFLAGS)
224 COMPILE.F=$(FC) $(FFLAGS) $(CPPFLAGS) -c
225 LINK.F=$(FC) $(FFLAGS) $(CPPFLAGS) $(LDFLAGS)
226 .f:
227 $(LINK.f) -o $@ $< $(LDLIBS)
228 .f~:
229 $(GET) $(GFLAGS) -p $< > $*.f
230 $(FC) $(FFLAGS) $(LDFLAGS) -o $@ $*.f
231 .f.o:
232 $(COMPILE.f) $(OUTPUT_OPTION) $<
233 .f~.o:
234 $(GET) $(GFLAGS) -p $< > $*.f
235 $(FC) $(FFLAGS) -c $*.f
236 .f.a:
237 $(COMPILE.f) -o $% $<
238 $(AR) $(ARFLAGS) $@ $%
239 $(RM) $%
240 .f~.a:
241 $(GET) $(GFLAGS) -p $< > $*.f
242 $(COMPILE.f) -o $% $*.f
243 $(AR) $(ARFLAGS) $@ $%
244 $(RM) $%
245 .for:
246 $(LINK.f) -o $@ $< $(LDLIBS)
247 .for~:
248 $(GET) $(GFLAGS) -p $< > $*.for
249 $(FC) $(FFLAGS) $(LDFLAGS) -o $@ $*.for
250 .for.o:
251 $(COMPILE.f) $(OUTPUT_OPTION) $<
252 .for~.o:
253 $(GET) $(GFLAGS) -p $< > $*.for
254 $(FC) $(FFLAGS) -c $*.for
255 .for.a:
256 $(COMPILE.f) -o $% $<
257 $(AR) $(ARFLAGS) $@ $%

```

```

258 $(RM) $$%
259 .for~.a:
260 $(GET) $(GFLAGS) -p $< > $*.for
261 $(COMPILE.f) -o $$% $*.for
262 $(AR) $(ARFLAGS) $$@ $$%
263 $(RM) $$%
264 .F:
265 $(LINK.F) -o $$@ $< $(LDLIBS)
266 .F~:
267 $(GET) $(GFLAGS) -p $< > $*.F
268 $(FC) $(FFLAGS) $(LDFFLAGS) -o $$@ $*.F
269 .F.o:
270 $(COMPILE.F) $(OUTPUT_OPTION) $<
271 .F~.o:
272 $(GET) $(GFLAGS) -p $< > $*.F
273 $(FC) $(FFLAGS) -c $*.F
274 .F.a:
275 $(COMPILE.F) -o $$% $<
276 $(AR) $(ARFLAGS) $$@ $$%
277 $(RM) $$%
278 .F~.a:
279 $(GET) $(GFLAGS) -p $< > $*.F
280 $(COMPILE.F) -o $$% $*.F
281 $(AR) $(ARFLAGS) $$@ $$%
282 $(RM) $$%

284 # FORTRAN section. ratfor.
285 RFLAGS=
286 COMPILE.r=$(FC) $(FFLAGS) $(RFLAGS) -c
287 LINK.r=$(FC) $(FFLAGS) $(RFLAGS) $(LDFFLAGS)
288 .r:
289 $(LINK.r) -o $$@ $< $(LDLIBS)
290 .r~:
291 $(GET) $(GFLAGS) -p $< > $*.r
292 $(LINK.r) -o $$@ $*.r $(LDLIBS)
293 .r.o:
294 $(COMPILE.r) $(OUTPUT_OPTION) $<
295 .r~.o:
296 $(GET) $(GFLAGS) -p $< > $*.r
297 $(COMPILE.r) $(OUTPUT_OPTION) $*.r
298 .r.a:
299 $(COMPILE.r) -o $$% $<
300 $(AR) $(ARFLAGS) $$@ $$%
301 $(RM) $$%
302 .r~.a:
303 $(GET) $(GFLAGS) -p $< > $*.r
304 $(COMPILE.r) -o $$% $*.r
305 $(AR) $(ARFLAGS) $$@ $$%
306 $(RM) $$%

308 # FORTRAN 90 section.
309 F90C=f90
310 F90FLAGS=
311 COMPILE.f90=$(F90C) $(F90FLAGS) -c
312 LINK.f90=$(F90C) $(F90FLAGS) $(LDFFLAGS)
313 COMPILE.ftn=$(F90C) $(F90FLAGS) -c
314 LINK.ftn=$(F90C) $(F90FLAGS) $(LDFFLAGS)
315 .f90:
316 $(LINK.f90) -o $$@ $< $(LDLIBS)
317 .f90~:
318 $(GET) $(GFLAGS) -p $< > $*.f90
319 $(LINK.f90) -o $$@ $*.f90 $(LDLIBS)
320 .f90.o:
321 $(COMPILE.f90) $(OUTPUT_OPTION) $<
322 .f90~.o:
323 $(GET) $(GFLAGS) -p $< > $*.f90

```

```

324 $(COMPILE.f90) $(OUTPUT_OPTION) $*.f90
325 .f90.a:
326 $(COMPILE.f90) -o $$% $<
327 $(AR) $(ARFLAGS) $$@ $$%
328 $(RM) $$%
329 .f90~.a:
330 $(GET) $(GFLAGS) -p $< > $*.f90
331 $(COMPILE.f90) -o $$% $*.f90
332 $(AR) $(ARFLAGS) $$@ $$%
333 $(RM) $$%
334 .ftn:
335 $(LINK.ftn) -o $$@ $< $(LDLIBS)
336 .ftn~:
337 $(GET) $(GFLAGS) -p $< > $*.ftn
338 $(LINK.ftn) -o $$@ $*.ftn $(LDLIBS)
339 .ftn.o:
340 $(COMPILE.ftn) $(OUTPUT_OPTION) $<
341 .ftn~.o:
342 $(GET) $(GFLAGS) -p $< > $*.ftn
343 $(COMPILE.ftn) $(OUTPUT_OPTION) $*.ftn
344 .ftn.a:
345 $(COMPILE.ftn) -o $$% $<
346 $(AR) $(ARFLAGS) $$@ $$%
347 $(RM) $$%
348 .ftn~.a:
349 $(GET) $(GFLAGS) -p $< > $*.ftn
350 $(COMPILE.ftn) -o $$% $*.ftn
351 $(AR) $(ARFLAGS) $$@ $$%
352 $(RM) $$%

354 # Modula-2 section.
355 M2C=m2c
356 M2FLAGS=
357 MODFLAGS=
358 DEFFLAGS=
359 COMPILE.def=$(M2C) $(M2FLAGS) $(DEFFLAGS)
360 COMPILE.mod=$(M2C) $(M2FLAGS) $(MODFLAGS)
361 .def.sym:
362 $(COMPILE.def) -o $$@ $<
363 .def~.sym:
364 $(GET) $(GFLAGS) -p $< > $*.def
365 $(COMPILE.def) -o $$@ $*.def
366 .mod:
367 $(COMPILE.mod) -o $$@ -e $$@ $<
368 .mod~:
369 $(GET) $(GFLAGS) -p $< > $*.mod
370 $(COMPILE.mod) -o $$@ -e $$@ $*.mod
371 .mod.o:
372 $(COMPILE.mod) -o $$@ $<
373 .mod~.o:
374 $(GET) $(GFLAGS) -p $< > $*.mod
375 $(COMPILE.mod) -o $$@ $*.mod
376 .mod.a:
377 $(COMPILE.mod) -o $$% $<
378 $(AR) $(ARFLAGS) $$@ $$%
379 $(RM) $$%
380 .mod~.a:
381 $(GET) $(GFLAGS) -p $< > $*.mod
382 $(COMPILE.mod) -o $$% $*.mod
383 $(AR) $(ARFLAGS) $$@ $$%
384 $(RM) $$%

386 # Pascal section.
387 PC=pc
388 PFLAGS=
389 COMPILE.p=$(PC) $(PFLAGS) $(CPPFLAGS) -c

```

```

390 LINK.p=$(PC) $(PFLAGS) $(CPPFLAGS) $(LDFLAGS)
391 .p:
392     $(LINK.p) -o $@ $< $(LDLIBS)
393 .p~:
394     $(GET) $(GFLAGS) -p $< > $*.p
395     $(LINK.p) -o $@ $*.p $(LDLIBS)
396 .p.o:
397     $(COMPILE.p) $(OUTPUT_OPTION) $<
398 .p~.o:
399     $(GET) $(GFLAGS) -p $< > $*.p
400     $(COMPILE.p) $(OUTPUT_OPTION) $*.p
401 .p.a:
402     $(COMPILE.p) -o $% $<
403     $(AR) $(ARFLAGS) $@ $%
404     $(RM) $%
405 .p~.a:
406     $(GET) $(GFLAGS) -p $< > $*.p
407     $(COMPILE.p) -o $% $*.p
408     $(AR) $(ARFLAGS) $@ $%
409     $(RM) $%

411 #      Assembly section.
412 AS=as
413 ASFLAGS=
414 COMPILE.s=$(AS) $(ASFLAGS)
415 COMPILE.S=$(CC) $(ASFLAGS) $(CPPFLAGS) -c
416 .s.o:
417     $(COMPILE.s) -o $@ $<
418 .s~.o:
419     $(GET) $(GFLAGS) -p $< > $*.s
420     $(COMPILE.s) -o $@ $*.s
421 .s.a:
422     $(COMPILE.s) -o $% $<
423     $(AR) $(ARFLAGS) $@ $%
424     $(RM) $%
425 .s~.a:
426     $(GET) $(GFLAGS) -p $< > $*.s
427     $(COMPILE.s) -o $% $*.s
428     $(AR) $(ARFLAGS) $@ $%
429     $(RM) $%
430 .S.o:
431     $(COMPILE.S) -o $@ $<
432 .S~.o:
433     $(GET) $(GFLAGS) -p $< > $*.S
434     $(COMPILE.S) -o $@ $*.S
435 .S.a:
436     $(COMPILE.S) -o $% $<
437     $(AR) $(ARFLAGS) $@ $%
438     $(RM) $%
439 .S~.a:
440     $(GET) $(GFLAGS) -p $< > $*.S
441     $(COMPILE.S) -o $% $*.S
442     $(AR) $(ARFLAGS) $@ $%
443     $(RM) $%

445 #      Shell section.
446 .sh:
447     $(RM) $@
448     cat $< > $@
449     chmod +x $@
450 .sh~:
451     $(GET) $(GFLAGS) -p $< > $*.sh
452     cp $*.sh $@
453     chmod a+x $@

455 #      NeWS section

```

```

456 CPS=cps
457 CPSFLAGS=
458 .cps.h:
459     $(CPS) $(CPSFLAGS) $*.cps
460 .cps~.h:
461     $(GET) $(GFLAGS) -p $< > $*.cps
462     $(CPS) $(CPSFLAGS) $*.cps

464 #      JAVA section
465 JAVAC=javac
466 JAVACFLAGS=
467 .java.class:
468     $(JAVAC) $(JAVACFLAGS) $<
469 .java~.class:
470     $(GET) $(GFLAGS) -p $< > $*.java
471     $(JAVAC) $(JAVACFLAGS) $<

473 #      Miscellaneous section.
474 LD=ld
475 LDFLAGS=
476 LDLIBS=
477 MAKE=make
478 RM=rm -f
479 AR=ar
480 ARFLAGS=rv
481 GET=get
482 GFLAGS=

484 markfile.o:      markfile
485     echo "static char _sccsid[] = \"'grep @'(#)' markfile\\\";" > markfile.c
486     cc -c markfile.c
487     $(RM) markfile.c

489 SCCSFLAGS=
490 SCCSGETFLAGS=-s
491 .SCCS_GET:
492     sccs $(SCCSFLAGS) get $(SCCSGETFLAGS) $@ -G$@

494 .SCCS_GET_POSIX:
495     sccs $(SCCSFLAGS) get $(SCCSGETFLAGS) $@

497 .GET_POSIX:
498     $(GET) $(GFLAGS) s.$@

```

new/usr/src/cmd/make/bin/make/common/misc.cc

1

```
*****
26889 Wed May 20 11:09:54 2015
new/usr/src/cmd/make/bin/make/common/misc.cc
make: remove SCCS ident stuff
*****
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 2005 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)misc.cc 1.50 06/12/12
27 */
29 #pragma ident      "@(#)misc.cc  1.34  95/10/04"

26 /*
27  *      misc.cc
28  *
29  *      This file contains various unclassified routines.  Some main groups:
30  *      getname
31  *      Memory allocation
32  *      String handling
33  *      Property handling
34  *      Error message handling
35  *      Make internal state dumping
36  *      main routine support
37  */

39 /*
40  * Included files
41  */
42 #include <errno.h>
43 #include <mk/defs.h>
44 #include <mksh/macro.h>      /* SETVAR() */
45 #include <mksh/misc.h>      /* enable_interrupt() */
46 #include <stdarg.h>         /* va_list, va_start(), va_end() */
47 #include <vroot/report.h>   /* SUNPRO_DEPENDENCIES */

49 #if defined(HP_UX) || defined(linux)
50 #include <unistd.h>
51 #endif

53 #ifdef TEAMWARE_MAKE_CMN
54 #define MAXJOBS_ADJUST_RFE4694000
56 #ifdef MAXJOBS_ADJUST_RFE4694000
```

new/usr/src/cmd/make/bin/make/common/misc.cc

2

```
57 extern void job_adjust_fini();
58 #endif /* MAXJOBS_ADJUST_RFE4694000 */
59 #endif /* TEAMWARE_MAKE_CMN */

61 #if defined(linux)
62 #include <time.h>          /* localtime() */
63 #endif

65 /*
66  * Defined macros
67  */

69 /*
70  * typedefs & structs
71  */

73 /*
74  * Static variables
75  */

77 /*
78  * File table of contents
79  */
80 static void      print_rule(register Name target);
81 static void      print_target_n_deps(register Name target);

83 /*****
84  *
85  *      getname
86  */

88 /*****
89  *
90  *      Memory allocation
91  */

93 /*
94  *      free_chain()
95  *
96  *      frees a chain of Name_vector's
97  *
98  *      Parameters:
99  *          ptr          Pointer to the first element in the chain
100 *                      to be freed.
101 *
102 *      Global variables used:
103  */
104 void
105 free_chain(Name_vector ptr)
106 {
107     if (ptr != NULL) {
108         if (ptr->next != NULL) {
109             free_chain(ptr->next);
110         }
111         free((char *) ptr);
112     }
113 }
_____unchanged_portion_omitted_____
```

new/usr/src/cmd/make/bin/make/common/nse.cc

1

```
*****
14545 Wed May 20 11:09:55 2015
new/usr/src/cmd/make/bin/make/common/nse.cc
make: remove SCCS ident stuff
*****
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 1994 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)nse.cc 1.15 06/12/12
27 */
28 #pragma ident      "@(#)nse.cc      1.15      06/12/12"
29
30 #ifdef NSE
31
32 /*
33  * Included files
34  */
35 #include <mk/defs.h>
36 #include <mksh/macro.h>          /* expand_value() */
37 #include <mksh/misc.h>          /* get_prop() */
38
39 /*
40  * This file does some extra checking on behalf of the NSE.
41  * It does some stuff that is analogous to lint in that it
42  * looks for things which may be legal but that give the NSE
43  * trouble.  Currently it looks for:
44  *
45  * 1) recursion by cd'ing to a directory specified by a
46  *    make variable that is defined from the shell environment.
47  * 2) recursion by cd'ing to a directory specified by a
48  *    make variable that has backquotes in it.
49  * 3) recursion to another makefile in the same directory.
50  * 4) a dependency upon an SCCS file (SCCS/s.*)
51  * 5) backquotes in a file name
52  * 6) a make variable being defined on the command-line that
53  *    ends up affecting dependencies
54  * 7) wildcards (*) in dependencies
55  * 8) recursion to the same directory
56  * 9) potential source files on the left-hand-side so
57  *    that they appear as derived files
58
59 * Things it should look for:
60 * 1) makefiles that are symlinks (why are these a problem?)
61 */

```

new/usr/src/cmd/make/bin/make/common/nse.cc

2

```
58 #define TARG_SUFFIX      "/usr/nse/lib/nse_targ.suffix"
59
60 typedef struct _Nse_suffix      *Nse_suffix, Nse_suffix_rec;
61 struct _Nse_suffix {
62     wchar_t                      *suffix;          /* The suffix */
63     struct _Nse_suffix          *next;            /* Linked list */
64 };
65
66 _____unchanged_portion_omitted_____

```

new/usr/src/cmd/make/bin/make/common/nse_printdep.cc

1

```
*****
12880 Wed May 20 11:09:55 2015
new/usr/src/cmd/make/bin/make/common/nse_printdep.cc
make: remove SCCS ident stuff
*****
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)nse_printdep.cc 1.18 06/12/12
27 */
28
29 #pragma ident      "@(#)nse_printdep.cc 1.18 06/12/12"
30
31
32 /*
33  * File table of contents
34  */
35 void print_dependencies(register Name target, register Property line);
36 static void print_deps(register Name target, register Property line);
37 static void print_more_deps(Name target, Name name);
38 static void print_filename(Name name);
39 static Boolean should_print_dep(Property line);
40 static void print_forest(Name target);
41 static void print_deplist(Dependency head);
42 void print_value(register Name value, Daemon daemon);
43 static void print_rule(register Name target);
44 static void print_rec_info(Name target);
45 static Boolean is_out_of_date(Property line);
46 extern void depvar_print_results(void);
47 extern int printf(const char *, ...);
48 extern int _flsbuf(unsigned int, FILE *);
49
50 /*
51  * print_dependencies(target, line)
52  *
53  * Print all the dependencies of a target. First print all the Makefiles.
54  * Then print all the dependencies. Finally, print all the .INIT
55  * dependencies.
56  */
```

new/usr/src/cmd/make/bin/make/common/nse_printdep.cc

2

```
57 * Parameters:
58 * target The target we print dependencies for
59 * line We get the dependency list from here
60 *
61 * Global variables used:
62 * done The Name ".DONE"
63 * init The Name ".INIT"
64 * makefiles_used List of all makefiles read
65 */
66 void
67 print_dependencies(register Name target, register Property line)
68 {
69     Dependency dp;
70     static Boolean makefiles_printed = false;
71
72 #ifdef SUNOS4_AND_AFTER
73     if (target_variants) {
74 #else
75     if (is_true(flag.target_variants)) {
76 #endif
77         depvar_print_results();
78     }
79
80     if (!makefiles_printed) {
81         /*
82          * Search the makefile list for the primary makefile,
83          * then print it and its inclusions. After that go back
84          * and print the default.mk file and its inclusions.
85          */
86         for (dp = makefiles_used; dp != NULL; dp = dp->next) {
87             if (dp->name == primary_makefile) {
88                 break;
89             }
90         }
91         if (dp) {
92             print_deplist(dp);
93             for (dp = makefiles_used; dp != NULL; dp = dp->next) {
94                 if (dp->name == primary_makefile) {
95                     break;
96                 }
97                 (void)printf("%s", dp->name->string_mb);
98             }
99         }
100         (void) printf("\n");
101         makefiles_printed = true;
102     }
103     print_deps(target, line);
104 #ifdef SUNOS4_AND_AFTER
105     /*
106      * print_more_deps(target, init);
107      * print_more_deps(target, done);
108      */
109     if (target_variants) {
110 #else
111     print_more_deps(target, cached_names.init);
112     print_more_deps(target, cached_names.done);
113     if (is_true(flag.target_variants)) {
114 #endif
115         print_forest(target);
116     }
117 }
_____unchanged_portion_omitted_____
```

new/usr/src/cmd/make/bin/make/common/parallel.cc

1

```
*****
61618 Wed May 20 11:09:56 2015
new/usr/src/cmd/make/bin/make/common/parallel.cc
make: remove SCCS ident stuff
*****
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 2005 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)parallel.cc 1.75 06/12/12
27 */
29 #pragma ident    "@(#)parallel.cc    1.75    06/12/12"

26 #ifdef TEAMWARE_MAKE_CMN

28 /*
29  *    parallel.cc
30  *
31  *    Deal with the parallel processing
32  */

34 /*
35  * Included files
36  */
37 #ifdef DISTRIBUTED
38 #include <avo/strings.h>        /* AVO_STRDUP() */
39 #include <dm/Avo_DoJobMsg.h>
40 #include <dm/Avo_MToolJobResultMsg.h>
41 #endif
42 #include <errno.h>             /* errno */
43 #include <fcntl.h>
44 #include <avo/util.h>          /* avo_get_user(), avo_hostname() */
45 #include <mk/defs.h>
46 #include <mksh/dosys.h>        /* redirect_io() */
47 #include <mksh/macro.h>        /* expand_value() */
48 #include <mksh/misc.h>         /* getmem() */
49 #include <sys/signal.h>
50 #include <sys/stat.h>
51 #include <sys/types.h>
52 #include <sys/utsname.h>
53 #include <sys/wait.h>
54 #include <unistd.h>

56 #ifdef SGE_SUPPORT
```

new/usr/src/cmd/make/bin/make/common/parallel.cc

2

```
57 #include <dmthread/Avo_PathNames.h>
58 #endif

61 /*
62  * Defined macros
63  */
64 #define MAXRULES                100

66 /*
67  * This const should be in avo_dms/include/AvoDmakeCommand.h
68  */
69 const int local_host_mask = 0x20;

72 /*
73  * typedefs & structs
74  */

77 /*
78  * Static variables
79  */
80 #ifdef TEAMWARE_MAKE_CMN
81 static Boolean      just_did_subtree = false;
82 static char         local_host[MAXNAMELEN] = "";
83 static char         user_name[MAXNAMELEN] = "";
84 #endif
85 static int          pmake_max_jobs = 0;
86 static pid_t        process_running = -1;
87 static Running      *running_tail = &running_list;
88 static Name         subtree_conflict;
89 static Name         subtree_conflict2;

92 /*
93  * File table of contents
94  */
95 #ifdef DISTRIBUTED
96 static void         append_dmake_cmd(Avo_DoJobMsg *dmake_job_msg, char *orig
97 static void         append_job_result_msg(Avo_MToolJobResultMsg *msg, char *
98 static void         send_job_result_msg(Running rp);
99 #endif
100 static void         delete_running_struct(Running rp);
101 static Boolean      dependency_conflict(Name target);
102 static Doname       distribute_process(char **commands, Property line);
103 static void         doname_subtree(Name target, Boolean do_get, Boolean impl
104 static void         dump_out_file(char *filename, Boolean err);
105 static void         finish_doname(Running rp);
106 static void         maybe_reread_make_state(void);
107 static void         process_next(void);
108 static void         reset_conditionals(int cnt, Name *targets, Property *loc
109 static pid_t        run_rule_commands(char *host, char **commands);
110 static Property     *set_conditionals(int cnt, Name *targets);
111 static void         store_conditionals(Running rp);

114 /*
115  *    execute_parallel(line, waitflg)
116  *
117  *    DMake 2.x:
118  *    parallel mode: spawns a parallel process to execute the command group.
119  *    distributed mode: sends the command group down the pipe to rxm.
120  *
121  *    Return value:
122  *
123  *    The result of the execution
```



```

123 *
124 *   Parameters:
125 *       line           The command group to execute
126 */
127 Doname
128 execute_parallel(Property line, Boolean waitflg, Boolean local)
129 {
130     int             argcnt;
131     int             cmd_options = 0;
132     char            *commands[MAXRULES + 5];
133     char            *cp;
134 #ifdef DISTRIBUTED
135     Avo_DoJobMsg    *dmake_job_msg = NULL;
136 #endif
137     Name            dmake_name;
138     Name            dmake_value;
139     int             ignore;
140     Name            make_machines_name;
141     char            **p;
142     Property        prop;
143     Doname          result = build_ok;
144     Cmd_line        rule;
145     Boolean         silent_flag;
146     Name            target = line->body.line.target;
147     Boolean         wrote_state_file = false;

149     if ((pmake_max_jobs == 0) &&
150         (dmake_mode_type == parallel_mode)) {
151         if (user_name[0] == '\0') {
152             avo_get_user(user_name, NULL);
153         }
154         if (local_host[0] == '\0') {
155             strcpy(local_host, avo_hostname());
156         }
157         MBSTOWCS(wcs_buffer, NOCATGETS("DMAKE_MAX_JOBS"));
158         dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
159         if (((prop = get_prop(dmake_name->prop, macro_prop)) != NULL) &&
160             ((dmake_value = prop->body.macro.value) != NULL)) {
161             pmake_max_jobs = atoi(dmake_value->string_mb);
162             if (pmake_max_jobs <= 0) {
163                 warning(catgets(catd, 1, 308, "DMAKE_MAX_JOBS ca
164 warning(catgets(catd, 1, 309, "setting DMAKE_MAX
165 pmake_max_jobs = PMAKE_DEF_MAX_JOBS;
166             }
167         } else {
168             /*
169             * For backwards compatibility w/ PMake 1.x, when
170             * DMake 2.x is being run in parallel mode, DMake
171             * should parse the PMake startup file
172             * $(HOME)/.make.machines to get the pmake_max_jobs.
173             */
174             MBSTOWCS(wcs_buffer, NOCATGETS("PMAKE_MACHINESFILE"));
175             dmake_name = GETNAME(wcs_buffer, FIND_LENGTH);
176             if (((prop = get_prop(dmake_name->prop, macro_prop)) !=
177                 ((dmake_value = prop->body.macro.value) != NULL)) {
178                 make_machines_name = dmake_value;
179             } else {
180                 make_machines_name = NULL;
181             }
182             if ((pmake_max_jobs = read_make_machines(make_machines_n
183                 pmake_max_jobs = PMAKE_DEF_MAX_JOBS;
184             }
185         }
186 #ifdef DISTRIBUTED
187         if (send_mtool_msgs) {
188             send_rsrc_info_msg(pmake_max_jobs, local_host, user_name

```

```

189     }
190 #endif
191 }

193     if ((dmake_mode_type == serial_mode) ||
194         ((dmake_mode_type == parallel_mode) && (waitflg))) {
195         return (execute_serial(line));
196     }

198 #ifdef DISTRIBUTED
199     if (dmake_mode_type == distributed_mode) {
200         if (local) {
201             // return (execute_serial(line));
202             waitflg = true;
203         }
204         dmake_job_msg = new Avo_DoJobMsg();
205         dmake_job_msg->setJobId(++job_msg_id);
206         dmake_job_msg->setTarget(target->string_mb);
207         dmake_job_msg->setImmediateOutput(0);
208         called_make = false;
209     } else
210 #endif
211     {
212         p = commands;
213     }

215     argcnt = 0;
216     for (rule = line->body.line.command_used;
217         rule != NULL;
218         rule = rule->next) {
219         if (posix && (touch || quest) && !rule->always_exec) {
220             continue;
221         }
222         if (vpath_defined) {
223             rule->command_line =
224                 vpath_translation(rule->command_line);
225         }
226         if (dmake_mode_type == distributed_mode) {
227             cmd_options = 0;
228             if (local) {
229                 cmd_options |= local_host_mask;
230             }
231         } else {
232             silent_flag = false;
233             ignore = 0;
234         }
235         if (rule->command_line->hash.length > 0) {
236             if (++argcnt == MAXRULES) {
237                 if (dmake_mode_type == distributed_mode) {
238                     /* XXX - tell rxm to execute on local ho
239                     /* I WAS HERE!!! */
240                 } else {
241                     /* Too many rules, run serially instead.
242                     return build_serial;
243                 }
244             }
245 #ifdef DISTRIBUTED
246             if (dmake_mode_type == distributed_mode) {
247                 /*
248                 * XXX - set assign_mask to tell rxm
249                 * to do the following.
250                 */
251                 /* From execute_serial():
252                 if (rule->assign) {
253                     result = build_ok;
254                     do_assign(rule->command_line, target);

```

```

255 */
256
257     if (0) {
258     } else if (report_dependencies_level == 0) {
259         if (rule->ignore_error) {
260             cmd_options |= ignore_mask;
261         }
262         if (rule->silent) {
263             cmd_options |= silent_mask;
264         }
265         if (rule->command_line->meta) {
266             cmd_options |= meta_mask;
267         }
268         if (rule->make_refd) {
269             cmd_options |= make_refd_mask;
270         }
271         if (do_not_exec_rule) {
272             cmd_options |= do_not_exec_mask;
273         }
274         append_dmake_cmd(dmake_job_msg,
275                         rule->command_line->str
276                         cmd_options);
277         /* Copying dosys().... */
278         if (rule->make_refd) {
279             if (waitflg) {
280                 dmake_job_msg->setImmedi
281             }
282             called_make = true;
283             if (command_changed &&
284                 !wrote_state_file) {
285                 write_state_file(0, fals
286                 wrote_state_file = true;
287             }
288         }
289     } else
290 #endif
291     {
292         if (rule->silent && !silent) {
293             silent_flag = true;
294         }
295         if (rule->ignore_error) {
296             ignore++;
297         }
298         /* XXX - need to add support for + prefix */
299         if (silent_flag || ignore) {
300             *p = getmem((silent_flag ? 1 : 0) +
301                       ignore +
302                       (strlen(rule->
303                           command_line->
304                           string_mb)) +
305                       1);
306             cp = *p++;
307             if (silent_flag) {
308                 *cp++ = (int) at_char;
309             }
310             if (ignore) {
311                 *cp++ = (int) hyphen_char;
312             }
313             (void) strcpy(cp, rule->command_line->st
314         } else {
315             *p++ = rule->command_line->string_mb;
316         }
317     }
318 }
319 }
320 if ((argc == 0) ||

```

```

321         (report_dependencies_level > 0)) {
322 #ifdef DISTRIBUTED
323         if (dmake_job_msg) {
324             delete dmake_job_msg;
325         }
326 #endif
327         return build_ok;
328     }
329 #ifdef DISTRIBUTED
330     if (dmake_mode_type == distributed_mode) {
331         // Send a DoJob message to the rxm process.
332         distribute_rxm(dmake_job_msg);
333
334         // Wait for an acknowledgement.
335         Avo_AcknowledgeMsg *ackMsg = getAcknowledgeMsg();
336         if (ackMsg) {
337             delete ackMsg;
338         }
339
340         if (waitflg) {
341             // Wait for, and process a job result.
342             result = await_dist(waitflg);
343             if (called_make) {
344                 maybe_reread_make_state();
345             }
346             check_state(temp_file_name);
347             if (result == build_failed) {
348                 if (!continue_after_error) {
349
350 #ifdef PRINT_EXIT_STATUS
351                 warning(NOCATGETS("I'm in execute_parall
352 #endif
353
354                 fatal(catgets(catd, 1, 252, "Command fai
355                 target->string_mb);
356             }
357             /*
358             * Make sure a failing command is not
359             * saved in .make.state.
360             */
361             line->body.line.command_used = NULL;
362         }
363         if (temp_file_name != NULL) {
364             free_name(temp_file_name);
365         }
366         temp_file_name = NULL;
367         Property spro = get_prop(sunpro_dependencies->prop, macr
368         if (spro != NULL) {
369             Name val = spro->body.macro.value;
370             if (val != NULL) {
371                 free_name(val);
372                 spro->body.macro.value = NULL;
373             }
374         }
375         spro = get_prop(sunpro_dependencies->prop, env_mem_prop)
376         if (spro) {
377             char *val = spro->body.env_mem.value;
378             if (val != NULL) {
379                 retmem_mb(val);
380                 spro->body.env_mem.value = NULL;
381             }
382         }
383         return result;
384     } else {
385         parallel_process_cnt++;
386         return build_running;

```

```
387     }
388   } else
389 #endif
390   {
391     *p = NULL;
392
393     Doname res = distribute_process(commands, line);
394     if (res == build_running) {
395       parallel_process_cnt++;
396     }
397
398     /*
399     * Return only those memory that were specially allocated
400     * for part of commands.
401     */
402     for (int i = 0; commands[i] != NULL; i++) {
403       if ((commands[i][0] == (int) at_char) ||
404           (commands[i][0] == (int) hyphen_char)) {
405         retmem_mb(commands[i]);
406       }
407     }
408     return res;
409   }
410 }
unchanged_portion_omitted
```

```

*****
11436 Wed May 20 11:09:56 2015
new/usr/src/cmd/make/bin/make/common/pmake.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)pmake.cc 1.9 06/12/12
27 */
28 #pragma ident      "@(#)pmake.cc  1.9      06/12/12"
29
30 #ifdef TEAMWARE_MAKE_CMN
31
32 /*
33  * Included files
34  */
35 #include <arpa/inet.h>
36 #include <mk/defs.h>
37 #include <mksh/misc.h>
38 #include <netdb.h>
39 #include <netinet/in.h>
40 #include <sys/socket.h>
41 #include <sys/stat.h>
42 #include <sys/types.h>
43 #include <sys/utsname.h>
44 #include <rpc/rpc.h>          /* host2netname(), netname2host() */
45 #ifdef linux
46 #include <unistd.h>          /* getdomainname() */
47 #endif
48
49 /*
50  * Defined macros
51  */
52
53 /*
54  * Typedef's & structs
55  */

```

```

57 /*
58  * File table of contents
59  */
60 static int          get_max(wchar_t **ms_address, wchar_t *hostname);
61 static Boolean      pskip_comment(wchar_t **cp_address);
62 static void         pskip_till_next_word(wchar_t **cp);
63 static Boolean      pskip_white_space(wchar_t **cp_address);
64
65
66 /*
67  *      read_make_machines(Name make_machines_name)
68  *
69  *      For backwards compatibility w/ PMake 1.x, when DMake 2.x is
70  *      being run in parallel mode, DMake should parse the PMake startup
71  *      file $(HOME)/.make.machines to get the PMake max jobs.
72  *
73  *      Return value:
74  *          int of PMake max jobs
75  *
76  *      Parameters:
77  *          make_machines_name      Name of .make.machines file
78  *
79  */
80 int
81 read_make_machines(Name make_machines_name)
82 {
83     wchar_t          c;
84     Boolean          default_make_machines;
85     struct hostent   *hp;
86     wchar_t          local_host[MAX_HOSTNAMELEN + 1];
87     char             local_host_mb[MAX_HOSTNAMELEN + 1] = "";
88     int              local_host_wslen;
89     wchar_t          full_host[MAXNETNAMELEN + 1];
90     int              full_host_wslen = 0;
91     char             *homedir;
92     Name             MAKE_MACHINES;
93     struct stat      make_machines_buf;
94     FILE             *make_machines_file;
95     wchar_t          *make_machines_list = NULL;
96     char             *make_machines_list_mb = NULL;
97     wchar_t          make_machines_path[MAXPATHLEN];
98     char             mb_make_machines_path[MAXPATHLEN];
99     wchar_t          *mp;
100    wchar_t          *ms;
101    int              pmake_max_jobs = 0;
102    struct utsname    uts_info;
103
104
105    MBSTOWCS(wcs_buffer, NOCATGETS("MAKE_MACHINES"));
106    MAKE_MACHINES = GETNAME(wcs_buffer, FIND_LENGTH);
107    /* Did the user specify a .make.machines file on the command line? */
108    default_make_machines = false;
109    if (make_machines_name == NULL) {
110        /* Try reading the default .make.machines file, in $(HOME). */
111        homedir = getenv(NOCATGETS("HOME"));
112        if ((homedir != NULL) && (strlen(homedir) < (sizeof(mb_make_mach
113            sprintf(mb_make_machines_path,
114                NOCATGETS("%s/.make.machines"), homedir);
115            MBSTOWCS(make_machines_path, mb_make_machines_path);
116            make_machines_name = GETNAME(make_machines_path, FIND_LE
117            default_make_machines = true;
118        }
119        if (make_machines_name == NULL) {
120            /*
121             * No $(HOME)/.make.machines file.
122             * Return 0 for PMake max jobs.

```

```

123         */
124         return(0);
125     }
126 }
127 /*
128 make_machines_list_mb = getenv(MAKE_MACHINES->string_mb);
129 */
130 /* Open the .make.machines file. */
131 if ((make_machines_file = fopen(make_machines_name->string_mb, "r")) ==
132     if (!default_make_machines) {
133         /* Error opening .make.machines file. */
134         fatal(catgets(catd, 1, 314, "Open of %s failed: %s"),
135             make_machines_name->string_mb,
136             errmsg(errno));
137     } else {
138         /*
139          * No $(HOME)/.make.machines file.
140          * Return 0 for PMake max jobs.
141          */
142         return(0);
143     }
144 /* Stat the .make.machines file to get the size of the file. */
145 } else if (fstat(fileno(make_machines_file), &make_machines_buf) < 0) {
146     /* Error stat'ing .make.machines file. */
147     fatal(catgets(catd, 1, 315, "Stat of %s failed: %s"),
148         make_machines_name->string_mb,
149         errmsg(errno));
150 } else {
151     /* Allocate memory for "MAKE_MACHINES=<contents of .m.m>" */
152     make_machines_list_mb =
153         (char *) getmem((int) (strlen(MAKE_MACHINES->string_mb) +
154             2 +
155             make_machines_buf.st_size));
156     sprintf(make_machines_list_mb,
157         "%s=",
158         MAKE_MACHINES->string_mb);
159     /* Read in the .make.machines file. */
160     if (fread(make_machines_list_mb + strlen(MAKE_MACHINES->string_m
161         sizeof(char),
162         (int) make_machines_buf.st_size,
163         make_machines_file) != make_machines_buf.st_size) {
164         /*
165          * Error reading .make.machines file.
166          * Return 0 for PMake max jobs.
167          */
168         warning(catgets(catd, 1, 316, "Unable to read %s"),
169             make_machines_name->string_mb);
170         (void) fclose(make_machines_file);
171         retmem_mb((caddr_t) make_machines_list_mb);
172         return(0);
173     } else {
174         (void) fclose(make_machines_file);
175         /* putenv "MAKE_MACHINES=<contents of .m.m>" */
176         *(make_machines_list_mb +
177             strlen(MAKE_MACHINES->string_mb) +
178             1 +
179             make_machines_buf.st_size) = (int) nul_char;
180         if (putenv(make_machines_list_mb) != 0) {
181             warning(catgets(catd, 1, 317, "Couldn't put cont
182                 make_machines_name->string_mb);
183         } else {
184             make_machines_list_mb += strlen(MAKE_MACHINES->s
185             make_machines_list = ALLOC_WC(strlen(make_machin
186                 (void) mbstowcs(make_machines_list,
187                     make_machines_list_mb,
188                     (strlen(make_machines_list_mb) +

```

```

189     }
190 }
191 }
192
193 uname(&uts_info);
194 strcpy(local_host_mb, &uts_info.nodename[0]);
195 MBSTOWCS(local_host, local_host_mb);
196 local_host_wslen = wslen(local_host);
197
198 // There is no getdomainname() function on Solaris.
199 // And netname2host() function does not work on Linux.
200 // So we have to use different APIs.
201 #ifdef linux
202 if (getdomainname(mbs_buffer, MAXNETNAMELEN+1) == 0) {
203     sprintf(mbs_buffer2, "%s.%s", local_host_mb, mbs_buffer);
204 #else
205 if (host2netname(mbs_buffer, NULL, NULL) &&
206     netname2host(mbs_buffer, mbs_buffer2, MAXNETNAMELEN+1)) {
207 #endif
208     MBSTOWCS(full_host, mbs_buffer2);
209     full_host_wslen = wslen(full_host);
210 }
211
212 for (ms = make_machines_list;
213      (ms) && (*ms);
214      ) {
215     /*
216      * Skip white space and comments till you reach
217      * a machine name.
218      */
219     pskip_till_next_word(&ms);
220
221     /*
222      * If we haven't reached the end of file, process the
223      * machine name.
224      */
225     if (*ms) {
226         /*
227          * If invalid machine name decrement counter
228          * and skip line.
229          */
230         mp = ms;
231         SKIPWORD(ms);
232         c = *ms;
233         *ms++ = '\0'; /* Append null to machine name. */
234         /*
235          * If this was the beginning of a comment
236          * (we overwrote a # sign) and it's not
237          * end of line yet, shift the # sign.
238          */
239         if ((c == '#' ) && (*ms != '\n') && (*ms)) {
240             *ms = '#';
241         }
242         WCSTOMBS(mbs_buffer, mp);
243         /*
244          * Print "Ignoring unknown host" if:
245          * 1) hostname is longer than MAX_HOSTNAMELEN, or
246          * 2) hostname is unknown
247          */
248         if ((wslen(mp) > MAX_HOSTNAMELEN) ||
249             ((hp = gethostbyname(mbs_buffer)) == NULL)) {
250             warning(catgets(catd, 1, 318, "Ignoring unknown
251                 mbs_buffer);
252             SKIPTOEND(ms);
253             /* Increment ptr if not end of file. */
254             if (*ms) {

```

```

255         ms++;
256     }
257 } else {
258     /* Compare current hostname with local_host. */
259     if (wslen(mp) == local_host_wslen &&
260         IS_WEQUALN(mp, local_host, local_host_wslen)
261         /*
262          * Bingo, local_host is in .make.machine
263          * Continue reading.
264          */
265         pmake_max_jobs = PMAKE_DEF_MAX_JOBS;
266     /* Compare current hostname with full_host. */
267     } else if (wslen(mp) == full_host_wslen &&
268         IS_WEQUALN(mp, full_host, full_host_w
269         /*
270          * Bingo, full_host is in .make.machines
271          * Continue reading.
272          */
273         pmake_max_jobs = PMAKE_DEF_MAX_JOBS;
274     } else {
275         if (c != '\n') {
276             SKIPTOEND(ms);
277             if (*ms) {
278                 ms++;
279             }
280         }
281         continue;
282     }
283     /* If we get here, local_host is in .make.machin
284     if (c != '\n') {
285         /* Now look for keyword 'max'. */
286         MBSTOWCS(wcs_buffer, NOCATGETS("max"));
287         SKIPSPACE(ms);
288         while ((*ms != '\n') && (*ms)) {
289             if (*ms == '#') {
290                 pskip_comment(&ms);
291             } else if (IS_WEQUALN(ms, wcs_bu
292                 /* Skip "max". */
293                 ms += 3;
294                 pmake_max_jobs = get_max
295                 SKIPSPACE(ms);
296             } else {
297                 warning(catgets(catd, 1,
298                     SKIPTOEND(ms);
299                 break;
300             }
301         }
302     }
303     break; /* out of outermost for() loop. */
304 }
305 }
306 }
307 retmem(make_machines_list);
308 return(pmake_max_jobs);
309 }

```

unchanged_portion_omitted

```

*****
58610 Wed May 20 11:09:57 2015
new/usr/src/cmd/make/bin/make/common/read.cc
make: remove SCCS ident stuff
*****
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 */
25 /*
26 * @(#)read.cc 1.64 06/12/12
27 */
28
29 #pragma ident      "@(#)read.cc  1.64  06/12/12"
30
31
32 /*
33  * Included files
34  */
35 #include <avo/avo_alloc.h>          /* alloc() */
36 #include <errno.h>                 /* errno */
37 #include <fcntl.h>                 /* fcntl() */
38 #include <mk/defs.h>
39 #include <mksh/macro.h>            /* expand_value(), expand_macro() */
40 #include <mksh/misc.h>            /* getmem() */
41 #include <mksh/read.h>            /* get_next_block_fn() */
42 #include <sys/uio.h>              /* read() */
43 #include <unistd.h>               /* read(), unlink() */
44
45 #if defined(HP_UX) || defined(linux)
46 #include <avo/types.h>
47 extern "C" Avo_err *avo_find_run_dir(char **dirp);
48 #endif
49
50 /*
51  * typedefs & structs
52  */
53
54 /*
55  * Static variables
56  */

```

```

58 static int line_started_with_space=0; // Used to diagnose spaces instead of tabs
59
60 /*
61  * File table of contents
62  */
63 static void parse_makefile(register Name true_makefile_name, register
64 static Source push_macro_value(register Source bp, register wchar_t *b
65 extern void enter_target_groups_and_dependencies(Name_vector target,
66 extern Name normalize_name(register wchar_t *name_string, register i
67
68 /*
69  * read_simple_file(makefile_name, chase_path, doname_it,
70  * complain, must_exist, report_file, lock_makefile)
71  *
72  * Make the makefile and setup to read it. Actually read it if it is stdio
73  *
74  * Return value:
75  * false if the read failed
76  *
77  * Parameters:
78  * makefile_name Name of the file to read
79  * chase_path Use the makefile path when opening file
80  * doname_it Call doname() to build the file first
81  * complain Print message if doname/open fails
82  * must_exist Generate fatal if file is missing
83  * report_file Report file when running -P
84  * lock_makefile Lock the makefile when reading
85  *
86  * Static variables used:
87  *
88  * Global variables used:
89  * do_not_exec_rule Is -n on?
90  * file_being_read Set to the name of the new file
91  * line_number The number of the current makefile line
92  * makefiles_used A list of all makefiles used, appended to
93  */
94
95 Boolean
96 read_simple_file(register Name makefile_name, register Boolean chase_path, regis
97 {
98     static short max_include_depth;
99     register Property makefile = maybe_append_prop(makefile_name,
100                                                     makefile_prop);
101     Boolean forget_after_parse = false;
102     static pathpt makefile_path;
103     register int n;
104     char *path;
105     register Source source = ALLOC(Source);
106     Property orig_makefile = makefile;
107     Dependency *dpp;
108     Dependency dp;
109     register int length;
110     wchar_t *previous_file_being_read = file_being_read;
111     int previous_line_number = line_number;
112     wchar_t previous_current_makefile[MAXPATHLEN];
113     Makefile_type save_makefile_type;
114     Name normalized_makefile_name;
115     register wchar_t *string_start;
116     register wchar_t *string_end;
117
118
119
120 #if defined(HP_UX) || defined(linux)
121     Avo_err *findrundir_err;
122     char *run_dir, makerules_dir[BUFSIZ];

```

```

123 #endif
125     wchar_t * wcb = get_wstring(makefile_name->string_mb);

127 #ifdef NSE
128     if (report_file){
129         wscopy(previous_current_makefile, current_makefile);
130         wscopy(current_makefile, wcb);
131     }
132 #endif
133     if (max_include_depth++ >= 40) {
134         fatal(catgets(catd, 1, 66, "Too many nested include statements")
135     }
136     if (makefile->body.makefile.contents != NULL) {
137         retmem(makefile->body.makefile.contents);
138     }
139     source->inp_buf =
140         source->inp_buf_ptr =
141         source->inp_buf_end = NULL;
142     source->error_converting = false;
143     makefile->body.makefile.contents = NULL;
144     makefile->body.makefile.size = 0;
145     if ((makefile_name->hash.length != 1) ||
146         (wcb[0] != (int) hyphen_char)) {
147         if ((makefile->body.makefile.contents == NULL) &&
148             (doname_it)) {
149             if (makefile_path == NULL) {
150                 add_dir_to_path(".",
151                     &makefile_path,
152                     -1);
153 #ifdef SUN5_0
154                 add_dir_to_path(NOCATGETS("/usr/share/lib/make")
155                     &makefile_path,
156                     -1);
157                 add_dir_to_path(NOCATGETS("/etc/default"),
158                     &makefile_path,
159                     -1);
160 #elif defined(HP_UX)
161                 findrundir_err = avo_find_run_dir(&run_dir);
162                 if (! findrundir_err) {
163                     (void) sprintf(makerules_dir, NOCATGETS(
164                         add_dir_to_path(makerules_dir,
165                             &makefile_path,
166                             -1);
167                 }
169                 add_dir_to_path(NOCATGETS("/opt/SUNWspro/share/l
170                     &makefile_path,
171                     -1);
172                 add_dir_to_path(NOCATGETS("/usr/share/lib/make")
173                     &makefile_path,
174                     -1);
175 #elif defined(linux)
176                 findrundir_err = avo_find_run_dir(&run_dir);
177                 if (! findrundir_err) {
178                     (void) sprintf(makerules_dir, NOCATGETS(
179                         add_dir_to_path(makerules_dir,
180                             &makefile_path,
181                             -1);
182                 }
184                 add_dir_to_path(NOCATGETS("/usr/SUNWspro/lib"),
185                     &makefile_path,
186                     -1);
187                 add_dir_to_path(NOCATGETS("/opt/SUNWspro/share/l
188                     &makefile_path,

```

```

189                                     -1);
190         add_dir_to_path(NOCATGETS("/usr/share/lib/make")
191             &makefile_path,
192             -1);
193     #else
194         add_dir_to_path(NOCATGETS("/usr/include/make"),
195             &makefile_path,
196             -1);
197 #endif
198     }
199     save_makefile_type = makefile_type;
200     makefile_type = reading_nothing;
201     if (doname(makefile_name, true, false) == build_dont_kno
202         /* Try normalized filename */
203         string_start=get_wstring(makefile_name->string_m
204         for (string_end=string_start+1; *string_end != N
205         normalized_makefile_name=normalize_name(string_s
206         if ((strcmp(makefile_name->string_mb, normalized
207             (doname(normalized_makefile_name, true,
208                 n = access_vroot(makefile_name->string_m
209                     4,
210                     chase_path ?
211                     makefile_path : NULL,
212                     VROOT_DEFAULT);
213         if (n == 0) {
214             get_vroot_path((char **) NULL,
215                 &path,
216                 (char **) NULL);
217             if ((path[0] == (int) period_cha
218                 (path[1] == (int) slash_char
219                     path += 2;
220             }
221             MBSTOWCS(wcs_buffer, path);
222             makefile_name = GETNAME(wcs_buff
223                 FIND_LENGTH);
224         }
225     }
226     retmem(string_start);
227     /*
228     * Commented out: retmem_mb(normalized_makefile_
229     * We have to return this memory, but it seems t
230     * in dmake or in Sun C++ 5.7 compiler (it works
231     * is compiled using Sun C++ 5.6).
232     */
233     // retmem_mb(normalized_makefile_name->string_mb
234     }
235     makefile_type = save_makefile_type;
236 }
237 source->string.free_after_use = false;
238 source->previous = NULL;
239 source->already_expanded = false;
240 /* Lock the file for read, but not when -n. */
241 if (lock_makefile &&
242     !do_not_exec_rule) {
244         make_state_lockfile = getmem(strlen(make_state->string_
245         (void) sprintf(make_state_lockfile,
246             NOCATGETS("%s.lock"),
247             make_state->string_mb);
248         (void) file_lock(make_state->string_mb,
249             make_state_lockfile,
250             (int *) &make_state_locked,
251             0);
252         if(!make_state_locked) {
253             printf(NOCATGETS("-- NO LOCKING for read\n"));
254             retmem_mb(make_state_lockfile);

```



```

255         make_state_lockfile = 0;
256         return failed;
257     }
258 }
259 if (makefile->body.makefile.contents == NULL) {
260     save_makefile_type = makefile_type;
261     makefile_type = reading_nothing;
262     if ((doname_it) &&
263         (doname(makefile_name, true, false) == build_failed)
264         if (complain) {
265             (void) fprintf(stderr,
266 #ifdef DISTRIBUTED
267                 catgets(catd, 1, 67, "dma
268 #else
269                 catgets(catd, 1, 237, "ma
270 #endif
271                 makefile_name->string_mb)
272             }
273             max_include_depth--;
274             makefile_type = save_makefile_type;
275             return failed;
276         }
277         makefile_type = save_makefile_type;
278         //
279         // Before calling exists() make sure that we have the ri
280         //
281         makefile_name->stat.time = file_no_time;

283         if (exists(makefile_name) == file_doesnt_exist) {
284             if (complain ||
285                 (makefile_name->stat.stat_errno != ENOENT))
286                 if (must_exist) {
287                     fatal(catgets(catd, 1, 68, "Can'
288                         makefile_name->string_mb,
289                         errmsg(makefile_name->
290                             stat.stat_errno));
291                 } else {
292                     warning(catgets(catd, 1, 69, "Ca
293                         makefile_name->string_mb
294                         errmsg(makefile_name->
295                             stat.stat_errno))
296                 }
297             }
298             max_include_depth--;
299             if (make_state_locked && (make_state_lockfile !=
300                 (void) unlink(make_state_lockfile);
301                 retmem_mb(make_state_lockfile);
302                 make_state_lockfile = NULL;
303                 make_state_locked = false;
304             }
305             retmem(wcb);
306             retmem_mb((char *)source);
307             return failed;
308         }
309         /*
310         * These values are the size and bytes of
311         * the MULTI-BYTE makefile.
312         */
313         orig_makefile->body.makefile.size =
314         makefile->body.makefile.size =
315         source->bytes_left_in_file =
316         makefile_name->stat.size;
317         if (report_file) {
318             for (dpp = &makefiles_used;
319                 *dpp != NULL;
320                 dpp = &(*dpp)->next);

```

```

321         dp = ALLOC(Dependency);
322         dp->next = NULL;
323         dp->name = makefile_name;
324         dp->automatic = false;
325         dp->stale = false;
326         dp->built = false;
327         *dpp = dp;
328     }
329     source->fd = open_vroot(makefile_name->string_mb,
330                             O_RDONLY,
331                             0,
332                             NULL,
333                             VROOT_DEFAULT);
334     if (source->fd < 0) {
335         if (complain || (errno != ENOENT)) {
336             if (must_exist) {
337                 fatal(catgets(catd, 1, 70, "Can'
338                     makefile_name->string_mb,
339                     errmsg(errno));
340             } else {
341                 warning(catgets(catd, 1, 71, "Ca
342                     makefile_name->string_mb
343                     errmsg(errno));
344             }
345             max_include_depth--;
346             return failed;
347         }
348     }
349     (void) fcntl(source->fd, F_SETFD, 1);
350     orig_makefile->body.makefile.contents =
351     makefile->body.makefile.contents =
352     source->string.text.p =
353     source->string.buffer.start =
354     ALLOC_WC((int) (makefile_name->stat.size + 2));
355     if (makefile_type == reading_cpp_file) {
356         forget_after_parse = true;
357     }
358     source->string.text.end = source->string.text.p;
359     source->string.buffer.end =
360     source->string.text.p + makefile_name->stat.size;
361 } else {
362     /* Do we ever reach here? */
363     source->fd = -1;
364     source->string.text.p =
365     source->string.buffer.start =
366     makefile->body.makefile.contents;
367     source->string.text.end =
368     source->string.buffer.end =
369     source->string.text.p + makefile->body.makefile.size
370     source->bytes_left_in_file =
371     makefile->body.makefile.size;
372 }
373 file_being_read = wcb;
374 } else {
375     char *stdin_text_p;
376     char *stdin_text_end;
377     char *stdin_buffer_start;
378     char *stdin_buffer_end;
379     *p_mb;
380     int num_mb_chars;
381     size_t num_wc_chars;

383     MBSTOWCS(wcs_buffer, NOCATGETS("Standard in"));
384     makefile_name = GETNAME(wcs_buffer, FIND_LENGTH);
385     /*
386     * Memory to read standard in, then convert it

```



```

*****
52411 Wed May 20 11:09:57 2015
new/usr/src/cmd/make/bin/make/common/read2.cc
make: remove SCCS ident stuff
*****
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 2005 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)read2.cc 1.53 06/12/12
27 */
29 #pragma ident      "@(#)read2.cc  1.53  06/12/12"

26 /*
27  *      read.c
28  *
29  *      This file contains the makefile reader.
30  */

32 /*
33  * Included files
34  */
35 #include <mk/defs.h>
36 #include <mksh/dosys.h>          /* sh_command2string() */
37 #include <mksh/macro.h>         /* expand_value() */
38 #include <mksh/misc.h>         /* retmem() */
39 #include <stdarg.h>             /* va_list, va_start(), va_end() */

41 /*
42  * Defined macros
43  */

45 /*
46  * typedefs & structs
47  */

49 /*
50  * Static variables
51  */
52 static Boolean      built_last_make_run_seen;

54 /*
55  * File table of contents
56  */

```

```

57 static Name_vector  enter_member_name(register wchar_t *lib_start, register
58 extern Name         normalize_name(register wchar_t *name_string, register i
59 static void         read_suffixes_list(register Name_vector depes);
60 static void         make_relative(wchar_t *to, wchar_t *result);
61 static void         print_rule(register Cmd_line command);
62 static void         sh_transform(Name *name, Name *value);

65 /*
66  *      enter_name(string, tail_present, string_start, string_end,
67  *                current_names, extra_names, target_group_seen)
68  *
69  *      Take one string and enter it as a name. The string is passed in
70  *      two parts. A make string and possibly a C string to append to it.
71  *      The result is stuffed in the vector current_names.
72  *      extra_names points to a vector that is used if current_names overflows.
73  *      This is allocad in the calling routine.
74  *      Here we handle the "lib.a[members]" notation.
75  *
76  *      Return value:
77  *                The name vector that was used
78  *
79  *      Parameters:
80  *      tail_present  Indicates if both C and make string was passed
81  *      string_start  C string
82  *      string_end    Pointer to char after last in C string
83  *      string        make style string with head of name
84  *      current_names Vector to deposit the name in
85  *      extra_names   Where to get next name vector if we run out
86  *      target_group_seen Pointer to boolean that is set if "+" is seen
87  *
88  *      Global variables used:
89  *      makefile_type When we read a report file we normalize paths
90  *      plus           Points to the Name "+"
91  */

93 Name_vector
94 enter_name(String string, Boolean tail_present, register wchar_t *string_start,
95 {
96     Name          name;
97     register wchar_t *cp;
98     wchar_t       ch;

100     /* If we were passed a separate tail of the name we append it to the */
101     /* make string with the rest of it */
102     if (tail_present) {
103         append_string(string_start, string, string_end - string_start);
104         string_start = string->buffer.start;
105         string_end = string->text.p;
106     }
107     ch = *string_end;
108     *string_end = (int) nul_char;
109     /*
110      * Check if there are any ( or [ that are not prefixed with $.
111      * If there are, we have to deal with the lib.a(members) format.
112      */
113     for (cp = (wchar_t *) wschr(string_start, (int) parenleft_char);
114          cp != NULL;
115          cp = (wchar_t *) wschr(cp + 1, (int) parenleft_char)) {
116         if (*(cp - 1) != (int) dollar_char) {
117             *string_end = ch;
118             return enter_member_name(string_start,
119                                     cp,
120                                     string_end,
121                                     current_names,
122                                     extra_names);

```

```
123     }
124 }
125 *string_end = ch;

127 if (makefile_type == reading_cpp_file) {
128     /* Remove extra ../ constructs if we are reading from a report f
129     name = normalize_name(string_start, string_end - string_start);
130 } else {
131     /*
132     * /tolik, fix bug 1197477/
133     * Normalize every target name before entering.
134     * ../obj/a.o and ../obj//a.o are not two different targets.
135     * There is only one target ../obj/a.o
136     */
137     /*name = GETNAME(string_start, string_end - string_start);*/
138     name = normalize_name(string_start, string_end - string_start);
139 }

141 /* Internalize the name. Detect the name "+" (target group here) */
142 if(current_names->used != 0 && current_names->names[current_names->used-1] == pl
143     if(name == plus) {
144         return current_names;
145     }
146 }
unchanged_portion_omitted
```

```
*****
```

```
12738 Wed May 20 11:09:58 2015
```

```
new/usr/src/cmd/make/bin/make/common/rep.cc
```

```
make: remove SCCS ident stuff
```

```
*****
```

```
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)rep.cc 1.25 06/12/12
27 */
28
29 #pragma ident      "@(#)rep.cc      1.25      06/12/12"
30
31
32 /*
33 * Included files
34 */
35 #include <mk/defs.h>
36 #include <mksh/misc.h>          /* retmem() */
37 #include <vroot/report.h>      /* NSE_DEPINFO */
38
39 /*
40 * Static variables
41 */
42 static Recursive_make recursive_list;
43 static Recursive_make *bpatch = &recursive_list;
44 static Boolean        changed;
45
46 /*
47 * File table of contents
48 */
49
50
51 /*
52 * report_recursive_init()
53 *
54 * Read the .nse_depinfo file and make a list of all the
55 * .RECURSIVE entries.
56 */
```

```
57 * Parameters:
58 *
59 * Static variables used:
60 *     bpatch           Points to slot where next cell should be added
61 *
62 * Global variables used:
63 *     recursive_name   The Name ".RECURSIVE", compared against
64 */
65
66 void
67 report_recursive_init(void)
68 {
69     char          *search_dir;
70     char          nse_depinfo[MAXPATHLEN];
71     FILE          *fp;
72     int           line_size, line_index;
73     wchar_t       *line;
74     wchar_t       *bigger_line;
75     wchar_t       *colon;
76     wchar_t       *dollar;
77     Recursive_make rp;
78
79     /*
80      * This routine can be called more than once, don't do
81      * anything after the first time.
82      */
83     if (depinfo_already_read) {
84         return;
85     } else {
86         depinfo_already_read = true;
87     }
88
89     search_dir = getenv(NOCATGETS("NSE_DEP"));
90     if (search_dir == NULL) {
91         return;
92     }
93     (void) sprintf(nse_depinfo, "%s/%s", search_dir, NSE_DEPINFO);
94     fp = fopen(nse_depinfo, "r");
95     if (fp == NULL) {
96         return;
97     }
98     line_size = MAXPATHLEN;
99     line_index = line_size - 1;
100    line = ALLOC_WC(line_size);
101    Wstring rns(recursive_name);
102    wchar_t *wcb = rns.get_string();
103    while (fgetws(line, line_size, fp) != NULL) {
104        while (wslen(line) == line_index) {
105            if (line[wslen(line) - 1] == '\n') {
106                continue;
107            }
108            bigger_line = ALLOC_WC(2 * line_size);
109            wscopy(bigger_line, line);
110            retmem(line);
111            line = bigger_line;
112            if (fgetws(&line[line_index], line_size, fp) == NULL)
113                continue;
114            line_index = 2 * line_index;
115            line_size = 2 * line_size;
116        }
117
118        colon = (wchar_t *) wschr(line, (int) colon_char);
119        if (colon == NULL) {
120            continue;
121        }
122        dollar = (wchar_t *) wschr(line, (int) dollar_char);
```

```
123     line[wslen(line) - 1] = (int) nul_char;
124     if (IS_WEQUALN(&colon[2], wcb,
125         (int) recursive_name->hash.length)) {
126         /*
127          * If this entry is an old entry, ignore it
128          */
129         MBSTOWCS(wcs_buffer, DEPINFO_FMT_VERSION);
130         if (dollar == NULL ||
131             !IS_WEQUALN(wcs_buffer, (dollar+1) - VER_LEN, VER_LE
132                 continue;
133         }
134         rp = ALLOC(Recursive_make);
135         (void) memset((char *) rp, 0, sizeof (Recursive_make_rec
136         /*
137          * set conditional_macro_string if string is present
138          */
139         rp->oldline = (wchar_t *) wsdup(line);
140         if ( dollar != NULL ){
141             rp->cond_macrostring =
142                 (wchar_t *) wsdup(dollar - VER_LEN + 1);
143         }
144         /*
145          * get target name into recursive struct
146          */
147         *colon = (int) nul_char;
148         rp->target = (wchar_t *) wsdup(line);
149         *bpatch = rp;
150         bpatch = &rp->next;
151     }
152 }
153 (void) fclose(fp);
154 }
```

unchanged_portion_omitted

```

*****
12702 Wed May 20 11:09:58 2015
new/usr/src/cmd/make/bin/make/common/state.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)state.cc 1.27 06/12/12
27 */
28
29 #pragma ident      "@(#)state.cc  1.27   06/12/12"
30
31
32 /*
33  * Included files
34  */
35 #include <mk/defs.h>
36 #include <mksh/misc.h>          /* errmsg() */
37 #include <setjmp.h>              /* setjmp() */
38 #include <unistd.h>              /* getpid() */
39 #include <errno.h>              /* errno */
40 #include <locale.h>             /* MB_CUR_MAX */
41
42 /*
43  * Defined macros
44  */
45 #define LONGJUMP_VALUE 17
46 #define XFWRITE(string, length, fd) {if (fwrite(string, 1, length, fd) == 0) \
47     longjmp(long_jump, LONGJUMP_VALUE);}
48 #define XPUTC(ch, fd) { \
49     if (putc((int) ch, fd) == EOF) \
50         longjmp(long_jump, LONGJUMP_VALUE); \
51     }
52 #define XFPUTS(string, fd) fputs(string, fd)
53
54 /*
55  * typedefs & structs
56  */

```

```

58 /*
59  * Static variables
60  */
61
62 /*
63  * File table of contents
64  */
65 static char * escape_target_name(Name np)
66 {
67     if(np->dollar) {
68         int len = strlen(np->string_mb);
69         char * buff = (char*)malloc(2 * len);
70         int pos = 0;
71         wchar_t wc;
72         int pp = 0;
73         while(pos < len) {
74             int n = mbtowlc(&wc, np->string_mb + pos, MB_CUR_MAX);
75             if(n < 0) { // error - this shouldn't happen
76                 (void)free(buff);
77                 return strdup(np->string_mb);
78             }
79             if(wc == dollar_char) {
80                 buff[pp] = '\\'; pp++;
81                 buff[pp] = '$'; pp++;
82             } else {
83                 for(int j=0; j<n; j++) {
84                     buff[pp] = np->string_mb[pos+j]; pp++;
85                 }
86             }
87             pos += n;
88         }
89         buff[pp] = '\\0';
90         return buff;
91     } else {
92         return strdup(np->string_mb);
93     }
94 }
_____unchanged_portion_omitted_____

```

4597 Wed May 20 11:09:59 2015
new/usr/src/cmd/make/bin/make/common/svr4.make.rules.file
make: remove SCCS ident stuff

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 1994 Sun Microsystems, Inc. All rights reserved.
22 # Use is subject to license terms.
23 #
24 # @(#)svr4.make.rules.file 1.4 06/12/12
24 #
25 .SUFFIXES: .o .c .c~ .y .y~ .l .l~ .s .s~ .sh .sh~ .h .h~ .f .f~ \
26 .C .C~ .Y .Y~ .L .L~
28 MAKE=make
29 BUILD=build
30 AR=ar
31 ARFLAGS=rv
32 AS=as
33 ASFLAGS=
34 CC=cc
35 CFLAGS=-O
36 F77=f77
37 FFLAGS=-O
38 GET=get
39 GFLAGS=
40 LD=ld
41 LDFLAGS=
42 LEX=lex
43 LFLAGS=
44 YACC=yacc
45 YFLAGS=
46 C++C=CC
47 C++FLAGS=-O

50 .c: \$(CC) \$(CFLAGS) \$< -o \$@ \$(LDFLAGS)
51
52 .c~: \$(GET) \$(GFLAGS) \$<
53 \$(CC) \$(CFLAGS) \$*.c -o \$@ \$(LDFLAGS)
54 -rm -f \$*.c
55
56 .f: \$(F77) \$(FFLAGS) \$< -o \$@ \$(LDFLAGS)
57
58 .f~: \$(GET) \$(GFLAGS) \$<
59 \$(F77) \$(FFLAGS) \$*.f -o \$@ \$(LDFLAGS)
60

61 -rm -f \$*.f
62 .s: \$(AS) \$(ASFLAGS) \$< -o \$@ \$(LDFLAGS)
63
64 .s~: \$(GET) \$(GFLAGS) \$<
65 \$(AS) \$(ASFLAGS) \$*.s -o \$* \$(LDFLAGS)
66 -rm -f \$*.s
67
68 .sh: cp \$< \$@; chmod 0777 \$@
69
70 .sh~: \$(GET) \$(GFLAGS) \$<
71 cp \$*.sh \$*; chmod 0777 \$@
72 -rm -f \$*.sh
73
74 .C: \$(C++C) \$(C++FLAGS) \$< -o \$@ \$(LDFLAGS)
75
76 .C~: \$(GET) \$(GFLAGS) \$<
77 \$(C++C) \$(C++FLAGS) \$*.C -o \$@ \$(LDFLAGS)
78 -rm -f \$*.C
79
81 .c.a: \$(CC) \$(CFLAGS) -c \$<
82 \$(AR) \$(ARFLAGS) \$@ \$*.o
83 -rm -f \$*.o
84
85 .c.o: \$(CC) \$(CFLAGS) -c \$<
86
87 .c~.a: \$(GET) \$(GFLAGS) \$<
88 \$(CC) \$(CFLAGS) -c \$*.c
89 \$(AR) \$(ARFLAGS) \$@ \$*.o
90 -rm -f \$*.[co]
91
92 .c~.c: \$(GET) \$(GFLAGS) \$<
93
94 .c~.o: \$(GET) \$(GFLAGS) \$<
95 \$(CC) \$(CFLAGS) -c \$*.c
96 -rm -f \$*.c
97
98 .f.a: \$(F77) \$(FFLAGS) -c \$*.f
99 \$(AR) \$(ARFLAGS) \$@ \$*.o
100 -rm -f \$*.o
101
102 .f.o: \$(F77) \$(FFLAGS) -c \$*.f
103
104 .f~.a: \$(GET) \$(GFLAGS) \$<
105 \$(F77) \$(FFLAGS) -c \$*.f
106 \$(AR) \$(ARFLAGS) \$@ \$*.o
107 -rm -f \$*.[fo]
108
109 .f~.f: \$(GET) \$(GFLAGS) \$<
110
111 .f~.o: \$(GET) \$(GFLAGS) \$<
112 \$(F77) \$(FFLAGS) -c \$*.f
113 -rm -f \$*.f
114
115 .h~.h: \$(GET) \$(GFLAGS) \$<
116
117 .l.c: \$(LEX) \$(LFLAGS) \$<
118 mv lex.yy.c \$@
119
120 .l.o: \$(LEX) \$(LFLAGS) \$<
121 \$(CC) \$(CFLAGS) -c lex.yy.c
122 -rm lex.yy.c; mv lex.yy.o \$@
123
124 .l~.c: \$(GET) \$(GFLAGS) \$<
125 \$(LEX) \$(LFLAGS) \$*.l
126


```

127 mv lex.yy.c @$
128 -rm -f $*.l
129 .l~.l: $(GET) $(GFLAGS) $<
130 $(GET) $(GFLAGS) $<
131 .l~.o: $(GET) $(GFLAGS) $<
132 $(GET) $(GFLAGS) $<
133 $(LEX) $(LFLAGS) $*.l
134 $(CC) $(CFLAGS) -c lex.yy.c
135 -rm -f lex.yy.c $*.l
136 mv lex.yy.o @$
137 .s.a: $(AS) $(ASFLAGS) -o $*.o $*.s
138 $(AR) $(ARFLAGS) $@ $*.o
139 .s.o: $(AS) $(ASFLAGS) -o $@ $<
140 .s~.a: $(GET) $(GFLAGS) $<
141 $(AS) $(ASFLAGS) -o $*.o $*.s
142 $(AR) $(ARFLAGS) $@ $*.o
143 -rm -f $*.[so]
144 .s~.o: $(GET) $(GFLAGS) $<
145 $(AS) $(ASFLAGS) -o $*.o $*.s
146 -rm -f $*.s
147 .s~.s: $(GET) $(GFLAGS) $<
148 .sh~.sh: $(GET) $(GFLAGS) $<
149 .y.c: $(YACC) $(YFLAGS) $<
150 mv y.tab.c @$
151 .y.o: $(YACC) $(YFLAGS) $<
152 $(CC) $(CFLAGS) -c y.tab.c
153 -rm y.tab.c
154 mv y.tab.o @$
155 .y~.c: $(GET) $(GFLAGS) $<
156 $(YACC) $(YFLAGS) $*.y
157 mv y.tab.c $*.c
158 -rm -f $*.y
159 .y~.o: $(GET) $(GFLAGS) $<
160 $(YACC) $(YFLAGS) $*.y
161 $(CC) $(CFLAGS) -c y.tab.c
162 -rm -f y.tab.c $*.y
163 mv y.tab.o $*.o
164 .y~.y : $(GET) $(GFLAGS) $<
165 .C.a: $(C++) $(C++FLAGS) -c $<
166 $(AR) $(ARFLAGS) $@ $*.o
167 -rm -f $*.o
168 .C.o: $(C++) $(C++FLAGS) -c $<
169 .C~.a: $(GET) $(GFLAGS) $<
170 $(C++) $(C++FLAGS) -c $*.C
171 $(AR) $(ARFLAGS) $@ $*.o
172 -rm -f $*.[Co]
173 .C~.C: $(GET) $(GFLAGS) $<
174 .C~.o: $(GET) $(GFLAGS) $<
175 $(C++) $(C++FLAGS) -c $*.C
176 -rm -f $*.C

```

```

193 .L.C: $(LEX) $(LFLAGS) $<
194 mv lex.yy.c @$
195 .L.o: $(LEX) $(LFLAGS) $<
196 $(C++) $(C++FLAGS) -c lex.yy.c
197 -rm lex.yy.c; mv lex.yy.o @$
198 .L~.C: $(GET) $(GFLAGS) $<
199 $(LEX) $(LFLAGS) $*.L
200 mv lex.yy.c @$
201 -rm -f $*.L
202 .L~.L: $(GET) $(GFLAGS) $<
203 .L~.o: $(GET) $(GFLAGS) $<
204 $(LEX) $(LFLAGS) $*.L
205 $(C++) $(C++FLAGS) -c lex.yy.c
206 -rm -f lex.yy.c $*.L
207 mv lex.yy.c @$
208 .Y.C: $(YACC) $(YFLAGS) $<
209 mv y.tab.c @$
210 .Y.o: $(YACC) $(YFLAGS) $<
211 $(C++) $(C++FLAGS) -c y.tab.c
212 -rm y.tab.c
213 mv y.tab.o @$
214 .Y~.C: $(GET) $(GFLAGS) $<
215 $(YACC) $(YFLAGS) $*.Y
216 mv y.tab.c $*.C
217 -rm -f $*.Y
218 .Y~.o: $(GET) $(GFLAGS) $<
219 $(YACC) $(YFLAGS) $*.Y
220 $(C++) $(C++FLAGS) -c y.tab.c
221 -rm -f y.tab.c $*.Y
222 mv y.tab.o $*.o
223 .Y~.Y : $(GET) $(GFLAGS) $<
224 markfile.o: markfile
225 echo "static char _sccsid[] = \"'grep @'(#)' markfile\\\";" > markfile.c
226 $(CC) -c markfile.c
227 -rm -f markfile.c
228 .SCCS_GET: $(GET) $(GFLAGS) s.$@
229

```

```
new/usr/src/cmd/make/include/avo/avo_alloc.h
```

1

```
*****
```

```
1530 Wed May 20 11:09:59 2015
```

```
new/usr/src/cmd/make/include/avo/avo_alloc.h
```

```
make: remove SCCS ident stuff
```

```
*****
```

```
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 1998 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)avo_alloc.h 1.4 06/12/12
27  */
29 #pragma ident    "@(#)avo_alloc.h 1.4    06/12/12"

26 #ifndef _AVO_ALLOC_H
27 #define _AVO_ALLOC_H

29 #include <alloca.h>

31 #ifdef __SunOS_5_4
32 // The following prototype declaration is necessary when compiling on Solaris
33 // 2.4 using 5.0 compilers. On Solaris 2.4 the necessary prototypes are not
34 // included in alloca.h. The 4.x compilers provide a workaround by declaring the
35 // prototype as a pre-defined type. The 5.0 compilers do not implement this work
36 // This can be removed when support for 2.4 is dropped

38 #include <stdlib.h>    // for size_t
39 extern "C" void *__builtin_alloca(size_t);

41 #endif // ifdef __SunOS_5_4

43 #endif // ifdef _AVO_ALLOC_H
```

new/usr/src/cmd/make/include/avo/intl.h

1

2109 Wed May 20 11:10:00 2015

new/usr/src/cmd/make/include/avo/intl.h

make: remove SCCS ident stuff

```
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 2001 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)intl.h 1.19 06/12/12
27 */
28
29 #pragma ident      "@(#)intl.h      1.19      06/12/12"
30
31 #ifndef _AVO_INTL_H
32 #define _AVO_INTL_H
33
34 #if defined(SUN4_x) || defined(HP_UX)
35 #include <avo/widefake.h>
36 #endif
37
38 /*
39  * For catgets
40  */
41 #include <nl_types.h>
42
43 #ifdef HP_UX
44 #ifdef __cplusplus
45 #ifndef _STDLIB_INCLUDED
46 #include <stdlib.h>          /* for wchar_t definition and HP-UX - */
47 #endif
48 #endif
49 extern "C" {
50 char *gettext(char *msg);
51 char *dgettext(const char *, const char *);
52 char *bindtextdomain(const char *, const char *);
53 char *textdomain(char *);
54 }
55 #endif
56
57 unchanged portion omitted
```

new/usr/src/cmd/make/include/bsd/bsd.h

1

1434 Wed May 20 11:10:00 2015

new/usr/src/cmd/make/include/bsd/bsd.h

make: remove SCCS ident stuff

```
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)bsd.h 1.6 06/12/12
27 */
28
29 #pragma ident      "@(#)bsd.h      1.6      06/12/12"
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
```

new/usr/src/cmd/make/include/mk/copyright.h

1

925 Wed May 20 11:10:01 2015

new/usr/src/cmd/make/include/mk/copyright.h

make: remove SCCS ident stuff

```
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 */
25 /*
26 * @(#)copyright.h 1.11 06/12/12
27 */
```

```

*****
16559 Wed May 20 11:10:01 2015
new/usr/src/cmd/make/include/mk/defs.h
make: remove SCCS ident stuff
*****
1 #ifndef _MK_DEFS_H
2 #define _MK_DEFS_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)defs.h 1.61 06/12/12
29 */
31 #pragma ident    "@(#)defs.h      1.61    06/12/12"

28 /*
29 * Included files
30 */
31 #ifdef DISTRIBUTED
32 #   include <dm/Avo_AcknowledgeMsg.h>
33 #   include <dm/Avo_DoJobMsg.h>
34 #   include <dm/Avo_JobResultMsg.h>
35 #endif

37 #include <mksh/defs.h>

39 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
40 #   include <rw/xdrstrea.h>
41 #endif

44 /*
45 * Defined macros
46 */

48 #define SKIPSPACE(x)   while (*x &&
49                       ((*x == (int) space_char) ||
50                       (*x == (int) tab_char) ||
51                       (*x == (int) comma_char))) {
52                       x++;
53                       }

55 #define SKIPWORD(x)   while (*x &&
56                       (*x != (int) space_char) &&

```

```

57                       (*x != (int) tab_char) &&
58                       (*x != (int) newline_char) &&
59                       (*x != (int) comma_char) &&
60                       (*x != (int) equal_char)) {
61                       x++;
62                       }

64 #define SKIPTOEND(x)   while (*x &&
65                       (*x != (int) newline_char)) {
66                       x++;
67                       }

69 #define PMAKE_DEF_MAX_JOBS      2          /* Default number of parallel jobs. */

71 #define OUT_OF_DATE(a,b) \
72     (((a) < (b)) || (((a) == file_doesnt_exist) && ((b) == file_doesnt_exist)

74 #define OUT_OF_DATE_SEC(a,b) \
75     (((a).tv_sec < (b).tv_sec) || (((a).tv_sec == file_doesnt_exist.tv_sec)

77 #define SETVAR(name, value, append) \
78     setvar_daemon(name, value, append, no_daemon, \
79     true, debug_level)

80 #ifdef SUN5_0
81 #define MAX(a,b)         (((a)>(b))? (a):(b))
82 /*
83  * New feature added to SUN5_0 make, invoke the vanilla svr4 make when
84  * the USE_SVR4_MAKE environment variable is set.
85  */
86 #define SVR4_MAKE       "/usr/ccs/lib/svr4.make"
87 #define USE_SVR4_MAKE   "USE_SVR4_MAKE"
88 #endif
89 /*
90  * The standard MAXHOSTNAMELEN is 64. We want 32.
91  */
92 #define MAX_HOSTNAMELEN      32

95 /*
96 * typedefs & structs
97 */
98 typedef enum {
99     no_state,
100    scan_name_state,
101    scan_command_state,
102    enter_dependencies_state,
103    enter_conditional_state,
104    enter_equal_state,
105    illegal_bytes_state,
106    illegal_eoln_state,
107    poorly_formed_macro_state,
108    exit_state
109 }
110 Reader_state;
111
112 unchanged_portion_omitted

```

new/usr/src/cmd/make/include/mksdmsi18n/mksdmsi18n.h

1

1126 Wed May 20 11:10:01 2015

new/usr/src/cmd/make/include/mksdmsi18n/mksdmsi18n.h

make: remove SCCS ident stuff

```
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 1996 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)mksdmsi18n.h 1.3 06/12/12
27 */
28
29 #pragma ident      "@(#)mksdmsi18n.h      1.3      06/12/12"
30
31 #ifndef _AVO_MKSDMSI18N_H
32 #define _AVO_MKSDMSI18N_H
33
34 #ifndef _AVO_INTL_H
35 #include <avo/intl.h>
36 #endif
37
38 extern nl_catd libmksdmsi18n_catd;
39
40 int libmksdmsi18n_init();
41 void libmksdmsi18n_fini();
42
43 #endif
```

new/usr/src/cmd/make/include/mksh/defs.h

1

```
*****
24558 Wed May 20 11:10:02 2015
new/usr/src/cmd/make/include/mksh/defs.h
make: remove SCCS ident stuff
*****
1 #ifndef _MKSH_DEFS_H
2 #define _MKSH_DEFS_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)defs.h 1.35 06/12/12
29 */
31 #pragma ident    "@(#)defs.h      1.35    06/12/12"

28 /*
29 * This is not "#ifdef TEAMWARE_MAKE_CMN" because we're currently
30 * using the TW fake i18n headers and libraries to build both
31 * SMake and PMake on SPARC/S1 and x86/S2.
32 */

34 #include <avo/intl.h>
35 #include <limits.h>           /* MB_LEN_MAX */
36 #include <stdio.h>
37 #include <stdlib.h>          /* wchar_t */
38 #include <string.h>          /* strcmp() */
39 #include <nl_types.h>        /* catgets() */
40 #include <sys/param.h>       /* MAXPATHLEN */
41 #include <sys/types.h>       /* time_t, caddr_t */
42 #include <vroot/vroot.h>     /* pathpt */
43 #include <sys/time.h>        /* timestruc_t */
44 #include <errno.h>           /* errno */

46 #if defined (HP_UX) || defined (linux)
47 #define MAXNAMELEN          256
48 #define RW_NO_OVERLOAD_WCHAR 1 /* Rogue Wave, belongs in <rw/compiler.h> */
49 #else
50 #include <wctype.h>
51 #include <wider.h>
52 #endif

54 #if defined (linux)
55 /*
56  * Definition of wchar functions.
```

new/usr/src/cmd/make/include/mksh/defs.h

2

```
57 */
58 #   include <wctype.h>
59 #   include <wchar.h>
60 #   define wsdup(x) wcsdup(x)
61 #   define wschr(x,y) wcschr(x,y)
62 #   define wscat(x,y) wscat(x,y)
63 #   define wsrchr(x,y) wsrchr(x,y)
64 #   define wslen(x) wcslen(x)
65 #   define wscopy(x,y) wscopy(x,y)
66 #   define wsncpy(x,y,z) wsncpy(x,y,z)
67 #   define wscmp(x,y) wscmp(x,y)
68 #   define wncmp(x,y,z) wncmp(x,y,z)
69 #endif

71 /*
72  * A type and some utilities for boolean values
73 */

75 #define false    BOOLEAN_false
76 #define true     BOOLEAN_true

78 typedef enum {
79     false =      0,
80     true  =      1,
81     failed =     0,
82     succeeded =  1
83 } Boolean;
_____unchanged_portion_omitted_
```


new/usr/src/cmd/make/include/mksh/dosys.h

1

2295 Wed May 20 11:10:03 2015

new/usr/src/cmd/make/include/mksh/dosys.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_DOSYS_H
2 #define _MKSH_DOSYS_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)dosys.h 1.9 06/12/12
29 */
31 #pragma ident    "@(#)dosys.h  1.9    06/12/12"

28 #include <mksh/defs.h>
29 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
30 #    include <rw/xdrstrea.h>
31 #endif
32 #include <vroot/vroot.h>

34 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
35 extern Boolean  await(register Boolean ignore_error, register Boolean silent_err
36 #else
37 extern Boolean  await(register Boolean ignore_error, register Boolean silent_err
38 #endif
39 extern int     doexec(register wchar_t *command, register Boolean ignore_error,
40 extern int     doshell(wchar_t *command, register Boolean ignore_error, Boolean
41 extern Doname  dosys_mksh(register Name command, register Boolean ignore_error,
42 extern void    redirect_io(char *stdout_file, char *stderr_file);
43 extern void    sh_command2string(register String command, register String desti

45 #endif
```

new/usr/src/cmd/make/include/mksh/globals.h

1

1005 Wed May 20 11:10:03 2015

new/usr/src/cmd/make/include/mksh/globals.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_GLOBALS_H
2 #define _MKSH_GLOBALS_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 1994 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)globals.h 1.2 06/12/12
29 */
31 #pragma ident    "@(#)globals.h 1.2    06/12/12"

28 #include <mksh/defs.h>

30 #endif
```

new/usr/src/cmd/make/include/mksh/i18n.h

1

1099 Wed May 20 11:10:04 2015

new/usr/src/cmd/make/include/mksh/i18n.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_I18N_H
2 #define _MKSH_I18N_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 1994 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)i18n.h 1.2 06/12/12
29 */
31 #pragma ident      "@(#)i18n.h      1.2      06/12/12"

28 #include <mksh/defs.h>

30 extern int      get_char_semantics_entry(wchar_t ch);
31 extern char     get_char_semantics_value(wchar_t ch);

33 #endif
```

new/usr/src/cmd/make/include/mksh/libmksh_init.h

1

1016 Wed May 20 11:10:04 2015

new/usr/src/cmd/make/include/mksh/libmksh_init.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_INIT_H
2 #define _MKSH_INIT_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 1995 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)libmksh_init.h 1.2 06/12/12
29 */
31 #pragma ident    "@(#)libmksh_init.h      1.2      06/12/12"

28 int libmksh_init()
29 void libmksh_fini();

31 #endif
```

new/usr/src/cmd/make/include/mksh/macro.h

1

1393 Wed May 20 11:10:05 2015

new/usr/src/cmd/make/include/mksh/macro.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_MACRO_H
2 #define _MKSH_MACRO_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2002 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)macro.h 1.3 06/12/12
29 */
31 #pragma ident    "@(#)macro.h  1.3   06/12/12"

28 #include <mksh/defs.h>

30 extern void      expand_macro(register Source source, register String destination
31 extern void      expand_value(Name value, register String destination, Boolean cm
32 extern Name      getvar(register Name name);

34 extern Property setvar_daemon(register Name name, register Name value, Boolean a

36 #endif
```

new/usr/src/cmd/make/include/mksh/misc.h

1

2241 Wed May 20 11:10:05 2015

new/usr/src/cmd/make/include/mksh/misc.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_MISC_H
2 #define _MKSH_MISC_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)misc.h 1.4 06/12/12
29 */
31 #pragma ident    "@(#)misc.h 1.4 06/12/12"
28 #include <mksh/defs.h>
30 extern void      append_char(wchar_t from, register String to);
31 extern Property append_prop(register Name target, register Property_id type);
32 extern void      append_string(register wchar_t *from, register String to, regist
33 extern void      enable_interrupt(register void (*handler) (int));
34 extern char      *errmsg(int errnum);
35 extern void      fatal_mksh(char * message, ...);
36 extern void      fatal_reader_mksh(char * pattern, ...);
37 extern char      *get_current_path_mksh(void);
38 extern Property get_prop(register Property start, register Property_id type);
39 extern char      *getmem(register int size);
40 extern Name      getname_fn(wchar_t *name, register int len, register Boolean don
41 extern void      store_name(Name name);
42 extern void      free_name(Name name);
43 extern void      handle_interrupt_mksh(int);
44 extern Property maybe_append_prop(register Name target, register Property_id typ
45 extern void      retmem(wchar_t *p);
46 extern void      retmem_mb(caddr_t p);
47 extern void      setup_char_semantics(void);
48 extern void      setup_interrupt(register void (*handler) (int));
49 extern void      warning_mksh(char * message, ...);
51 extern void      append_string(register char *from, register String to, register
52 extern wchar_t   *get_wstring(char * from);
55 #endif
```

new/usr/src/cmd/make/include/mksh/mksh.h

1

1436 Wed May 20 11:10:06 2015

new/usr/src/cmd/make/include/mksh/mksh.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_MKSH_H
2 #define _MKSH_MKSH_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 2003 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)mksh.h 1.7 06/12/12
29 */
31 #pragma ident      "@(#)mksh.h      1.7      06/12/12"
29 /*
30 * Included files
31 */
32 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
33 #include <dm/Avo_DmakeCommand.h>
34 #endif
36 #include <mksh/defs.h>
37 #include <unistd.h>
39 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
41 extern int      do_job(Avo_DmakeCommand *cmd_list[], char *env_list[], char *std
43 #endif /* TEAMWARE_MAKE_CMN */
45 #endif
```

new/usr/src/cmd/make/include/mksh/read.h

1

1058 Wed May 20 11:10:06 2015

new/usr/src/cmd/make/include/mksh/read.h

make: remove SCCS ident stuff

```
1 #ifndef _MKSH_READ_H
2 #define _MKSH_READ_H
3 /*
4  * CDDL HEADER START
5  *
6  * The contents of this file are subject to the terms of the
7  * Common Development and Distribution License (the "License").
8  * You may not use this file except in compliance with the License.
9  *
10 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
11 * or http://www.opensolaris.org/os/licensing.
12 * See the License for the specific language governing permissions
13 * and limitations under the License.
14 *
15 * When distributing Covered Code, include this CDDL HEADER in each
16 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
17 * If applicable, add the following below this CDDL HEADER, with the
18 * fields enclosed by brackets "[]" replaced with your own identifying
19 * information: Portions Copyright [yyyy] [name of copyright owner]
20 *
21 * CDDL HEADER END
22 */
23 /*
24 * Copyright 1994 Sun Microsystems, Inc. All rights reserved.
25 * Use is subject to license terms.
26 */
27 /*
28 * @(#)read.h 1.2 06/12/12
29 */
30 #pragma ident    "@(#)read.h      1.2      06/12/12"
31 #include <mksh/defs.h>
32 extern Source    get_next_block_fn(register Source source);
33 #endif
```


new/usr/src/cmd/make/include/vroot/args.h

1

1962 Wed May 20 11:10:06 2015

new/usr/src/cmd/make/include/vroot/args.h

make: remove SCCS ident stuff

```
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 1999 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)args.h 1.7 06/12/12
27 */
28
29 #pragma ident      "@(#)args.h      1.7      06/12/12"
30
31 #ifndef _ARGS_H_
32 #define _ARGS_H_
33
34 #include <sys/syscall.h>
35 #include <errno.h>
36 #include <sys/time.h>
37 #include <sys/param.h>
38 #include <stdio.h>
39 #include <fcntl.h>
40 #include <sys/types.h>
41 #include <sys/stat.h>
42 #include <sys/file.h>
43
44 typedef enum { rw_read, rw_write} rwt, *rwpt;
45
46 extern void translate_with_thunk(register char *filename, int (*thunk) (char
47
48 union Args {
49     struct { int mode;} access;
50     struct { int mode;} chmod;
51     struct { int user; int group;} chown;
52     struct { int mode;} creat;
53     struct { char **argv; char **environ;} execve;
54     struct { struct stat *buffer;} lstat;
55     struct { int mode;} mkdir;
56     struct { char *name; int mode;} mount;
57     struct { int flags; int mode;} open;
58     struct { char *buffer; int buffer_size;} readlink;
59     struct { struct stat *buffer;} stat;
60 #ifndef SUN5_0
61     struct { struct statfs *buffer;} statfs;
62 #endif
63 }
```

new/usr/src/cmd/make/include/vroot/args.h

2

```
58 #endif
59     struct { int length;} truncate;
60     struct { struct timeval *time;} utimes;
61 };
62
63 unchanged_portion_omitted
```

new/usr/src/cmd/make/include/vroot/report.h

1

1758 Wed May 20 11:10:07 2015

new/usr/src/cmd/make/include/vroot/report.h

make: remove SCCS ident stuff

```
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 1994 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)report.h 1.6 06/12/12
27 */
29 #pragma ident "@(#)report.h 1.6 06/12/12"
26 #ifndef _REPORT_H_
27 #define _REPORT_H_
29 #include <stdio.h>
31 extern FILE *get_report_file(void);
32 extern char *get_target_being_reported_for(void);
33 extern void report_dependency(register char *name);
34 extern int file_lock(char *name, char *lockname, int *file_locked, int time)
35 #ifdef NSE
36 extern char *setenv(char *name, char *value);
37 #endif
39 #define SUNPRO_DEPENDENCIES "SUNPRO_DEPENDENCIES"
40 #define LD "LD"
41 #define COMP "COMP"
43 /* the following definitions define the interface between make and
44 * NSE - the two systems must track each other.
45 */
46 #define NSE_DEPINFO ".nse_depinfo"
47 #define NSE_DEPINFO_LOCK ".nse_depinfo.lock"
48 #define NSE_DEP_ENV "NSE_DEP"
49 #define NSE_TFS_PUSH "/usr/nse/bin/tfs_push"
50 #define NSE_TFS_PUSH_LEN 8
51 #define NSE_VARIANT_ENV "NSE_VARIANT"
52 #define NSE_RT_SOURCE_NAME "Shared_Source"
54 #endif
```

new/usr/src/cmd/make/include/vroot/vroot.h

1

2325 Wed May 20 11:10:08 2015

new/usr/src/cmd/make/include/vroot/vroot.h

make: remove SCCS ident stuff

```
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)vroot.h 1.10 06/12/12
27 */
28 #pragma ident "@(#)vroot.h 1.10 06/12/12"
29
30 #ifndef _VROOT_H_
31 #define _VROOT_H_
32
33 #include <stdio.h>
34 #include <nl_types.h>
35
36 #define VROOT_DEFAULT ((pathpt)-1)
37
38 typedef struct {
39     char *path;
40     short length;
41 } pathcellt, *pathcellpt, patht;
42
43 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/bsd/src/bsd.cc

1

4269 Wed May 20 11:10:09 2015

new/usr/src/cmd/make/lib/bsd/src/bsd.cc

make: remove SCCS ident stuff

```
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)bsd.cc 1.6 06/12/12
27 */
28
29 #pragma ident    "@(#)bsd.cc 1.6 06/12/12"
30
31 #include <signal.h>
32
33 #include <bsd/bsd.h>
34
35 /* External references.
36 */
37
38 /* Forward references.
39 */
40
41 /* Static data.
42 */
```

new/usr/src/cmd/make/lib/bsd/src/bsd.cc

2

```
39 extern SIG_PF
40 bsd_signal (int Signal, SIG_PF Handler)
41 {
42     auto SIG_PF          previous_handler;
43 #ifdef SUN5_0
44 #ifdef sun
45     previous_handler = sigset (Signal, Handler);
46 #else
47     auto struct sigaction    new_action;
48     auto struct sigaction    old_action;
49
50     new_action.sa_flags = SA_SIGINFO;
51     new_action.sa_handler = (void (*) ()) Handler;
52     (void) sigemptyset (&new_action.sa_mask);
53     (void) sigaddset (&new_action.sa_mask, Signal);
54
55     (void) sigaction (Signal, &new_action, &old_action);
56
57     previous_handler = (SIG_PF) old_action.sa_handler;
58 #endif
59 #elif defined(linux)
60     previous_handler = sigset (Signal, Handler);
61 #else
62     previous_handler = signal (Signal, Handler);
63 #endif
64     return previous_handler;
65 }
_____unchanged_portion_omitted_____
```

new/usr/src/cmd/make/lib/makestate/src/ld_file.c

1

3639 Wed May 20 11:10:09 2015

new/usr/src/cmd/make/lib/makestate/src/ld_file.c

make: remove SCCS ident stuff

```
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 1998 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)ld_file.c 1.7 06/12/12
27 */
28 #pragma ident    "@(#)ld_file.c 1.7    06/12/12"

26 #pragma init(ld_support_init)

28 #include <stdio.h>
29 #include <unistd.h>
30 #include <stdlib.h>
31 #include <string.h>
32 #include <libelf.h>
33 #include <sys/param.h>
34 #include <link.h>

36 #define SUNPRO_DEPENDENCIES    "SUNPRO_DEPENDENCIES"

38 /*
39 * Linked list of strings - used to keep lists of names
40 * of directories or files.
41 */

43 struct Stritem {
44     char *      str;
45     void *      next;
46 };
unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/makestate/src/lock.c

1

```
*****
4838 Wed May 20 11:10:09 2015
new/usr/src/cmd/make/lib/makestate/src/lock.c
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)lock.c 1.5 06/12/12
27 */

26 #include <stdio.h>
27 #include <stdlib.h>
28 #include <unistd.h>
29 #include <string.h>
30 #include <fcntl.h>
31 #include <sys/types.h>
32 #include <sys/param.h>
33 #include <sys/stat.h>
34 #include <sys/errno.h>
35 #include <errno.h>          /* errno */

37 #if defined(_LP64)
38 /*
39  * The symbols _sys_errlist and _sys_nerr are not visible in the
40  * LP64 libc. Use strerror(3C) instead.
41  */
42 #else /* !_LP64 */
43 extern char *      sys_errlist[];
44 extern int        sys_nerr;
45 #endif /* !_LP64 */

47 static void      file_lock_error();

49 /*
50 * This code stolen from the NSE library and changed to not depend
51 * upon any NSE routines or header files.
52 *
53 * Simple file locking.
54 * Create a symlink to a file. The "test and set" will be
55 * atomic as creating the symlink provides both functions.
56 *
57 * The timeout value specifies how long to wait for stale locks
58 * to disappear. If the lock is more than 'timeout' seconds old
```

new/usr/src/cmd/make/lib/makestate/src/lock.c

2

```
59 * then it is ok to blow it away. This part has a small window
60 * of vulnerability as the operations of testing the time,
61 * removing the lock and creating a new one are not atomic.
62 * It would be possible for two processes to both decide to blow
63 * away the lock and then have process A remove the lock and establish
64 * its own, and then then have process B remove the lock which accidentally
65 * removes A's lock rather than the stale one.
66 *
67 * A further complication is with the NFS. If the file in question is
68 * being served by an NFS server, then its time is set by that server.
69 * We can not use the time on the client machine to check for a stale
70 * lock. Therefore, a temp file on the server is created to get
71 * the servers current time.
72 *
73 * Returns an error message. NULL return means the lock was obtained.
74 *
75 */
76 char *
77 file_lock(char * name, char * lockname, int timeout)
78 {
79     int          r;
80     int          fd;
81     struct stat  statb;
82     struct stat  fs_statb;
83     char         tmpname[MAXPATHLEN];
84     static char  msg[MAXPATHLEN];

86     if (timeout <= 0) {
87         timeout = 15;
88     }
89     for (;;) {
90         r = symlink(name, lockname);
91         if (r == 0) {
92             return (NULL);
93         }
94         if (errno != EEXIST) {
95             file_lock_error(msg, name,
96                 (const char *)"symlink(%s, %s)", name, lockname);
97             return (msg);
98         }
99         for (;;) {
100             (void) sleep(1);
101             r = lstat(lockname, &statb);
102             if (r == -1) {
103                 /*
104                  * The lock must have just gone away - try
105                  * again.
106                  */
107                 break;
108             }

109             /*
110              * With the NFS the time given a file is the time on
111              * the file server. This time may vary from the
112              * client's time. Therefore, we create a tmpfile in
113              * the same directory to establish the time on the
114              * server and use this time to see if the lock has
115              * expired.
116              */
117             (void) sprintf(tmpname, "%s.XXXXXX", lockname);
118             (void) mktemp(tmpname);
119             fd = creat(tmpname, 0666);
120             if (fd != -1) {
121                 (void) close(fd);
122             } else {
123                 file_lock_error(msg, name,
124
```

```
125         (const char *)"creat(%s)", tmpname);
126         return (msg);
127     }
128     if (stat(tmpname, &fs_statb) == -1) {
129         file_lock_error(msg, name,
130             (const char *)"stat(%s)", tmpname);
131         return (msg);
132     }
133     (void) unlink(tmpname);
134     if (statb.st_mtime + timeout < fs_statb.st_mtime) {
135         /*
136          * The lock has expired - blow it away.
137          */
138         (void) unlink(lockname);
139         break;
140     }
141 }
142 }
143 /* NOTREACHED */
144 }
```

unchanged_portion_omitted

new/usr/src/cmd/make/lib/mksdmsi18n/src/libmksdmsi18n_init.cc

1

1579 Wed May 20 11:10:10 2015

new/usr/src/cmd/make/lib/mksdmsi18n/src/libmksdmsi18n_init.cc

make: remove SCCS ident stuff

```
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 1996 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)libmksdmsi18n_init.cc 1.5 06/12/12
27 */
28
29 #pragma ident      "@(#)libmksdmsi18n_init.cc      1.5      06/12/12"
30
31
32 #include <avo/intl.h>
33 #include <stdio.h>
34 #include <stdlib.h>
35
36 nl_catd libmksdmsi18n_catd;
37
38 /*
39  * Open the catalog file for libmksdmsi18n.  Users of this library must set
40  * NSLPATH first.  See avo_l18n_init().
41  */
42 int
43 libmksdmsi18n_init()
44 {
45     char          name[20];
46
47     if (getenv(NOCATGETS("NLSPATH")) == NULL) {
48         fprintf(stderr, NOCATGETS("Internal error: Set NLSPATH before op
49         return 1;
50     }
51     sprintf(name, NOCATGETS("libmksdmsi18n_%d"), I18N_VERSION);
52     libmksdmsi18n_catd = catopen(name, NL_CAT_LOCALE);
53     return 0;
54 }
55
56 _____
57 unchanged portion omitted
```


new/usr/src/cmd/make/lib/mksh/src/dosys.cc

1

```
*****
22997 Wed May 20 11:10:11 2015
new/usr/src/cmd/make/lib/mksh/src/dosys.cc
make: remove SCCS ident stuff
*****
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 2005 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)dosys.cc 1.38 06/12/12
27 */
29 #pragma ident      "@(#)dosys.cc  1.38  06/12/12"

27 /*
28  *      dosys.cc
29  *
30  *      Execute one commandline
31  */

33 /*
34  * Included files
35  */
36 #include <sys/wait.h>          /* WIFEXITED(status) */
37 #include <avo/avo_alloc.h>     /* alloca() */

39 #if defined(TEAMWARE_MAKE_CMN) || defined(MAKETOOL) /* tolik */
40 #   include <avo/strings.h> /* AVO_STRDUP() */
41 #if defined(DISTRIBUTED)
42 #   include <dm/Avo_CmdOutput.h>
43 #   include <rw/xdrstrea.h>
44 #endif
45 #endif

47 #include <stdio.h>             /* errno */
48 #include <errno.h>             /* errno */
49 #include <fcntl.h>             /* open() */
50 #include <mksh/dosys.h>
51 #include <mksh/macro.h>        /* getvar() */
52 #include <mksh/misc.h>        /* getmem(), fatal_mksh(), errmsg() */
53 #include <mkstdmsi18n/mkstdmsi18n.h> /* libmkstdmsi18n_init() */
54 #include <sys/signal.h>        /* SIG_DFL */
55 #include <sys/stat.h>          /* open() */
56 #include <sys/wait.h>          /* wait() */
57 #include <ulimit.h>            /* ulimit() */
```

new/usr/src/cmd/make/lib/mksh/src/dosys.cc

2

```
58 #include <unistd.h>           /* close(), dup2() */

60 #if defined(HP_UX) || defined(linux)
61 #   include <sys/param.h>
62 #   include <wctype.h>
63 #   include <wchar.h>
64 #endif

66 #if defined(linux)
67 #   define wslen(x) wcslen(x)
68 #   define wscopy(x,y) wcscopy(x,y)
69 #endif

71 /*
72  * Defined macros
73  */
74 #if defined(DISTRIBUTED) || defined(MAKETOOL) /* tolik */
75 #define SEND_MTOOL_MSG(cmds) \
76     if (send_mtool_msgs) { \
77         cmds \
78     }
79 #else
80 #define SEND_MTOOL_MSG(cmds)
81 #endif

83 /*
84  * typedefs & structs
85  */

87 /*
88  * Static variables
89  */

91 /*
92  * File table of contents
93  */
94 static Boolean  exec_vp(register char *name, register char **argv, char **envp,

96 /*
97  * Workaround for NFS bug. Sometimes, when running 'open' on a remote
98  * dmake server, it fails with "Stale NFS file handle" error.
99  * The second attempt seems to work.
100 */
101 int
102 my_open(const char *path, int oflag, mode_t mode) {
103     int res = open(path, oflag, mode);
104 #ifdef linux
105     // Workaround for NFS problem: even when all directories in 'path'
106     // exist, 'open' (file creation) fails with ENOENT.
107     int nattempt = 0;
108     while (res < 0 && (errno == ESTALE || errno == EAGAIN || errno == ENOENT
109         nattempt++;
110         if(nattempt > 30) {
111             break;
112         }
113         sleep(1);
114 #else
115     if (res < 0 && (errno == ESTALE || errno == EAGAIN)) {
116 #endif
117         /* Stale NFS file handle. Try again */
118         res = open(path, oflag, mode);
119     }
120     return res;
121 }
    unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/mksh/src/globals.cc

1

```
*****
3106 Wed May 20 11:10:11 2015
new/usr/src/cmd/make/lib/mksh/src/globals.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)globals.cc 1.16 06/12/12
27 */
28
29 #pragma ident      "@(#)globals.cc 1.16 06/12/12"
30
31 /*
32  *
33  *
34  *
35  *
36  *
37  *
38  *
39  *
40  *
41  *
42  *
43  *
44  *
45  *
46  *
47  * Global variables
48  */
49 char          char_semantics[CHAR_SEMANTICS_ENTRIES];
50 wchar_t      char_semantics_char[] = {
51     ampersand_char,
52     asterisk_char,
53     at_char,
54     backquote_char,
55     backslash_char,
56     bar_char,
57     bracketleft_char,
```

new/usr/src/cmd/make/lib/mksh/src/globals.cc

2

```
58     bracketright_char,
59     colon_char,
60     dollar_char,
61     doublequote_char,
62     equal_char,
63     exclam_char,
64     greater_char,
65     hat_char,
66     hyphen_char,
67     less_char,
68     newline_char,
69     numbersign_char,
70     parenleft_char,
71     parenright_char,
72     percent_char,
73     plus_char,
74     question_char,
75     quote_char,
76     semicolon_char,
77 #ifdef SGE_SUPPORT
78     space_char,
79     tab_char,
80 #endif
81     nul_char
82 };
unchanged_portion_omitted_
```

new/usr/src/cmd/make/lib/mksh/src/i18n.cc

1

2281 Wed May 20 11:10:11 2015

new/usr/src/cmd/make/lib/mksh/src/i18n.cc

make: remove SCCS ident stuff

```
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 2003 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)i18n.cc 1.3 06/12/12
27 */
28
29 #pragma ident      "@(#)i18n.cc  1.3  06/12/12"
30
31 /*
32  *
33  *
34  * Included files
35  */
36 #include <mksh/i18n.h>
37 #include <mksh/misc.h>          /* setup_char_semantics() */
38 #if defined (linux)
39 #   include <wctype.h>
40 #   include <wchar.h>
41 #   define wschr(x,y) wcschr(x,y)
42 #endif
43
44 /*
45  *   get_char_semantics_value(ch)
46  *
47  *   Return value:
48  *       The character semantics of ch.
49  *
50  *   Parameters:
51  *       ch           character we want semantics for.
52  *
53  */
54 char
55 get_char_semantics_value(wchar_t ch)
56 {
57     static Boolean  char_semantics_setup;
```

new/usr/src/cmd/make/lib/mksh/src/i18n.cc

2

```
59     if (!char_semantics_setup) {
60         setup_char_semantics();
61         char_semantics_setup = true;
62     }
63     return char_semantics[get_char_semantics_entry(ch)];
64 }
```

unchanged_portion_omitted

```

*****
2725 Wed May 20 11:10:12 2015
new/usr/src/cmd/make/lib/mksh/src/libmksh.msg
make: remove SCCS ident stuff
*****

2 $quote "

5 $set 1
6 $
7 $ CDDL HEADER START
8 $
9 $ The contents of this file are subject to the terms of the
10 $ Common Development and Distribution License (the "License").
11 $ You may not use this file except in compliance with the License.
12 $
13 $ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
14 $ or http://www.opensolaris.org/os/licensing.
15 $ See the License for the specific language governing permissions
16 $ and limitations under the License.
17 $
18 $ When distributing Covered Code, include this CDDL HEADER in each
19 $ file and include the License file at usr/src/OPENSOLARIS.LICENSE.
20 $ If applicable, add the following below this CDDL HEADER, with the
21 $ fields enclosed by brackets "[]" replaced with your own identifying
22 $ information: Portions Copyright [yyyy] [name of copyright owner]
23 $
24 $ CDDL HEADER END
25 $
26 $ Copyright 1996 Sun Microsystems, Inc. All rights reserved.
27 $ Use is subject to license terms.
28 $
29 $ @(#)libmksh.msg 1.2 06/12/12
29 $
30 89 "ulimit() failed: %s"
31 90 "Couldn't open standard out temp file '%s': %s"
32 91 "Couldn't open standard error temp file '%s': %s"
33 92 "Could not load '/usr/bin/nice': %s"
34 93 "Could not load Shell from '%s': %s"
35 94 "fork failed: %s"
36 95 "Command '%s' has more than %d arguments"
37 96 "Cannot load command '%s': %s"
38 97 "fork failed: %s"
39 98 "wait() failed: %s"
40 99 "Could not open filter file for -X"
41 100 "Could not stat filter file for -X"
42 101 "\n**** Error: Directory %s Target %s:\n%s\n"
43 102 "**** Error: Directory %s Target %s\n"
44 103 "**** Error code %d"
45 104 "**** Error code %d"
46 105 "**** Signal %d"
47 106 "**** Signal %d"
48 107 " - core dumped"
49 108 " - core dumped"
50 109 " (ignored)"
51 110 " (ignored)"
52 111 "Could not run command '%s' for :sh transformation"
53 112 "The command '%s' returned status '%d'"
54 113 "Loop detected when expanding macro value '%s'"
55 114 "'$' at end of string '%s'"
56 115 "'$' at end of line"
57 116 "Unmatched '%c' in string '%s'"
58 117 "Premature EOF"
59 118 "Unmatched '%c' on line"
60 119 "Illegal macro reference '%s'"

```

```

61 120 "= missing from replacement macro reference"
62 121 "= missing from replacement macro reference"
63 122 "% missing from replacement macro reference"
64 123 "% missing from replacement macro reference"
65 124 "Too many %% in pattern"
66 125 "Conditional macro '%s' referenced on line %d"
67 126 "Out of memory"
68 127 "Error %d"
69 128 "mksh: Fatal error: "
70 129 "Current working directory %s\n"
71 131 "mksh: Fatal error in reader: "
72 133 "Current working directory %s\n"
73 134 "mksh: Warning: "
74 135 "Current working directory %s\n"
75 136 "Internal error. Unknown prop type %d"
76 137 "'cd %s' failed, and conversion of %s to automounter pathname also failed"
77 138 "'cd %s' and 'cd %s' both failed"
78 139 "The following command caused the error:\n%s\n"
79 140 "Error reading '%s': Premature EOF"
80 141 "Error reading '%s': %s"

```

```

*****
39266 Wed May 20 11:10:13 2015
new/usr/src/cmd/make/lib/mksh/src/macro.cc
make: remove SCCS ident stuff
*****
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 */
25 /*
26 * @(#)macro.cc 1.22 06/12/12
27 */
28
29 #pragma ident      "@(#)macro.cc 1.22 06/12/12"
30
31 /*
32  * macro.cc
33  *
34  * Handle expansion of make macros
35  */
36
37 /*
38  * Included files
39  */
40 #include <mksh/dosys.h>          /* sh_command2string() */
41 #include <mksh/il8n.h>          /* get_char_semantics_value() */
42 #include <mksh/macro.h>
43 #include <mksh/misc.h>          /* retmem() */
44 #include <mksh/read.h>         /* get_next_block_fn() */
45 #include <mkstdmsil8n/mkstdmsil8n.h> /* libmkstdmsil8n_init() */
46
47 /*
48  * File table of contents
49  */
50
51 static void      add_macro_to_global_list(Name macro_to_add);
52 #ifdef NSE
53 static void      expand_value_with_daemon(Name name, register Property macro, reg
54 #else
55 static void      expand_value_with_daemon(Name, register Property macro, register
56 #endif
57
58 static void      init_arch_macros(void);
59 static void      init_mach_macros(void);
60 static Boolean   init_arch_done = false;
61 static Boolean   init_mach_done = false;

```

```

59 long env_alloc_num = 0;
60 long env_alloc_bytes = 0;
61
62 /*
63  *   getvar(name)
64  *
65  *   Return expanded value of macro.
66  *
67  *   Return value:
68  *
69  *   The expanded value of the macro
70  *
71  *   Parameters:
72  *   name
73  *   The name of the macro we want the value for
74  *
75  *   Global variables used:
76  */
77 Name
78 getvar(register Name name)
79 {
80     String_rec      destination;
81     wchar_t         buffer[STRING_BUFFER_LENGTH];
82     register Name   result;
83
84     if ((name == host_arch) || (name == target_arch)) {
85         if (!init_arch_done) {
86             init_arch_done = true;
87             init_arch_macros();
88         }
89     }
90     if ((name == host_mach) || (name == target_mach)) {
91         if (!init_mach_done) {
92             init_mach_done = true;
93             init_mach_macros();
94         }
95     }
96     INIT_STRING_FROM_STACK(destination, buffer);
97     expand_value(maybe_append_prop(name, macro_prop)->body.macro.value,
98                 &destination,
99                 false);
100     result = GETNAME(destination.buffer.start, FIND_LENGTH);
101     if (destination.free_after_use) {
102         retmem(destination.buffer.start);
103     }
104     return result;
105 }

```

unchanged_portion_omitted

```

*****
25893 Wed May 20 11:10:14 2015
new/usr/src/cmd/make/lib/mksh/src/misc.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)misc.cc 1.31 06/12/12
27 */
29 #pragma ident      "@(#)misc.cc 1.31 06/12/12"

27 /*
28  *      misc.cc
29  *
30  *      This file contains various unclassified routines. Some main groups:
31  *      getname
32  *      Memory allocation
33  *      String handling
34  *      Property handling
35  *      Error message handling
36  *      Make internal state dumping
37  *      main routine support
38  */

40 /*
41  * Included files
42  */
43 #include <bsd/bsd.h>          /* bsd_signal() */
44 #include <mksh/i18n.h>       /* get_char_semantics_value() */
45 #include <mksh/misc.h>
46 #include <mkstdmsi18n/mkstdmsi18n.h>
47 #include <stdarg.h>          /* va_list, va_start(), va_end() */
48 #include <stdlib.h>          /* mbstowcs() */
49 #include <sys/signal.h>      /* SIG_DFL */
50 #include <sys/wait.h>        /* wait() */

52 #ifdef SUN5_0
53 #include <string.h>          /* strerror() */
54 #endif

56 #if defined (HP_UX) || defined (linux)
57 #include <unistd.h>

```

```

58 #endif

60 /*
61  * Defined macros
62  */

64 /*
65  * typedefs & structs
66  */

68 /*
69  * Static variables
70  */
71 #ifdef SUN5_0
72 extern "C" void      (*sigvalue)(int) = SIG_DFL;
73 extern "C" void      (*siggvvalue)(int) = SIG_DFL;
74 extern "C" void      (*sigtvalue)(int) = SIG_DFL;
75 extern "C" void      (*sighvalue)(int) = SIG_DFL;
76 #else
77 static void          (*sigvalue)(int) = (void (*) (int)) SIG_DFL;
78 static void          (*siggvvalue)(int) = (void (*) (int)) SIG_DFL;
79 static void          (*sigtvalue)(int) = (void (*) (int)) SIG_DFL;
80 static void          (*sighvalue)(int) = (void (*) (int)) SIG_DFL;
81 #endif

83 long      getname_bytes_count = 0;
84 long      getname_names_count = 0;
85 long      getname_struct_count = 0;

87 long      freename_bytes_count = 0;
88 long      freename_names_count = 0;
89 long      freename_struct_count = 0;

91 long      expandstring_count = 0;
92 long      getwstring_count = 0;

94 /*
95  * File table of contents
96  */
97 static void      expand_string(register String string, register int length);

99 #define FATAL_ERROR_MSG_SIZE 200

101 /*
102  *      getmem(size)
103  *
104  *      malloc() version that checks the returned value.
105  *
106  *      Return value:
107  *                          The memory chunk we allocated
108  *
109  *      Parameters:
110  *          size              The size of the chunk we need
111  *
112  *      Global variables used:
113  */
114 char *
115 getmem(register int size)
116 {
117     register char      *result = (char *) malloc((unsigned) size);
118     if (result == NULL) {
119         char buf[FATAL_ERROR_MSG_SIZE];
120         sprintf(buf, NOCATGETS("*** Error: malloc(%d) failed: %s\n"), si
121             strcat(buf, catgets(libmkstdmsi18n_catd, 1, 126, "mksh: Fatal err
122             fputs(buf, stderr);
123 #ifdef SUN5_0

```

new/usr/src/cmd/make/lib/mksh/src/misc.cc

3

```
124             exit_status = 1;
125 #endif
126             exit(1);
127         }
128         return result;
129 }
unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/mksh/src/mksh.cc

1

```
*****
7598 Wed May 20 11:10:14 2015
new/usr/src/cmd/make/lib/mksh/src/mksh.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)mksh.cc 1.22 06/12/12
27 */
28
29 #pragma ident      "@(#)mksh.cc  1.22  06/12/12"
30
31 /*
32  *      mksh.cc
33  *
34  *      Execute the command(s) of one Make or DMake rule
35  */
36
37 /*
38  * Included files
39  */
40 #if defined(TEAMWARE_MAKE_CMN) || defined(MAKETOOL) /* tolik */
41 #include <avo/util.h>
42 #endif
43
44 #include <mksh/dosys.h>          /* redirect_io() */
45 #include <mksh/misc.h>         /* retmem() */
46 #include <mksh/mksh.h>
47 #include <mksdmsil8n/mksdmsil8n.h>
48 #include <errno.h>
49 #include <signal.h>
50
51 #ifdef HP_UX
52 extern void (*sigset(int, void (*)(&__harg)))(__harg);
53 #endif
54
55 /*
56  * Workaround for NFS bug. Sometimes, when running 'chdir' on a remote
57  * dmake server, it fails with "Stale NFS file handle" error.
58  * The second attempt seems to work.
59  */
60 int
61 my_chdir(char * dir) {
```

new/usr/src/cmd/make/lib/mksh/src/mksh.cc

2

```
58     int res = chdir(dir);
59     if (res != 0 && (errno == ESTALE || errno == EAGAIN)) {
60         /* Stale NFS file handle. Try again */
61         res = chdir(dir);
62     }
63     return res;
64 }
```

unchanged_portion_omitted


```

*****
4653 Wed May 20 11:10:15 2015
new/usr/src/cmd/make/lib/mksh/src/read.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)read.cc 1.11 06/12/12
27 */

29 #pragma ident      "@(#)read.cc  1.11  06/12/12"

27 /*
28  *      read.c
29  *
30  *      This file contains the makefile reader.
31  */

33 /*
34  * Included files
35  */
36 #include <mksh/misc.h>          /* retmem() */
37 #include <mksh/read.h>
38 #include <mkstdmsi18n/mkstdmsi18n.h>
39 #include <sys/uid.h>           /* read() */
40 #include <unistd.h>           /* close(), unlink(), read() */

42 #define STRING_LEN_TO_CONVERT  (8*1024)

44 /*
45  *      get_next_block_fn(source)
46  *
47  *      Will get the next block of text to read either
48  *      by popping one source bvsizEOFlock of the stack of Sources
49  *      or by reading some more from the makefile.
50  *
51  *      Return value:
52  *
53  *          The new source block to read from
54  *
55  *      Parameters:
56  *
57  *          source      The old source block
58  *
59  *      Global variables used:

```

```

58  *          file_being_read The name of the current file, error msg
59  */
60 Boolean      make_state_locked;
61 Source
62 get_next_block_fn(register Source source)
63 {
64     register off_t      to_read;
65     register int        length;
66     register size_t     num_wc_chars;
67     char                ch_save;
68     char                *ptr;
69
70     if (source == NULL) {
71         return NULL;
72     }
73     if ((source->fd < 0) ||
74         ((source->bytes_left_in_file <= 0) &&
75          (source->inp_buf_ptr >= source->inp_buf_end))) {
76         /* We can't read from the makefile, so pop the source block */
77         if (source->fd > 2) {
78             (void) close(source->fd);
79             if (make_state_lockfile != NULL) {
80                 (void) unlink(make_state_lockfile);
81                 retmem_mb(make_state_lockfile);
82                 make_state_lockfile = NULL;
83                 make_state_locked = false;
84             }
85         }
86         if (source->string.free_after_use &&
87             (source->string.buffer.start != NULL)) {
88             retmem(source->string.buffer.start);
89             source->string.buffer.start = NULL;
90         }
91         if (source->inp_buf != NULL) {
92             retmem_mb(source->inp_buf);
93             source->inp_buf = NULL;
94         }
95         source = source->previous;
96         if (source != NULL) {
97             source->error_converting = false;
98         }
99         return source;
100    }
101    if (source->bytes_left_in_file > 0) {
102        /*
103         * Read the whole makefile.
104         * Hopefully the kernel managed to prefetch the stuff.
105         */
106        to_read = source->bytes_left_in_file;
107        source->inp_buf_ptr = source->inp_buf + getmem(to_read + 1);
108        source->inp_buf_end = source->inp_buf + to_read;
109        length = read(source->fd, source->inp_buf, (unsigned int) to_read);
110        if (length != to_read) {
111            WCSTOMBS(mbs_buffer, file_being_read);
112            if (length == 0) {
113                fatal_mksh(catgets(libmkstdmsi18n_catd, 1, 140, "
114                                mbs_buffer);
115            } else {
116                fatal_mksh(catgets(libmkstdmsi18n_catd, 1, 141, "
117                                mbs_buffer,
118                                errmsg(errno));
119            }
120        }
121        *source->inp_buf_end = nul_char;
122        source->bytes_left_in_file = 0;
123    }

```

```
124     /*
125     * Try to convert the next piece.
126     */
127     ptr = source->inp_buf_ptr + STRING_LEN_TO_CONVERT;
128     if (ptr > source->inp_buf_end) {
129         ptr = source->inp_buf_end;
130     }
131     for (num_wc_chars = 0; ptr > source->inp_buf_ptr; ptr--) {
132         ch_save = *ptr;
133         *ptr = nul_char;
134         num_wc_chars = mbstowcs(source->string.text.end,
135                                source->inp_buf_ptr,
136                                STRING_LEN_TO_CONVERT);
137         *ptr = ch_save;
138         if (num_wc_chars != (size_t)-1) {
139             break;
140         }
141     }
142     if ((int) num_wc_chars == (size_t)-1) {
143         source->error_converting = true;
144         return source;
145     }
146
147     source->error_converting = false;
148     source->inp_buf_ptr = ptr;
149     source->string.text.end += num_wc_chars;
150     *source->string.text.end = 0;
151
152     if (source->inp_buf_ptr >= source->inp_buf_end) {
153         if (*(source->string.text.end - 1) != (int) newline_char) {
154             WCSTOMBS(mbs_buffer, file_being_read);
155             warning_mksh(catgets(libmksdmsi18n_catd, 1, 142, "newlin
156                                mbs_buffer);
157             *source->string.text.end++ = (int) newline_char;
158             *source->string.text.end = (int) nul_char;
159             *source->string.buffer.end++;
160         }
161         if (source->inp_buf != NULL) {
162             retmem_mb(source->inp_buf);
163             source->inp_buf = NULL;
164         }
165     }
166     return source;
167 }
168 }
```

unchanged portion omitted

new/usr/src/cmd/make/lib/vroot/src/access.cc

1

1355 Wed May 20 11:10:15 2015

new/usr/src/cmd/make/lib/vroot/src/access.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)access.cc 1.4 06/12/12
27 */
29 #pragma ident "@(#)access.cc 1.4 06/12/12"
27 #include <unistd.h>
28 #include <vroot/vroot.h>
29 #include <vroot/args.h>
31 static int access_thunk(char *path)
32 {
33     vroot_result= access(path, vroot_args.access.mode);
34     return((vroot_result == 0) || (errno != ENOENT));
35 }
_____unchanged_portion_omitted_
```

new/usr/src/cmd/make/lib/vroot/src/args.cc

1

1019 Wed May 20 11:10:15 2015

new/usr/src/cmd/make/lib/vroot/src/args.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)args.cc 1.3 06/12/12
27 */
29 #pragma ident "@(#)args.cc 1.3 06/12/12"
27 #include <vroot/vroot.h>
28 #include <vroot/args.h>
30 union Args      vroot_args;
31 int             vroot_result;
```

new/usr/src/cmd/make/lib/vroot/src/chdir.cc

1

1263 Wed May 20 11:10:16 2015

new/usr/src/cmd/make/lib/vroot/src/chdir.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)chdir.cc 1.4 06/12/12
27  */
28
29 #pragma ident      "@(#)chdir.cc  1.4   06/12/12"
30
31 #include <unistd.h>
32 #include <vroot/vroot.h>
33 #include <vroot/args.h>
34
35 static int      chdir_thunk(char *path)
36 {
37     vroot_result= chdir(path);
38     return(vroot_result == 0);
39 }
_____unchanged_portion_omitted_
```

new/usr/src/cmd/make/lib/vroot/src/chmod.cc

1

1402 Wed May 20 11:10:16 2015

new/usr/src/cmd/make/lib/vroot/src/chmod.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)chmod.cc 1.4 06/12/12
27  */
28 #pragma ident      "@(#)chmod.cc  1.4   06/12/12"
29
30 #include <sys/types.h>
31 #include <sys/stat.h>
32
33 extern int chmod(const char *path, mode_t mode);
34
35 #include <vroot/vroot.h>
36 #include <vroot/args.h>
37
38 static int      chmod_thunk(char *path)
39 {
40     {
41         vroot_result= chmod(path, vroot_args.chmod.mode);
42         return(vroot_result == 0);
43     }
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/chown.cc

1

1480 Wed May 20 11:10:17 2015

new/usr/src/cmd/make/lib/vroot/src/chown.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)chown.cc 1.4 06/12/12
27  */
28
29 #pragma ident      "@(#)chown.cc  1.4   06/12/12"
30
31
32 #include <unistd.h>
33 #include <sys/types.h>
34
35 extern int chown(const char *path, uid_t owner, gid_t group);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      chown_thunk(char *path)
41 {
42     vroot_result= chown(path, vroot_args.chown.user, vroot_args.chown.group)
43     return(vroot_result == 0);
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/chroot.cc

1

1306 Wed May 20 11:10:17 2015

new/usr/src/cmd/make/lib/vroot/src/chroot.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)chroot.cc 1.4 06/12/12
27 */
29 #pragma ident "@(#)chroot.cc 1.4 06/12/12"
27 #include <unistd.h>
29 extern int chroot(const char *path);
31 #include <vroot/vroot.h>
32 #include <vroot/args.h>
34 static int chroot_thunk(char *path)
35 {
36     vroot_result= chroot(path);
37     return(vroot_result == 0);
38 }
unchanged_portion_omitted
```


new/usr/src/cmd/make/lib/vroot/src/creat.cc

1

1422 Wed May 20 11:10:18 2015

new/usr/src/cmd/make/lib/vroot/src/creat.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)creat.cc 1.4 06/12/12
27 */
28
29 #pragma ident      "@(#)creat.cc  1.4   06/12/12"
30
31 #include <sys/types.h>
32 #include <sys/stat.h>
33 #include <fcntl.h>
34
35 extern int creat(const char *path, mode_t mode);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      creat_thunk(char *path)
41 {
42     vroot_result= creat(path, vroot_args.creat.mode);
43     return(vroot_result >= 0);
44 }
45
46 _____unchanged_portion_omitted_____
```

new/usr/src/cmd/make/lib/vroot/src/execve.cc

1

1527 Wed May 20 11:10:18 2015

new/usr/src/cmd/make/lib/vroot/src/execve.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)execve.cc 1.4 06/12/12
27  */
28
29 #pragma ident      "@(#)execve.cc 1.4      06/12/12"
29
30
31 #include <unistd.h>
32
33 extern int execve (const char *path, char *const argv[], char *const envp[]);
34
35 #include <root/vroot.h>
36 #include <root/args.h>
37
38 static int      execve_thunk(char *path)
39 {
40     execve(path, vroot_args.execve.argv, vroot_args.execve.environ);
41     switch (errno) {
42         case ETXTBSY:
43             case ENOEXEC: return 1;
44         default: return 0;
45     }
46 }
47
48 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/libvroot.msg

1

1224 Wed May 20 11:10:19 2015

new/usr/src/cmd/make/lib/vroot/src/libvroot.msg

make: remove SCCS ident stuff

2 \$quote "

5 \$set 1

6 \$

7 \$ CDDL HEADER START

8 \$

9 \$ The contents of this file are subject to the terms of the
10 \$ Common Development and Distribution License (the "License").
11 \$ You may not use this file except in compliance with the License.
12 \$
13 \$ You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
14 \$ or <http://www.opensolaris.org/os/licensing>.
15 \$ See the License for the specific language governing permissions
16 \$ and limitations under the License.

17 \$

18 \$ When distributing Covered Code, include this CDDL HEADER in each
19 \$ file and include the License file at usr/src/OPENSOLARIS.LICENSE.
20 \$ If applicable, add the following below this CDDL HEADER, with the
21 \$ fields enclosed by brackets "[]" replaced with your own identifying
22 \$ information: Portions Copyright [yyyy] [name of copyright owner]

23 \$

24 \$ CDDL HEADER END

25 \$

26 \$ Copyright 1996 Sun Microsystems, Inc. All rights reserved.

27 \$ Use is subject to license terms.

28 \$

29 \$ @(#)libvroot.msg 1.2 06/12/12

30 \$

29 142 "file_lock: file %s is already locked.\n"
30 143 "file_lock: will periodically check the lockfile %s for two minutes.\n"
31 144 "Current working directory %s\n"
32 145 "Could not lock file '%s'; "
33 146 " failed - "
34 147 "Couldn't write to %s"
35 148 "Cannot open '%s' for writing\n"
36 149 "Cannot open %s for writing\n"

new/usr/src/cmd/make/lib/vroot/src/lock.cc

1

```
*****
5511 Wed May 20 11:10:19 2015
new/usr/src/cmd/make/lib/vroot/src/lock.cc
make: remove SCCS ident stuff
*****
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)lock.cc 1.17 06/12/12
27  */
28
29 #pragma ident      "@(#)lock.cc  1.17  06/12/12"
30
31 #include <avo/intl.h>  /* for NOCATGETS */
32 #include <stdio.h>
33 #include <stdlib.h>
34 #include <string.h>
35 #include <sys/errno.h>
36 #include <sys/param.h>
37 #include <sys/stat.h>
38 #include <sys/types.h>
39 #include <unistd.h>
40 #include <vroot/vroot.h>
41 #include <mksdms18n/mksdms18n.h>
42 #include <signal.h>
43 #include <errno.h>          /* errno */
44
45 #if !defined(linux)
46 extern char      *sys_errlist[];
47 extern int       sys_nerr;
48 #endif
49
50 static void      file_lock_error(char *msg, char *file, char *str, int ar
51
52 #define BLOCK_INTERRUPTS sigfillset(&newset) ; \
53 sigprocmask(SIG_SETMASK, &newset, &oldset)
54
55 #define UNBLOCK_INTERRUPTS \
56 sigprocmask(SIG_SETMASK, &oldset, &newset)
57
58 /*
59  * This code stolen from the NSE library and changed to not depend
60  * upon any NSE routines or header files.
61  */
```

new/usr/src/cmd/make/lib/vroot/src/lock.cc

2

```
57 * Simple file locking.
58 * Create a symlink to a file.  The "test and set" will be
59 * atomic as creating the symlink provides both functions.
60 *
61 * The timeout value specifies how long to wait for stale locks
62 * to disappear.  If the lock is more than 'timeout' seconds old
63 * then it is ok to blow it away.  This part has a small window
64 * of vulnerability as the operations of testing the time,
65 * removing the lock and creating a new one are not atomic.
66 * It would be possible for two processes to both decide to blow
67 * away the lock and then have process A remove the lock and establish
68 * its own, and then then have process B remove the lock which accidentally
69 * removes A's lock rather than the stale one.
70 *
71 * A further complication is with the NFS.  If the file in question is
72 * being served by an NFS server, then its time is set by that server.
73 * We can not use the time on the client machine to check for a stale
74 * lock.  Therefore, a temp file on the server is created to get
75 * the servers current time.
76 *
77 * Returns an error message.  NULL return means the lock was obtained.
78 *
79 * 12/6/91 Added the parameter "file_locked".  Before this parameter
80 * was added, the calling procedure would have to wait for file_lock()
81 * to return before it sets the flag.  If the user interrupted "make"
82 * between the time the lock was acquired and the time file_lock()
83 * returns, make wouldn't know that the file has been locked, and therefore
84 * it wouldn't remove the lock.  Setting the flag right after locking the file
85 * makes this window much smaller.
86 */
87
88 int
89 file_lock(char *name, char *lockname, int *file_locked, int timeout)
90 {
91     int          counter = 0;
92     static char  msg[MAXPATHLEN+1];
93     int          printed_warning = 0;
94     int          r;
95     struct stat  statb;
96     sigset_t     newset;
97     sigset_t     oldset;
98
99     *file_locked = 0;
100    if (timeout <= 0) {
101        timeout = 120;
102    }
103    for (;;) {
104        BLOCK_INTERRUPTS;
105        r = symlink(name, lockname);
106        if (r == 0) {
107            *file_locked = 1;
108            UNBLOCK_INTERRUPTS;
109            return 0; /* success */
110        }
111        UNBLOCK_INTERRUPTS;
112
113        if (errno != EEXIST) {
114            file_lock_error(msg, name, NOCATGETS("symlink(%s, %s)"),
115                (int) name, (int) lockname);
116            fprintf(stderr, "%s", msg);
117            return errno;
118        }
119
120        counter = 0;
121        for (;;) {
122            sleep(1);
```

```
123         r = lstat(lockname, &statb);
124         if (r == -1) {
125             /*
126              * The lock must have just gone away - try
127              * again.
128              */
129             break;
130         }
131
132         if ((counter > 5) && (!printed_warning)) {
133             /* Print waiting message after 5 secs */
134 #if defined(SUN5_0) || defined(HP_UX) || defined(linux)
135             (void) getcwd(msg, MAXPATHLEN);
136 #else
137             (void) getwd(msg);
138 #endif
139             fprintf(stderr,
140                 catgets(libmksdmsil8n_catd, 1, 162, "fil
141                 name);
142             fprintf(stderr,
143                 catgets(libmksdmsil8n_catd, 1, 163, "fil
144                 lockname);
145             fprintf(stderr,
146                 catgets(libmksdmsil8n_catd, 1, 144, "Cur
147                 msg);
148
149             printed_warning = 1;
150         }
151
152         if (++counter > timeout) {
153             /*
154              * Waited enough - return an error..
155              */
156             return EEXIST;
157         }
158     }
159     /* NOTREACHED */
160 }
161 }
162 }
163 }
164 }
165 }
166 }
167 }
168 }
169 }
170 }
171 }
172 }
173 }
174 }
175 }
176 }
177 }
178 }
179 }
180 }
181 }
182 }
183 }
184 }
185 }
186 }
187 }
188 }
189 }
190 }
191 }
192 }
193 }
194 }
195 }
196 }
197 }
198 }
199 }
200 }
201 }
202 }
203 }
204 }
205 }
206 }
207 }
208 }
209 }
210 }
211 }
212 }
213 }
214 }
215 }
216 }
217 }
218 }
219 }
220 }
221 }
222 }
223 }
224 }
225 }
226 }
227 }
228 }
229 }
230 }
231 }
232 }
233 }
234 }
235 }
236 }
237 }
238 }
239 }
240 }
241 }
242 }
243 }
244 }
245 }
246 }
247 }
248 }
249 }
250 }
251 }
252 }
253 }
254 }
255 }
256 }
257 }
258 }
259 }
260 }
261 }
262 }
263 }
264 }
265 }
266 }
267 }
268 }
269 }
270 }
271 }
272 }
273 }
274 }
275 }
276 }
277 }
278 }
279 }
280 }
281 }
282 }
283 }
284 }
285 }
286 }
287 }
288 }
289 }
290 }
291 }
292 }
293 }
294 }
295 }
296 }
297 }
298 }
299 }
300 }
301 }
302 }
303 }
304 }
305 }
306 }
307 }
308 }
309 }
310 }
311 }
312 }
313 }
314 }
315 }
316 }
317 }
318 }
319 }
320 }
321 }
322 }
323 }
324 }
325 }
326 }
327 }
328 }
329 }
330 }
331 }
332 }
333 }
334 }
335 }
336 }
337 }
338 }
339 }
340 }
341 }
342 }
343 }
344 }
345 }
346 }
347 }
348 }
349 }
350 }
351 }
352 }
353 }
354 }
355 }
356 }
357 }
358 }
359 }
360 }
361 }
362 }
363 }
364 }
365 }
366 }
367 }
368 }
369 }
370 }
371 }
372 }
373 }
374 }
375 }
376 }
377 }
378 }
379 }
380 }
381 }
382 }
383 }
384 }
385 }
386 }
387 }
388 }
389 }
390 }
391 }
392 }
393 }
394 }
395 }
396 }
397 }
398 }
399 }
400 }
401 }
402 }
403 }
404 }
405 }
406 }
407 }
408 }
409 }
410 }
411 }
412 }
413 }
414 }
415 }
416 }
417 }
418 }
419 }
420 }
421 }
422 }
423 }
424 }
425 }
426 }
427 }
428 }
429 }
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }
454 }
455 }
456 }
457 }
458 }
459 }
460 }
461 }
462 }
463 }
464 }
465 }
466 }
467 }
468 }
469 }
470 }
471 }
472 }
473 }
474 }
475 }
476 }
477 }
478 }
479 }
480 }
481 }
482 }
483 }
484 }
485 }
486 }
487 }
488 }
489 }
490 }
491 }
492 }
493 }
494 }
495 }
496 }
497 }
498 }
499 }
500 }
501 }
502 }
503 }
504 }
505 }
506 }
507 }
508 }
509 }
510 }
511 }
512 }
513 }
514 }
515 }
516 }
517 }
518 }
519 }
520 }
521 }
522 }
523 }
524 }
525 }
526 }
527 }
528 }
529 }
530 }
531 }
532 }
533 }
534 }
535 }
536 }
537 }
538 }
539 }
540 }
541 }
542 }
543 }
544 }
545 }
546 }
547 }
548 }
549 }
550 }
551 }
552 }
553 }
554 }
555 }
556 }
557 }
558 }
559 }
560 }
561 }
562 }
563 }
564 }
565 }
566 }
567 }
568 }
569 }
570 }
571 }
572 }
573 }
574 }
575 }
576 }
577 }
578 }
579 }
580 }
581 }
582 }
583 }
584 }
585 }
586 }
587 }
588 }
589 }
590 }
591 }
592 }
593 }
594 }
595 }
596 }
597 }
598 }
599 }
600 }
601 }
602 }
603 }
604 }
605 }
606 }
607 }
608 }
609 }
610 }
611 }
612 }
613 }
614 }
615 }
616 }
617 }
618 }
619 }
620 }
621 }
622 }
623 }
624 }
625 }
626 }
627 }
628 }
629 }
630 }
631 }
632 }
633 }
634 }
635 }
636 }
637 }
638 }
639 }
640 }
641 }
642 }
643 }
644 }
645 }
646 }
647 }
648 }
649 }
650 }
651 }
652 }
653 }
654 }
655 }
656 }
657 }
658 }
659 }
660 }
661 }
662 }
663 }
664 }
665 }
666 }
667 }
668 }
669 }
670 }
671 }
672 }
673 }
674 }
675 }
676 }
677 }
678 }
679 }
680 }
681 }
682 }
683 }
684 }
685 }
686 }
687 }
688 }
689 }
690 }
691 }
692 }
693 }
694 }
695 }
696 }
697 }
698 }
699 }
700 }
701 }
702 }
703 }
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }
721 }
722 }
723 }
724 }
725 }
726 }
727 }
728 }
729 }
730 }
731 }
732 }
733 }
734 }
735 }
736 }
737 }
738 }
739 }
740 }
741 }
742 }
743 }
744 }
745 }
746 }
747 }
748 }
749 }
750 }
751 }
752 }
753 }
754 }
755 }
756 }
757 }
758 }
759 }
760 }
761 }
762 }
763 }
764 }
765 }
766 }
767 }
768 }
769 }
770 }
771 }
772 }
773 }
774 }
775 }
776 }
777 }
778 }
779 }
780 }
781 }
782 }
783 }
784 }
785 }
786 }
787 }
788 }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799 }
800 }
801 }
802 }
803 }
804 }
805 }
806 }
807 }
808 }
809 }
810 }
811 }
812 }
813 }
814 }
815 }
816 }
817 }
818 }
819 }
820 }
821 }
822 }
823 }
824 }
825 }
826 }
827 }
828 }
829 }
830 }
831 }
832 }
833 }
834 }
835 }
836 }
837 }
838 }
839 }
840 }
841 }
842 }
843 }
844 }
845 }
846 }
847 }
848 }
849 }
850 }
851 }
852 }
853 }
854 }
855 }
856 }
857 }
858 }
859 }
860 }
861 }
862 }
863 }
864 }
865 }
866 }
867 }
868 }
869 }
870 }
871 }
872 }
873 }
874 }
875 }
876 }
877 }
878 }
879 }
880 }
881 }
882 }
883 }
884 }
885 }
886 }
887 }
888 }
889 }
890 }
891 }
892 }
893 }
894 }
895 }
896 }
897 }
898 }
899 }
900 }
901 }
902 }
903 }
904 }
905 }
906 }
907 }
908 }
909 }
910 }
911 }
912 }
913 }
914 }
915 }
916 }
917 }
918 }
919 }
920 }
921 }
922 }
923 }
924 }
925 }
926 }
927 }
928 }
929 }
930 }
931 }
932 }
933 }
934 }
935 }
936 }
937 }
938 }
939 }
940 }
941 }
942 }
943 }
944 }
945 }
946 }
947 }
948 }
949 }
950 }
951 }
952 }
953 }
954 }
955 }
956 }
957 }
958 }
959 }
960 }
961 }
962 }
963 }
964 }
965 }
966 }
967 }
968 }
969 }
970 }
971 }
972 }
973 }
974 }
975 }
976 }
977 }
978 }
979 }
980 }
981 }
982 }
983 }
984 }
985 }
986 }
987 }
988 }
989 }
990 }
991 }
992 }
993 }
994 }
995 }
996 }
997 }
998 }
999 }
1000 }
```

unchanged_portion_omitted

new/usr/src/cmd/make/lib/vroot/src/lstat.cc

1

1424 Wed May 20 11:10:20 2015

new/usr/src/cmd/make/lib/vroot/src/lstat.cc

make: remove SCCS ident stuff

```
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 1998 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)lstat.cc 1.6 06/12/12
27 */
28
29 #pragma ident "@(#)lstat.cc 1.6 06/12/12"
30
31
32 #include <sys/types.h>
33 #include <sys/stat.h>
34
35 extern int lstat(const char *path, struct stat *buf);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int lstat_thunk(char *path)
41 {
42     vroot_result= lstat(path, vroot_args.lstat.buffer);
43     return(vroot_result == 0);
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/mkdir.cc

1

1403 Wed May 20 11:10:20 2015

new/usr/src/cmd/make/lib/vroot/src/mkdir.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)mkdir.cc 1.4 06/12/12
27  */
28
29 #pragma ident      "@(#)mkdir.cc  1.4   06/12/12"
30
31
32 #include <sys/types.h>
33 #include <sys/stat.h>
34
35 extern int mkdir(const char *path, mode_t mode);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      mkdir_thunk(char *path)
41 {
42     vroot_result= mkdir(path, vroot_args.mkdir.mode);
43     return(vroot_result == 0);
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/mount.cc

1

1513 Wed May 20 11:10:20 2015

new/usr/src/cmd/make/lib/vroot/src/mount.cc

make: remove SCCS ident stuff

```
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 1995 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)mount.cc 1.5 06/12/12
27  */
28 #pragma ident    "@(#)mount.cc 1.5 06/12/12"
29
30 #include <sys/types.h>
31 #include <sys/mount.h>
32
33 #ifndef HP_UX
34 extern int mount(const char *spec, const char *dir, int mflag, ...);
35 #endif
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      mount_thunk(char *path)
41 {
42     vroot_result= mount(path, vroot_args.mount.name, vroot_args.mount.mode);
43     return(vroot_result == 0);
44 }
45
46 _____unchanged_portion_omitted_____
```


new/usr/src/cmd/make/lib/vroot/src/open.cc

1

1535 Wed May 20 11:10:21 2015

new/usr/src/cmd/make/lib/vroot/src/open.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)open.cc 1.4 06/12/12
27 */
28
29 #pragma ident      "@(#)open.cc  1.4   06/12/12"
30
31 #include <sys/types.h>
32 #include <sys/stat.h>
33 #include <fcntl.h>
34
35 extern int open(const char *path, int oflag, ...);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      open_thunk(char *path)
41 {
42     vroot_result= open(path, vroot_args.open.flags, vroot_args.open.mode);
43     return(vroot_result >= 0);
44 }
45
46 _____unchanged_portion_omitted_____
```

new/usr/src/cmd/make/lib/vroot/src/readlink.cc

1

1518 Wed May 20 11:10:21 2015

new/usr/src/cmd/make/lib/vroot/src/readlink.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)readlink.cc 1.4 06/12/12
27  */
28
29 #pragma ident    "@(#)readlink.cc      1.4      06/12/12"
29
30
31 #include <unistd.h>
32
33 extern int readlink(const char *path, void *buf, size_t bufsiz);
34
35 #include <vroot/vroot.h>
36 #include <vroot/args.h>
37
38 static int      readlink_thunk(char *path)
39 {
40     vroot_result= readlink(path, vroot_args.readlink.buffer, vroot_args.read
41     return(vroot_result >= 0);
42 }
43
44 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/report.cc

1

9528 Wed May 20 11:10:22 2015

new/usr/src/cmd/make/lib/vroot/src/report.cc

make: remove SCCS ident stuff

```
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)report.cc 1.17 06/12/12
27  */
29 #pragma ident    "@(#)report.cc 1.17 06/12/12"

26 #include <stdio.h>
27 #include <stdlib.h>
28 #include <string.h>
29 #include <sys/param.h>
30 #include <sys/wait.h>
31 #include <unistd.h>

33 #include <vroot/report.h>
34 #include <vroot/vroot.h>
35 #include <mksdmsil8n/mksdmsil8n.h>
36 #include <avo/intl.h> /* for NOCATGETS */
37 #include <mk/defs.h> /* for tmpdir */

39 static FILE *report_file;
40 static FILE *command_output_fp;
41 static char *target_being_reported_for;
42 static char *search_dir;
43 static char command_output_tmpfile[30];
44 static int is_path = 0;
45 static char sfile[MAXPATHLEN];
46 extern "C" {
47 static void (*warning_ptr) (char *, ...) = (void (*) (char *, ...)) NULL;
48 }
unchanged portion omitted
```

new/usr/src/cmd/make/lib/vroot/src/rmdir.cc

1

1301 Wed May 20 11:10:23 2015

new/usr/src/cmd/make/lib/vroot/src/rmdir.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)rmdir.cc 1.4 06/12/12
27 */
28
29 #pragma ident      "@(#)rmdir.cc  1.4   06/12/12"
30
31 #include <unistd.h>
32
33 extern int rmdir(const char *path);
34
35 #include <vroot/vroot.h>
36 #include <vroot/args.h>
37
38 static int      rmdir_thunk(char *path)
39 {
40     vroot_result= rmdir(path);
41     return(vroot_result == 0);
42 }
43
44 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/setenv.cc

1

```
*****
1846 Wed May 20 11:10:23 2015
new/usr/src/cmd/make/lib/vroot/src/setenv.cc
make: remove SCCS ident stuff
*****
```

```
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 1994 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)setenv.cc 1.6 06/12/12
27 */
28
29 #pragma ident "@(#)setenv.cc 1.6 06/12/12"
30
31 #include <stdlib.h>
32 #include <string.h>
33 #include <unistd.h>
34
35 extern char **environ;
36
37 static short setenv_made_new_vector= 0;
38
39 char *setenv(char *name, char *value)
40 {
41     char *p= NULL, **q;
42     int length= 0, vl;
43
44     if ((p= getenv(name)) == NULL) { /* Allocate new vector */
45         for (q= environ; *q != NULL; q++, length++);
46         q= (char **)malloc((unsigned)(sizeof(char *)*(length+2)));
47         memcpy(((char *)q)+sizeof(char *), (char *)environ, sizeof(char *)*length);
48         if (setenv_made_new_vector++)
49             free((char *)environ);
50         length= strlen(name);
51         environ= q;
52     } else { /* Find old slot */
53         length= strlen(name);
54         for (q= environ; *q != NULL; q++)
55             if (!strcmp(*q, name, length))
56                 break;
57     }
58
59     vl= strlen(value);
60     if (!p || (length+vl+1 > strlen(p)))
61         *q= p= (char *) malloc((unsigned)(length+vl+2));
62     else
63         p= *q;
64     (void)strcpy(p, name); p+= length;
65 }
```

new/usr/src/cmd/make/lib/vroot/src/setenv.cc

2

```
58     *p++= '=';
59     (void)strcpy(p, value);
60     return(value);
61 }
62
63 _____
64 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/stat.cc

1

1417 Wed May 20 11:10:24 2015

new/usr/src/cmd/make/lib/vroot/src/stat.cc

make: remove SCCS ident stuff

```
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 1998 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)stat.cc 1.6 06/12/12
27 */
28
29 #pragma ident      "@(#)stat.cc  1.6   06/12/12"
30
31
32 #include <sys/types.h>
33 #include <sys/stat.h>
34
35 extern int stat(const char *path, struct stat *buf);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      stat_thunk(char *path)
41 {
42     vroot_result= stat(path, vroot_args.stat.buffer);
43     return(vroot_result == 0);
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/statfs.cc

1

1374 Wed May 20 11:10:24 2015

new/usr/src/cmd/make/lib/vroot/src/statfs.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)statfs.cc 1.5 06/12/12
27 */
28 #pragma ident "@(#)statfs.cc 1.5 06/12/12"
29
30 #ifndef SUN5_0
31
32 #include <vroot/vroot.h>
33 #include <vroot/args.h>
34 #include <sys/vfs.h>
35
36 static int statfs_thunk(char *path)
37 {
38     vroot_result= statfs(path, vroot_args.statfs.buffer);
39     return(vroot_result == 0);
40 }
41
42 _____ unchanged_portion_omitted_
```

new/usr/src/cmd/make/lib/vroot/src/truncate.cc

1

1407 Wed May 20 11:10:24 2015

new/usr/src/cmd/make/lib/vroot/src/truncate.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26  * @(#)truncate.cc 1.4 06/12/12
27 */
28
29 #pragma ident      "@(#)truncate.cc      1.4      06/12/12"
30
31 #include <unistd.h>
32
33 extern int truncate(const char *path, off_t length);
34
35 #include <vroot/vroot.h>
36 #include <vroot/args.h>
37
38 static int      truncate_thunk(char *path)
39 {
40     vroot_result= truncate(path, vroot_args.truncate.length);
41     return(vroot_result == 0);
42 }
43
44 unchanged_portion_omitted
```



```
new/usr/src/cmd/make/lib/vroot/src/unlink.cc
```

1

```
*****
```

```
1306 Wed May 20 11:10:25 2015
```

```
new/usr/src/cmd/make/lib/vroot/src/unlink.cc
```

```
make: remove SCCS ident stuff
```

```
*****
```

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)unlink.cc 1.4 06/12/12
27 */
29 #pragma ident "@(#)unlink.cc 1.4 06/12/12"
27 #include <unistd.h>
29 extern int unlink(const char *path);
31 #include <vroot/vroot.h>
32 #include <vroot/args.h>
34 static int unlink_thunk(char *path)
35 {
36     vroot_result= unlink(path);
37     return(vroot_result == 0);
38 }
unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/unmount.cc

1

1305 Wed May 20 11:10:25 2015

new/usr/src/cmd/make/lib/vroot/src/unmount.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)unmount.cc 1.5 06/12/12
27 */
28
29 #pragma ident    "@(#)unmount.cc 1.5    06/12/12"
30
31
32 #ifndef SUN5_0
33 #include <vroot/vroot.h>
34 #include <vroot/args.h>
35
36 extern int      unmount(char *name);
37
38 static int      unmount_thunk(char *path)
39 {
40     vroot_result= unmount(path);
41     return(vroot_result == 0);
42 }
43
44 _____
45 unchanged portion omitted
```

new/usr/src/cmd/make/lib/vroot/src/utimes.cc

1

1423 Wed May 20 11:10:26 2015

new/usr/src/cmd/make/lib/vroot/src/utimes.cc

make: remove SCCS ident stuff

```
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 1993 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)utimes.cc 1.4 06/12/12
27 */
28
29 #pragma ident      "@(#)utimes.cc 1.4      06/12/12"
30
31
32 #include <sys/types.h>
33 #include <sys/time.h>
34
35 extern int utimes(char *file, struct timeval *tvp);
36
37 #include <vroot/vroot.h>
38 #include <vroot/args.h>
39
40 static int      utimes_thunk(char *path)
41 {
42     vroot_result= utimes(path, vroot_args.utimes.time);
43     return(vroot_result == 0);
44 }
45
46 unchanged_portion_omitted
```

new/usr/src/cmd/make/lib/vroot/src/vroot.cc

1

9163 Wed May 20 11:10:26 2015

new/usr/src/cmd/make/lib/vroot/src/vroot.cc

make: remove SCCS ident stuff

```
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 2004 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25 /*
26 * @(#)vroot.cc 1.11 06/12/12
27 */
29 #pragma ident      "@(#)vroot.cc  1.11  06/12/12"

27 #include <stdlib.h>
28 #include <string.h>

30 #include <vroot/vroot.h>
31 #include <vroot/args.h>

33 #include <string.h>
34 #include <sys/param.h>
35 #include <sys/file.h>

37 #include <avo/intl.h>  /* for NOCATGETS */

39 typedef struct {
40     short      init;
41     pathpt     vector;
42     char       *env_var;
43 } vroot_patht;
unchanged_portion_omitted
```