

new/usr/src/tools/env/illumos.sh

1

```
*****
10378 Fri Dec 14 15:11:22 2018
new/usr/src/tools/env/illumos.sh
10062 illumos.sh could be re-organized
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 # Copyright (c) 2005, 2010, Oracle and/or its affiliates. All rights reserved.
22 # Copyright 2015 Nexenta Systems, Inc. All rights reserved.
23 # Copyright 2012 Joshua M. Clulow <josh@sysmgr.org>
24 # Copyright 2015, OmniTI Computer Consulting, Inc. All rights reserved.
25 # Copyright 2018 OmniOS Community Edition (OmniOSce) Association.
26 # Copyright (c) 2018, Joyent, Inc.
27 #
28 # - This file is sourced by "bldenv.sh" and "nightly.sh" and should not
29 #   be executed directly.
30 # - This script is only interpreted by ksh93 and explicitly allows the
31 #   use of ksh93 language extensions.
32 #
33 # -----
34 # Parameters you are likely to want to change
35 # -----
36 #
37 #
38 # Configuration variables for the runtime environment of the nightly
39 # build script and other tools for construction and packaging of
40 # releases.
41 # This example is suitable for building an illumos workspace, which
42 # will contain the resulting archives. It is based off the onnv
43 # release. It sets NIGHTLY_OPTIONS to make nightly do:
44 #   DEBUG build only (-D, -F)
45 #   do not bringover from the parent (-n)
46 #   runs 'make check' (-C)
47 #   checks for new interfaces in libraries (-A)
48 #   sends mail on completion (-m and the MAILTO variable)
49 #   creates packages for PIT/RE (-p)
50 #   checks for changes in ELF runpaths (-r)
51 #   build and use this workspace's tools in $SRC/tools (-t)
52 #
53 # - This file is sourced by "bldenv.sh" and "nightly.sh" and should not
54 #   be executed directly.
55 # - This script is only interpreted by ksh93 and explicitly allows the
56 #   use of ksh93 language extensions.
57 #
58 # export NIGHTLY_OPTIONS='-FnCDAmprt'
59 #
60 # Some scripts optionally send mail messages to MAILTO.
61 #export MAILTO=
```

new/usr/src/tools/env/illumos.sh

2

```
51 # CODEMGR_WS - where is your workspace at
52 #export CODEMGR_WS="$HOME/ws/illumos-gate"
53 export CODEMGR_WS="`git rev-parse --show-toplevel`"
54 # Compilers may be specified using the following variables:
55 # PRIMARY_CC - primary C compiler
56 # PRIMARY_CCC - primary C++ compiler
57 #
58 # SHADOW_CCS - list of shadow C compilers
59 # SHADOW_CCCS - list of shadow C++ compilers
60 #
61 # Each entry has the form <name>,<path to binary>,<style> where name is a
62 # free-form name (possibly used in the makefiles to guard options), path is
63 # the path to the executable. style is the 'style' of command line taken by
64 # the compiler, currently either gnu (or gcc) or sun (or cc), which is also
65 # used by Makefiles to guard options.
66 #
67 # __SUNC and __GNUC must still be set to reflect the style of the primary
68 # compiler (and to influence the default primary, otherwise)
69 #
70 # for example:
71 # export PRIMARY_CC=gcc4,/opt/gcc/4.4.4/bin/gcc,gnu
72 # export PRIMARY_CCC=gcc4,/opt/gcc/4.4.4/bin/g++,gnu
73 # export SHADOW_CCS=studio12,/opt/SUNWspr0/bin/cc,sun
74 # export SHADOW_CCCS=studio12,/opt/SUNWspr0/bin/CC,sun
75 #
76 # There can be several space-separated entries in SHADOW_* to run multiple
77 # shadow compilers.
78 #
79 # To disable shadow compilation, unset SHADOW_* or set them to the empty string.
80 #
81 export SHADOW_CCS=gcc7,/usr/gcc/7/bin/gcc,gnu
82 export SHADOW_CCCS=gcc7,/usr/gcc/7/bin/g++,gnu
83 #
84 # Comment this out to disable support for SMB printing, i.e. if you
85 # don't want to bother providing the CUPS headers this needs.
86 export ENABLE_SMB_PRINTING=
87 #
88 # If your distro uses certain versions of Perl, make sure either Makefile.master
89 # contains your new defaults OR your .env file sets them.
90 # These are how you would override for building on OmniOS r151028, for example.
91 #export PERL_VERSION=5.28
92 #export PERL_ARCH=i86pc-solaris-thread-multi-64int
93 #export PERL_PKGVERS=
94 #
95 # If your distro uses certain versions of Python, make sure either
96 # Makefile.master contains your new defaults OR your .env file sets them.
97 #export PYTHON_VERSION=2.7
98 #export PYTHON_PKGVERS=-27
99 #export PYTHON_SUFFIX=
100 #export PYTHON3_VERSION=3.5
101 #export PYTHON3_PKGVERS=-35
102 #export PYTHON3_SUFFIX=m
103 #
104 # To disable building with either Python2 or Python 3 (or both), uncomment
105 # these lines:
106 #export BUILDPY2='#'
107 #export BUILDPY3='#'
108 #
109 # Set if your distribution has different package versioning
110 #export PKGVERS_BRANCH=2018.0.0.17900
111 #
112 # Skip Java 8 builds on distributions that don't support it
113 #export BLD_JAVA_8=
```

```

115 # POST_NIGHTLY can be any command to be run at the end of nightly. See
116 # nightly(1) for interactions between environment variables and this command.
117 #POST_NIGHTLY=

119 # -----
120 # You are less likely to need to modify parameters below.
121 # -----

123 # Maximum number of dmake jobs. The recommended number is 2 + NCPUS,
124 # where NCPUS is the number of logical CPUs on your build system.
125 function maxjobs
126 {
127     nameref maxjobs=$1
128     integer ncpu
129     integer -r min_mem_per_job=512 # minimum amount of memory for a job

131     ncpu=$(builtin getconf ; getconf 'NPROCESSORS_ONLN')
132     (( maxjobs=ncpu + 2 ))

134     # Throttle number of parallel jobs launched by dmake to a value which
135     # gurantees that all jobs have enough memory. This was added to avoid
136     # excessive paging/swapping in cases of virtual machine installations
137     # which have lots of CPUs but not enough memory assigned to handle
138     # that many parallel jobs
139     if [[ $(/usr/sbin/prtconf 2>' /dev/null') == ~(E)Memory\ size:\ ([[[:digit
140         integer max_jobs_per_memory # parallel jobs which fit into physi
141         integer physical_memory # physical memory installed

143     # The array ".sh.match" contains the contents of capturing
144     # brackets in the last regex, .sh.match[1] will contain
145     # the value matched by ([[[:digit:]]+), i.e. the amount of
146     # memory installed
147     physical_memory="10#${.sh.match[1]}"

149     ((
150         max_jobs_per_memory=round(physical_memory/min_mem_per_jo
151         maxjobs=fmax(2, fmin(maxjobs, max_jobs_per_memory))
152     ))
153     fi

155     return 0
156 }

158 maxjobs DMAKE_MAX_JOBS # "DMAKE_MAX_JOBS" passed as ksh(1) name reference
159 export DMAKE_MAX_JOBS

161 # path to onbld tool binaries
162 ONBLD_BIN="/opt/onbld/bin"

164 # PARENT_WS is used to determine the parent of this workspace. This is
165 # for the options that deal with the parent workspace (such as where the
166 # proto area will go).
167 export PARENT_WS=""

169 # CLONE_WS is the workspace nightly should do a bringover from.
170 # The bringover, if any, is done as STAFFER.
171 export CLONE_WS="ssh://anonhg@hg.illumos.org/illumos-gate"

103 # The bringover, if any, is done as STAFFER.
173 # Set STAFFER to your own login as gatekeeper or developer
174 # The point is to use group "staff" and avoid referencing the parent
175 # workspace as root.
107 # Some scripts optionally send mail messages to MAILTO.
108 #
176 export STAFFER="$LOGNAME"
177 export MAILTO="${MAILTO:-$STAFFER}"

```

```

110 export MAILTO="$STAFFER"

179 # If you wish the mail messages to be From: an arbitrary address, export
180 # MAILFROM.
181 #export MAILFROM="user@example.com"

183 # The project (see project(4)) under which to run this build. If not
184 # specified, the build is simply run in a new task in the current project.
185 export BUILD_PROJECT=""

187 # You should not need to change the next three lines
188 export ATLOG="$CODEMGR_WS/log"
189 export LOGFILE="$ATLOG/nightly.log"
190 export MACH="$(uname -p)"

192 #
193 # The following macro points to the closed binaries. Once illumos has
194 # totally freed itself, we can remove this reference.
195 #
196 # Location of encumbered binaries.
197 export ON_CLOSED_BINS="$CODEMGR_WS/closed"

199 # REF_PROTO_LIST - for comparing the list of stuff in your proto area
200 # with. Generally this should be left alone, since you want to see differences
201 # from your parent (the gate).
202 #
203 export REF_PROTO_LIST="$PARENT_WS/usr/src/proto_list_${MACH}"

206 export ROOT="$CODEMGR_WS/proto/root_${MACH}"
207 export SRC="$CODEMGR_WS/usr/src"
208 export MULTI_PROTO="no"

210 #
211 # build environment variables, including version info for mcs, motd,
212 # motd, uname and boot messages. Mostly you shouldn't change this except
213 # when the release slips (nah) or you move an environment file to a new
214 # release
215 #
216 export VERSION="'git describe --long --all HEAD | cut -d/ -f2-'"

218 #
219 # the RELEASE and RELEASE_DATE variables are set in Makefile.master;
220 # there might be special reasons to override them here, but that
221 # should not be the case in general
222 #
223 # export RELEASE='5.11'
224 # export RELEASE_DATE='October 2007'

226 # proto area in parent for optionally depositing a copy of headers and
227 # libraries corresponding to the protolibs target
228 # not applicable given the NIGHTLY_OPTIONS
229 #
230 export PARENT_ROOT="$PARENT_WS/proto/root_${MACH}"
231 export PARENT_TOOLS_ROOT="$PARENT_WS/usr/src/tools/proto/root_${MACH}-nd"

233 # Package creation variables. You probably shouldn't change these,
234 # either.
235 #
236 # PKGARCHIVE determines where the repository will be created.
237 #
238 # PKGPUBLISHER_REDIST controls the publisher setting for the repository.
239 #
240 export PKGARCHIVE="{CODEMGR_WS}/packages/{MACH}/nightly"
241 # export PKGPUBLISHER_REDIST='on-redist'

```

```

243 # Package manifest format version.
244 export PKGFMT_OUTPUT='v1'

246 # we want make to do as much as it can, just in case there's more than
247 # one problem.
248 export MAKEFLAGS='k'

250 # Magic variables to prevent the devpro compilers/teamware from checking
251 # for updates or sending mail back to devpro on every use.
252 export SUNW_NO_UPDATE_NOTIFY='1'
253 export UT_NO_USAGE_TRACKING='1'

255 # Build tools - don't change these unless you know what you're doing. These
256 # variables allows you to get the compilers and onbld files locally.
257 # Set BUILD_TOOLS to pull everything from one location.
258 # Alternately, you can set ONBLD_TOOLS to where you keep the contents of
259 # SUNWonbld and SPRO_ROOT to where you keep the compilers. SPRO_VROOT
260 # exists to make it easier to test new versions of the compiler.
261 export BUILD_TOOLS='/opt'
262 #export ONBLD_TOOLS='/opt/onbld'
263 export SPRO_ROOT='/opt/SUNWspro'
264 export SPRO_VROOT="$SPRO_ROOT"

199 # Compilers may be specified using the following variables:
200 # PRIMARY_CC - primary C compiler
201 # PRIMARY_CCC - primary C++ compiler
202 #
203 # SHADOW_CCS - list of shadow C compilers
204 # SHADOW_CCCS - list of shadow C++ compilers
205 #
206 # Each entry has the form <name>,<path to binary>,<style> where name is a
207 # free-form name (possibly used in the makefiles to guard options), path is
208 # the path to the executable. style is the 'style' of command line taken by
209 # the compiler, currently either gnu (or gcc) or sun (or cc), which is also
210 # used by Makefiles to guard options.
211 #
212 # __SUNC and __GNUC must still be set to reflect the style of the primary
213 # compiler (and to influence the default primary, otherwise)
214 #
215 # for example:
216 # export PRIMARY_CC=gcc4,/opt/gcc/4.4.4/bin/gcc,gnu
217 # export PRIMARY_CCC=gcc4,/opt/gcc/4.4.4/bin/g++,gnu
218 # export SHADOW_CCS=studio12,/opt/SUNWspro/bin/cc,sun
219 # export SHADOW_CCCS=studio12,/opt/SUNWspro/bin/CC,sun
220 #
221 # There can be several space-separated entries in SHADOW_* to run multiple
222 # shadow compilers.
223 #
224 # To disable shadow compilation, unset SHADOW_* or set them to the empty string.
225 #
226 export SHADOW_CCS=gcc7,/usr/gcc/7/bin/gcc,gnu
227 export SHADOW_CCCS=gcc7,/usr/gcc/7/bin/g++,gnu

266 # This goes along with lint - it is a series of the form "A [y|n]" which
267 # means "go to directory A and run 'make lint'" Then mail me (y) the
268 # difference in the lint output. 'y' should only be used if the area you're
269 # linting is actually lint clean or you'll get lots of mail.
270 # You shouldn't need to change this though.
271 #export LINTDIRS="$SRC y"

273 # Set this flag to 'n' to disable the use of 'checkpaths'. The default,
274 # if the 'N' option is not specified, is to run this test.
275 #CHECK_PATHS='y'

240 # POST_NIGHTLY can be any command to be run at the end of nightly. See
241 # nightly(1) for interactions between environment variables and this command.

```

```

242 #POST_NIGHTLY=

244 # Comment this out to disable support for SMB printing, i.e. if you
245 # don't want to bother providing the CUPS headers this needs.
246 export ENABLE_SMB_PRINTING=

248 # If your distro uses certain versions of Perl, make sure either Makefile.master
249 # contains your new defaults OR your .env file sets them.
250 # These are how you would override for building on Omnios r151028, for example.
251 #export PERL_VERSION=5.28
252 #export PERL_ARCH=i86pc-solaris-thread-multi-64int
253 #export PERL_PKGVERS=

255 # If your distro uses certain versions of Python, make sure either
256 # Makefile.master contains your new defaults OR your .env file sets them.
257 #export PYTHON_VERSION=2.7
258 #export PYTHON_PKGVERS=-27
259 #export PYTHON_SUFFIX=
260 #export PYTHON3_VERSION=3.5
261 #export PYTHON3_PKGVERS=-35
262 #export PYTHON3_SUFFIX=m

264 # To disable building with either Python2 or Python 3 (or both), uncomment
265 # these lines:
266 #export BUILDPY2='#'
267 #export BUILDPY3='#'

```