

new/usr/src/tools/scripts/nightly.sh

1

```
*****
57847 Tue Mar 6 13:59:33 2018
new/usr/src/tools/scripts/nightly.sh
9231 nightly should create a log/latest/ symlink
*****
_unchanged_portion_omitted_
995 newdirlist=

997 [ -d $CODEMGR_WS ] || newdir $CODEMGR_WS || exit 1

999 # since this script assumes the build is from full source, it nullifies
1000 # variables likely to have been set by a "ws" script; nullification
1001 # confines the search space for headers and libraries to the proto area
1002 # built from this immediate source.
1003 ENVLDLIBS1=
1004 ENVLDLIBS2=
1005 ENVLDLIBS3=
1006 ENVCPPFLAGS1=
1007 ENVCPPFLAGS2=
1008 ENVCPPFLAGS3=
1009 ENVCPPFLAGS4=
1010 PARENT_ROOT=

1012 export ENVLDLIBS3 ENVCPPFLAGS1 ENVCPPFLAGS2 ENVCPPFLAGS3 ENVCPPFLAGS4 \
1013         ENVLDLIBS1 ENVLDLIBS2 PARENT_ROOT

1015 PKGARCHIVE_ORIG=$PKGARCHIVE

1017 #
1018 # Juggle the logs and optionally send mail on completion.
1019 #

1021 function logshuffle {
1022     LLOG="$ATLOG/log.`date '+%F.%H:%M'`"
1023     if [ -f $LLOG -o -d $LLOG ]; then
1024         LLOG=$LLOG.$$
1025     fi

1027     rm -f "$ATLOG/latest" 2>/dev/null
1028     mkdir -p $LLOG
1029     mkdir $LLOG
1030     export LLOG

1031     if [ "$build_ok" = "y" ]; then
1032         mv $ATLOG/proto_list_{$MACH} $LLOG

1034         if [ -f $ATLOG/proto_list_tools_{$MACH} ]; then
1035             mv $ATLOG/proto_list_tools_{$MACH} $LLOG
1036         fi

1038         if [ -f $TMPDIR/wsdiff.results ]; then
1039             mv $TMPDIR/wsdiff.results $LLOG
1040         fi

1042         if [ -f $TMPDIR/wsdiff-nd.results ]; then
1043             mv $TMPDIR/wsdiff-nd.results $LLOG
1044         fi
1045     fi

1047     #
1048     # Now that we're about to send mail, it's time to check the noise
1049     # file. In the event that an error occurs beyond this point, it will
1050     # be recorded in the nightly.log file, but nowhere else. This would
1051     # include only errors that cause the copying of the noise log to fail
1052     # or the mail itself not to be sent.
1053     #
```

new/usr/src/tools/scripts/nightly.sh

2

```
1055     exec >>$LOGFILE 2>&1
1056     if [ -s $build_noise_file ]; then
1057         echo "\n=== Nightly build noise ===\n" |
1058             tee -a $LOGFILE >>$mail_msg_file
1059         cat $build_noise_file >>$LOGFILE
1060         cat $build_noise_file >>$mail_msg_file
1061         echo | tee -a $LOGFILE >>$mail_msg_file
1062     fi
1063     rm -f $build_noise_file

1065     case "$build_ok" in
1066         y)
1067             state=Completed
1068             ;;
1069         i)
1070             state=Interrupted
1071             ;;
1072         *)
1073             state=Failed
1074             ;;
1075     esac

1077     if [[ $state != "Interrupted" && $build_extras_ok != "y" ]]; then
1078         state=Failed
1079     fi

1081     NIGHTLY_STATUS=$state
1082     export NIGHTLY_STATUS

1084     run_hook POST_NIGHTLY $state
1085     run_hook SYS_POST_NIGHTLY $state

1087     #
1088     # mailx(1) sets From: based on the -r flag
1089     # if it is given.
1090     #
1091     mailx_r=
1092     if [[ -n "${MAILFROM}" ]]; then
1093         mailx_r="-r ${MAILFROM}"
1094     fi

1096     cat $build_time_file $build_envron_file $mail_msg_file \
1097         > ${LLOG}/mail_msg
1098     if [ "$u_FLAG" = "y" ]; then
1099         cat ${LLOG}/mail_msg | /usr/bin/mailx ${mailx_r} -s \
1100             "Nightly ${MACH} Build of 'basename ${CODEMGR_WS}' ${state}." \
1101             ${MAILTO}
1102     fi

1104     if [ "$u_FLAG" = "y" -a "$build_ok" = "y" ]; then
1105         staffer cp ${LLOG}/mail_msg $PARENT_WS/usr/src/mail_msg-{$MACH}
1106         staffer cp $LOGFILE $PARENT_WS/usr/src/nightly-{$MACH}.log
1107     fi

1109     mv $LOGFILE $LLOG

1111     ln -s "$LLOG" "$ATLOG/latest"
1112 }
_unchanged_portion_omitted_
```