

BrutusConfig NightlyBuild

BRUTUS NO LONGER RUNS NIGHTLY BUILDS

Gump is not so suitable for nightly builds as its not trusted ([read this](#)). Might make sense to configure some nightly build support on brutus. This document details how I set the basics for that up on brutus. If you want brutus to build nightlies for you, [request a nightly build](#).

```
# create user for nightly builds
useradd -d /home/nightlybuild -g nightlybuild nightlybuild
passwd nightlybuild # something sensible
su - nightlybuild
echo 'general@gump.apache.org' > ~/.forward

# set up basic profile
echo #! /bin/bash > ~/.profile
echo export JAVA_HOME=/usr/local/j2sdk1.4.2_04 >> ~/.profile
echo export PATH=$PATH:$JAVA_HOME/bin >> ~/.profile

# set up dir structure
mkdir ~/bin
echo export PATH=$PATH:/home/nightlybuild/bin >> ~/.profile
mkdir ~/src
mkdir ~/opt
mkdir ~/tmp
mkdir ~/cvs
mkdir -p ~/svn/asf

# set up maven
echo export MAVEN_HOME=/home/nightlybuild/opt/maven >> ~/.profile
echo export PATH=$PATH:$MAVEN_HOME/bin >> ~/.profile
wget http://dist.apache.easynet.nl/maven/binaries/maven-1.0-rc3.tar.bz2
mv maven-1.0-rc3.tar.bz2 ~/src
tar -jxf ~/src/maven-1.0-rc3.tar.bz2
mv maven-1.0-rc3 ~/opt
ln -s ~/opt/maven-1.0-rc3 ~/opt/maven

# set up ant
echo export ANT_HOME=/home/nightlybuild/opt/ant >> ~/.profile
echo export PATH=$PATH:$ANT_HOME/bin >> ~/.profile
wget http://apache.mirror.transip.nl/ant/binaries/apache-ant-1.6.1-bin.tar.bz2
mv apache-ant-1.6.1-bin.tar.bz2 ~/src
tar -jxf ~/src/apache-ant-1.6.1-bin.tar.bz2
mv apache-ant-1.6.1 ~/opt
ln -s ~/opt/maven-1.0-rc3 ~/opt/maven
ln -s ~/opt/apache-ant-1.6.1 ~/opt/ant

# import new settings
. ~/.profile

# initialize maven
cd ~/tmp
maven -Dpackage=com.mycompany.app genapp
maven dist
maven site
maven clover
maven junit-report:report
rm -Rf ~/tmp/*

# copy some commonly needed jars from another machine
scp -r ~/.maven/repository/jsse ~/.maven/repository/javamail ~/.maven/repository/jms nightlybuild@brutus.apache.org:~/.maven/repository

# save this basic maven setup
cp -r ~/.maven ~/.maven-basic-profile

# install the most common optional ant tasks
mkdir -p ~/.ant/lib
cp ~/.maven/repository/junit/jars/junit-3.8.1.jar ~/.ant/lib/
cp ~/.maven/repository/jsse/jars/jsse-1.0.3.jar ~/.ant/lib/
cp ~/.maven/repository/javamail/jars/mailapi-1.3.1.jar ~/.ant/lib/
```

```
wget http://www.apache.org/dist/java-repository/bsf/jars/bsf-2.3.0.jar \
http://www.ibiblio.org/maven/rhino/jars/js-1.5R4.1.jar \
http://www.apache.org/dist/java-repository/xalan/jars/xalan-2.5.1.jar \
http://www.ibiblio.org/maven/jython/jars/jython-2.1.jar \
http://www.ibiblio.org/maven/bsh/jars/bsh-2.0b1.jar \
http://www.ibiblio.org/maven/jsch/jars/jsch-0.1.8.jar
mv *.jar ~/.ant/lib

# save this basic ant setup
cp -r ~/.ant ~/.ant-basic-profile
```

This script for running a build goes into ~/bin/build:

```
#!/bin/bash
#
# Author: Leo Simons
# Copyright (c) 2004 The Apache Software Foundation. All rights reserved.
#
# sample invocations:
#   build svn maven site excalibur trunk
#   build cvs ant dist ant
#   build cvs maven jar:jar jakarta-commons collections

# initialize...
repo=$1      # svn|cvs|...
cmd=$2       # ant|maven|...
target=$3    # jar|site|...
project=$4   # ant|excalibur|jakarta-commons
dir=$5       # .|trunk|logging|...
date=`date +%Y%m%d`
time=`date +%H:%M:%S`

# use a clean ant and a clean maven...
rm -Rf /home/nightlybuild/.maven
cp -r /home/nightlybuild/.maven-basic-profile /home/nightlybuild/.maven
rm -Rf /home/nightlybuild/.ant
cp -r /home/nightlybuild/.ant-basic-profile /home/nightlybuild/.ant

# log stuff
logdir=~/.public_html/builds/$project/$date
mkdir -p $logdir
logsubname=$(eval echo $dir | sed -e 's/\/_/g')
log=$logdir/time:$time-target:$target-dir:$logsubname.txt

echo Starting $target build of $project/$dir at $time >> $log

# clean up
builddir=/home/nightlybuild/$repo/asf/$project/$dir
cd $builddir 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log

echo Working directory: $builddir >> $log
echo >> $log

echo Cleaning up .. >> $log
find -name '*.log' -or -name target -or -name build | xargs rm -Rf 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
echo >> $log

echo Running command: $repo up 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
echo -----
$eval $repo up 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
echo -----

echo Running command: $cmd $target 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
echo ----- >> $log
eval $cmd $target 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
echo ----- >> $log

echo looking for license... >> $log
```

```

licenses=$(find . \
  \( -path '*build*' -or -path '*target*' \) \
  -and \( \
    -name '*.gz' \
    -or -name '*.zip' \
    -or -name '*.bz2' \
    -or -name '*.jar' \
  \) \
  \( -not -path '*.svn*' \) \
)
if [ -n "$licenses" ]; then
  echo found license, copying results... >> $log
  cp -f -v --target-directory $logdir $files 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log

  files=$(find . \
    \( -path '*build*' -or -path '*target*' \) \
    -and \( \
      -name '*.gz' \
      -or -name '*.zip' \
      -or -name '*.bz2' \
      -or -name '*.jar' \
    \) \
    \( -not -path '*.svn*' \) \
  )
  if [ -n "$files" ]; then
    echo $files >> $log
    cp -f -v --target-directory $logdir $files 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
  fi
fi
echo ----- >> $log
echo >> $log

echo Attempting to clean old logs out... >> $log
echo
cd $logdir/..
year=`date +%Y`
month=`date +%m`
day=`date +%d`

oldyear=$(( $year - 1 ))
oldmonth=$(( $month - 1 ))
oldday=$(( $day - 5 ))

files=$(find . \
  -type d -maxdepth 1 \
  -name $oldyear* \
  -or -name $year$oldmonth* \
  -or -name $year$month$oldday* )

if [ -n "$files" ]; then
  echo "deleting..." >> $log
  echo $files >> $log
  rm -Rf $files 2>&1 3>&1 4>&1 5>&1 6>&1 7>&1 8>&1 9>&1 >> $log
fi

echo ...done. >> $log
echo ----- >> $log
echo >> $log

```

Set up an extensible crontab:

```
echo '#! /bin/bash

. /etc/profile
. /home/nightlybuild/.profile
for i in `ls /home/nightlybuild/etc/cron.d/*.sh`; do
  sh $i 2>&1 3>&1 > /dev/null
done' > ~/bin/cron.d.sh

chmod ug+x ~/bin/cron.d.sh
mkdir -p ~/etc/cron.d
echo '5 18 * * * /home/nightlybuild/bin/cron.d.sh' > ~/etc/crontab
crontab ~/etc/crontab
```

Add projects to build:

```
echo 'build svn maven site excalibur trunk/site' > ~/etc/cron.d/00-build-excalibur-site.sh
# ...
```

Do a test run:

```
bash ~/bin/cron.d.sh
```

And take a look at the contents at

<http://brutus.apache.org/~nightlybuild/builds/>