Smuxi - Feature # 504: PPA with daily snapshots

Status:	Closed	**Priority:**	Normal
Author:	Mirco Bauer	**Category:**	Other
Created:	09/23/2010	**Assigned to:**	Mirco Bauer
Updated:	11/27/2011	**Due date:**	
Complexity:	Medium		
Subject:	PPA with daily snapshots		

Description: This is an autobuilder script for banshee's daily PPA:

<pre>

```
#!/bin/bash

REPOSITORY="$HOME/src/debian/pkg-cli/banshee-daily/"
PKG=banshee
UPLOAD_TARGET=banshee-daily
export GIT_DIR="$REPOSITORY/.git"
export DEBEMAIL="hyperair@ubuntu.com"
separator="+"
export GPG_AGENT_INFO=$(cat $HOME/.gpg-agent-info)
export DISPLAY=:0.0

function fail()
{
    echo "$@"
    exit 1
}

function status_msg()
{
    echo "$(basename $0) => $@"
}

function get_hash()
{
    git log --no-color -n1 upstream --format=%h
}

function get_prev_hash()
{
    dpkg-parsechangelog -l"$REPOSITORY/debian/changelog" | \
        sed -n 's/^Version: //p' | \
        sed -re 's/.*git[0-9]+\.r[0-9]+\.([a-f0-9]+)-.*/\1/'
}

function get_latest_date()
{
    git log --no-color -n1 upstream --format=%ci | \
         cut -f1 -d' ' | \
         sed -e s/-//g
}

function get_prev_date()
{
```

```
    dpkg-parsechangelog -l"$REPOSITORY/debian/changelog" | \
        sed -n 's/^Version: //p' | \
        sed -e 's/.*git//; s/\..*$//';
}


get_build_rev()
{
    if ! [ $(get_latest_date) = $(get_prev_date) ]; then
        echo 1
        return 0
    fi
    oldrev=$(dpkg-parsechangelog -l"$REPOSITORY/debian/changelog" | \
        sed -n 's/^Version: //p' | \
        sed -re 's/.*git[0-9]+//; s/^\.r//; s/\.[^.]+$//;')
    [ -z "$oldrev" ] && echo 1 && return 0
    let oldrev++
    echo $oldrev
}

function get_version()
{
    head -n5 "$REPOSITORY/configure.ac" | tail -n3 | \
        egrep -ho '\[[0-9]\]' | \
        egrep -ho '[0-9]' | \
        (while read digit; do echo -n $digit.; done) | \
        sed -e 's/\.$//'
}

function get_upstream_version()
{
    echo $(get_version)${separator}git$(get_latest_date).r$(get_build_rev).$(get_hash)
}

function get_debian_revision()
{
    echo 0ubuntu1
}

function get_full_version()
{
    echo $(get_upstream_version)-$(get_debian_revision)
}

function prepare_changelog()
{
    git log --no-color $(get_prev_hash)..$(get_hash) --format="[%h] %s" | \
        (while read line; do
            if [ -z $first ]; then
                dch --changelog="$REPOSITORY/debian/changelog" \
                    -v $FULLVERSION \
                    -D $1 \
                    "$line" < /dev/null
```

```
            else
                dch --changelog="$REPOSITORY/debian/changelog" \
                    -a \
                    "$line" < /dev/null
            fi
            first=true;
        done)
    cd "$REPOSITORY"
    git commit -i debian/changelog -m "New upstream revision"
}

function update_repository()
{
    git checkout upstream || return 1
    git pull || return 1
}

function checkout_packaging()
{
    git checkout ubuntu/$1 || return 1
    git merge upstream || return 1
}

function make_tarball()
{
    orig_tarball="$REPOSITORY/../build-area/${PKG}_$UPSTREAMVERSION.orig.tar"
    hyena_tarball="$REPOSITORY/../build-area/$PKG-hyena-$UPSTREAMVERSION.tar"
    git archive upstream --prefix="$PKG-$UPSTREAMVERSION/" > "$orig_tarball" || return 1
                                    GIT_DIR="$REPOSITORY/src/Hyena/.git"          git          archive          HEAD
--prefix="$PKG-$UPSTREAMVERSION/src/Hyena/" > "$hyena_tarball" || return 1
    tar -Af "$orig_tarball" "$hyena_tarball" || return 1
    rm $hyena_tarball || return 1
    gzip -9fn $orig_tarball || return 1
}

status_msg "Chdir'ing into $REPOSITORY"
cd "$REPOSITORY" || fail "Couldn't chdir to $REPOSITORY"

unset GIT_DIR


status_msg "Updating $REPOSITORY"
update_repository || fail "Couldn't update $REPOSITORY"

DISTRIBUTIONS="karmic lucid"
for DIST in $DISTRIBUTIONS; do
    status_msg "Checking out $DIST packaging"
    checkout_packaging $DIST

    status_msg "Preparing debian/changelog"
    FULLVERSION=$(get_full_version)+$DIST
    UPSTREAMVERSION=$(get_upstream_version)
    [ "$(get_hash)" = "$(get_prev_hash)" ] && fail "Already up to date."
```

```
    prepare_changelog $DIST || fail "Couldn't prepare debian/changelog"

    status_msg "Building"
    if [ -z $CHECK_FIRST ]; then
      # we have to manually generate the tarball here
      git submodule update --init
      status_msg "Creating tarball"
      make_tarball || fail "Could not make tarball"
      DEBEMAIL=hyperair@gmail.com git bpsa --git-no-pristine-tar || \
        fail "Could not build"
      CHECK_FIRST=1
    else
      DEBEMAIL=hyperair@gmail.com git bps --git-no-pristine-tar || \
        fail "Could not build"
    fi

    status_msg "Signing"
    debsign -pgpg-no-tty -sgpg "$REPOSITORY/../build-area/${PKG}_${FULLVERSION}_source.changes" \
        || fail "Could not sign"

    status_msg "Uploading"
    until dput \
      $UPLOAD_TARGET \
      "$REPOSITORY/../build-area/${PKG}_${FULLVERSION}_source.changes"; do
      sleep 15m;
    done
  done
  </pre>
```

History

09/29/2010 03:47 PM - Mirco Bauer

Jorge Castro gave a nice talk about providing daily packages:
http://irclogs.ubuntu.com/2010/09/28/%23ubuntu-classroom.html#t20:00

11/27/2011 01:48 AM - Mirco Bauer

- Status changed from New to Closed

- Target version set to 0.8.9

- % Done changed from 0 to 100

Daily development builds for Ubuntu which are compatible with Debian are now available on:
https://launchpad.net/~meebey/+archive/smuxi-daily