

PREAMBULE

The objective of this tutorial is to create a debian package (.deb) for yours projects.

This is interesting for :

- > make a complete backup project,
- > install or distribute a project to others computeurs or users.

For this feature we will use the Proview Demo project package structure.

For this tutorial, we will considere the project name as : **myproject**

Note this guide is useable for a project with only 1 root volume.

Use 'pwrp' session with admin rights.

1/ DOWNLOAD PROVIEW DEMO PROJECT

Go to www.proview.se to download the demo project :

- > for Proview5.4 (64bit) pwrdemo54_5.4.0-1_amd64.deb

2/ UNPACK THE DEBIAN PACKAGE

Before to unpack the pwrdemo debian package, create a folder named « **myproject** » for example in the 'Documents' folder.

Then, unpack the pwrdemo54...amd64.deb in the 'myproject' folder .

For this, right click on pwrdemo54...amd64.deb and select « open with Archive Manager » and extract files in 'myproject' folder.

3/ DEBIAN PACKAGE STRUCTURE

After extration, you do have the following structure :

```
myproject ____DEBIAN ____ control
      |                |__ postinst
      |                |__ perm
      |
      |__ usr____      pwrp____      pwrdemo54____      bld
                                |
                                |__ cnf
                                |__ scr
                                |__ .gitignore
                                |
                                |__ share____      applications
                                        |__ doc____      pwrdemo54
```


 don't modify this structure

3-1/ FILE « control »

[myproject/DEBIAN/control](#)

open file with text editor :

```
Package: myproject54
Version: 5.4.0-1
Section: base
Priority: optional
Architecture: amd64
Depends: pwr54 (>= 5.4.0-1)
Replaces:
Maintainer: Proview <postmaster@proview.se>
Description: myproject Demo Project package
             5.4.0-1 Base release
```

 replace 'pwrdemo' by **myproject** (don't remove '54')

3-2/ FILE « postinst »

[myproject/DEBIAN/postinst](#)

open with text editor (display) :

```
#!/bin/bash
ver="54"
pwre_target="os_linux/hw_x86_64"
#!/bin/bash
#
set -e
# Added by build.sh :
# ver=""
# pwre_target=""
#
echo "Here in postinst..."
# Automatically added by dh_installdocs
if [ "$1" = "configure" ]; then
if [ -d /usr/doc -a ! -e /usr/doc/pwrtest -a -d /usr/share/doc/pwrtest ]; then
ln -sf ../share/doc/pwrtest /usr/doc/pwrtest
```



```
fi
fi
# End automatically added section

aroot="/usr/pwrp/adm"
export pwra_db=$aroot/db

echo "Change owner of files to pwrp"
chown -R pwrp:pwrp /usr/pwrp/myproject$ver

# Register volume
if [ -e $aroot/db/pwr_volumelist.dat ]; then
set +e
ptst=`eval grep "\bVolMyproject\b" $aroot/db/pwr_volumelist.dat | grep "\bmyproject\b"`
set -e
if [ "$ptst" = "" ]; then
echo " VolMyproject 0.254.254.199 myproject" >> $aroot/db/pwr_volumelist.dat
fi
fi

fi
# Insert base in projectlist
if [ -e $aroot/db/pwr_projectlist.dat ]; then
set +e
ptst=`eval grep "\bmyproject$ver\b" $aroot/db/pwr_projectlist.dat`
set -e
if [ "$ptst" = "" ]; then
echo "myproject$ver V${ver:0:1}.${ver:1:1} /usr/pwrp/myproject$ver myproject$ver \"\" " >>
$aroot/db/pwr_projectlist.dat
fi
fi

# Insert nodename in DirectoryVolume and create bootfile
nodename=`eval uname -n`
initsh=/tmp/myproject_init.sh
initpwr=/tmp/myproject_init.pwr_com

# Create a shellsript that attaches to projects and runs wb_cmd
cat > $initsh <<EOF
#!/bin/bash
```



See comments to follow

```
source $aroot/db/pwr_setup.sh
source \ $pwr_exe/pwrp_env.sh set project myproject $ver
```

```
wb_cmd @$initpwr
EOF
```

```
chmod a+x $initsh
```

```
# Create a wb_cmd script that inserts nodename and creates bootfile
cat > $initpwr <<EOF
set attr/name=Sim999-DemoNode/attr=NodeName/value="$nodename"/noconf
save
create boot/node=DemoNode
exit
EOF
```

erratum

See comments to follow

```
# Execute shellscript
sudo -u pwrp $initsh
```

```
rm $initsh
rm $initpwr
```

```
if [ -e /home/pwrp/Desktop ]; then
cp /usr/pwrp/myproject $ver/cnf/myproject $ver.desktop /home/pwrp/Desktop/
chown pwrp:pwrp /home/pwrp/Desktop/myproject $ver.desktop
chmod a+x /home/pwrp/Desktop/myproject $ver.desktop
fi
```

 replace 'pwrdemo' by **myproject**



0.254.254.199 this is the volume ID, e.g. 0.1.1.1, 0.1.1.2,...

In Proview dev env, go to toolbar File/Open/GlobalVolumeList to identify the volume ID

Be careful : before to install your deb package on an other computer, assure that the volume ID is not already use (by an other project).

Vol**Myproject** when you create your project, the volume name is generate automatically don't modify this, keep this syntax : VolMyproject (note the 'M' uppercase !)

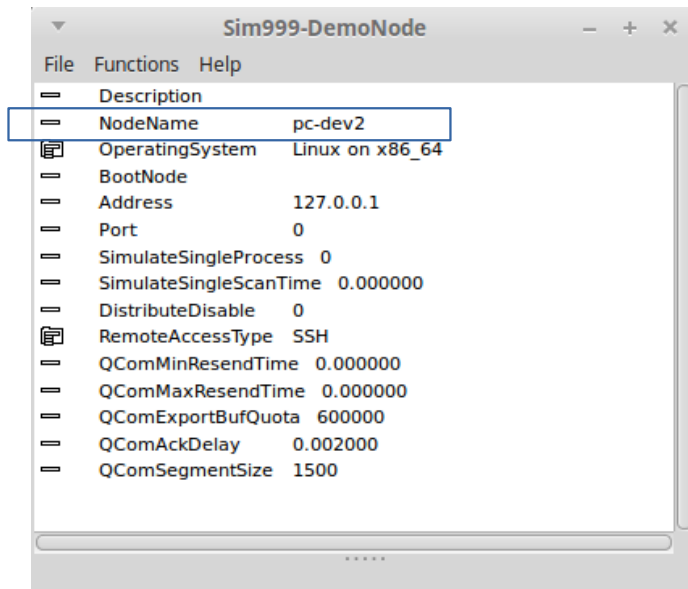
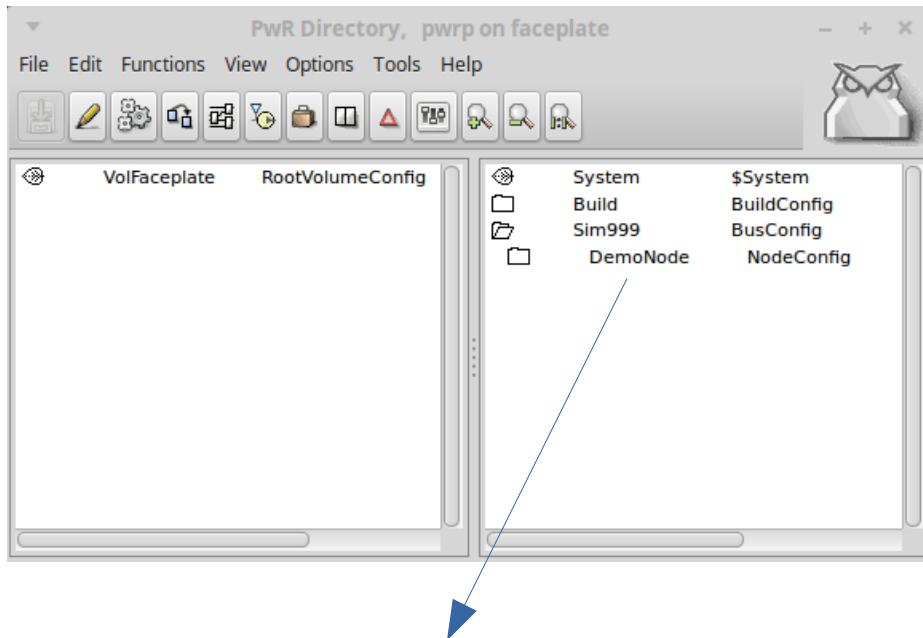
erratum



set attr/name=Sim999-DemoNode/attr=NodeName/value="\$nodename"/noconf
 create boot/node=DemoNode

This script line is used to define the nodename in « NodeConfig » object.

Explanations : when you create your project on the dev station, you define the « NodeName » attribute of the « NodeConfig » object with the dev station name (e.g. pc-dev2, see example bellow) :

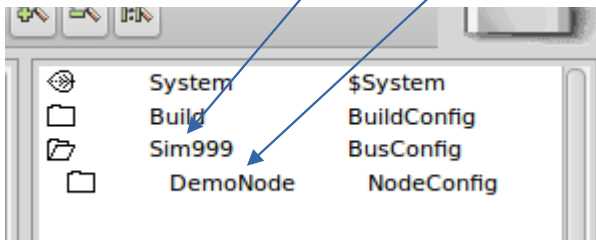


Now if you install your project (with the debian package) on other computer, of course the NodeName is different.

During the installation, the script line :

set attr/name=Sim999-DemoNode/attr=NodeName/value="\$nodename"/noconf
retrieve the host's nodename (variable \$nodename) and update the NodeName attribute (attr=NodeName) in the « NodeConfig » object.

Note « Sim999-DemoNode » is the path to the « NodeConfig » object :



erratum

Important : assure that in your project the « BusConfig » name and the « NodeConfig » name are the same as write in the line :

set attr/name=Sim999-DemoNode/attr=NodeName/value="\$nodename"/noconf

and this line :

create boot/node=DemoNode

3-3/ FILE « prerm »

myproject/DEBIAN/prerm

open with text editor (display) :

```
#!/bin/sh
ver="54"
#!/bin/bash
#
# ver='version' is added by build.sh
#

set -e
echo "Here in prerm..."
# Automatically added by dh_installdocs
if [ \( "$1" = "upgrade" -o "$1" = "remove" \) -a -L /usr/doc/pwrtest ]; then
  rm -f /usr/doc/pwrtest
```

```
fi
# End automatically added section


if [ -e /usr/pwrp/myproject$ver ]; then
  rm -r /usr/pwrp/myproject$ver
fi
```

 replace 'pwrdemo' by **myproject**

3-4/ FOLDER « myproject/usr/share/applications »

This folder contain the 'icon' file definition :

[myproject54.desktop](#)

 modify the name (without extension, with 54)
when you press Enter the .desktop extension is add automatically


right-clic in 'myproject54.desktop' and chose 'properties'.
In the 'command' field, modify **pwrdemo54** by **myproject54**

3-5/ FOLDER « myproject/usr/share/doc »

This folder contain '[pwrdemo54](#)' folder.
Modify this folder name as : '**myproject54**'
Don't modify the contain of this folder.

3-6/ FOLDER « myproject/usr/pwrp »

This folder will contain your project folders.

 Modify the folder name 'pwrdemo54' by '**myproject54**'

Into « myproject54 » folder, remove '**bld**' and '**scr**' folders.

Go to your project folder : [/usr/local/pwrp/myproject/](#)

Copy '**bld**' and '**scr**' folders and paste them to « myproject54 »

This folder contain a hide file : .gitignore (don't modify)

3-7/ FOLDER « myproject/usr/pwrp/myproject54/cnf/ »

This folder contain the 'icon' file definition :

[myproject54.desktop](#)



modify the name (without extention, with 54)
when you press Enter the .desktop extention is add automaticly

right-clic in 'myproject54.desktop' and chose 'properties'.

In the 'command' field, modify **pwrdemo54** by **myproject54**

4/ CREATE THE DEBIAN PACKAGE

Open a terminal :

```
~ $ sudo -s
```

```
[sudo] password for pwrp: (enter : pwrp)
```

```
~ # cd Documents (if the 'myproject' package folder is in 'Documents' folder)
```

```
Documents # dpkg-deb --build myproject (execute dpkg-deb utility)
```

```
Documents # dpkg-deb : "building package 'myproject' in 'myproject.deb'." (package in progress)
```

After few seconds or minutes (depend to your project) you find in « Documents » folder your project debian package : [myproject.deb](#)

5/ INSTALL YOUR .DEB PROJECT PACKAGE



Note : verify that the 'pwrp' user has a administrator rights !

From this step, you are ready to install your .deb package on other computer.

To install your .deb package I advise you to use « gdebi » utility (debian package installer).

This program is include in Linux Mint 17.3, but must be install in Ubuntu 14.04

The advantages of « gdebi » :

- > automatic install and un-install processes,
- > dependencies are manage automaticaly (if necessary)



After installation, you must to have an Proview icon (with your project name) in your desktop.

Double-clic to start the runtime and start your project (you can also run it with dev env.). Enjoy !

Good programming, Ben (Proview forum member) [release : 13fev2017]

[Thanks to Proview team \(Claes\) for this debian package script files !](#)