

LHCb software local installation at Edinburgh

Welcome!

The following page is intended to walk you through the steps required to start using the LHCb software installed in our cluster (not ECDF). It also provides a set of instructions on how to install individual LHCb packages in your home area. A section is dedicated to the software manager providing the basics steps to install LHCb software. These set of instructions are applicable if you want to install the LHCb software in your laptop.

For the USER

As part of LHCb collaboration, you should have a CERN account on the LXPLUS system and a Grid certificate. The LXPLUS account has to be a LHCb account. You can check by logging in to CERN and typing the command `echo $GROUP` which should return `z5`.

System environment flags

Lets start a session on your desktop and open a terminal. At the root of your HOME area there should be a directory called `".hepix"`. If it does not exist, please make it.

```
% cd ~
```

```
> ls .hepix
```

```
ls: .hepix: No such file or directory
```

```
> mkdir .hepix
```

We need to create three files in this directory: `"lhcb"`, `"grid"`, and `"cern-user-name"`. These are files that set flags to tell the system that you will be working for LHCb and using the grid. The `"cern-user-name"` file is required if your login name at LXPLUS is different from the one assigned to you at Edinburgh.

```
> cd .hepix
```

```
> touch lhcb grid cern-user-name
```

The touch command creates three empty files. Edit them with your favourite editor (use pico for instance) and type in the following content in each file (this is self-explanatory):

File Content

```
lhcbyesgridyes
```

```
cern-user-nameyour-login-name-at-cern
```

You can close the terminal now. Open a new terminal and you should see now:

```
*****
```

```
* WELCOME LHCb software *
```

```
*****
```

```
--- $SITEROOT is set to //Disk/lochnagar0/lhcb/lhcb-soft
```

```
--- $CMBSITE is set to LOCAL
```

```
--- $CMTROOT is set to /Disk/lochnagar0/lhcb/lhcb-soft/contrib/CMT/v1r20p20070208
```

```
--- $CMTCONFIG is set to slc4_ia32_gcc34
--- to compile and link in debug mode : export CMTCONFIG=$CMTDEB ; gmake

--- $LHCBPROJECTPATH is set to $!LHCb_release_area:$LCG_release_area
--- projects will be searched in $CMTPROJECTPATH set by the user
-----
```

You are now ready to start working with the locally installed LHCb software. Try for instance:

```
> SetupProject DaVinci v19r5
```

```
Environment for DaVinci v19r5 ready
> which DaVinci.exe
```

```
/Disk/lochnagar0/lhcb/lhcb-soft/lhcb/DAVINCI/DAVINCI_v19r5/InstallArea/slc4_ia32_gcc34/bin/DaVinci.exe
```

In the case of DaVinci, please choose the appropriate version according to your analysis requirements. Consult the DaVinci homepage. The next step on the list is preparation for the "getpack" script to download packages to your home area.

Note added Apr.20-06: Users report the LHCb software does not initialise. If that is the case, one way of solving the problem is to add the following code to the .bashrc file (if using bash or equivalent if using other shell):

```
#!/LHCb env
if [ "$TERM" = "dumb" ]
then
source /etc/profile.d/ifp-lhcb-env.sh
fi
```

CMT/CVS - SSH setup

LHCb software is kept and released at CERN, hence you might need to checkout source from the CERN CVS repository using a secure connection via SSH. We will use SSH keys to simplify the process, in this way you won't need to type a password every time a package is requested.

Start a new terminal, move into the directory .ssh (if it doesn't exist, please make it) and execute:

```
% cd ~/.ssh
% ssh-keygen -t rsa
Generating public/private rsa key pair.
...
```

You will be asked for a file name: just press enter for the default name. In this case it will be "identity". Then a pass-phrase will be asked. Pick something reasonable.

You have now two files: "identity" and "identity.pub". Copy the last file, that is the Public Key, to your LXPLUS HOME area for example using the scp command (or using your afs token):

```
> scp identity.pub lxplus.cern.ch:~ -l cern-login-name
```

Now log on to LXPLUS and run the CERN script to set SSH:

```
> ssh lxplus.cern.ch
```

```
...
> afs/cern.ch/project/cvs/dist/bin/set_ssh
```

You can now add your Public Key to the authorised keys file (ignore this step as the previous script does it for you -just check it is actually done):

```
> cat identity.pub >> ~/.ssh/authorized_keys
```

We are done with the CVS server side configuration. We need now to get back to Edinburgh and configure SSH, so log out from the LXPLUS system. Move into the .ssh directory and modify (create if necessary) the file "config" by adding:

```
#Normal CERN connection
Host lxplus.cern.ch lxplus
User your-cern-login-name
Protocol 2
PubkeyAuthentication no
PasswordAuthentication yes
```

```
#For CERN cvs server
Host isscvs.cern.ch isscvs
User your-cern-login-name
Protocol 1
ForwardX11 no
IdentityFile ~/.ssh/identity
PasswordAuthentication no
```

One last consideration about the ".ssh/config" file: SSH demands this file to have specific permissions. Use %
 chmod 644 config to modify to set the correct permissions.

If your terminal session is under a C shell (if you are not sure execute % echo \$SHELL), you can test the installation by creating a script testSSH.csh:

```
# For csh this could go in $HOME/.cshrc
if ( ! $?SSH_AUTH_SOCKET ) then

eval `ssh-agent -c`
endif
ssh-add
```

Alternatively, if you are under a bash shell:

```
# For bash could go in HOME/.bashrc

# $HOME/.profile
if [ -z "$SSH_AUTH_SOCKET" ] ; then
eval `ssh-agent -s`
fi
ssh-add
```

The script will initialise the ssh-agent and add your key (you will be asked for the passphrase). Try it now:

```
% source testSSH.sh
Enter passphrase for ...
% ssh -l isscvs.cern.ch -l your-cern-login-name
```

You will be asked to accept a server key. Please don't, unless the fingerprint is 16:7a:1d:fc:6d:4b:e3:f0:ae:5e:0d:98:63:d3:c5:75 - that's you! You shouldn't be asked for a password.

The above piece of code could be added to the `.cshrc` or `.bashrc` file. However, you will need to modify the code to ask for a passphrase only once.

Grid certificate placement

You might have a grid certificate imported into your web browser in the PKCS12 format. If so we want to convert it to PEM format and put the files in the appropriate directory so it can be seen by the system.

First we need to create a directory called ".globus" at the root of your HOME area. This is the place to store your grid certificate. Without going into specific details, you should be able to export/save your certificate from your web browser. This is usually done from the "Preferences" menu. Put the created PKCS12 file (having extension .p12) inside the just created directory. Then move in this directory and execute the following commands:

```
% cd ~/.globus

% ls
my_cert.p12
% openssl pkcs12 -in my_cert.p12 -nocerts -out userkey.pem
% openssl pkcs12 -in my_cert.p12 -clcerts -nokeys -out usercert.pem
```

You will be asked for the password that locks your certificate. After this you end up with three files: my_cert.p12, userkey.pem and usercert.pem.

One more step is needed. Change the permissions of each of the newly created files, they won't work anyway (in order to secure them):

```
% chmod go-rw $HOME/.globus/userkey.pem
```

You can do now:

```
% grid-cert-info
Certificate:
...
```

to display your grid certificate information. It is also important you keep a backup of these files in your preferred media (USB key, iPod ...). You can delete the '.p12' file at the end of the procedure to avoid security issues.

Emacs setup

Emacs is a very powerful editor and the LHCb collaboration has produced a customised set of commands that can assist you when working in the LHCb environment. Documentation can be found at:

<http://lhcb-doc.web.cern.ch/lhcb-doc/userdocs/emacs.htm> and

<http://agenda.cern.ch/askArchive.php?a01680/a01680s6t2/transparencies>

As many other programs, Emacs has a hidden configuration file ".emacs" located at the top of your home area. Open it (or create the file) and copy the following lines:

```
;; LHCb framework
;; Load the EDT keypad
(load (expand-file-name "$EMACSDIR/edt"))

;; LHCb general features

(load (expand-file-name "$EMACSDIR/lhcb"))
```

Next time you create a new project (any file with .h .cpp ... file extension), you will be ask what type of code do you intend writing and it will load a skeleton file made for the purpose.

The getpack script

A package is a block of code or data that does one task. It can lead to the compilation of an application such as DaVinci. You might face the situation where you want control over a package and therefore have it under your HOME area. The getpack script downloads (checks out) packages of the Gaudi framework from the CERN repository. Once it has downloaded the code it also performs some CMT configurations.

You are encouraged to install your packages under a directory called "cmtuser" located at the root of your HOME area (create it if it doesn't exist).

If you haven't include the suggested code to initialise the ssh-agent into your shell configuration file (shown at the previous section) please run it again in order to have your identity added. If you are not sure execute the ssh-agent command. It should return some output.

```
% ssh-agent
SSH_AUTH_SOCK=/tmp/ssh-HaUQ2790/agent.2790; export SSH_AUTH_SOCK;
SSH_AGENT_PID=2791; export SSH_AGENT_PID;
echo Agent pid 2791;
```

Having your key in place, you are ready to invoke the getpack script and start getting code from CERN CVS repository. If you need some instructions try:

```
% getpack -h
```

As an example, lets say you want DaVinci v19r5. The final command is:

```
% setenv!DaVinci v19r5
% getpack Phys!/DaVinci v19r5
```

...

Notice you are not prompted for a password! The script will start downloading the files. If everything is ok, you can proceed to compile it:

```
% cd Phys!/DaVinci/v19r5/cmt
% make
```

...

```
> DaVinci ok
```

```
> (constituents.make) DaVinci done
all ok.
% echo $DAVINCIROOT
your-home-area/cmtuser/!DaVinci_v19r5/Phys/!DaVinci/v19r5/
```

One possible problem you might encounter is that your package depends on other packages not present in the local installation. Given that case please, contact your closest Software Manager to help you solve this or any other problem.

For the Software Manager

Introduction

With the CERN batch system becoming increasingly heavily used it has become practical for sites remote from CERN to install local versions of the LHCb analysis software to allow local users to conduct their analysis on local resources. At Edinburgh a local install of DaVinci, Panoramix and other packages has been undertaken. This has been done on a Red Hat Enterprise Linux (RHEL) 4.0 machine using the same recipe used for SL4.

This page will give instructions on installing the DaVinci analysis package and the Panoramix visualisation package from the LHCb software framework. On route the reconstruction package (Brunel) will also be installed. The simulation (Gauss) and digitisation (Boole) packages will not be installed although the method used could be extended to include these packages.

A brief introduction to the software, getpack and CMT can be found at this document

!LHCbsoftware.pdf.

Software installation

The method for installing the LHCb software now utilises a python script, created by Florence Ranjard, which can be used to easily install/compile/maintain the lhcb software. The web-page for generic software installation is available at:

http://lhcb-comp.web.cern.ch/lhcb-comp/Support/html/NEW_Install.htm

For simplicity the instructions contained there are repeated here. Before beginning install must first choose a directory in which to use as the install root. This will later be defined as \$MYSITEROOT. So having chosen the desired location:

```
% setenv MYSITEROOT 'path/of/desired/install'
% cd $MYSITEROOT
% wget http://lhcbproject.web.cern.ch/lhcbproject/dist/install_project.py
```

This will obtain the python scripts that can be used to download and install the lhcb software. Before installing it is also necessary to set the CMTCONFIG environment variable. In this way it will be possible to download binaries of the desired packages. To begin the install the python script can be called as follows:

```
python install_project.py -p [project] -v [version] [ -b ] [ -m [globalselect] ] [ -d ]
```

A full explanation of these options is available on the install webpage mentioned above or through the '-h' option. The project option is used to specify the DaVinci package. The available versions for each of the packages is available at:

<http://lhcbproject.web.cern.ch/lhcbproject/dist/distribution.html>

If the CMTCONFIG environment variable was set then it is possible to simply download the binaries and avoid compiling the package. Otherwise the '-m' option allows the compilation of all the required pre-requisite packages or simply just the package itself.

For the DaVinci version to analyse DC06 data:

```
% python install_project.py -p DaVinci -v v19r5 -b
```

OR

```
% python install_project.py -p DaVinci -v v19r5 -m global
```

Ganga installation

Ganga is a "front end for job definition and management of analysis jobs to run locally and in an distributed environment" written in Python. At Edinburgh we are interested to use it for Grid jobs submission and to smaller local batch queues. Installation follows the instructions found at the Ganga web site:

<http://ganga.web.cern.ch/ganga/user/index.html>

The installation is done using a Python script. Download first the script to the LHCb software installation directory:

```
% wget http://cern.ch/ganga/download/ganga-install
```

Then use it (--help for assistance) to get the desired version:

```
% python ganga-install --prefix=$MYSITEROOT/Ganga 4.4.1
```

We need to provide the --prefix target_dir, otherwise the script will install Ganga on the default directory (which will be your \$HOME area

In order to install additional packages, you may also specify the following options:

```
# For LHCb: --extern=GangaLHCb,GangaGUI,GangaPlotter
# GUI only: --extern=GangaGUI,GangaPlotter
```

At CERN, the user invokes the GangaEnv alias to add Ganga to the environment. It sources a shell script which in turns executes a Python script setting up the environment. This is not explained at the above web site. Hence, we proceeded to copy those scripts and modified them according to our local setup.

```
% cp /afs/cern.ch/sw/ganga/install/etc/setup.csh $TARGETDIR/ganga_env_setup.csh
```

```
% cp /afs/cern.ch/sw/ganga/install/etc/setup.sh $TARGETDIR/ganga_env_setup.sh
```

```
% cp /afs/cern.ch/sw/ganga/install/etc/ganga_setup.py $TARGETDIR/
```

We selected \$MYSITEROOT/scripts directory as the place to keep those scripts. These scripts need to be modified according to local setup. The ganga_env_setup.c{sh} scripts are easy changed. However "ganga_guess" is a bit more complicated, and this is under current investigation. The GangaEnv alias is added to the local configuration script, to be explained in the next section.

Configuration script

When users start a session, the shell reads and executes commands available at /etc/profile.d and sets the environment. The invoked ifp-lhcb-env.{c}sh script checks if the user has the ".hepix/lhcb" file and if it

contains the "yes" flag. If it does, then it executes the "local_setup.{c}sh" script located inside \$MYSITEROOT/scripts directory.

Therefore the "local_setup.{c}sh" script needs to be modified according to what LHCb users will need. The software manager has to edit the local-setup.{c}sh so that environment variables \$LHCB_DIR and \$CMTCONFIG are setup correctly to allow users to get packages from CERN CVS using the "getpack" script. The CERN CVS can be accessed via Kerberos or SSH. In this instance we have selected SSH.

The local_setup.{c}sh scripts are maintained and available from our CVS repository:

```
% cvs co Scripts/lhcb-soft
```

Emacs-!LHCb files

Unfortunately, the LHCb customisation for Emacs cannot be downloaded as a package or obtained via the installation script. Thus we simply took the latest version and copied across the installation areas. They are located at:

```
/afs/cern.ch/lhcb/software/releases/TOOLS/Tools/Emacs
```

It is made available to users through the environment variable EMACSDIR.

More Information

We are keeping up-to-date information in our group TWiki page. You can check on the Software support section to see what's installed locally and what local files are available in our file system.

The official !DaVinci web-page is useful to find out about new versions of DaVinci available.

There are regular DaVinci tutorials held at CERN for those wishing to use DaVinci for physics analysis. Previous tutorial notes for the CERN course as well as a GridPP sponsored !LHCb-UK training course are available.

- ◆ Set USERCOLORSURL = <https://twiki.cern.ch/twiki/pub/Main/LHCbEdinburghGroupSW/colors.css>

This topic: Main > LHCbEdinburghGroupSW
Topic revision: r4 - 2008-02-18 - GreigCowan



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback