

The Lofar Login Environment (LLE)

This page describes the standard login environment for new users on the LOFAR cluster nodes. Existing accounts can be easily modified; see below for a manual to do this.

General

The Lofar Login Environment sets a few things for your account, and allows you to easily initialise often used packages and tools at login time. To this purpose we provide a few simple standardized login-scripts.

Use (t)osh or bash!

To be able to use the LLE environment you must either use a (t)osh or bash login shell. We do not support scripts for any other shells.

What to do when you get an account

(t)osh

If you have chosen for the (t)osh shell, you do as follows:

- Log in to the frontend node lhn001 or lfe001 and go to your (empty) \$HOME directory
- > ln -s /opt/cep/login/cshrc .cshrc.
- Log out and login again; you should see a welcome message (and no errors...).

bash

If you have chosen for the bash shell, you do as follows:

- Log in to lhd002 and go to your (empty) \$HOME directory
- > ln -s /opt/cep/login/bashrc .bashrc
- > ln -s /opt/cep/login/profile .profile
- Log out and login again; you should see a welcome message (and no errors...).

How to transform existing accounts?

To use the LLE in an existing account, rename your existing .cshrc or .bashrc in your \$HOME and follow the instructions given above.

Make sure that possibly existing files \$HOME/.login, \$HOME/.setenv and \$HOME/.alias that you

still want to use at login time are renamed to `$HOME/.mylogin`, `$HOME/.mysetenv`, `$HOME/.myalias`.

The LLE scripts

Some of the details are presented here.

What is `$APS_LOCAL` ?

The root directory for the scripts is provided in environment variable `$APS_LOCAL`, whose value is set in the `.cshrc` and `.bashrc` scripts. The actual value is:

- CEP processing clusters (CEP1, CEP2, CEP3): `/opt/cep`

Login scripts

In directory `${APS_LOCAL}/login` are a number of default login scripts. The scripts ending with `.bash` are for the bash shell, the others for the `(t)csh` shell:

- `cshrc` → Sets `APS_LOCAL` and calls the other scripts for `(t)csh` environments
- `bashrc` → Sets `APS_LOCAL` and calls the other scripts for bash environments
- `profile` → Needed for bash users on Ubuntu systems; calls `bashrc`.
- `login` → Displays welcome message; no settings
- `setenv` → adds some items to `$PATH`, sets your prompt, etc.
- `alias` → some default aliases to make life easy
- `setpackages` → script to define packages to initialize at login-time

Users should **NEVER** modify these default scripts. In their `$HOME` sthey should be symlinks to the versions in `/opt/cep/login`. Personalization of your login is possible through other scripts like `.myalias` and `.mysetenv`.

Personal command aliases: `.myalias`

Personal command aliases can be added to a file `$HOME/.myalias`. When this file exists, the `.cshrc` or `.bashrc` script will read this file after reading the default `${APS_LOCAL}/login/alias(.bash)`.

Personal environment settings: `.mysetenv`

Personal extensions to `$PATH`, personal environment variables, personal prompt setting, or overloaded existing environment variables should be done in a file `$HOME/.mysetenv`. If this file exists, the `cshrc` or `bashrc` script will read this file.

Package initialisation

Many packages are available from the distribution of the Operating System. Several packages are added later on. To activate these, environment variables like `PATH`, `LD_LIBRARY_PATH`, `PYTHONPATH` must be set correctly. To help you, we have created package initialization scripts that you can call to set these parameters correctly.

A list of packages available on the offline processing cluster can be found [on this page](#).

On the commandline

To initialize a package, you must know the name of the initialization file. These can be found in directory `${APS_LOCAL}/scripts`. An example of such a filename is `doLofIm`. This file initializes the `LofIm` package. You can execute it by typing:

```
> use LofIm
```

It can be quite cumbersome to have to initialize all your required packages each time in each shell window you open. Therefore we have made available an option to initialize a list of packages at login time.

For packages that are build daily and that have a version available for all days of the week (`LUS`, `LofIm`), you can also specify a day of week on the commandline:

```
> use LofIm Tue
```

.mypackages

To initialize the use of installed packages at login time you must create a file `$HOME/.mypackages`. If this file exists, it is used by the LLE script `setpackages`.

In the directory `${APS_LOCAL}/scripts` you will find available package initialization scripts, named like `do<package>` (e.g., `doCasa`, `doLofIm`). Whenever you add a package to the file `$HOME/.mypackages`, the associated script in `$APS_LOCAL/scripts` will be source'd.

The file `$HOME/.mypackages` can look like this example:

```
Casa  
LofIm
```

Provide only one package per line. Also note the use of `<package>` and `do<package>`; both are accepted. If an initialization script for a package cannot be found, it will be reported to the user.

For packages that are build daily and that have a version available for all days of the week (`LUS`, `LofIm`), you can also specify a day of week in the `.mypackages` file:

```
LofIm Tue
```

How to add a personal package

Apart from the systemwide `do<package>` files provided in directory `$APS_LOCAL/scripts`, users can add their personal `do<package>` files in their `$HOME` and have these run at login time. If there is a `do<package>` file both in `$HOME` and in `$APS_LOCAL/scripts`, the version in `$HOME` has preference and will be executed. This can be used to test a different version of a package, etc... To add a new, personal, package, act as follows:

- Install the package
- Create a `$HOME/do<package>` initialization script (e.g., adding the installation directory to your `$PATH`)
- Add `<package>` to the list of packages in file `$HOME/.mypackages`

Starting X environment

To start up an X environment you have to make sure that

- The files `$HOME/.xinitrc` and `$HOME/.xsession` are removed

Determine the colordepth that you need (8, 16, 24 bit display) and type:

```
startx -- :1 -depth <colordepth>
```

This should start your X environment at the proper colordepth. This will create an additional X Server next to the one you are already running. You can access this with Alt-F8 or Cntl-Alt-F8. The F7 variant will give you back your normal X.

Due to the large latency and large bandwidth required for X-traffic, it is advisable to connect to the CEP systems with the NX-client on your system. See [this page for more info](#).

Help!

If you need help, or have questions, or want to give any other comments, contact Arno Schoenmakers or Adriaan Renting.

New `do<xxxx>`-files can be added at any time if you have a new package that should be available systemwide. Contact Arno Schoenmakers or Adriaan Renting in this case.

From:
<https://www.astron.nl/lofarwiki/> - **LOFAR Wiki**

Permanent link:
<https://www.astron.nl/lofarwiki/doku.php?id=public:lle&rev=1493793360>

Last update: **2017-05-03 06:36**

