

# Notes on Processing at the Juropa Cluster

This is going to be the Wiki page for the Lofar Software installation at the Juelich Supercomputing Centre. I will update this page as the installation progresses.

Currently the software is being tested on the Juropa system. Everything that is part of the Calibration and the Target Pipeline is working. The awimager causes problems but might be working in an experimental build (details at the end).

The software is available in the home directory of user zdv596. The root path of the install is

`/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14`

You can find the lofar software in "lofar/release". The environment you need is loaded with the script "variables\_lofar.sh"

In addition you might need a copy of the measurement data `/lustre/jhome9/lofar/zdv596/dataCEP` in your home directory and point to it in a file `.casarc` (just contains: "measures.directory: [yourhome]/dataCEP")

If you require access to the GlobalSkyModel database, there is a copy of the database from the CEP Cluster (hopefully) running on the Juropa login node `jj28l02`. Access the database "gsm" on port 51000 with user "gsm" and pass "msss"

How to keep the measurement and gsm data up to date and distributed has to be discussed

You can now run and test the executables on the login node from "lofar/release/bin" or run python scripts (your own or pipeline scripts in "local/lib/python2.7/site-packages/lofarpipe/recipes").

To run your jobs on the compute nodes you first have to setup and submit a job via the batch system. A detailed description can be found on the Juropa homepage

<http://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/JUOPA/UserInfo/QuickIntroduction.html>

Here is a simple example of the procedure. Basically you use two scripts. One to configure the job and one to setup the environment for your program and run it.

Job configuration is pretty basic right now because we can only utilize one node per job. Do not get confused by the use of comments '#'. The '#' in front of MSUB commands is necessary for the command to be recognized from the Moab batch system.

You submit the job with the command "msub [yourscript]". Check your status with "showq -u 'username' ". To see the whole machine with a gui try the "llview" program.

Contents of 'lofarmsub.sh':

```
#!/bin/bash -x
#MSUB -N Lofar-test
# just the name
#MSUB -l nodes=1:ppn=8
#MSUB -l walltime=00:30:00
#MSUB -e error.txt
# if keyword omitted : default is submitting directory
#MSUB -o output.txt
```

```
# if keyword omitted : default is submitting directory
#MSUB -M your@mail.de
#Mailadress
#MSUB -m eab
#send mail on end, abort, begin
./lofarCalibratorPipelinePy2.7.sh
```

The walltime is the time your job will be running on the machine. If it is too low and the job is not finished it will be killed. If it is too high your job might have to wait longer in queue but only the real computing time will be booked.

The maximum walltime is 24h.

The number of nodes has to stay at 1 for the time being. You can experiment with ppn (process per node) which is used for openmp enabled programs.

It is best to name the log files error and output with some job specific parameters and maybe the date.

You can choose to have mails sent to you about the status of your job.

Contents of 'lofarCalibratorPipelinePy2.7.sh':

```
#!/bin/sh!
#start of jobscript
export OMP_NUM_THREADS=8
#
#
export
PYTHONPATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/local/lib/python2.7/site-packages:$PYTHONPATH
export
PYTHONPATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/lofar/release/lib/python2.7/site-packages:$PYTHONPATH
#
export PATH=/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14/local/bin:$PATH
export
PATH=/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14/lofar/release/bin:$PATH
export
PATH=/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14/lofar/release/sbin:$PATH
#
export
LD_LIBRARY_PATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/lofar/release/lib:$LD_LIBRARY_PATH
export
LD_LIBRARY_PATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/lofar/release/lib64:$LD_LIBRARY_PATH
export
LD_LIBRARY_PATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/local/lib:$LD_LIBRARY_PATH
export
LD_LIBRARY_PATH=/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/local/lib64:$LD_LIBRARY_PATH
#
```

```
export LOFARR00T=/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14
#
/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/lofar/release/bin/msss_calibrator_pipeline.py
/lustre/jwork/lofar/zdv596/Pipeline_testdata/calibrator_pipeline/Observation
64405
-C
/lustre/jhome9/lofar/zdv596/LOFAR-
Release-1_14/lofar/release/share/pipeline/pipeline.cfg
--job calibrator_branch_regression -d
```

Simply replace the pipeline call with the command you want to run in your job. Example of Alexanders bbs test: `/lustre/jhome9/lofar/zdv596/LOFAR-Release-1_14/lofar/release/bin/calibrate-stand-alone -v -n -f L104244_SB200_uv.dppp.MS BBS.parset skymodel.parset` One important remark for your working directory. Use the Filesystem mounted under \$WORK for your data and jobs.

From the Juropa home page:

\$WORK

File system for large temporary files with high I/O bandwidth demands (scratch file system). No backup of files residing here. Files not used for more than 28 days will be automatically deleted!

If you want to try the awimager try another build directly in the home directory

`/lustre/jhome9/lofar/zdv596/lofar/release.`

There are some weird problems with pyrap and this version uses a different compiler. The awimager seems to work but there are other problems which extend has not been analyzed yet.

I hope these information are sufficient for some first tests and experiments.

Good luck and let me know of any problems and feel free to give some feedback.

on holidays for one week... after that more updates

From:

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

Permanent link:

[https://www.astron.nl/lofarwiki/doku.php?id=public:processing\\_at\\_juropa&rev=1374847242](https://www.astron.nl/lofarwiki/doku.php?id=public:processing_at_juropa&rev=1374847242)

Last update: **2013-07-26 14:00**

