LOFAR Docker Images

We offer the LOFAR software as Docker images, allowing anyone to run our software using the same configuration, without having to build it. Our images can be browsed at https://hub.docker.com/r/lofar/.

LOFAR Interferometry Post-Processing Software

To run the lofar pipeline software, you need to:

1. Install and configure Docker on your computer (Mac/Linux/Windows)
2. Download and start the LOFAR image by running:

   ```
   docker run -it --rm -u $UID -e USER -e HOME -v $HOME:$HOME lofar/lofar-pipeline:LOFAR-Release-2_19
   ```

   To list all available versions, go to https://hub.docker.com/r/lofar/lofar-pipeline/tags/.

3. You now have the LOFAR software available at your finger tips, and your home directory available. You can run for example:

   ```
   you@3617438dfc63:/$ NDPPP
   Usage: DPPP [-v] [parsetfile] [parsetkeys...]
     parsetfile: a file containing one parset key=value pair per line
     parsetkeys: any number of parset key=value pairs, e.g. msin=my.MS
   ```

   or:

   ```
   you@3617438dfc63:/$ long_baseline_pipeline.py
   /opt/lofar/lib/python2.7/site-packages/lofarpipe/support/utilities.pyc
   : Using default subprocess module!
   Usage: /opt/lofar/bin/long_baseline_pipeline.py <parset-file>
   [options]
   Results:
   ```

   or one of our dependencies:

   ```
   you@3617438dfc63:/$ /opt/aoflagger/bin/aoflagger
   AOFlagger 2.8.0 (2016-06-21) command line application
   This program will execute an RFI strategy as can be created with the
   RFI gui
   and executes it on one or several observations.

   Author: André Offringa (offringa@gmail.com)
   ```

```
Usage: ./aoflagger [options] <obs1> [<obs2> [..]]
...
```

LOFAR Pulsar Post-Processing Software

To run the lofar pipeline software, you need to:

1. Install and configure Docker on your computer (Mac/Linux/Windows)
2. Download and start the LOFAR image by running:

   ```
   docker run -it --rm -u $UID -e USER -e HOME -v $HOME:$HOME lofar/lofar-pulp:LOFAR-Release-2_19
   ```

   To list all available versions, go to https://hub.docker.com/r/lofar/lofar-pulp/tags/.

3. You now have the LOFAR pulp software available at your finger tips, and your home directory available. You can run for example:

   ```
   you@3617438dfc63:/$ pulp.py
   Usage: pulp.py <--id ObsID> [-h|--help] [OPTIONS]
   ```

Q&A

Installation

Q: Where can I find a list of available LOFAR software versions?

A: For `lofar-pipeline`, surf to https://hub.docker.com/r/lofar/lofar-pipeline/tags/. For `lofar-pulp`, surf to https://hub.docker.com/r/lofar/lofar-pulp/tags/.

Q: What is included in the image?

We put the following in our `lofar-pipeline` image:

- LOFAR Pipeline Framework & Recipes
- NDPPP
- AWImager
- AOFlagger
- DAL2
- Casacore + casarest + python-casacore

Running the software

Q: I can't see my data or parset?

Once you have the Docker image up and running, you will need some data to work on, and likely a parset with configuration settings. If you've put both in your home directory, they're immediately available when running with the commands described on this page. If not, you can use additional ''-v'' parameters to make more directories available in your Docker container.

Q: I get "Illegal instruction" when running some of the software?

A: The LOFAR software is compiled for a 2015-era processor (Intel Xeon E5-2603v3 to be exact) for performance reasons, and uses CPU instructions not available on older machines. Please run our images on a newer machine.

Development (extending the software)

Q: How do I obtain your Dockerfiles?

A: By running:

```
svn co -N https://svn.astron.nl/LOFAR/branches/LOFAR-Release-2_19/
cd LOFAR-Release-2_19
mkdir -p build/gnu_opt && cd build/gnu_opt

cmake ../.. -DBUILD_PACKAGES=Docker -DUSE_LOG4CPLUS=OFF -DUSE_CASACORE=OFF
&& make -j 16 install
```

which will put the Dockerfiles in subdirectories in `LOFAR-Release-2_19/build/gnu_opt/installed/share/docker`.

Q: How do I build your Docker images from scratch?

A: By obtaining the Dockerfiles (see above), and running:

```
# load LOFAR environment variables, used to determine tags etc
source installed/lofarinit.sh
# build all LOFAR images, in the order required by their interdependencies
docker-build-all.sh
```

Q: How do I add files/modifications permanently?

A: The best way to add stuff to our images is by creating a new image based on ours. To do so, create a directory containing the files you want to add. Add to this directory a `Dockerfile` that describes your extension, for example:

```
FROM lofar-pipeline:LOFAR-Release-2_19

COPY my_extension /opt/my_extension
```

Finally, you run

```
docker build -t myimage:mytag directory/
```

where `directory` is the directory containing the Dockerfile, and `myimage:mytag` is the name and tag of your new image (instead of `lofar-pipeline:LOFAR-Release-2_19`). See also the Docker manual on how to make and use Dockerfiles.

Q: How do I add initialisation stuff? (PYTHONPATH, etc)

A: Our Docker images execute `/opt/bashrc` upon entry, which in turn reads and sources all the files in `/opt/bashrc.d/` in order. You can thus add bash files to this directory that contain your initialisation statements, f.e. a file called `20-myextension` containing `export PYTHONPATH=$PYTHONPATH:/opt/my_extension`.

Note: if you add eggs to the image, you need to put the egg *filename* to the PYTHONPATH. It is not enough to add the directory containing the egg.