NTT Pipeline Installation Guide

Tomás Müller, Cosimo Inserra
email: t.e.muller-bravo@soton.ac.uk

January 2020

1   Introduction

I made this guide to install the NTT pipeline on my computer. The OS used here is Ubuntu (18.04 at the time I am writing this), although many of the steps might also work or be similar for Mac.

Before starting, check that your system is up to date. Open a terminal and type "`sudo apt update`" and "`sudo apt-get dist-upgrade`". You might need to reboot the computer after this. This is the only step not needed on MacOs.

2   Git

Git is mainly used for version control. Many packages use it, so it might be a good idea to install it. Just type "`sudo apt install git`". In MacOs you can download the installer from the website or use conda, homebrew (`brew install git`) or macports.

3   Python

The best way to install Python (v3.7, current version) is through Anaconda. Once you download the file (e.g., `Anaconda3-5.0.1-Linux-x86_64.sh`) you need to run it by typing "`bash Anaconda3-5.0.1-Linux-x86_64.sh`" in a terminal. For MacOS it is only needed to download the installer.
It can be found at `https://www.anaconda.com/distribution/`

3.1   Iraf env (+ds9)

Iraf is a cornerstone of Astronomy and it is now available in Anaconda. The steps to install it are listed in the following link. Basically what you need to do is:

1. Configure Anaconda to install packages directly from the AstroConda repository.

    ```
    conda config --add channels http://ssb.stsci.edu/astroconda
    ```

2. Create the Python 2.7 package with IRAF by using the command:

    ```
    conda create -n iraf27 python=2.7 iraf-all pyraf-all stsci
    ```

3. **Optionally**, you can create the Python 3 AstroConda environment. This is not necessary to run the NTT pipeline, but python 2.7 support will stop on the 1st of January 2020. However, at the moment, there is no plan on migrating the pipeline to Python 3. Anyway, if you want the Python 3 AstroConda environment use the following command

    ```
    conda create -n astroconda stsci
    ```

While creating the AstroConda environment I encountered some errors (broken files maybe?). I found the solution and some explanations here. If you encounter the same error, you needed to type `conda clean --all` to remove the problematic files (and many others which where not being used) and then try again. Be patient, all these installations take A WHILE!

To setup Iraf you need to create a `login.cl` file. You might want to create an "iraf" directory under your home directory and do the following:

```
mkdir /home/user/iraf
cd /home/user/iraf
mkiraf
```

Something useful to have together with `Iraf` is `ds9` (unsurprisingly used by the pipeline) which, fortunately, is also available in Anaconda. Ds9[1] should have been installed within the `Iraf` environment, but if doesn't work for any reason (it might tell you that it didn't find the `libssl.so.10` file, do not pay attention to this!), follow these instructions. You want to install it in the same environment as `Iraf`, so you will first need to activate it by typing "`source activate iraf27`" (unless you have changed the name of the environment). To install `ds9` you need to type the next command:

```
conda install -c pkgw-forge ds9
```

and that is all.

3.2   NTT pipeline

There are two ways to install the PESSTO NTT pipeline. The first and shorter option is the clone the github repository and run the installation. Follow the next commands:

```
cd /where/you/want/to/have/the/pipeline/directory
git clone https://github.com/svalenti/pessto.git
cd pessto/trunk
python setup.py install
```

and that is all. **Warning**: this will not necessarily install an stable version as the pipeline is sometimes being updated, but very rarely will this lead to a "broken" version (use this option under your own risk!). You will also need to have "Swarp" installed (see below). You will need to add the path of the pipeline `/bin` directory to your `.bashrc` file for Ubuntu or choose the appropriate file depending on your OS and/or shell type.

The second option is a bit longer, but you can be sure that you are going to install the latest stable version. The first thing to do is to download the pipeline from the wiki page (2.4.1 is the current version at the moment I wrote this. Remember, the pipeline should NOT be installed with Ureka (which is deprecated), but with Anaconda (following this guide, not the old one). If you already have your "iraf27" environment installed, you can proceed by creating the `login.cl` file if you haven't it already (see above).

The pipeline requires "Swarp" to be installed. You can install it from source by downloading the software from the SWarp website (download here, there might be a more recent version at the time you read this) and then following the next commands (the first two commands are in case you haven't manually downloaded the package yet - but you need to have installed wget):

```
cd ~/Downloads
wget www.astromatic.net/download/swarp/swarp-2.38.0.tar.gz
cd /tmp
tar xfz ~/Downloads/swarp-2.38.0.tar.gz
cd swarp-2.38.0
./configure
make
sudo make install
```

Now starts the fun part! Follow the next commands (again, the first two commands are in case you haven't manually downloaded the package yet):

```
cd ~/Downloads
wget https://github.com/svalenti/pessto/archive/v2.4.1.tar.gz
tar -zxvf v2.4.1.tar.gz
```

I like to have all the main packages, software, pipelines, etc., in my home directory. However, you can install the pipeline wherever you want. Let's continue:

---

[1]If you are a Star trek fan, such name might be familiar. If you are wondering about any connection with the Star Trek TV show, you are on the money! It was name after the DS9 Star Trek series and it follows its predecessor displaying tool called TNG, named after Star Trek The New Generation.

```
cd ~   # move to your home directory

mkdir nttpipeline

cd nttpipeline

cp -r ~/Downloads/pessto-2.4.1/trunk/* .
# this copies all the files recursively to the nttpipeline directory
# (you current directory, see the dot "." at the end of the command)

conda activate iraf27 # remember the pipeline works
                      # with the iraf27 environment

python setup.py install (--record NTTpipe_install_files.txt)
```

Note, in the pipeline .tar file, there is also a README.txt.
Remember to add the path of the pipeline `/bin` directory to your `.bashrc` file for Ubuntu or choose the appropriate file depending on your OS and/or shell type. If for some divine intervention you didn't encounter any issue (fingers crossed!), that is all! You can see the guide to test if the pipeline was installed correctly.

4   Aliases and Export Paths

Add the following to your *.bashrc* file if you are using Ubuntu or the appropriate file depending on your OS and/or shell type.

```
# anaconda
export PATH="/home/<user>/anaconda3/bin:$PATH"

# iraf (useful to quickly access the "iraf27" environment)
alias iraf27="source activate iraf27"

# NTT pipeline
export PATH=$PATH:/home/<user>/nttpipeline/bin
```

5   Reported Bugs (last update: 01 September, 2019)

There's a conflict between macOS 10.14.6 and Tkinter:
https://discussions.apple.com/thread/250549297
https://www.python.org/download/mac/tcltk/built-in-8-6-8

If you are using macOS 10.6 or later, the Apple-supplied Tcl/Tk 8.5 has serious bugs that can cause application crashes.

Running the pipeline (and pyraf) will cause a window server crash (on python 2.7.16). This has also seems to affect DS9 (they released a beta to address this).

FIX (Courtesy of Seán Brennan): I managed to fix this by doing `conda install python=2.7.15` (rather than python 2.7.16) on my iraf environment and that has seemed to allow the pipeline/pyraf to open the interactive window.