Using *conda to provide python dependencies for esoreflex

Paola Popesso - 2025-02-20 - Comments (0) - Data processing and analysis software resources

Using *conda to provide python dependencies for esoreflex

Whether installed via the package managers (RPM or MacPorts) or via the install script, esoreflex will be configured to use the system python (usually /usr/bin/python), or for MacPorts /opt/local/bin/python3.X). And even for script installations instructions are provided here to install the required python packages via package managers covering macOS, RPM- and APT-based systems.

But it is possible to configure esoreflex to use any python installation. But note, except in the case of package manager installations, it is the user's responsibility to insure that all the required python packages are installed.

The following is a step-by-step guide to setting up a conda environment for esoreflex.

The following assumes that esoreflex and the pipelines you want are already installed. See the Pipeline pages at http://eso.org/sci/software/pipelines/index.html if you need help with that.

Before starting, please note that lines formatted like the following:

```
ls
```

mean that you should execute the command (preferably by copy/paste) in a terminal. In general it is safest to copy them line by line, but note some commands are broken onto multiple lines by a '\' character at the end of the line, in this case copy/paste all sequential lines ending in '\' plus the next line with no '\' at the end of the line.

The following assumes that your shell is bash, so if you do not use bash shell, or if you are not sure which shell you use type

```
bash
```

If for some reason you do not want to use bash, then it is up to you to adapt the following to whatever shell you are using.

Install *conda

If you don't already have a *conda installation, we suggest installing miniconda.

Under Linux, miniconda is available via RPM/apt package directly from conda.io (recommended) see:

https://docs.conda.io/projects/conda/en/latest/user-guide/install/rpm-debian.html

or via third-party sources (e.g. EPEL).

For macOS it is possible to install miniconda as follows:

```
cd /tmp
MoL=$(uname -s | sed -e 's/Darwin/MacOSX/')
curl -O https://repo.anaconda.com/miniconda/Miniconda3-latest-${MoL}-x86_64.sh
chmod u+x ./Miniconda3-latest-${MoL}-x86_64.sh
./Miniconda3-latest-${MoL}-x86_64.sh
```

Note, to avoid setting the *conda environment as your default environment, at the following prompt near the end of the installation:

```
Do you wish the installer to initialize Miniconda3
by running conda init? [yes|no]
[yes] >>>
```

Answer **no**.

Create an esoreflex conda environment

Now assuming *conda is already installed, activate if not already activated:

```
eval "$(conda shell.bash hook)"
```

Note: you may need to specify the full path to conda if it is not in your PATH, e.g. for the above Miniconda installation, ${HOME}/miniconda3/bin/conda.

Now with conda activated and esoreflex already installed, assuming that the “esoreflex” command can be found in the PATH then do:

```
ESOREFLEX_INSTALL_DIR=$(dirname $(dirname "$(which esoreflex)"))
```

Create a *conda esoreflex environment -- wxpython is no longer available from the default repos, so we suggest conda-forge:

```
ESOREFLEX_ENV_NAME=esoreflex
conda create -n ${ESOREFLEX_ENV_NAME} -c conda-forge python=3.12 matplotlib wxpython
astropy
```

From time to time, one or other, or more of the required packages may fail to install. We can not help you with this, you will have to resolve any and all such issues yourself. It may also be that esoreflex is not compatible with a given version of anyone of these packages. Again, we can not help you with this, you will have to resolve any and all such issues yourself.

Configure esoreflex to use the esoreflex conda environment

Configure esoreflex to use the esoreflex *conda environment

```
conda activate ${ESOREFLEX_ENV_NAME}
```

Please follow the instructions below, as appropriate for your operating system/distribution.

on Linux:

system-wide:

```
sudo sed -i \
  -e "s[^esoreflex.python-command=.*[esoreflex.python-
command=$CONDA_PREFIX/bin/python[" \
  ${ESOREFLEX_INSTALL_DIR}/etc/esoreflex.rc
```

personal:

Use this option only if the system-wide option is not possible (e.g. because you don't have sudo/admin access on your computer).

```
esoreflex -create-config
sed -i \
  -e "s[^esoreflex.python-command=.*[esoreflex.python-
command=$CONDA_PREFIX/bin/python[" \
  ${HOME}/.esoreflex/esoreflex.rc
```

on macOS:

```
conda install python.app
```

system-wide:

```
sudo sed -i '' \
  -e "s[^esoreflex.python-command=.*[esoreflex.python-
command=$CONDA_PREFIX/bin/pythonw[" \
  ${ESOREFLEX_INSTALL_DIR}/etc/esoreflex.rc
```

personal:

Use this option only if the system-wide option is not possible (e.g. because you don't have sudo/admin access on your computer).

```
esoreflex -create-config
sed -i '' \
  -e "s[^esoreflex.python-command=.*[esoreflex.python-
command=$CONDA_PREFIX/bin/pythonw[" \
  ${HOME}/.esoreflex/esoreflex.rc
```

Install additional/optional packages for particular workflows

astroscrappy & PyCosmic for GIRAFFE:

```
conda install -y scipy
python -m pip install
https://files.pythonhosted.org/packages/source/a/astroscrappy/astroscrappy-1.0.8.tar.g
z
python -m pip install https://ftp.eso.org/pub/usg/PyCosmic/PyCosmic-0.5.3.tar.gz
```

scikit-learn for MUSE-ZAP:

```
conda install scikit-learn
```

Completing the installation...

Deactivate the conda esoreflex environment:

```
conda deactivate
```

Deactivate the conda base environment (optional):

```
conda deactivate
```

Running day-to-day...

Note, once you've performed the above procedure, you do not need to activate the esoreflex conda or base conda environment to use esoreflex, you can simply execute from any environment.

```
esoreflex
```

- Tags
- conda
- esoreflex