Using venv/virtualenv to provide python dependencies for esoreflex

John Pritchard - 2025-02-20 - Comments (0) - Data processing and analysis software resources

Using venv/virtualenv 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 python-venv 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 python

You must have a functional version of Python installed. We can not provide instructions for that here. If you don't know how to at least do this, then this method is almost certainly not for you. Also note that as Python version 3.3, the venv module is built in to Python. For earlier Python versions, notably 2.7, you would need to install the virtualenv python module. Again, if you don't know how to do this, then this method is almost certainly not for you.

Create an esoreflex venv/virtualenv environment

The functionality of venv and virtualenv (as required below) is essentially the same. So we will proceed with venv, but if you want to use virtualenv instead, simply replace venv with virtualenv in the specified commands.

With 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 venv or virtualenv esoreflex environment and install the required python modules:

```
ESOREFLEX_ENV_NAME=esoreflex
mkdir -pv ${HOME}/envs
python -m venv ${HOME}/envs/${ESOREFLEX_ENV_NAME}
. ${HOME}/envs/${ESOREFLEX_ENV_NAME}/bin/activate
python -m pip install 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 venv/virtualenv environment

With the esoreflex venv/virtualenv environment still activated as above, 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=${HOME}/envs/${ESOREFLEX_ENV_NAME}/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=${HOME}/envs/${ESOREFLEX_ENV_NAME}/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=${HOME}/envs/${ESOREFLEX_ENV_NAME}/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=${HOME}/envs/${ESOREFLEX_ENV_NAME}/bin/python[" \
  ${HOME}/.esoreflex/esoreflex.rc
```

Install additional/optional packages for particular workflows

astroscrappy & PyCosmic for GIRAFFE:

```
python -m pip install 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:

```
python -m pip install scikit-learn
```

Completing the installation…

Deactivate the esoreflex venv/virtualenv environment:

```
deactivate
```

Running day-to-day…

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

```
esoreflex
```