How to run ESO pipelines under Microsoft Windows (*)

Lowell Tacconi-Garman - 2024-10-29 - Comments (0) - Data processing and analysis software resources

How to run ESO pipelines under Microsoft Windows (*)

Introduction

ESO officially supports running its pipelines under recent versions of MacOS and several flavors of Linux. The Microsoft Store for recent version of Windows 10 and Windows 11 include the Windows Subsystem for Linux Version 2 (WSL2). WSL2 allows to run a GNU/Linux environment -- including command-line tools, utilities, and applications-- unmodified directly on Windows, without recompilation or porting, and without the overhead of a traditional virtual machine or dual-boot setup. With the inclusion of WSLg, Windows can be used to run almost any Linux application, including those with GUIs.

Such an installation can be used to run ESO pipelines either standalone or within the EsoReflex environment. ESO does not officially support running pipelines under Windows. However, we have reports of problem free experiences in running pipelines under Windows in that manner. After installation, all documents and tutorials written for Linux are applicable.

ESO is committed to helping users on a best effort base with problems while running pipelines under Windows.

Below, we give a basic summary of how to install pipelines under Windows.

Prerequisites

WSL2, supported on Windows 10 & 11, is required.

Windows 10 versions earlier than 1903 did not allow to install WSL2. Its predecessor, WSL, worked in a different manner, and we **strongly advise against** trying to run the ESO pipelines with it. Before installing ESO pipelines, it is therefore strongly advised to upgrade Windows to its latest version and the following instructions assume that you have done so.

For a pleasant user experience, we recommend a computer with at least 8 GB of RAM, 50 GB of free disk space, and a modern CPU with at least 4 physical cores. Complex pipelines like MUSE require significantly more memory and computational resources.

Step-by-Step Guide to install WSL2 and then ESO Pipelines/EsoReflex

The following instructions have been tested with the Windows 10 operating system but should work with all recent versions of Windows 10 and 11.

We recommend using the Ubuntu Linux distribution since it is provided directly by Microsoft and is free. Recent versions of WSL2 include the WSLg, which is automatically activated when WSL2 is installed. It is therefore no longer necessary to install a third party X-server.

Step 1: install and configure Ubuntu

1.1 The most reliable way to install WSL2 is to use the Microsoft Store. Simply open the Microsoft Store, search for “Ubuntu 20.04 LTS” or later, and install it.

1.2 Open Ubuntu with a double click, provide the requested information to add a user account.

Step 2: Install pipelines

2.1 Install pre-requisites by typing into the Ubuntu window:

```
sudo apt-get update
sudo apt-get install openjdk-11-jre-headless
sudo apt-get install g++ zlib1g-dev make gzip bzip2
sudo apt-get install tar perl gawk sed grep coreutils pkg-config
sudo apt-get install python3-matplotlib python3-wxgtk4.0
sudo apt-get install python3-astropy python3-numpy default-jre
libffi-dev
sudo apt install ftools-fv wish
```

2.2 Install ESOReflex and pipelines by typing into the Ubuntu window:

```
mkdir eso_pipelines
cd eso_pipelines
wget https://ftp.eso.org/pub/dfs/reflex/install_esoreflex
chmod +x install_esoreflex
./install_esoreflex
```

Follow instructions to install selected pipelines, demo data and EsoReflex.

Running the pipelines in EsoReflex

Before running the pipeline, make sure the Xserver and Ubuntu have been started (1.2 and 2.3 above). If the installation was done following the instructions above, then the executable for EsoReflex will be in ~/eso_pipelines/install/bin/esoreflex.

To find out what workflows have been installed, type into the Ubuntu window:

```
~/eso_pipelines/install/bin/esoreflex -l
```

To start EsoReflex with one of the installed for_imaging pipeline, type:

```
~/eso_pipelines/install/bin/esoreflex fors_imaging
```

Tutorials on how to run EsoReflex are available at https://eso.org/pipelines .

FAQs

1. Q: I started an EsoReflex workflow, but it seems to hang without any progress.

A: In most cases, the reason is that EsoReflex is waiting for input, but the input window is closed. Check the task bar for windows related to EsoReflex and open them.

2. Q: My data are stored in a folder “my_data_dir” on my C: disk. How can I configure EsoReflex to use them?
   A: Files on disk C: can be accessed from within WSL as /mnt/c/my_data_dir. This path should be specified as EsoReflex’ BOOKKEEPING_DIR either from the command line or on the workflow. For further details, see any of the Reflex tutorials.
3. Q: The Ubuntu window does not support a feature I use in my Linux terminal emulator. How can I work around it?
   A: Any terminal emulator available under Ubuntu can be installed and used. For example, xterm can be installed from the ubuntu window using “sudo apt-get xterm”.
4. Q: I want to use some other astronomical software to display or examine the files within EsoReflex. How can I do this?
   A: Virtually any software that is available under Ubuntu such as DS9 can be installed and used using the regular Linux installing procedure (e.g. apt-get).
5. Q: When starting ESOReflex, I get the error message "Can't connect to X11 window server using 'localhost:0.0' as the value of the DISPLAY variable." What is going on?
   A: This means that the Xserver is not waiting for input as local display 0.0. Make sure that VcXsrv is installed and started as described in Step 2. You might want to test the Xserver installation by starting some other program such as xterm from the command line. ESOReflex will not be able to start as long as the communication with the Xserver does not work.

(*) The original version of this article has been kindly contributed by Wolfram Freudling and Pedro Fluxa.

Tags

Data Reduction

Windows