

Running PWI4 under Linux

Contents

Introduction	2
Installing and running PWI4.....	2
Installing the PlateSolve3 star catalog.....	3
Installing Mono in CentOS 7	4
Installing Mono in Ubuntu	6
Identifying Serial Ports and Setting Permissions (L-series mounts).....	7

Introduction

PWI4 runs in Linux using the Mono runtime environment. Similar to how programs written in Python generally require a Python environment to be installed on the computer, or programs written in Java require the Java runtime environment to be installed, Mono will need to be installed on your Linux system in order to run PWI4.

For more information about Mono, please refer to <https://www.mono-project.com/>

Installing and running PWI4

Prior to running PWI4, make sure that the necessary Mono packages are installed on your system. Instructions for various Linux distributions are included below.

Once Mono is installed, download the .tar.gz archive containing the PWI4 program files. In the example below, we will assume the file is named `pwi-4.0.3beta7.tar.gz`

Change to the directory where you would like the PWI4 application to be located. For example, to keep the program in your home directory, type:

```
cd ~
```

Extract the archive:

```
tar zxvf pwi-4.0.3beta7.tar.gz
```

A subdirectory will be created containing the PWI4 application. In this case, the subdirectory will be named `pwi-4.0.3beta7`.

Change to that directory:

```
cd pwi-4.0.3beta7
```

Run PWI4:

```
./run-pwi4
```

Installing the PlateSolve3 star catalog

PlateSolve3 is a program that matches stars detected in an image to a catalog of known stars. Without any prior knowledge of where the telescope was pointing when taking an image, PlateSolve3 can determine the center coordinates of the image. This is particularly useful when building a pointing model for a telescope.

The star catalog used by PlateSolve3 is based on the U.S. Naval Observatory's UCAC4 catalog. The files used by PlateSolve3 require approximately 3.8 GB of disk space.

A .zip archive of the star catalog can be downloaded from:

<http://planewave.com/files/catalogs/Kepler.zip>

By default, PWI4 and the command-line version of PlateSolve3 (ps3cli) will look for a directory named "Kepler" in the user's home directory. The Kepler directory will contain (at minimum) 2 subdirectories named "UC4" and "Orca", and may optionally include a subdirectory named "UC4Mag14" to aid in wide-field platesolves.

Once the file is downloaded, you can extract it in your home directory as follows:

```
cd ~  
  
unzip path/to/Kepler.zip  
  
ls ~/Kepler/      # Should show Orca, UC4, and UC4Mag14
```

Installing Mono in CentOS 7

The instructions below are based on the following:

<https://www.mono-project.com/download/stable/#download-lin-centos>

The mono packages for CentOS are hosted in a package repository that is not included by default with CentOS 7. It is necessary to first add the repository so that the yum package manager can locate and install the packages.

1. Open a shell with root permissions.
For example, from an existing shell, use the command `su` or `sudo -i` to access a root shell.
2. Import the key that is used to sign the mono packages. This allows the system to verify the authenticity of the downloaded packages.

(Note: the following command should be entered on a single line)

```
rpm --import  
"https://keyserver.ubuntu.com/pks/lookup?op=get&search=0x3FA7E032808  
1BFF6A14DA29AA6A19B38D3D831EF"
```

3. (optional) Verify that the key was installed.

```
rpm -qa | grep 831ef
```

Expected output: `gpg-pubkey-d3d831ef-53dfa827`

4. Add the mono/centos7 package repository to the system:

(Note: the following command should be entered on a single line. Also, simply prepending the command with `sudo` will not work correctly since the piped output will not be written with elevated permissions. It is easiest to run the command from a root shell.)

```
curl https://download.mono-project.com/repo/centos7-stable.repo |  
tee /etc/yum.repos.d/mono-centos7-stable.repo
```

5. (optional) Verify that the repository was added.

```
ls /etc/yum.repos.d/mono*
```

Expected output: `/etc/yum.repos.d/mono-centos7-stable.repo`

6. Install the required mono packages and dependencies:

```
yum install mono-devel mono-basic.noarch
```

Installing Mono in Ubuntu

The latest recommended instructions for installing Mono under Ubuntu can be found on the official Mono project website:

<https://www.mono-project.com/download/stable/#download-lin-ubuntu>

The following installation method has been tested under **Ubuntu 18.04**:

```
sudo apt install gnupg ca-certificates
```

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys  
3FA7E0328081BFF6A14DA29AA6A19B38D3D831EF
```

```
echo "deb https://download.mono-project.com/repo/ubuntu stable-bionic  
main" | sudo tee /etc/apt/sources.list.d/mono-official-stable.list
```

```
sudo apt update
```

```
sudo apt install mono-devel
```

```
sudo apt install mono-basic-dbg NOTE: Required, but not mentioned in linked page
```

Identifying Serial Ports and Setting Permissions (L-series mounts)

When an L-series mount is plugged in via USB, the system will identify **two virtual serial devices**. The names and permissions on these ports will vary from one distribution to the next.

You can use the **dmesg** command to look at kernel log messages shortly after plugging in or powering on the L mount. This will often give you a clue to help identify the names of these ports.

```
kmi@kmi-vbox:~$ dmesg | tail
[ 1871.345505] usb 1-2: new full-speed USB device number 5 using ohci-pci
[ 1871.662607] usb 1-2: New USB device found, idVendor=1cbe, idProduct=0267, bcd
Device= 1.00
[ 1871.662609] usb 1-2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 1871.662610] usb 1-2: Product: Virtual COM Port
[ 1871.662611] usb 1-2: Manufacturer: PlaneWave Instruments
[ 1871.662612] usb 1-2: SerialNumber: 12345678
[ 1871.865660] cdc_acm 1-2:1.0: ttyACM0: USB ACM device
[ 1871.882929] cdc_acm 1-2:1.2: ttyACM1: USB ACM device
[ 1871.890535] usbcore: registered new interface driver cdc_acm
[ 1871.890536] cdc_acm: USB Abstract Control Model driver for USB modems and ISD
N adapters
kmi@kmi-vbox:~$
```

On **Ubuntu 18.04**, these ports are named **/dev/ttyACM0** and **/dev/ttyACM1**, and they are owned by the **dialout** group:

```
kmi@kmi-vbox:~$ ls -l /dev/ttyACM*
crw-rw---- 1 root dialout 166, 0 Jul 29 11:03 /dev/ttyACM0
crw-rw---- 1 root dialout 166, 1 Jul 29 11:03 /dev/ttyACM1
kmi@kmi-vbox:~$
```

In order to access these ports, your user account must also be a member of the **dialout** group. This is not true by default:

```
kmi@kmi-vbox:~$ groups
kmi adm cdrom sudo dip plugdev lpadmin sambashare
kmi@kmi-vbox:~$
```

To add yourself to this group, run the command:

```
sudo usermod -a -G dialout $USER
```

Then you will need to log out and log back in for the new group membership to take effect. As a shortcut, you can temporarily start a new login session with the updated group membership using the following command:

```
su - $USER
```

For example, the following session shows that the user is not identified as being a member of **dialout** until a new login session is started via “su”:

```
kmi@kmi-vbox:~$ groups
kmi adm cdrom sudo dip plugdev lpadmin sambashare
kmi@kmi-vbox:~$ sudo usermod -a -G dialout $USER
kmi@kmi-vbox:~$ groups
kmi adm cdrom sudo dip plugdev lpadmin sambashare
kmi@kmi-vbox:~$ su - $USER
Password:
kmi@kmi-vbox:~$ groups
kmi adm dialout cdrom sudo dip plugdev lpadmin sambashare
kmi@kmi-vbox:~$
```