



Digital POWRR How-to Tech Tutorial

FITS-Demo Only In VM-Linux

This is a demonstration of how to run the File Information Toolset (FITS) in Ubuntu Linux inside of a Virtual Machine (VM). It is a command-line based tool. It is a tool that is available from Harvard: <https://projects.iq.harvard.edu/fits>.

What is FITS?

From the Harvard website:

<https://projects.iq.harvard.edu/fits/fitsfaq>

“FITS is a free and open source tool for identifying and validating file formats, extracting metadata embedded within files, and outputting the metadata in various formats. It was created to do some of the file processing tasks needed to support digital preservation repositories and applications.”

Where to download FITS?

<https://projects.iq.harvard.edu/fits/downloads>

Choose the latest version, usually found near the top of the page. As of the writing of this guide, the current release version is “fits-1.2.0.zip” released on August 15, 2017 at 128.9MB.

Download the file from the page above and save it to a flash drive so that we will have access to it later in Ubuntu Linux.

Pre-requisites:

Make a directory in your /home/YOURUSERNAME/ directory called DigitalCollectionFiles. This is where you will place all of your input files that will be processed in FITS.

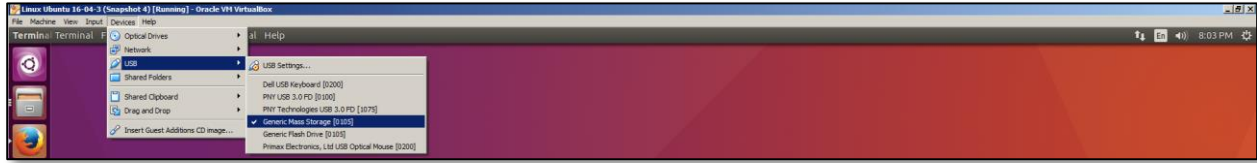
Open a Terminal Window in Linux by using the following key combination: (Ctrl-Alt-T)

Type the following into the terminal window:

- `cd ~`
This command will take you back to your home directory; this is the same location that you are located when you first log into Linux.
- `ls -l`
This command will list all of the files and directories in your home directory; check for a folder called DigitalCollectionFiles. If one is not already present (most likely it is not), then follow the next step. If DigitalCollectionFiles is already present then skip the next immediate step and proceed further.
- `mkdir DigitalCollectionFiles`
This command will create the directory DigitalCollectionFiles in your home directory. Only do this step if DigitalCollectionFiles folder is not already present.
- `ls -l`
Use this command to list all of the files and directories in your home directory again. Check to make sure that the DigitalCollectionFiles directory was created.

Locate a flash drive or other storage media that has a few files on the root directory of the flash drive that can be used for testing FITS in the Linux VirtualBox. Insert the flash drive into the HOST computer.

- On the VirtualBox File Menu, select **Devices** ➤ **USB** and then locate the flash drive that you just inserted into the HOST computer. Selecting it will place a checkmark next to that storage device and enable it for use on the Linux VM.

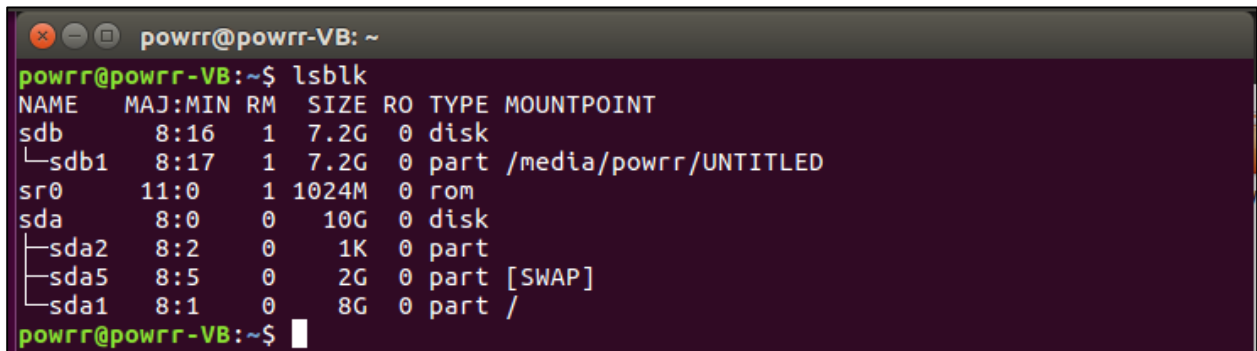


Open the Terminal Window (Ctrl-Alt-T),

Type the following:

- lsblk

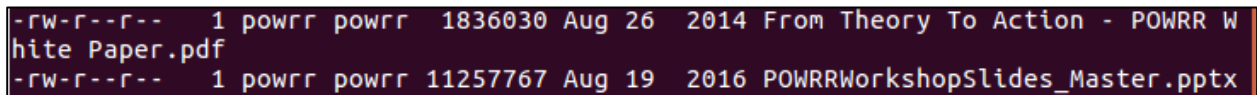
As you can see in the following screenshot, the flash drive was mounted as /media/powrr/UNTITLED



- cd /media/powrr/UNTITLED

This command will direct the command line to go to the flash drive.

- ls -l



As you can see in this screenshot from this example flash drive, there are two possible test files on the root directory of this flash drive that we could use for testing.

From Theory to Action – POWRR White Paper.pdf

POWRRWorkshopSlides_Master.pptx

We will now copy these two files over to the location /home/powrr/DigitalCollectionFiles (we had previously created this folder in our home directory; note that **powrr** should be replaced with your username).

- cp From*.pdf /home/powrr/DigitalCollectionFiles

- cp POWRR*.pptx /home/powrr/DigitalCollectionFiles

Please note that the use of an asterisk here is to wildcard the name so that we do not have to type the long filename, but we are still including the extension to limit the copying to these specific files in that directory.

- cd /home/powrr/DigitalCollectionFiles

After we have completed the copying, we will change directories to ensure that the two files are now in that directory.

- ls -l

```
powrr@powrr-VirtualBox: ~/DigitalCollectionFiles
powrr@powrr-VirtualBox:~/DigitalCollectionFiles$ ls -l
total 12792
-rw-r--r-- 1 powrr powrr 1836030 Oct  9 01:42 From Theory To Action - POWRR White Paper.pdf
-rw-r--r-- 1 powrr powrr 11257767 Oct  9 01:44 POWRRWorkshopSlides_Master.pptx
powrr@powrr-VirtualBox:~/DigitalCollectionFiles$
```

Make a directory in your /home/YOURUSERNAME/ directory called FitsOutputFiles. This is where you will place all of the output files that will be generated by FITS.

Open a Terminal Window in Linux by using the following key combination: (Ctrl-Alt-T)

Type the following into the terminal window:

- `cd ~`
This command will take you back to your home directory; this is the same location that you are located when you first log into Linux.
- `ls -l`
This command will list all of the files and directories in your home directory; check for a folder called FitsOutputFiles. If one is not already present (most likely it is not), then follow the next step. If FitsOutputFiles is already present then skip the next immediate step and proceed further.
- `mkdir FitsOutputFiles`
This command will create the directory FitsOutputFiles in your home directory. Only do this step if FitsOutputFiles folder is not already present.
- `ls -l`
Use this command to list all of the files and directories in your home directory again. Check to make sure that the FitsOutputFiles directory was created.

We have completed the process of copying some test files to be used as inputs in the FITS program later and created a folder that can be used to store the output files from FITS. Now we proceed to installing Java, which is required by the FITS program.

Pre-requisites: Java installation

FITS requires a Java Runtime Environment (JRE) of version 1.7 or higher. (OpenJDK version 9 does work with FITS) Furthermore, for MediaInfo to work, only Java 64-bit versions are supported. If you do not need to validate media items (audio and video filetypes), then FITS will work on non-media file types but may throw exception errors in the command prompt window regarding MediaInfo while running (please see the end of this document for an example). However, it will still process non-media file types, and produce XML output files that can be viewed.

Here is how to check to see if the Ubuntu Linux OS version installed is 32-bit or 64-bit. Open a Terminal Window in Linux (Ctrl-Alt-T) and type "**uname -a**".

If it is a 32-bit Linux system, you will see something similar to this:

```
Linux powrr-VB 4.10.0-28-generic #32~16.04.2-Ubuntu SMP Thu Jul 20 10:19:48 UTC 2017 i686 i686 i386 GNU/Linux
```

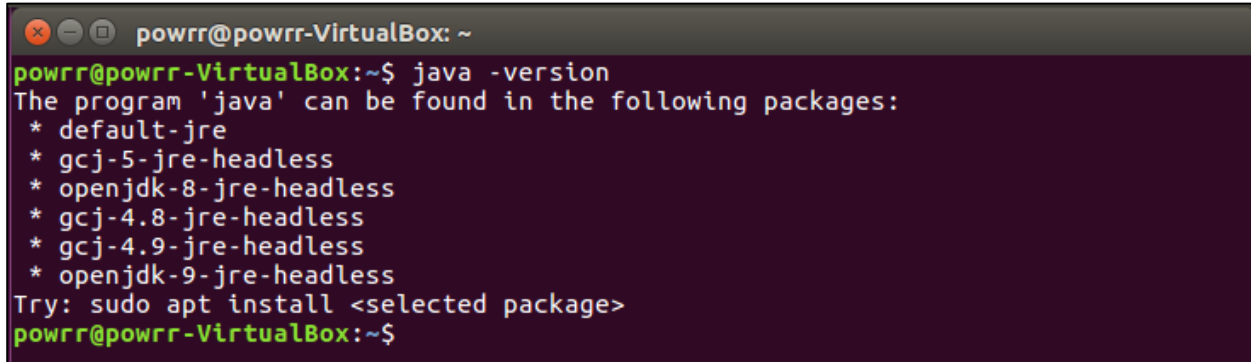
If it is a 64-bit Linux system, you will see:

```
Linux powrr-VB 4.10.0-28-generic #32~16.04.2-Ubuntu SMP Thu Jul 20 10:19:48 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
```

For the purposes of simplicity, this guide will assume that a 64-bit version of Ubuntu Linux is installed.

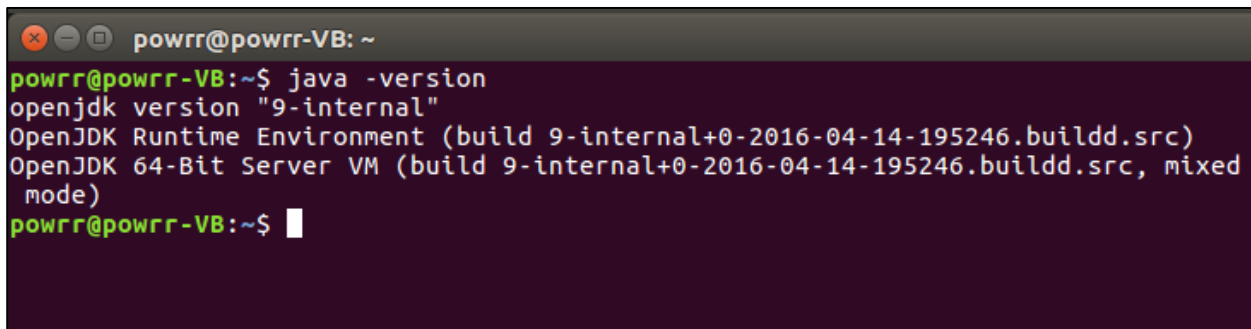
Open a Terminal Window in Linux (Ctrl-Alt-T) and type "**java -version**". We will use this command to see if Java is installed.

If Java is not installed, you will see something similar to the following screen. Please install Java using the instructions listed in the next section: Installing Java.



```
powrr@powrr-VirtualBox: ~
powrr@powrr-VirtualBox:~$ java -version
The program 'java' can be found in the following packages:
 * default-jre
 * gcj-5-jre-headless
 * openjdk-8-jre-headless
 * gcj-4.8-jre-headless
 * gcj-4.9-jre-headless
 * openjdk-9-jre-headless
Try: sudo apt install <selected package>
powrr@powrr-VirtualBox:~$
```

If Java is already installed, you will see something similar to the following screen. You can skip the "Installing Java" section below.



```
powrr@powrr-VB: ~
powrr@powrr-VB:~$ java -version
openjdk version "9-internal"
OpenJDK Runtime Environment (build 9-internal+0-2016-04-14-195246.buildd.src)
OpenJDK 64-Bit Server VM (build 9-internal+0-2016-04-14-195246.buildd.src, mixed
mode)
powrr@powrr-VB:~$
```

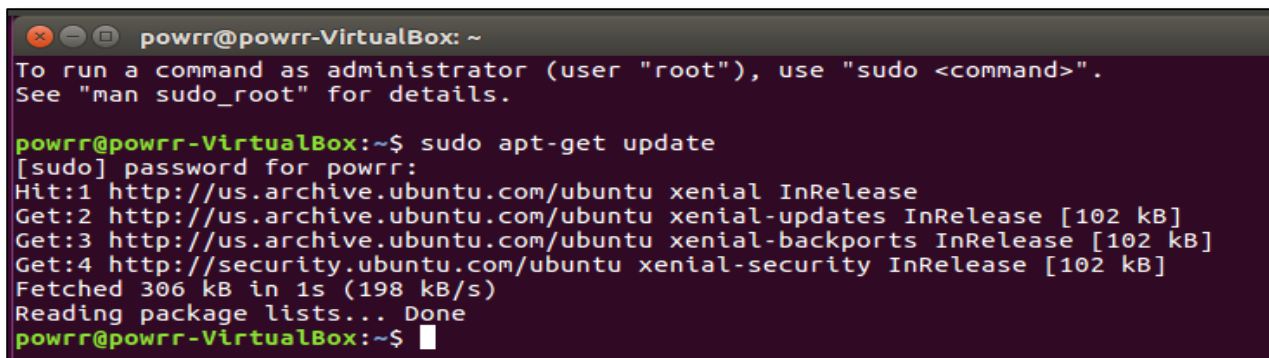
Installing Java:

First, we will update the apt-get packages before installing Java. Software is constantly being updated. Your Ubuntu Linux installation's listing of what is the current software available to be downloaded may be out-of-date. Updating the apt-get packages will allow Ubuntu to know which installer packages are currently available.

In the terminal window type: **sudo apt-get update**

Ubuntu will ask for the password. Type your password; hit the Enter key.

As you can see from the following screenshot, the apt-get packages have been updated.



```
powrr@powrr-VirtualBox: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

powrr@powrr-VirtualBox:~$ sudo apt-get update
[sudo] password for powrr:
Hit:1 http://us.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Get:4 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Fetched 306 kB in 1s (198 kB/s)
Reading package lists... Done
powrr@powrr-VirtualBox:~$
```

Next, in the terminal window type: **java -version**

```
powrr@powrr-VirtualBox: ~
powrr@powrr-VirtualBox:~$ java -version
The program 'java' can be found in the following packages:
* default-jre
* gcj-5-jre-headless
* openjdk-8-jre-headless
* gcj-4.8-jre-headless
* gcj-4.9-jre-headless
* openjdk-9-jre-headless
Try: sudo apt install <selected package>
powrr@powrr-VirtualBox:~$
```

We will install the OpenJDK version 9 of the Java Runtime Environment as it works with FITS.
As the terminal window suggests, type: **sudo apt install openjdk-9-jre-headless**
If the terminal requests your password, please type it in and hit Enter.

```
powrr@powrr-VirtualBox: ~
powrr@powrr-VirtualBox:~$ java -version
The program 'java' can be found in the following packages:
* default-jre
* gcj-5-jre-headless
* openjdk-8-jre-headless
* gcj-4.8-jre-headless
* gcj-4.9-jre-headless
* openjdk-9-jre-headless
Try: sudo apt install <selected package>
powrr@powrr-VirtualBox:~$ sudo apt install openjdk-9-jre-headless
[sudo] password for powrr:
```

The terminal will ask if you would like to continue the installation, type: **y**.

```
powrr@powrr-VirtualBox:~$ sudo apt install openjdk-9-jre-headless
[sudo] password for powrr:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java java-common
Suggested packages:
  default-jre fonts-dejavu-extra fonts-ipafont-gothic fonts-ipafont-mincho
  ttf-wqy-microhei | ttf-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java java-common openjdk-9-jre-headless
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 183 MB of archives.
After this operation, 307 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

When the installation completes, you will see done at the bottom and the command prompt will return to its normal blinking cursor.

```
powrr@powrr-VirtualBox: ~
Adding debian:TeliaSonera_Root_CA_v1.pem
Adding debian:thawte_Primary_Root_CA_-_G3.pem
Adding debian:Swisscom_Root_CA_1.pem
Adding debian:GeoTrust_Primary_Certification_Authority_-_G3.pem
Adding debian:DigiCert_Global_Root_CA.pem
Adding debian:SwissSign_Gold_CA_-_G2.pem
Adding debian:Amazon_Root_CA_4.pem
Adding debian:GeoTrust_Primary_Certification_Authority_-_G2.pem
Adding debian:Taiwan_GRCA.pem
Adding debian:PSCProcert.pem
Adding debian:Certinomis_-_Autorité_Racine.pem
done.
Processing triggers for ca-certificates (20170717~16.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
done.
powrr@powrr-VirtualBox:~$
```


As a confirmation that Java installed properly, type: **java -version**

You will see a screen similar to the screenshot below if Java installed properly. Now FITS is ready to be installed and run.

```
powrr@powrr-VB: ~
powrr@powrr-VB:~$ java -version
openjdk version "9-internal"
OpenJDK Runtime Environment (build 9-internal+0-2016-04-14-195246.buildd.src)
OpenJDK 64-Bit Server VM (build 9-internal+0-2016-04-14-195246.buildd.src, mixed
mode)
powrr@powrr-VB:~$
```

Installing FITS:

Earlier, the FITS installer zip file was downloaded and saved to a flash drive. Insert that flash drive into the HOST computer so that it can be accessed from the Ubuntu Linux VM.

- On the VirtualBox File Menu, select **Devices**  **USB** and then locate the flash drive that you just inserted into the HOST computer. Selecting it will place a checkmark next to that storage device and enable it for use on the Linux VM.



Start up a Terminal Window (Ctrl-Alt-T),

Type the following:

- lsblk

As you can see in the following screenshot, the flash drive was mounted as /media/powrr/UNTITLED


```
powrr@powrr-VB: ~
powrr@powrr-VB:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO  TYPE MOUNTPOINT
sdb          8:16   1   7.2G  0  disk
└─sdb1       8:17   1   7.2G  0  part /media/powrr/UNTITLED
sr0         11:0    1 1024M  0  rom
sda          8:0    0   10G  0  disk
├─sda2       8:2    0    1K  0  part
├─sda5       8:5    0    2G  0  part [SWAP]
└─sda1       8:1    0    8G  0  part /
powrr@powrr-VB:~$
```

- `cd /media/powrr/UNTITLED`
This command will direct the command line to go to the flash drive.

- `ls -l`

```
-rw-r--r-- 1 powrr powrr 122894643 Oct  9 02:42 fits-1.2.0.zip
```

As you can see on this flash drive, the fits-1.2.0.zip file is present on the drive (it was downloaded earlier from the Internet and placed onto this drive).

We will now copy this file over to the location /home/powrr/ (note that **powrr** should be replaced with your username).

- `cp fits-1.2.0.zip /home/powrr/`
- `cd /home/powrr/`
After we have completed the copying, we will change directories to ensure that the zip file is in that directory.
- `ls -l`

```
powrr@powrr-VirtualBox: ~
powrr@powrr-VirtualBox:~$ ls -l
total 120068
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Desktop
drwxrwxr-x 2 powrr powrr 4096 Oct  9 01:44 DigitalCollectionFiles
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Documents
drwxr-xr-x 2 powrr powrr 4096 Oct  9 02:44 Downloads
-rw-r--r-- 1 powrr powrr 8980 Oct  8 22:36 examples.desktop
-rw-r--r-- 1 powrr powrr 122894643 Oct  9 02:49 fits-1.2.0.zip
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Music
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Pictures
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Public
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Templates
drwxr-xr-x 2 powrr powrr 4096 Oct  8 22:46 Videos
powrr@powrr-VirtualBox:~$
```

Now, extract the zip file using the following command (it should take about 1 minute):

- `unzip fits-1.2.0.zip -d /home/powrr/Fits`

```
powrr@powrr-VirtualBox: ~
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/jpg.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/logfile.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/mp3.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/mpeg4.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/msworks.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/nlnz_file.xsd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/nlnz_presmet.xsd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/nlnz_types.xsd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/openoffice.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/pdf.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/pdfbox.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/powerpoint.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/tiff_6_0.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/warc.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/wav.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/word.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/wordperfect.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/nlnz/xml.dtd
inflating: /home/powrr/Fits/fits-1.2.0/xml/prettyprint.xslt
inflating: /home/powrr/Fits/fits-1.2.0/xml/xslt_map.xsd
inflating: /home/powrr/Fits/fits-1.2.0/fits-env.sh
inflating: /home/powrr/Fits/fits-1.2.0/fits-ngserver.sh
inflating: /home/powrr/Fits/fits-1.2.0/fits.sh
powrr@powrr-VirtualBox:~$
```

Once the unzipping is complete, then you will be returned to the blinking cursor on the command line.

- `cd Fits`
This command will change directories into the Fits directory.
- `ls -l`
This command allows you to see the contents of the Fits directory.

This concludes the installation portion of FITS. The program is now ready to run.

DEMONSTRATION BEGINS HERE

Running FITS:

The FITS program was previously installed in your home directory. It should be in `/home/YOURUSERNAME/Fits`. If a Terminal Window is not open already, open a Terminal Window in Linux (Ctrl-Alt-T) and navigate to your Fits folder. Type the following commands:

- `pwd`
Use this command to see where you are currently located. You should be in `/home/USERNAME`. If not, please type `"cd ~"`. That will take you back to your home directory.
- `ls -l`
Use this command to see what folders are present. The Fits folder should be there.
- `cd Fits`
Use this command to change directories to the Fits folder.
- `ls -l`
Use this command to see what is inside of the Fits folder. You will most likely see a subfolder called `fits-1.2.0` (that folder will be slightly different if a different version of FITS was installed, use the folder name you see present in your terminal window)
- `cd fits-1.2.0`
Use this command to change directories to the `fits-1.2.0` folder.

- `ls -l`

Use this command to see what is inside of the fits-1.2.0 folder. This should be the folder that contains the fits.sh file that is needed to start FITS in Linux. Please look at this small screenshot that shows the fits.sh file is present.

```
-rwxr-xr-x 1 powrr powrr 474 Aug 16 15:44 fits.sh
```

Also, note that on the left side of this line, the permissions for this file start with “-rwx”. These first four characters of the file permissions tell you whether you can read, write, or execute the file as the owner of the file. The dash indicates that this is a file and not a directory. The r represents that the owner of the file (you) can read it; the w indicates that the owner of the file (you) can write to the file; the x indicates that the owner of the file (you) can ‘execute’ the file or run it. If the “x” is not present (you only see -rw-), then you will need to run the following command to give the fits.sh file the ‘execute’ permission, otherwise, FITS will not run.

- `Chmod +x fits.sh`

- `ls -l`

Use this command to review the change made to execute permission on the fits.sh file. Make sure that the permissions now show as -rwx for that file.

- `./fits.sh`

Use this command to start the FITS program. The information returned will explain briefly how to run the program. Below should be what you see.

```
powrr@powrr-VB:~/Fits/fits-1.2.0$ ./fits.sh
Invalid CLI options
usage: fits
-h          print this message
-i <arg>   input file or directory
-o <arg>   output file or directory if -i is a directory
-r         process directories recursively when -i is a directory
-v         print version information
-x         convert FITS output to a standard metadata schema
-xc       output using a standard metadata schema and include FITS xml
powrr@powrr-VB:~/Fits/fits-1.2.0$
```

Based on the above information returned from the program, to run FITS, you will need to provide the program with the following:

- 1) An input file or directory (that is the -i argument) that we wish to process through FITS. The input directory was created earlier; it is called: **DigitalCollectionFiles**. For the username powrr, it is located in **/home/powrr/DigitalCollectionFiles**.
- 2) An output file or directory (that is the -o argument) that we wish to process through FITS. The output directory was created earlier; it is called **FitsOutputFiles**. For the username powrr, it is located in **/home/powrr/FitsOutputFiles**
- 3) Optionally, the -r argument is included if a directory is given to the program so that directories are processed recursively.

For example, if our input directory of files to be processed are located in /home/powrr/DigitalCollectionFiles, then the following command will run FITS on those files and place the output files in /home/powrr/FitsOutputFiles:

```
./fits.sh -i /home/powrr/DigitalCollectionFiles -o /home/powrr/FitsOutputFiles
```

```
powrr@powrr-VirtualBox:~/Fits/fits-1.2.0$ ./fits.sh -i /home/powrr/DigitalCollectionFiles -o /home/powrr/FitsOutputFiles
```

As you can see from this screenshot above, the program is now running.

```
powrr@powrr-VirtualBox:~/Fits/fits-1.2.0$ ./fits.sh -i /home/powrr/DigitalCollectionFiles -o /home/powrr/FitsOutputFiles
powrr@powrr-VirtualBox:~/Fits/fits-1.2.0$
```

When the program completes, the command prompt will return to the cursor as you see above.

- `cd /home/powrr/FitsOutputFiles`

Switch to the FitsOutputFiles to look at the XML file created by FITS.

- `ls -l`

Look at the contents of that output directory.

```
powrr@powrr-VirtualBox: ~/FitsOutputFiles
powrr@powrr-VirtualBox:~/FitsOutputFiles$ ls -l
total 12
-rw-rw-r-- 1 powrr powrr 5705 Oct  9 03:28 From Theory To Action - POWRR White Paper.pdf.fits.xml
-rw-rw-r-- 1 powrr powrr 3916 Oct  9 03:28 POWRRWorkshopSlides_Master.pptx.fits.xml
powrr@powrr-VirtualBox:~/FitsOutputFiles$
```

- `vi From*.xml`
This command will open up the From Theory to Action...xml file in the “vi” editor.
- `:q!`
Use this command in vi to quit the editor.

You have now successfully installed and run FITS in Ubuntu Linux via the command line.