

## Preparation:

On your Windows10 system enable the Windows Subsystem for Linux.

Open the powershell as administrator. Do do so type powershell in the search window, right click on it and choose to run as administrator. Execute the command:

**Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux**

You should now reboot your computer.

Open the Microsoft Store, search for Ubuntu and choose Ubuntu 18.04 LTS and click on Install.

The Ubuntu system will open with a new window and ask:

**Enter a new UNIX username:**

Choose any suitable name, which need not coincide with your Windows user name.

**Enter new UNIX password:**

**Retype new UNIX password:**

The password need not and probably should not coincide with your Windows password.

The compilation requires several libraries, especially their development versions, not all of them may be installed automatically. Especially Ubuntu leaves off many development versions.... Some of these libraries might be present on your system with different version numbers or slightly different names/paths, please be flexible:

libX11-devel	! X11 development library
libm	! Usually installed
libreadline6-dev	! Needed for command editing
libpng16-devel	! PNG Graphics library currently version 16
	! The current version is 16, but older versions seem to work fine
	! as well
libpgplot	! Might already be installed, Best left off, as the installation script
	! does provide a version and installs it.
	! See below
OpenMP	! DIFFUSE needs this for parallel processing
openmpi-dev	! DIFFUSE needs this for parallel processing

### Optional Libraries

Python	
mpich	! An alternative, OpenMP is preferred
	! DIFFUSE needs this for parallel processing
NeXuS	! In the future DISCUS will use this for 3D data

### Mandatory development tools:

cmake, cmake, make	
gcc	
gfortran	! At least Version 4.6

As an example, with UBUNTU 16.04 or 18.04 use apt-get or simply apt to install packages:

```
sudo apt-get install build-essential
sudo apt-get install libx11-dev
sudo apt-get install libxmu-dev
sudo apt-get install libreadline-dev
sudo apt-get install libopenmpi-dev
sudo apt-get install libopenmpi-bin
sudo apt-get install gfortran
sudo apt-get install cmake
sudo apt-get install cmake-curses-gui
```

For the interactive plot in DISCUS I prefer Jmol, which requires java as well.

```
sudo apt-get install jmol
```

As of Ubuntu 18.04 Open Java should be installed with the default system installation.

## Installation:

### One touch installation

Copy the current DISCUS\_INSTALL-v.5.99.13.tar.gz to your home directory

```
cp Downloads/DISCUS_INSTALL-v.5.99.13.tar.gz $HOME
```

If necessary adjust the source path...

Unpack this archive:

```
cd $HOME
tar -zxf DISCUS_INSTALL-v.5.99.13.tar.gz
```

This will create a directory called **DIFFUSE\_INSTALL**.

Copy the current source code archive into this directory

The archive is called DiffuseCode-vV.M.P.tar.gz, where V.M.P stands for the major Version, the Minor version and the Patch numbers, currently 5.99.13

```
cd $HOME/DIFFUSE_INSTALL
cp Downloads/DiffuseCode-5.99.13.tar.gz .
```

Install everything including the PGPLOT library by running the shell script install\_discus\_suite.sh:

```
sudo ./install_discus_suite.sh DiffuseCode-5.99.13.tar.gz
or ./install_discus_suite.sh DiffuseCode-5.99.13.tar.gz
```

If run with “sudo”, the install\_discus\_suite.sh script will copy the PGPLOT source code into **/usr/local/src**, the PGPLOT installation directory into **/usr/local/pgplot**. The discus\_suite program sources will be placed into **\$HOME/DIFFUSE\_INSTALL/develop/** and **\$HOME/DIFFUSE\_INSTALL/develop/DiffuseBuild**. The compiled programs will be in **/usr/local/bin**.

Else, PGPLOT will be installed locally into your HOME directory at **\$HOME/pgplot** and the programs be installed into **\$HOME/bin**, with the manuals and on-line help in **\$HOME/share**. Feel free to adjust paths.

In order to use the PGPLOT library, you need the following environment variables:

See the files in **DIFFUSE\_INSTALL/SHELL** for templates. The installation script will place these environment variables into **\$HOME/.profile.local** and in addition, if run as sudo into **/etc/profile.d/profile.local**. If your local **\$HOME/.bashrc** exists, a line will source **\$HOME/.profile.local** will be appended

```
PGPLOT_DIR="/usr/local/pgplot"; export PGPLOT_DIR
PGPLOT_DEV="/XSERVE"; export PGPLOT_DEV
PGPLOT_FONT="/usr/local/pgplot/grfont.dat"; export PGPLOT_FONT
```

define these within **\$HOME/.bashrc.local** if a „bash“ is used.



## Individual manual Installation:

Download the latest source code archive from GitHub at:

[github.com/tproffen/DiffuseCode/releases](https://github.com/tproffen/DiffuseCode/releases)

The archive is called DiffuseCode-V.M.P.tar.gz, where V.M.P stands for the major Version, the Minor version and the Patch numbers, currently 5.29.0

Copy the source code archive to a suitable directory and unpack:

```
mkdir -p $HOME/develop
cp DiffuseCode-5.29.0.tar.gz $HOME/develop
cd $HOME/develop
tar -zxf DiffuseCode-5.29.0.tar.gz
```

create a „build“ directory, and change to build directory:

```
mkdir -p $HOME/develop/DiffuseBuild
cd $HOME/develop/DiffuseBuild
```

execute cmake with source code directory as parameter.  
cmake should open a graphical interface:

```
cmake ../DiffuseCode-5.29.0/
```

cmake operates mostly via one letter commands, the main are:

```
c    for configure
e    exit the message screen
g    to generate the make files and exit cmake
```

In cmake toggle OFF the options:

DIFFUSE\_PYTHON, DISCUS\_CUDA, DISCUS\_NEXUS, DISCUS\_OMP

In cmake toggle ON the options:

DIFFEV\_MPI

press „t“ to toggle to advanced mode. Go down with cursor and inspect pgplot settings  
they should point to the directory in which the pgplot library is found:  
/usr/local/pgplot OR may be: /usr/local/lib64/pgplot

The pgplot library need at least the following files in this directory:

```
grfont.dat
libcpgplot.a or libpgplot.so
libpgplot.a or libpgplot.so
pgxwin_server
```

Especially if you use a pgplot installation provided by the linux system, these files might be in different directories. It might be best to create a directory

/usr/local/pgplot

and to copy these files into this directory or to create symbolic links within this directory that point to the actual files.

To edit an entry within ccmake hit the „Enter key“ then type or change text.

ccmake wants an entry for „CMAKE\_BUILD\_TYPE“, edit this field and leave it blank.

Once done hit „c“ to configure ccmake

You will get an info screen with hopefully no error messages.

If errors are listed, type „e“ and then „q“ and fix the error

If no errors occur hit „e“ to leave the info screen

Hit „g“ to generate the actual make files and to exit ccmake

then you need to compile the program, type without options

**make**

If this worked out without error messages you can install DISCUS, DIFFEV etc.

Our default installation directory is /usr/local/bin thus you can:

A) do it with „sudo“

B) Change the ownership of /usr/local to your own account:

A)

**sudo make install**

B)

**sudo chown -R /usr/local your\_user\_name**

**make install**

With UBUNTU the style B) seems to create issues with other packages and is discouraged.

To clean up type

**make clean**

for the on-line help to work, a couple of environment variables should be set:

**PGPLOT\_DIR="/usr/local/pgplot"; export PGPLOT\_DIR**

**PGPLOT\_DEV="/XSERVE"; export PGPLOT\_DEV**

**PGPLOT\_FONT="/usr/local/pgplot/grfont.dat"; export PGPLOT\_FONT**

define these within \$HOME/.bashrc.local if a „bash“ is used.

## PGPLOT Library

With Ubuntu the PGPLOT library can be installed using the package manager but needs fine tuning. On other systems you might have to install the PGPLOT library locally.

PGPLOT library needs the files:

libpgplot.a  
libpgplot.so  
grfont.dat  
pgxwin\_server

in the directory under the PGPLOT\_INCLUDE\_DIR entry in ccmake you need the files:

cpgplot.h  
grpckg1.inc  
pgplot.inc  
pgxwin\_server

---

Manual installation of LIBPGPLOT:

Make /usr/local your own, or proceed throughout with sudo. This is best on an UBUNTU system, as a change of ownership seems to interfere with other packages and their updates.... Changing the ownership is not an issue with SuSe

**NOT on UBUNTU:** `sudo chown -R your_user_name /usr/local`

INSTALL x11-dev, libreadline6, libpng, libpng-dev

I made small adjustments to the PGPlot library to ease the installation on a linux system. Please use the archive **DISCUS\_INSTALLATION\_CYGWIN.tar.gz** from GITHUB along with the DIFFUSE source code. The following instructions apply to this version.

copy **DISCUS\_INSTALLATION\_CYGWIN.tar.gz** to your home directory and unpack:

```
cp DISCUS_INSTALLATION_CYGWIN.tar.gz $HOME
tar -zxf DISCUS_INSTALLATION_CYGWIN.tar.gz
cd DIFFUSE_INSTALLATION
tar -zxf DIFFUSE_CODE_pgplot.tar.gz
```

You will have a good template for the pgplot installation within DIFFUSE\_INSTALL. See the directories **pgplot** and **src/pgplot** within **DIFFUSE\_INSTALLATION**.

Unless you want to change the selection of graphics drivers for pgplot, I recommend to proceed with the compilation script: `compile_pgplot_linux.sh` :

```
sudo ./compile_pgplot_linux.sh $(pwd)
```

For a fully manual installation do:

```
sudo mkdir -p /usr/local/src
sudo cp -r src/pgplot /usr/local/src
```

Create pgplot directory

```
sudo mkdir /usr/local/pgplot
cd /usr/local/pgplot
```

Copy your drivers.list to /usr/local/pgplot

```
sudo cp /usr/local/src/pgplot/drivers.list /usr/local/pgplot
```

Edit drivers.list and uncomment any driver you need. The DISCUS\_SUITE needs at least

```
NUDRIV
PSDRIV 1 to 4
PNDRIV 1
XWDRIV 1 and 2
```

Create makefile:

```
cd /usr/local/pgplot
sudo /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux gfortran_gcc
```

ONLY if you have the older version pgplot.5.2.tar.gz, do the following instead:

```
cd /usr/local/pgplot
sudo /usr/local/src/pgplot/makemake /usr/local/src/pgplot linux f77_gcc
```

Edit makefile

change line 25 and 26 to:

```
25: FCOMPL=gfortran
```

```
26: FFLAGC=-ffixed-form -ffixed-line-length-none -u -Wall -fPIC -O
```

```
48: Remove "-lf2c"
```

Copy line 875(?), change to  
pndriv.o:

Exit makefile

run makefile with:

```
sudo make
sudo make cpg
sudo make clean
```

If you use the „bash“ then



edit /etc/profile.d/profile.local to contain:

```
PGPLOT_DIR=/usr/local/pgplot
#PGPLOT_DEV=/XSERVE
PGPLOT_DEV=/XWINDOW
PGPLOT_FONT=/usr/local/pgplot/grfont.dat
export PGPLOT_DIR
export PGPLOT_DEV
export PGPLOT_FONT
```

Edit your local ".bashrc", add at end:  
source /etc/profile.d/profile.local

Alternatively you can of course edit your local .bashrc.local .

Note on current releases of png\_dev libpng16 and later. As the file pngconf has been modified, you might get an error while compiling file pndriv.c. If this occurs, please edit pndriv.c in the folder drivers and comment lines 225 to 233.

Finally run one or more of the pdgemo programs to verify that the installation proceeded properly. Sometimes, if graphics libraries are missing, the PGPLOT make file seems to quietly turn off the corresponding driver in drivers.list. In this case verify that you have installed the required graphics libraries, especially in their “devel” version. Make sure you edit drivers.list again before compiling the pgplot library.