



Installing and Running

FLUKA Beginner's Course

How to download and install FLUKA

Two ways of downloading the FLUKA software:

- From the FLUKA website <http://www.fluka.org>
- From NEA databank <http://www.nea.fr> through the liaison officer from your institute

It is **mandatory** to be registered as FLUKA user.

Follow the link:

<http://www.fluka.org/download.html>

After registration (or using your user-id and password) you can proceed in downloading the latest official release version.

How to download and install FLUKA

First identify the location of the FLUKA distribution file: `fluka2011.2-linuxAA.tar.gz`
Depending on the operating system and the method you used most probably will be located in one of the following directories:

```
    /media/FLUKA/Software           # in case you are using the USB stick  
or  $HOME                          # if you downloaded from the web  
    $HOME/Desktop                  # depending on your browser
```

We will create a directory FLUKA under your home directory to install FLUKA.
The following commands issued from a **terminal/console window** will perform the entire installation.

```
    cd                               # changes directory to your home  
    mkdir FLUKA                     # creates a directory called FLUKA  
    cd FLUKA                        # changes to the FLUKA directory  
    tar xzf /media/FLUKA/Software/fluka2011.2-linuxAA.tar.gz  
                                     # expands the FLUKA package  
  
    # set FLUPRO environment variable  
    export FLUPRO=$HOME/FLUKA      # sets FLUPRO in bash shell or similar  
or  setenv FLUPRO $HOME/FLUKA     # sets FLUPRO in tcsh shell or similar  
    make                            # compiles a FLUKA executable and  
                                     auxiliary programs
```

Persistent settings

To make these settings persistent on your computer, *i.e.*, you don't have to set the FLUPRO environment variable again when you open a new terminal or log into your computer, we will add the following lines into your shell configuration file in your main directory.

bash users:

```
cd
```

```
emacs [or any editor] .bashrc
```

add the following:

```
export FLUPRO=${HOME}/FLUKA
```

```
export PATH=${PATH}:${FLUPRO}:${FLUPRO}/flutil
```

tcsh users:

```
cd
```

```
emacs [or any editor] .tcshrc
```

add the following:

```
setenv FLUPRO ${HOME}/FLUKA
```

```
setenv PATH ${PATH}:${FLUPRO}:${FLUPRO}/flutil
```

The changes will be activated on the next login or if you type the command

```
source .bashrc
```

```
source .tcshrc
```

FLUKA release: main directory \$FLUPRO

Main Library:

libflukahp.a (object collection)

Physics data files:

sigmapl.bin
elasct.bin
brems_fin.bin
cohff.bin
gxsect.bin
neuxsc-ind_260.bin
nuclear.bin
fluodt.dat
e6r1nds3.fyi
jef2.fyi
jendl3.fyi
xnloan.dat
Fad/*
DDS/*

Basic Scripts: (in \$FLUPRO/flutil)

rfluka
lfluka
fff

Random Number seed

random.dat

Important Directories

flukapro/	all FLUKA commons
usermvax/	user routines
flutil/	general utilities

What's inside the physics data files:

sigmapl.bin:	pion-N double-diff. cross sections
elasct.bin:	elastic scattering cross sections
brems_fin.bin:	Bremsstrahlung cross sections
cohff.bin:	atomic form factor tabulations
gxsect.bin	photon cross sections
neuxsc-ind_260.bin:	low energy neutron multi-group cross sections (260 groups)
nuclear.bin:	nuclear masses, mass excesses, levels, and many other nuclear data for evaporation, pre- equilibrium, Fermi break up and photonuclear cross sections gamma and beta databases
fluodt.dat:	Fluorescence data (photoelectric effect)
e6r1nds3.fyi:	} Fission products (for neutrons with E<20 MeV)
jef2.fyi:	
jendl3.fyi:	
xnloan.dat:	
Fad/* :	BME pre-equilibrium particle angular distribution
DDS/* :	BME pre-equilibrium particle energy spectra

Available Documentation

- **fluka2011.manual** ASCII version of the manual (easy to edit)
- **FM.pdf** current version of the FLUKA manual
- **CERN-2005-10.pdf** official reference for FLUKA (manual not up to date)
- or navigate the manual, online version (www.fluka.org)
- or (when using FLAIR) press **F1** to get an interactive manual (which can be also called on prompt level by calling *'fm.py'*)
- or (at a further stage) the **FAQ** available at:
<http://www.fluka.org/fluka.php?id=faq&mm2=3>
- or (at a further stage) the archive of **fluka-discuss**:
<http://www.fluka.org/MailingList.html>
- **Release notes**

A New Way to "Go FLUKA" - **FLUPIX**

Besides the current FLUKA distribution you also got the current version of FLUPIX distributed on your USB stick.

FLUPIX

(FLUka in knopPIX)

© Vasilis.Vlachoudis@cern.ch 2008

FLUPIX is a Fedora (originally **KNOPPIX**, www.knoppix.org) based version of the Live CD, with pre-installed FLUKA and flair and all the necessary tools in for performing FLUKA runs.

FLUPIX can run from a CD/DVD, bootable USB or through any virtual machine from any host operating system (Ms Windows, Mac OS, Linux, Solaris etc.). FLUPIX includes all the additions of VirtualBox (www.virtualbox.org) a free and open source Virtual machine supported by Sun, that provides easy installation and high performance.

Introduction

- FLUPIX (**FLUKA** in **KNOPPIX***) is a bare-bones Live ISO containing:
 - **FLUKA**
 - **Flair**
 - All necessary tools for analysis (gnuplot, processing programs...)
 - **LXDE** – Lightweight Desktop
 - VirtualBox additions
- Minimal size of ~300MB
- The ISO is specially mastered to run under VirtualBox (www.virtualbox.org) an open source virtual machine by Sun. It is available under many platforms:
 - all Linux
 - MS Windows
 - Mac OS
 - OpenSolaris

* Originally it was based on knoppix. The present version is based on Fedora ₉

VirtualBox

- **VirtualBox** is an [x86 virtualization](#) software package originally created by Innotek and now being developed by [Sun Microsystems](#) as part of its [Sun xVM](#) virtualization platform.
- It is installed on an existing **host** operating system (OS); within this application, additional operating systems, each known as a *Guest OS*, can be loaded and run, each with its own virtual environment.
- For example, [Linux](#) can be guest hosted on a single virtual machine running [Microsoft Windows XP](#) as the *Host OS*; or, XP and [Windows Vista](#) can run as guest operating systems on a machine running [OpenSolaris](#).

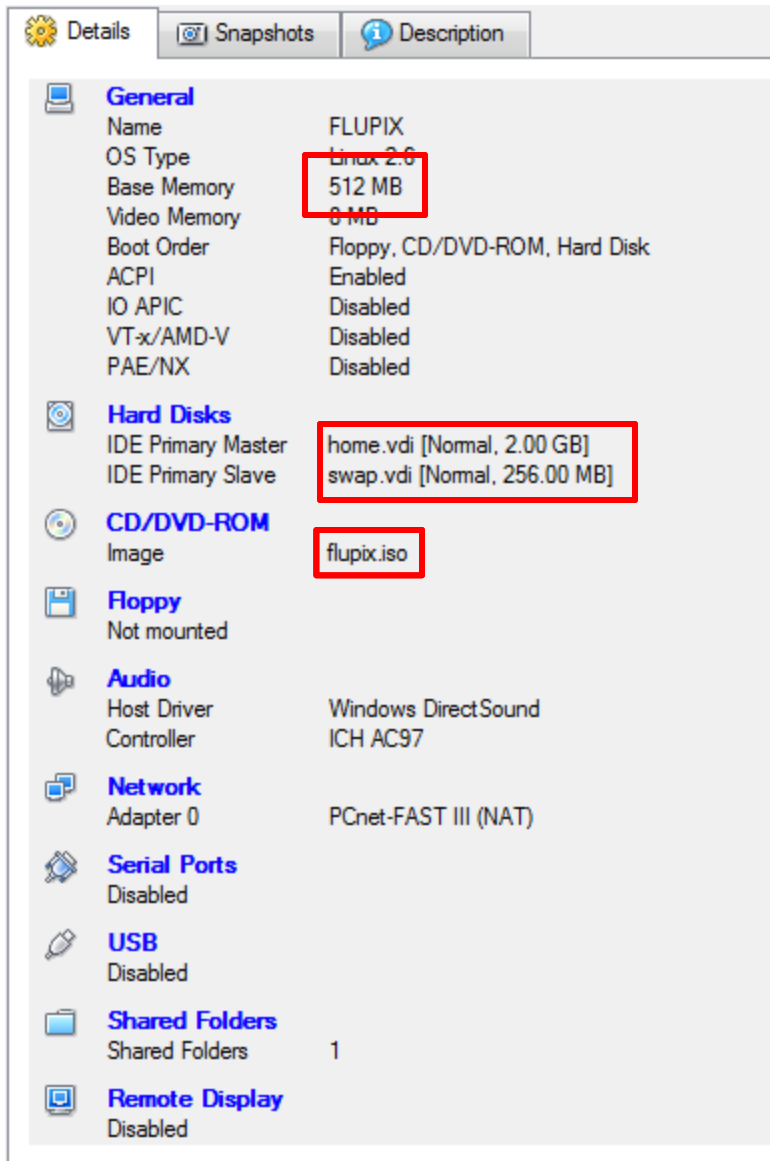
Installation of FLUPIX for VirtualBox

- You will need the following packages
 1. **VirtualBox-X.Y.Z-Win_x86.msi**
The windows setup program of VirtualBox. Install this program in your Windows OS or Mac OS.
 2. **flupix-20XX-YYY.iso**
The FLUPIX bootable CD-ISO image.
 3. **flupix-vdi.zip**
The default configuration of FLUPIX for VirtualBox. Unpack the content of the zip file to copy them to
C:\Documents And Settings\username\.VirtualBox
or
C:\Users\username\.VirtualBox

Setting up

- The flupix-vdi.zip contains a predefined VM named FLUPIX that is attaching two Virtual disks
 - **home.vdi** Dynamic size disk up to 8 Gb, formatted in ext4 used for working space
 - **swap.vdi** Fixed size disk of 256MB used for swapping
- Set the location of the ISO image
 - Select the **CD/DVD Images** tab.
 - Select the flupix iso image (if present) and click on **Release** button, then click on **Remove** button
 - Click on **Add** button and locate the correct ISO image
- Then on the CD/DVD Rom tab
 - Check the **Mount CD/DVD Drive**
 - Check the **ISO Image File**
 - Select the flupix ISO image

FLUPIX Settings



Memory Settings:

- Minimum requirement
RAM: 512 Mb
Swap: 256 Mb
- Recommended:
RAM: 1 Gb
Swap: 512 Mb
- Linux needs at least 256MB to run
- FLUKA needs ~400 Mb with DPMJET ~500 Mb
- Some FLUKA tools need ~500Mb
- FLAIR memory is dynamic

Starting the VM

- Select the FLUPIX VM and click on the Start  button



- It will start the boot in 1s.
[If you want to change the parameters press Tab]
- Booting will take about 20-30s

Working space

- The linux root directory "/" is mounted as readonly from the FLUPIX iso file.
- However you have the possibility of writing and installing extra programs on the "/" **ONLY temporary for the session**
- You can use super-user privileges with the "sudo" command with no password
- The home.vdi if flagged with a label "/home" and will be mounted as /home and it contains the following:
 - **flupix**: your persistent user **flupix** home directory.
 - The swap.vdi contains a memory swap disk of 256MB

Interface

- FLUPIX has precompiled the utilities from VirtualBox that allows
 - **Mouse integration** (only in X11). The mouse of the host is used as such from the guest system. In all other displays the guest is getting the FULL control of the mouse
 - A special driver for **X11 video**, for faster, smoother and hardware accelerated graphics.
 - **Time synchronization** with the host system
 - **Folder sharing** from the host to the guest system.
- **Remember** the “**Right-Ctrl**” key is the default **Host key** of your Virtual Machine. With the use of this key you can redirect all input (keyboard+mouse) from your host to your guest system and many other.

Accessing your host directories

- There are several ways of accessing directories from the host system.
- The easiest way is through the use of a Shared Folder
- Create a Shared folder from your VirtualBox and give a name e.g. *home*
- From FLUPIX call the command
`vboxmount home ~/home`
- The command will create a directory `~/home` and mount the Shared folder *home* to it
- Add the command to your profile script to be executed on every login.
- **You cannot run FLUKA inside a shared folder** since symbolic links (needed by FLUKA) are not supported

Shutting down FLUPIX

- Always try to shutdown correctly the VirtualMachine, from the menu.
Otherwise you can end up with a corrupted persistent image.
- You can even save the machine state, which is equivalent like StandBy. It will create a file equal to the size of the RAM defined 512MB

Input example

- FLUKA is driven by the user almost completely by means of an input file (**.inp**) which contains directives issued in the form of **DATA CARDS**
- The standard release provides a simple case to test the installation: **example.inp** (*Production of particles in p-Be collisions with a 50GeV/c proton beam.*)
- **Different examples** are used along this course, which will be varied in different ways for didactic reasons
- We will start with a minimum input file and after each lecture we will enhance our example with more and more functionality
- It is strongly recommended that for every exercise you create a **subdirectory** *i.e.*, **ex1, ex2, ex3** where all the necessary input and output file will be stored
- For better clarity before starting a new exercise you will get the solution of the previous one, to be picked up at the course website: <http://www.fluka.org/fluka.php?id=course&sub=program&navig=2&which=heidelberg2011>

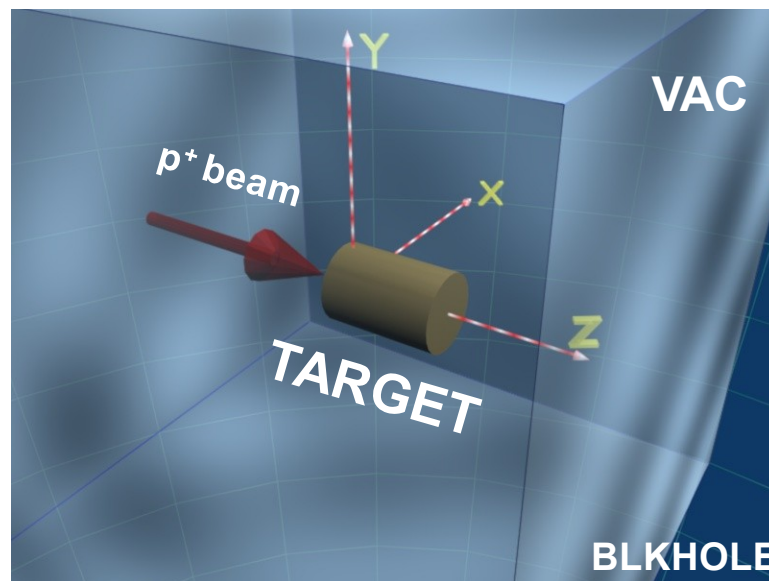
A Simple Example

Geometry

```

TITLE
FLUKA Course Exercise
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...*
DEFAULTS
NEW-DEFA
BEAM          -3.5 -0.082425    -1.7    0.0    0.0    1.0PROTON
BEAMPOS       0.0    0.0    0.1    0.0    0.0    0.0
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...*
GEOBEGIN
0 0          Cylindrical Target
SPH BLK 0.0  0.0  0.0  10000.
* vacuum box
RPP VOI -1000. 1000. -1000. 1000. -1000. 1000.
* Lead target
RCC TARG 0.0 0.0 0.0 0.0 0.0 10. 5.
END
* Regions
* Black Hole
BLKHOLE 5  +BLK -VOI
* Void around
VAC      5  +VOI -TARG
* Target
TARGET  5  +TARG
END
GEOEND
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...*
ASSIGNMA      BLCKHOLE    BLKHOLE
ASSIGNMA      VACUUM      VAC
ASSIGNMA      LEAD        TARGET
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...*
RANDOMIZ       1.0
START         10.0      0.0
STOP

```



Prepare the working space

- We don't want to run inside the \$FLUPRO directories therefore:
- Go to your **home** directory and create a subdirectory named **ex1**:

```
cd  
mkdir ex1  
cd ex1
```

- Get the source example file from the course website (copy all the **ex1*** files to your subdirectory: **ex1**)

<http://www.fluka.org/fluka.php?id=course&sub=program&navig=2&which=heidelberg2011>

```
download ex1*
```

Now let's test the installation

After you have created your standard FLUKA we can run the first example:

No. of previous
cycle (default is 0)

No. of Last cycle
(default is 5)

```
$FLUPRO/flutil/rfluka -e $FLUPRO/flukahp -N0 -M1 ex1
```

Specifies the executable name: if it is **flukahp** in **\$FLUPRO** (default) then it can be omitted

Name of the **input file**. It must be a file named *****.inp** and **.inp** has to be omitted.

What rfluka does:

It creates a temporary subdirectory: `$PWD/fluka_nnnn`
(`$PWD` means the current directory)
where `nnnn` is the system process-id assigned to FLUKA.
There all necessary assignments are defined and output files
are written.

<code>elasct.bin</code>	→ <code>\$FLUPRO/elasct.bin</code>
<code>fluodt.dat</code>	→ <code>\$FLUPRO/fluodt.dat</code>
<code>fort.1</code>	→ <code>../ranex_1001</code>
<code>fort.11</code>	→ <code>ex_1001.out</code>
<code>fort.12</code>	→ <code>libec_thihecufealw_10t.pemf</code>
<code>fort.15</code>	→ <code>ex_1001.err</code>
<code>fort.16</code>	→ “geometry scratch”
<code>fort.2</code>	→ <code>ranex_1002</code>
<code>neuxsc.bin</code>	→ <code>\$FLUPRO/neuxsc-ind_260.bin</code>
<code>nuclear.bin</code>	→ <code>\$FLUPRO/nuclear.bin</code>
<code>sigmapl.bin</code>	→ <code>\$FLUPRO/sigmapl.bin</code>
<code>xnloan.dat</code>	→ <code>\$FLUPRO/xnloan.dat</code>

At the end of the FLUKA run:

If everything is OK the temporary directory disappears
And the relevant results are copied in the start directory:

Removing links by default you have **ex100n.log**, **ex100n.out**,
ex100n.err and **ranex100m** (seed for cycle $m = n+1$)

Removing temporary files

Saving output and random number seed

Saving additional files from scoring requested by the user

Moving fort.33 to /home/username/work/ex1/ex1001_fort.33
Moving fort.47 to /home/username/work/ex1/ex1001_fort.47
Moving fort.48 to /home/username/work/ex1/ex1001_fort.48
Moving fort.49 to /home/username/work/ex1/ex1001_fort.49
Moving fort.50 to /home/username/work/ex1/ex1001_fort.50

End of FLUKA run

Checking FLUKA during the run

Look in the temporary directory:

- Initialization phase ends when the ***.err** file is created.
- Inside ***.err** file and (at the end of ***.out** file) the progress in the number of events is given in the line immediately following the one which starts by **"NEXT SEEDS"**:

```
NEXT SEEDS: C8888D    0    0    0    0    0 0 33B49B1    0    0    0
              1          9          9          0.0000000E+00    1.0000000E+30
0
NEXT SEEDS: C88894    0    0    0    0    0 0 33B49B1    0    0    0
              2          8          8          5.0010681E-03    1.0000000E+30
0
NEXT SEEDS: C8889A    0    0    0    0    0 0 33B49B1    0    0    0
              3          7          7          3.3340454E-03    1.0000000E+30
0
```

.....
EVENTS ALREADY COMPLETED

EVENTS TO BE COMPLETED

AVERAGE CPU TIME CONSUMED PER EVENT

Tips & Tricks

How to make a “clean” stop of FLUKA run

- Here “clean” means closing all files, writing scoring output and removing the temporary directory and files.
- In the temporary run directory:

touch fluka.stop	To stop the present cycle
or touch <u>r</u> fluka.stop	To stop all remaining cycles

- The clean stop will occur at the next CPU-time check, *i.e.*, at the same time when printing the random number calls : see **START** card instructions (5th parameter) for the frequency of these checks!!
- If the check is never performed it means that the program has entered an infinite loop (probably a fault in user code)