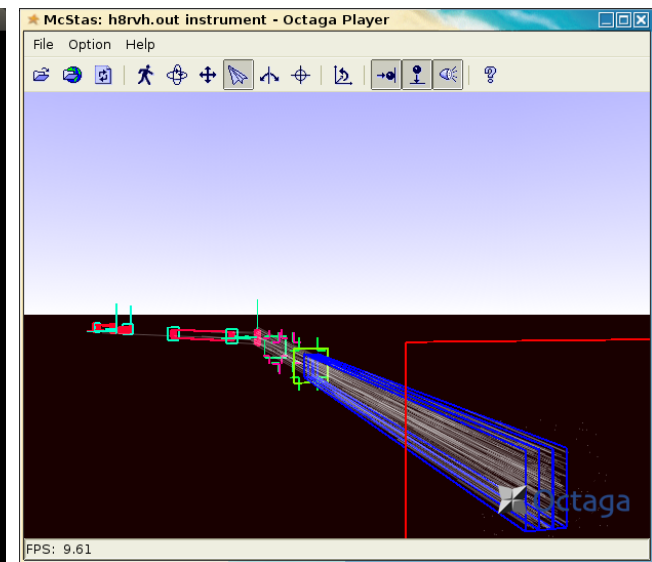
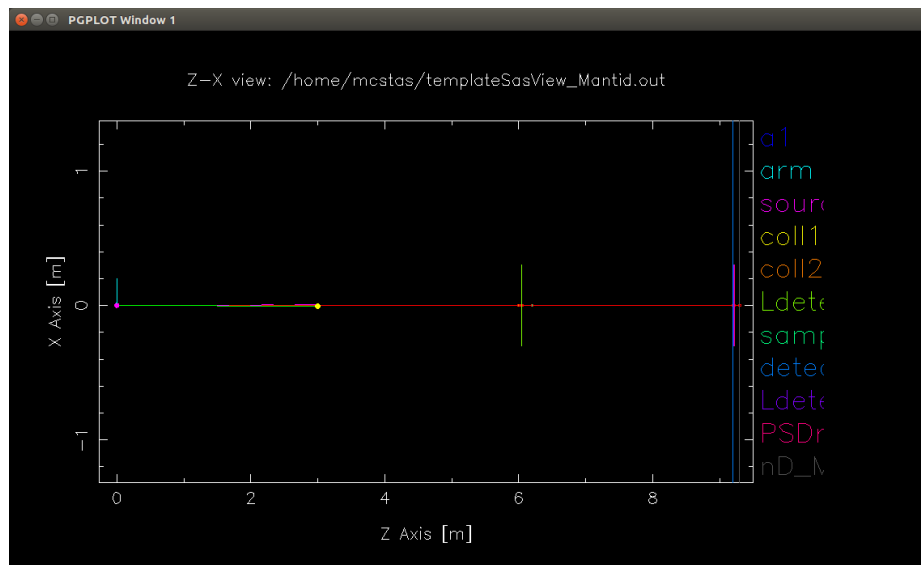


The McStas Raytracing Toolkit for Simulating Neutron Scattering Instruments and Experiments



Monte Carlo Simulation of Triple Axis Spectrometers



- Developed at/supported by DTU Physics, ILL, PSI, Uni CPH, ESS DMSC.
- Freely available software. Licensed under GNU GPL v2.
- Flexible, general simulation utility for neutron scattering experiments.
- Allows virtual instruments to be built on a component by component basis.
- Used for instrumentation, planning, construction, virtual experiments, data analysis and teaching.
- Experimental data can be output to Nexus format (normally HDF5)



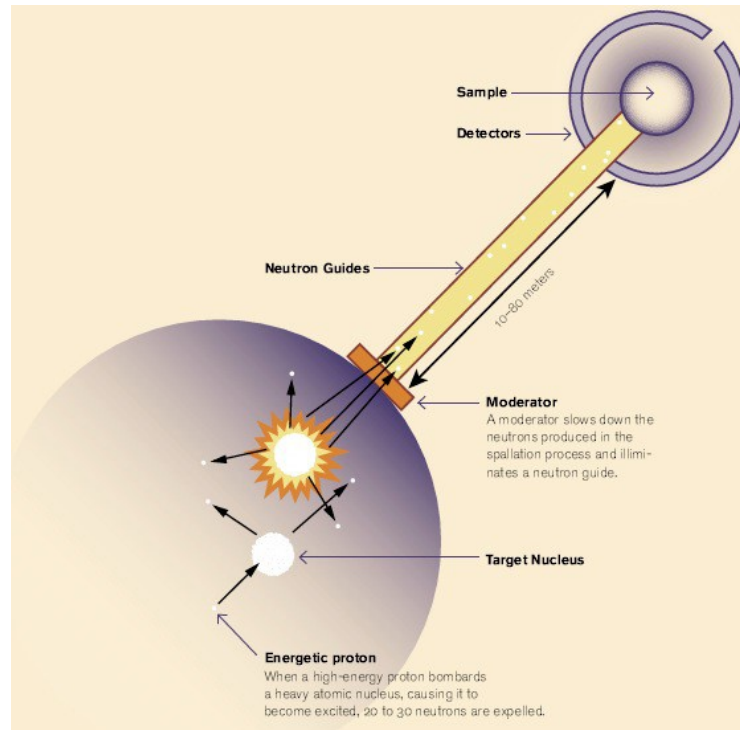
System Requirements and Supported Platforms

- A C Compiler (gcc/icc recommended)
- Perl and Perl-Tk
- Matlab OR (pgplot+pgperl+PDL) OR Gnuplot
- Linux, Windows and MacOS. (Instructions on website)



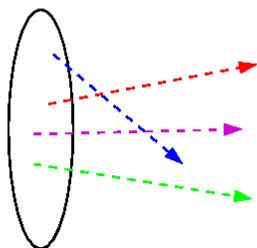
Supported Components

- Moderators
- Neutron Guides
- Collimators & Slits
- Choppers
- Samples
- Detectors (Monitors)
- And More



How it works (Simple)

1. Particles emitted with random starting conditions via MC



2. Particles are "ray-traced" through space



3. Will eventually meet other objects e.g. a studied experimental sample and get scattered via MC again



4. At various points in the instrument the particle states are measured in so-called monitors or detectors



Brief Note on McStas Design/Architecture

- McStas is based on a meta-language specifically designed for neutron scattering instruments.
- McStas compiles this metalanguage into ISO-C programs.
- There are three levels of source code:
 1. Instrument File (*.instr) - input
 2. Component Files (*.comp) - input
 3. ANSI c code (*.c) - output



Science & Technology Facilities Council

ISIS

Defining an Instrument

```
DEFINE INSTRUMENT My_Instrument(DIST=10)

/* Here comes the TRACE section, where the actual      */
/* instrument is defined as a sequence of components    */
TRACE

/* The Arm() class component defines reference points and orientations */
/* in 3D space.                                          */
COMPONENT Origin = Arm(
)
AT (0, 0, 0) ABSOLUTE

COMPONENT Source = Source_simple(
radius = 0.1, yheight = 0.1, xwidth = 0.1, dist = 10, E0 = 5,
dE = 1)
AT (0, 0, 0) RELATIVE Origin

COMPONENT Emon = E_monitor(
filename = "Emon.dat", xmin = -0.1, xmax = 0.1, ymin = -0.1,
ymax = 0.1, Emin = 0, Emax = 10)
AT (0, 0, DIST) RELATIVE Origin

COMPONENT PSD = PSD_monitor(
nx = 128, ny = 128, filename = "PSD.dat", xmin = -0.1,
xmax = 0.1, ymin = -0.1, ymax = 0.1)
AT (0, 0, 1e-10) RELATIVE Emon

/* The END token marks the instrument definition end*/
END
```

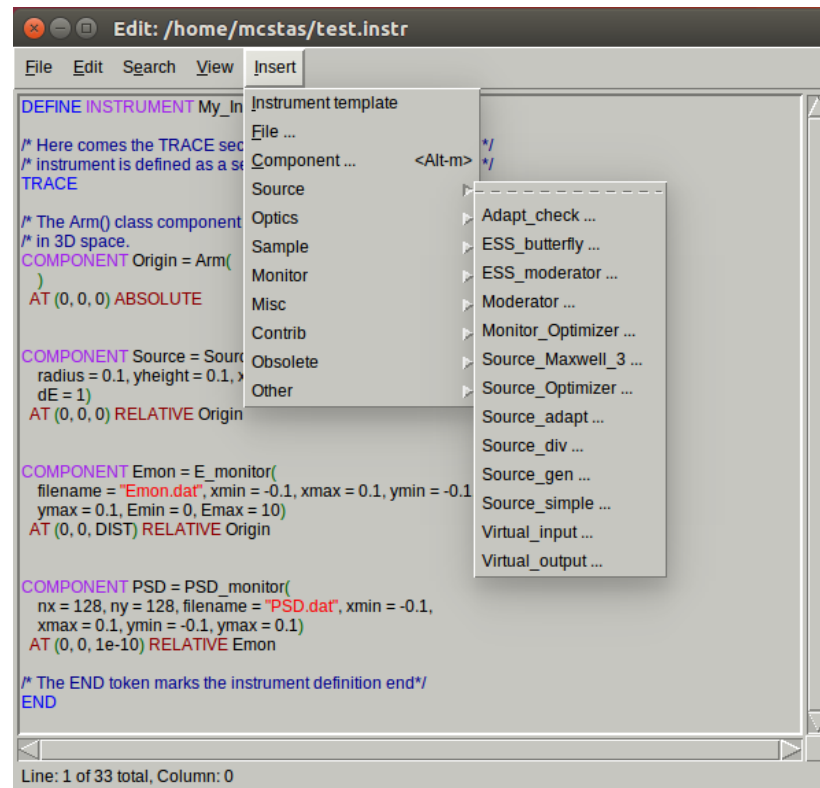
- Component Manual - <http://www.mcstas.org/documentation/manual/mcstas-2.3-components.pdf>
- Component Descriptions (and source) - <http://www.mcstas.org/download/components/>



McGUI (Creating/Editing Instruments)



McGUI (Selecting Components)



McGUI (Tracing by Hand)

Run simulation /home/mcstas/test.instr

Instrument source: /home/mcstas/test.instr [HTML docs](#)

Instrument parameters (D=floating point, I=integer, S=string):

DIST (D):

Output to (dir): ☐ overwrite [Browse...](#)

Neutron count: ☐ gravity (BEWARE) Random seed:

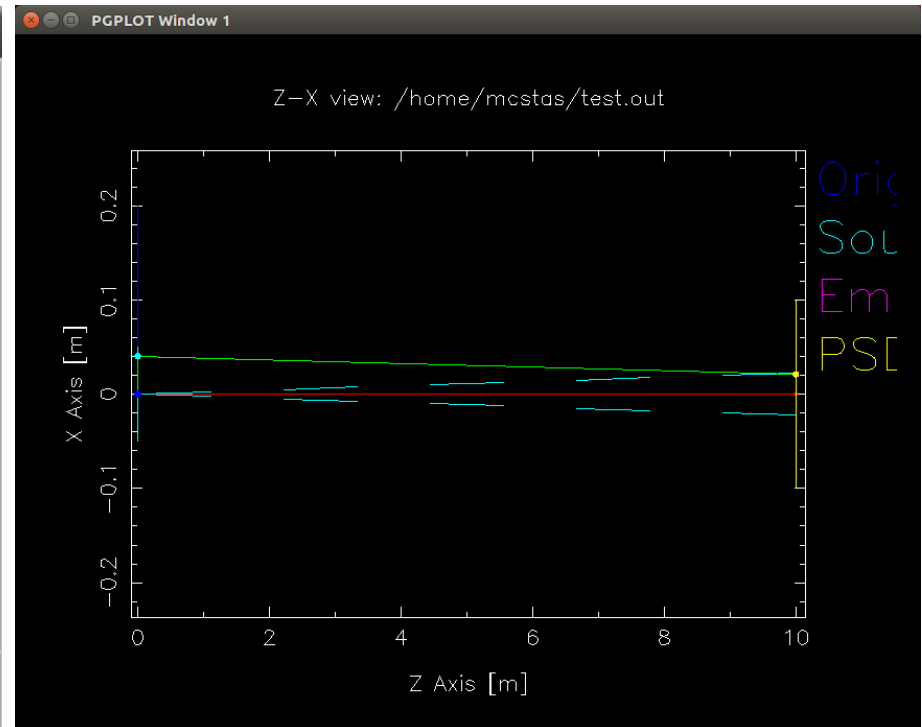
[Trace \(3D view\)](#) # steps: ☐ Plot results with: [PGPLOT](#)

Clustering: [None \(single CPU\)](#) Number of nodes:

Inspect component:

First component:

Last component:



McGUI (Running the simulation)

Run simulation /home/mcstas/test.instr

Instrument source: /home/mcstas/test.instr [HTML docs](#)

Instrument parameters (D=floating point, I=integer, S=string):
DIST (D): 10

Output to (dir): ☐ overwrite [Browse...](#)

Neutron count: 1000000 ☐ gravity (BEWARE) Random seed:

[Simulate](#) # steps 0 ☐ Plot results with: [PGPLOT](#)

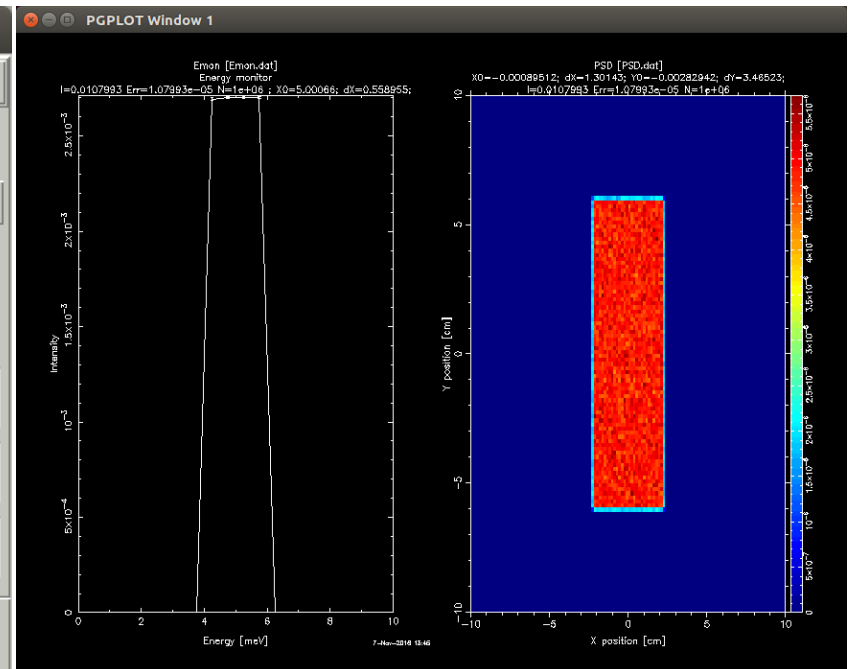
Clustering: None (single CPU) Number of nodes: 2

Inspect component: Origin Source

First component: Origin Source

Last component: Origin Source

[Start](#) [Cancel](#)



Extras

- Interoperability with Mantid. Components are written as Mantid components into an IDF.
- IDF embedded in Nexus output and written as a xml file using mcdisplay.
- Requires special naming conventions.
 - sourceMantid
 - sampleMantid
 - nd_Monitor_n
- Uses OFF file types for more complex geometries.



Questions?



Science & Technology Facilities Council

ISIS