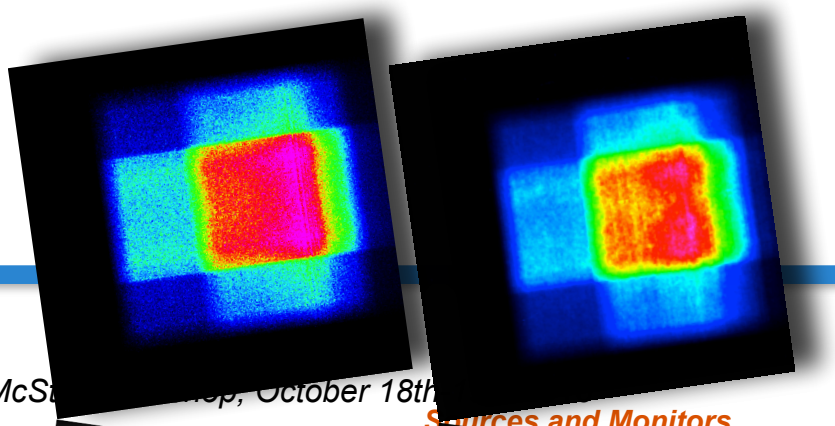
# McStas monitors & sources

Presenter: Ulrich Wildgruber



# Access the docs

October 2018

*In this session:*

- *Overview of existing Source and Monitor components*
- *Detailed description of the most commonly used ones*
- *How to 'call' them into a *.instr file*
- *Practical Exercise using sources and monitors*

***IMPORTANT:***

*All (and more) of this information can be found in the online pdf component documentation, e.g.*
*http://www.mcstas.org/documentation/manual/mcstas-2.4.1-components.pdf*

*- is also distributed with your McStas installation - mcdoc -c*

*The component documentation along with the "mcdoc component_you_are_searching_for" command, are your best friends when using McStas*

# Sources



- *Mathematical:*
  - *Source_simple.comp*
  - *Source_div.comp*
- *Pulsed sources:*
  - *ESS_moderator.comp*
  - *Moderator.comp*
  - *SNS_source.comp (*)*
  - *ISIS_moderator.comp (*)*

*Reactors :*

*Source_Maxwell_3.comp*

*Source_gen.comp*

*Source_gen4.comp*

*Source_multi_surfaces.comp (*)*

*(*) contributed (can be found in /mcstas/installation/folder/**contrib** )*

# Sources

October 2018

*Typing "mcdoc", mcdoc source" or "mcdoc moderator" in your command shell, will reveal a list of available sources and moderators. Or you can search in the directories 'sources', 'contrib', and 'obsolete'.*

## Components and Instruments from the Library for *McStas*

Names in **Boldface** denote components that are properly documented with comments in the source code.

**Sources**

| *Name* | *Origin* | *Author(s)* | *Source code* | *Description* |
|---|---|---|---|---|
| **Adapt_check** | Risoe | Kristian Nielsen | comp | Optimization specifier for the Source_adapt component. |
| **ESS_butterfly** | DTU | Peter Willendrup and Esben Klinkby | comp | ESS butterfly moderator |
| **ESS_moderator** | DTU | P Willendrup and E Klinkby, February 2014, derived from K Lefmann ESS_moderator_long | comp | A parametrised pulsed source for modelling ESS long pulses. |
| **Moderator** | Risoe | KN, M.Hagen | comp | A simple pulsed source for time-of-flight. |
| **Monitor_Optimizer** | ILL (France) | Emmanuel Farhi | comp | To be used after the **Source_Optimizer** component |
| **Source_Maxwell_3** | Risoe | Kim Lefmann | comp | Source with up to three Maxwellian distributions |
| **Source_Optimizer** | ILL (France) | Emmanuel Farhi | comp | A component that optimizes the neutron flux passing through the Source_Optimizer in order to have the maximum flux at the **Monitor_Optimizer** position. |
| **Source_adapt** | Risoe | Kristian Nielsen | comp | Neutron source with adaptive importance sampling |
| **Source_div** | Risoe | KL | comp | Neutron source with Gaussian or uniform divergence |
| **Source_gen** | ILL/Risoe | Emmanuel Farhi, Kim Lefmann | comp | Circular/squared neutron source with flat or Maxwellian energy/wavelength spectrum |
| **Source_simple** | Risoe | Kim Lefmann | comp | A circular neutron source with flat energy spectrum and arbitrary flux |
| **Virtual_input** | ILL | E. Farhi | comp | Source-like component that generates neutron events from an ascii 'virtual source' filename. |
| **Virtual_output** | ILL | E. Farhi | comp | Detector-like component that writes neutron state parameters into an ascci-format 'virtual source' neutron file. |

*(*) contributed (can be found in /mcstas/installation/folder/**contrib** )*

# Sources



*Typing "mcdoc", mcdoc source" or "mcdoc moderator" in your command shell, will reveal a list of available sources and moderators. Or you can search in the directories 'sources', 'contrib', and 'obsolete'.*

## Components and Instruments from the Library for *McStas*

Names in **Boldface** denote components that are properly documented with comments in the source code.

**Sources**

| *Name* | *Origin* | *Author(s)* | *Source code* | *Description* |
|---|---|---|---|---|
| Adapt_check | Risoe | Kristian Nielsen | comp | Optimization specifier for the Source_adapt component. |
| ESS_butterfly | DTU | Peter Willendrup and Esben Klinkby | comp | ESS butterfly moderator |
| ESS_moderator | DTU | P Willendrup and E Klinkby, February 2014, derived from K Lefmann ESS_moderator_long | comp | A parametrised pulsed source for modelling ESS long pulses. |
| Moderator | Risoe | KN, M.Hagen | comp | A simple pulsed source for time-of-flight. |
| Monitor_Optimizer | ILL (France) | Emmanuel Farhi | comp | To be used after the **Source_Optimizer** component |
| Source_Maxwell_3 | | | | |
| Source_Optimizer | ILL (France) | Emmanuel Farhi | comp | A co in or |
| Source_adapt | Risoe | Kristian Nielsen | comp | Neut |
| Source_div | | | | |
| Source_gen | | Kim Lefmann | | spect |
| Source_simple | | | | |
| Virtual_input | ILL | E. Farhi | comp | Sour filen |
| Virtual_output | ILL | E. Farhi | comp | Detector-like component that writes neutron state parameters into an ascci-format 'virtual source' neutron file. |

***Steady-state sources (Reactors & PSI… )***

*folder/**contrib** )*

# Sources

October 2018

*Typing "mcdoc", mcdoc source" or "mcdoc moderator" in your command shell, will reveal a list of available sources and moderators. Or you can search in the directories 'sources', 'contrib', and 'obsolete'.*

## Components and Instruments from the Library for *McStas*

Names in **Boldface** denote components that are properly documented with comments in the source code.

**Sources**

| *Name* | *Origin* | *Author(s)* | *Source code* | *Description* |
|---|---|---|---|---|
| Adapt_check | Risoe | Kristian Nielsen | comp | Optimization specifier for the Source_adapt component. |
| ESS_butterfly | DTU | Peter Willendrup and Esben Klinkby | comp | ESS butterfly moderator |
| ESS_moderator | | P Willendrup and E Klinkby, February 2014, derived from K Lefmann ESS_moderator_long | comp | A pa |
| Moderator | | | | |
| Monitor_Optimizer | ILL (France) | Emmanuel Farhi | comp | To b |
| Source_Maxwell_3 | Risoe | Kim Lefmann | comp | Sour |
| Source_Optimizer | ILL (France) | Emmanuel Farhi | comp | A co in or |
| Source_adapt | Risoe | Kristian Nielsen | comp | Neut |
| Source_div | Risoe | KL | comp | Neutron source with Gaussian or uniform divergence |
| Source_gen | ILL/Risoe | Emmanuel Farhi, Kim Lefmann | comp | Circular/squared neutron source with flat or Maxwellian energy/wavelength spectrum |
| Source_simple | Risoe | Kim Lefmann | comp | A circular neutron source with flat energy spectrum and arbitrary flux |
| Virtual_input | ILL | E. Farhi | comp | Source-like component that generates neutron events from an ascii 'virtual source' filename. |
| Virtual_output | ILL | E. Farhi | comp | Detector-like component that writes neutron state parameters into an ascci-format 'virtual source' neutron file. |

***Pulsed sources***

***(also SNS_source and ISIS_moderator in the contrib category)***

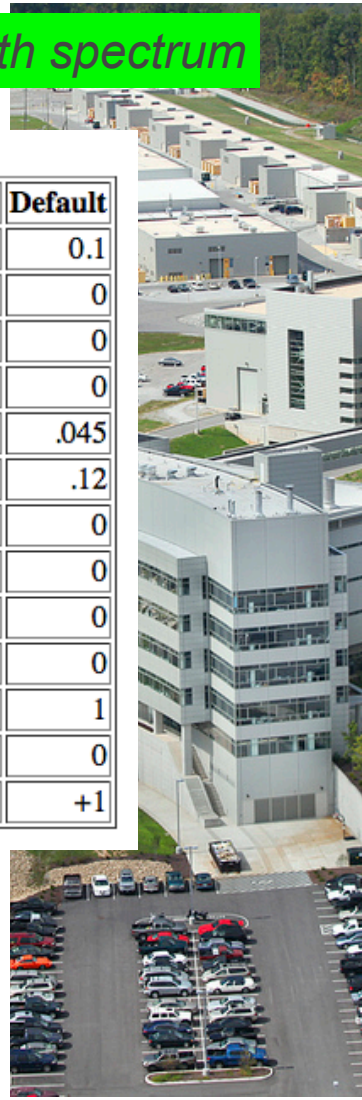
*folder/**contrib** )*

# Sources

October 2018

## Source_simple.comp

*A Simple continuous source with flat energy/wavelength spectrum*



Parameters in **boldface** are required; the others are optional.

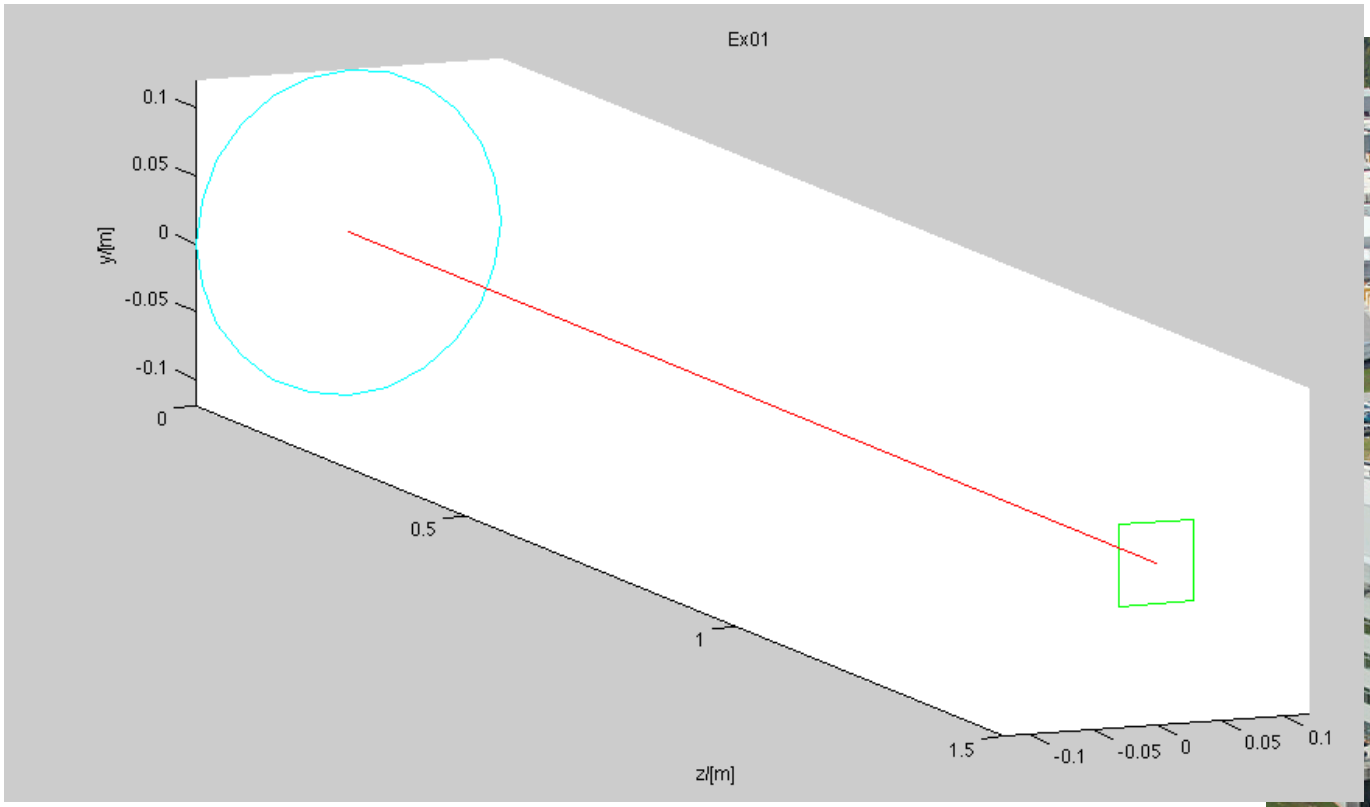
| Name | Unit | Description | Default |
|---|---|---|---|
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. | 0.1 |
| yheight | m | Height of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| xwidth | m | Width of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target | .045 |
| focus_yh | m | Height of target | .12 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy half spread of neutrons (flat or gaussian sigma). | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength half spread of neutrons. | 0 |
| flux | 1/(s*cm**2*st*energy unit) | flux per energy unit, Angs or meV if flux=0, the source emits 1 in 4*PI whole space. | 1 |
| gauss | 1 | Gaussian (1) or Flat (0) energy/wavelength distribution | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

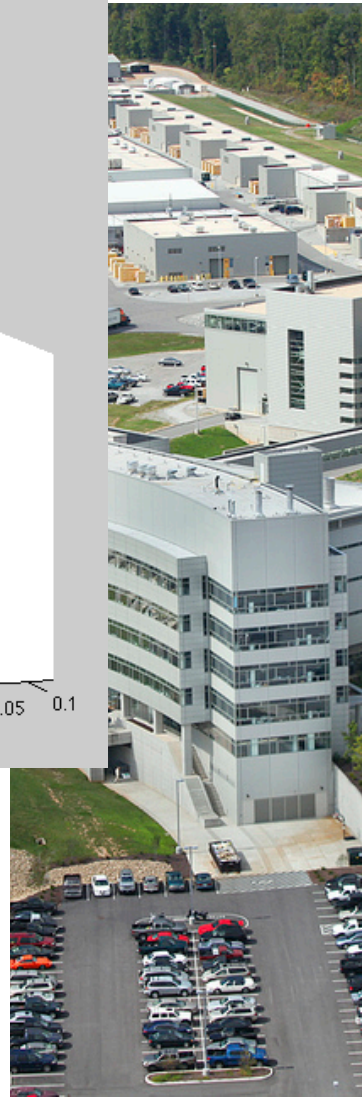
*What do we need the distance and target size for?*

*(let us take a small break and investigate this)*

# Sources

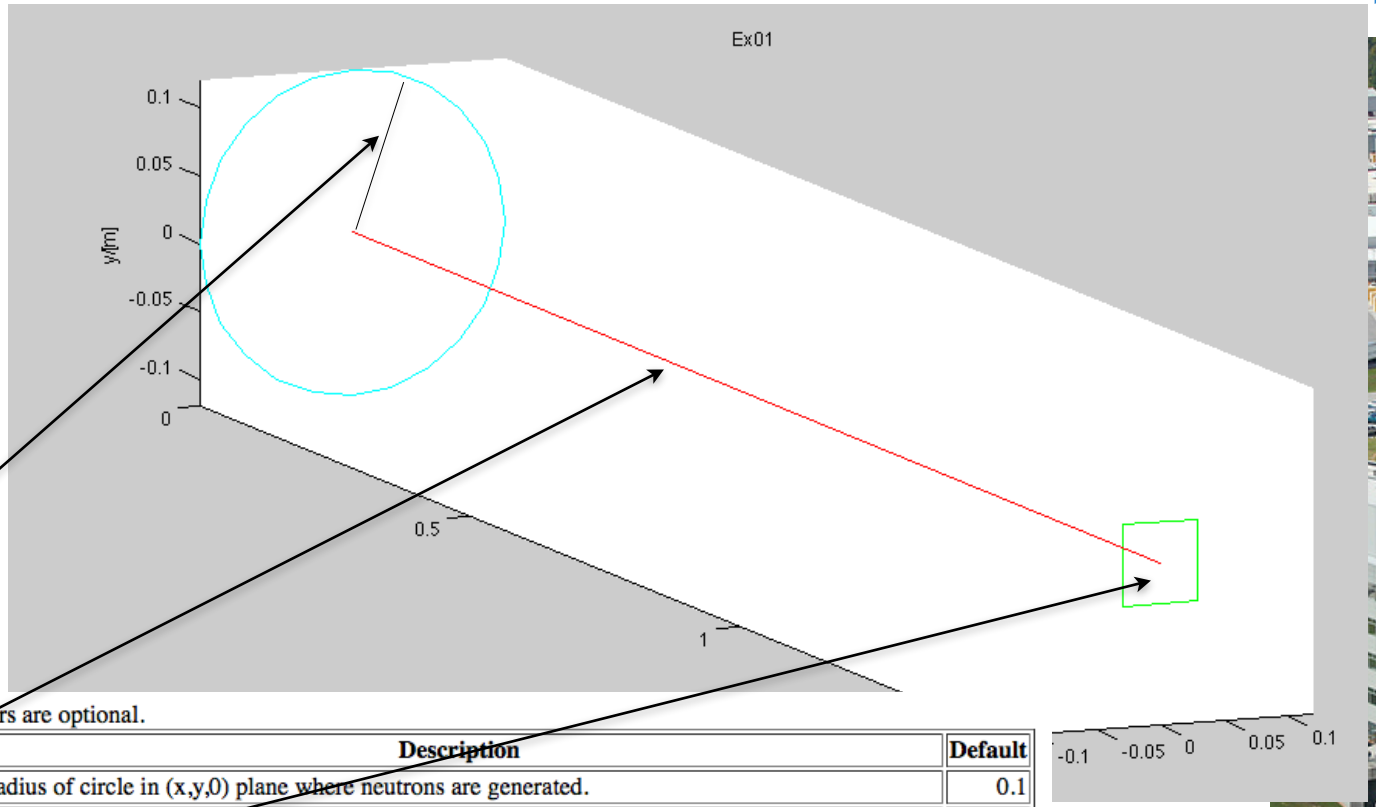


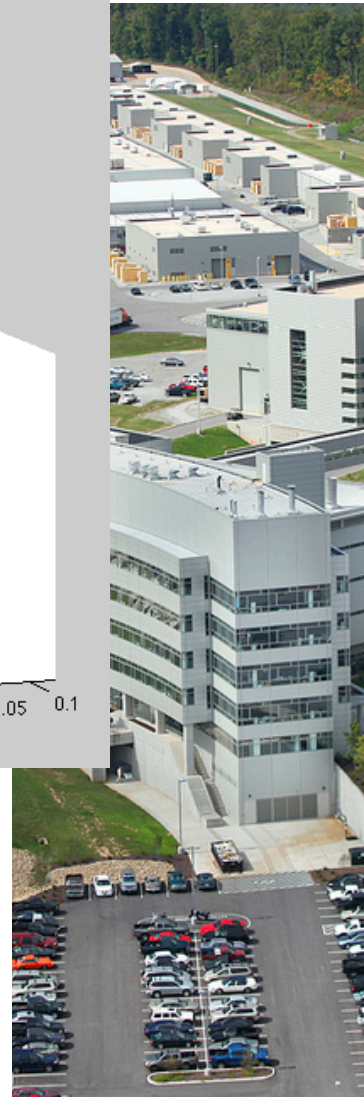


# Sources





Parameters in **boldface** are required; the others are optional.

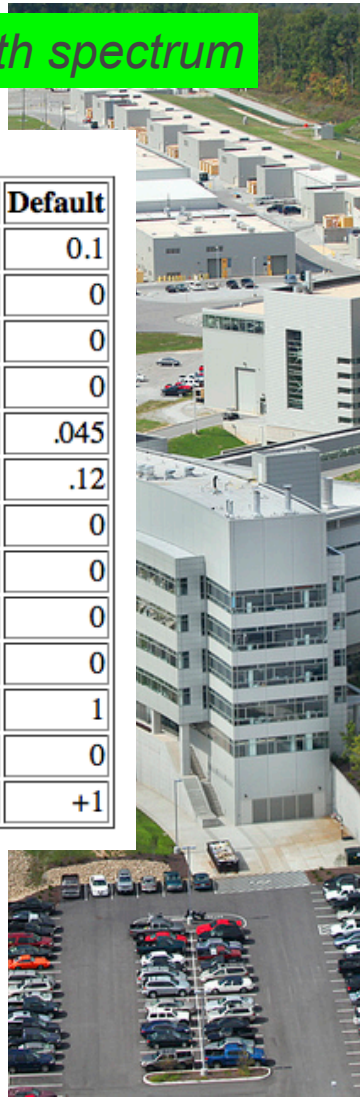
| Name | Unit | Description | Default |
|---|---|---|---|
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. | 0.1 |
| yheight | m | Height of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| xwidth | m | Width of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target | .045 |
| focus_yh | m | Height of target | .12 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy half spread of neutrons (flat or gaussian sigma). | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength half spread of neutrons. | 0 |
| flux | 1/(s*cm**2*st*energy unit) | flux per energy unit, Angs or meV if flux=0, the source emits 1 in 4*PI whole space. | 1 |
| gauss | 1 | Gaussian (1) or Flat (0) energy/wavelength distribution | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

# Source_simple

October 2018

Source_simple.comp

*A Simple continuous source with flat energy/wavelength spectrum*



Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. | 0.1 |
| yheight | m | Height of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| xwidth | m | Width of rectangle in (x,y,0) plane where neutrons are generated. | 0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target | .045 |
| focus_yh | m | Height of target | .12 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy half spread of neutrons (flat or gaussian sigma). | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength half spread of neutrons. | 0 |
| flux | 1/(s*cm**2*st*energy unit) | flux per energy unit, Angs or meV if flux=0, the source emits 1 in 4*PI whole space. | 1 |
| gauss | 1 | Gaussian (1) or Flat (0) energy/wavelength distribution | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

Example:

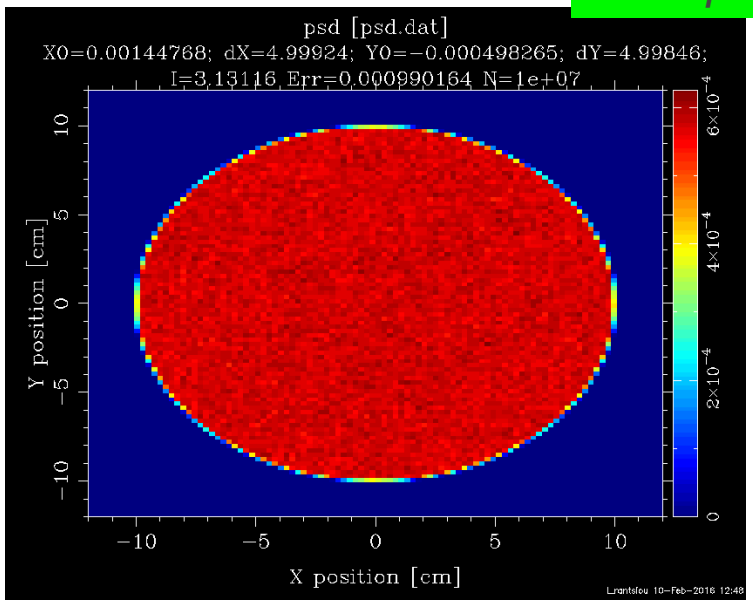
```
COMPONENT my_simple_source = Source_simple(radius=0.1, dist=2.0,
                             focus_xw=0.1, focus_yh=0.1, E0=14.0,
                             dE=2.0)
```

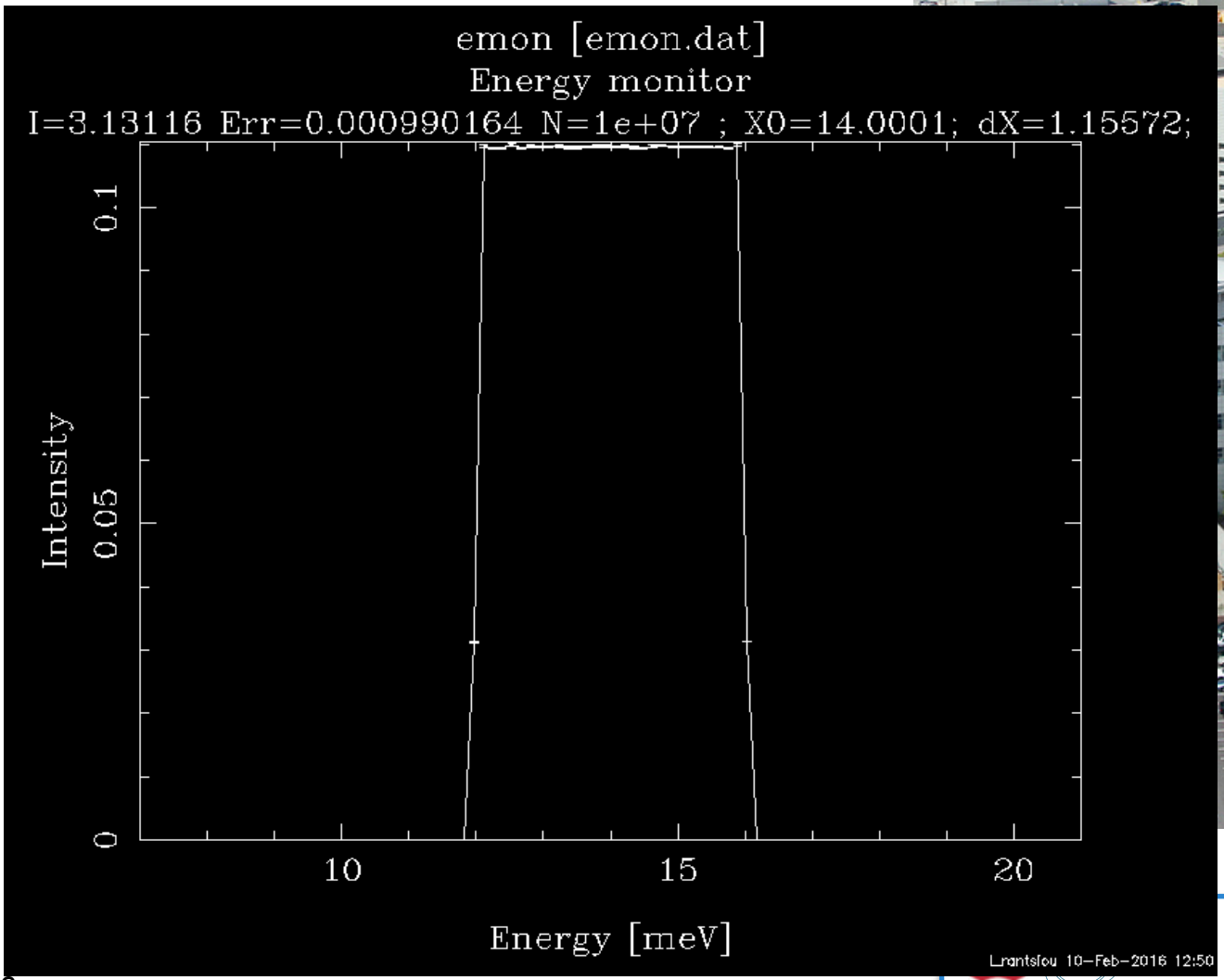
# Monitor output PSD / Emon

October 2018

[Source_simple.comp](#)

*A Simple continuous source with flat energy/wavelength spectrum*



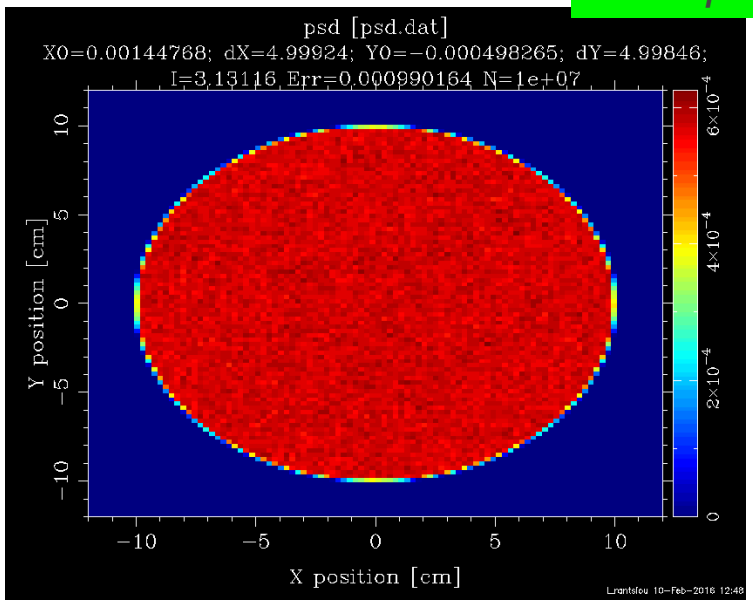



# Monitor output PSD / Emon

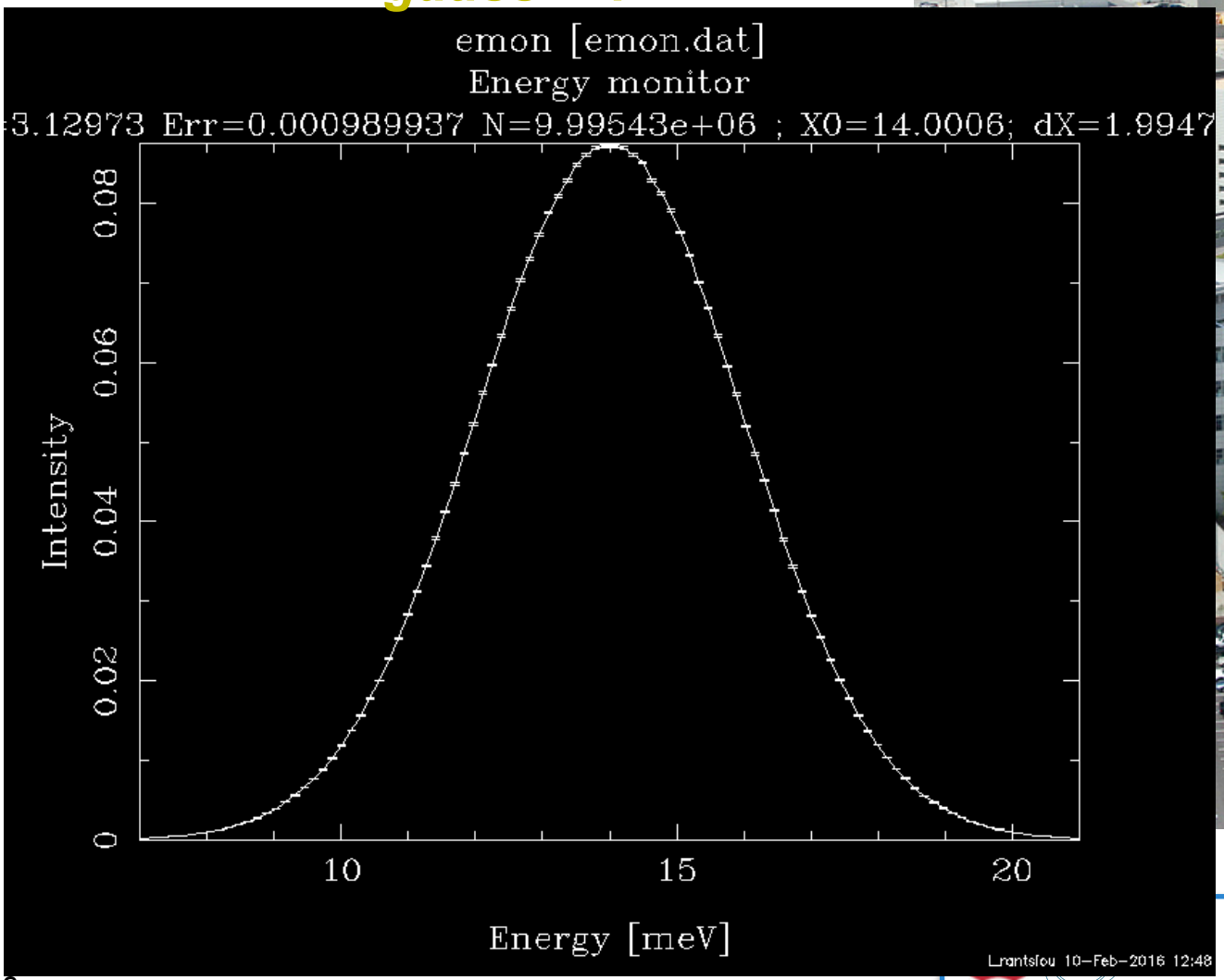
October 2018

[Source_simple.comp](#)

*A Simple continuous source with flat energy/wavelength spectrum*




*gauss = 1*



# Source_div

October 2018

Source_div.comp

## Input parameters

*A Simple continuous source with flat energy/wavelength spectrum*

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| **xwidth** | m | Width of source | |
| **yheight** | m | Height of source | |
| **focus_aw** | deg | FWHM (Gaussian) or maximal (uniform) horz. width divergence | |
| **focus_ah** | deg | FWHM (Gaussian) or maximal (uniform) vert. height divergence | |
| E0 | meV | Mean energy of neutrons. | 0.0 |
| dE | meV | Energy half spread of neutrons. | 0.0 |
| lambda0 | Ang | Mean wavelength of neutrons (only relevant for E0=0) | 0.0 |
| dlambda | Ang | Wavelength half spread of neutrons. | 0.0 |
| gauss | 0\|1 | Criterion: 0: uniform, 1: Gaussian distributions | 0 |
| flux | 1/(s*cm**2*st*energy unit) | flux per energy unit, Angs or meV | 1 |

Example:

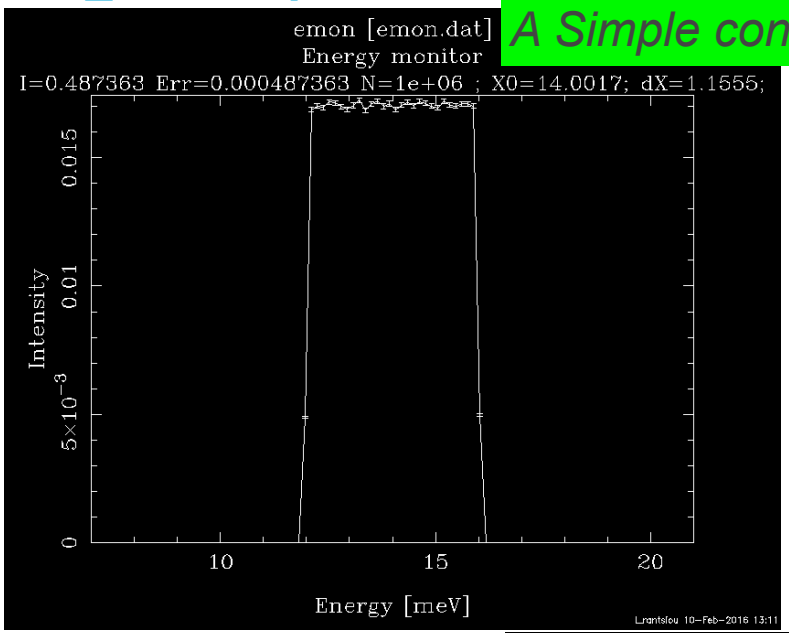
```
COMPONENT my_source_div = Source_div(xwidth=0.1, yheight=0.1,
                         focus_aw=2,focus_ah=2, E0=14, dE=2, gauss=0)
```

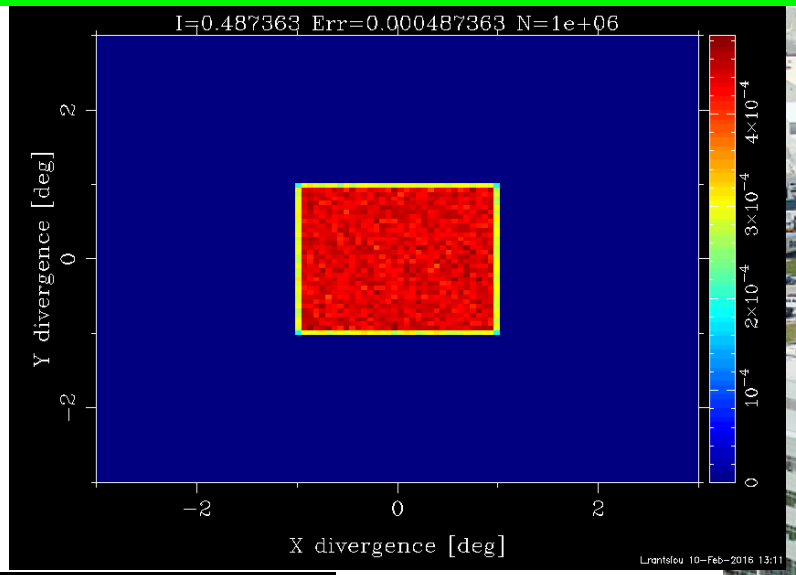
# Source_div

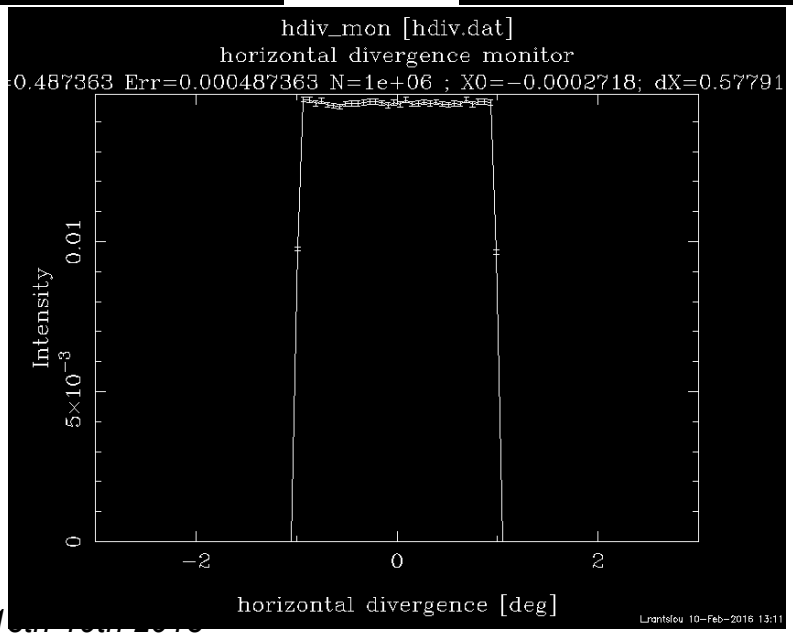
October 2018

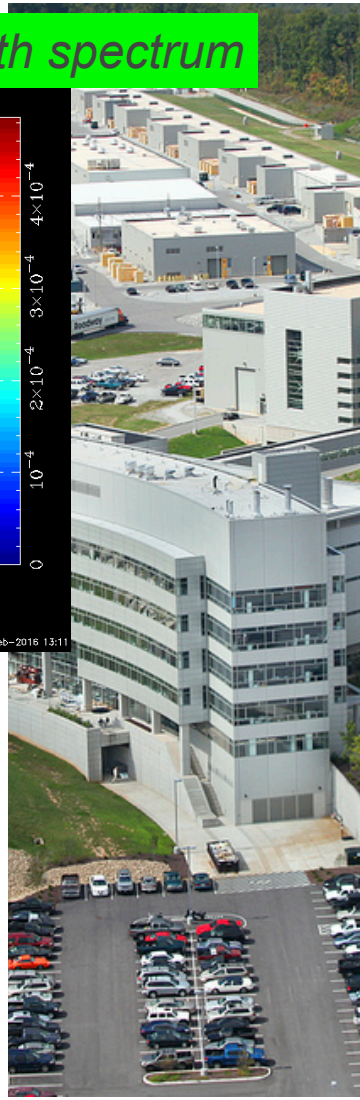
[Source_div.comp](#)

A Simple continuous source with flat energy/wavelength spectrum



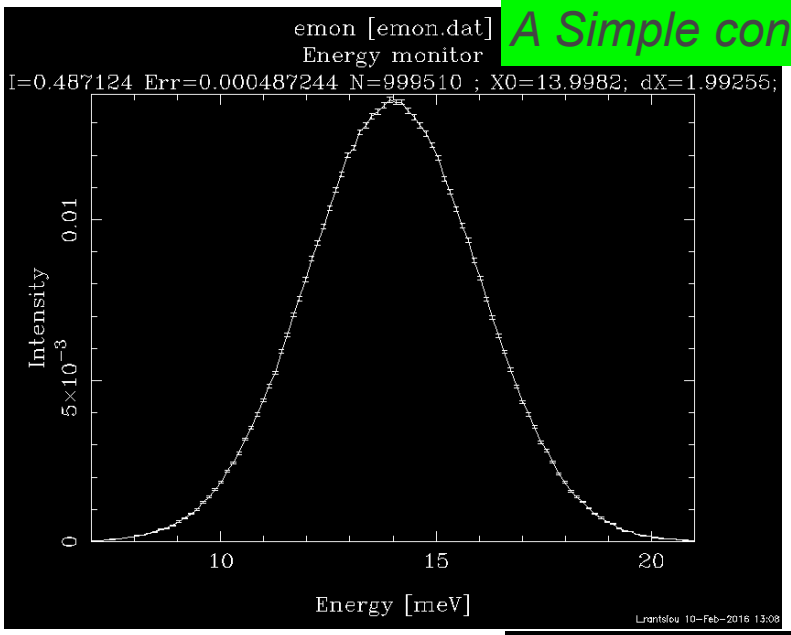



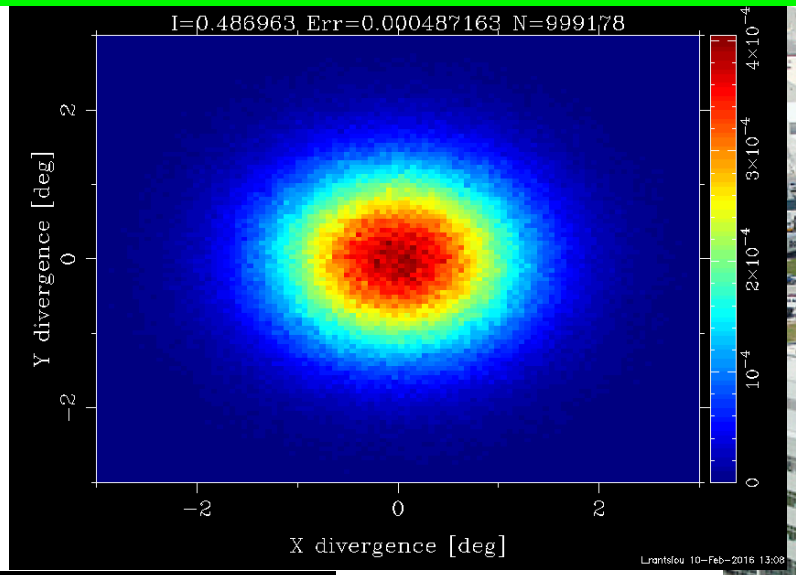
# Source_div

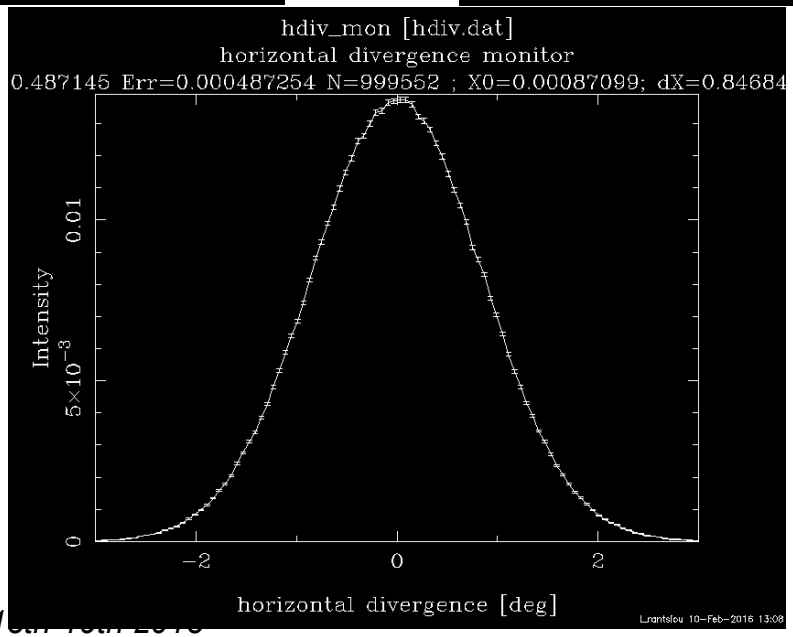
October 2018

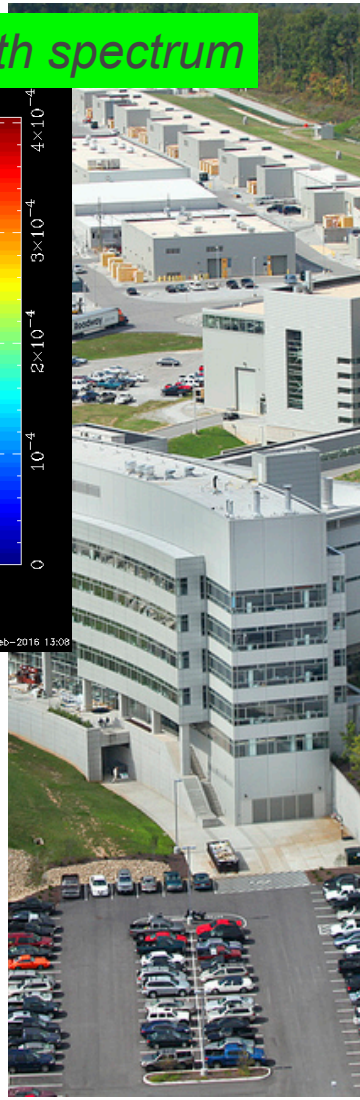
[Source_div.comp](#)

A Simple continuous source with flat energy/wavelength spectrum



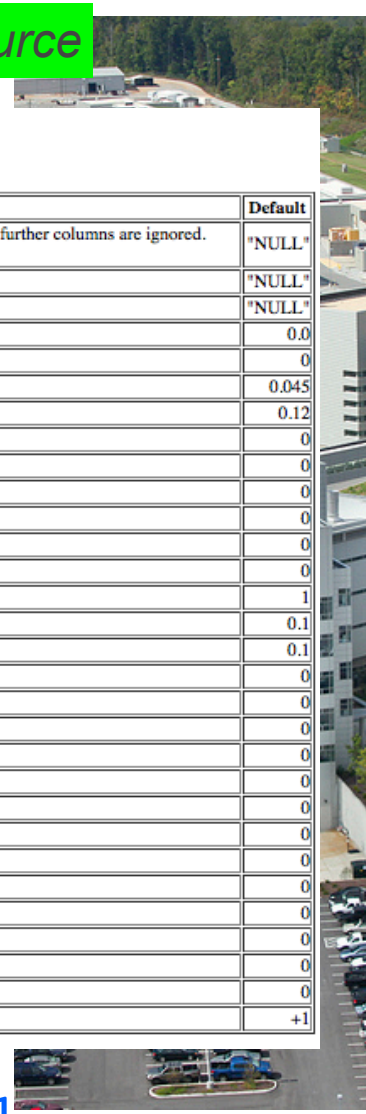
gauss = 1

# Source_gen

October 2018

Source_gen.comp

A general continuous source



**Input parameters**

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| flux_file | str | Name of a two columns [lambda flux] text file that contains the wavelength distribution of the flux in **either** [1/(s*cm**2*st)] **or** [1/(s*cm**2*st*AA)] (see flux_file_perAA flag) Comments (#) and further columns are ignored. Format is compatible with McStas/PGPLOT wavelength monitor files. When specified, temperature and intensity values are ignored. | "NULL" |
| xdiv_file | str | Name of the x-horiz. divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: xmin xmax xdiv_min xdiv_max' | "NULL" |
| ydiv_file | str | Name of the y-vert. divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: ymin ymax ydiv_min ydiv_max' | "NULL" |
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. You may also use 'yheight' and 'xwidth' for a square source | 0.0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target. | 0.045 |
| focus_yh | m | Height of target. | 0.12 |
| focus_aw | deg | maximal (uniform) horz. width divergence | 0 |
| focus_ah | deg | maximal (uniform) vert. height divergence | 0 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy spread of neutrons, half width. | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength spread of neutrons,half width | 0 |
| I1 | 1/(cm**2*sr*AA) | Source flux per solid angle, area and Angstrom if I1=0, the source emits 1 in 4*PI whole space. | 1 |
| yheight | m | Source y-height, then does not use radius parameter | 0.1 |
| xwidth | m | Source x-width, then does not use radius parameter | 0.1 |
| verbose | 0/1 | display info about the source. -1 unactivate source. | 0 |
| T1 | K | Temperature of the Maxwellian source, 0=none | 0 |
| flux_file_perAA | 1 | When true (1), indicates that flux file data is already per Angstroem. If false, file data is per wavelength bin. | 0 |
| flux_file_log | 1 | When true, will transform the flux table in log scale to improve the sampling. | 0 |
| Lmin | AA | Minimum wavelength of neutrons | 0 |
| Lmax | AA | Maximum wavelength of neutrons | 0 |
| Emin | meV | Minimum energy of neutrons | 0 |
| Emax | meV | Maximum energy of neutrons | 0 |
| T2 | K | Second Maxwellian source Temperature, 0=none | 0 |
| I2 | 1/(cm**2*sr*AA) | Second Maxwellian Source flux | 0 |
| T3 | K | Third Maxwellian source Temperature, 0=none | 0 |
| I3 | 1/(cm**2*sr*AA) | Third Maxwellian Source flux | 0 |
| zdepth | m | Source z-zdepth, not anymore flat | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

Example:

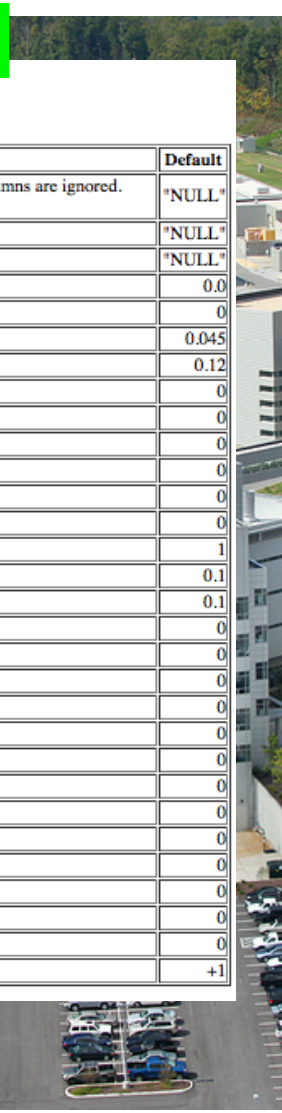
```
COMPONENT my_source_gen = Source_gen(yheight=0.1, xwidth=0.1, Emin=1,
                    Emax=3, I1=1e13, verbose=1, focus_xw=0.01,
                    focus_yh=0.01)
```

# Source_gen

October 2018

## Source_gen.comp

A general continuous source



### Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| flux_file | str | Name of a two columns [lambda flux] text file that contains the wavelength distribution of the flux in **either** [1/(s*cm**2*st)] **or** [1/(s*cm**2*st*AA)] (see flux_file_perAA flag) Comments (#) and further columns are ignored. Format is compatible with McStas/PGPLOT wavelength monitor files. When specified, temperature and intensity values are ignored. | "NULL" |
| xdiv_file | str | Name of the x-horiz divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: xmin xmax xdiv_min xdiv_max' | "NULL" |
| ydiv_file | str | Name of the y-ver divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: ymin ymax ydiv_min ydiv_max' | "NULL" |
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. You may also use 'yheight' and 'xwidth' for a square source | 0.0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target. | 0.045 |
| focus_yh | m | Height of target. | 0.12 |
| focus_aw | deg | maximal (uniform) horz. width divergence | 0 |
| focus_ah | deg | maximal (uniform) vert. height divergence | 0 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy spread of neutrons, half width. | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength spread of neutrons,half width | 0 |
| I1 | 1/(cm**2*sr*AA) | Source flux per solid angle, area and Angstrom if I1=0, the source emits 1 in 4*PI whole space. | 1 |
| yheight | m | Source y-height, then does not use radius parameter | 0.1 |
| xwidth | m | Source x-width, then does not use radius parameter | 0.1 |
| verbose | 0/1 | display info about the source. -1 unactivate source. | 0 |
| T1 | K | Temperature of the Maxwellian source, 0=none | 0 |
| flux_file_perAA | 1 | When true (1), indicates that flux file data is already per Angstroem. If false, file data is per wavelength bin. | 0 |
| flux_file_log | 1 | When true, will transform the flux table in log scale to improve the sampling. | 0 |
| Lmin | AA | Minimum wavelength of neutrons | 0 |
| Lmax | AA | Maximum wavelength of neutrons | 0 |
| Emin | meV | Minimum energy of neutrons | 0 |
| Emax | meV | Maximum energy of neutrons | 0 |
| T2 | K | Second Maxwellian source Temperature, 0=none | 0 |
| I2 | 1/(cm**2*sr*AA) | Second Maxwellian Source flux | 0 |
| T3 | K | Third Maxwellian source Temperature, 0=none | 0 |
| I3 | 1/(cm**2*sr*AA) | Third Maxwellian Source flux | 0 |
| zdepth | m | Source z-zdepth, not anymore flat | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

Source input can be an ASCII file!!

Example:

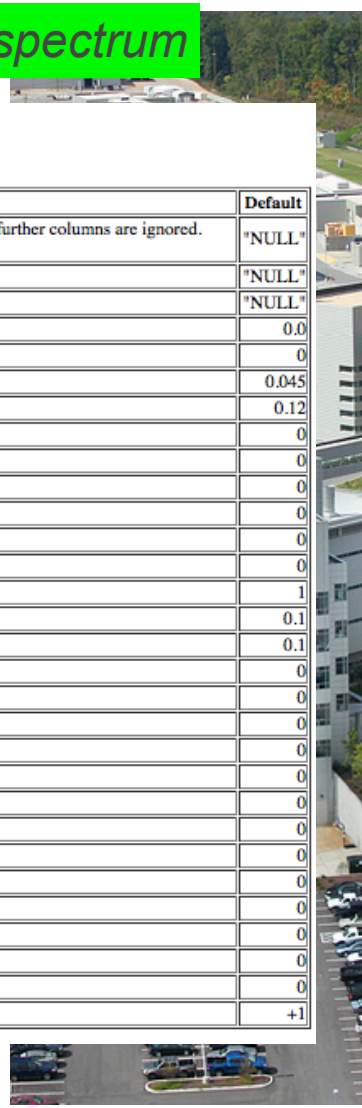
```
COMPONENT my_source_gen = Source_gen(yheight=0.1, xwidth=0.1,
                          dist=1.5, focus_xw=0.01, focus_yh=0.01,
                          lambda0=1, dlambda=8,
                          flux_file='source.dat')
```

# Source_Maxwell 3 ~ Source_gen

October 2018

## Source_Maxwell_3.comp

*A continuous source with a maxwellian spectrum*



### Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| flux_file | str | Name of a two columns [lambda flux] text file that contains the wavelength distribution of the flux in **either** [1/(s*cm**2*st)] **or** [1/(s*cm**2*st*AA)] (see flux_file_perAA flag) Comments (#) and further columns are ignored. Format is compatible with McStas/PGPLOT wavelength monitor files. When specified, temperature and intensity values are ignored. | "NULL" |
| xdiv_file | str | Name of the x-horiz. divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: xmin xmax xdiv_min xdiv_max' | "NULL" |
| ydiv_file | str | Name of the y-vert. divergence distribution file, given as a free format text matrix, preceeded with a line '# xylimits: ymin ymax ydiv_min ydiv_max' | "NULL" |
| radius | m | Radius of circle in (x,y,0) plane where neutrons are generated. You may also use 'yheight' and 'xwidth' for a square source | 0.0 |
| dist | m | Distance to target along z axis. | 0 |
| focus_xw | m | Width of target. | 0.045 |
| focus_yh | m | Height of target. | 0.12 |
| focus_aw | deg | maximal (uniform) horz. width divergence | 0 |
| focus_ah | deg | maximal (uniform) vert. height divergence | 0 |
| E0 | meV | Mean energy of neutrons. | 0 |
| dE | meV | Energy spread of neutrons, half width. | 0 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength spread of neutrons,half width | 0 |
| I1 | 1/(cm**2*sr*AA) | Source flux per solid angle, area and Angstrom if I1=0, the source emits 1 in 4*PI whole space. | 1 |
| yheight | m | Source y-height, then does not use radius parameter | 0.1 |
| xwidth | m | Source x-width, then does not use radius parameter | 0.1 |
| verbose | 0/1 | display info about the source. -1 unactivate source. | 0 |
| T1 | K | Temperature of the Maxwellian source, 0=none | 0 |
| flux_file_perAA | 1 | When true (1), indicates that flux file data is already per Angstroem. If false, file data is per wavelength bin. | 0 |
| flux_file_log | 1 | When true, will transform the flux table in log scale to improve the sampling. | 0 |
| Lmin | AA | Minimum wavelength of neutrons | 0 |
| Lmax | AA | Maximum wavelength of neutrons | 0 |
| Emin | meV | Minimum energy of neutrons | 0 |
| Emax | meV | Maximum energy of neutrons | 0 |
| T2 | K | Second Maxwellian source Temperature, 0=none | 0 |
| I2 | 1/(cm**2*sr*AA) | Second Maxwellian Source flux | 0 |
| T3 | K | Third Maxwellian source Temperature, 0=none | 0 |
| I3 | 1/(cm**2*sr*AA) | Third Maxwellian Source flux | 0 |
| zdepth | m | Source z-zdepth, not anymore flat | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |

# Source_Maxwell_3

October 2018

Source_Maxwell_3.comp

A continuous source with a maxwellian spectrum

## Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| size | m | Edge of cube shaped source (for backward compatibility) | 0 |
| yheight | m | Height of rectangular source | 0 |
| xwidth | m | Width of rectangular source | 0 |
| **Lmin** | AA | Lower edge of lambda distribution | |
| **Lmax** | AA | Upper edge of lambda distribution | |
| **dist** | m | Distance from source to focusing rectangle; at (0,0,dist) | |
| **focus_xw** | m | Width of focusing rectangle | |
| **focus_yh** | m | Height of focusing rectangle | |
| **T1** | K | 1st temperature of thermal distribution | |
| T2 | K | 2nd temperature of thermal distribution | 300 |
| T3 | K | 3nd temperature of - - - | 300 |
| **I1** | in flux units, see above | [1/(cm**2*st*AA)] flux, 1 | |
| I2 | in flux units, see above | [1/(cm**2*st*AA)] flux, 2 | 0 |
| I3 | 1/(cm**2*st*AA) | flux, 3 - - - | 0 |
| target_index | 1 | relative index of component to focus at, e.g. next is +1 this is used to compute 'dist' automatically. | +1 |
| lambda0 | AA | Mean wavelength of neutrons. | 0 |
| dlambda | AA | Wavelength spread of neutrons. | 0 |

```
COMPONENT source = Source_Maxwell_3(yheight=0.156, xwidth=0.126,
                   Lmin=0.1, Lmax=9.0, dist=1.5, focus_xw = 0.025,
                   focus_yh = 0.12, T1=296.16, I1=8.5E11,
              T2=40.68, I2=5.2E11)
```

# Special sources



## Special Sources

*"Feedback" components:*

- *Adapt_check.comp*
- *Source_adapt.comp*
- *Source_Optimizer.comp*
- *Monitor_Optimizer.comp*

*I/O mechanisms:*

- *Virtual_input.comp*
- *Virtual_output.comp*
- ***Virtual_mcnp_ss_input.comp***
- *Virtual_tripoli4_input.comp*
- ***Virtual_mcnp_ss_output.comp***
- *Virtual_tripoli4_output.comp*
- *Vitess_input.comp*
- *Vitess_output.comp*
- *MCPL_input.comp (useful for MCNP/McStas coupling (*))*
- *MCPL_output.comp (useful for MCNP/McStas coupling (*))*

(*) Kittelmann, et. al (2017) https://arxiv.org/abs/1609.02792

# Special sources

October 2018

## Special Sources

*"Feedback" components:*

- *Adapt_check.comp*
- *Source_adapt.comp*
- *Source_Optimizer.comp*
- *Monitor_Optimizer.comp*

**Need more practical info on that?**
**Ask the instructors (could say something on friday)**

- *Virtual_output.comp*
- ***Virtual_mcnp_ss_input.comp***
- *Virtual_tripoli4_input.comp*
- ***Virtual_mcnp_ss_output.comp***
- *Virtual_tripoli4_output.comp*
- *Vitess_input.comp*
- *Vitess_output.comp*
- *MCPL_input.comp (useful for MCNP/McStas coupling (*))*
- *MCPL_output.comp (useful for MCNP/McStas coupling (*))*

(*) Kittelmann, et. al (2017) https://arxiv.org/abs/1609.02792

# Monitors

October 2018

**Detectors and monitors**

| *Name* | *Origin* | *Author(s)* | *Source code* | *Description* |
|---|---|---|---|---|
| DivLambda_monitor | Risoe | Kristian Nielsen | comp | Divergence/wavelength monitor. |
| DivPos_monitor | Risoe | Kristian Nielsen | comp | Divergence/position monitor (acceptance diagram). |
| Divergence_monitor | Risoe | Kim Lefmann | comp | Horizontal+vertical divergence monitor. |
| EPSD_monitor | Risoe | Kim Lefmann | comp | A monitor measuring neutron intensity vs. position, x, and neutron energy, E |
| E_monitor | Risoe | Kristian Nielsen and Kim Lefmann | comp | Energy-sensitive monitor. |
| Hdiv_monitor | Risoe | KL, | comp | A divergence sensitive monitor. |
| L_monitor | Risoe | Kristian Nielsen and Kim Lefmann | comp | Wavelength-sensitive monitor. |
| MeanPolLambda_monitor | Risoe | Peter Christiansen | comp | Polarisation and wavelength sensitive monitor. |
| Monitor | Risoe | Kim Lefmann | comp | Simple single detector/monitor. |
| Monitor_4PI | Risoe | Kim Lefmann and Kristian Nielsen | comp | Monitor that detects ALL non-absorbed neutrons. Example: Monitor_4PI() |
| Monitor_nD | ILL | Emmanuel Farhi | comp | This component is a general Monitor that can output 0/1/2D signals (Intensity or signal vs. [something] and vs. [something] ...) |
| PSD_monitor | Risoe | Kim Lefmann | comp | Position-sensitive monitor. |
| PSD_monitor_4PI | Risoe | Kim Lefmann and Kristian Nielsen | comp | Spherical position-sensitive detector. |
| PSDcyl_monitor | Risoe | Kim Lefmann | comp | A 2D Position-sensitive monitor. The shape is cylindrical with the axis vertical. The monitor covers the whole cylinder (360 degrees). |
| PSDlin_monitor | Risoe | Kim Lefmann | comp | Rectangular 1D PSD, measuring intensity vs. vertical position, x |
| PolLambda_monitor | Risoe | Peter Christiansen | comp | Polarisation and wavelength sensitive monitor. |
| Pol_monitor | Risoe | Peter Christiansen | comp | Polarisation sensitive monitor. |
| PreMonitor_nD | ILL (France) | Emmanuel Farhi | comp | Neutron parameters cross-correlation monitor. |
| Res_monitor | Risoe | Kristian Nielsen | comp | Monitor for resolution calculations |
| TOF2E_monitor | Risoe | Kim Lefmann and Helmuth Schoeber | comp | TOF-sensitive monitor, converting to energy |
| TOFLambda_monitor | Risoe | KL | comp | Time-of-flight/wavelength monitor. |
| TOF_cylPSD_monitor | Risoe | Kim Lefmann | comp | Cylindrical (2pi) PSD Time-of-flight monitor. |
| TOF_monitor | Risoe | KN, M. Hagen | comp | Rectangular Time-of-flight monitor. |
| TOFlog_mon | Risoe | Kim Lefmann | comp | Rectangular Time-of-flight monitor with logarithmic time binning. |

# Monitors

October 2018

## Semantics

- *In reality:*
  - *Monitor: intensity probe of the beam; transparent to neutrons; has efficiency <1%*
  - *Detector: should detect each and every neutron; as high efficiency as possible.*

- *In McStas:*
  - *In simulations we can program monitors and detectors to behave any way we like. We refer to both of those indistinguishably as 'monitors'.*

  *(With exception of PSD_Detector that models a "physical" $He^3$ detector)*

# L_monitor



L_monitor.comp

1-D wavelength sensitive monitor

## Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| nL | 1 | Number of wavelength channels | 20 |
| filename | text | Name of file in which to store the detector image | 0 |
| xmin | m | Lower x bound of detector opening | -0.05 |
| xmax | m | Upper x bound of detector opening | 0.05 |
| ymin | m | Lower y bound of detector opening | -0.05 |
| ymax | m | Upper y bound of detector opening | 0.05 |
| xwidth | m | Width of detector. Overrides xmin,xmax. | 0 |
| yheight | m | Height of detector. Overrides ymin,ymax. | 0 |
| **Lmin** | AA | Minimum wavelength to detect | |
| **Lmax** | AA | Maximum wavelength to detect | |
| restore_neutron | 1 | If set, the monitor does not influence the neutron state | 0 |

**Example:**

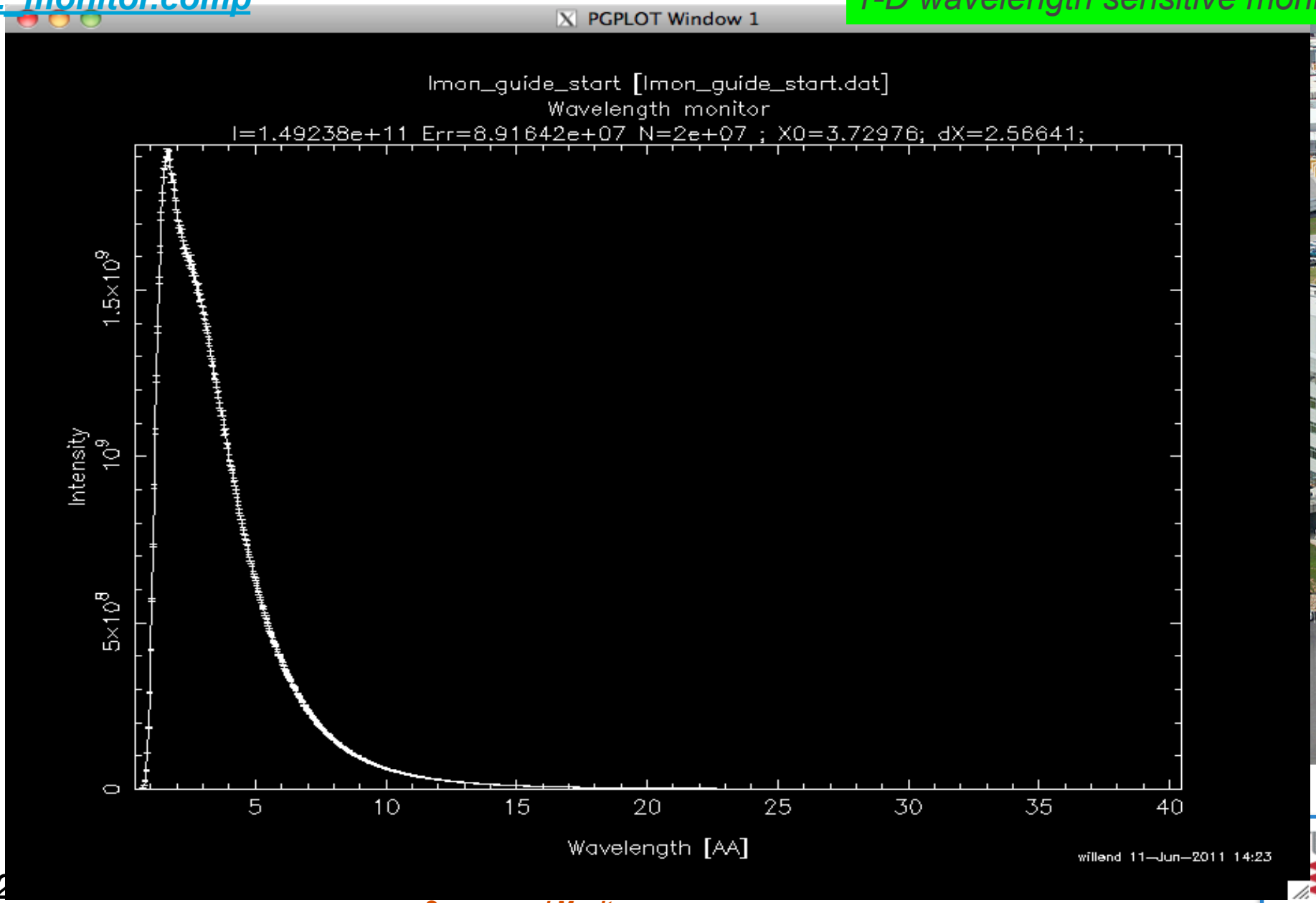
```
COMPONENT my_L_monitor = L_monitor(xmin=-0.1, xmax=0.1, ymin=-0.1,
                         ymax=0.1, nL=20, filename="Output.L",
                         Lmin=2, Lmax=10)
```

# L_monitor

October 2018

[L_monitor.comp](#)

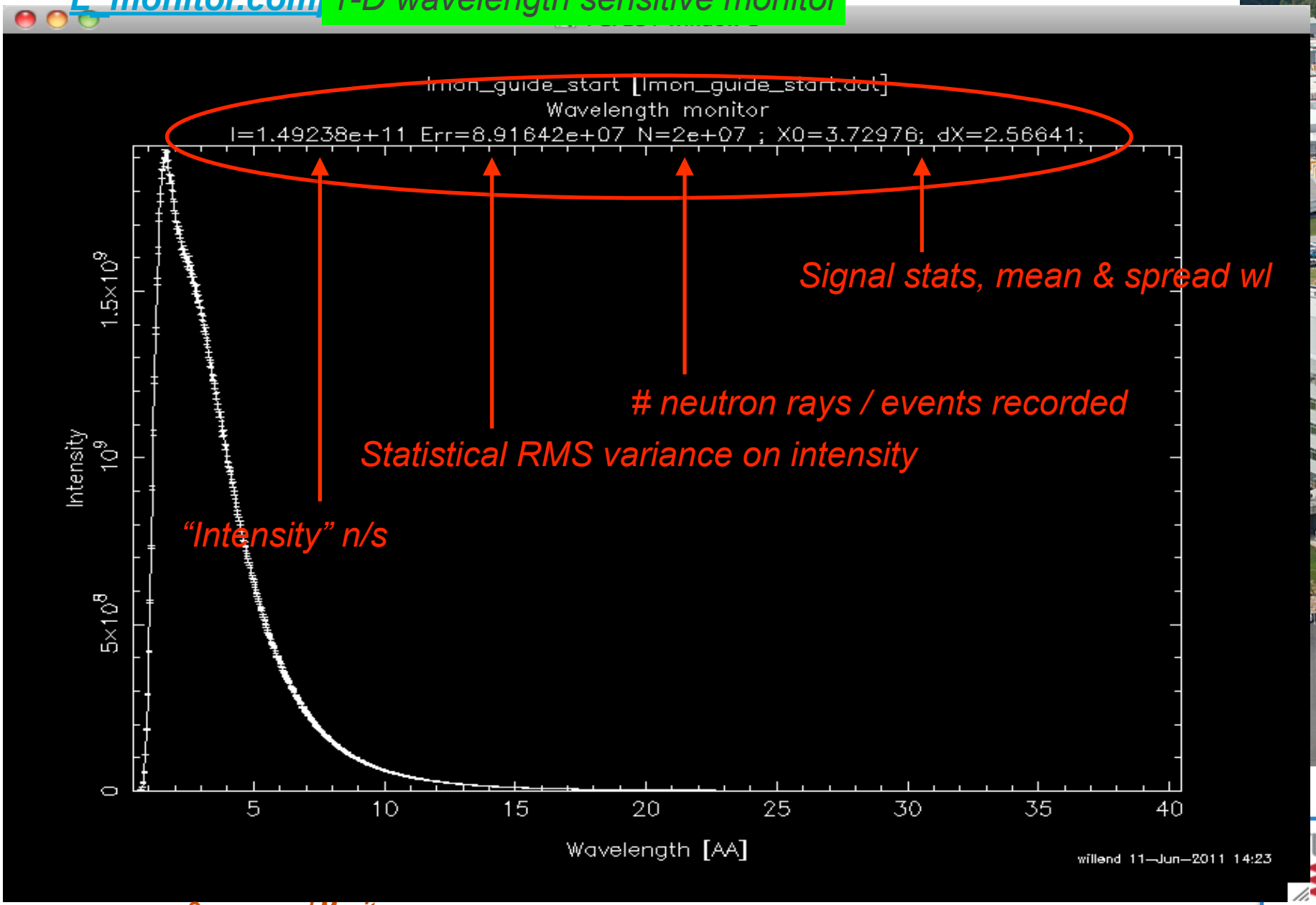
1-D wavelength sensitive monitor




# L_monitor



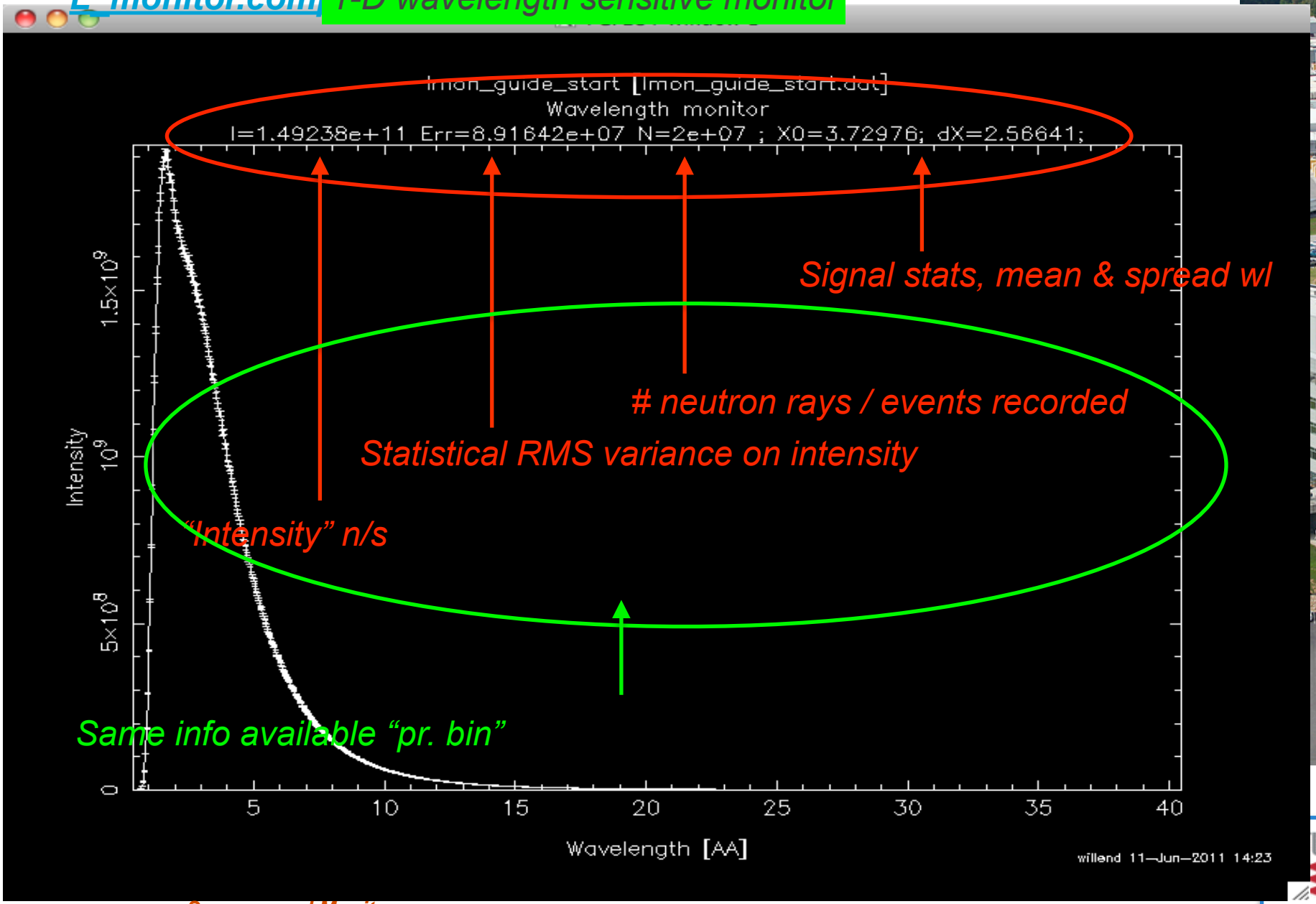
L_monitor.com 1-D wavelength sensitive monitor




# L_monitor

October 2018

L_monitor.com 1-D wavelength sensitive monitor




# PSD_monitor



PSD_monitor.comp

2-D position sensitive monitor

## Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| nx | 1 | Number of pixel columns | 90 |
| ny | 1 | Number of pixel rows | 90 |
| restore_neutron | 1 | If set, the monitor does not influence the neutron state | 0 |
| filename | text | Name of file in which to store the detector image | 0 |
| xmin | m | Lower x bound of detector opening | -0.05 |
| xmax | m | Upper x bound of detector opening | 0.05 |
| ymin | m | Lower y bound of detector opening | -0.05 |
| ymax | m | Upper y bound of detector opening | 0.05 |
| xwidth | m | Width of detector. Overrides xmin,xmax. | 0 |
| yheight | m | Height of detector. Overrides ymin,ymax. | 0 |

**Example:**

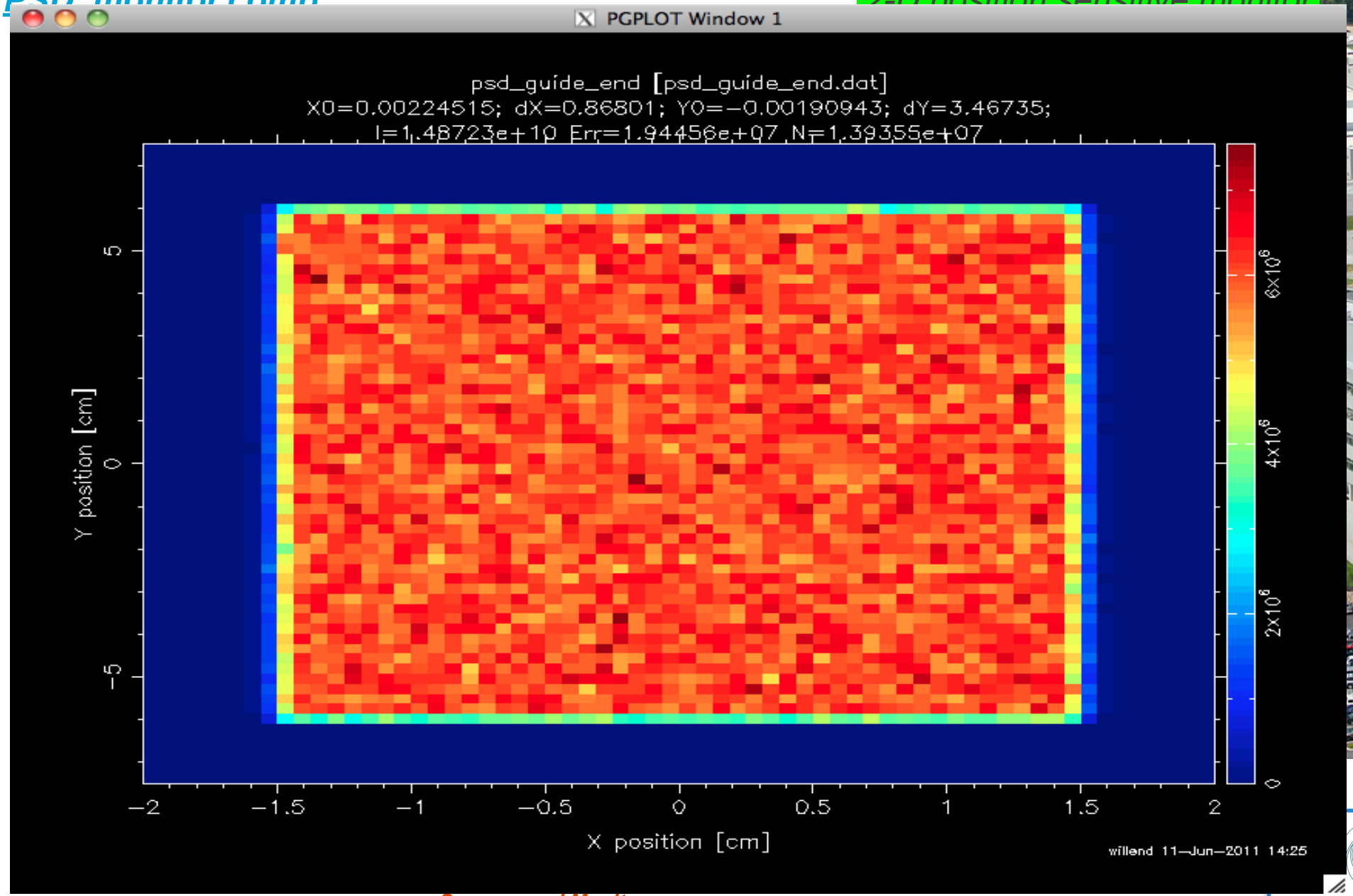
```
COMPONENT my_PSD_monitor = PSD_monitor(xmin=-0.1, xmax=0.1, ymin=-0.1
                    ymax=0.1, nx=90, ny=90,
                    filename="Output.psd")
```

# PSD_monitor

October 2018

PSD_monitor.comp

2-D position sensitive monitor




Sources and Monitors

# More monitors



## More Monitors

- TOF_monitor.comp
- E_monitor.comp (1-D energy sensitive monitor)
- Res_monitor.comp
- TOFLambda_monitor.comp (2-D TOF vs. wavelength monitor)
- Divergence_monitor.comp (2-D divergence monitor)
- EPSD_monitor.comp
- DivPos_monitor.comp (2-D divergence and position monitor)
- Brilliance_monitor.comp
- DivLambda_monitor.comp
- Monitor.comp
- Hdiv_monitor.comp
- PSDlin_monitor.comp
- PSD_monitor_4PI.comp
- Monitor_Sqw.comp
- Pol_monitor.comp
- Monitor_4PI.comp
- .....
- Monitor_nD

# More monitors



## More Monitors

- TOF_monitor.comp
- E_monitor.comp (1-D energy sensitive monitor)
- Res_monitor.comp
- TOFLambda_monitor.comp (2-D TOF vs. wavelength monitor)
- Divergence_monitor.comp (2-D divergence monitor)
- EPSD_monitor.comp
- DivPos_monitor.comp (2-D divergence and position monitor)
- Brilliance_monitor.comp
- DivLambda_monitor.comp
- Monitor.comp
- Hdiv_monitor.comp
- PSDlin_monitor.comp
- PSD_monitor_4PI.comp
- Monitor_Sqw.comp
- Pol_monitor.comp
- Monitor_4PI.comp
- .....
- Monitor_nD

# More monitors



## More Monitors

- TOF_monitor.comp
- E_monitor.comp (1-D energy sensitive monitor)
- Res_monitor.comp
- TOFLambda_monitor.comp (2-D TOF vs. wavelength monitor)
- Divergence_monitor.comp (2-D divergence monitor)
- EPSD_monitor.comp
- DivPos_monitor.comp (2-D divergence and position monitor)
- Brilliance_monitor.comp
- DivLambda_monitor.comp
- Monitor.comp
- Hdiv_monitor.comp
- PSDlin_monitor.comp
- PSD_monitor_4PI.comp
- Monitor_Sqw.comp
- Pol_monitor.comp
- Monitor_4PI.comp
- .....
- Monitor_nD

# Monitor_nD

October 2018

Monitor_nD.comp

A general monitor for 0D/1D/2D records

*The all-in-one , swiss-army-knife of monitors*

***Monitor_nD*** *can have almost any shape, and record any requested standard quantities*

# Monitor_nD

October 2018

[Monitor_nD.comp](#)


A general monitor for 0D/1D/2D records

## Input parameters

Parameters in **boldface** are required; the others are optional.

| Name | Unit | Description | Default |
|---|---|---|---|
| user1 | variable | Variable assigned to User1 | FLT_MAX |
| user2 | variable | Variable assigned to User2 | FLT_MAX |
| user3 | variable | Variable assigned to User3 | FLT_MAX |
| xwidth | m | Width of detector. | 0 |
| yheight | m | Height of detector. | 0 |
| zdepth | m | Thickness of detector (z). | 0 |
| xmin | m | Lower x bound of opening | 0 |
| xmax | m | Upper x bound of opening | 0 |
| ymin | m | Lower y bound of opening | 0 |
| ymax | m | Upper y bound of opening | 0 |
| zmin | m | Lower z bound of opening | 0 |
| zmax | m | Upper z bound of opening | 0 |
| bins | 1 | Number of bins to force for all variables. Use 'bins' keyword in 'options' for heterogeneous bins | 0 |
| min | u | Minimum range value to force for all variables Use 'min' or 'limits' keyword in 'options' for other limits | -1e40 |
| max | u | Maximum range value to force for all variables Use 'max' or 'limits' keyword in 'options' for other limits | 1e40 |
| restore_neutron | 0\|1 | If set, the monitor does not influence the neutron state. Equivalent to setting the 'parallel' option. | 0 |
| radius | m | Radius of sphere/banana shape monitor | 0 |
| options | str | String that specifies the configuration of the monitor The general syntax is "[x] options..." (see **Descr.**). | "NULL" |
| filename | str | Output file name (overrides file=XX option). | "NULL" |
| geometry | str | Name of an OFF file to specify a complex geometry detector | "NULL" |
| username1 | str | Name assigned to User1 | "NULL" |
| username2 | str | Name assigned to User2 | "NULL" |
| username3 | str | Name assigned to User3 | "NULL" |

EUROPEAN SPALLATION SOURCE

# Monitor_nD

October 2018

Monitor_nD.comp


A general monitor for 0D/1D/2D records

**EXAMPLES:**

```
COMPONENT MyMon = Monitor_nD( xwidth = 0.1, yheight = 0.1, zdepth = 0,
                  options = "intensity per cm2 angle,limits=[-5 5],
                  bins=10,with borders, file = mon1")
              options = "banana, theta limits=[10,130], bins=120, y"

options = "multiple kx ky kz, auto abs log t, and list all neutrons"
```

```
COMPONENT MyMo = Monitor_nD(xwidth = 0.1, yheight = 0.1,
                    user1=age, username1="Age of the Captain
                  [years]",
                    options="user1, auto")
```

- *Mathematical:*
  - *Source_simple.comp*
  - *Source_div.comp*

- *Pulsed sources:*
  - *ESS_moderator.comp*
  - *Moderator.comp*
  - *SNS_source.comp (*)*
  - *ISIS_moderator.comp (*)*

*Reactors :*

- *Source_Maxwell_3.comp*
- *Source_gen.comp*
- *Source_gen4.comp*
- *Source_multi_surfaces.comp (*)*

*(*) contributed (can be found in /mcstas/installation/folder/**contrib** )*

EXERCISE 1

# A job for you..

*To be performed at beginning of the split session for all*

- *Create a new instrument file, named 'sources_monitors_ex.instr'.*
  *HINT: you can use the 'template.instr' file that exists in today's class folder and you can open it with the editor of your liking, **or** open it from the McStas gui by clicking either:*
  *File —> New (python) or*
  *neutron site --> Templates --> template_simple (perl)*

- *Add a source using the Source_gen.comp component, with:*
  - *source dimensions: (w)0.132m X (h)0.164m*
  - *distance to target : 1.5 m*
  - *focus area: (w)0.03m X (h)0.12m*
  - *wavelength range: 0.1Å to 9.9Å*
  - *T1=27.63[K], I1=2.4E12 [n/s/cm²/st/AA], T2=130.76[K], I2=4.03E12[n/s/cm²/st/AA] ,T3=309.33[K], I3=1.24E13[n/s/cm²/st/AA]*

- *Add the following monitors at two different distances from the source, at 1.5m and 4.5m:*
  - *PSD monitor (PSD_monitor)*
  - *A linear PSD monitor for the y-direction (PSDlin_monitor)*
  - *Wavelength monitor (L_monitor)*
  - *2D Divergence monitor (Divergence_monitor)*
  - *Divergence-position monitor for the x-direction (DivPos_monitor)*