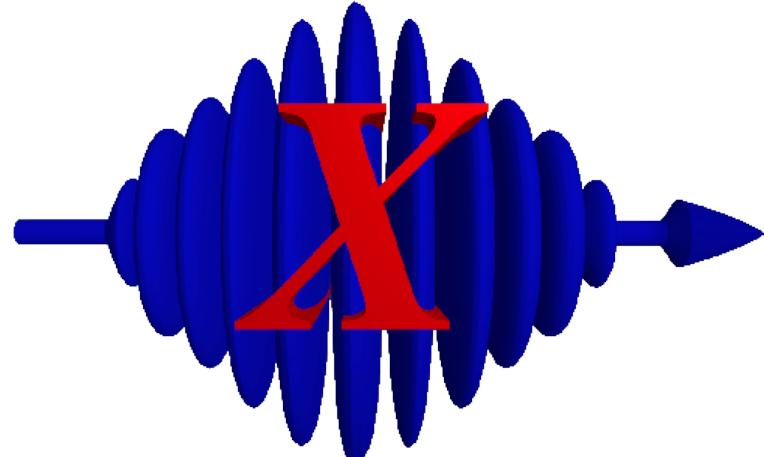




# McXtrace



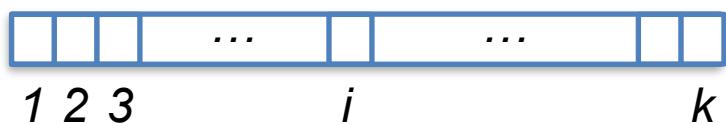
FAQ, advanced grammar, tips and tricks

Peter Willendrup ([pkwi@fysik.dtu.dk](mailto:pkwi@fysik.dtu.dk))



- ❖ McXtrace sources generally provide “intensity” in units of photons/s (into a chosen solid angle)
- ❖ That intensity is carried through the instrument on a discrete set of “X-ray rays”

- Imagine a histogram, e.g.  $I(\lambda)$

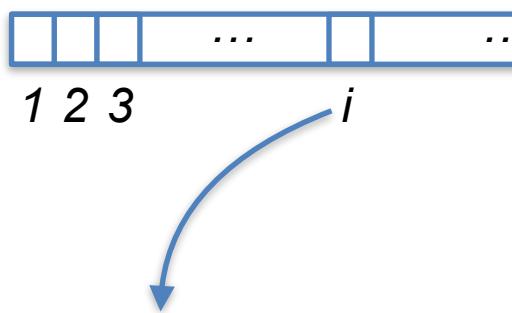


*In bin  $i$ ,  $N$  events each carrying a fractional intensity  $p_j$  so that*

$$I = \frac{\sum p_j}{N}$$

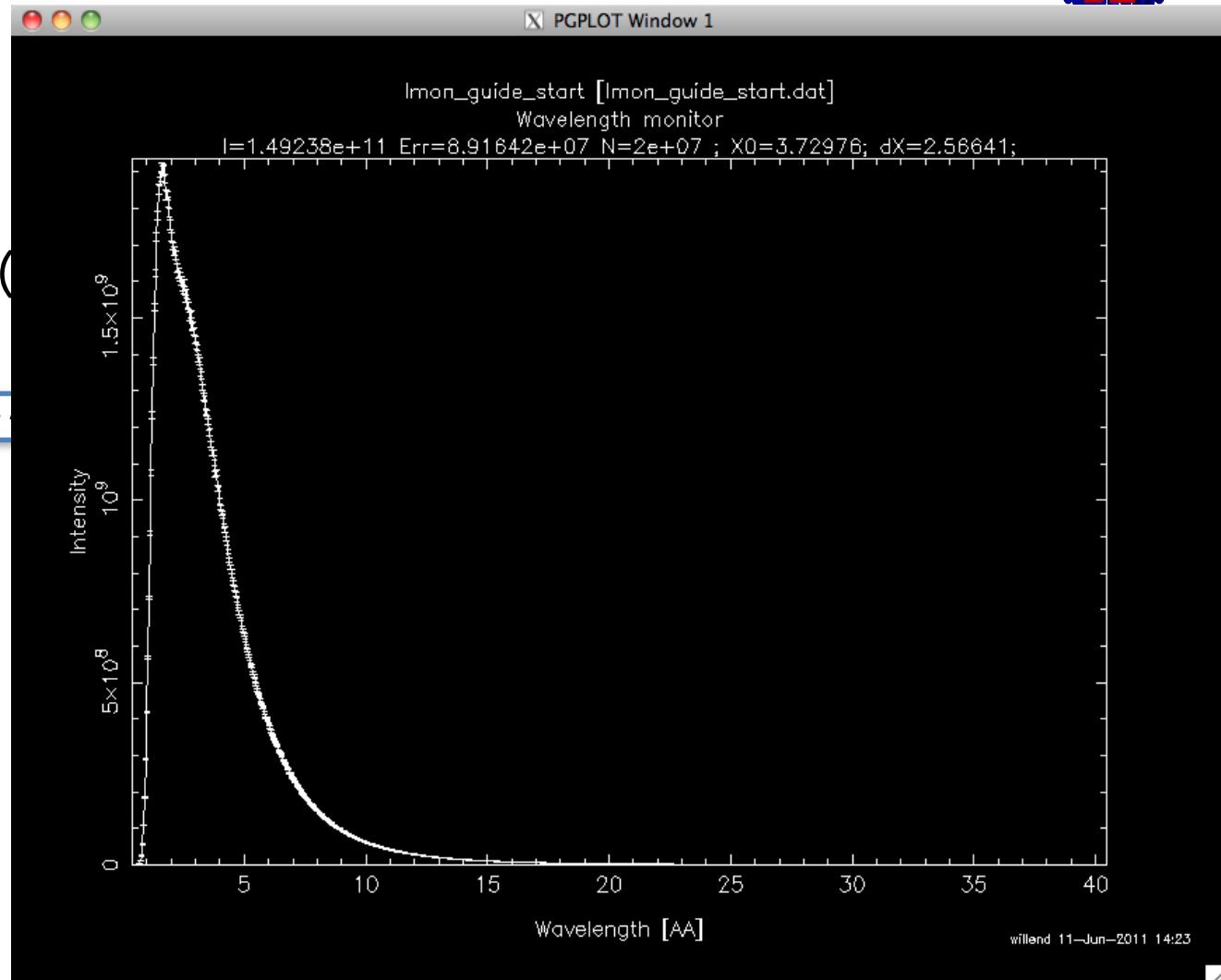
- The RMS variance over that set becomes our statistical error bar  $E$

Imagine a histogram, e.g.  $I($



*In bin  $i$ ,  $N$  events each carrying a fractional intensity  $p_j$  so that*

$$I = \sum_N p_j$$



The RMS variance over that set becomes our statistical error bar  $E$

From "Virtual experiments - the ultimate aim of neutron ray-tracing simulations", K. Lefmann et al., Journal of Neutron Research 16, 97-111 (2008)

Let  $n$  be the number of neutron rays reaching the detector, and let the rays have (different) weights,  $w_i$ . The simulated intensity is then given by

$$I = \sum_{i=1}^n w_i. \quad (1)$$

The estimate of the error on this number is calculated in the McStas manual [1], and the standard deviation is approximated by

$$\sigma^2(I) = \sum_{i=1}^n w_i^2. \quad (2)$$

In real experiments,  $w_i = 1$ , whence we reach  $I = n$  and  $\sigma(I) = \sqrt{I}$  as expected (for counts exceeding 10). Let the virtual time be denoted by  $t$ . The simulated counts during this time becomes

$$C = tI, \quad (3)$$

From "Virtual experiments - the ultimate aim of neutron ray-tracing simulations", K. Lefmann et al., Journal of Neutron Research 16, 97-111 (2008)

and its error bar estimate is

$$\sigma^2(C) = t^2 \sigma^2(I). \quad (4)$$

However, to simulate a realistic counting statistics, we must fulfill

$$\sigma_{\text{VE}}(C_{\text{VE}}) = \sqrt{C_{\text{VE}}}. \quad (5)$$

This is obtained by adding to (3) a Gaussian noise  $E(\Sigma)$  of mean value zero and standard deviation  $\Sigma$ :

$$C_{\text{VE}} = tI + E(\Sigma). \quad (6)$$

The standard deviation for the VE becomes

$$\sigma_{\text{VE}}^2(C) = t^2 \sigma^2(I) + \Sigma^2. \quad (7)$$

Now, the requirement (5) allows us to determine  $\Sigma$ :

$$\Sigma^2 = tI - t^2 \sigma^2(I). \quad (8)$$

Since  $\Sigma^2$  must remain positive, we reach an upper limit on  $t$

$$t_{\max} = \frac{I}{\sigma^2(I)}. \quad (9)$$

1. On a given McXtrace histogram
2. For the non-zero bins, calculate  $t_{\max} = \frac{I}{\sigma^2(I)}.$
3. The *smallest*  $t_{\max}$  defines the “maximal counting time” allowed by your statistics
4. Preferably a “background” should be added - use a “known experimental value” or an estimate...

1. Your simulation will only contain elements you provided / defined
2. ... to the precision you defined
3. Answers the questions you posed
4. Background essentially only from “sample”, or sample-near objects

Lefmann, K., Willendrup, P. K., Uddy, L., Lebech, B., Mortensen, K., Birk, J. O., ... Farhi, E. (2008).  
*Virtual experiments: the ultimate aim of neutron ray-tracing simulations.*  
Journal of Neutron Research, 16(4-4), 97. <https://doi.org/10.1080/10238160902819684>

Plus the attached note from Annette Vickery:

A note about McStas errorbars

A. Vickery, P. K. Willendrup and E. Knudsen

January 31, 2014

**Contents**

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Estimation of counting times and derivation of virtual data from simulations</b>	<b>2</b>
2.1	Real vs simulated data . . . . .	2
2.2	Counting time and data quality . . . . .	3
2.3	Estimating the appearance of a “real” data set . . . . .	3
2.4	Example . . . . .	3

- ♦ Apply focusing techniques
  - ♦ At the source (spatially, temporally, in wavelength...)
  - ♦ At the sample, if possible
- ♦ (carefully!) Apply SPLIT - but only if immediately followed by Monte Carlo choices, e.g. in sample
- ♦ Alternatively use MCPL o/i which allows repetition - beware of biases!

- ♦ Apply focusing techniques
  - ♦ At the source (spatially, temporally, in wavelength...)
  - ♦ At the sample, if possible
- ♦ (carefully!) Apply SPLIT - but only if immediately followed by Monte Carlo choices, e.g. in sample
- ♦ Alternatively use MCPL o/i which allows repetition - beware of biases!

*All of this can be considered "variance reduction"  
or biasing*

- ❖ Use MPI parallelisation - included in macOS install, easy to get on Linux...
- ❖ The Intel C compiler is known to give ~factor of 2 wrt. gcc in most cases
- ❖ - **Still** consider if you are asking the right question if runtimes reach days/weeks...

- mcsub cluster scripts

```
./mcsub_slurm.pl
Usage: ./mcsub_slurm.pl [options] [mcrun params]
-h      --help          Show this help
-rN     --runtime=N     Specify maximum runtime (hours) [default 1]
-qQNAME --queue=QNAME  Specify wanted SLURM queue [default 'express']
-e<mail> --email=<mail> Specify address to notify in reg. sim status [default
none]
--nodes=NUM            Specify wanted number of nodes [default 1]
--name=NAME            Specify slurm job name [default
"McSub_<USERNAME>_<TIMESTAMP>"]

After running ./mcsub_slurm.pl NAME.batch is ready for submission using the sbatch
command
```

- Takes a “mxrun commandline”
- Writes batch file “template” for use with PBS or slurm cluster queue systems
- <https://github.com/McStasMcXtrace/McCode/tree/master/tools/cluster-scripts>

## *Examples Directory*

- *Simple grep* →
  - *25 examples of Monitor\_nD*
  - *4 examples of Source\_lab*
  - *4 examples of Undulator*
  - *5 examples of PowderN*

*Unix/Linux:*

```
grep Monitor_nD /usr/share/mcxtrace/1.5/examples/*
```

## *Examples Directory*

- *Simple grep* →

*They have an author name!*

*For instance:*

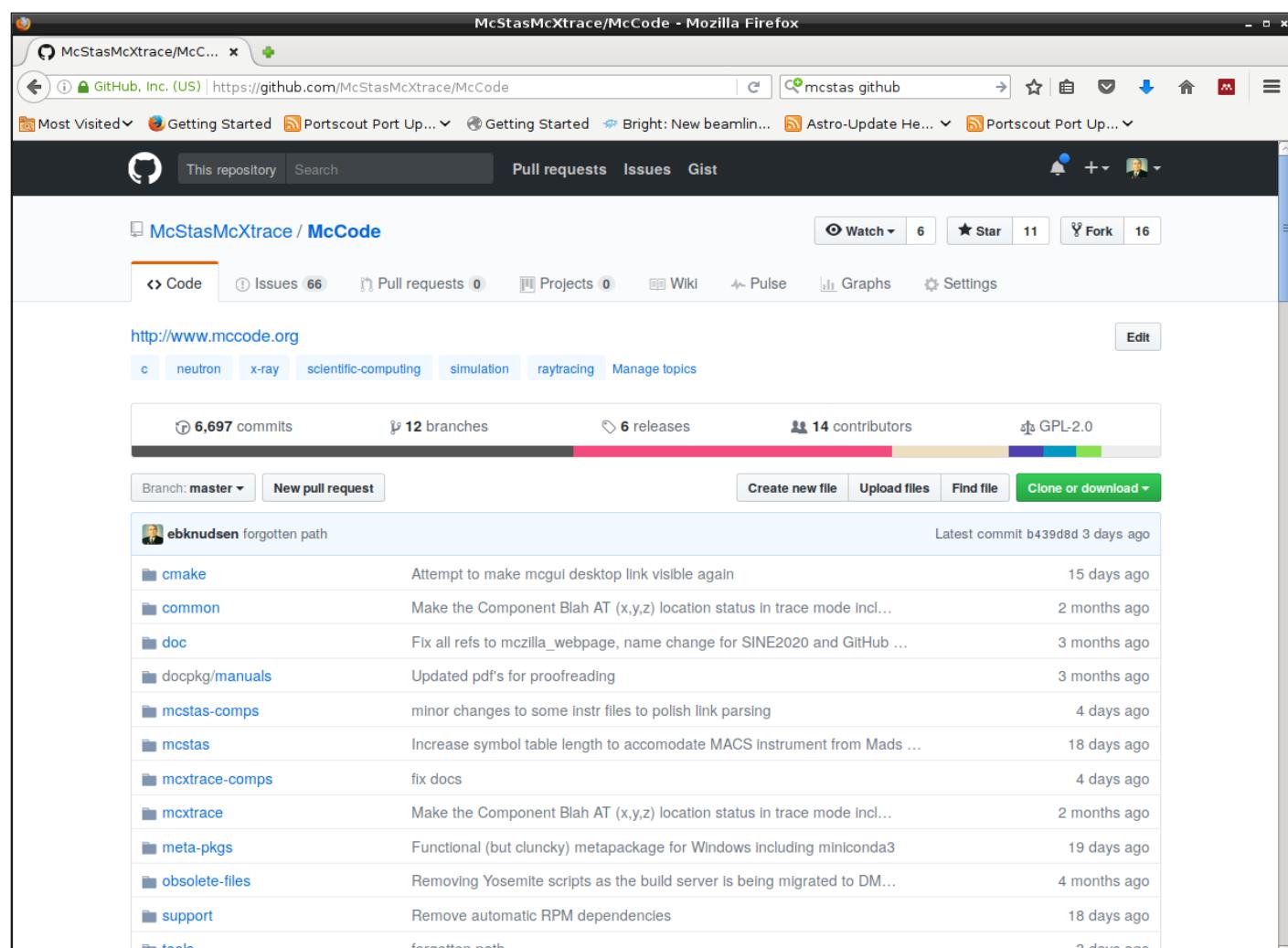
*DBD\_IBM\_Si\_analyzer* → *Marcus H. Mendenhall*

*Test\_SX* → *E. Farhi*

*Pump\_probe* → *E. Knudsen*

*GitHub.com*

*<https://github.com/McStasMcXtrace/McCode>*



McStasMcXtrace/McCode - Mozilla Firefox

GitHub, Inc. (US) | https://github.com/McStasMcXtrace/McCode

This repository Search Pull requests Issues Gist

Watch 6 Star 11 Fork 16

McStasMcXtrace / McCode

Code Issues 66 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

http://www.mccode.org Edit

c neutron x-ray scientific-computing simulation raytracing Manage topics

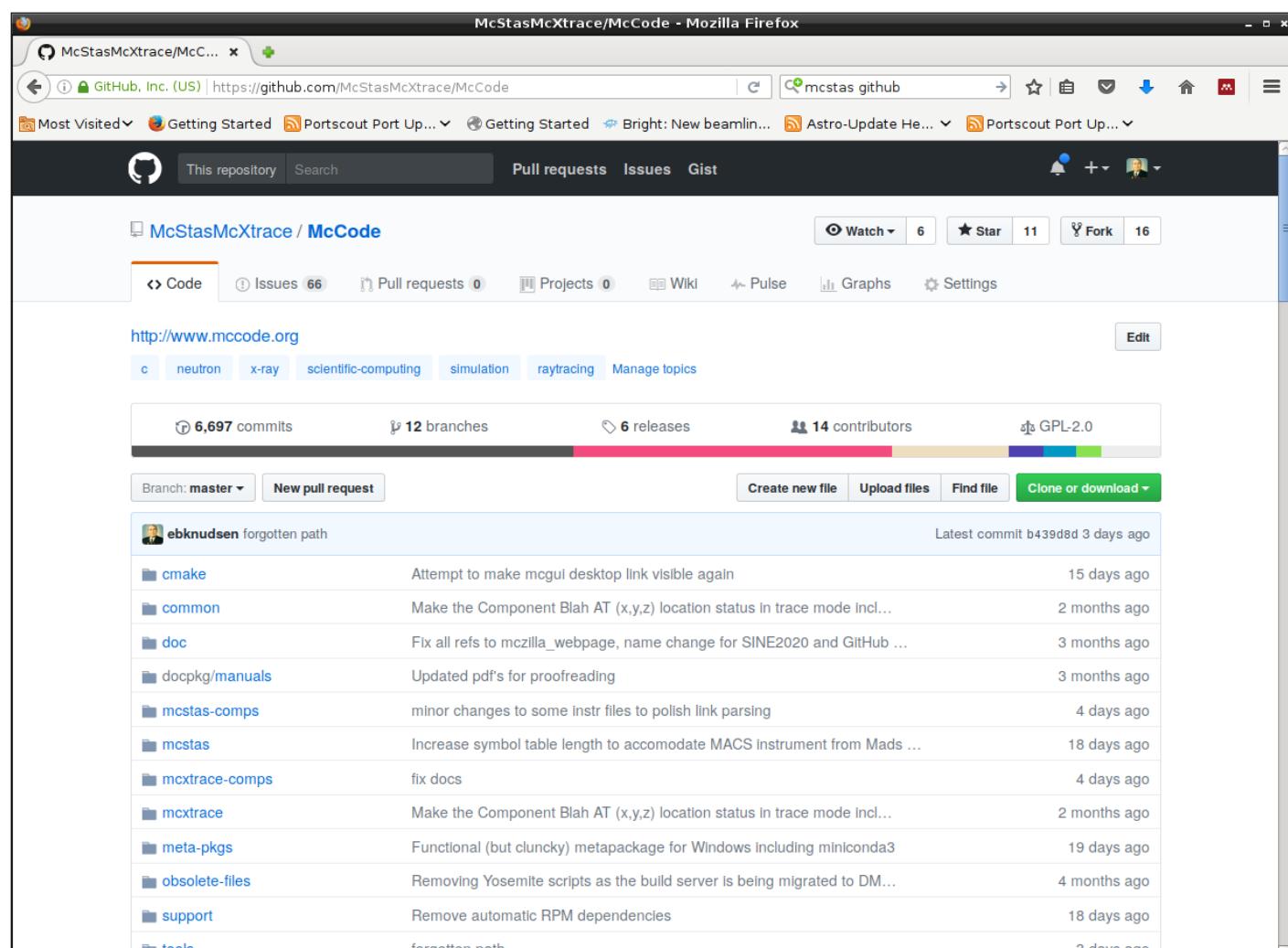
6,697 commits 12 branches 6 releases 14 contributors GPL-2.0

Branch: master New pull request Create new file Upload files Find file Clone or download

Author	Commit Message	Date
ebknudsen	forgotten path	Latest commit b439d8d 3 days ago
cmake	Attempt to make mcgui desktop link visible again	15 days ago
common	Make the Component Blah AT (x,y,z) location status in trace mode incl...	2 months ago
doc	Fix all refs to mczilla_webpage, name change for SINE2020 and GitHub ...	3 months ago
docpkg/manuals	Updated pdf's for proofreading	3 months ago
mcstas-comps	minor changes to some instr files to polish link parsing	4 days ago
mcstas	Increase symbol table length to accomodate MACS instrument from Mads ...	18 days ago
mcxtrace-comps	fix docs	4 days ago
mcxtrace	Make the Component Blah AT (x,y,z) location status in trace mode incl...	2 months ago
meta-pkgs	Functional (but cluncky) metapackage for Windows including miniconda3	19 days ago
obsolete-files	Removing Yosemite scripts as the build server is being migrated to DM...	4 months ago
support	Remove automatic RPM dependencies	18 days ago
tools	forgotten path	3 days ago

*GitHub.com*

*<https://github.com/McStasMcXtrace/McCode>*



The screenshot shows the GitHub repository page for `McStasMcXtrace/McCode`. The repository has 6,697 commits, 12 branches, 6 releases, and 14 contributors. The master branch is selected, showing recent commits from `ebknudsen` and others. The repository is licensed under GPL-2.0.

Commit	Description	Date
<code>cmake</code>	Attempt to make mcgui desktop link visible again	15 days ago
<code>common</code>	Make the Component Blah AT (x,y,z) location status in trace mode incl...	2 months ago
<code>doc</code>	Fix all refs to mczilla_webpage, name change for SINE2020 and GitHub ...	3 months ago
<code>docpkg/manuals</code>	Updated pdf's for proofreading	3 months ago
<code>mcstas-comps</code>	minor changes to some instr files to polish link parsing	4 days ago
<code>mcstas</code>	Increase symbol table length to accomodate MACS instrument from Mads ...	18 days ago
<code>mcxtrace-comps</code>	fix docs	4 days ago
<code>mcxtrace</code>	Make the Component Blah AT (x,y,z) location status in trace mode incl...	2 months ago
<code>meta-pkgs</code>	Functional (but cluncky) metapackage for Windows including miniconda3	19 days ago
<code>obsolete-files</code>	Removing Yosemite scripts as the build server is being migrated to DM...	4 months ago
<code>support</code>	Remove automatic RPM dependencies	18 days ago
<code>tools</code>	forgotten path	2 days ago

## Welcome to the McCode wiki!

### General information

- How McStas/McXtrace works overview
- Known issues and FAQ's
- Access to the code tree

### Developer guides

- Developer literature
- Building McStas/McXtrace
- Builds and platforms overview
- Release procedure/notes
- UI test checklist
- How to use Eclipse with PyDev
- Debugging the c-code

### For users & comp developers

- McStas tutorial: simplified SANS instrument
- Writing Components
- Single crystal - and generating its input
- Generate Vitess modules via mcstas2vitess

### Interlinks with other software

- McStas and Mantid
- How to update the MCPL shipped with McStas/McXtrace (McStas and MCNP/Geant4/SIMRES/...)
- MCPL-related issue on Ubuntu 17.04 (McStas and MCNP/Geant4/SIMRES/...)

### Documentation on the McCode tools

- User documentation for the 2017- Python tool set'
- mcrun variants - table overview
- mcplot variants - table overview
- mcdisplay variants - table overview



### Guides for McWeb (e-neutrons.org simulator and more)

- <https://github.com/McStasMcXtrace/McWeb/wiki>

▼ Pages 33
<input type="text" value="Find a Page..."/>
<a href="#">Home</a>
<a href="#">Access to the code tree</a>
<a href="#">Building McStas McXtrace</a>
<a href="#">Builds and platforms overview</a>
<a href="#">Debugging the c code</a>
<a href="#">Developer literature</a>
<a href="#">Generate Vitess modules via mcstas2vitess</a>
<a href="#">How McStas McXtrace works overview</a>
<a href="#">How to update the MCPL shipped with McStas McXtrace</a>
<a href="#">How to use Eclipse with PyDev</a>
<a href="#">Known issues and FAQ's</a>
<a href="#">mcdisplay pyqtgraph</a>
<a href="#">mcdisplay variants table overview</a>
<a href="#">mcdisplay webgl</a>
<a href="#">mcgui</a>
<a href="#">Show 18 more pages...</a>

+ Add a custom sidebar

Clone this wiki locally

<https://github.com/McStasMcXtrace/McCode-Wiki> 

## Welcome to the McCode wiki!

### General information

- How McStas/McXtrace works overview
- Known issues and FAQ's
- Access to the code tree

### Developer guides

- Developer literature
- Building McStas/McXtrace
- Builds and platforms overview
- Release procedure/notes
- UI test checklist
- How to use Eclipse with PyDev
- Debugging the c-code

### For users & comp developers

- McStas tutorial: simplified SANS instrument
- Writing Components
- Single crystal - and generating its input
- Generate Vitess modules via mcstas2vitess

### Interlinks with other software

- McStas and Mantid
- How to update the MCPL shipped with McStas/McXtrace (McStas/MCNP/Geant4/SIMRES/...)
- MCPL-related issue on Ubuntu 17.04 (McStas and MCNP/Geant4/SIMRES/...)

### Documentation on the McCode tools

- User documentation for the 2017- Python tool set
- mcrun variants - table overview
- mcplot variants - table overview
- mcdisplay variants - table overview

### Guides for McWeb (e-neutrons.org simulator and more)

- <https://github.com/McStasMcXtrace/McWeb/wiki>

## mcdisplay variants table overview

Peter Willendrup edited this page on Jan 18, 2018 · 7 revisions

[Edit](#) [New Page](#)

Default trace visualiser "mcdisplay" indicated in bold below

More columns if you scroll... ----->

Variant	Type	2D/3D	Special cmdline switches
mcdisplay-pyqtgraph	Python 3 + pyqtgraph	2D x 3 planes	--invcanvas (invert colors)
mcdisplay-webgl	Python 3 + WebGL	Full 3D	--nobrowse to avoid spawning browser
mcdisplay-mantid	Python 3 + Mantid	3D	None
mcdisplay.pl (mcplot-pl on Windows)	Perl 5 + PGPlot	2D x 3 with PGPlot, 3D with Matlab	-pPLOTTER --format=PLOTTER can be used to forward output to PGPlot, Gnuplot, Matlab, ... -ps/-psc/-gif save hardcopy --complete  When outputting XML, also describe component geometry --tmax=VAL  ToF axis limit when in --TOF mode
Plotters	below	this	not

▼ Pages 33

Find a Page...

[Home](#)

[Access to the code tree](#)

[Building McStas McXtrace](#)

[Builds and platforms overview](#)

[Debugging the c code](#)

[Developer literature](#)

[Generate Vitess modules via mcstas2vitess](#)

[How McStas McXtrace works overview](#)

[How to update the MCPL shipped with McStas McXtrace](#)

[How to use Eclipse with PyDev](#)

[Known issues and FAQ's](#)

[mcdisplay pyqtgraph](#)

[mcdisplay variants table overview](#)

[mcdisplay webgl](#)

[mcgui](#)

Show 18 more pages...

+ Add a custom sidebar

Clone this wiki locally

<https://github.com/McStasMcXtrace/McCode/wiki> 

McStasMcXtrace / McCode

Unwatch 11 Unstar 30 Fork 29

Code Issues 194 Pull requests 0 Actions Projects 9 Wiki Security

Insights Settings

Filters is:issue is:open Labels 55 Milestones 0 New issue

① 194 Open	Author	Labels	Projects	Milestones	Assignee
✓ 610 Closed					
Sort ▾					
① MxDisplay/MxPlot with matlab: missing 'mxdisplay' and 'mxplot' functions	C: McCode tools	P: minor bug			
#902 opened 12 minutes ago by farhi					
① McDisplay 3D rendering: not adapted to very long instruments				1	
P: minor bug					
#901 opened 14 minutes ago by farhi					
① McXtrace: Lens_parab display is very slow.	McXtrace 1.5				
annownce					
#900 opened 15 hours ago by ebknudsen					
① cif2hkl: Propagate fixes present in iFit to McCode.					
#899 opened 17 hours ago by ebknudsen					
① McXtrace: Lens_parab Lens shadow is square - consider changing to round	McXtrace 1.5	Picked for McXtrace 1.6	enhancement		
#898 opened 17 hours ago by ebknudsen					
① McXtrace 1.5: mxdoc (py) apparently missing in suite-package(s)				1	
#897 opened 21 hours ago by willend					
① Messed-up axes with pyqtgraph plotter on Windows 10					
#896 opened 2 days ago by willend					
① 2D->1D logic (in case of e.g. nx=1 in PSD_monitor) seems to add a '' to generated xvar					
#895 opened 2 days ago by willend					
① McXtrace PowderN: reuse Tau-search when in SPLIT as in McStas	C: McXtrace component	P: minor	Picked for McXtrace 1.6		
enhancement					
#894 opened 5 days ago by farhi					
① McXtrace Single crystal: reuse Tau-search when in SPLIT as in McStas	C: McXtrace component	P: minor	Picked for McXtrace 1.6		
enhancement					
#893 opened 5 days ago by farhi					

## Mailing List



# McXtrace - An X-ray ray-trace simulation package



### McXtrace

Download  
Components  
Linux Installation  
(deb/rpm)  
Mac OSX Installation  
Unix Install (src code)  
Windows Installation

**Project Status**  
Project Partners  
Project People  
Goal

Mailing list

Links

Publications

Minutes of Meetings

McXtrace Art

About McXtrace

**Documentation**  
Commands  
Tutorial

**Search**

### **mcxtrace-users mailing list**

To subscribe to mcxtrace-users, fill in your email address and press subscribe:

To access the mailing list interface, go to <https://mailman2.mcxttrace.org/mailman/listinfo/mcxtrace-users>

To access archived messages, go to <https://mailman2.mcxttrace.org/pipermail/mcxtrace-users>.

To search in the mailing list archive you may use: [added](#) to the website!

[Printable version](#)

Last Modified:

[Printable version](#)

Last Modified: Monday, 18-Nov-2019 11:05:20 CET



Integrated SCM & Project Management



## Mailing List

### mcxtrace-users --

#### About mcxtrace-users

English (USA)

To see the collection of prior postings to the list, visit the [mcxtrace-users Archives](#).

#### Using mcxtrace-users

To post a message to all the list members, send email to [mcxtrace-users@mailman2.mcxtrace.org](mailto:mcxtrace-users@mailman2.mcxtrace.org).

You can subscribe to the list, or change your existing subscription, in the sections below.

#### Subscribing to mcxtrace-users

Subscribe to mcxtrace-users by filling out the following form. You will be sent email requesting confirmation, to prevent others from gratuitously subscribing you. This is a private list, which means that the list of members is not available to non-members.

Your email address:

Your name (optional):

You may enter a privacy password below. This provides only mild security, but should prevent others from messing with your subscription. **Do not use a valuable password** as it will occasionally be emailed back to you in cleartext.

If you choose not to enter a password, one will be automatically generated for you, and it will be sent to you once you've confirmed your subscription. You can always request a mail-back of your password when you edit your personal options. Once a month, your password will be emailed to you as a reminder.

Pick a password:

Reenter password to confirm:

Which language do you prefer to display your messages?

English (USA)

Would you like to receive list mail batched in a daily digest?

No  Yes

[Subscribe](#)

#### mcxtrace-users Subscribers

(The subscribers list is only available to the list members.)

Enter your address and password to visit the subscribers list:

Address:

Password:

[Visit Subscriber List](#)

## Mailing List - Archive

# The mcxtrace-users Archives

You can get [more information about this list](#).

Archive	View by:	Downloadable version
Fourth quarter 2019:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 5 KB]</a>
Fourth quarter 2017:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 2 KB]</a>
Second quarter 2017:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 4 KB]</a>
Fourth quarter 2016:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 9 KB]</a>
Third quarter 2016:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 30 KB]</a>
First quarter 2016:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 3 KB]</a>
Fourth quarter 2015:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 17 KB]</a>
Third quarter 2015:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 28 KB]</a>
First quarter 2014:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 1 KB]</a>
Fourth quarter 2013:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 2 KB]</a>
First quarter 2013:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 4 KB]</a>
Fourth quarter 2012:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 4 KB]</a>
Third quarter 2012:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 6 KB]</a>
Fourth quarter 2011:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 2 KB]</a>
Fourth quarter 2010:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 1 KB]</a>
Third quarter 2009:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 969 bytes]</a>
First quarter 2009:	<a href="#">[Thread]</a> <a href="#">[Subject]</a> <a href="#">[Author]</a> <a href="#">[Date]</a>	<a href="#">[Text 3 KB]</a>

User forum and help since 2009!!

Spirit of the mailinglist:

- Very friendly and open!
- There is **no such thing** as a stupid question!
- Often answers are given from users rather than developers, especially in areas that include very specific domain science.
- contact: [mcxtrace-users@mcxtrace.org](mailto:mcxtrace-users@mcxtrace.org)

- Advanced language features:

*Tips and tricks for your  
instrument*



```
{SPLIT} COMPONENT name = comp(parameters) {WHEN condition}  
AT (...) [RELATIVE [reference|PREVIOUS] | ABSOLUTE]  
{ROTATED {RELATIVE [reference|PREVIOUS] | ABSOLUTE} }  
{GROUP group_name}  
{EXTEND C_code}  
{JUMP [reference|PREVIOUS|MYSELF|NEXT] [ITERATE number_of_times  
| WHEN condition] }
```



- Use the DECLARE section define user variables and functions.

```
DECLARE %{
    double myvar;
%}
```

- Use INITIALIZE for initialization of user variables and calculations.

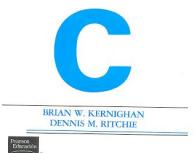
```
INITIALIZE %{
    myvar = sqrt(PI*input_var)*rand01();
%}
```

- - Both use normal c-syntax.

- BEWARE: (example) What you do in the c-style areas is c-standard, e.g. trigonometric functions from math.h use radians! - McXtrace placement specifiers work in degrees, etc...



K & R / GNU



## Useful physics constants:

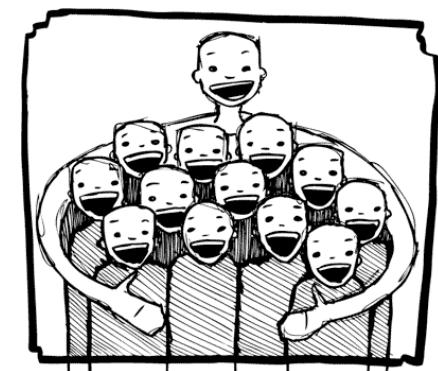
```
#define CELE      1.602176487e-19 /* [C] Elementary charge CODATA 2006*/
#define M_C        299792458   /* [m/s] speed of light CODATA 2006*/
#define E2K        0.506773091264796 /* Convert k[1/AA] to E [keV] (CELE/(HBAR*M_C)*1e-10)*1e3 */
#define K2E        1.97326972808327 /*Convert E[keV] to k[1/AA] (1e10*M_C*HBAR/CELE)/1e3 */
#define RE         2.8179402894e-5  /*[AA] Thomson scattering length*/
```

plus e.g. DEG2RAD, RAD2DEG, and these math constants

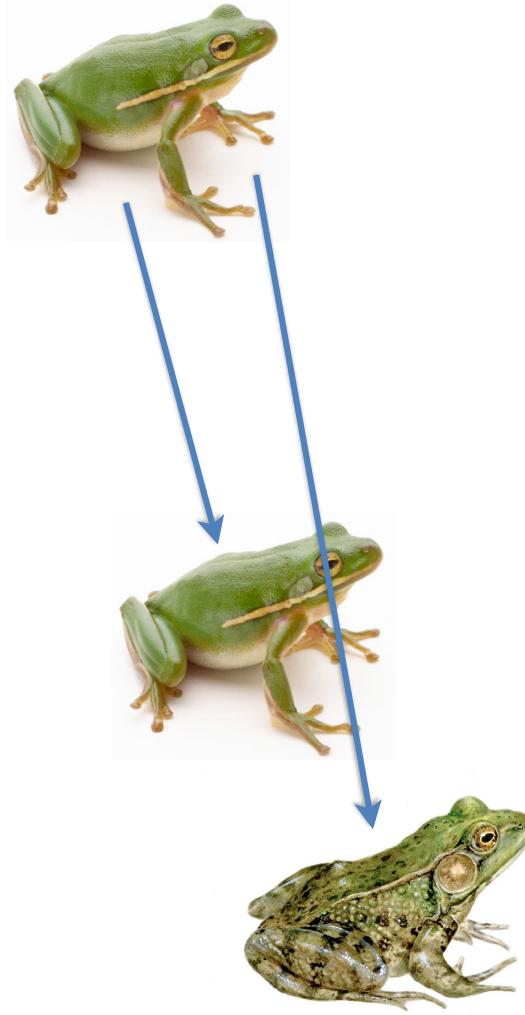
```
( # define PI 3.14159265358979323846 )
# define M_PI PI
# define M_PI_2 M_PI/2.0
# define M_PI_4 M_PI/4.0
# define M_1_PI 1.0/M_PI
# define M_2_PI 2*M_1_PI
# define M_2_SQRTPI 2/sqrt(M_PI)
# define M_SQRT2 sqrt(2)
# define M_SQRT1_2 sqrt(1/2)
# endif
```

plus anything you can imagine in terms of trigonometric functions from C <math.h>  
(beware, these take radians as input - as opposed to our ROTATED statements)

- Instrumentfiles can include external c-code or other instrumentfiles... (These are examples from McStas)
- ILL\_H15\_IN6.instr:%include "monitor\_nd-lib"
- ILL\_H16\_IN5.instr:%include "ILL\_H16.instr"
- ILL\_H25\_IN22.instr:%include "ILL\_H25.instr"
- ILL\_H25\_IN22.instr:%include "templateTAS.instr"
- Used in the DECLARE section



- In instruments: (see e.g. ILL\_H25.instr)
  - COMPONENT H25\_1 = Guide\_gravity(
    - w1=0.03, h1=0.2, w2=0.03, h2=0.2, l=L\_H25\_1,
    - R0=gR0, Qc=gQc, alpha=gAlpha, m=m, W=gW)
  - AT (0,0,AI\_Thickness+gGap) RELATIVE PREVIOUS
  - ROTATED (0,Rh\_H25\_1,0) RELATIVE PREVIOUS
- COMPONENT **MYcopy** = **COPY(H25\_1)**
  - AT (0,0,L\_H25\_1+gGap) RELATIVE PREVIOUS
  - ROTATED (0,Rh\_H25\_1,0) RELATIVE PREVIOUS
- COMPONENT **COPY(H25\_1)** = **COPY(H25\_1)(W=2\*gW)**
  - AT (0,0,L\_H25\_1+gGap) RELATIVE PREVIOUS
  - ROTATED (0,Rh\_H25\_1,0) RELATIVE PREVIOUS





*AT (0,0,-LMM) RELATIVE Cradle ROTATED (0,A1/2,0) RELATIVE Cradle*

**GROUP IN6Monoks**

*AT (0,0,0) RELATIVE Cradle ROTATED (0,A2/2,0) RELATIVE Cradle*

**GROUP IN6Monoks**

- One comp after the particle is “tried” in sequential order until the it becomes SCATTERED.



- Syntax:

COMPONENT Mine = Yours(blah, blah)

**WHEN** (c-expression) AT (...)

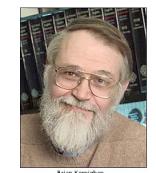
- Is very powerful when combined with EXTEND and user variables, or as a method to let input parameters select if certain components are active.
- Example: Use EXTEND to flag if X-ray was scattered on one monochromator blade or another. Then later use WHEN to only show contribution from blade N at sample position?

COMPONENT Mon = PSD\_monitor(...)

**WHEN** (myvar==1) AT (0,0,0) RELATIVE Sample



BRIAN W. KERNIGHAN  
DENNIS M. RITCHIE



Brian Kernighan



Dennis Ritchie



- Enrich component behaviour using EXTEND:

```
COMPONENT Mono1 = Monochromator_curved(...)
```

```
AT (0,0, -LMM) RELATIVE Cradle ROTATED (0,A1/2,0) RELATIVE Cradle
```

```
GROUP IN6Monoks
```

## EXTEND

```
%{
  if (SCATTERED) { myvar = 1; }
}%
```

```
...
```

```
COMPONENT Mono2 = Monochromator_curved(...)
```

```
AT (0,0, 0) RELATIVE Cradle ROTATED (0,A2/2,0) RELATIVE Cradle
```

```
GROUP IN6Monoks
```

## EXTEND

```
%{
  if (SCATTERED) { myvar = 2 ;}
}%
```



*K & R / GNU*



SEGUNDA EDICIÓN  
COPIA LIBRE  
EL  
 LENGUAJE DE  
 PROGRAMACIÓN

C

BRIAN W. KERNIGHAN  
DENNIS M. RITCHIE  
Prentice Hall

## Combined example: Decompose multiple scattering from Single\_crystal

```

DECLARE %{
    double multiple_scatt;
%}

...
COMPONENT Crystal = Single_crystal(... order=0 ...)
AT (0,0,0) RELATIVE somewhere
EXTEND %{
    multiple_scatt=SCATTERED;
%}

...
COMPONENT PSD_single=PSD_monitor(...)
WHEN (multiple_scatt==1) AT (0,0,0) RELATIVE somewhere_else

COMPONENT PSD_multiple=PSD_monitor(...)
WHEN (multiple_scatt > 1) AT (0,0,0) RELATIVE somewhere_else

```

- A goto. Be careful. Can be used in two situations:
  - **JUMP** to myself
  - **JUMP** to an Arm
- 
- No coordinate transformations are applied... (Meaning that if the Arms you JUMP between do not coincide you will “move” / “reorient” the X-rays...)
- 
- Syntaxes:
  - COMPONENT a=b(...)
  - **WHEN** (expr) AT (...) **JUMP** somewhere
- 
- COMPONENT a=b(...)
  - **WHEN** (expr) AT (...) **JUMP** myself



- A goto. Be careful. Can be used in two situations:
- **JUMP** to myself
- **JUMP** to an Arr

***BEWARE - This IS a GOTO!***

- No coordinate transformations are applied... (Meaning that if the Arms you JUMP between do not coincide you will “move” / “reorient” the X-rays...)
- Syntaxes:
- COMPONENT a=b(...)
- **WHEN** (expr) AT (...) **JUMP** somewhere
- COMPONENT a=b(...)
- **WHEN** (expr) AT (...) **JUMP** myself



- A goto. Be careful. Can be used in two situations:

- **JUMP** to myself
- **JUMP** to an Arr

- No coordinate trans do not coincide you

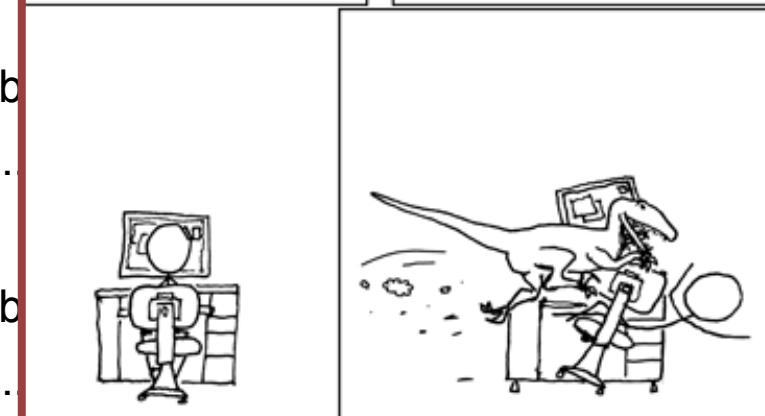
- Syntaxes:
- COMPONENT a=b
- **WHEN** (expr) AT (.

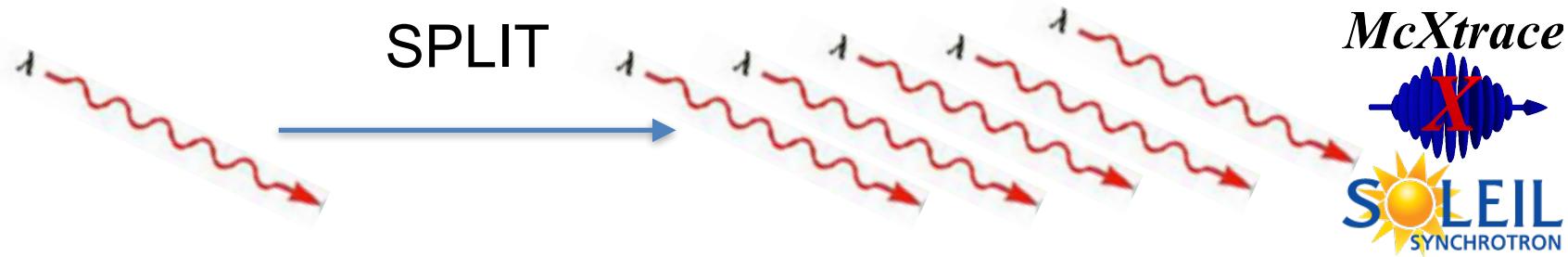
- COMPONENT a=b
- **WHEN** (expr) AT (.



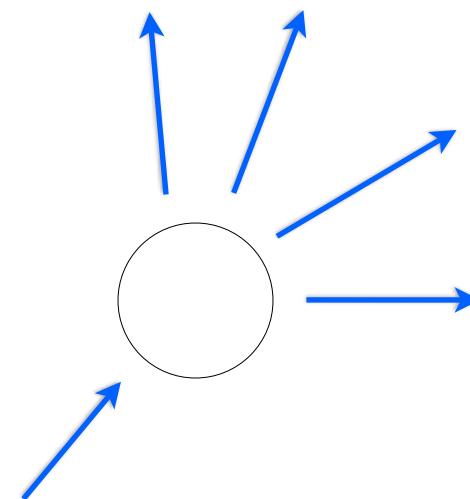
**a GOTO!**

at if the Arms you JUMP between

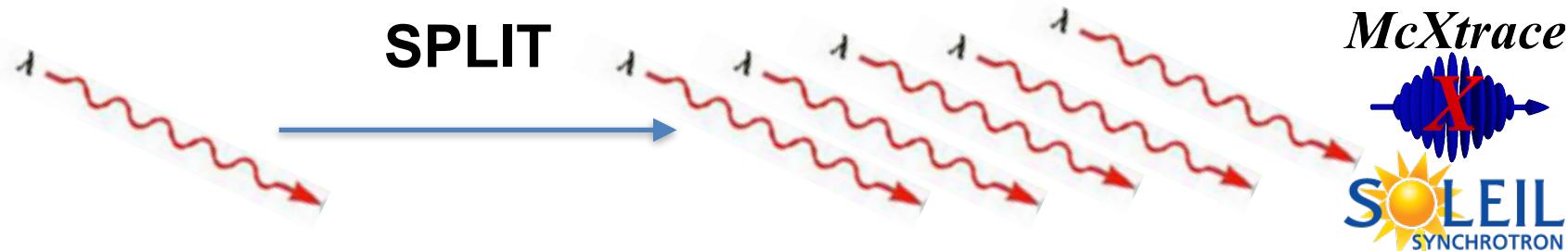




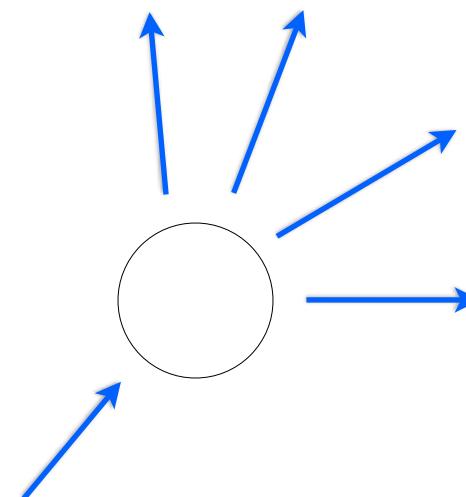
- Increase statistics beyond this point in the instrumentfile
  
- SPLIT n MyArm = Arm()
- AT somewhere
  
- will “formulate an if-statement”:
  
- for j=1:n
- comp1
- comp2
- comp3
- ...
- end (of instrument)
  
- **ONLY** meaningful in case of Monte Carlo choices after SPLIT point...



# SPLIT



- Increase statistics beyond this point in the instrumentfile
- SPLIT n MyArm = Arm()
- AT somewhere
- will “formulate an if-statement”:
- for j=1:n
  - comp1
  - comp2
  - comp3
  - ...
- end (of instrument)
- **ONLY** meaningful in case of Monte Carlo choices after SPLIT point...



Works very well together with e.g. monochromators, Single\_crystal, PowderN

