New features in McStas

Peter Willendrup[1,5], Emmanuel Farhi[2], Erik Knudsen[1,5], Esben Klinkby[3,6] Emmanouela Rantsiou[4,7], Torben Nielsen[4,5,6], Kim Lefmann[4,6]

[1]Physics Department, Technical University of Denmark, Denmark
[2]Calcul Scientifique, Institut Laue-Langevin, France
[3]Center for Nuclear Technologies, Technical University of Denmark, Denmark
[4]Laboratory for Developments and Methods, Paul Scherrer Institute, Switzerland
[4]Niels Bohr Institute, University of Copenhagen, Denmark
[5]ESS Data Management and Software Center,Copenhagen, Denmark
[6]ESS design update programme, Denmark
[7]ESS design update programme, Switzerland

Agenda

- New features in McStas 2.0
- McStas 2.1 and beyond...
- New project areas

Core / runtime highlights

- **Backward compatibility 1:** Some backward-compatibility is lost in the new version of the code-generation, as we no longer use the information defined in the component STATE_PARAMETERS and POLARISATION_PARAMETERS macros. A warning is issued in case of components using these macros

- **Backward compatibility 2**: We have standardized naming of component parameters - see the file NOMENCLATURE installed in the McStas library. The code generation will give error messages of e.g. using xw and yh where one should now be using xwidth and yheight.

- **Polarization 1:** Support for tabulated magnetic fields. New code for handling fast interpolation in sparse data is included in the share/interpolation/ area. A new version of the SE_example.instr takes use of this feature.

- **Polarization 2:** We allow overlapping magnetic fields. All components that use the PROP_ routines for propagation implicitly allows larmor precession.

- **Future keyword:** ASSEMBLY allowing ‘metacomponents’

Future support tool situation

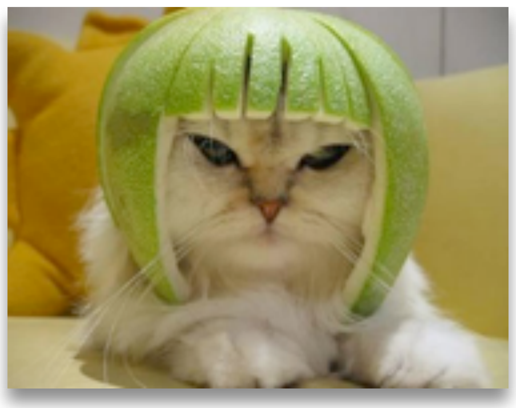
- Basic calc/sim functionality support still provided by C
- Perl tools are there - but should be abandoned

- Python will provide future scripting and glue
- GUI widgets likely wxwidgets or Qt or standard editor??
- Plotting using matplotlib (also considering q'n'd GNUPLOT hooks)
- High-level support tools provided using e.g.

- **Johan Brinch done the first developments in this direction for McStas 2.0**

Modernized build/package system - McStas 2.0

- CMake+CPack replace ./configure && make && home grown scripts
- Supported platforms:



Modernized build/package system - McStas 2.1

- CMake+CPack replace ./configure && make && home grown scripts
- Supported platforms:

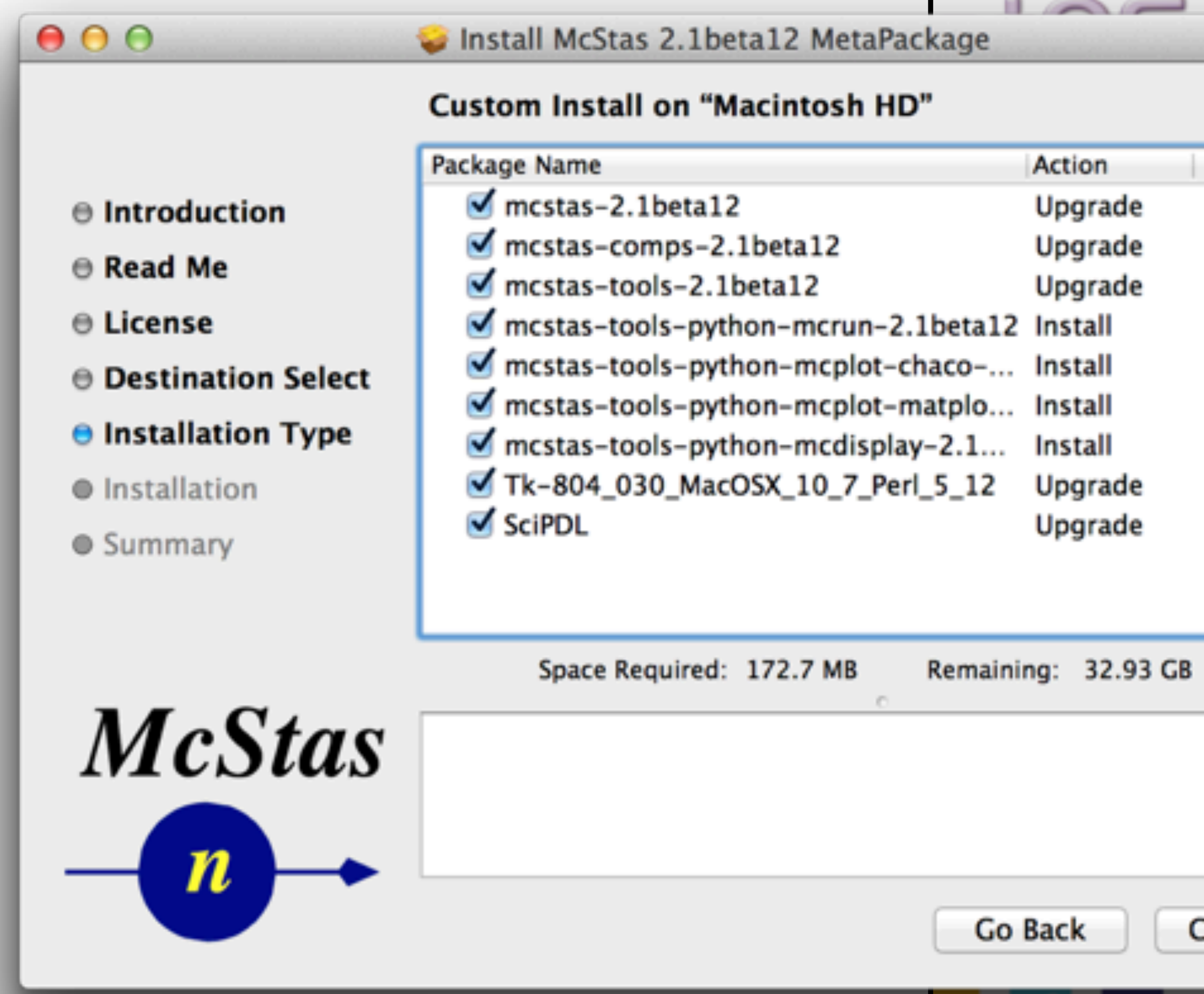
Metapackages!!!

Easy to install on Linux systems via dependency-requirements, i.e.

apt-get install mcstas-suite
apt-get install mcstas-suite-perl
apt-get install mcstas-suite-python

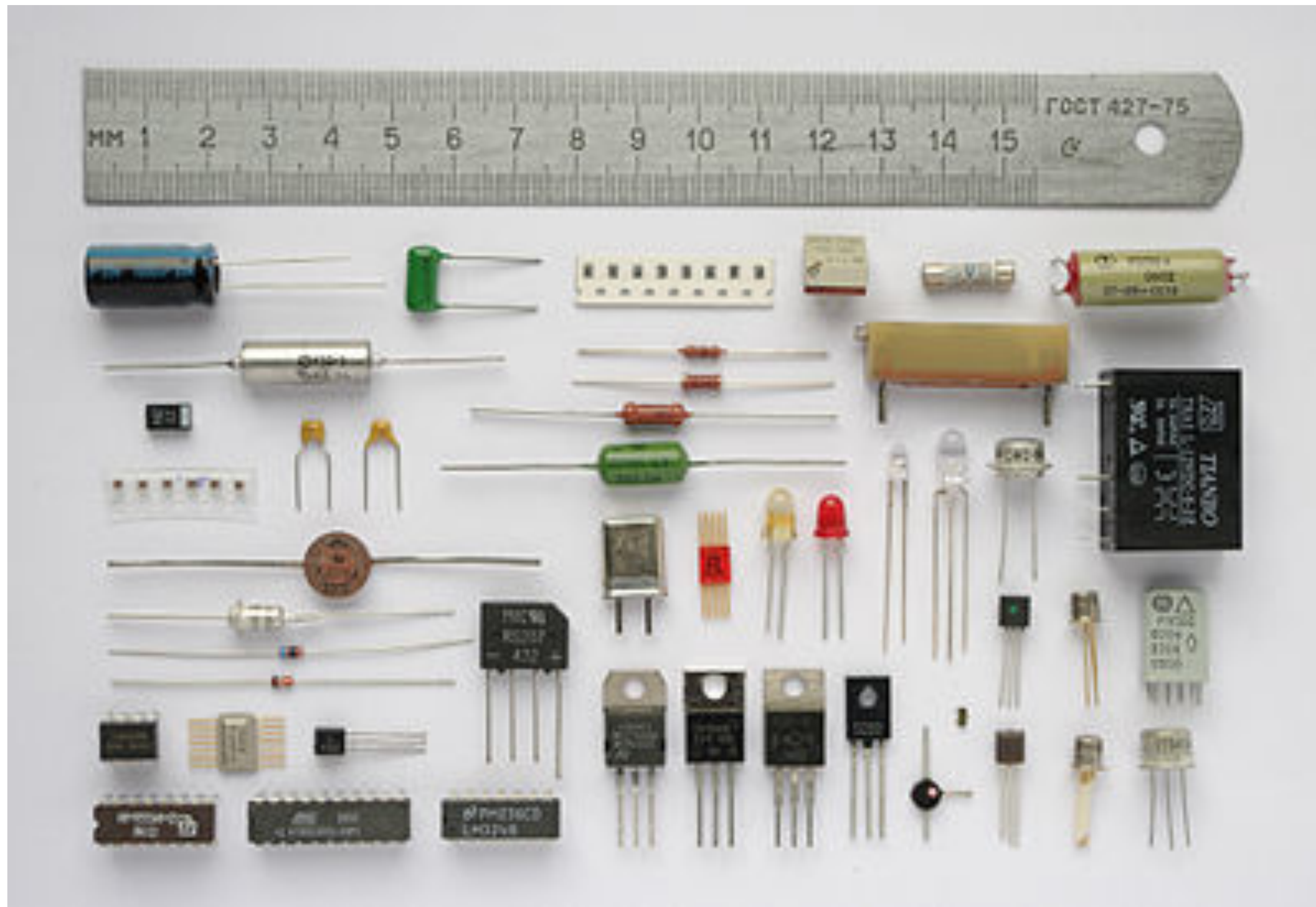
Meta-package now defined on the Mac

Will come for 2.1 on windows also...
- will allow install without internet connection

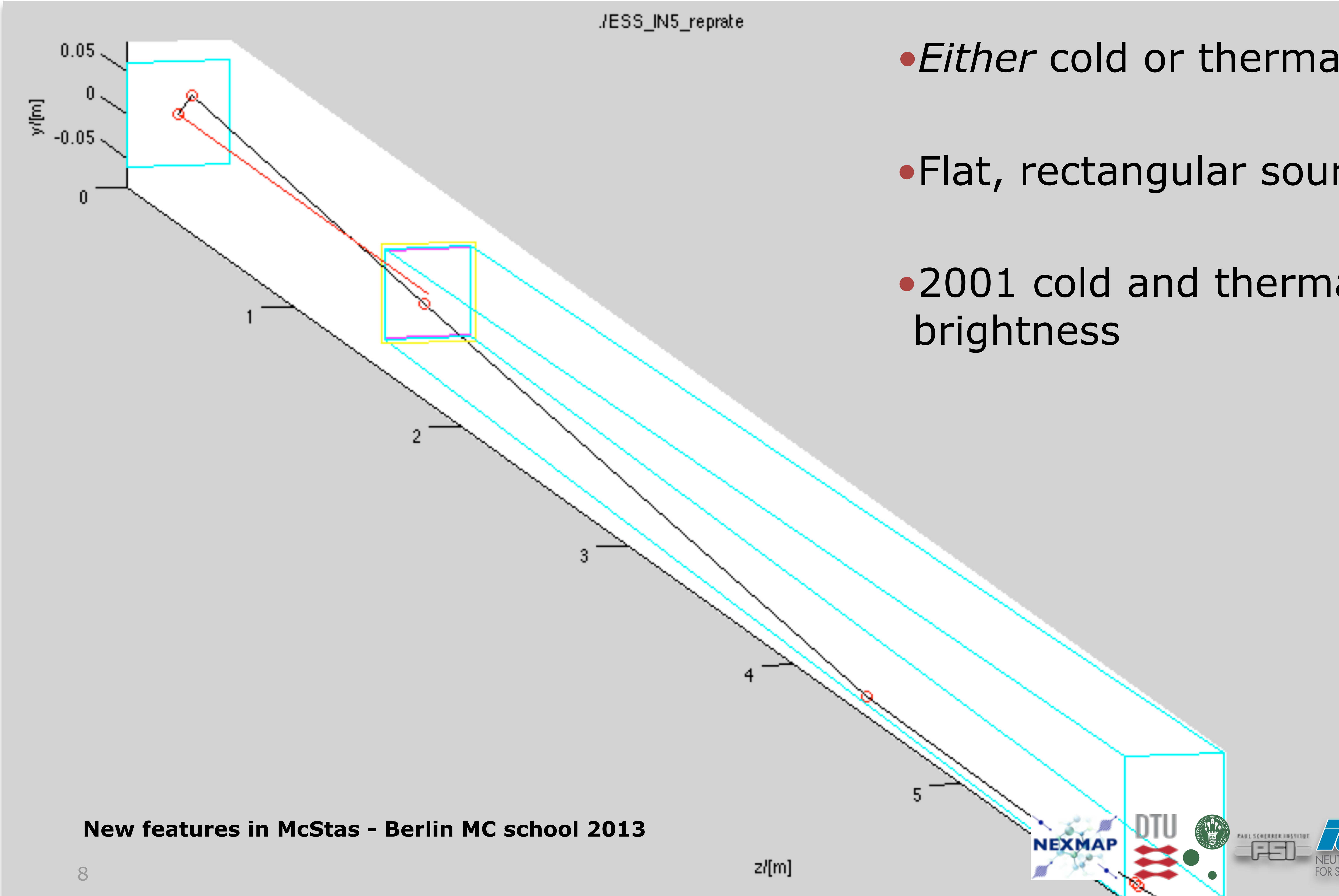


Component highlights

- - **Sample_nxs.comp** from Mirko Boin, HZB. Diffraction/imaging oriented sample where one defines the structure
- by means of the NXS crystallography library which is now also included with McStas. Comes with the test-
- instruments Test_Sample_nxs_diffraction.instr and Test_Sample_nxs_imaging.instr. NOTE: Special compilation
- required, see instrument source codes!
- - **Elliptic_guide_gravity.comp** from Henrik Carlsen, NBI. Analytical approach to describing an elliptical guide
- geometry where gravity is taken into account. Included in the Reflectometer.instr from Anette Vickery, NBI.
- - **Suite of SANS-samples** from Martin Cramer Pedersen, NBI. Various approaches to describing SANS diffraction
- e.g. using the PDB data bank. Component names are SANSShells.comp SANSPDBFast.comp SANSPDB.comp
- SANSNanodiscsWithTagsFast.comp SANSNanodiscsWithTags.comp SANSNanodiscsFast.comp SANSNanodiscs.comp
- SANSLiposomes.comp SANSEllipticCylinders.comp SANSCylinders.comp SANSCurve.comp SANSSpheres.comp. Included
- (some as comments) in the TestSANS.instr.
- - **SANSQMonitor.comp** also from Martin Cramer Pedersen, NBI. Q-monitor for SANS also included in TestSANS.instr.
- - **TOF2Q_cylPSD_monitor** from Anette Vickery, NBI. Time-of-flight vs. q monitor of cylindrical shape. Included
- in the Reflectometer.instr test instrument.
- - **SNS_source_analytic.comp** from Franz X. Gallmeier, SNS. Smooth-fit description of the SNS-moderators. Fits are
- derived from the same underlying raw-data as the ordinary SNS_source.comp is using directly with linear
- interpolation.
- - **Brilliance_monitor.comp** from Peter Willendrup, DTU Fysik. Monitor for easy evaluation of mean and instantaneous
- source brilliance for source comparison. Used in the ESS_brilliance.instr test instrument.
- - **TOF_PSD_monitor_rad.comp** from Kim Lefmann, KU. Derived from PSD_monitor_rad by Henrich Frielinghaus, FZJ.
- Position-sensitive TOF monitor with radially averaging.
- - **PSD_TOF_monitor.comp** from Peter Willendrup, DTU Fysik. PSD-monitor with a number of independent time-slices.
- Derived from PSD_monitor by Kim Lefmann, KU.
- - **Source_gen4.comp** from Jonas O Birk, NBI and Uwe Filges PSI. Version of source_gen with PSI-specific changes,
- e.g. a high-energy tail contribution. Included in the RITA-II.instr from Linda Udby, NBI.
- - **Absorber.comp** from Peter Willendrup, DTU Fysik. Slab of perfectly absorbing material. Included in the RITA-II.instr
- from Linda Udby, NBI.
- - **PSD_monitor_psf.comp** and PSD_monitor_psf_eff.comp from Kim Lefmann and Linda Udby, KU. Two derivatives of
- PSD_monitor.comp both with gaussian point-spread-function and _eff with a 1/k efficiency parameter. Included
- in the RITA-II.instr from Linda Udby, NBI.
- - **Virtual_mcnp_ss_input.comp** and Virtual_mcnp_ss_output.comp from Esben Klinkby, DTU Nutech. Allows to read and write
- MCNP/MCNPX 'source surfaces'. For use in simulations where neutrons need transport in both codes.
- - **Virtual_mcnp_ss_Guide.comp** from Esben Klinkby, DTU Nutech. Single guide piece sitting in a 'sandwich' between
- an input and an output MCNP/X source surface.
- - **ESS_moderator_long.comp** patches from Kim Lefmann KU (multiple-pulses, TOF-focusing) and Esben Klinkby DTU Nutech
- (geometry and spectrum from ESS MCNPX models). Thanks to Klaus Lieutenant from Vitess/HZB for providing adjusted
- parameters for the 'Mezei moderator' and a wavelength-dependent corretion term.
- - Many other components received updates...

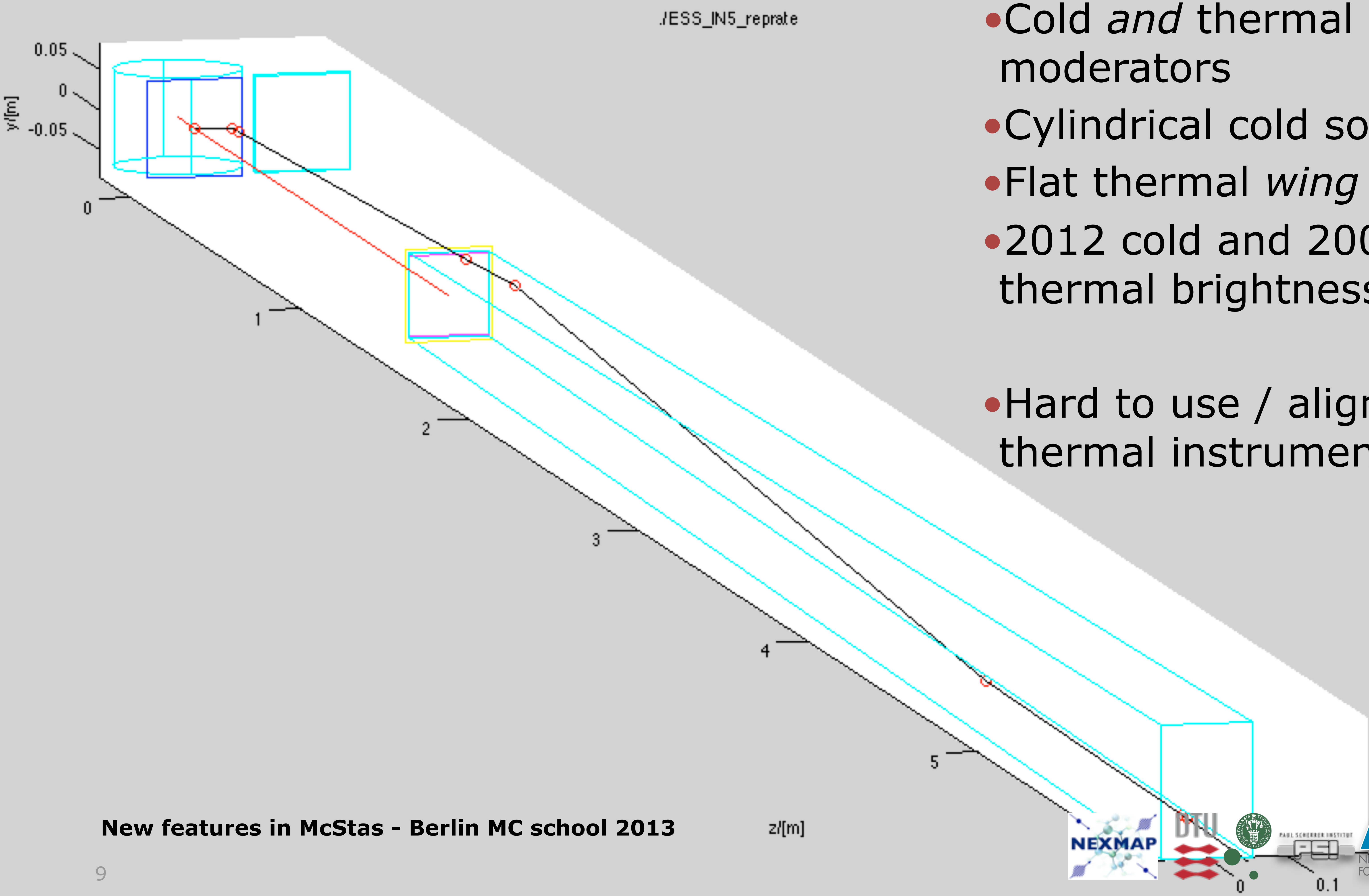


The 1.12c ESS source (released)



- *Either* cold or thermal
- Flat, rectangular source
- 2001 cold and thermal brightness

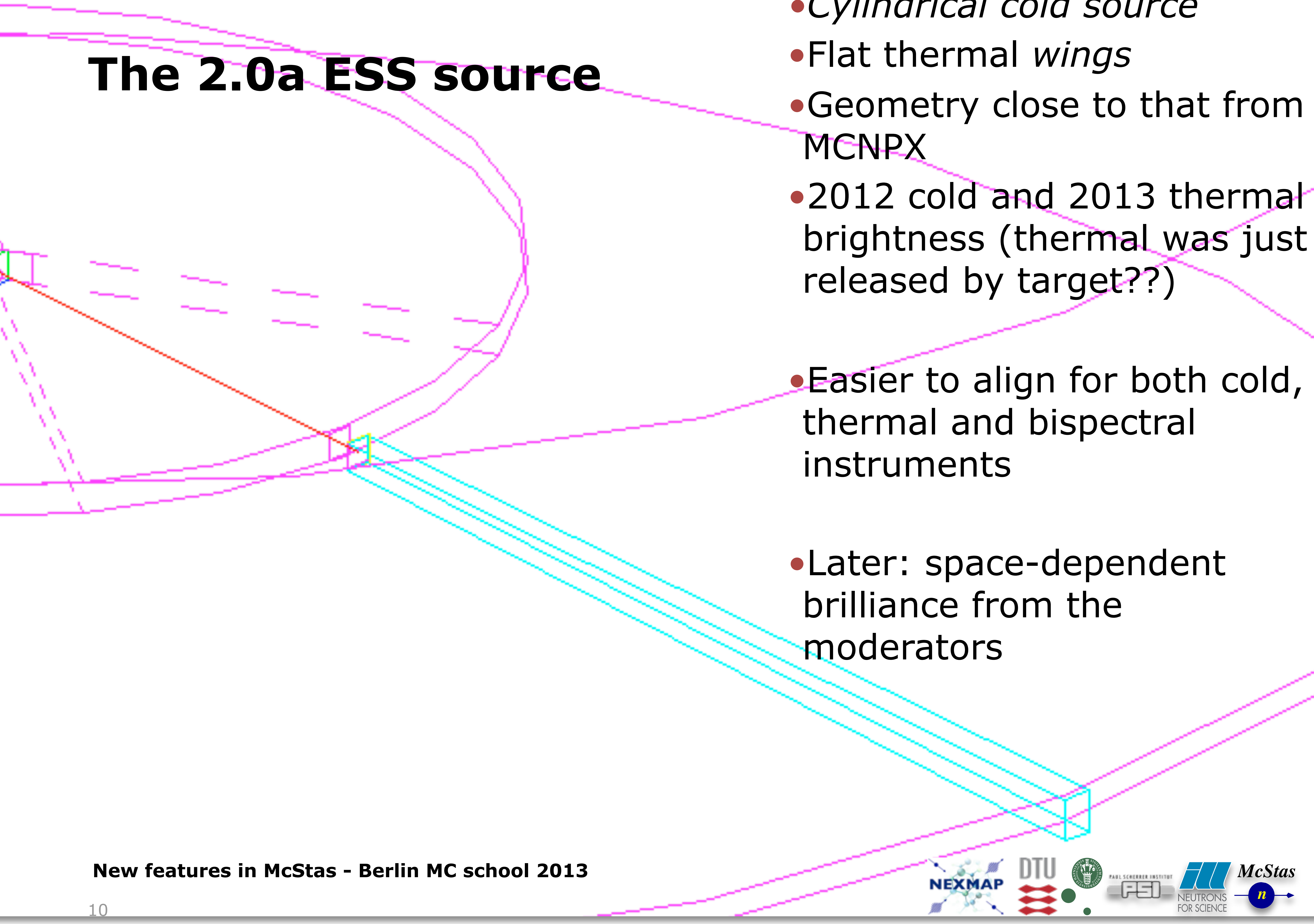
The 2.0 ESS source (released)



- Cold *and* thermal moderators
- Cylindrical cold source
- Flat thermal *wing*
- 2012 cold and 2001 thermal brightness

- Hard to use / align for thermal instruments

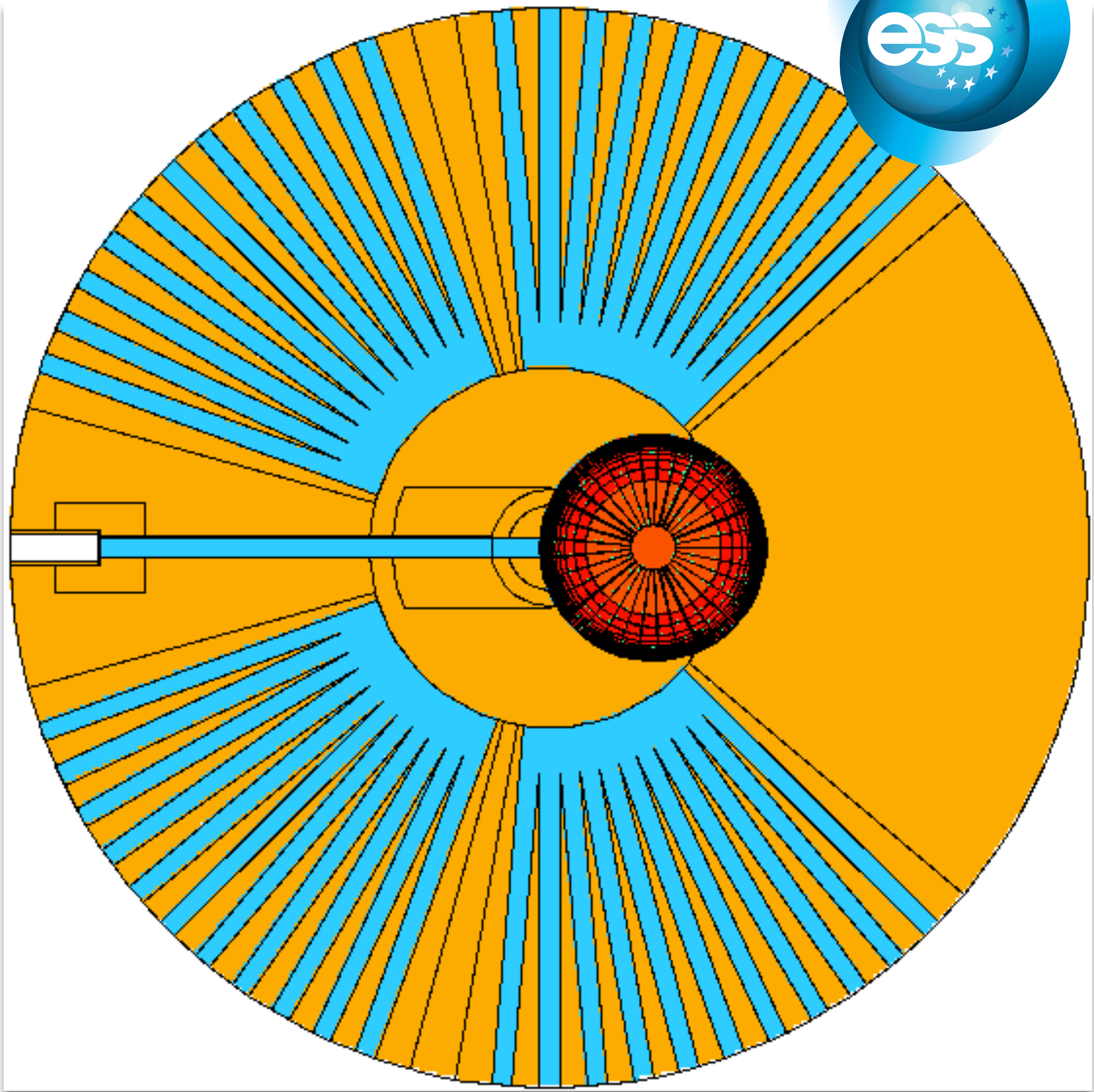
The 2.0a ESS source

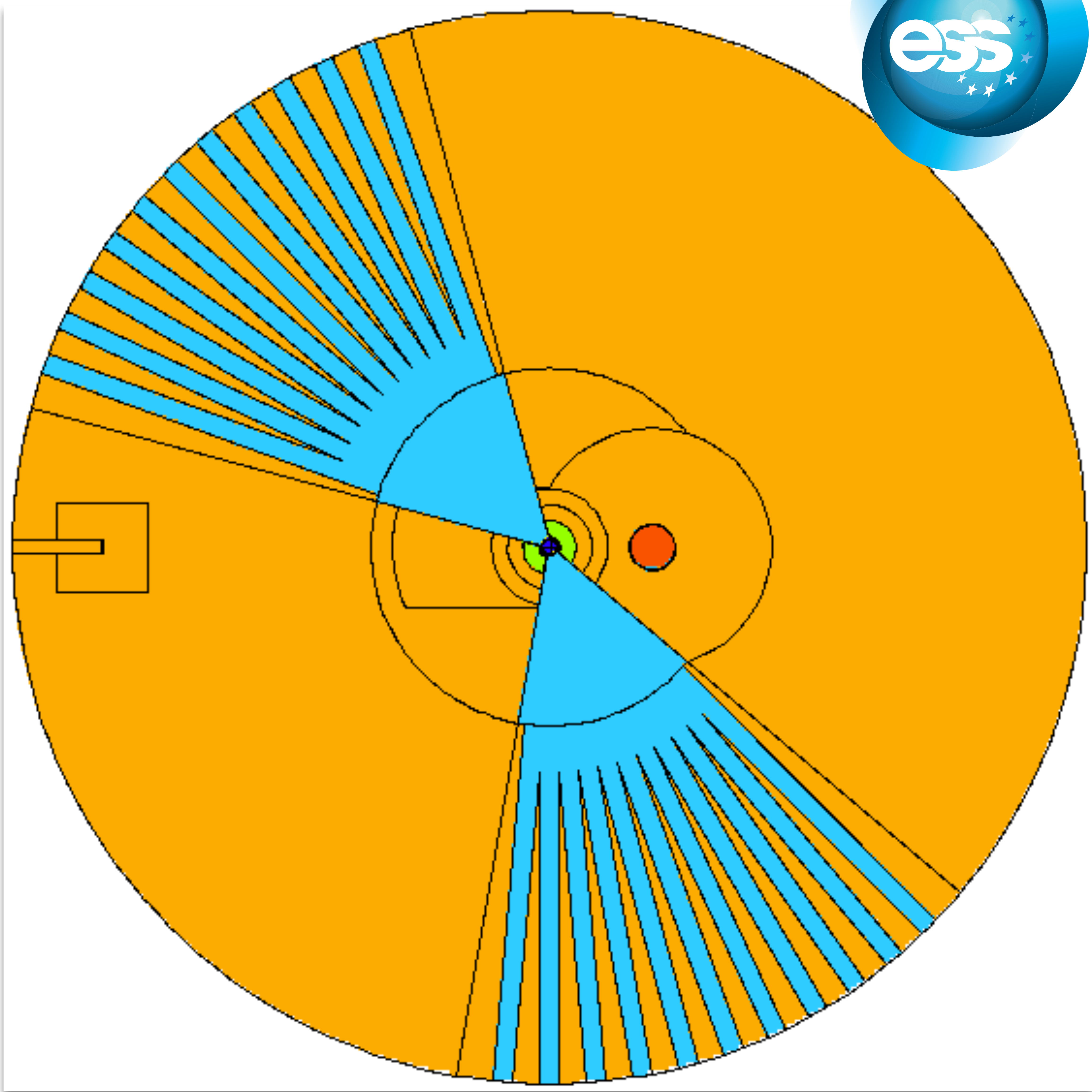


- Cold *and* thermal moderators
- *Cylindrical cold source*
- Flat thermal *wings*
- Geometry close to that from MCNPX
- 2012 cold and 2013 thermal brightness (thermal was just released by target??)

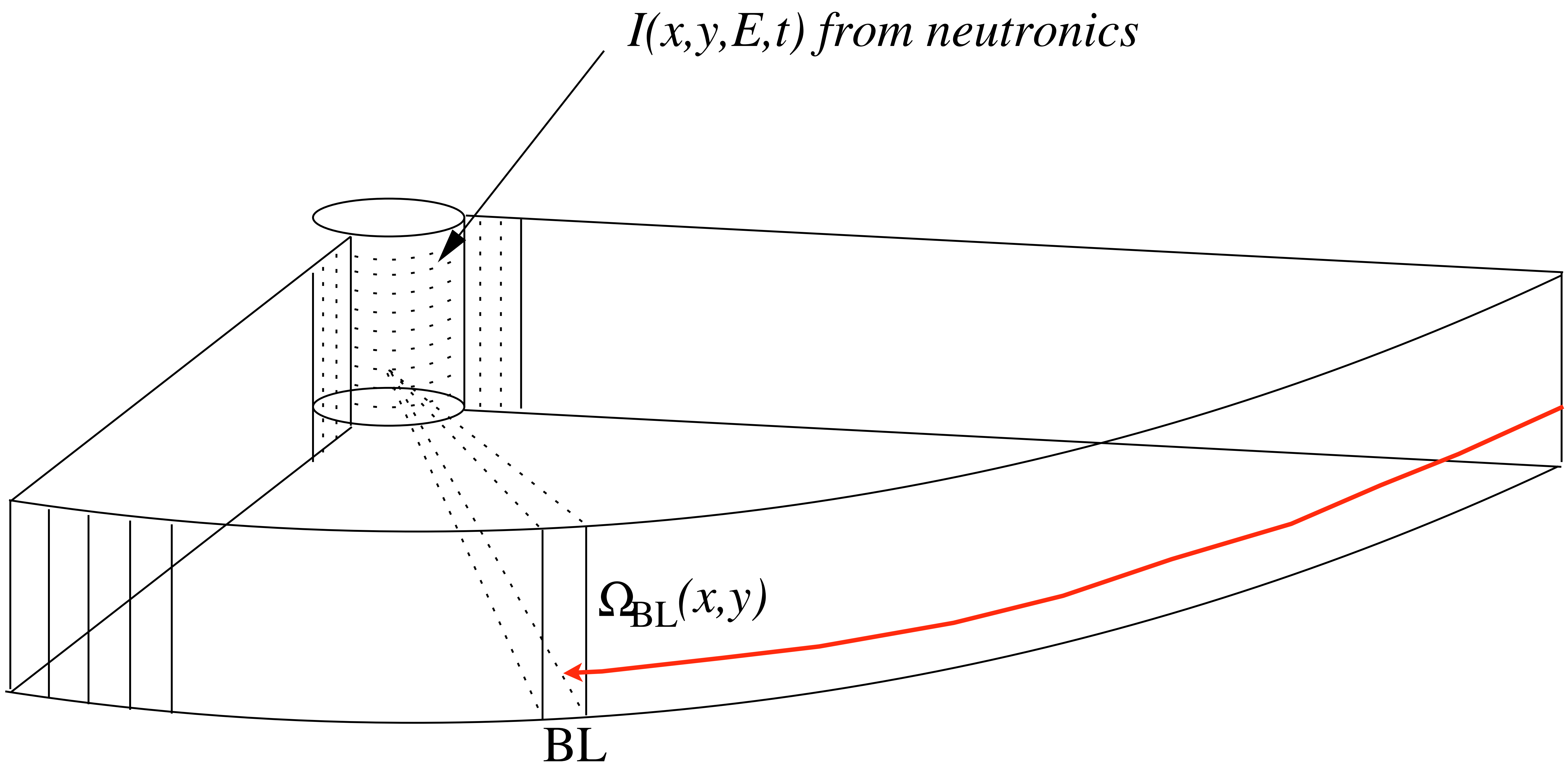
- Easier to align for both cold, thermal and bispectral instruments

- Later: space-dependent brilliance from the moderators





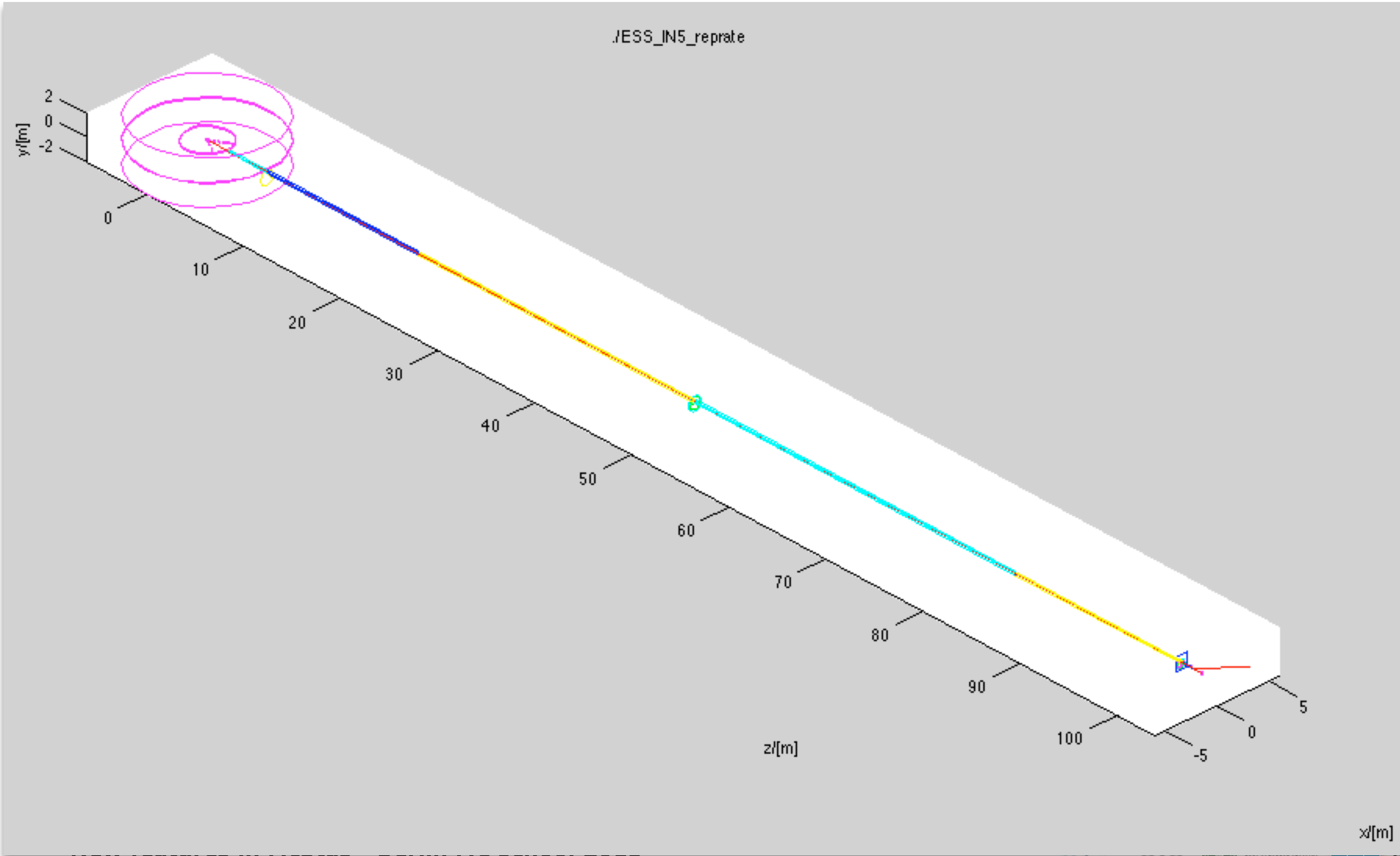
The 2.0a source (in the pipe)



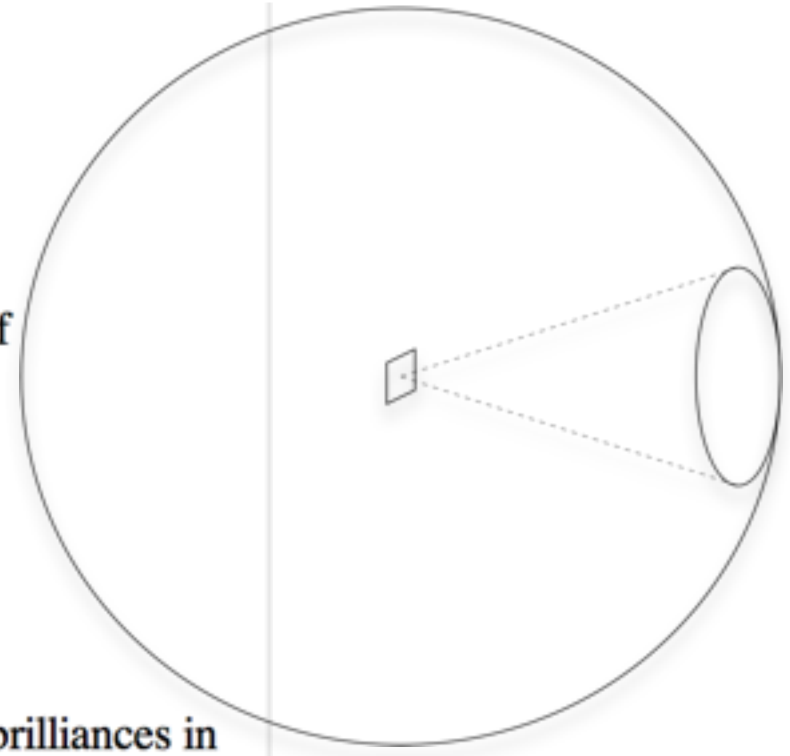
Per beamline:

$$I_{BL}(x,y,E,t) = \frac{\Omega_{BL}(x,y)}{4\pi} I(x,y,E,t)$$

Overview of full monolith, including in-pile optics, shielding etc.



$\mathfrak{B}\ [n/cm^2/s/ster/\text{Å}]$



[Identification | Description | Input parameters | Output parameters | Links]

The `Brilliance_monitor` Component

Special "Brilliance" monitor of FIXED size 1x1cm. If used in the right setting, will output "instantaneous" and "mean" brilliances in units of Neutrons/cm^2/ster/AA/s. Conditions for proper units:

- Use a with a source of area 1x1cm
- The source must illuminate/focus to an area of 1x1cm a 1m distance
- Parametrise the Brilliance_monitor with the frequency of the source
- To not change the source TOF distribution, place the Brilliance monitor close to the source!

with a source of area 1x1cm illuminating/focusing to an area of 1x1cm a 1m distance, this monitor will output "instantaneous" and "mean" brilliances in units of Neutrons/cm^2/ster/AA/s

Identification

- **Author:** Peter Willendrup, derived from TOF_lambda_monitor.comp
- **Origin:** DTU Physics
- **Date:** May 23, 2012
- **Version:** 1.1

Description

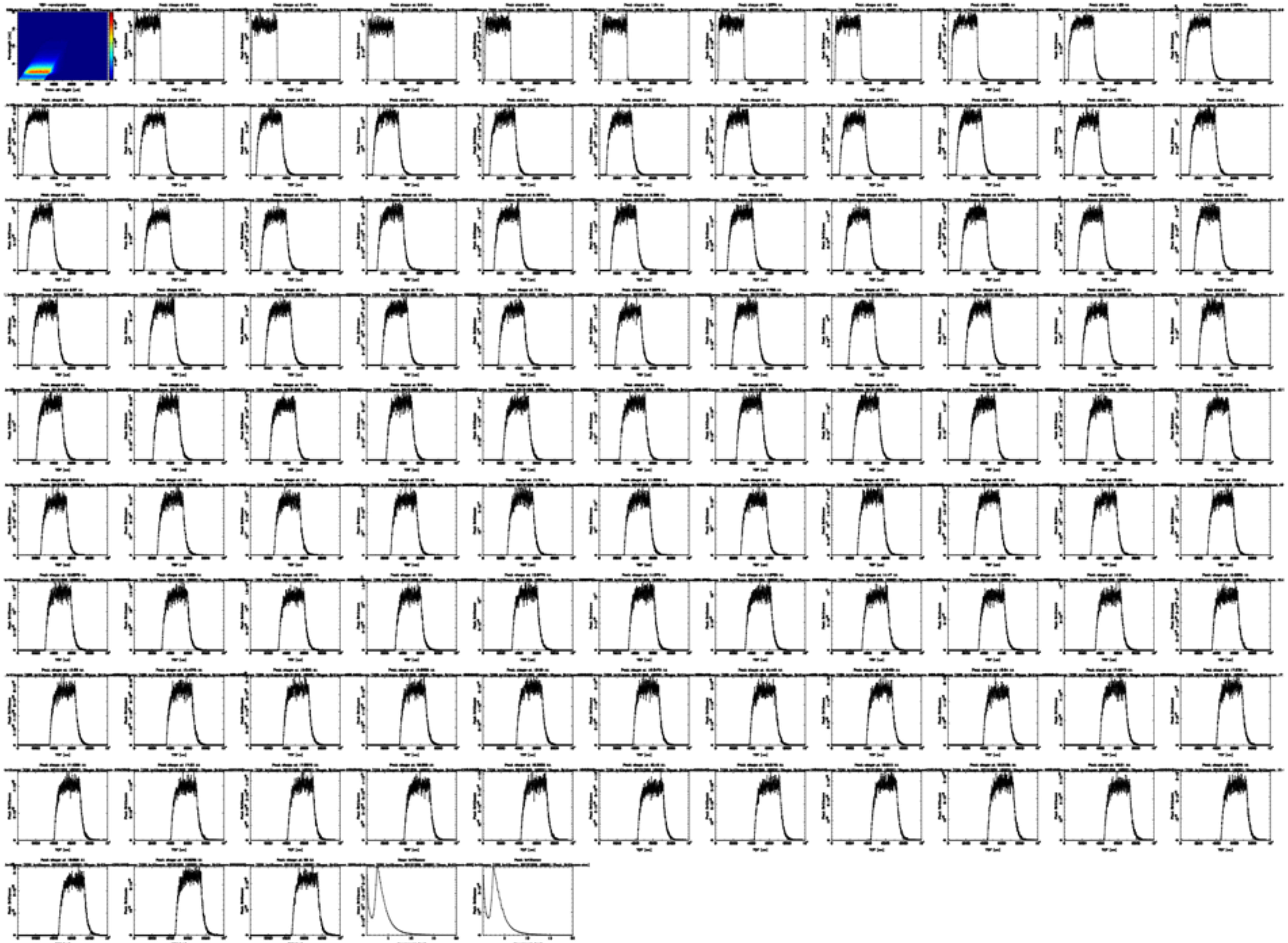
```
Here is an example of the use of the component. Note how the mentioned Unit conditions are implemented in instrument code.

COMPONENT Source = ESS_moderator_long(
   l_low = lambdamin, l_high = lambdamax, dist = 1, xw = 0.01, yh = 0.01,
   freq = 14, T=50, tau=287e-6, tau1=0, tau2=20e-6,
   n=20, n2=5, d=0.00286, chi2=0.9, I0=6.9e11, I2=27.6e10,
   branch1=0, branch2=0.5, twopulses=0, size=0.01)
 AT (0, 0, 0) RELATIVE Origin

COMPONENT BRIL = Brilliance_monitor(nlam=196,nt=401,filename="bril.sim",
        t_0=0,t_1=4000,lambda_0=lambdamin,
        lambda_1=lambdamax, Freq=14)
AT (0,0,0.000001) RELATIVE Source
```

Input parameters

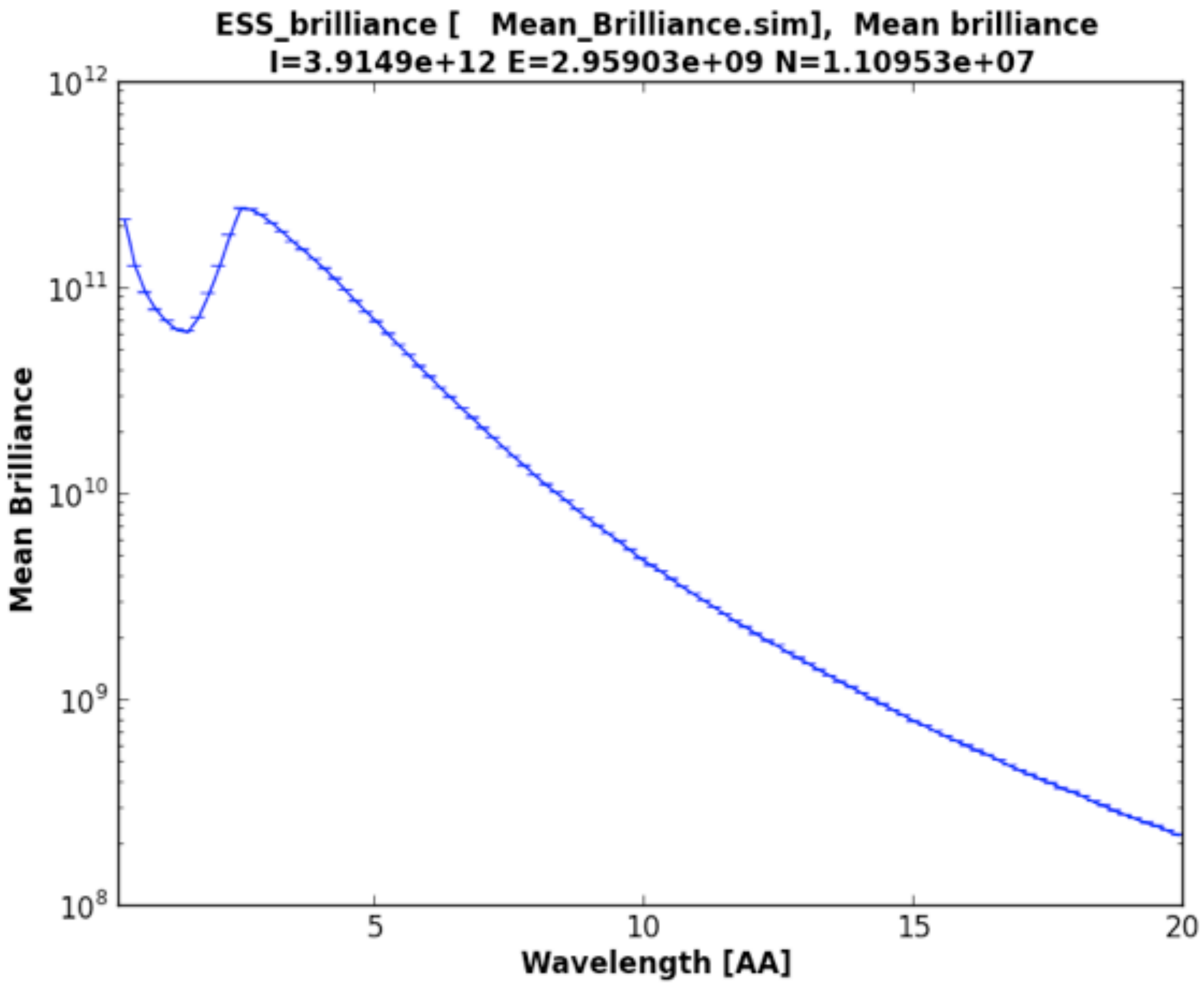
Parameters in **boldface** are required; the others are optional.

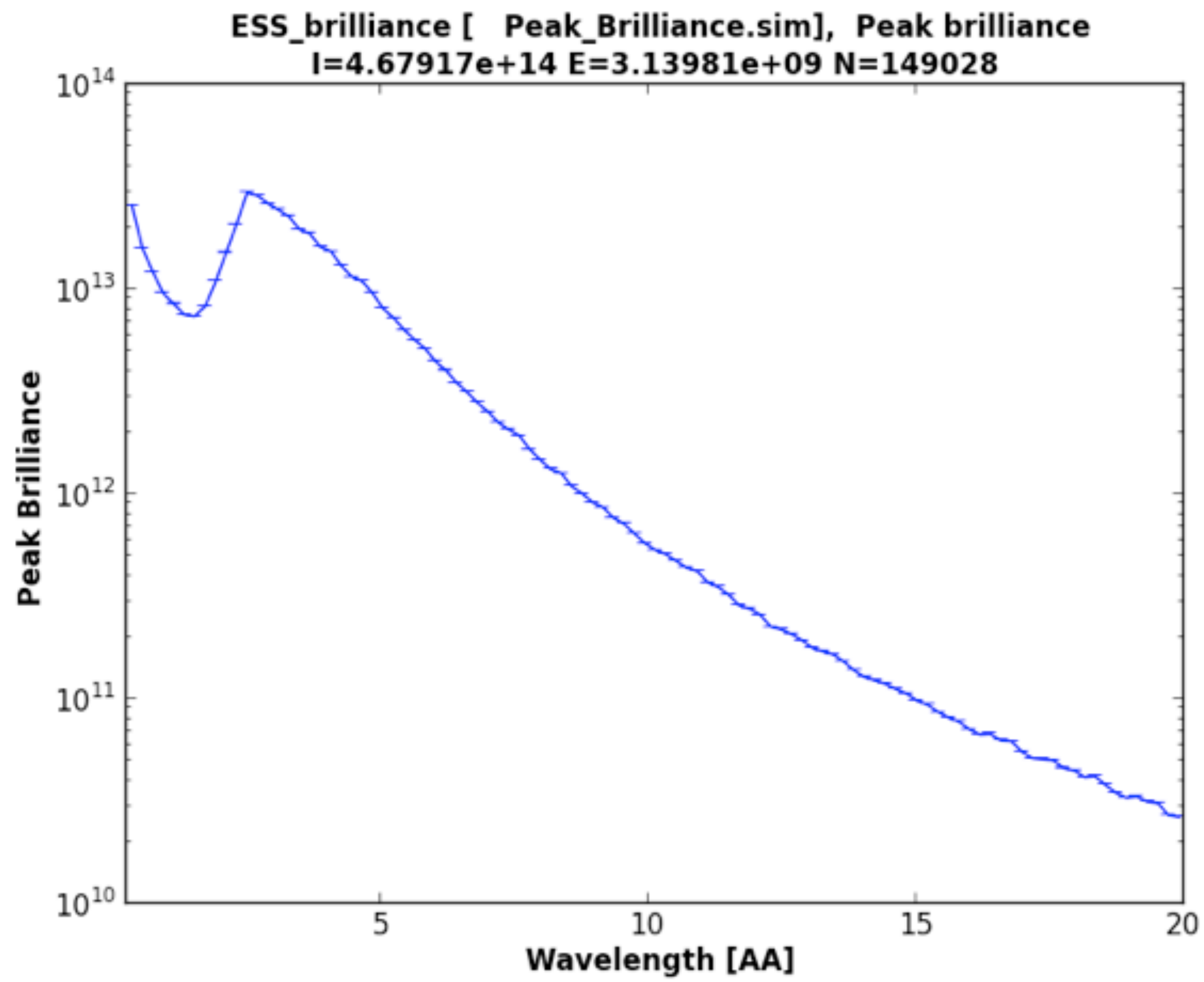
Name	Unit	Description	Default
nlam	1	Number of bins in wavelength	101
nt	1	Number of bins in TOF	1001
filename	string	Defines filenames for the detector images. Stored as: Peak_<filename> and Mean_<filename>	
t_0	us	Minimum time	0
t_1	us	Maximum time	20000
lambda_0	AA	Minimum wavelength detected	0
lambda_1	AA	Maximum wavelength detected	20
restore_neutron	1	If set, the monitor does not influence the neutron state	0
Freq	Hz	Source frequency. Use freq=1 for reactor source	

Brilliance_monitor output



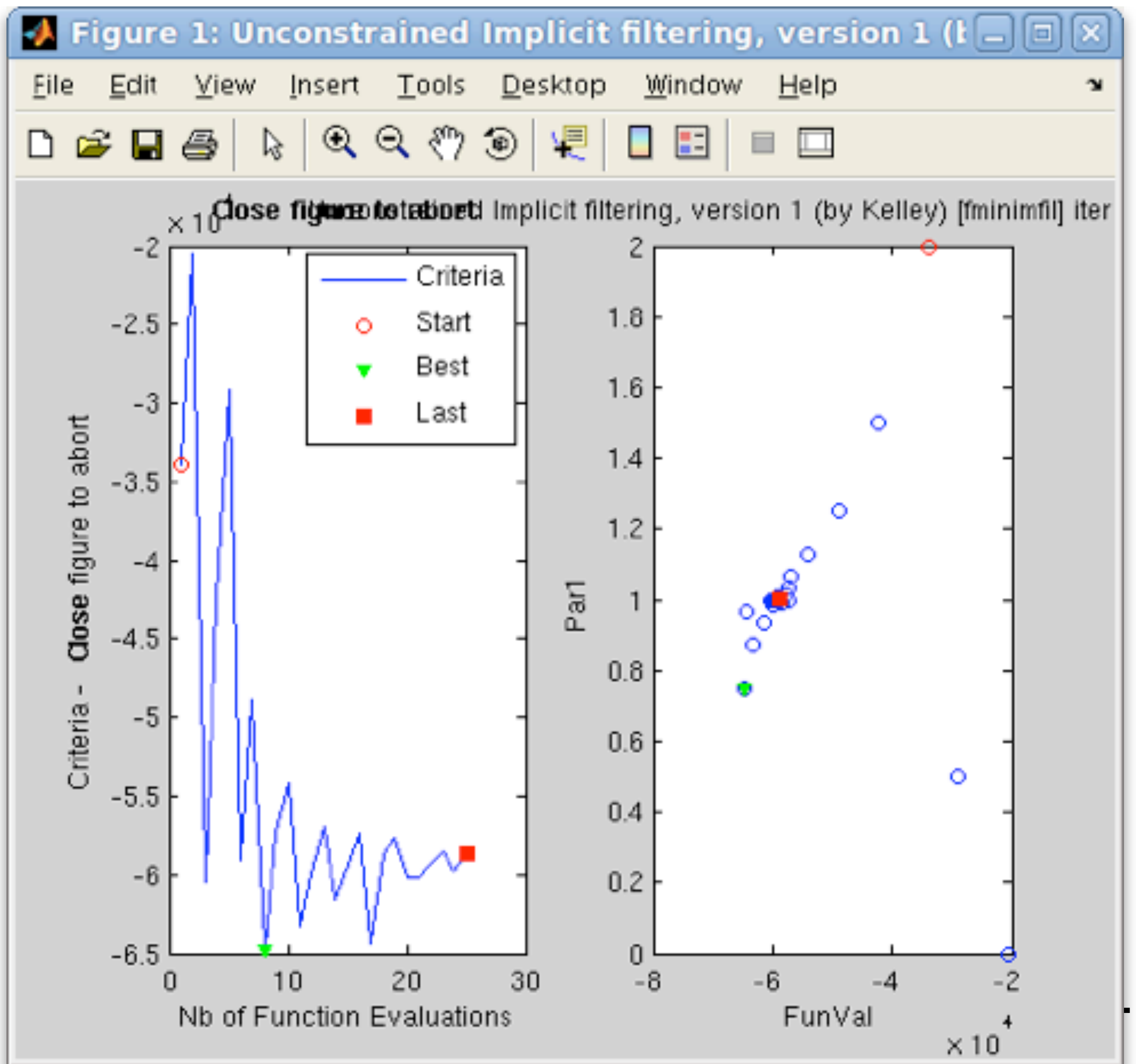
Easy way to get brilliance curves

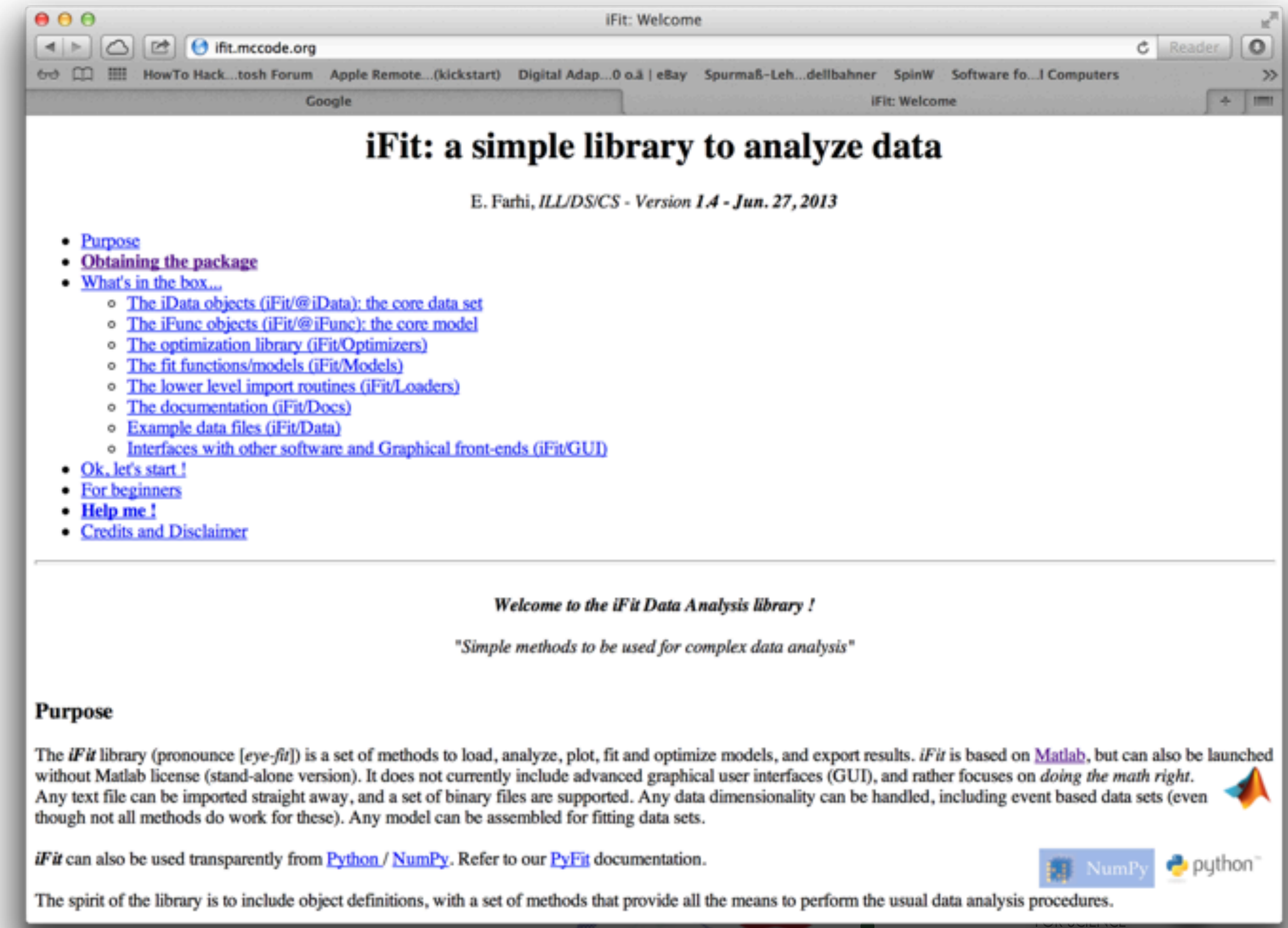




Interface to iFit

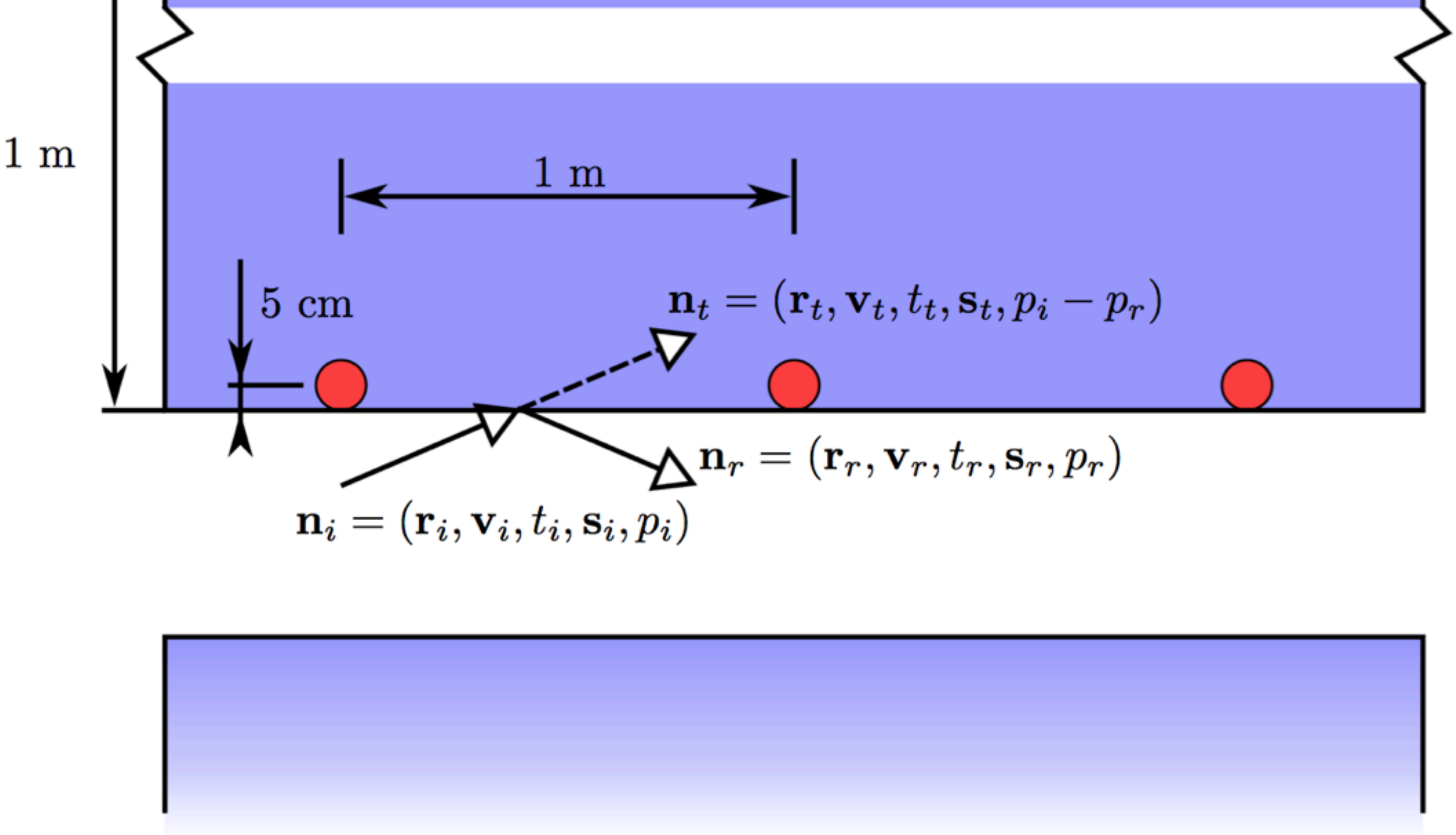
- Interface-code for the iFit data analysis package http://ifit.mccode.org
- Visualisation and analysis of simulation data and instrument (like Matlab but no need for the license)
- Suite of optimizers (see tomorrows exercise on that topic)

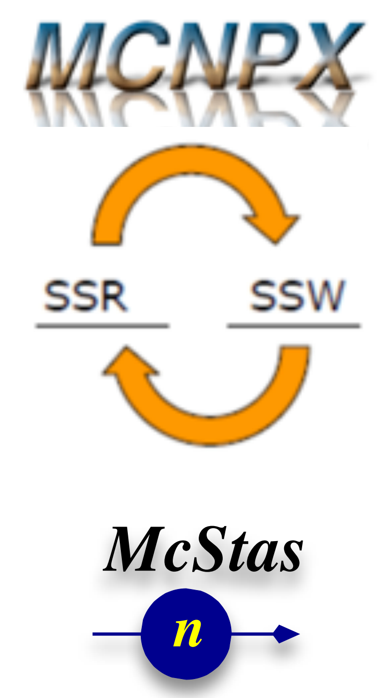


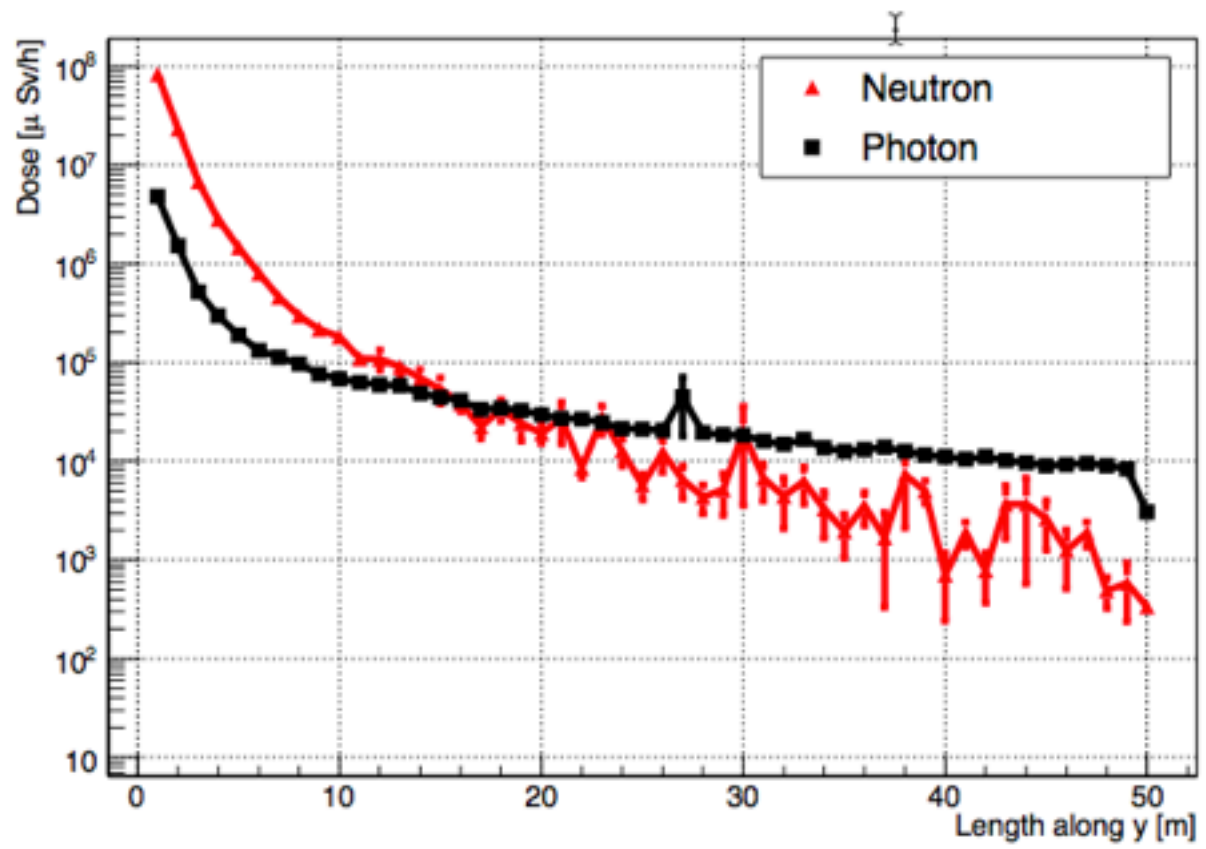


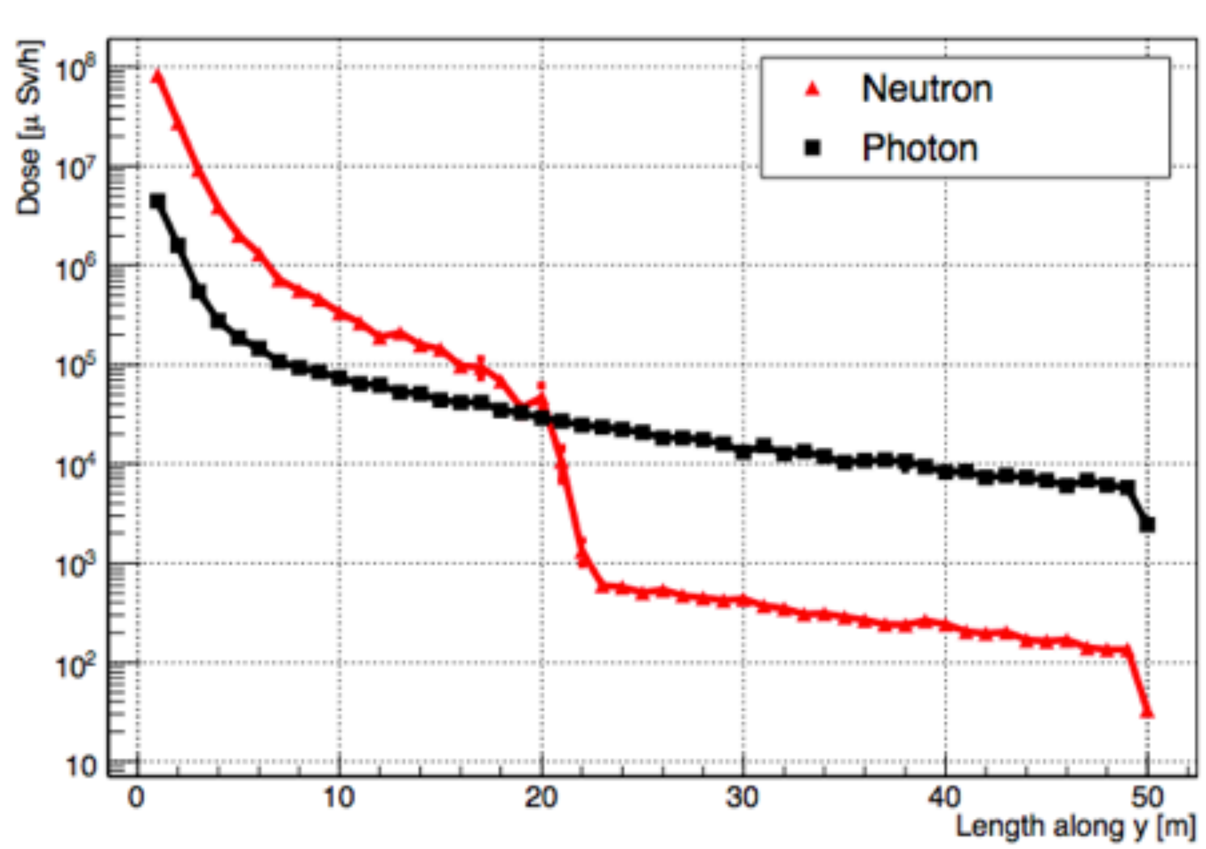
.3

McStas-MCNPX interfaces for beam losses









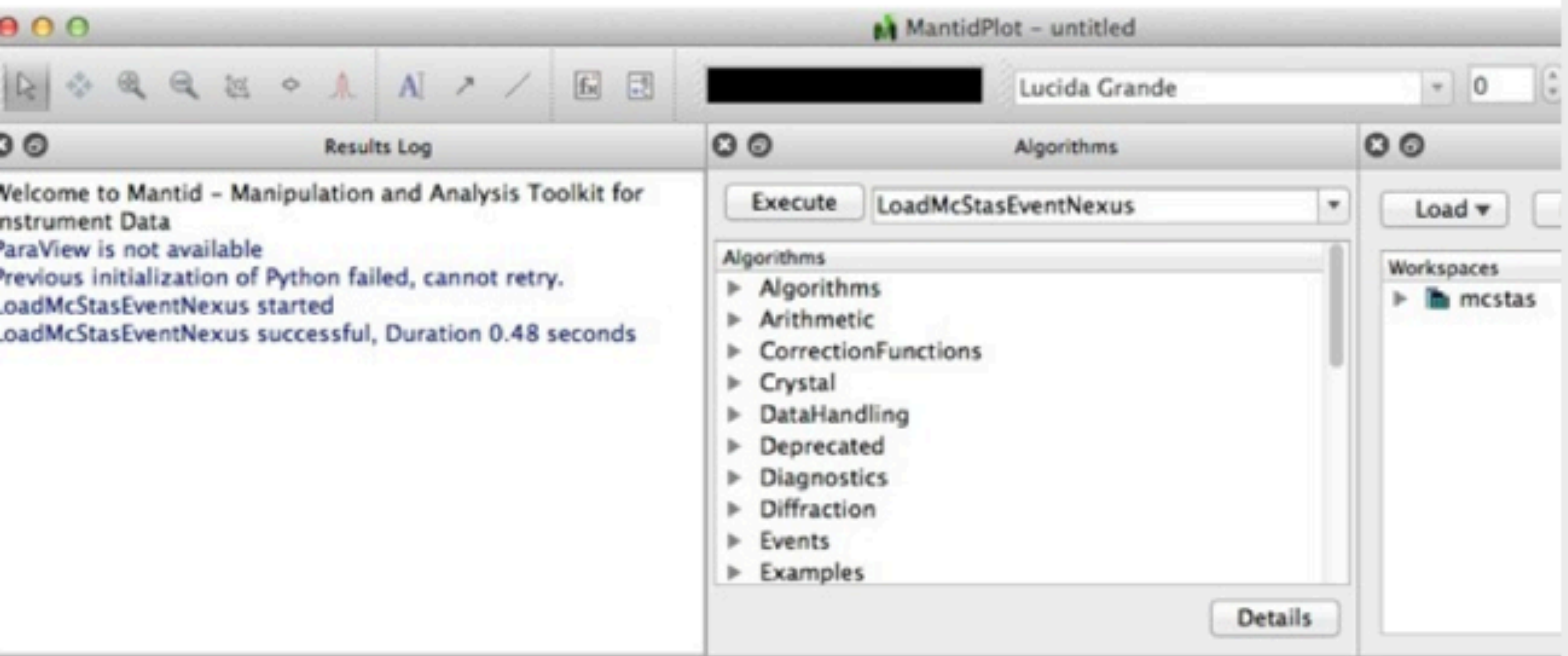
Collaboration
DTU Nutech
DTU Fysik
PSI
ESS

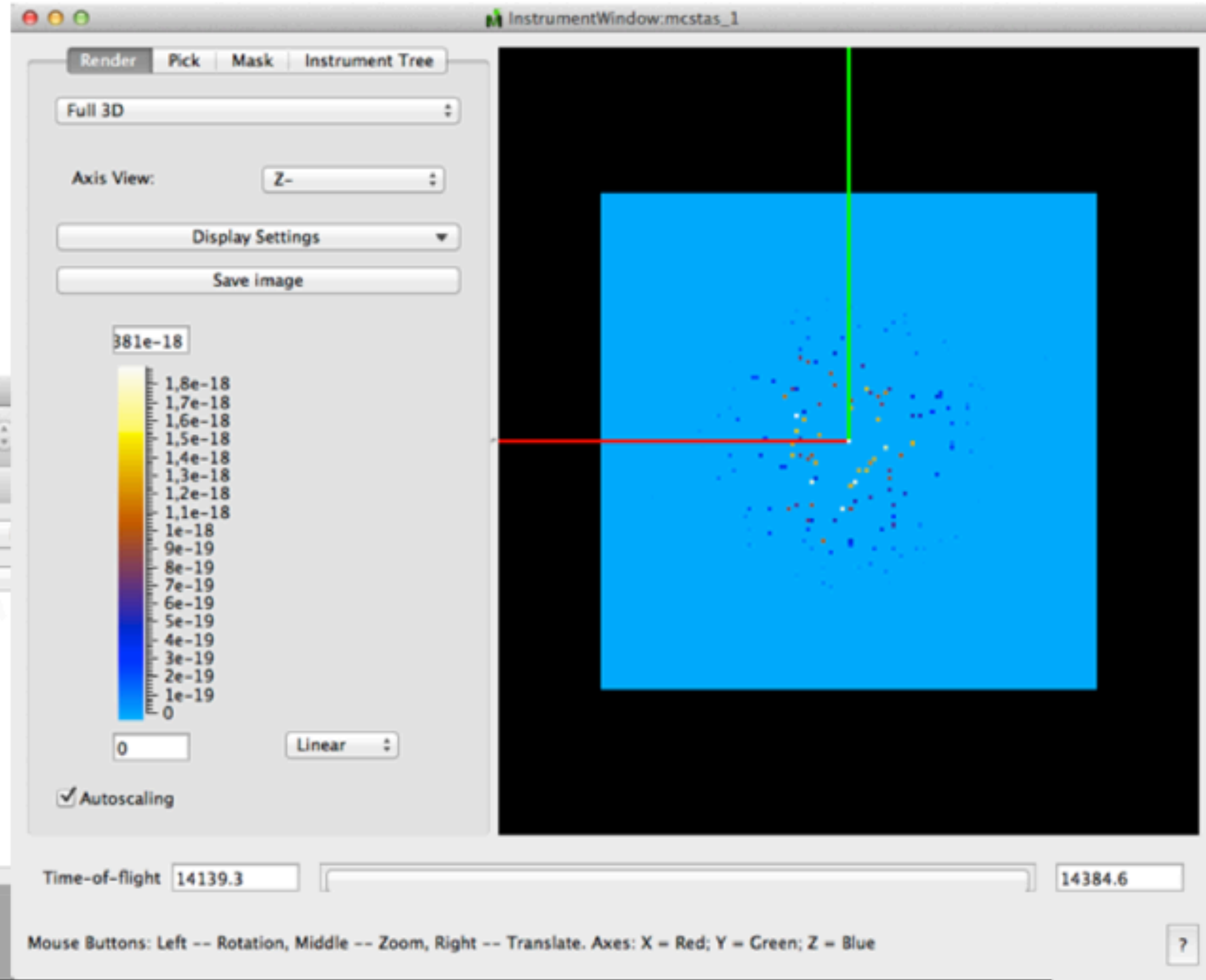
Coollaboration

KU/ESS-DMSC, DTU Fysik, Mantid

McStas + Mantid integration

- COMPONENT nD_Mantid_1 = Monitor_nD(
- options ="mantid square x limits=[-0.2 0.2] bins=128 y limits=[-0.2 0.2] bins=128, neutron pixel t, list all neutrons",
- xmin = -0.2,
- xmax = 0.2,
- ymin = -0.2,
- ymax = 0.2,
- restore_neutron = 1,
- filename = "bank01_events.dat")
- AT (0, 0, 3.2) RELATIVE sample

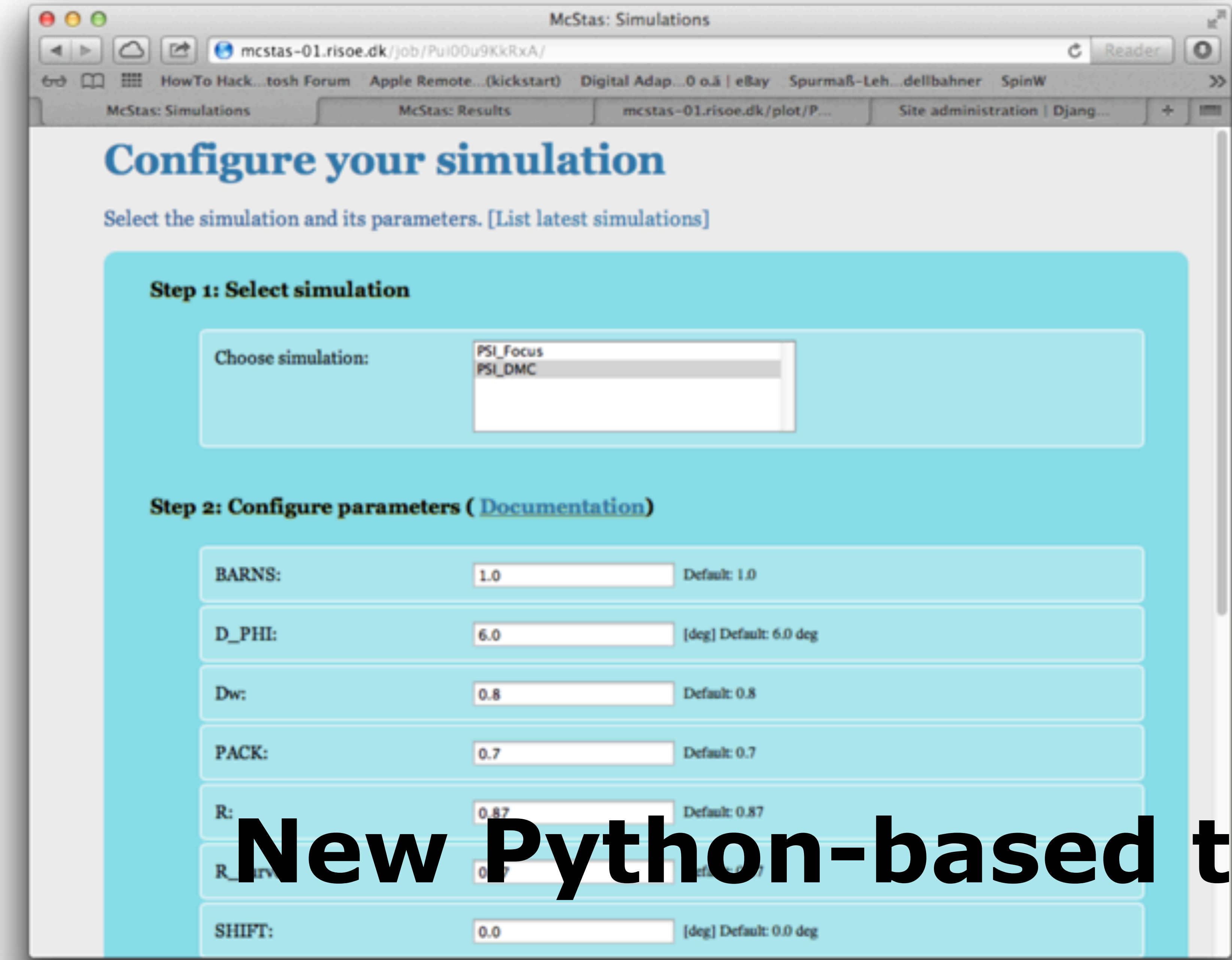


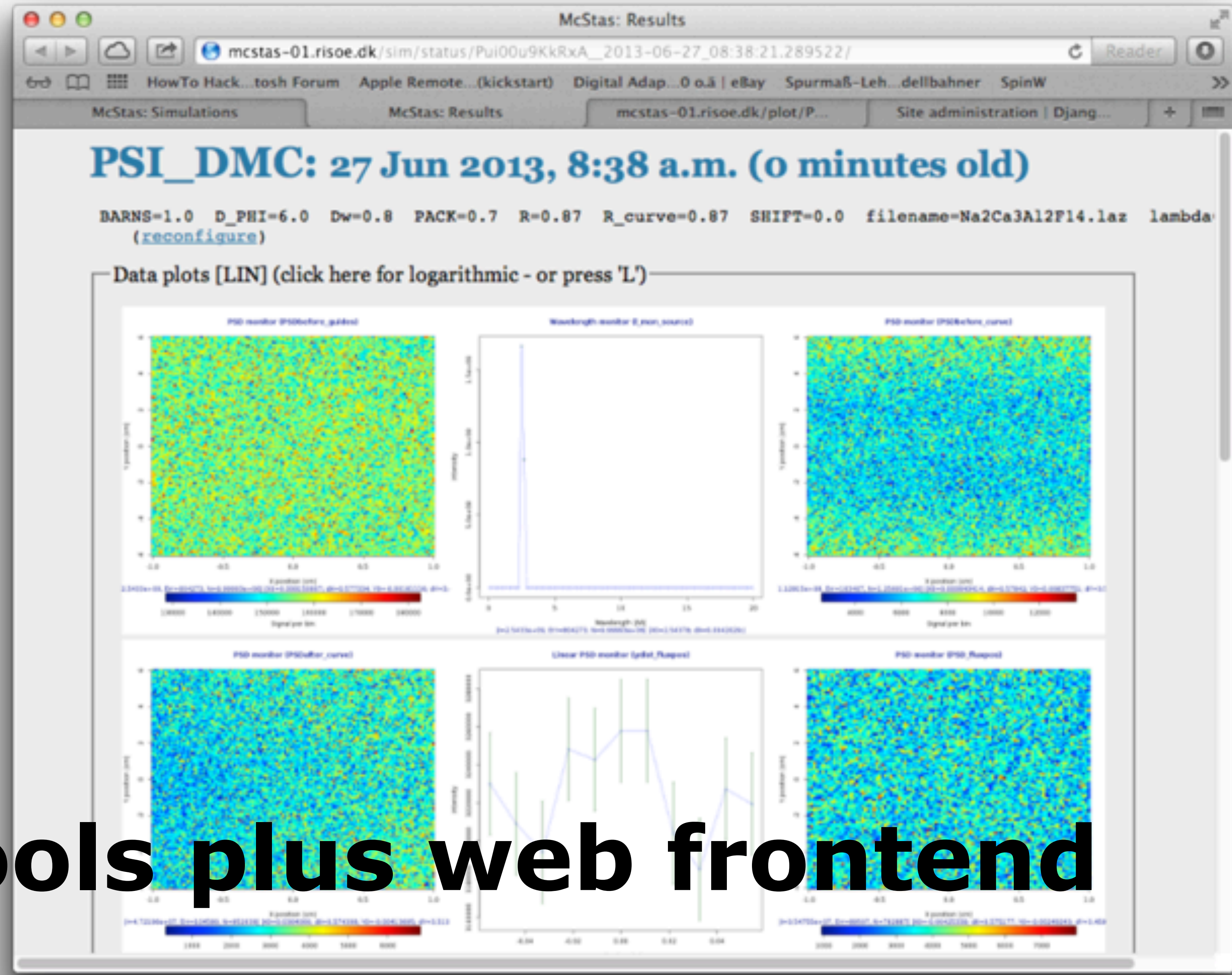


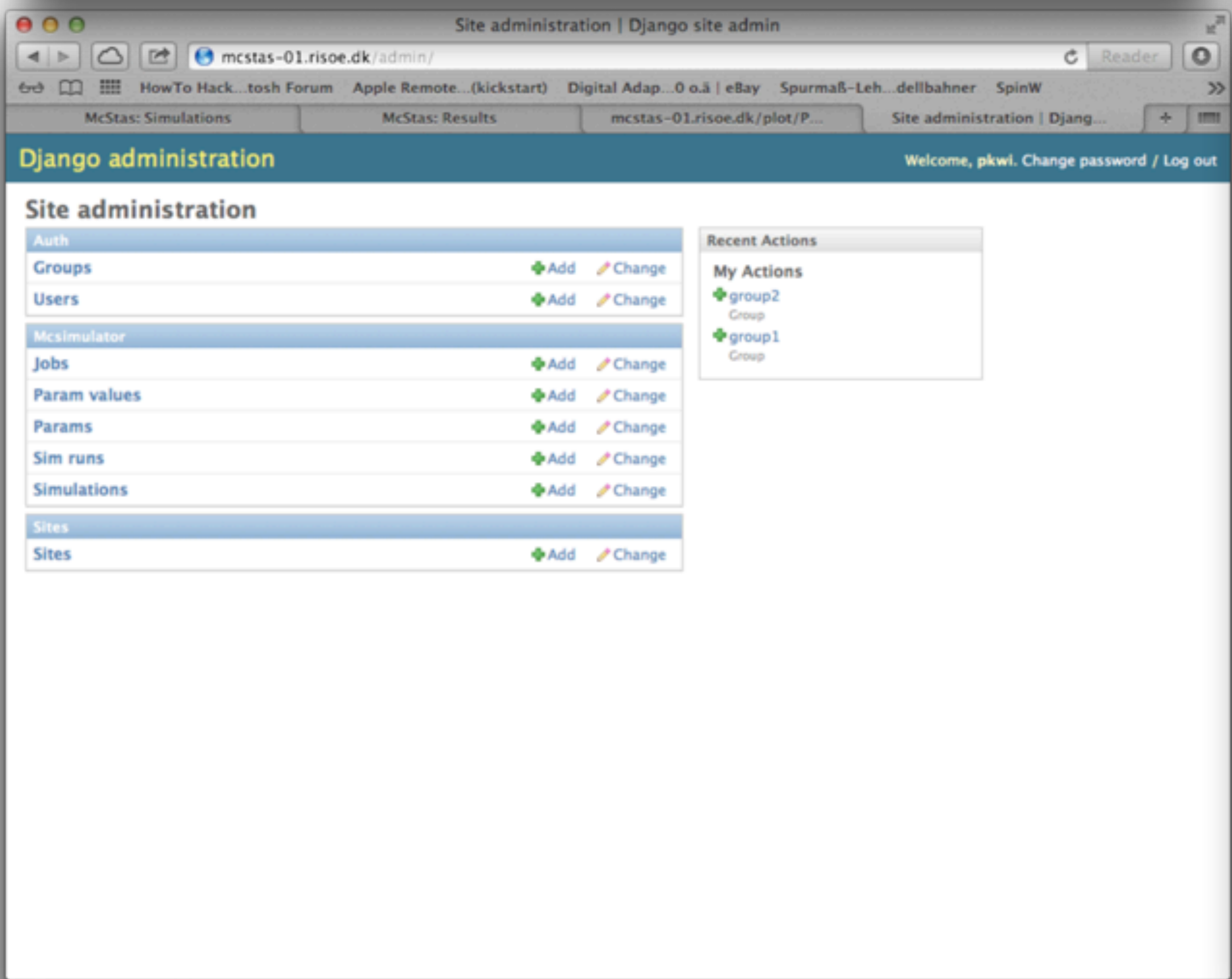
Proof of concept solution: One detector geometry so far, yet general:

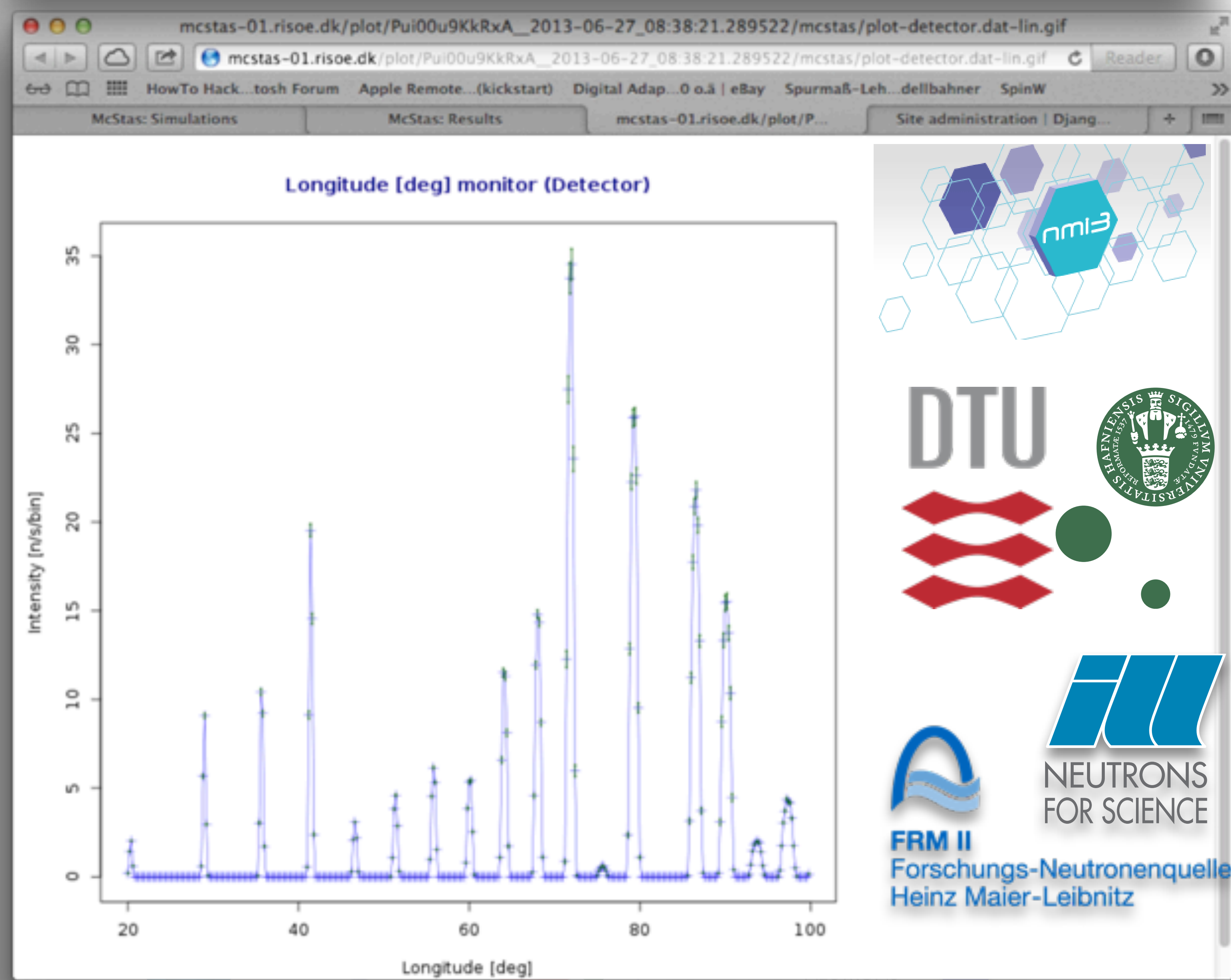
Allows to define an IDF XML instrument description using a standard McStas. NeXus needs to be installed for producing NeXus output.

New Python-based tools plus web frontend







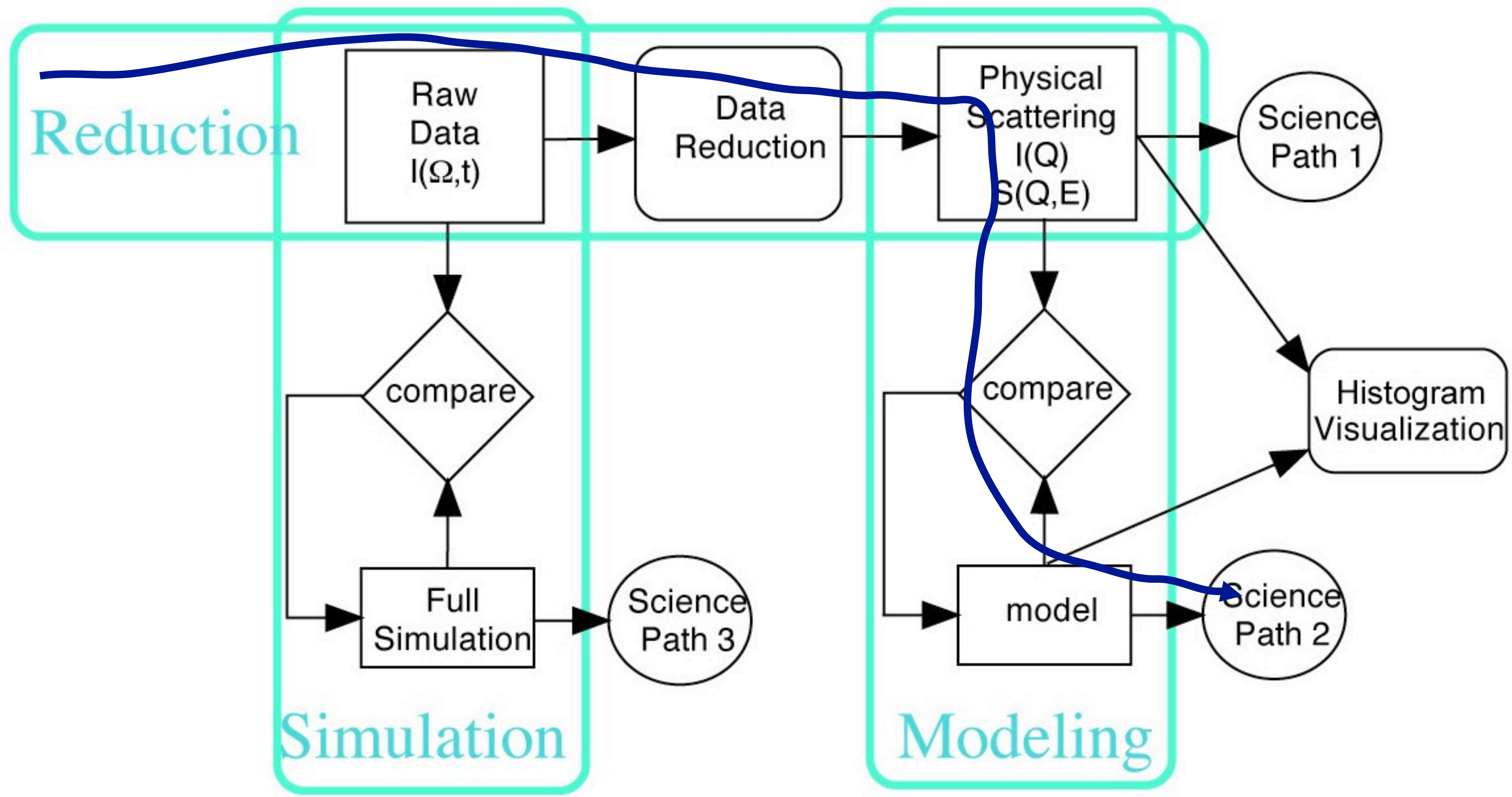


New Rietveld refinement methodology using McStas virtual experiment models

E. Farhi and J. Rodriguez-Carvajal, ILL

Science path: conventional

Raw data

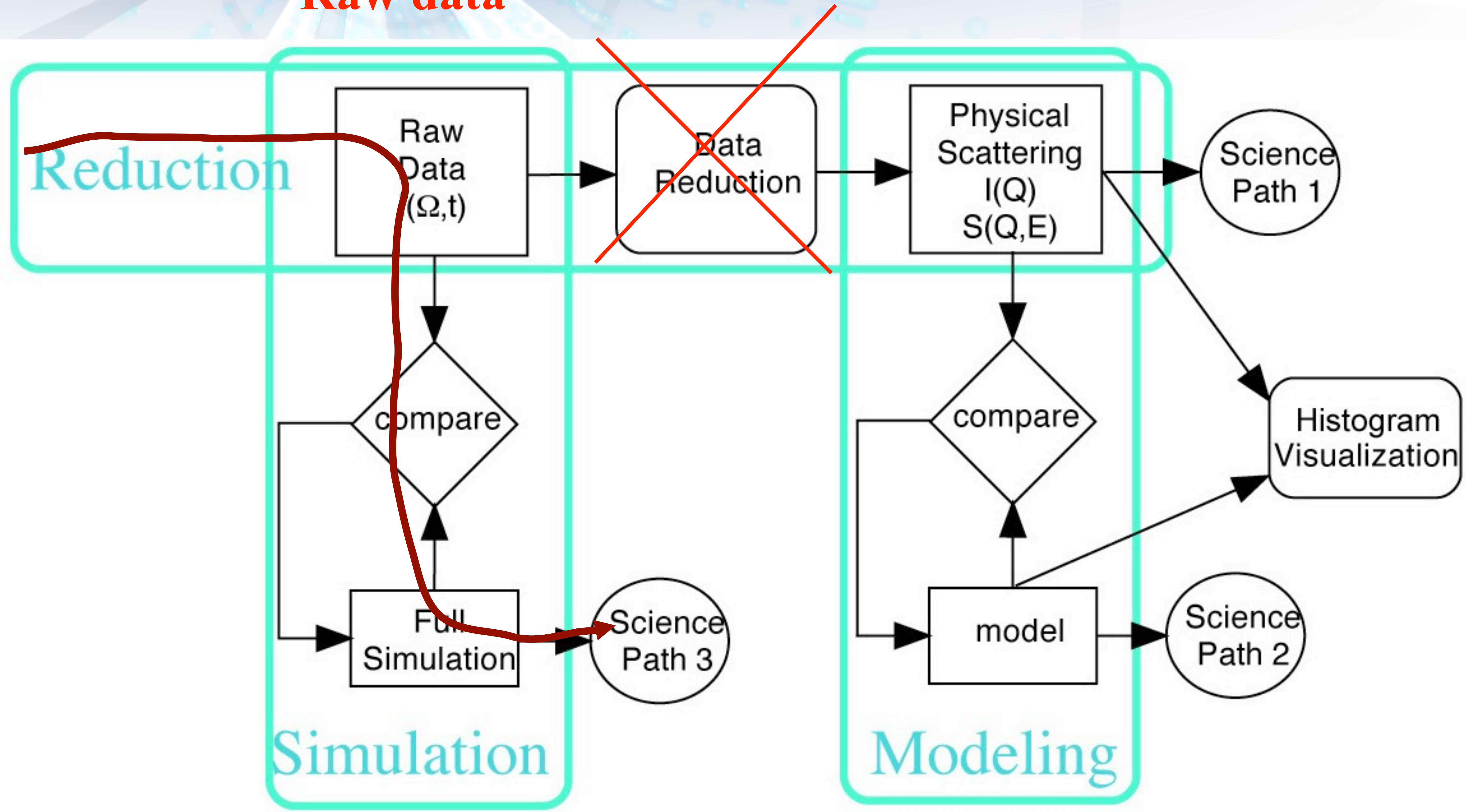


Conventional Rietveld

Picture from B. Fultz, DANSE

Science path: direct

Raw data

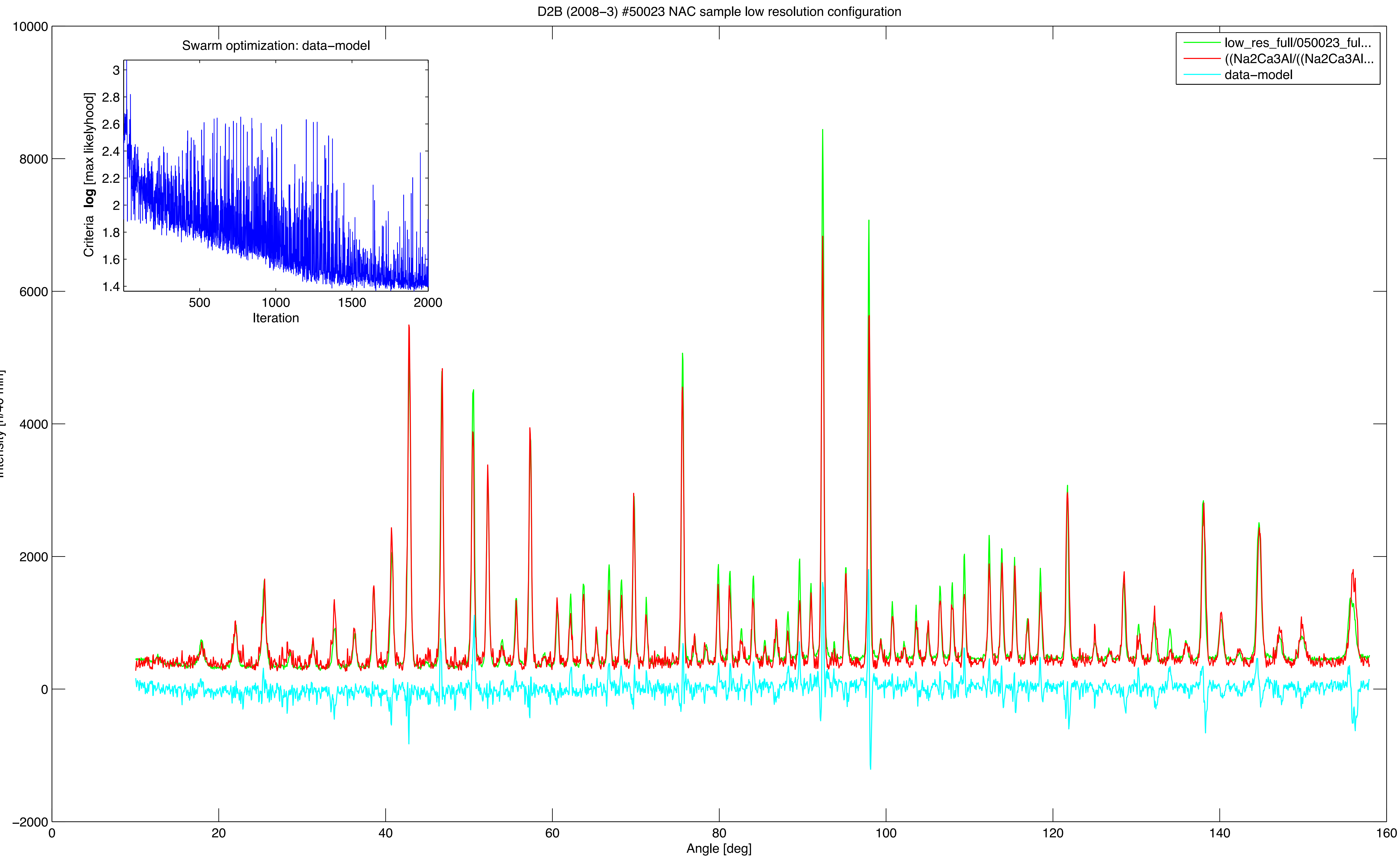


McStas with CrysFML calls
Optimisation with iFit

Picture from B. Fultz, DANSE

Implementation overview

- We set-up a diffractometer model from its geometry. No 'effective' parameters. The model should contain a realistic detector geometry (incl. ^{3}He gas).
- We define the sample as part of the diffractometer description, with structural parameters as input (CIF/CFL/ShelX).
- We get some measured data.
- We refine the structural parameters on the measurement, using an *ad-hoc* optimizer.



That's not perfect, but a reasonable fit. Much slower than traditional Rietveld, but no a-priori assumptions about peak shape or resolution.

People

- The success of the project is also about the people:

  - Present McStas team members






- K Lefmann E Farhi P Willendrup E Knudsen U Filges J Brinch

  - Past McStas team members




- K Nielsen PO Åstrand K Lieutenant P Christiansen