

McStas 25 year celebration

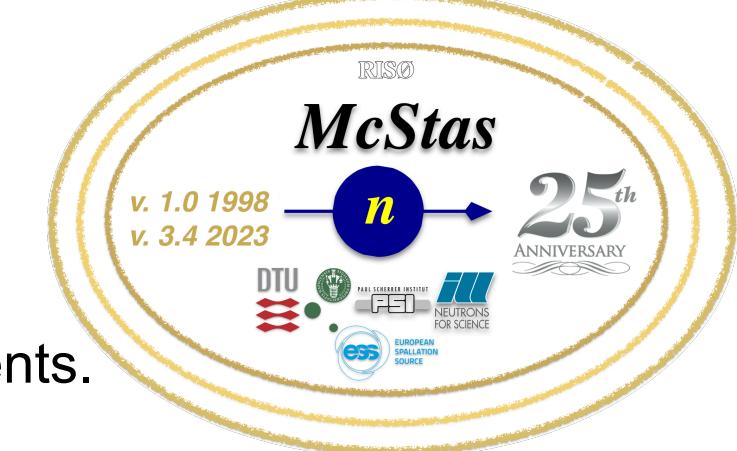
2023/11/16

PETER WILLENDRUP, ESS DMSC & DTU PHYSICS



McStas Introduction

- Flexible, general simulation utility for neutron scattering experiments.
- Original design for Monte carlo Simulation of triple axis spectrometers
- Developed at DTU Physics, ILL, PSI, Uni CPH, ESS DMSC
- V. 1.0 by **K Nielsen & K Lefmann** (1998) RISØ
- Currently ~6 people on joint McStas-McXtrace team but only **2 full time**, based at ESS DMSC / DTU



GNU GPL v3 license
Open Source



The screenshot shows the McStas homepage in a web browser. The page features a header with the McStas logo and navigation links for "Getting Started", "Latest Headlines", "Documentation", "Download", "Mailing list", "Search web/mailinglist", "Documentation", "Workshops/conferences", "Developments", "Links", "Report bugs", "CVS", and "McStas Ubuntu live-dvd". The main content area has a section titled "McStas - A neutron ray-trace simulation package" with a sub-section about simulated scattering from a hollow cylinder vanadium sample. It also includes sections for "Recent news" and "May 18th, 2009: McStas related slides / posters from ICNS".



Project website at
<http://www.mcstas.org>

mcstas-users@mcstas.org mailinglist

McXtrace - since jan 2009 similar for X-rays



Main Page – McXtraceWiki

http://www.mcxtace.org/index.php?title=Main_Page

Most Visited ▾ Getting Started Latest Headlines ▾ Geekblog ▾ Nyheder ▾ dr.dk ▾ open streami... Log in / create account

article discussion edit history

McXtrace

Main Page

[edit]

McXtrace

McXtrace - Monte Carlo Xray ray-tracing is a joint venture by

Risø DTU ESRF JJ X-RAY

Funding from NABIIT, DSF and the above parties.

Our code will be based on technology from

For information on our progress, please subscribe to our user mailinglist.

<mailto:webmaster@mcxtace.org>

navigation

- Main Page
- Partners
- Project People
- Project Status
- Vacancies
- Project Goal
- Mailing List
- Links
- SMEXOS talks
- SRI09 abstracts

search

Go Search

toolbox

- What links here
- Related changes
- Upload file
- Special pages
- Printable version
- Permanent link

This page was last modified 13:15, 25 February 2009.

This page has been accessed 2,049 times.

Privacy policy

About McXtraceWiki

Disclaimers



- Synergy, knowledge transfer, shared infrastructure, repo etc.



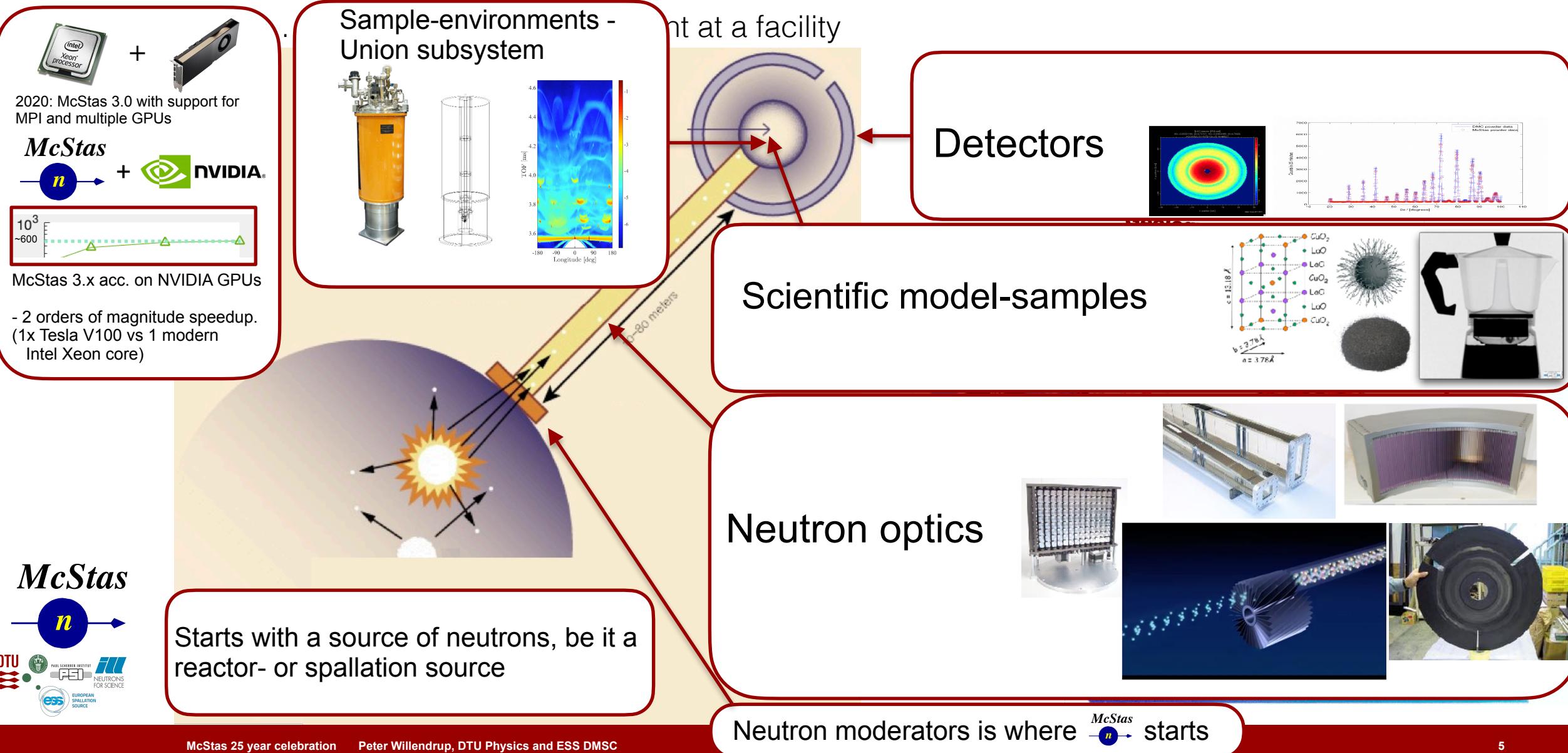
Funding sources

During its lifetime, McStas efforts have been supported through several European Union RTD and JRA programmes, plus several instrument development projects for facilities.

Project	Program type	Funding period
XENNI	RTD (EU FP4)	1996 - 2000
Cool Neutrons	RTD (EU FP4)	1998 - 2001
SCANS	RTD (EU FP5)	2000 - 2004
MCNSI	JRA in NMI3 (EU FP6)	2004 - 2006
MCNSI7	JRA in NMI3/FP7 (EU FP7)	2006 - 2008
NMI3-II/FP7 outreach project	JRA in NMI3/FP7 (EU FP7)	2012 - 2016
ISIS TS2 EU project	Infrastructure project in (EU FP6)	2006-2009
Instrument simulations for the ESS design update	Danish in-kind project toward the ESS	2009-2012
Secondment of P Willendrup from DTU for supporting instrument simulations for the ESS	33% part of the ESS DMSC	2014-2022
SINE2020	Part of WP3 and WP8 (EU H2020)	2016 - 2019
PaNOSC	Part of Software and E-learning (EU H2020)	2019 - 2022
HighNESS	Contributions to WPs 6,8,9 (EU H2020)	2020 - 2023
Secondment of P Willendrup from DTU for supporting instrument simulations for the ESS	100% part of the ESS DMSC	2023-2027



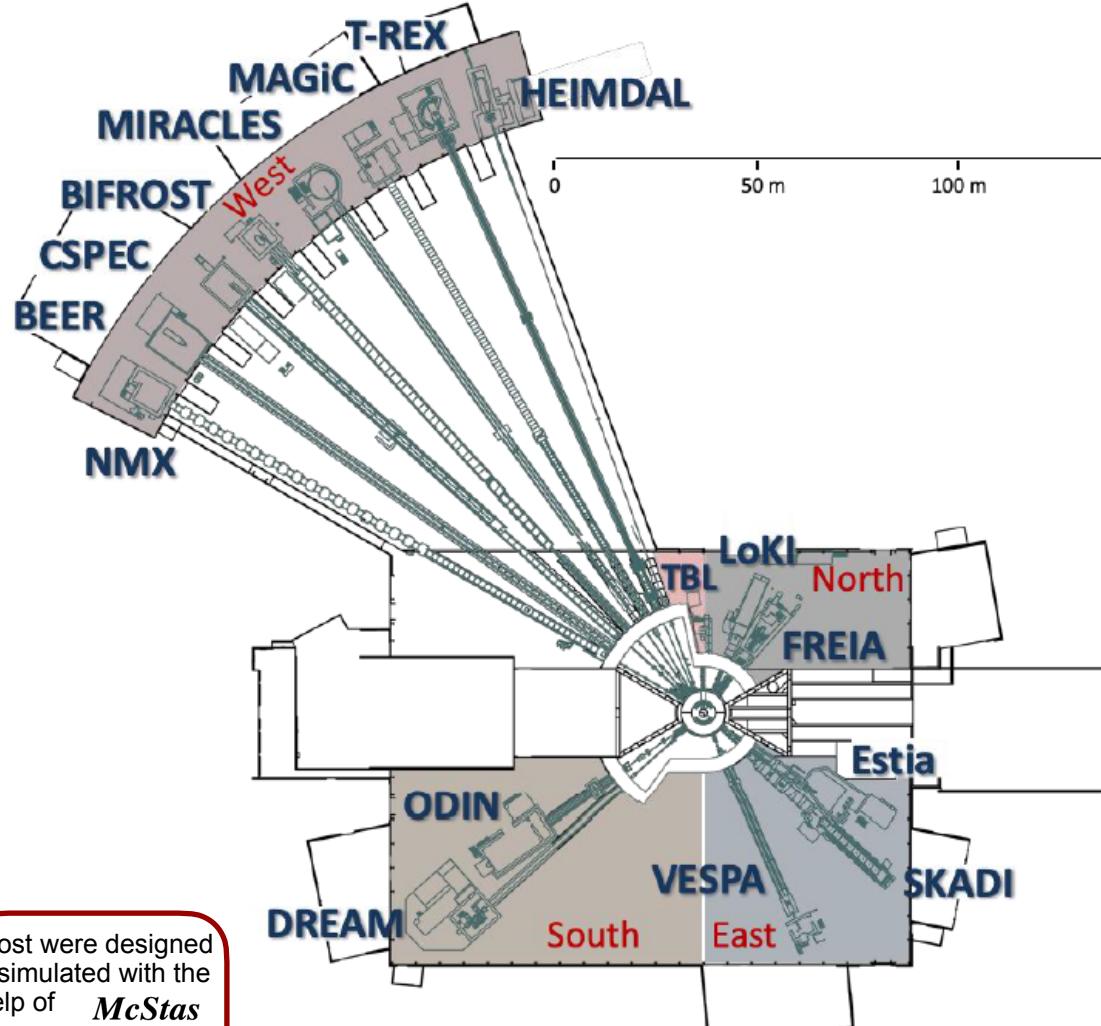
McStas: simulation toolkit for neutron scattering instruments, virtual experiments



Used in many places

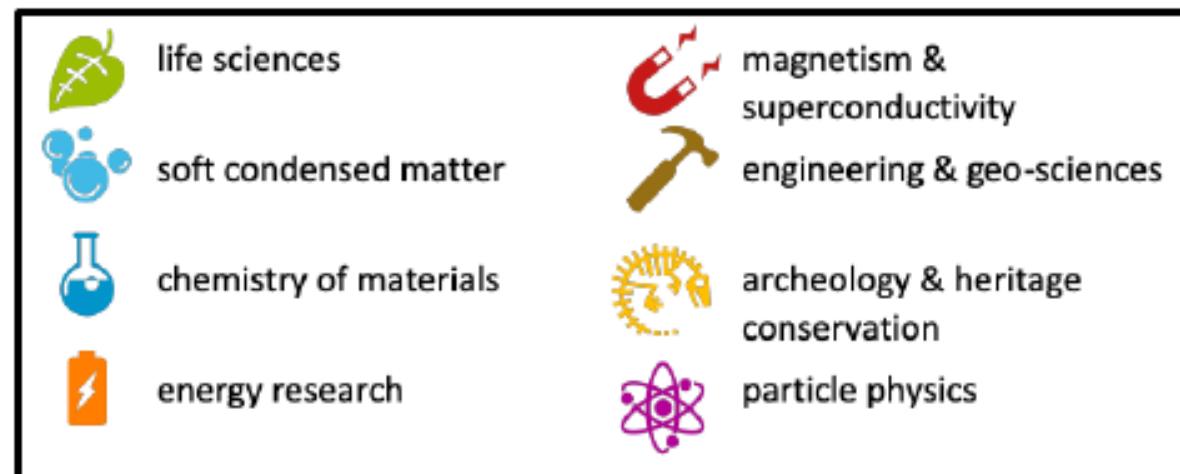


Instrument suite at ESS

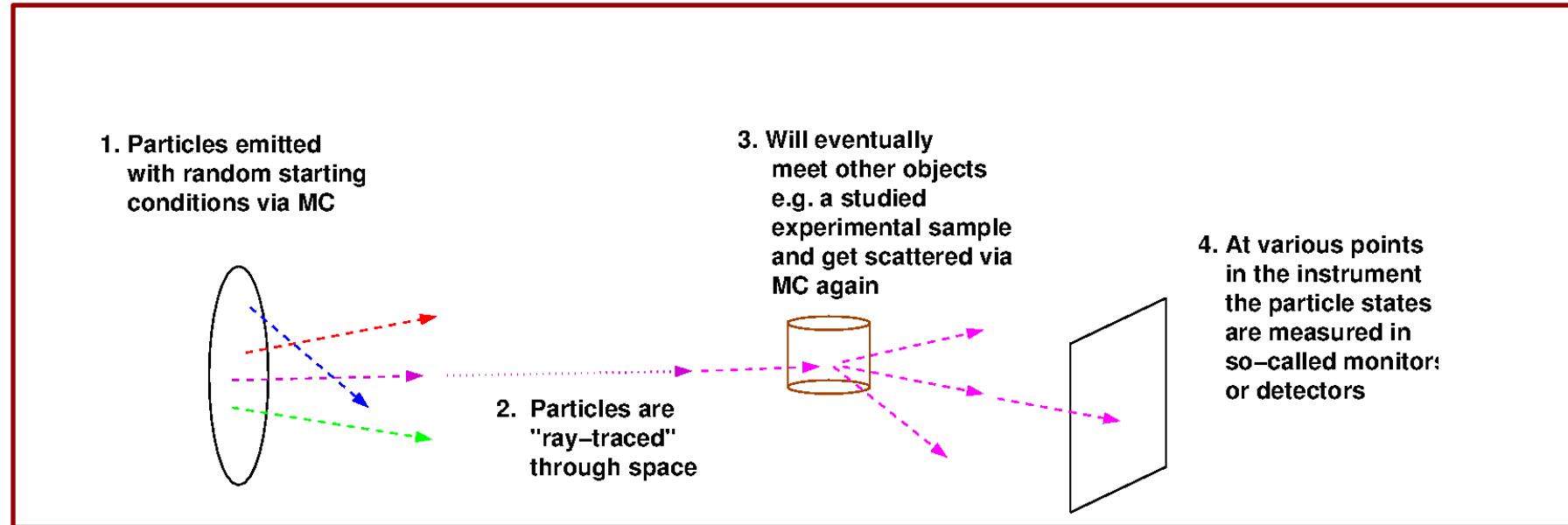


15 instruments

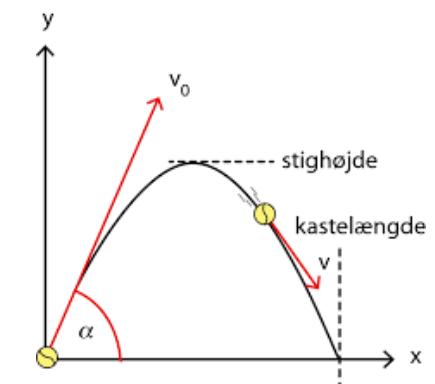
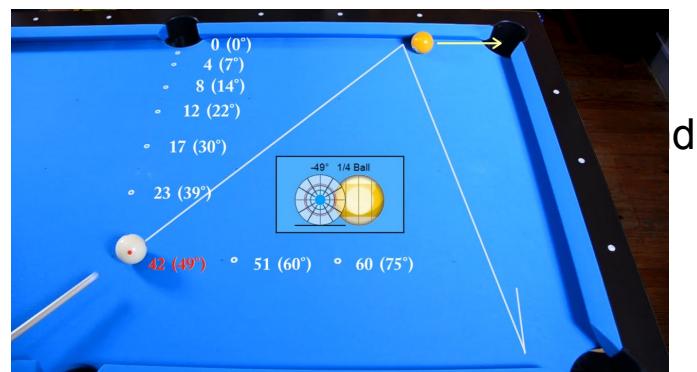
- 5 Large-scale structure instruments
- 5 Diffractometers
- 5 Spectrometer



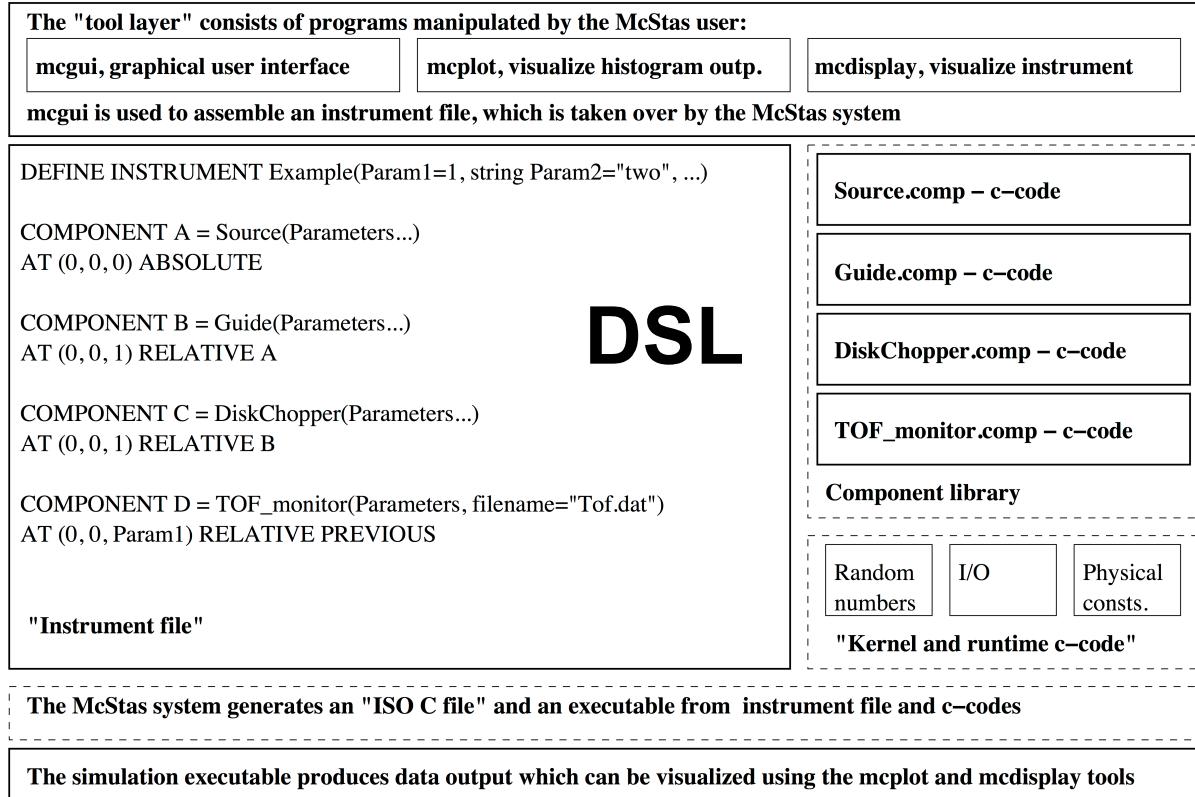
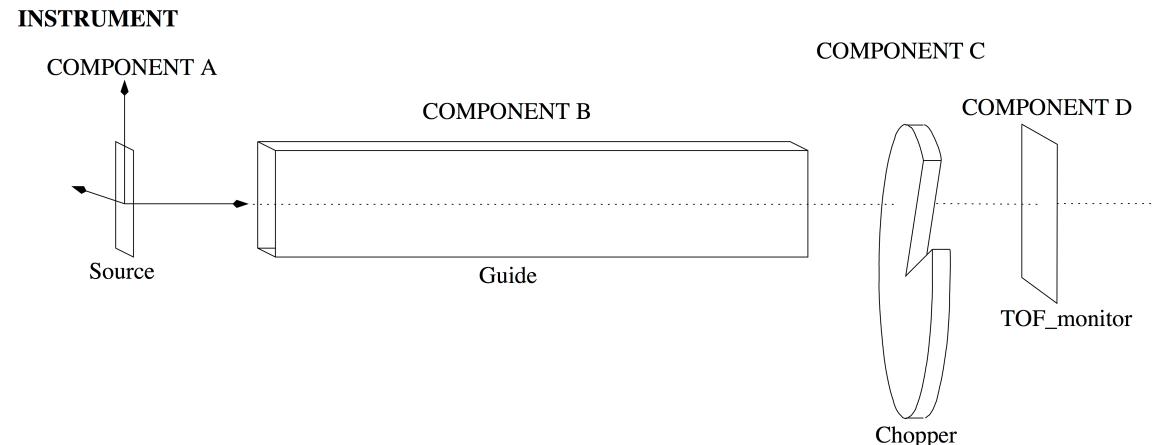
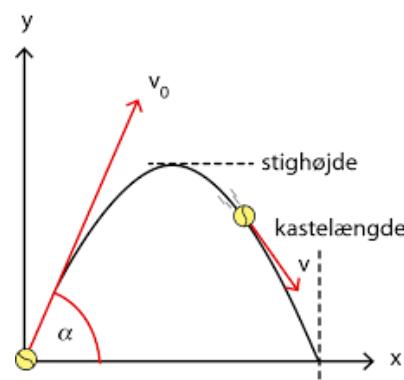
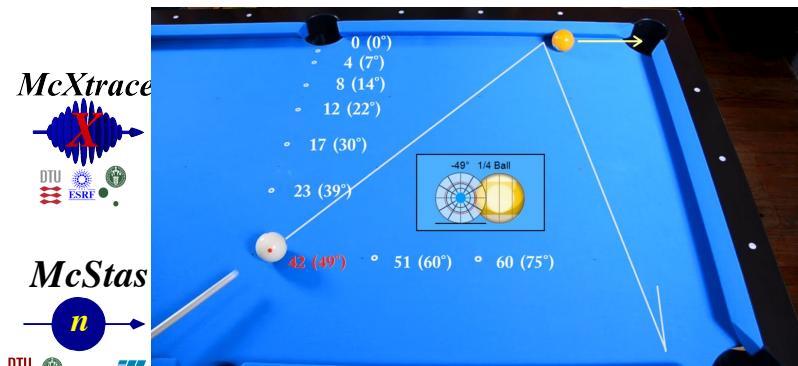
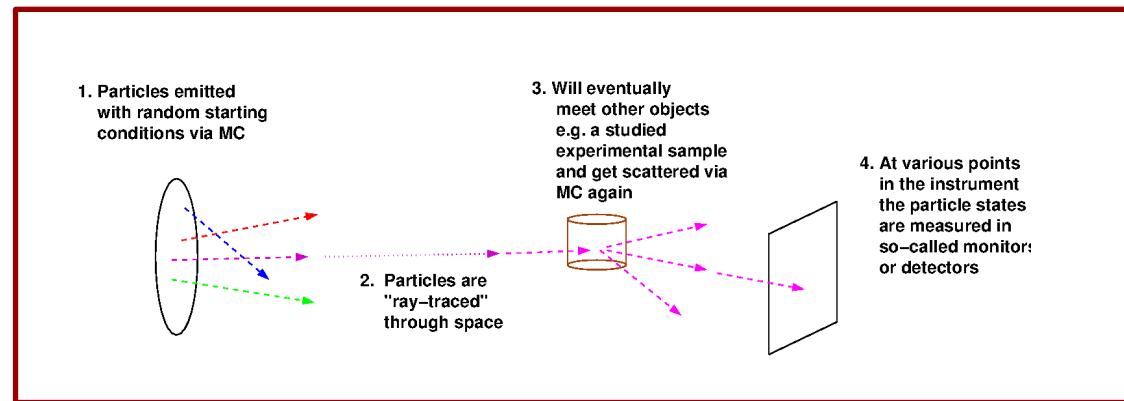
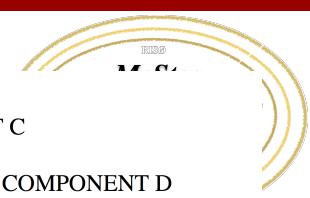
McStas and McXtrace are Monte Carlo ray-tracers



- For the neutrons, gravity kicks in... A cold neutron falls $\sim 10\text{cm}$ over 150m!
- Classical Newtonian mechanics, i.e.
- (independent, particles though...)



McStas and McXtrace Monte Carlo ray-tracers

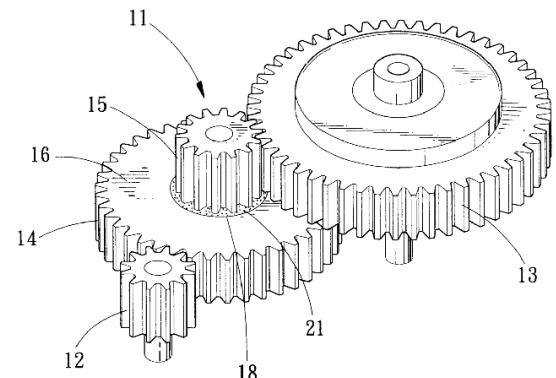
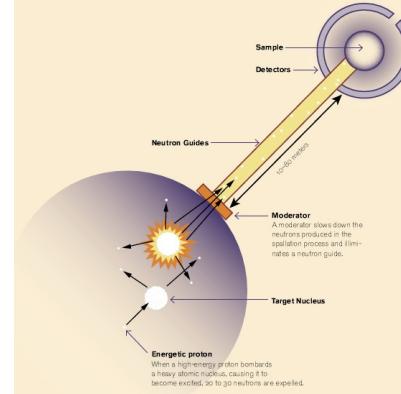


McStas tech overview

- Portable code (Unix/Linux/Mac/Windoze)
 - On the CPU-side, ran on everything from iPhone to 1000+ node cluster, intel, Alpha, PA-RISC etc.
- ‘Component’ files (>300) inserted from library
 - Sources
 - Optics
 - Samples
 - Monitors
 - If needed, write your own comps - **many are USER developments ~200-line “physicist” codes**



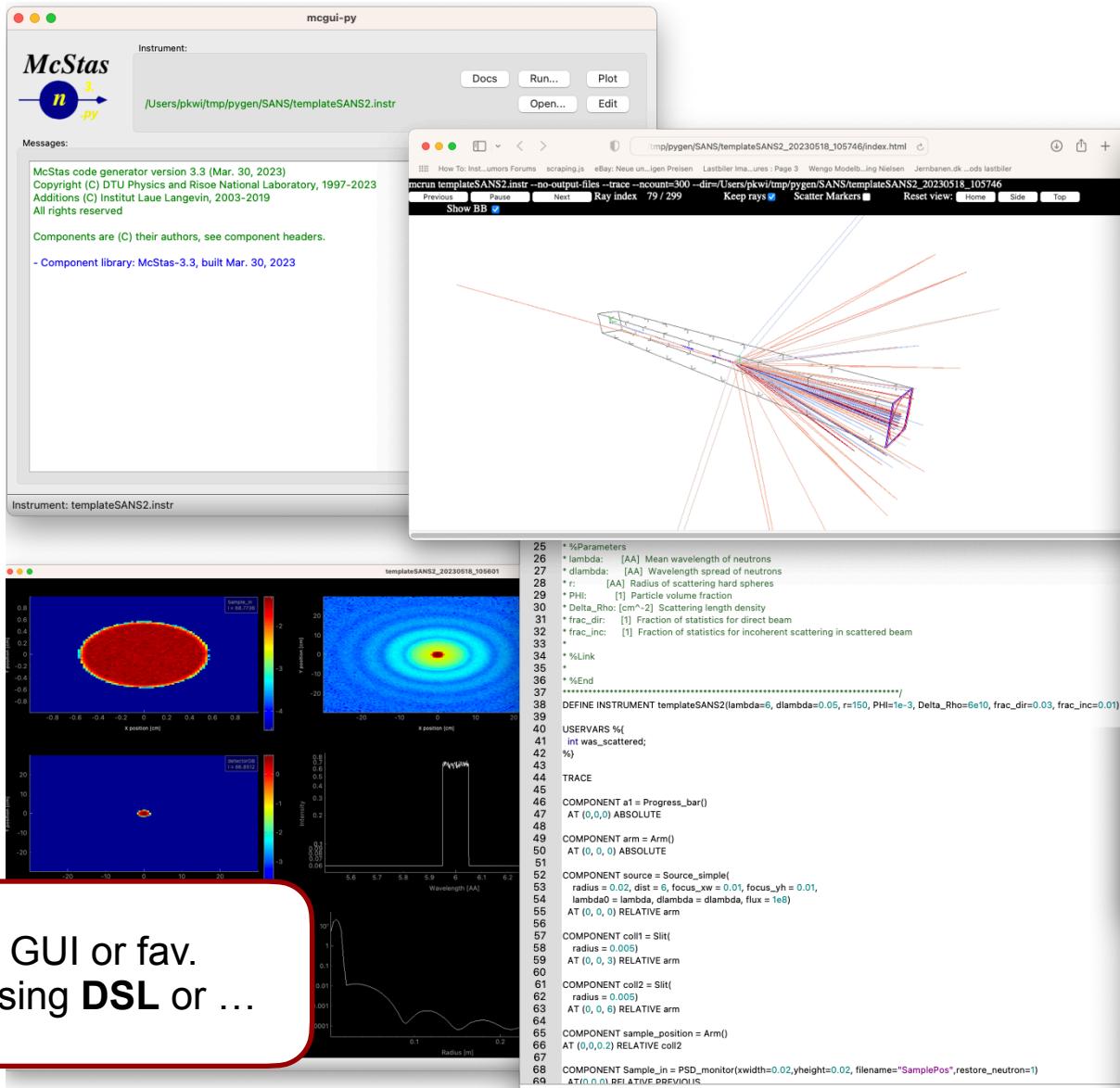
- DSL + ISO-C code-gen. (compiler technology / LeX+Yacc)
 - Simple Instrument language ISO C
- Component codes realizing beamline parts (including user contribs)
- Library of common functions
 - I/O
 - Random numbers
 - Physical constants
 - Propagation
 - Precession in fields
 - ...



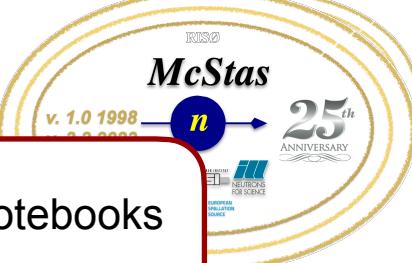
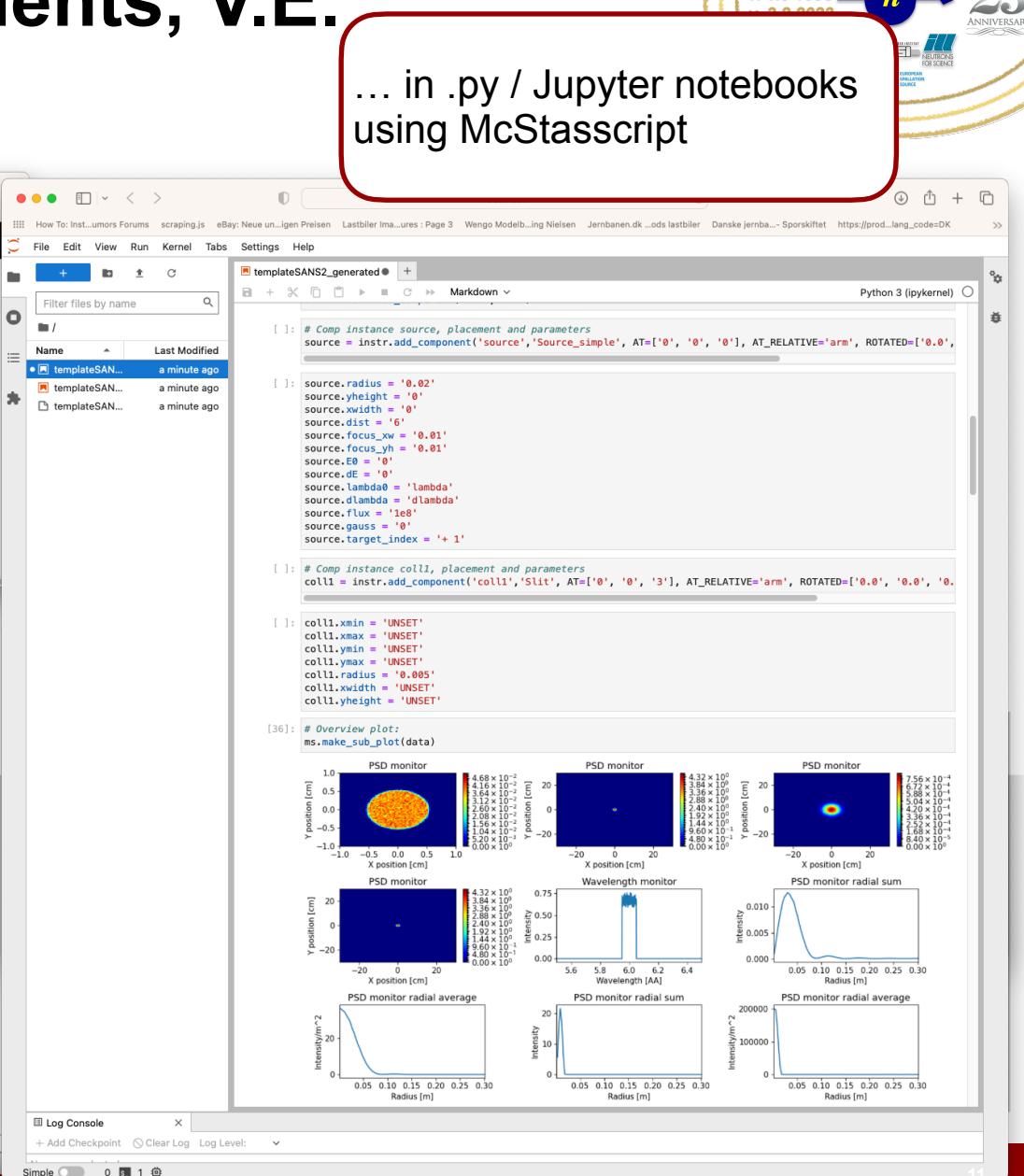
User experience:

- Write instrument
- Launch simulation (generates binary and runs simulation)
- Look at output data

McStas: simulation toolkit for neutron scattering instruments, V.E.



Work in GUI or fav.
editor using **DSL** or ...



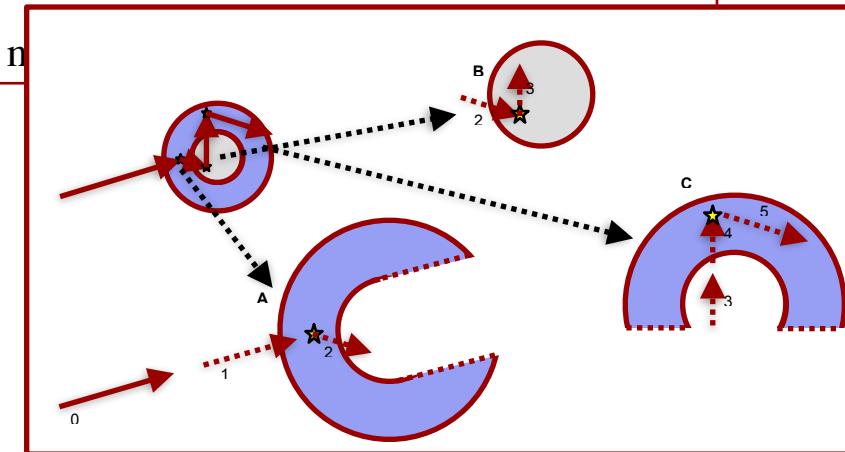
McStas is by design a “linear chain” of components

- But:
 - We have syntaxes/logic to e.g. GROUP components. (Think: XOR and similar logic)

```
{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] }
```

McStas is by design a “linear chain” of components

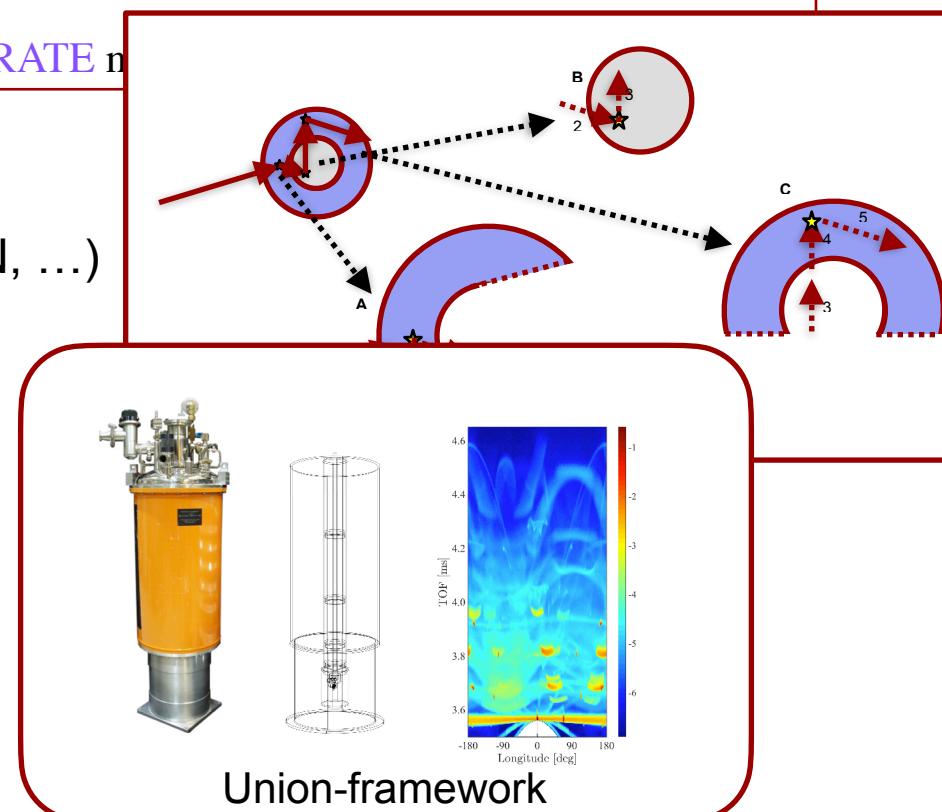
- But: AT (...) [RELATIVE [reference|PREVIOUS] | ABSOLUTE]
 • We {ROTATED {RELATIVE [reference|PREVIOUS] | ABSOLUTE} }
 cor {GROUP group_name}
 {EXTEND C_code}
 {JUMP [reference|PREVIOUS|MYSELF|NEXT] [ITERATE n]}
- Material-assemblies may be arranged in
 “concentric” onion-shells (Single_crystal, PowderN, ...)



McStas is by design a “linear chain” of components

- But:
 - We can now do:
 - {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 n]}

- Material-assemblies may be arranged in “concentric” onion-shells (Single_crystal, PowderN, ...)
- The Union subsystem (Mads Bertelsen) has been added, defining region(s) of the instrument where geometry and materials are decoupled and we completely deviate from the linear approximation
- NCrystal may be used to describe materials, also within Union. cfg=“materials_galore.ncmat”



McStas is by design a “linear chain” of components

(Other recent grammar additions....)

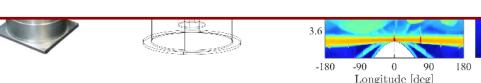
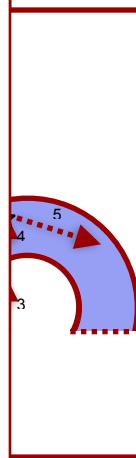
- But:
 - We use “CPU” to label non-GPU execution of a component in mixed CPU/GPU setups
 - We now use “SHELL” for cogen/compile-time system-hacks (create an instr snippet using e.g. bash)


```
SEARCH "/the/path/to/add/" COMPONENT ...
SEARCH SHELL "the_executable --and --some --options" COMPONENT ...
```
- Materials


```
"component" COMPONENT Origin = Progress_bar()
  AT (0,0,0) ABSOLUTE
```
- The instruments have been updated to support JSON metadata


```
METADATA JSON eniius_data %{
  ...{JSON string}...
%}
```
- Instruments can now define extra Python functions

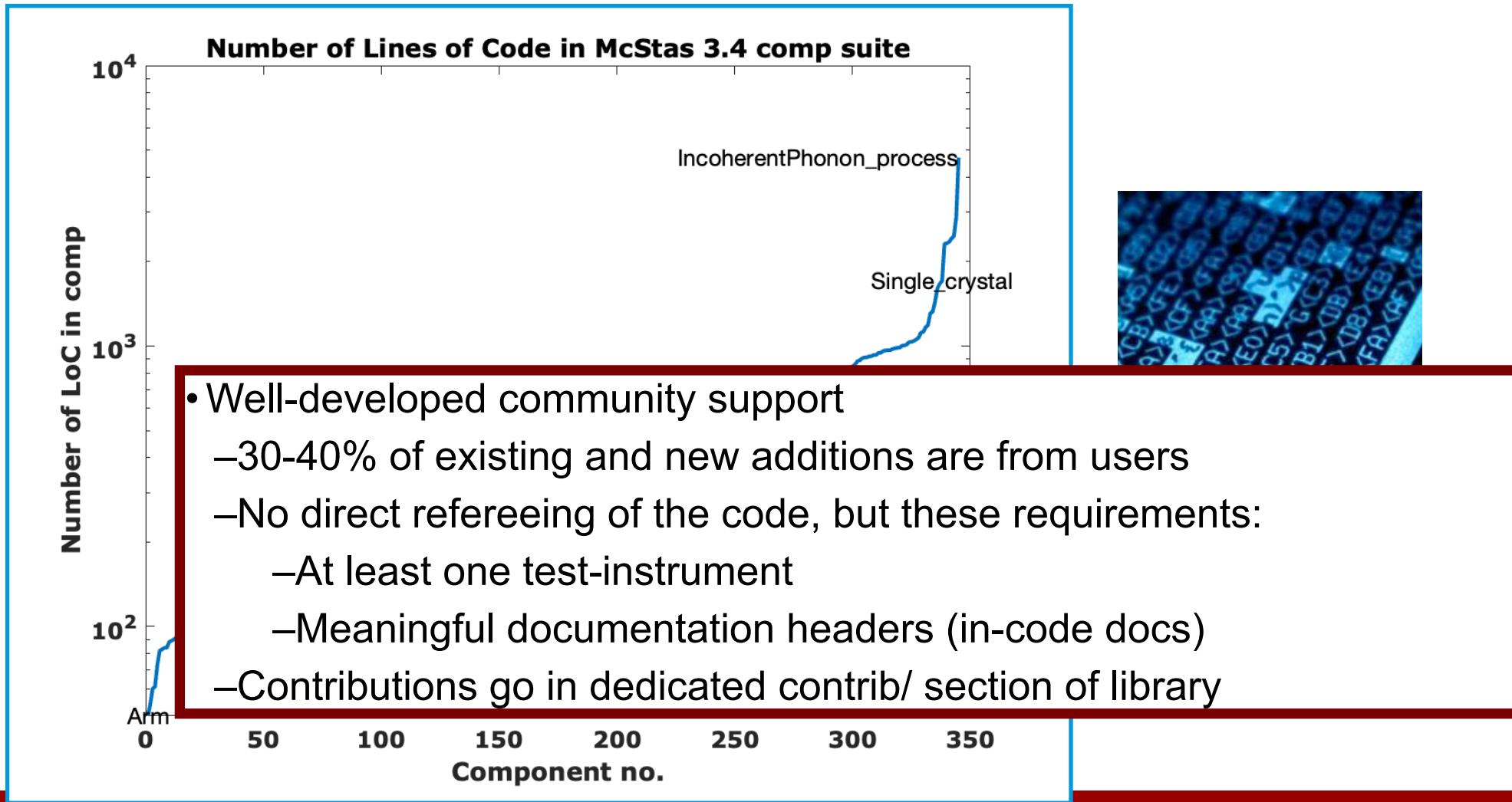

```
METADATA Python extra_function %{
  def the_function(a, b, c):
    return a + b / c
%}
```
- NCystal may be used to describe materials, also within Union. cfg="materials_galore.ncmat"



Union-framework

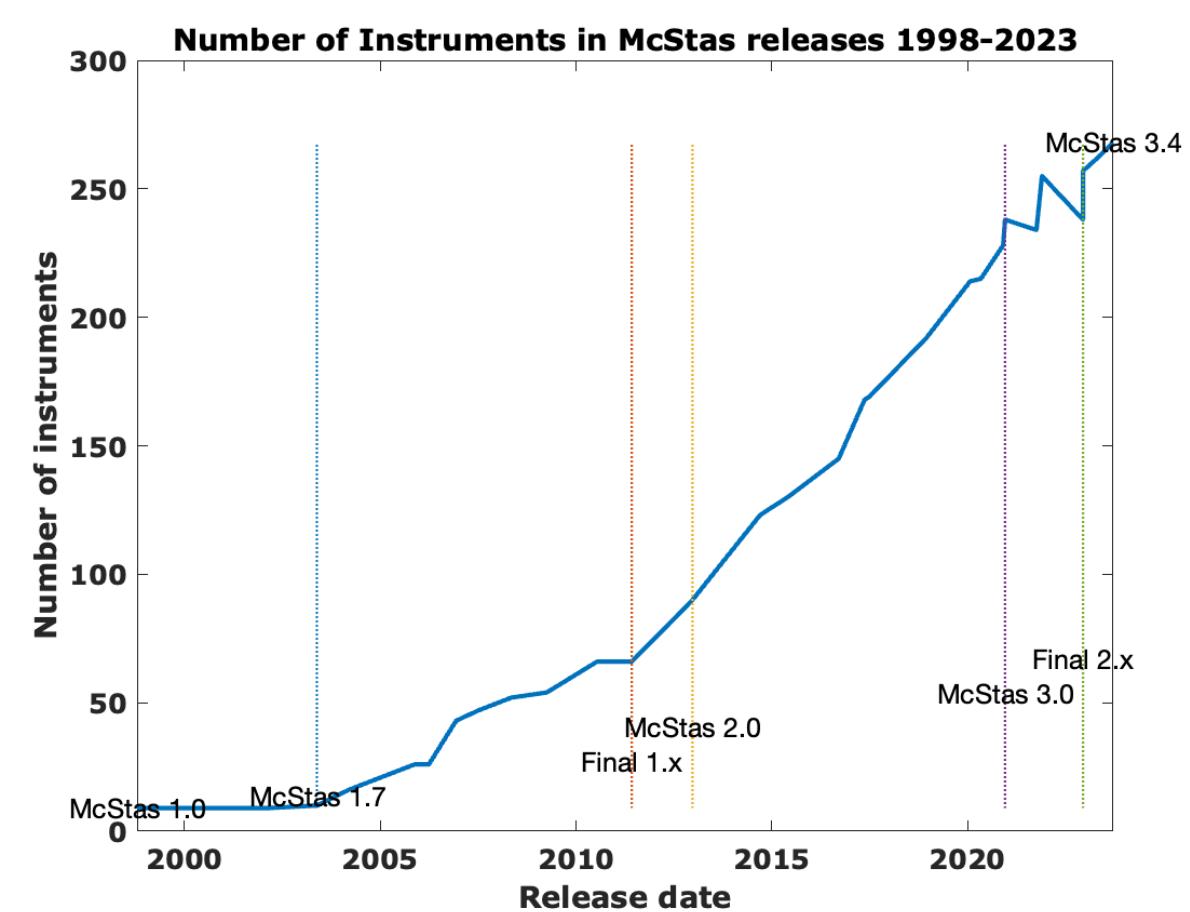
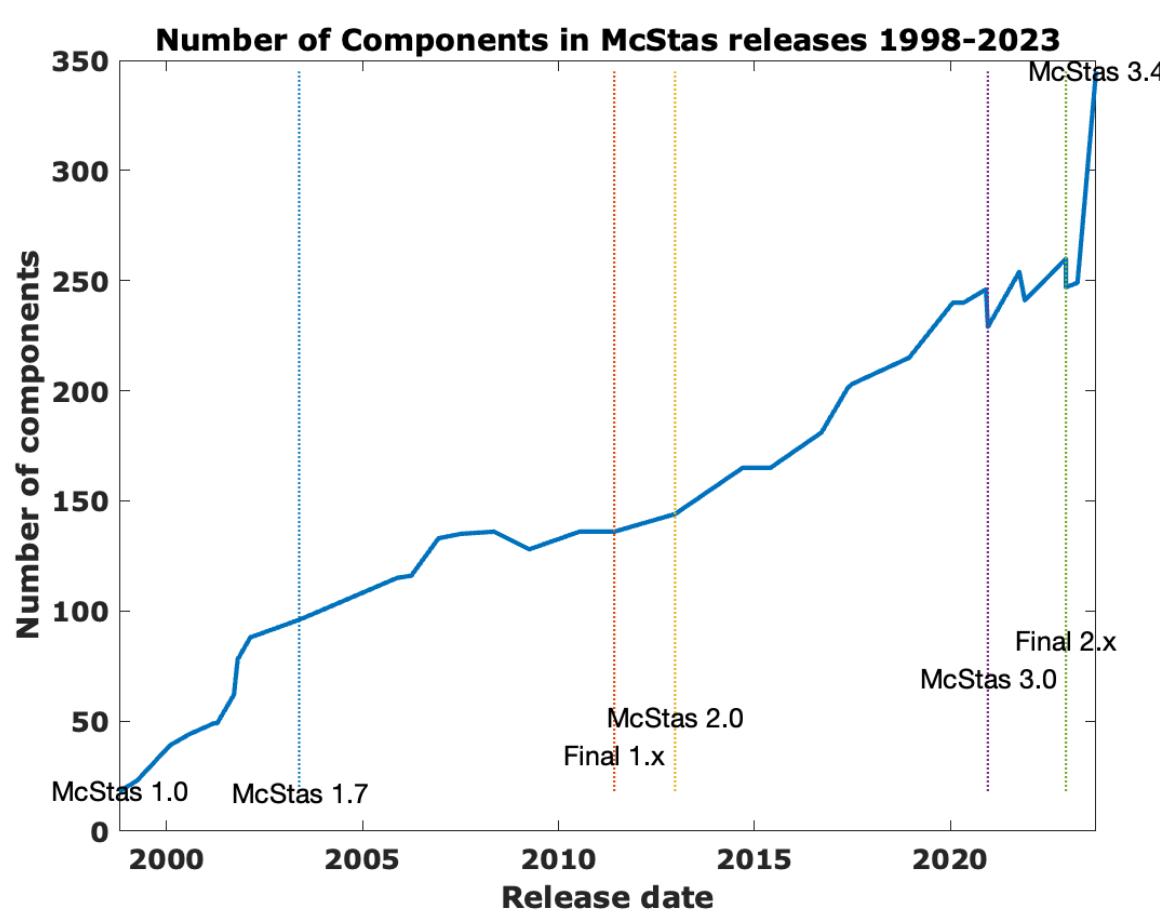
Writing new comps or understanding existing is not so complex...

- Many of them are quite simple and short... Statistics:



Component lib development

More than 350 comps, more than 250 instr examples

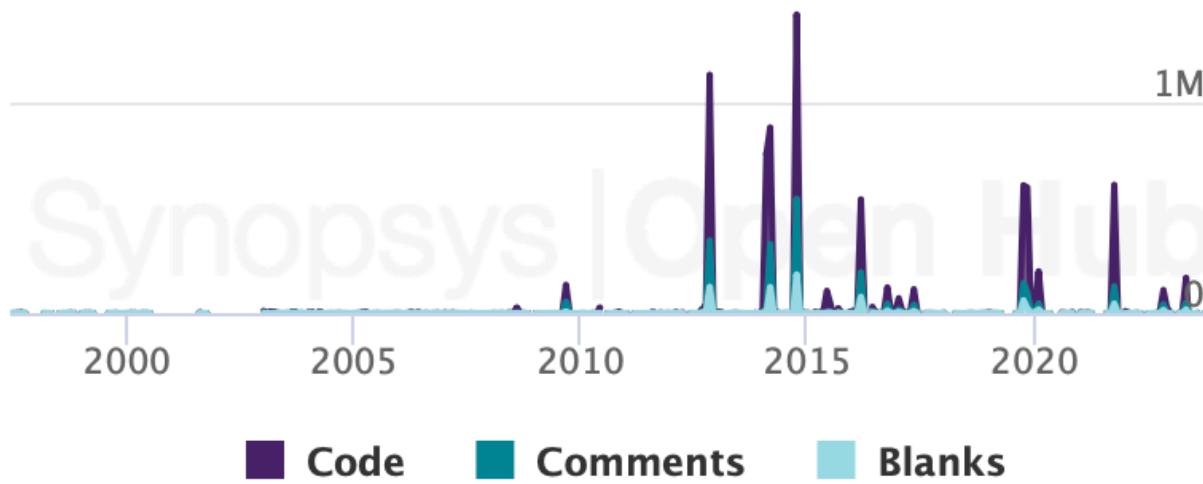


<https://www.openhub.net/p/mccode>

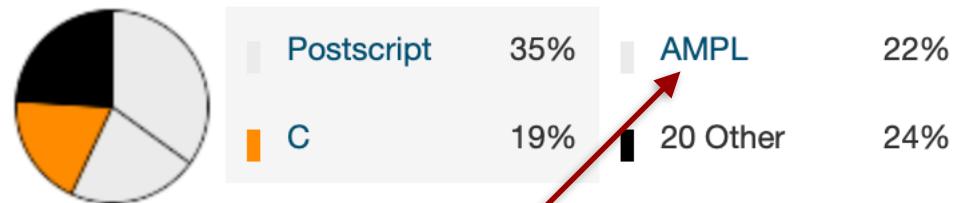
Stats and information on the codebase



Lines of Code



Languages



Nope, that's our DSL and grammar. :-) Which is close to "English".

In a Nutshell, McCode...

... has had 24,640 commits made by 60 contributors representing 2,177,413 lines of code

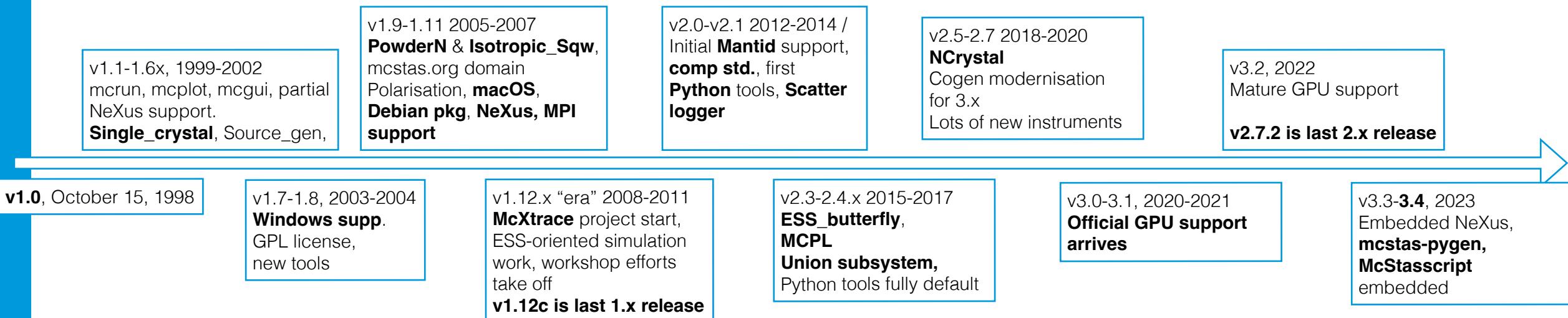
... is mostly written in Postscript with a low number of source code comments

... has a well established, mature codebase maintained by a large development team with increasing Y-O-Y commits

... took an estimated 636 years of effort (COCOMO model) starting with its first commit in June, 1997 ending with its most recent commit about 1 month ago

McStas timeline at a glance

When did what functionality arrive



McStas collaboration

Thanks and acknowledgment

Kristian Nielsen

Got input from RISØ physicists, (Incl. Kim & Henrik) architect behind “internals” and LeX-Yacc gram.



Releases 1.0-1.4



Henrik Rønnow
Drafted v 1.0
grammar w. Kim
& Kristian



Kim Lefmann

Author of many components, orig. “power user”, supervised many KU stud. and secured funding.

Part of McStas all along the way.



Kurt Clausen:

Came up with original idea of “framework”, secured first EU funding.



and ESS DMSC

... lots of people...



Emmanuel Farhi

First non-DK “McStas system developer”,

PW mentor for the deeper tech.



Grammar ext.
advanced comps.
+ lots more
Now works
mainly on
McXtrace.



McStas collaboration

Thanks and acknowledgments

Kristian Nielsen

Got input from RISØ physicists, (Incl. Kim & Henrik) architect behind "internals" and LeX-Yacc gram.



Releases 1.0-1.4

Henrik Rønnow
Drafted v 1.0 grammar w. Kim & Kristian



Kim Le

Author of complex "power" supervisory KU student secure

Part along

Kurt Clause

Came up with original idea "framework" secured first funding.



Peter Christiansen
1st pol. solution



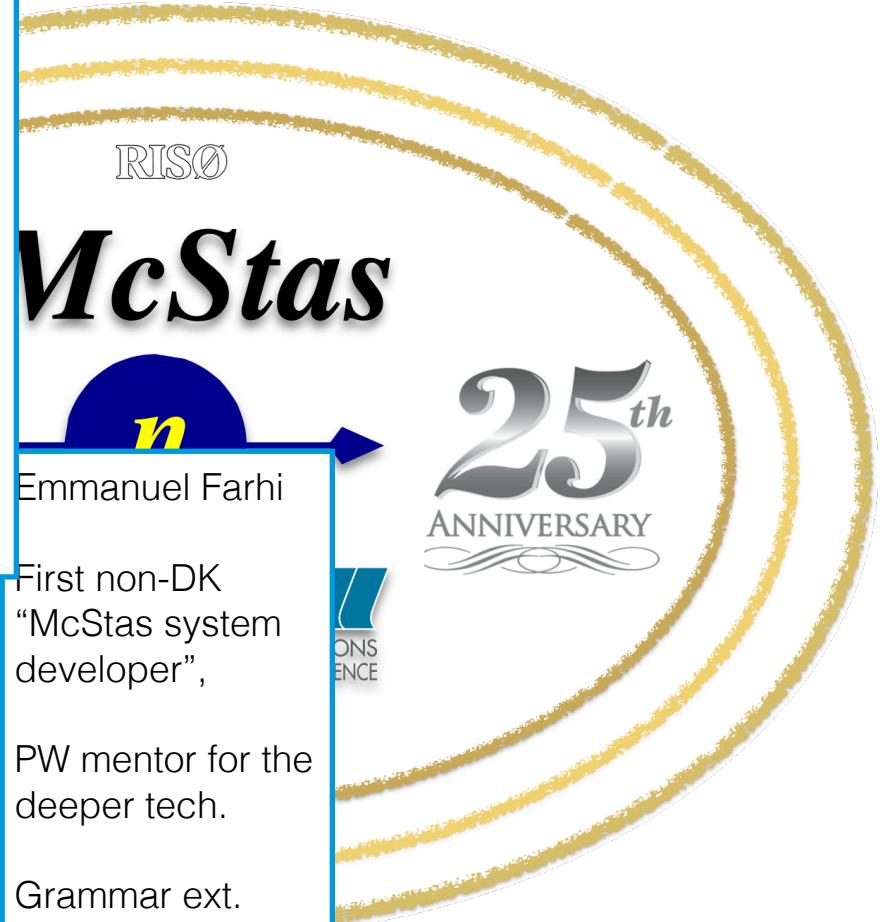
PW:

"Mr. McStas"

2002- ?

Custodian, community caretaker, sustained workforce.

Support for newcomers and power users alike.



Emmanuel Farhi

First non-DK "McStas system developer",

PW mentor for the deeper tech.

Grammar ext. advanced comps. + lots more
Now works mainly on McXtrace.



McStas collaboration

Thanks and acknowledgments

Kristian Nielsen

Got input from RISØ physicists, (Incl. Kim & Henrik) architect behind "internals" and LeX-Yacc gram.



Releases 1.0-1.4

Mads Bertelsen

"Next generation", Phd with KL, contributed adv. systems "around" and "in" McStas:

guide_bot and Union systems.

Irik Rønnow



Kim Le

Author of compo "power" superv KU stu secure

Part alon

Kurt Clause

Came up w original idea "framework" secured fir fund



Peter Christiansen

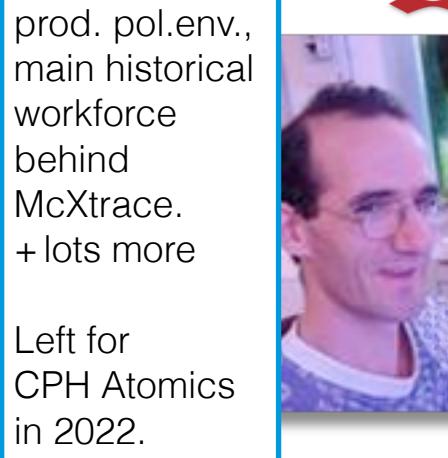
1st pol. solution



Erik Knudsen:

Developer of prod. pol.env., main historical workforce behind McXtrace. + lots more

Left for CPH Atomics in 2022.



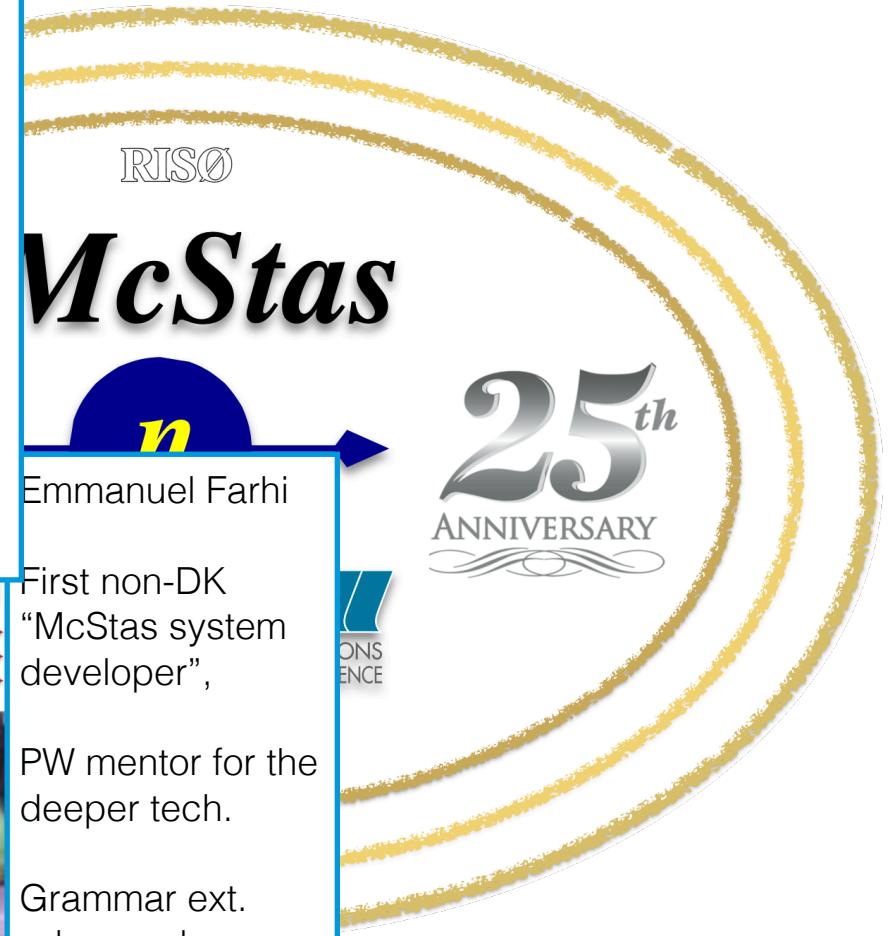
PW:

"Mr. McStas"

2002- ?

Custodian, community caretaker, sustained workforce.

Support for newcomers and power users alike.



McStas collaboration

Thanks and acknowledgments

Kristian Nielsen

Got input from RISØ physicists, (Incl. Kim & Henrik) architect behind "internals" and LeX-Yacc gram.

Releases 1.0-1.4



Kim L

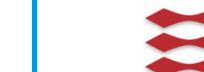
Author compo
"power"
superv
KU stu
secure

Part
alon



Peter Christiansen

1st pol. solution



PW:

"Mr. McStas"

2002- ?

Custodian,
community
caretaker,
sustained
workforce.

Support for
newcomers
and power
users alike.



DTU Fysik



Mads Bertelsen

"Next generation",
Phd with KL,
contributed adv.
systems "around"
and "in" McStas:

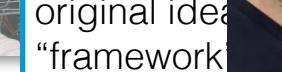
guide_bot and
Union systems.

Irik Rønnow



Kurt Clause

Came up w/
original idea
"framework"
secured fir
func



Erik Knudsen:

Developer of
prod. pol.env.,
main historical
workforce
behind
McXtrace.
+ lots more

Left for
CPH Atomics
in 2022.



Esben Klinkby
Mcnp-hooks,
scatter-logger



Jakob Garde
2nd round of .py
tools, GPU
deeper tech.



Grammar ext.
advanced comps.
+ lots more
Now works
mainly on
McXtrace.

Johan Brinch
Transfer to CMake
1st round of .py
tools

25th

McStas collaboration

Thanks and acknowledgments

Kristian Nielsen

Got input from RISØ physicists, (Incl. Kim & Henrik) architect behind "internals" and LeX-Yacc gram.

Releases 1.0-1.4

Mads Bertelsen

"Next generation", Phd with KL, contributed adv. systems "around" and "in" McStas:

guide_bot and Union systems.



Kim Le

Author compo
"power"
superv
KU stu
secure



PW:

"Mr. McStas"

2002- ?

Custodian,
community
caretaker,
sustained
workforce.

Support for
newcomers



Torben R. Nielsen:

Contributed solution
for connection with
"Mantid" data
reduction,
work on SASmodel
integration

Gregory Tucker

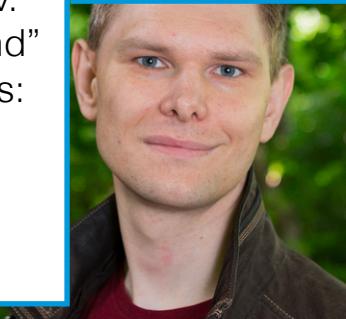
ESS event-formation
hookup, recent
syntax / code-
generator additions

Thomas Kittelmann

Main developer of
MCPL particle list
Format and
NCrystal structure/
dynamics lib for MC



Plus MANY others among the user community at neutron- facilities, students etc.



behind
McXtrace.
+ lots more

Left for
CPH Atomics
in 2022.



deeper tech.

Grammar ext.
advanced comps.
+ lots more
Now works
mainly on
McXtrace.



Johan Brinch
Transfer to CMake
1st round of .py
tools



Demo time?? :-)

