Digital Twin Design and Implementation at the Institut Laue-Langevin

Shervin Nourbakhsh (Institut Laue-Langevin)

NOBUGS 2024

Objectives

Enrich the offer of user tools with the possibility to run a virtual experiment.

Training:

- newcomers to the instrument control system
- users new to a particular instrument, its configuration and capabilities

Settings optimization: study and optimize instrument settings for a specific configuration to maximize some figure of merit (e.g. intensity vs resolution)

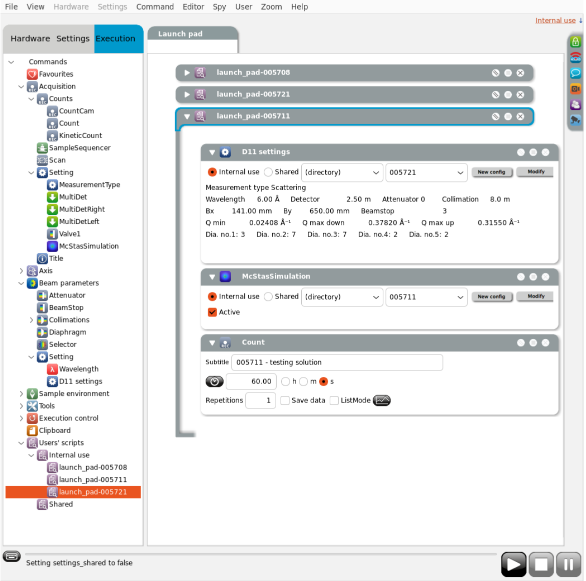
Analysis: improve analysis with better understanding of some background sources and uncertainties (e.g. effect of a possible mis-alignment)

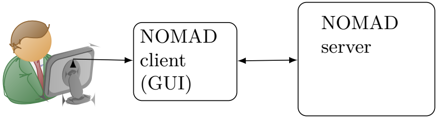
Support material: enrich proposals for demanding beam time with results from simulated data with the specific instrument taking into account its capabilities

Design Requirements

- usable by users with **basic knowledge** in neutron ray-tracing and interaction simulation
- use the **familiar interface** of instrument control to configure and start acquisition
- simulated data must be treated as the real data → written to disk in **same format**
- use **state-of-the-art** simulation software (e.g. McStas for neutron ray-tracing)

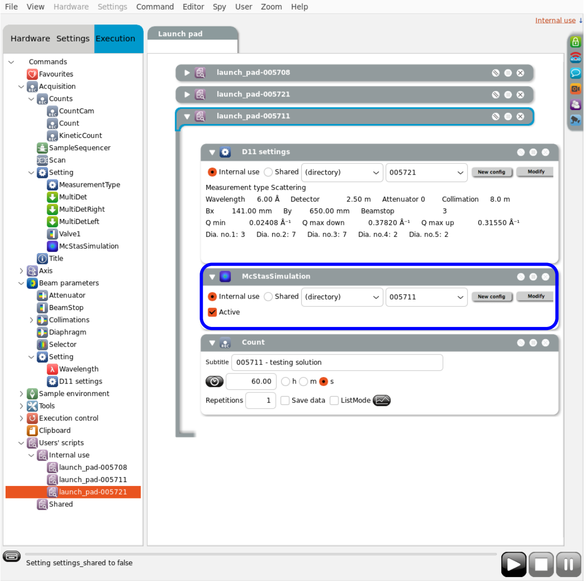
Instrument Control GUI

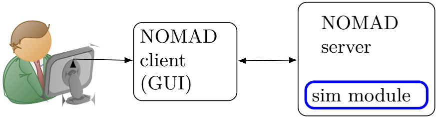




- NOMAD is ILL's Instrument Control System
- The GUI is a java client connecting to the server
- NOMAD core is a c++ server
- from the GUI users can:
  - control the instrument settings
  - program the acquisition workflow

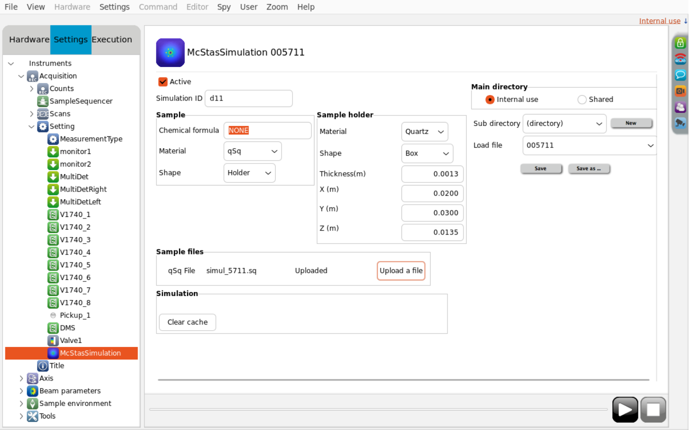
Instrument Control GUI





- NOMAD is ILL's Instrument Control System
- The GUI is a java client connecting to the server
- NOMAD core is a c++ server
- from the GUI users can:
  - control the instrument settings
  - program the acquisition workflow

Specific module for simulation settings



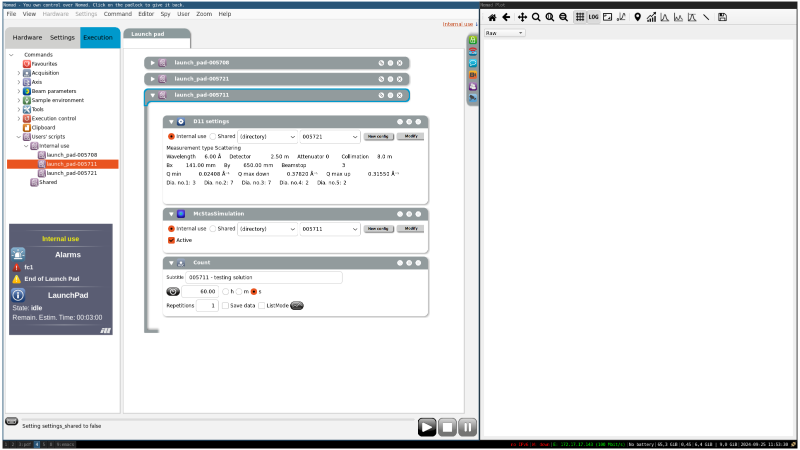
Sample settings

- shape and size
- material or information about scattering probability from theoretical calculations

Sample holder

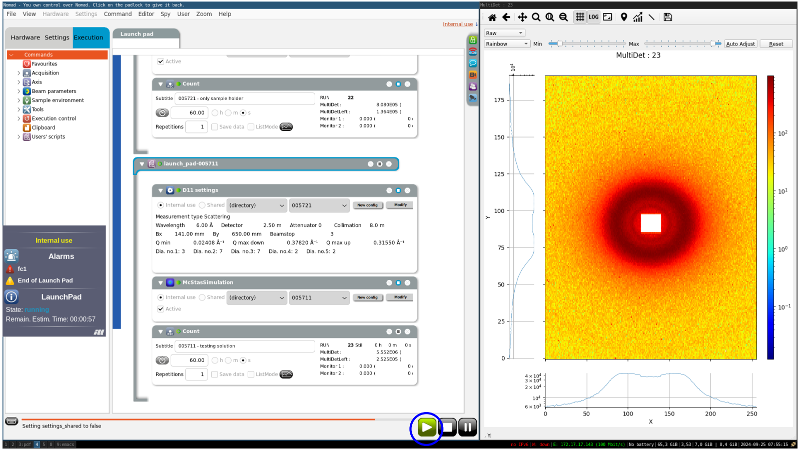
- shape and size
- material

Virtual acquisition



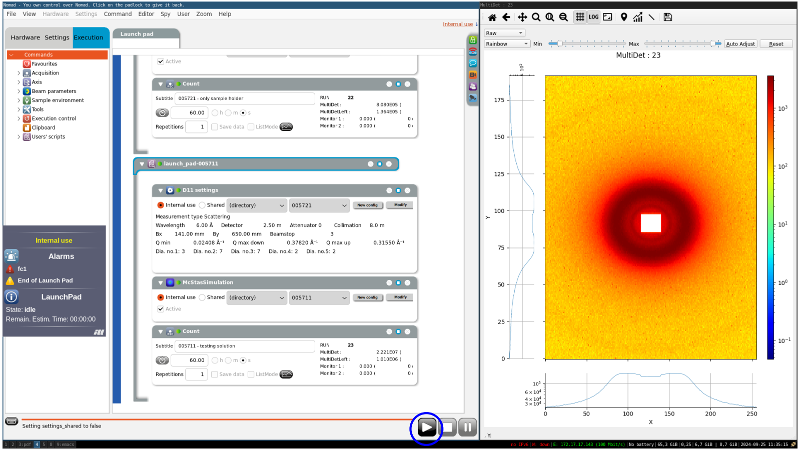
- User can start a simulation as used to do with the experiment.

Virtual acquisition



- User can start a simulation as used to do with the experiment.
- Feedback on progress

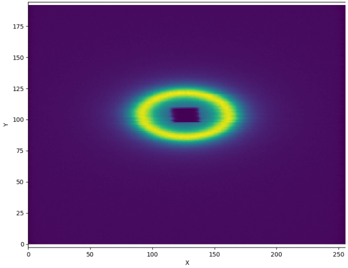
Virtual acquisition

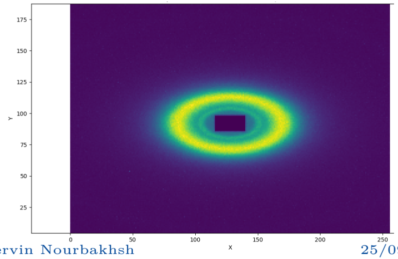


- User can start a simulation as used to do with the experiment.
- Feedback on progress
- Results updated at time intervals

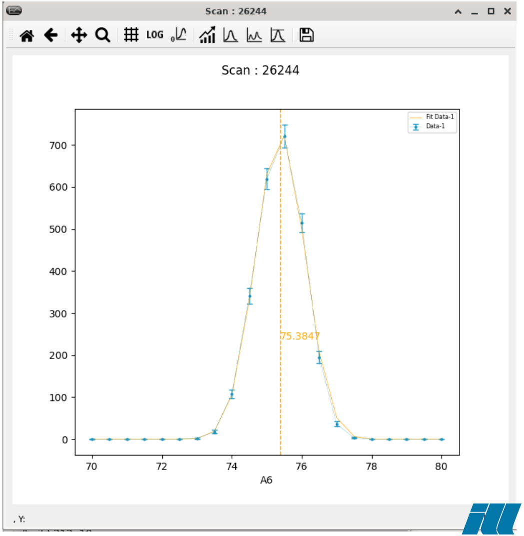
Simulation result

Scattering on calibration sample with sample holder (D11)



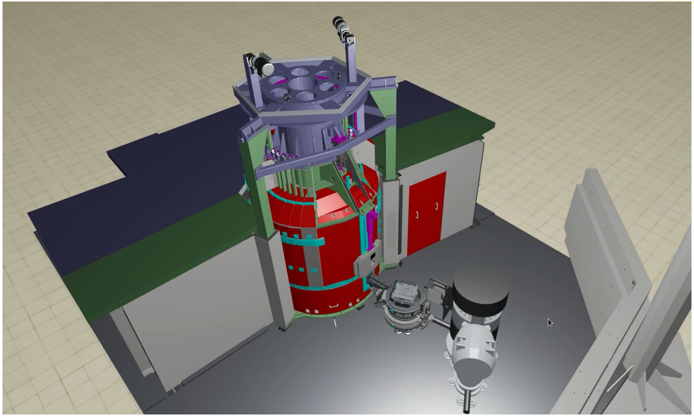


Scan: one simulation per point (THALES)



3D view

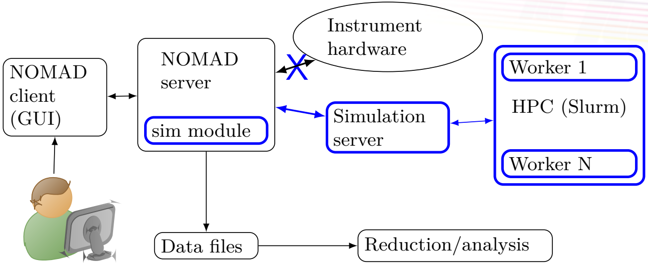
Y. Le Goc et al.



Better understanding of the instrument, its current configuration and possible movements.

- Live 3D view of current configuration.
- Animated showing moving parts at change of configuration.

Overview



C++ client API: sends simulation requests with instrument and sample parameters and receives results

SIM server: receives requests, dispatches workers, packs and returns results

Workers on **HPC (Slurm)** running the instrument simulation executables (e.g. McStas)

3D server: receives instrument parameters, generates 3D view in HMTL5 page

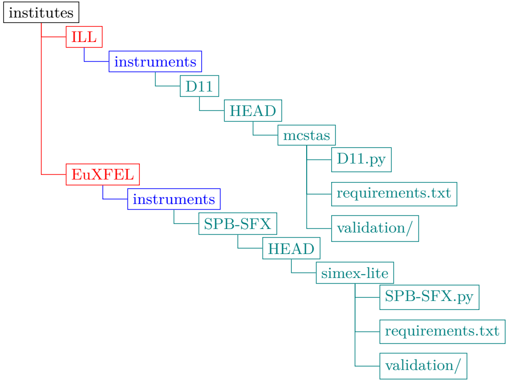
CAMEO middleware (see Le Goc's poster) provides:

- APP management (start/stop) also on remote machines
- Communication between managed APPs

Simulation Executable

- Workers can run any program to perform the simulation
- McStas used as state-of-the-art neutron ray-tracing software
- Instrument description
  - in Python using **McStasScript**[a] and **libpyvinyl** library[b]
  - retrieved from **PANOSC Vinyl** public "instrument description repository" (Github)
  - usable also in Jupyter notebook
- Instrument executables compiled for **CPU and GPU** and binary packages created and installed on HPC nodes.

Instrument description repository structure:



[a] Mads Bartelsen

[b] Carsten Fortmann-Grote, Mads Bertelsen, Juncheng E, Shervin Nourbakhsh

Summary

Objectives achieved:

- A prototype setup for a digital twin (DT) at ILL has been developed
- The DT can be used by users with no knowledge about simulation
- Data are available in the usual format, ready for reduction and analysis via the normal workflow.
- The client-server model allows further development of different interfaces to the simulation server (e.g. Jupyter notebooks via python API, or other client program)
- The parallelization on HCP (Slurm) has been implemented.
- Workers are flexible to allow different simulation softwares.
- McStas simulation binaries running on both CPUs and GPUs.
- 3D live view of the instrument in current configuration and showing movements at configuration change available as a web application.

Further steps:

- Add non-simulated background from real data.
- Develop surrogate models (ML) for faster simulations.