

Thomas Löhnert, Aaron James Long (ISIS, STFC)

John Holt (Tessella)

Generalising the High-Level Geometry System for Reflectometry Instruments at ISIS

Introduction

Reflectometers have complex motion requirements as equipment needs to maintain its position and angle to a neutron beam with a changing beam path.



As part of the ongoing migration to our new EPICS-based control system, we have re-implemented the beamline-level geometry layer used to hide this complexity from the user.

We have used this opportunity to redesign the architecture of the system with the goal of being easily reconfigurable and extensible for future beamline developments.

Server Architecture

- IBEX is client-server
- Reflectometry driver is a python service running locally on beamline server
- Uses PCASpy to interface to channel access – acts like any EPICS IOC to the outside



Schematic of a typical reflectometry beamline

teraction			
	Front Ranel All Darameters Driver Constitutions	In Motion: 🥚	STOP ALL MOTION
	Vertical Slit Gaps (mm)	Sample Stack	Important Beamline Parameters
	RBV SP:RBV SP	Phi: 0.000 Set: 0.000 deg	RBV SP:RBV SP
Front Panel is	S1VG: 0.500 0.500 0.500		M Theta: 0.530 1.000 1.000 Move
	S2VG: 0.500 0.500 0.500	Psi: 0.000 Set. 0.000 deg	SM Angle: 0.000 0.000 1.000 Move
implemented	S3VG: 0.500 0.500 0.500	Height: 0.000 Set: 0.000 mm	M SM In Beam: IN IN OUT IN Move
	S4VG: 0.040 0.500 0.500	Trans: 0.000 Set: 0.000 mm	Sample In Beam: IN IN OUT IN Move
as CSS OPI	Horizontal Slit Gaps (mm)		
	RBV SP:RBV SP S1HG: 40.000 40.000 40.000		
Scrinting	S2HG: 15.020 30.000 30.000		
Scripting.	S3HG: 0.000 30.000 30.000		
values are	S4HG: 0.000 40.000 40.000	Operation Mode	
1 1 1	Beam Footprint	NR	
exposed like	Sample Length: 200.000		-Server Status-
	For Setpoint For Setpoint RBV For Readback	Polarised NR	Status: OKAY
any other	Resolution: 565.439 Qmin: 0.034		

Layers

Server architecture overview

- Parameters: relevant beamline parameters, relative to the current beam path
- **Components:** Building blocks of the beam path model. Responsible for conversion between beamline parameters and (absolute) motor values
- **Drivers:** Push and read values to/from low-level motor PVs, handle (simple) move synchronization, apply engineering corrections

Interaction between items is coordinated by a toplevel **Beamline** object. Parameters can be unlinked from beam model depending on active **Mode**.

any device variable in IBEX

Inter

Fro

Scri





Example: Processing a move of the supermirror angle

Future Work: Continuous Scanning

Experiment data is stitched from datasets at various θ (incident angle Theta). More datasets means better data, but the overhead of starting/stopping data collection currently limits us to few data points

Ideally, we would like to take data while performing a continuous sweep over θ instead, producing many datasets annotated with real-time positions. θ is a compound axis \rightarrow this requires truly synchronous motion, i.e. axes know and correct where they are in relation to each other in real time.

This is currently limited by our motion controllers, but we are planning to roll-out a more sophisticated Beckhoff-based motion control system at ISIS. These controllers can run synchronization logic on an embedded real-time OS, instead of in Composite Driver layer of Reflectometry Server.