Australian Government

ANSTO

MX3 Beamline: MXCuBE Status Update

Joint MXCuBE-ISPyB Meeting (17/05/2022)

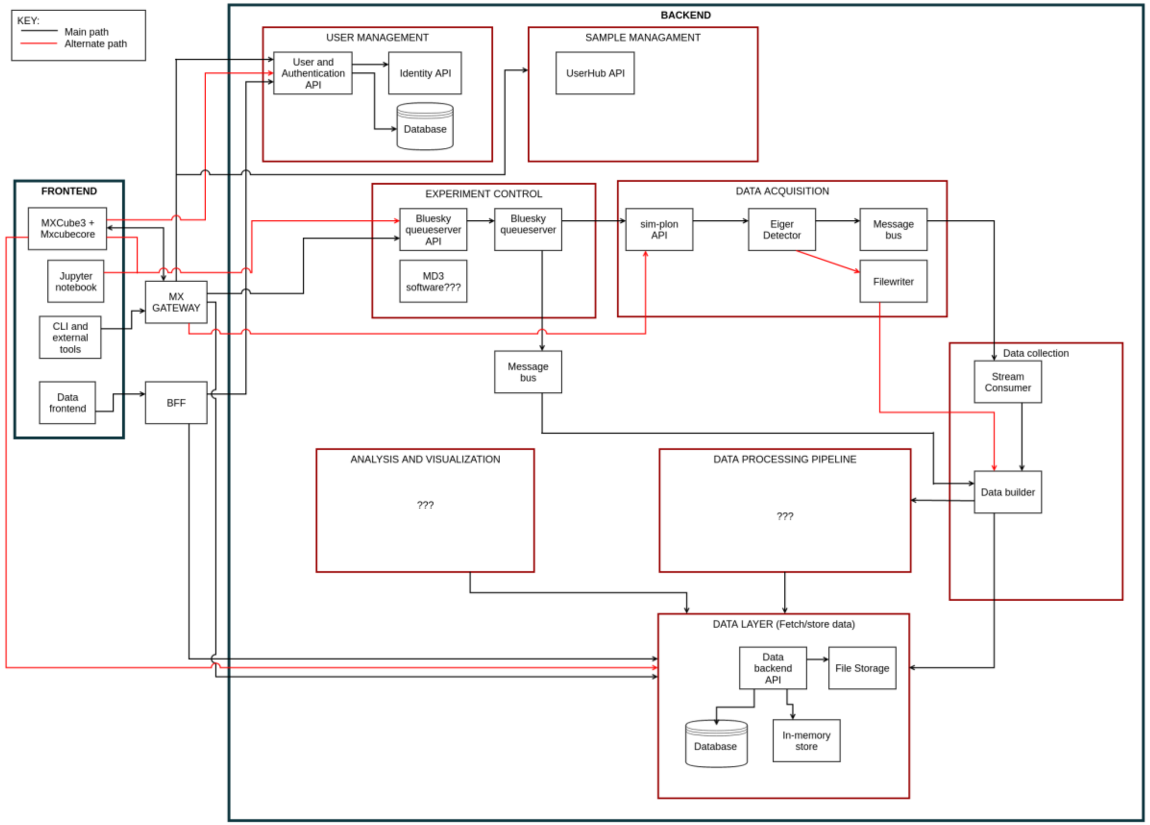
Jacob Oldfield – Scientific Computing (MX3)
Australian Synchrotron

ANSTO

Current Progress

- Drive testrig motors form the MXCuBE frontend
- Connect the testrig camera to MXCuBE
- Execute Bluesky plans from MXCuBE workflows using the Bluesky-queueserver REST API
- Simulate Eiger Detector REST API & StreamV2 ZeroMQ stream interface from HDF5 master files
- Execute "Screen and Collect" and "Raster" workflows via calls to Bluesky plans

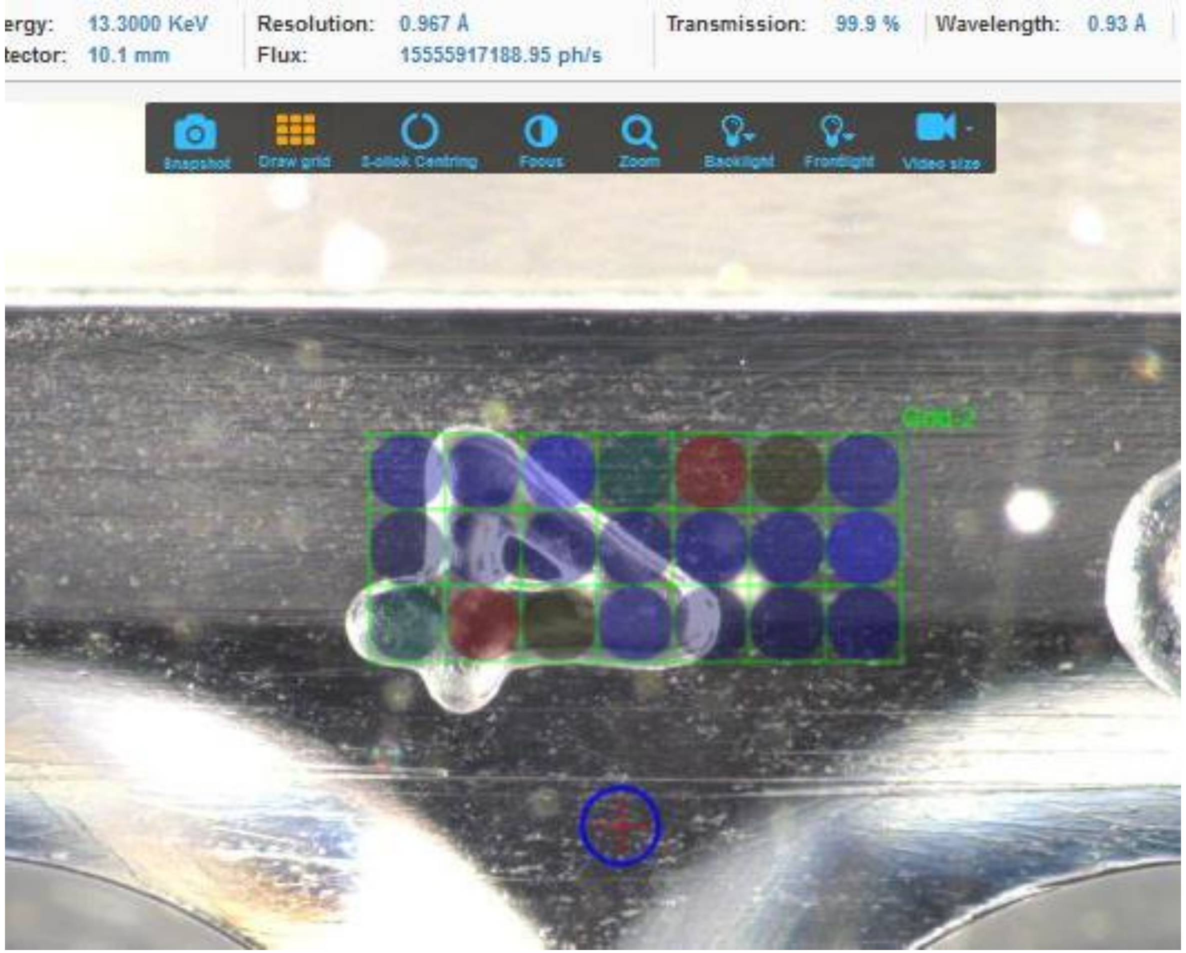
Software Architecture



*Very early draft diagram

- MXCuBE acts as the primary user interface for experiment control
- Secondary frontend will be built for sample changer & results management
- Aim is to do most of the heavy lifting outside of MXCuBE, exposing functionality as RESTful API's as required

Raster Workflow



ergy: 13.3000 KeV
tector: 10.1 mm

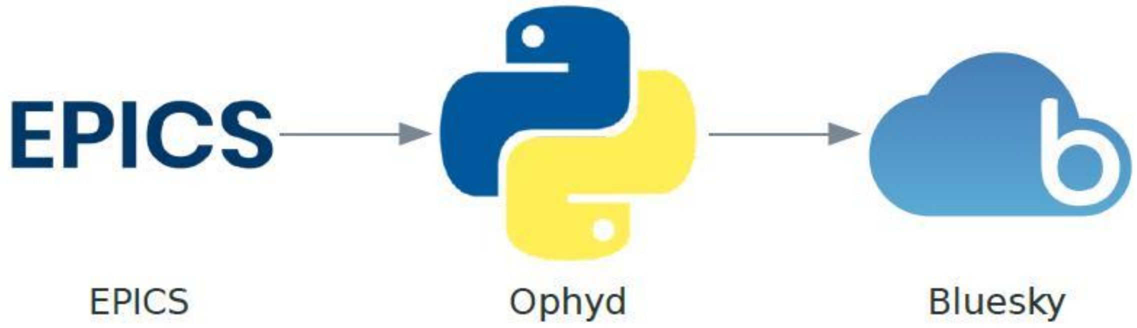
Resolution: 0.967 Å
Flux: 15555917188.95 ph/s

Transmission: 99.9 %

Wavelength: 0.93 Å

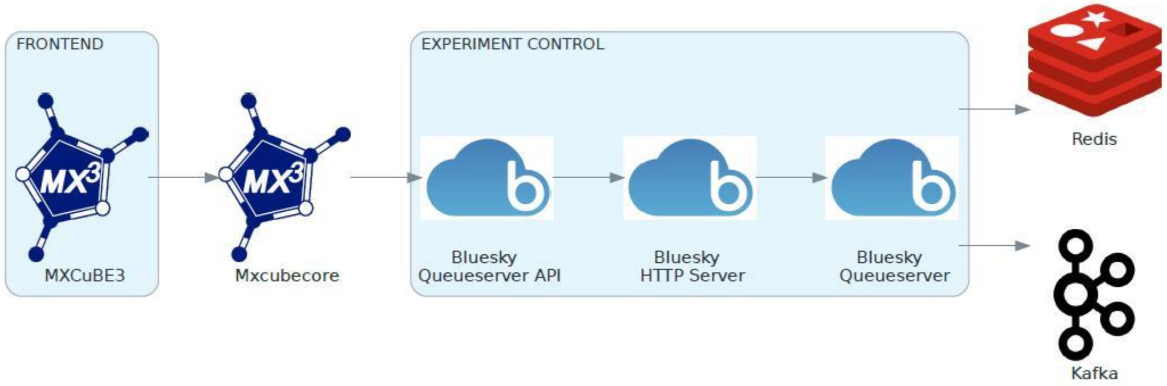
- Workflow calls a Bluesky plan via the Bluesky Queueserver REST API
- Currently the data is being simulated, so it's nonsensical, but provides a proof of concept

Experiment Control



- Use Bluesky plans for experiment orchestration
- Use Ophyd as a hardware abstraction layer
- Bluesky encodes an experimental procedure as a plan

MXCuBE Workflows - Bluesky Plans



- MXCuBE workflow triggers a Bluesky plan
- Bluesky documents are produced and written to a Kafka topic for service consumption
- Metadata recorded during Bluesky plan execution is recorded to Redis

Current Areas of Focus

- Data processing workflows – Dials & RPF
- MXCuBE Kubernetes deployment
- Sample Changer REST API & associated hardware class in MXCuBE
- MXCuBE user account integration with facility services
- Greater integration between MXCuBE hardware classes and Ophyd device definitions