DRAC UI & Micro-frontends

NOBUGS - 27/09/2024

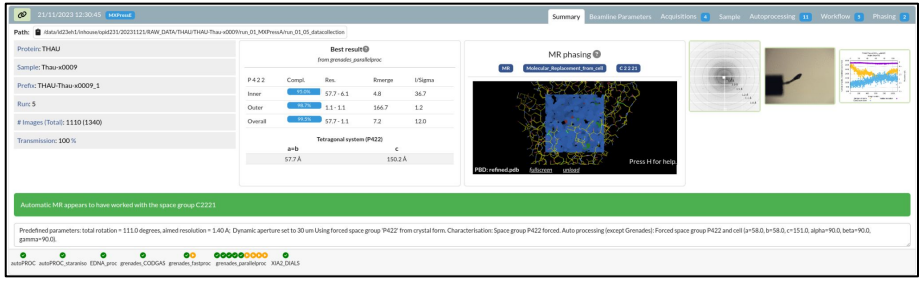
Maël Gaonach
Marjolaine Bodin
Alex De Maria

ESRF
The European Synchrotron

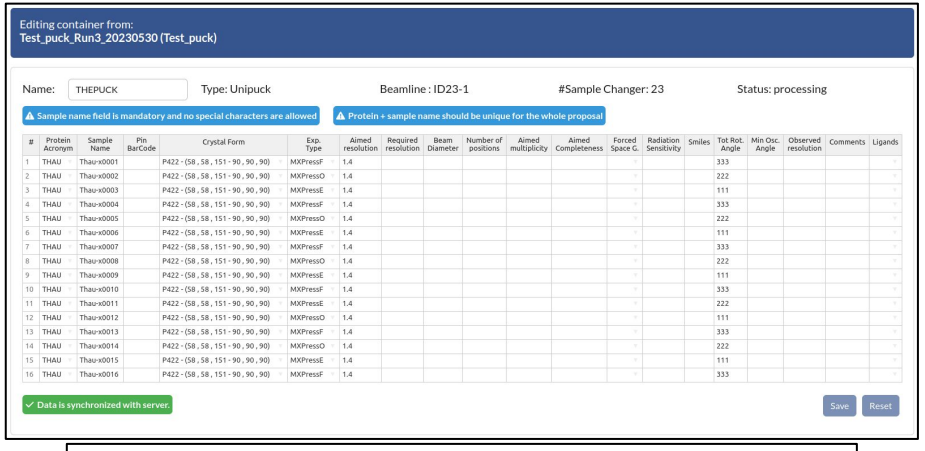
Starting point

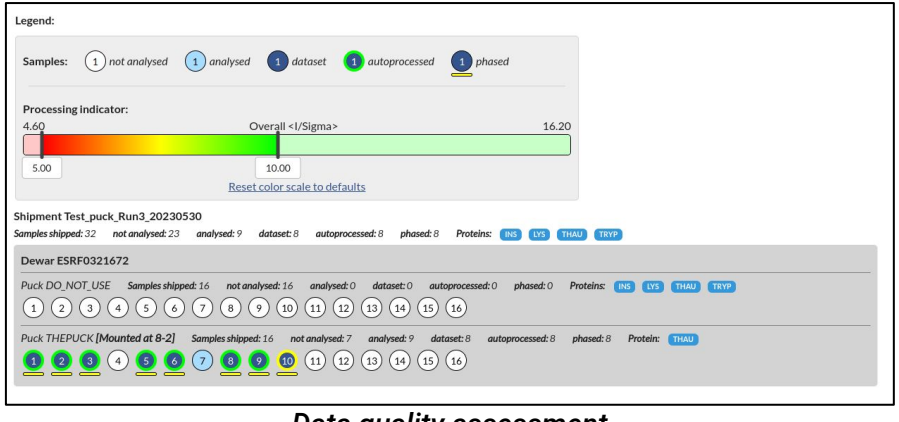
ISPyB

Growing amount of essential features for SB

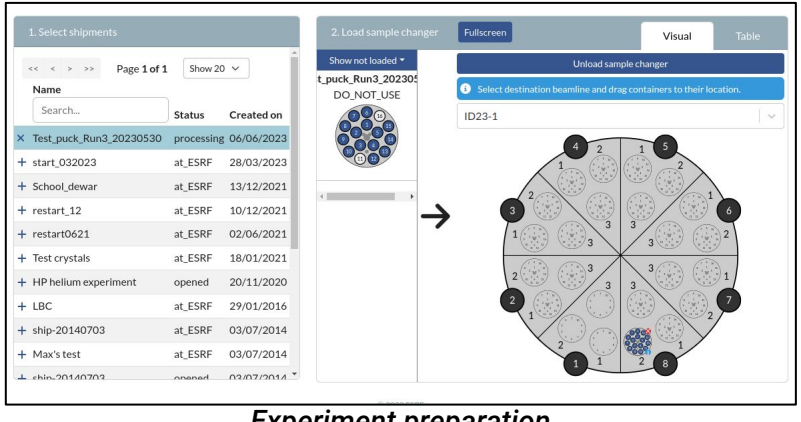


Specialized data visualisation





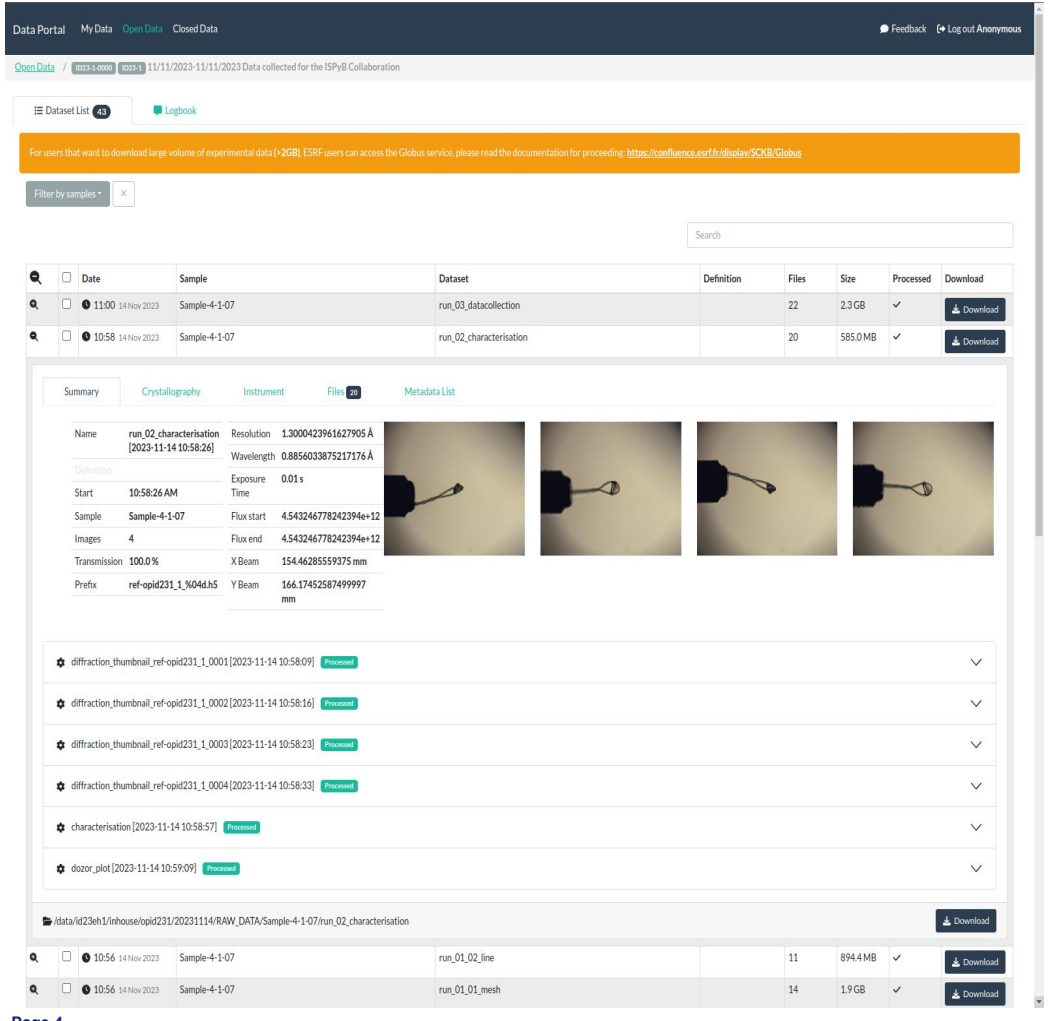
Data quality assessment



Experiment preparation

...BUT isolated from the rest of the facility

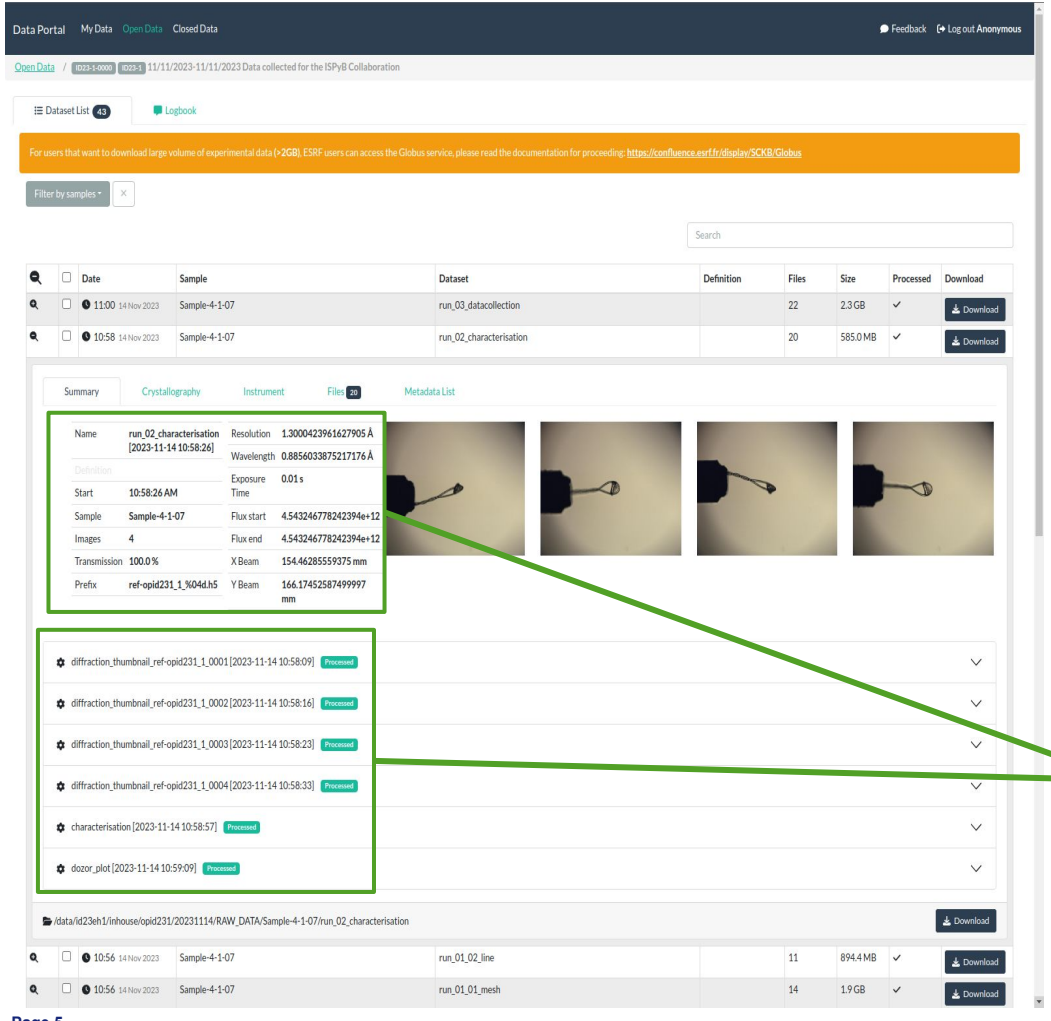
The ESRF Data Portal



+ Tons of essential features for the whole facility (**<u>including SB!</u>**)
  - Data policy implementation (Raw & Processed data)
  - Data download
  - Electronic logbook
  - DOI minting
  - Sample tracking
  - Reprocessing

- Limited data visualization : list of parameters, images…
  - Needs to support wide variety of techniques

The ESRF Data Portal



+ Tons of essential features for the whole facility (**including SB!**)
  - Data policy implementation (Raw & Processed data)
  - Data download
  - Electronic logbook
  - DOI minting
  - Sample tracking
  - Reprocessing

- Limited data visualization : list of parameters, images…
  - Needs to support wide variety of techniques

But all the data we need is already there!

Could we federate these features?

How to manage the diversity?

ISPyB focuses on SB and is already too complex. It would be impossible to manage an all-techniques monolith…

We need to split up things:

Generic features

- Logbook
- Sample tracking

Core features

- User management
- Proposal/session management
- General navigation
- Data fetching
- Generic visualization

Specialized features

- MX visualization
- Cryo EM/ET visualization
- SAXS visualization
- …other techniques

How to manage the diversity?

ISPyB focuses on SB and is already too complex. It would be impossible to manage an all-techniques monolith…

We need to split up things:

Micro Frontends = Independent apps

Generic features

- Logbook
- Sample tracking

Core features

- User management
- Proposal/session management
- General navigation
- Data fetching
- Generic visualization

Specialized features

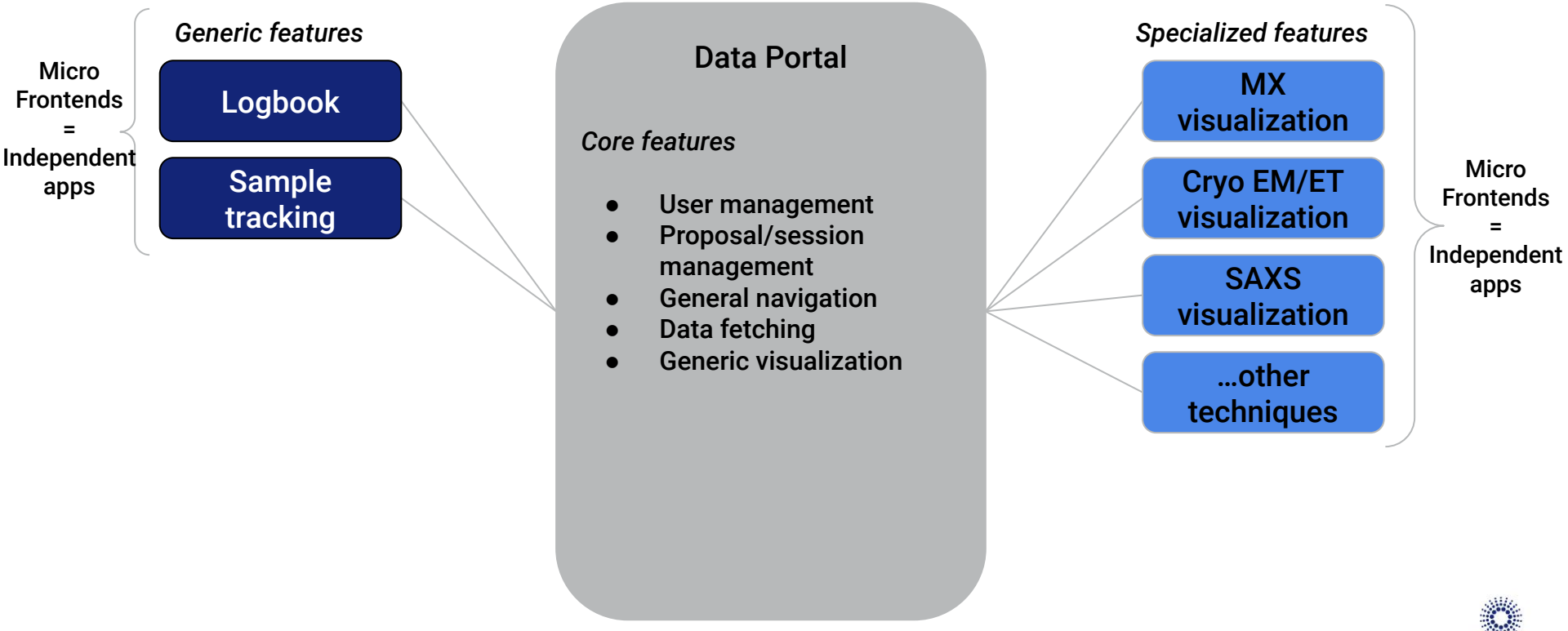
- MX visualization
- Cryo EM/ET visualization
- SAXS visualization
- …other techniques

Micro Frontends = Independent apps

How to manage the diversity?

ISPyB focuses on SB and is already too complex. It would be impossible to manage an all-techniques monolith…

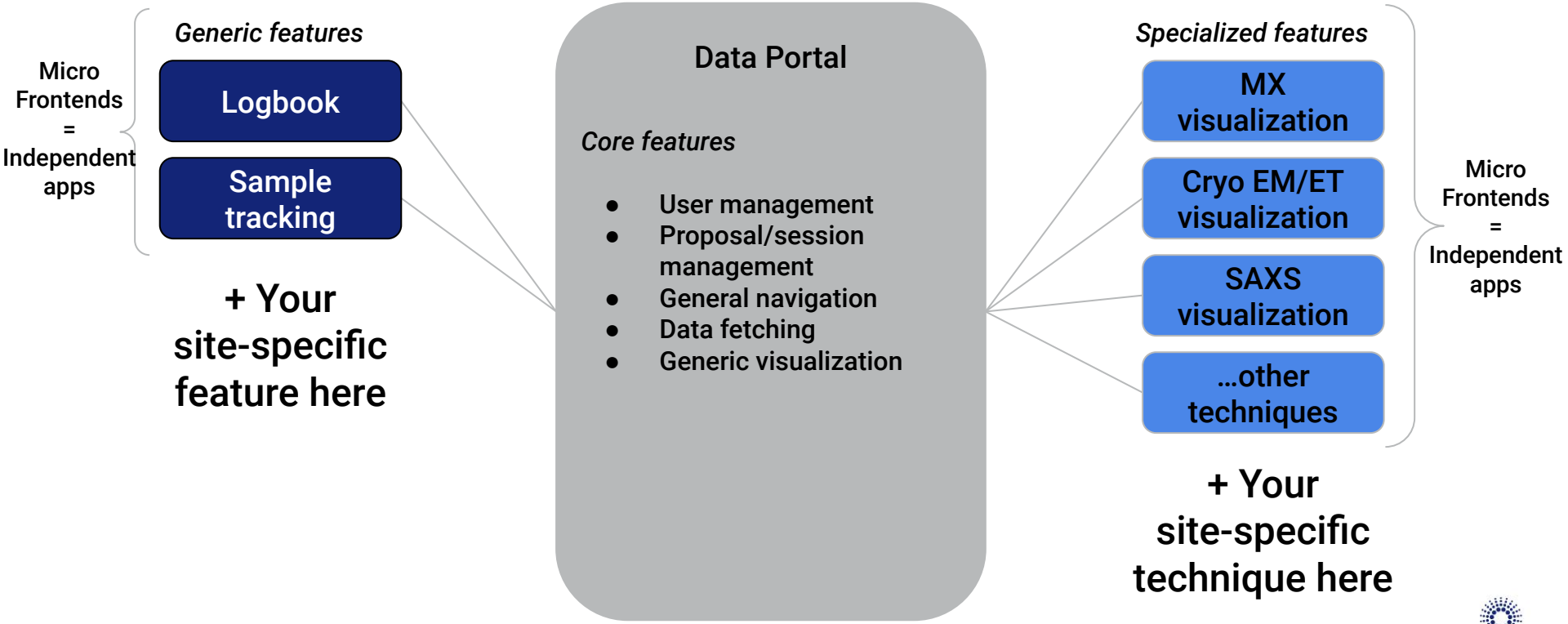
We need to split up things:



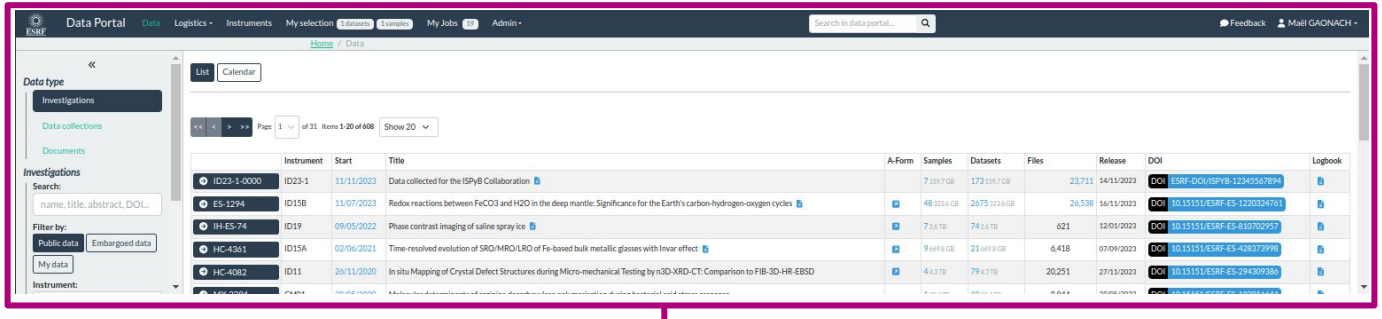
How to manage the diversity?

ISPyB focuses on SB and is already too complex. It would be impossible to manage an all-techniques monolith…

We need to split up things:

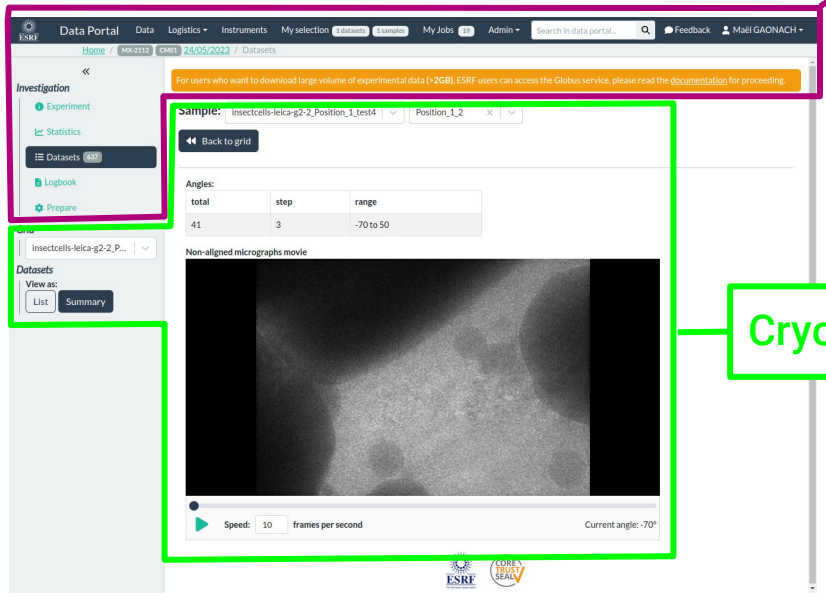


We can load the appropriate app depending on the session…



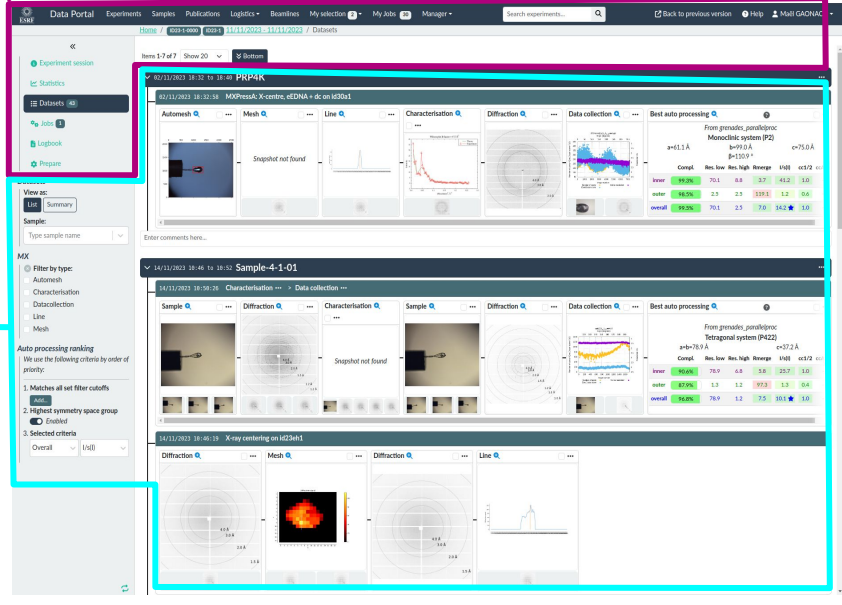
Portal

This includes all the portal features previously mentioned



CryoET

MX

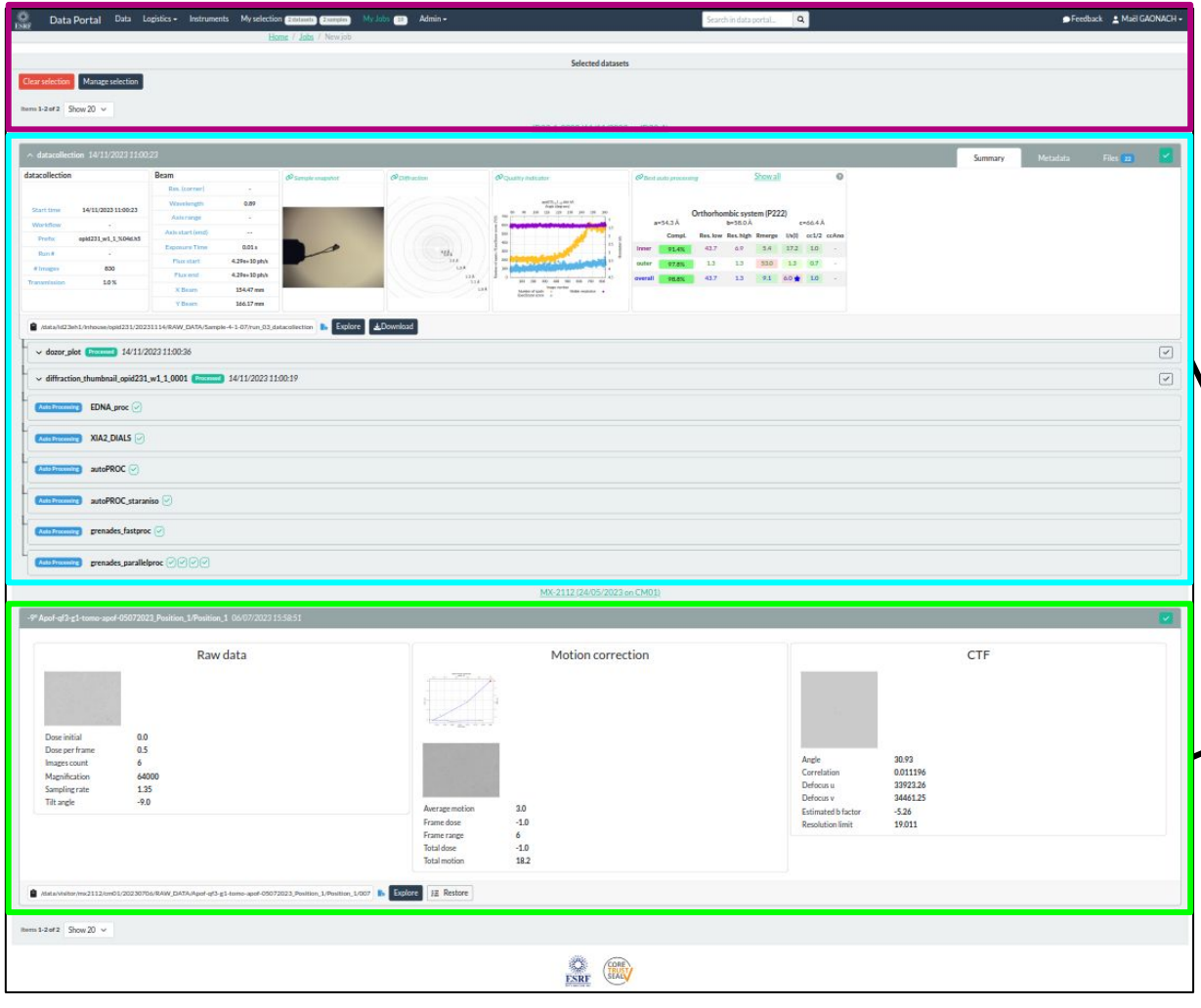


… Or mix them together

portal

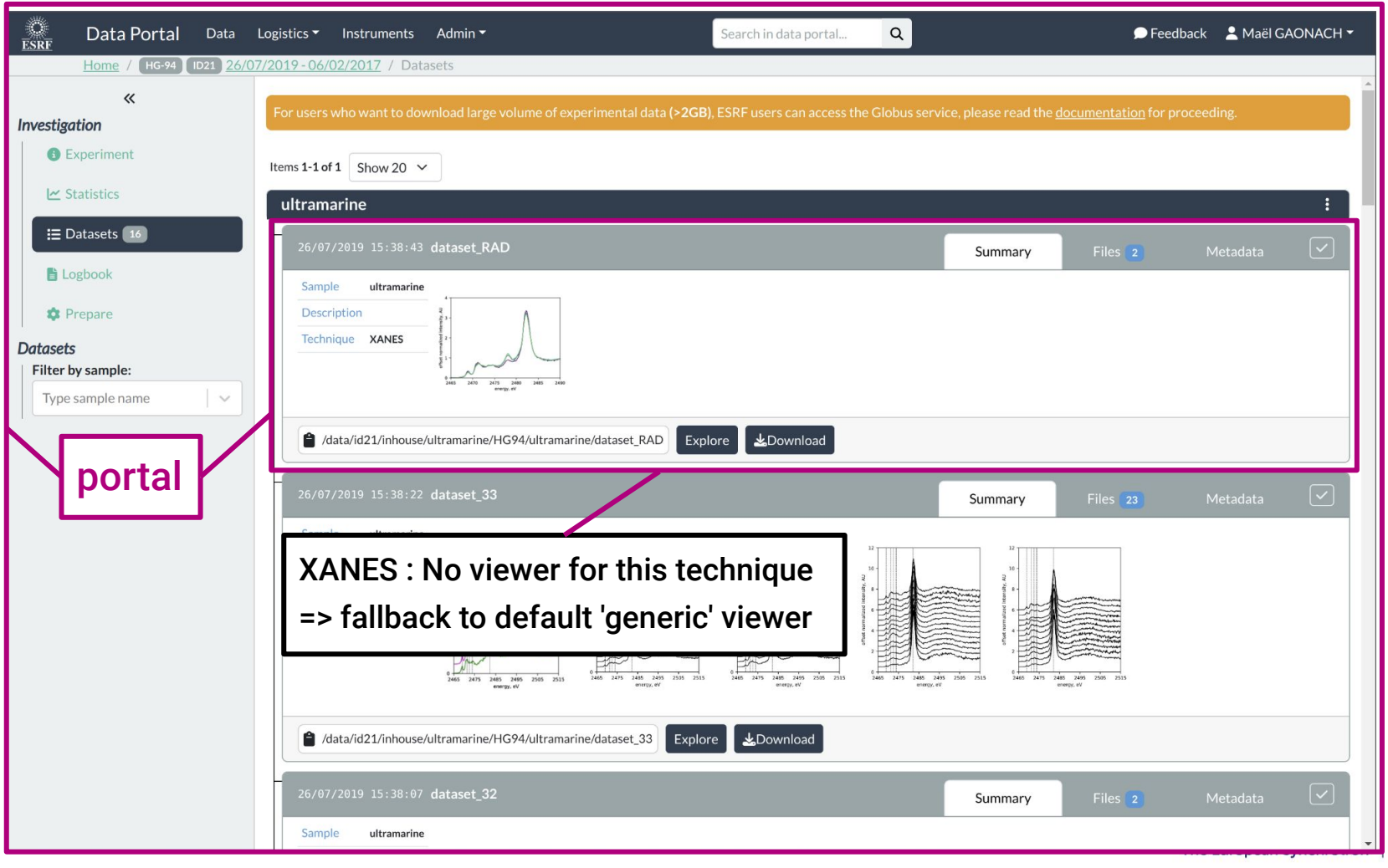
MX

CryoET



These could be the same sample/protein analysed with different techniques on the same page.

Or fallback to a default 'generic' viewer



Implementation

https://gitlab.esrf.fr/icat/data-portal

Switching micro-frontends

https://gitlab.esrf.fr/icat/data-portal

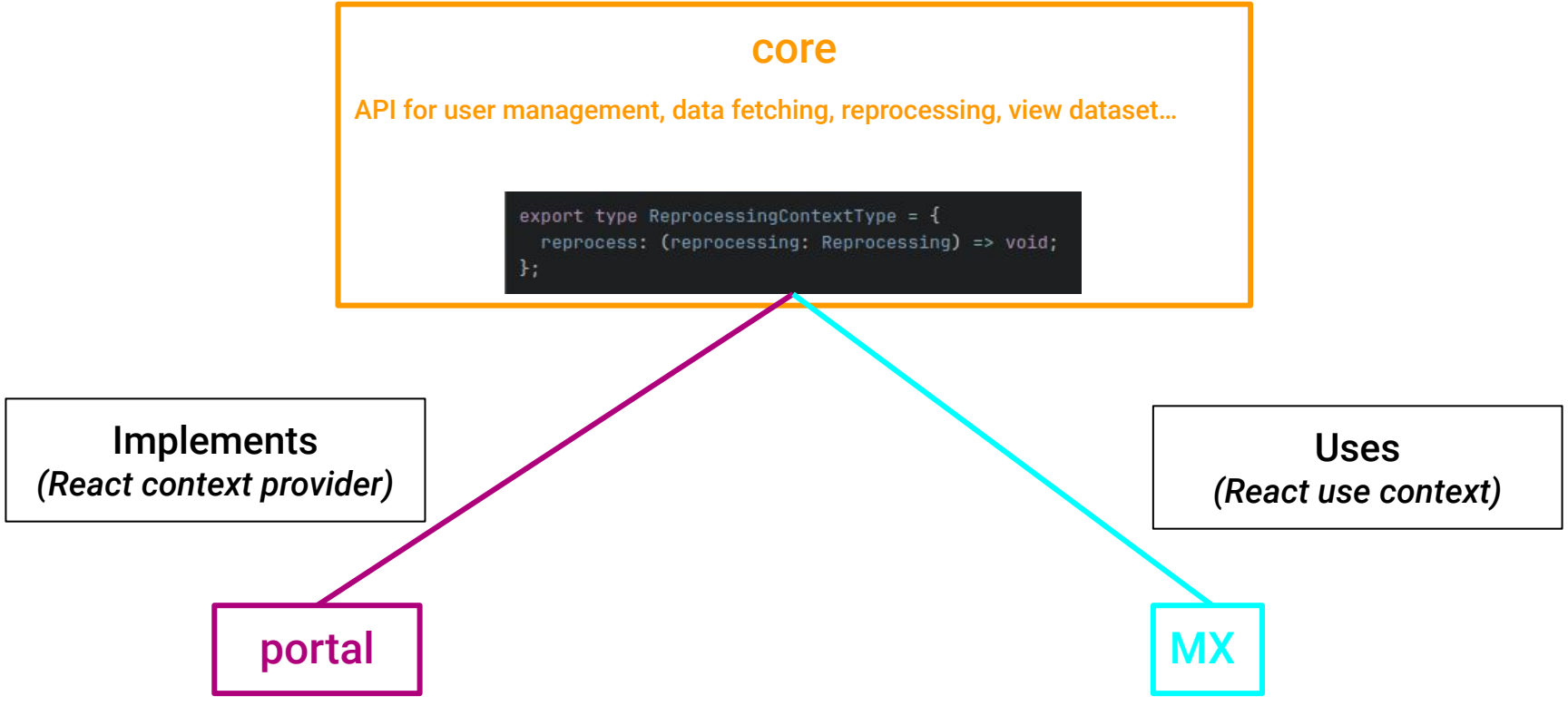
portal

```
export const investigationViewers: InvestigationViewerDefinition[] = [
  {
    date: '*',
    beamline: 'CM01',
    component: (props: any) => (
      <RemoteComponent component={CryoETInvestigationViewer} {...props} />
    ),
  },
  {
    date: '24/10/2023',
    beamline: /^(ID23.*|ID30.*)$/,
    component: (props: any) => (
      <RemoteComponent component={MXInvestigationViewer} {...props} />
    ),
  },
  {
    date: '*',
    beamline: 'BM29',
    component: (props: any) => <GenericInvestigationViewer {...props} />,
  },
```

```
remotes: {
  sessionViewer: process.env.VITE_FEDERATION_SESSION_VIEWER_URL,
  logistics: process.env.VITE_FEDERATION_LOGISTICS_URL,
  cryoet: process.env.VITE_FEDERATION_CRYOET_URL,
  mx: process.env.VITE_FEDERATION_MX_URL,
  remoteDatasetViewerApp:
    process.env.VITE_FEDERATION_DATASET_VIEWER_URL,
  logbook: process.env.VITE_FEDERATION_LOGBOOK_URL,
},
```

Micro-frontends are independents but can work together

https://gitlab.esrf.fr/icat/data-portal



Thank you!