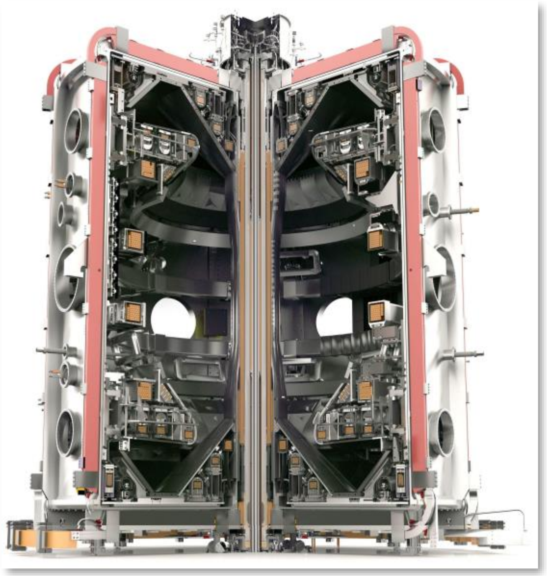
UK Atomic Energy Authority

# Towards Making Fusion Data Open and FAIR with IRIS Resources

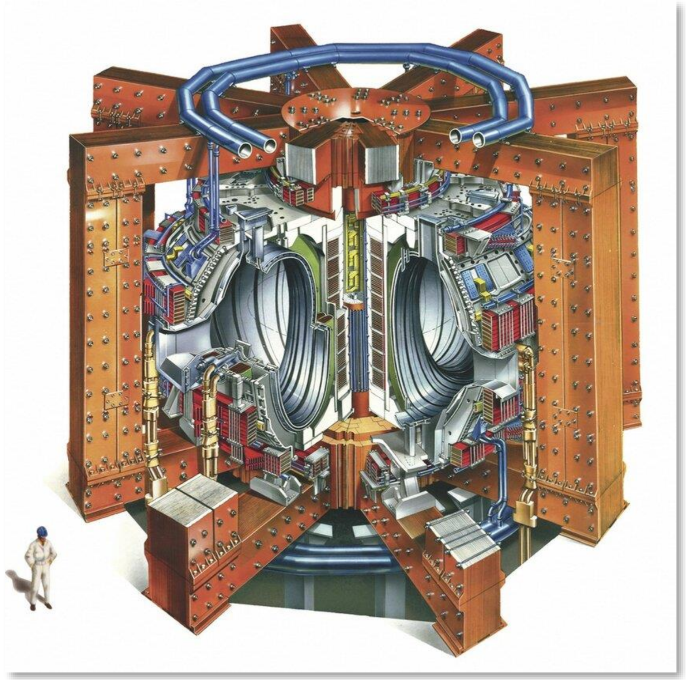
Samuel Jackson

# UKAEA

- **UKAEA: United Kingdon Atomic Energy Authority**
- Responsible for the national fusion energy research programme
- Magnetic confinement fusion though tokamak experiments
- Development of support facilities to develop UK’s fusion industry and supply chain



MAST-U

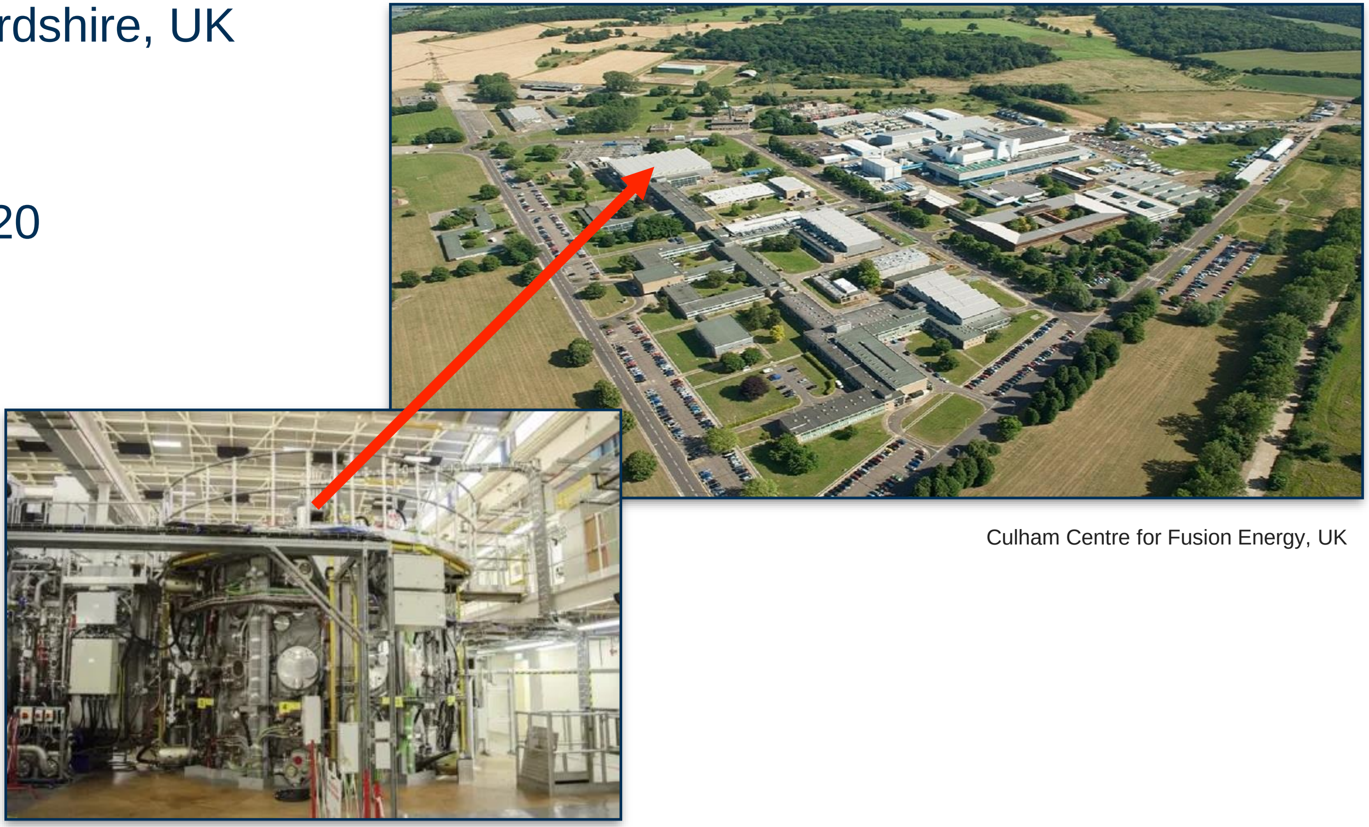


JET

JET’s record-breaking shot
69.26 Megajoules!

# MAST

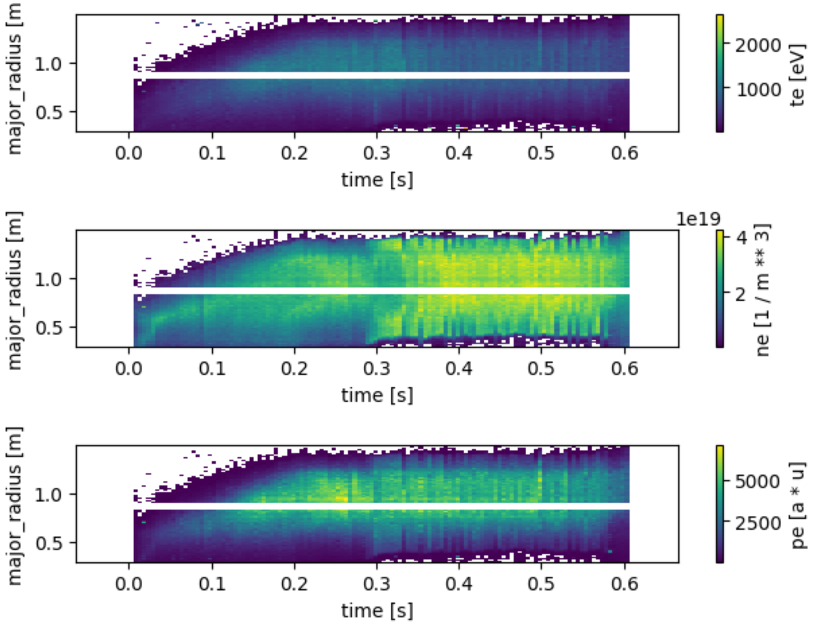
- **MAST (Mega Ampere Spherical Tokamak)**
- Spherical tokamak design commissioned by EURATOM/UKAEA
- Built at Culham Centre for Fusion Energy, Oxfordshire, UK
- Experiments ran from 1999 through to 2013
- Produced ~30,000 shots over its history
- Succeeded by MAST Upgrade (MAST-U) in 2020



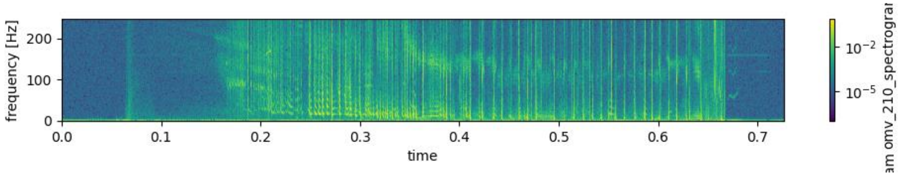
Culham Centre for Fusion Energy, UK

MAST-U Tokamak

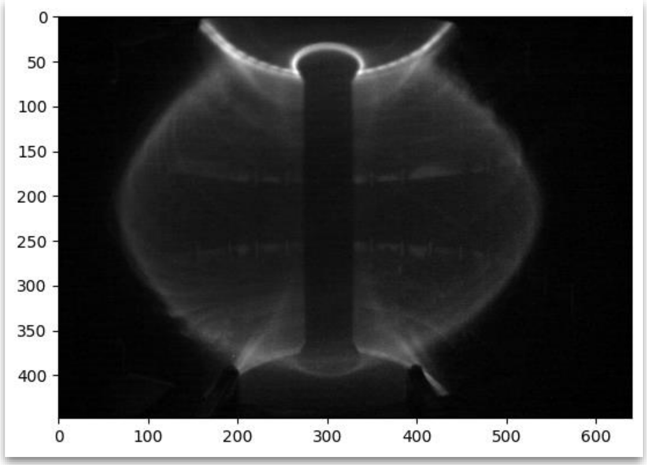
# MAST Data



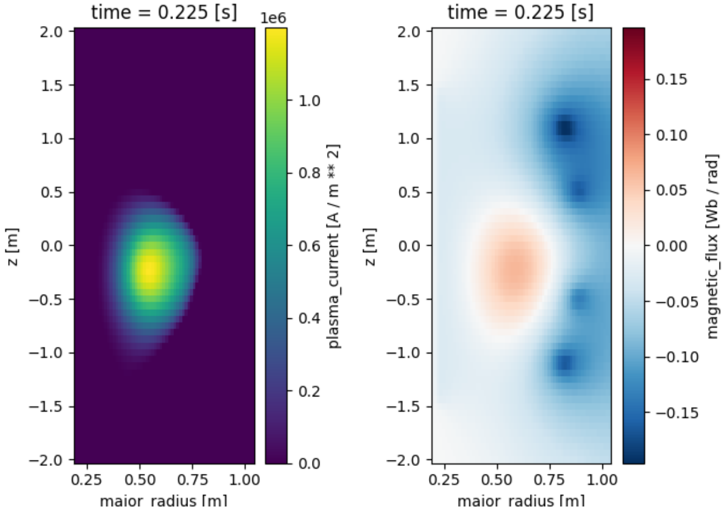
2D Profiles: Thomson Scattering Data



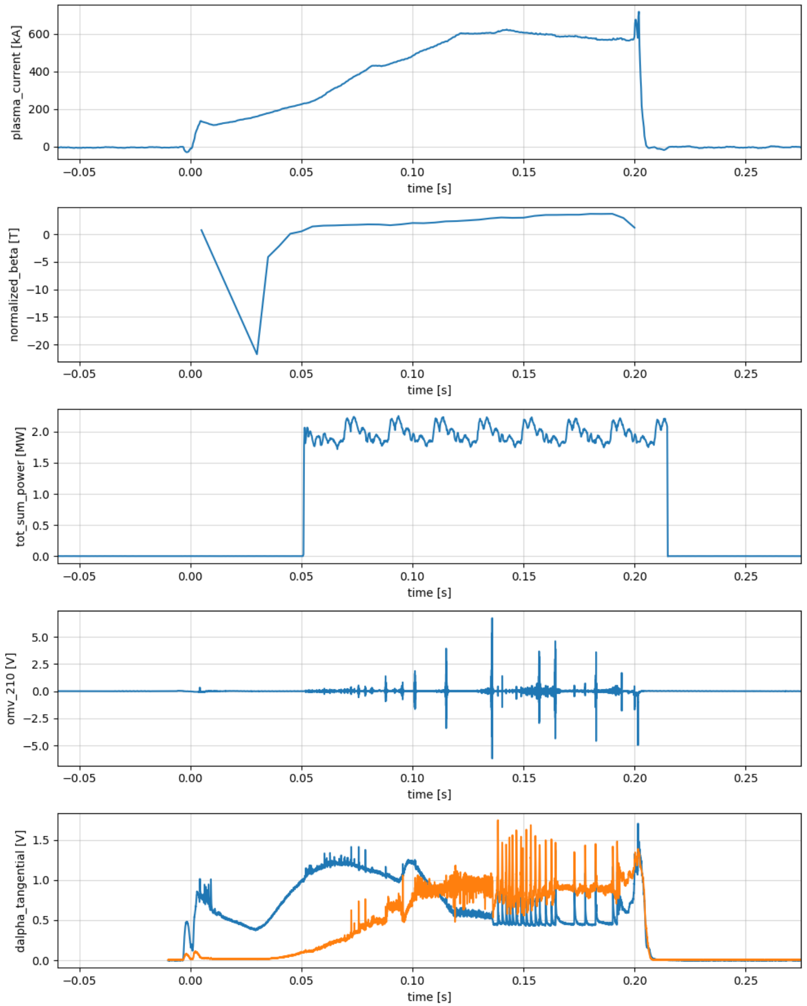
2D Spectrogram: Mirnov Coil Spectrograms



2D+t Video: Centre Column Camera data



3D profiles: Equilibrium Reconstruction Data



1D Time Traces

# FAIR MAST Project

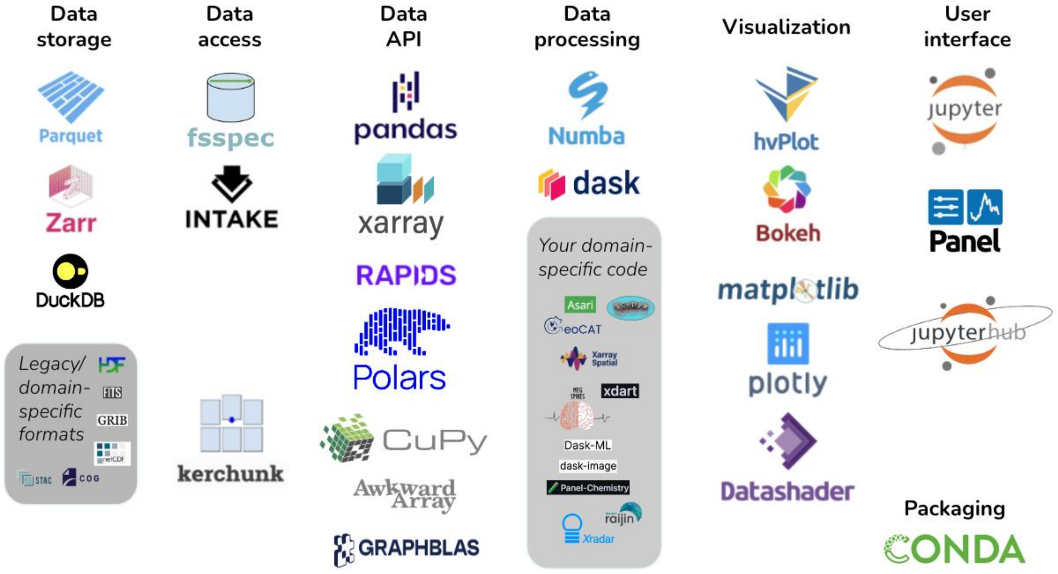
***Challenges:***

- Data were only accessible from an internal cluster
  - Must be on UKAEA network or whitelisted IP
  - Must have account access to internal cluster
- Data were only accessible though domain specific access layer (UDA)
- Data had inconsistent naming, dimensions, units
- Data had significant amount of redundancy
- Data representation not interoperable with common analysis tools

***Goal**: “To produce a framework for public access to MAST data in a FAIR (Findable, Accessible, Interoperable, and Reusable) manner”.*

- Data must be easily **findable** through the metadata
- Data must be in exposed in an **interoperable** format
- Minimise **loading** and **transferring** data (lazy loading)
- Support **data analysis** and **ML/AI** frameworks
- Support larger-than-memory & parallel computation
- Be **publicly** accessible

## Pandata Stack

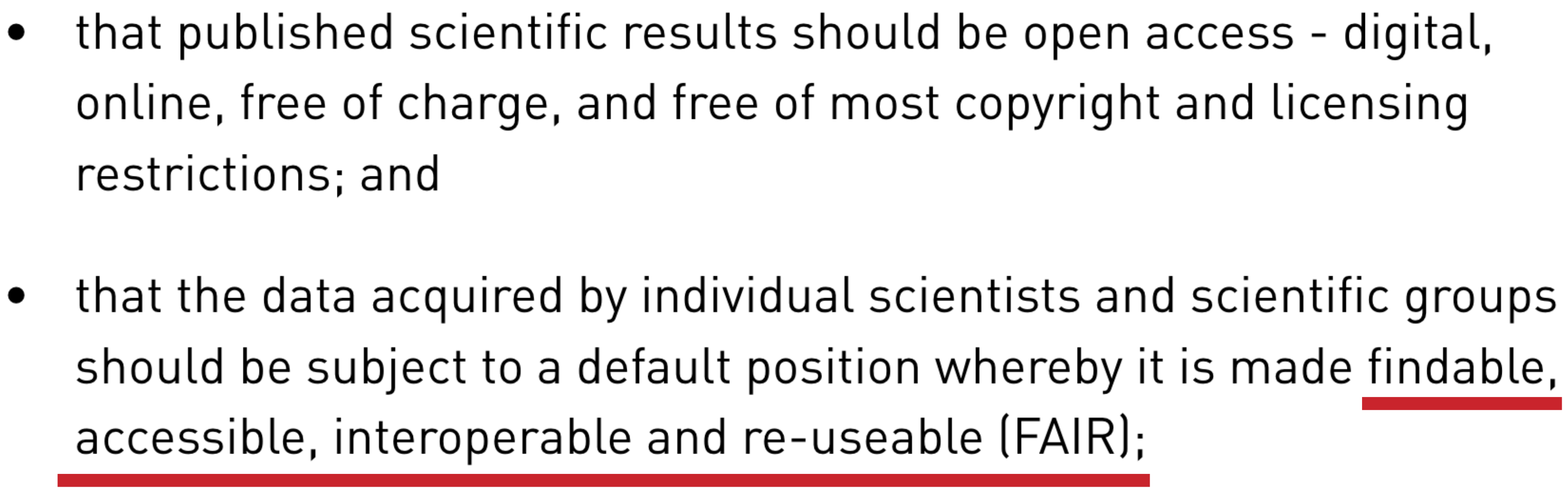


[Bednar J.A., Durant M. The Pandata Scalable Open-Source Analysis Stack.](#)

# Motivation

**Because our funders tell us too…**

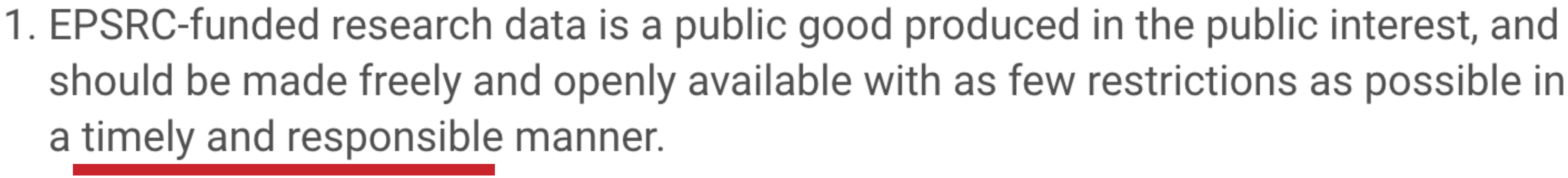
UKRI Open Research Data Taskforce:



- that published scientific results should be open access - digital, online, free of charge, and free of most copyright and licensing restrictions; and

- that the data acquired by individual scientists and scientific groups should be subject to a default position whereby it is made findable, accessible, interoperable and re-useable (FAIR);

EPSRC Research Data Policy:

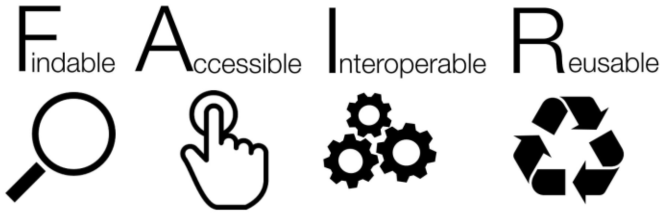


1. EPSRC-funded research data is a public good produced in the public interest, and should be made freely and openly available with as few restrictions as possible in a timely and responsible manner.

Final Report of the UKRI Open Research Task Force
EPSRC Data Policy

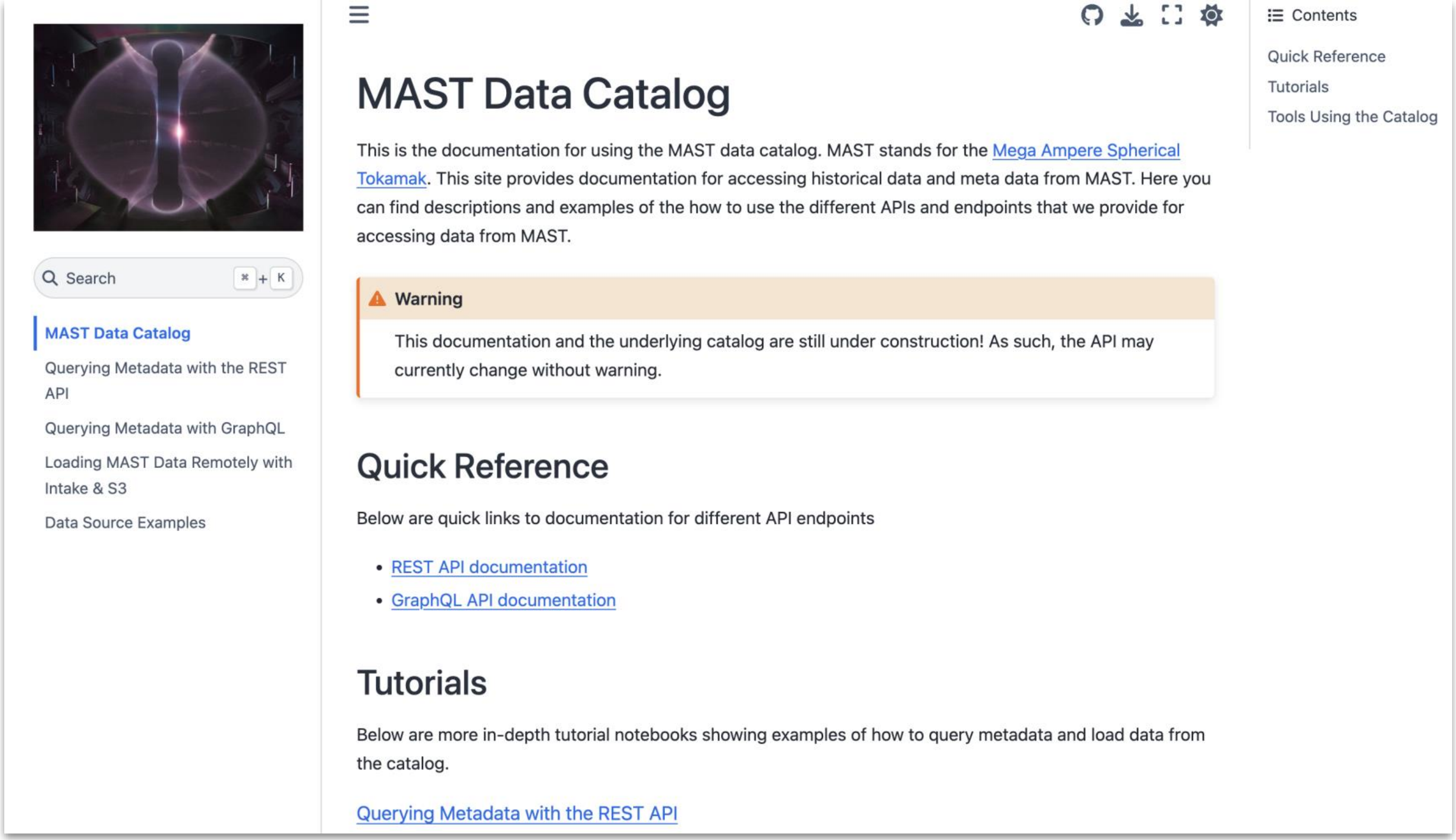
# FAIR Data

- **Findable -** Metadata and data should be easy to find for both humans and computers
- **Accessible** - It should be clear how to access the data once found.
- **Interoperable -** Data can be integrated with other data and interoperate with applications or workflows for analysis, storage, and processing.
- **Reusable -** Metadata and data should be well-described so that they can be replicated and/or combined in different settings.



GO FAIR: [https://www.go-fair.org/fair-principles/](https://www.go-fair.org/fair-principles/)
Wilkinson, M. D. et al. The FAIR Guiding Principles for scientific data management and stewardship. (2016).
Strand, P. et al. A FAIR based approach to data sharing in Europe. (2022).

# FAIR MAST Project


Search
⌘ + K
MAST Data Catalog
Querying Metadata with the REST API
Querying Metadata with GraphQL
Loading MAST Data Remotely with Intake & S3
Data Source Examples
Contents
Quick Reference
Tutorials
Tools Using the Catalog

## MAST Data Catalog

This is the documentation for using the MAST data catalog. MAST stands for the Mega Ampere Spherical Tokamak. This site provides documentation for accessing historical data and meta data from MAST. Here you can find descriptions and examples of the how to use the different APIs and endpoints that we provide for accessing data from MAST.

> ⚠ **Warning**
>
> This documentation and the underlying catalog are still under construction! As such, the API may currently change without warning.

### Quick Reference

Below are quick links to documentation for different API endpoints

- REST API documentation
- GraphQL API documentation

### Tutorials

Below are more in-depth tutorial notebooks showing examples of how to query metadata and load data from the catalog.

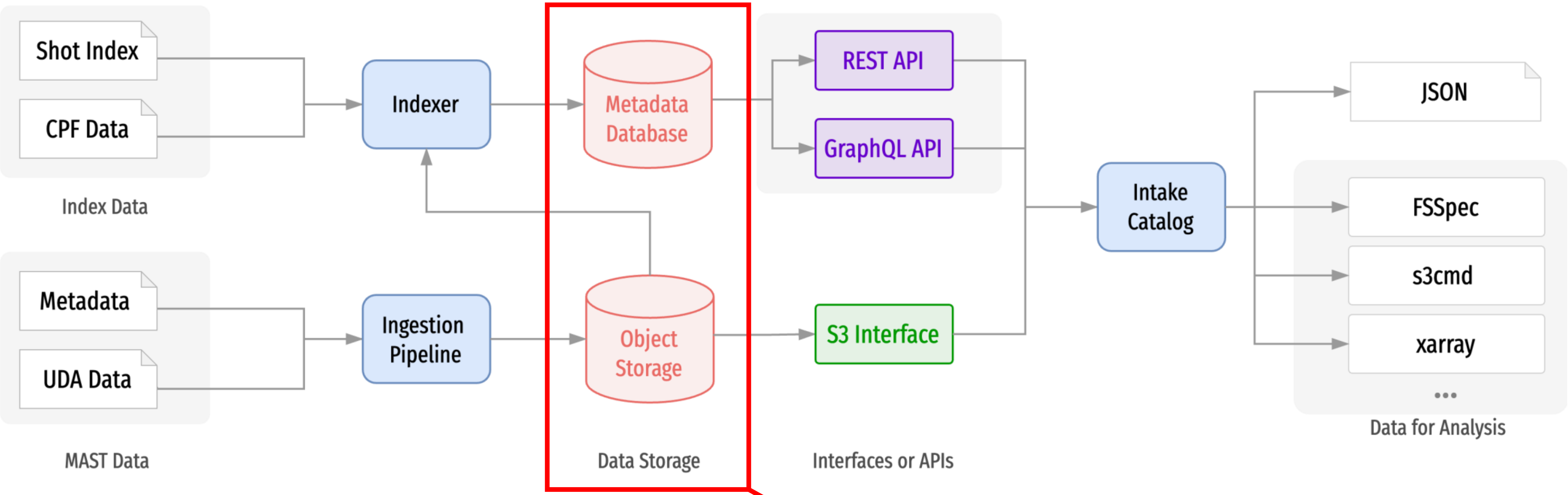
Querying Metadata with the REST API

https:// mastapp.site/

Hosted by IRIS on STFC Cloud

Jackson, Samuel, et al. "FAIR-MAST: A fusion device data management system." *SoftwareX* 27 (2024): 101869.

# System Architecture



- **Object storage**
  - Holding shot, source, and signal data in a self-describing, cloud optimised file format.
  - Accessible by S3 protocol.

- **Metadata database**
  - Indexing data in the object storage
  - For searching and finding data in the object storage
  - Accessible by web APIs

Jackson, Samuel, et al. "FAIR-MAST: A fusion device data management system." *SoftwareX* 27 (2024): 101869.

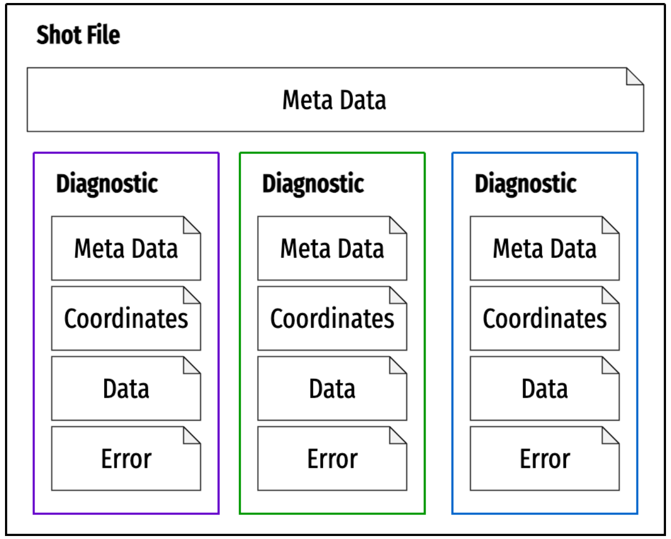
# File Format

We choose to use a hierarchical self-describing file format.

- Group data by shot
- Group signals by diagnostic
- Each group may contain metadata
- Coordinate axes are also defined

For our implementation we choose Zarr format

- Hierarchical format
- HDF-like interface
- Consolidated metadata
- Parallel read/write
- Cloud optimised
- Interoperable with different languages
- Lazy loading



Shot File

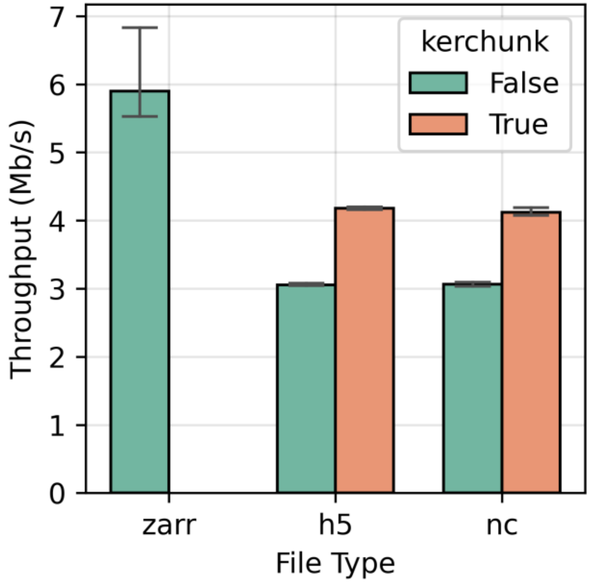
Meta Data

Diagnostic
- Meta Data
- Coordinates
- Data
- Error

Diagnostic
- Meta Data
- Coordinates
- Data
- Error

Diagnostic
- Meta Data
- Coordinates
- Data
- Error

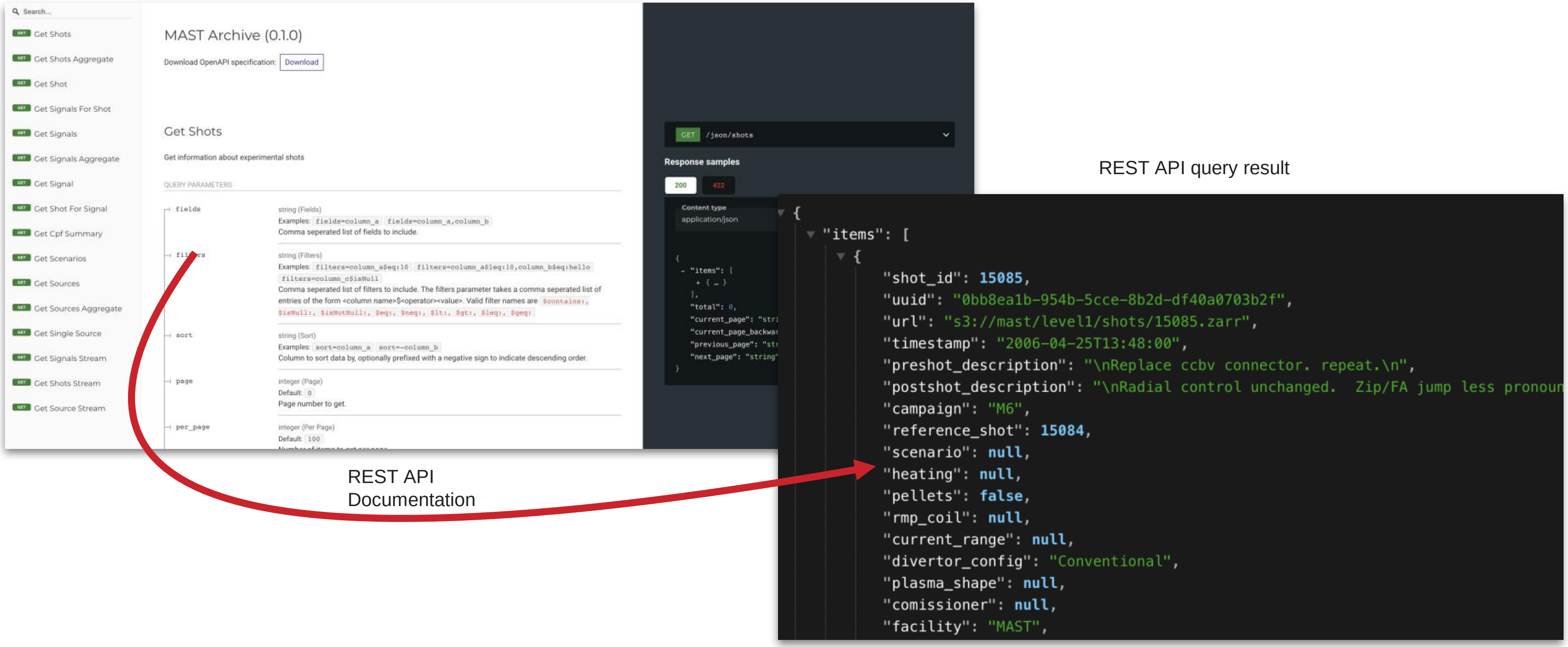
Above: File format structure



Above: Performance comparison of Zarr/NetCDF/HDF with and without Kerchunk RBB camera data.

Zarr File Format

# Metadata APIs: REST



Search...

- GET Get Shots
- GET Get Shots Aggregate
- GET Get Shot
- GET Get Signals For Shot
- GET Get Signals
- GET Get Signals Aggregate
- GET Get Signal
- GET Get Shot For Signal
- GET Get Cpf Summary
- GET Get Scenarios
- GET Get Sources
- GET Get Sources Aggregate
- GET Get Single Source
- GET Get Signals Stream
- GET Get Shots Stream
- GET Get Source Stream

MAST Archive (0.1.0)

Download OpenAPI specification: Download

Get Shots

Get information about experimental shots

QUERY PARAMETERS

| | |
|---|---|
| `fields` | string (Fields)<br>Examples: `fields=column_a` `fields=column_a,column_b`<br>Comma seperated list of fields to include. |
| `filters` | string (Filters)<br>Examples: `filters=column_a$eq:10` `filters=column_a$leq:10,column_b$eq:hello` `filters=column_c$isNull`<br>Comma seperated list of filters to include. The filters parameter takes a comma seperated list of entries of the form <column name>$<operator><value>. Valid filter names are `$contains:, $isNull:, $isNotNull:, $eq:, $neq:, $lt:, $gt:, $leq:, $geq:` |
| `sort` | string (Sort)<br>Examples: `sort=column_a` `sort=-column_b`<br>Column to sort data by, optionally prefixed with a negative sign to indicate descending order. |
| `page` | integer (Page)<br>Default: `0`<br>Page number to get. |
| `per_page` | integer (Per Page)<br>Default: `100` |

GET /json/shots

Response samples

200 422

Content type
application/json

```
{
  - "items": [
      + { … }
    ],
    "total": 0,
    "current_page": "stri
    "current_page_backwar
    "previous_page": "str
    "next_page": "string"
}
```

REST API Documentation

REST API query result

```
{
  "items": [
    {
      "shot_id": 15085,
      "uuid": "0bb8ea1b-954b-5cce-8b2d-df40a0703b2f",
      "url": "s3://mast/level1/shots/15085.zarr",
      "timestamp": "2006-04-25T13:48:00",
      "preshot_description": "\nReplace ccbv connector. repeat.\n",
      "postshot_description": "\nRadial control unchanged.  Zip/FA jump less pronoun
      "campaign": "M6",
      "reference_shot": 15084,
      "scenario": null,
      "heating": null,
      "pellets": false,
      "rmp_coil": null,
      "current_range": null,
      "divertor_config": "Conventional",
      "plasma_shape": null,
      "comissioner": null,
      "facility": "MAST",
```

REST API implemented with `fastapi`, `sqlmodel`, and `sqlalchemy`

Hosted on IRIS through STFC Cloud

# Metadata Querying

## Querying for shot metadata using the JSON API

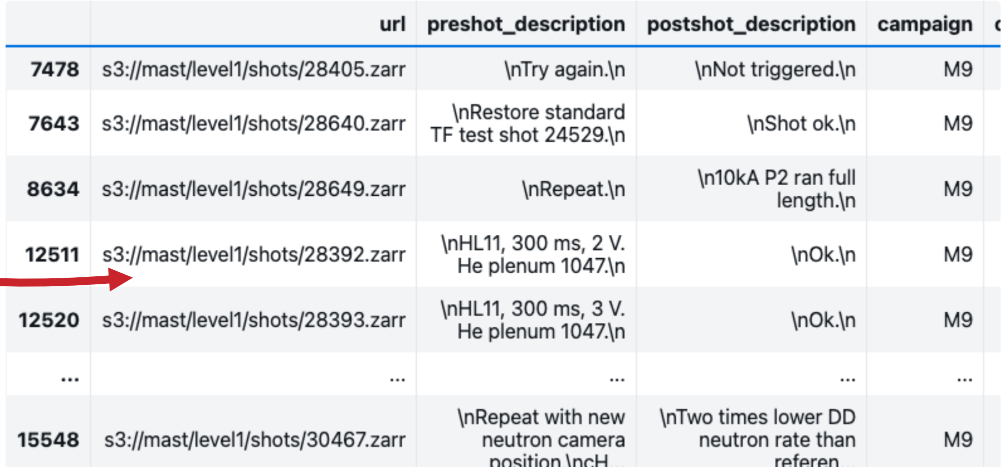
```
import requests
import pandas as pd

response = requests.get('https://mastapp.site/json/shots?filters=campaign$eq:M9')
items = response.json()['items’]
Summary = pd.DataFrame(items)
```

## Querying for shot metadata using Intake library to directly get a pandas DataFrame

```
import intake
import pandas as pd

catalog = intake.open_catalog(f'https://mastapp.site/intake/catalog.yml')
df = pd.DataFrame(catalog.index.level1.shots().read())
summary = df.loc[df.campaign == 'M9']
summary
```



| | url | preshot_description | postshot_description | campaign |
|---|---|---|---|---|
| **7478** | s3://mast/level1/shots/28405.zarr | \nTry again.\n | \nNot triggered.\n | M9 |
| **7643** | s3://mast/level1/shots/28640.zarr | \nRestore standard TF test shot 24529.\n | \nShot ok.\n | M9 |
| **8634** | s3://mast/level1/shots/28649.zarr | \nRepeat.\n | \n10kA P2 ran full length.\n | M9 |
| **12511** | s3://mast/level1/shots/28392.zarr | \nHL11, 300 ms, 2 V. He plenum 1047.\n | \nOk.\n | M9 |
| **12520** | s3://mast/level1/shots/28393.zarr | \nHL11, 300 ms, 3 V. He plenum 1047.\n | \nOk.\n | M9 |
| **...** | ... | ... | ... | ... |
| **15548** | s3://mast/level1/shots/30467.zarr | \nRepeat with new neutron camera position.\ncH | \nTwo times lower DD neutron rate than referen | M9 |

Hosted by IRIS on STFC Cloud

# Data Access: Xarray, Dask, S3

Loading MAST data in **2 lines of code**:

```
import xarray as xr
dataset = xr.open_zarr("https://s3.echo.stfc.ac.uk/mast/level1/shots/30420.zarr/amc")
```

A more explicit example with S3:

```
import s3fs
import xarray as xr
import matplotlib.pyplot as plt

# s3 storage location
endpoint_url = 'https://s3.echo.stfc.ac.uk'
# URL of data we want to load
url = 's3://mast/level1/shots/30420.zarr/amc'

# fsspec handle to remote file system
s3 = s3fs.S3FileSystem(anon=True, endpoint_url=endpoint_url)

# open the dataset
dataset = xr.open_zarr(s3.get_mapper(url))

# data only loaded at this point!
plt.plot(dataset['time'], dataset[‘plasma_current'])
```

Hosted by IRIS on
STFC Echo Storage!

# User Access: Bulk Download

UK Atomic Energy Authority

Bulk download of data can be done using your favorite S3 command line tool.
For example, `s5cmd` is a fast parallel transfer tool.

**Download one whole shot**

```
s5cmd --no-sign-request -—endpoint-url https://s3.echo.stfc.ac.uk \
  cp "s3://mast/level1/shots/30420.zarr/*" ./data/30420.zarr
```

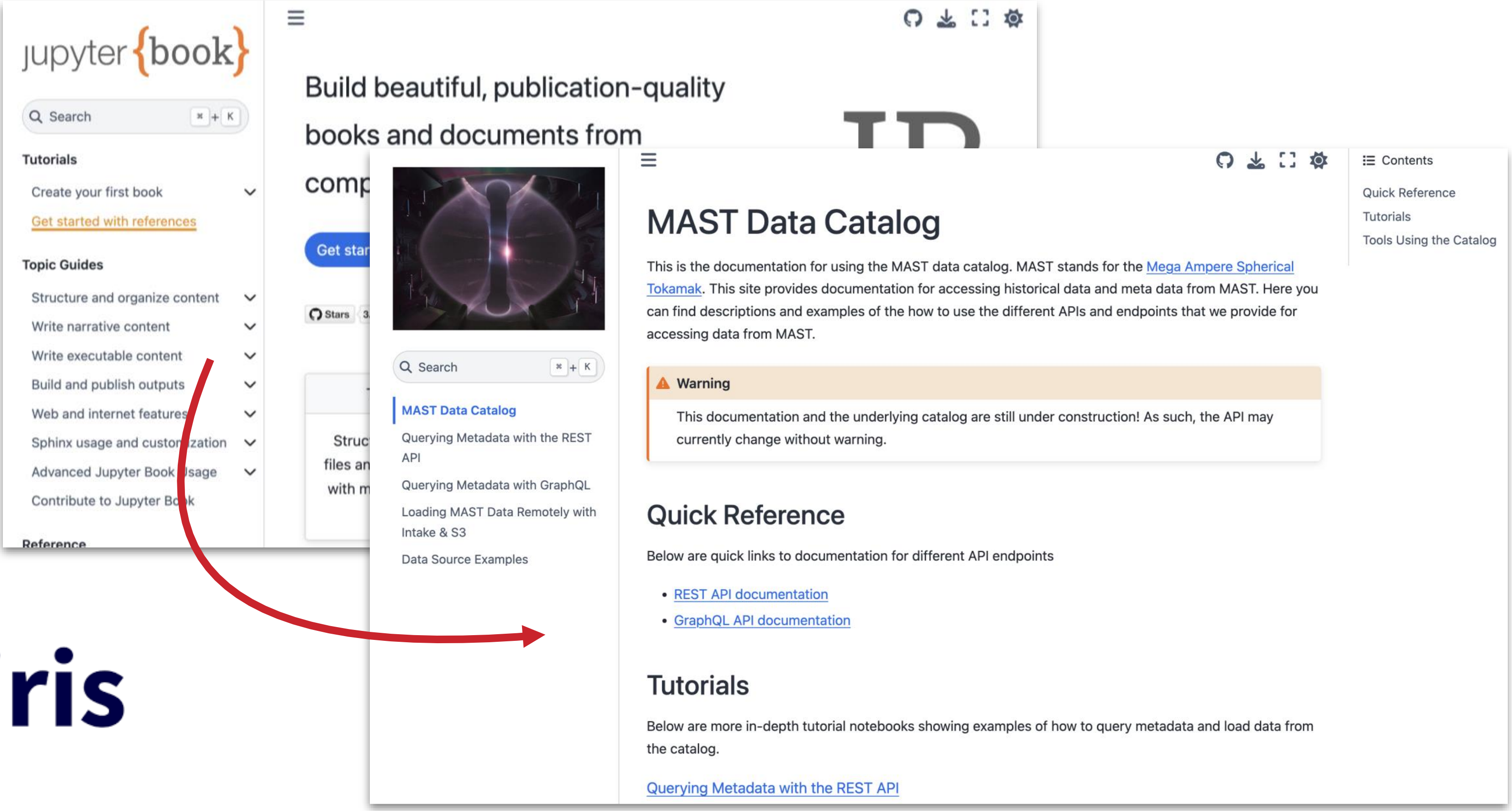
**Download a single source for all shots**

```
s5cmd –no-sign-request -—endpoint-url https://s3.echo.stfc.ac.uk \
  cp "s3://mast/level1/shots/*.zarr/rbb/*" ./data
```

s5cmd github

# User Documentation

Using Jupyter book to build documentation that is also executable


jupyter {book}
Search
Tutorials
Create your first book
Get started with references
Topic Guides
Structure and organize content
Write narrative content
Write executable content
Build and publish outputs
Web and internet features
Sphinx usage and customization
Advanced Jupyter Book usage
Contribute to Jupyter Book
Reference
Get star
Stars

Build beautiful, publication-quality books and documents from comp

Struc files an with m

Search
MAST Data Catalog
Querying Metadata with the REST API
Querying Metadata with GraphQL
Loading MAST Data Remotely with Intake & S3
Data Source Examples
Contents
Quick Reference
Tutorials
Tools Using the Catalog

## MAST Data Catalog

This is the documentation for using the MAST data catalog. MAST stands for the Mega Ampere Spherical Tokamak. This site provides documentation for accessing historical data and meta data from MAST. Here you can find descriptions and examples of the how to use the different APIs and endpoints that we provide for accessing data from MAST.

Warning

This documentation and the underlying catalog are still under construction! As such, the API may currently change without warning.

### Quick Reference

Below are quick links to documentation for different API endpoints

- REST API documentation
- GraphQL API documentation

### Tutorials

Below are more in-depth tutorial notebooks showing examples of how to query metadata and load data from the catalog.

Querying Metadata with the REST API

Hosted by IRIS on
STFC Cloud

Jupyter Book Project

# Summary

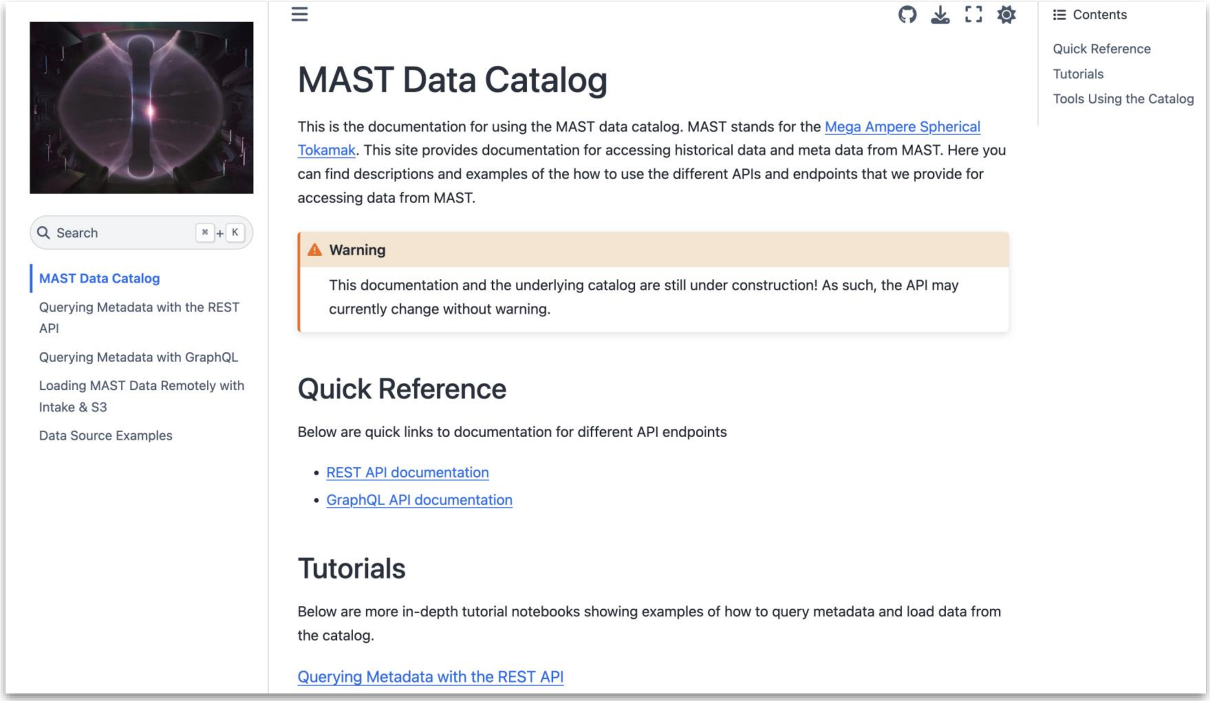
UK Atomic Energy Authority

We developed a data infrastructure solution for the history of the MAST experiment

We provide a public REST API for the metadata

We provide a public the history of the MAST data in cloud object storage

**All developed and hosted on IRIS resources!**



- **MAST Data Catalog**
- Querying Metadata with the REST API
- Querying Metadata with GraphQL
- Loading MAST Data Remotely with Intake & S3
- Data Source Examples

## MAST Data Catalog

This is the documentation for using the MAST data catalog. MAST stands for the Mega Ampere Spherical Tokamak. This site provides documentation for accessing historical data and meta data from MAST. Here you can find descriptions and examples of the how to use the different APIs and endpoints that we provide for accessing data from MAST.

> **Warning**
>
> This documentation and the underlying catalog are still under construction! As such, the API may currently change without warning.

## Quick Reference

Below are quick links to documentation for different API endpoints

- REST API documentation
- GraphQL API documentation

## Tutorials

Below are more in-depth tutorial notebooks showing examples of how to query metadata and load data from the catalog.

Querying Metadata with the REST API

- Quick Reference
- Tutorials
- Tools Using the Catalog

Demo site:
https://mastapp.site/

# With Thanks

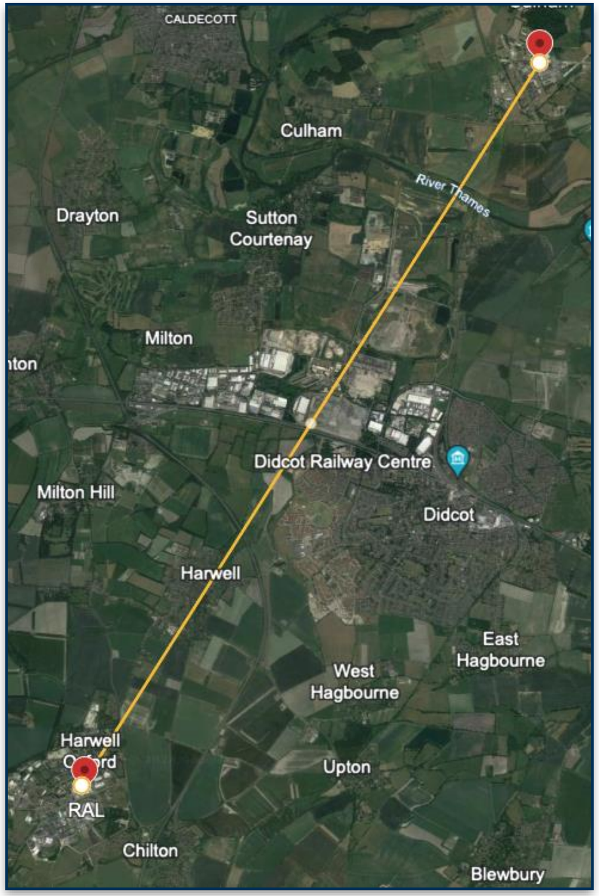
A cross-organisation collaboration between STFC and UKAEA and was funded as part of the Fusion Computing Lab programme.

## STFC

Saiful Khan

Jeyan Thiyagalingam

Rutherford Appleton Laboratory



Culham Centre for Fusion Energy

## UKAEA

Samuel Jackson

Nathan Cummings

James Hodson

Khalid Lawal

Larisa Dorman-Gajic

Daniel Brennand

Shaun De Witt

Stanislas Pamela

Rob Akers

With special thanks to Jonathan Hollocombe, Stephen Dixon, Jimmy Measures, Lucy Kogan, Adam Parker, Deniza Chekrygina, Alejandra Gonzalez-Beltran, IRIS, the STFC Cloud and STFC Data Services Groups.

(C) Google (2024) Didcot Area, Accessed June 2024