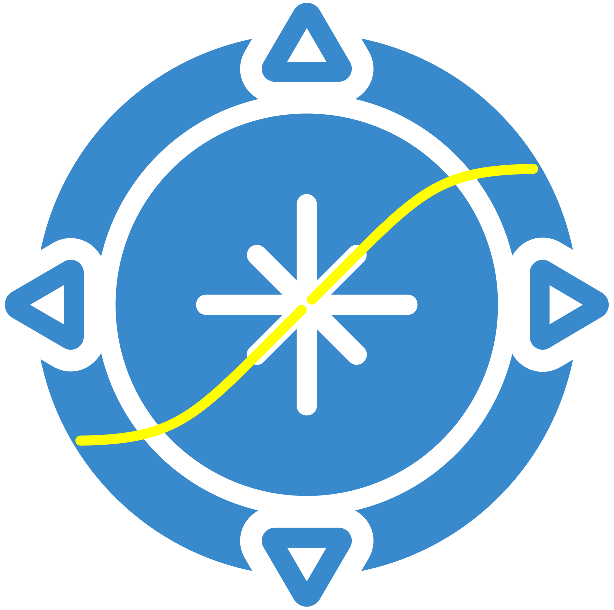
# ami Ecosystem

P.-A. Delsart, J. Fulachier, F. Lambert, J. Odier

# Presentation outline

**A**TLAS **M**etadata **I**nterface (**AMI**)

A generic ecosystem for scientific metadata



1 **What is AMI?**

2 **AMI backend**

3 **AMI frontend**

4 **Metadata aggregation**

5 **AMI live demo**

01

# What is AMI ?

ami

# What is AMI?

- **AMI (ATLAS Metadata Interface) is a generic ecosystem dedicated to scientific metadata:**
  - Heterogenous datasource / Web service connectivity
    - → AMI can connect to any existing DB or Web service
  - High level primitives for metadata extraction and processing
  - High level tools for performing data selection by metadata criteria

- **The ecosystem has development kits for:**
  - Developing server-side metadata-oriented subsystems (in JAVA)
  - Developing client-side metadata-oriented Web applications (in JS) and scripts (in Python, JS, ...)

- **AMI is designed to work in big-data context:**
  - Scalability, evolutivity and maintainability

LPSC GRENOBLE | MODANE

cnrs IN2P3 Les deux infinis

UGA Université Grenoble Alpes

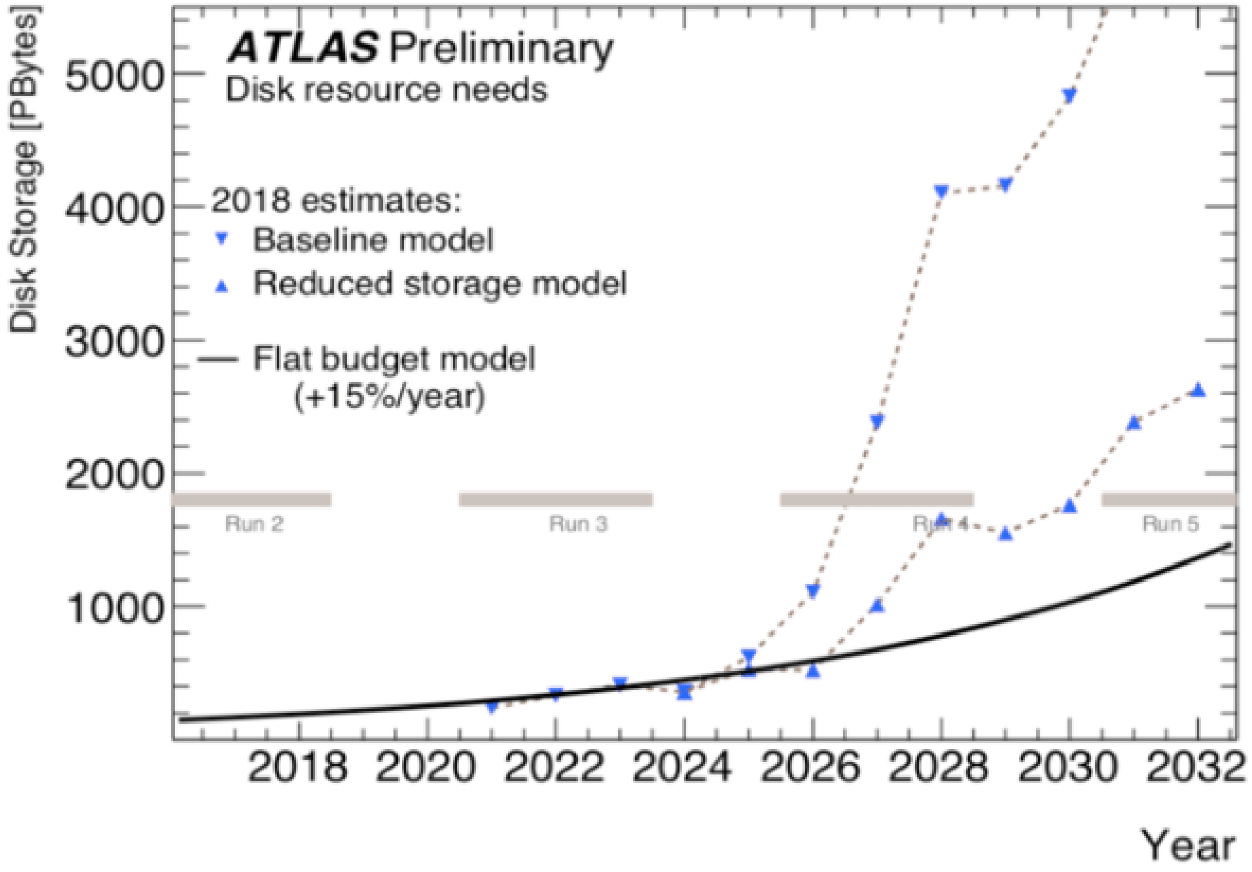
CERN

ATLAS EXPERIMENT

What is AMI?

# What is AMI?

- > 22 years of experience within the ATLAS collaboration at CERN
- AMI hosts metadata for O(10^6) datasets and O(10^9) files



ATLAS EXPERIMENT

What is AMI?

# Typical usage

Aggregating scientific metadata from heterogeneous datasources

Searching scientific data by metadata criteria

Storing software & job metadata

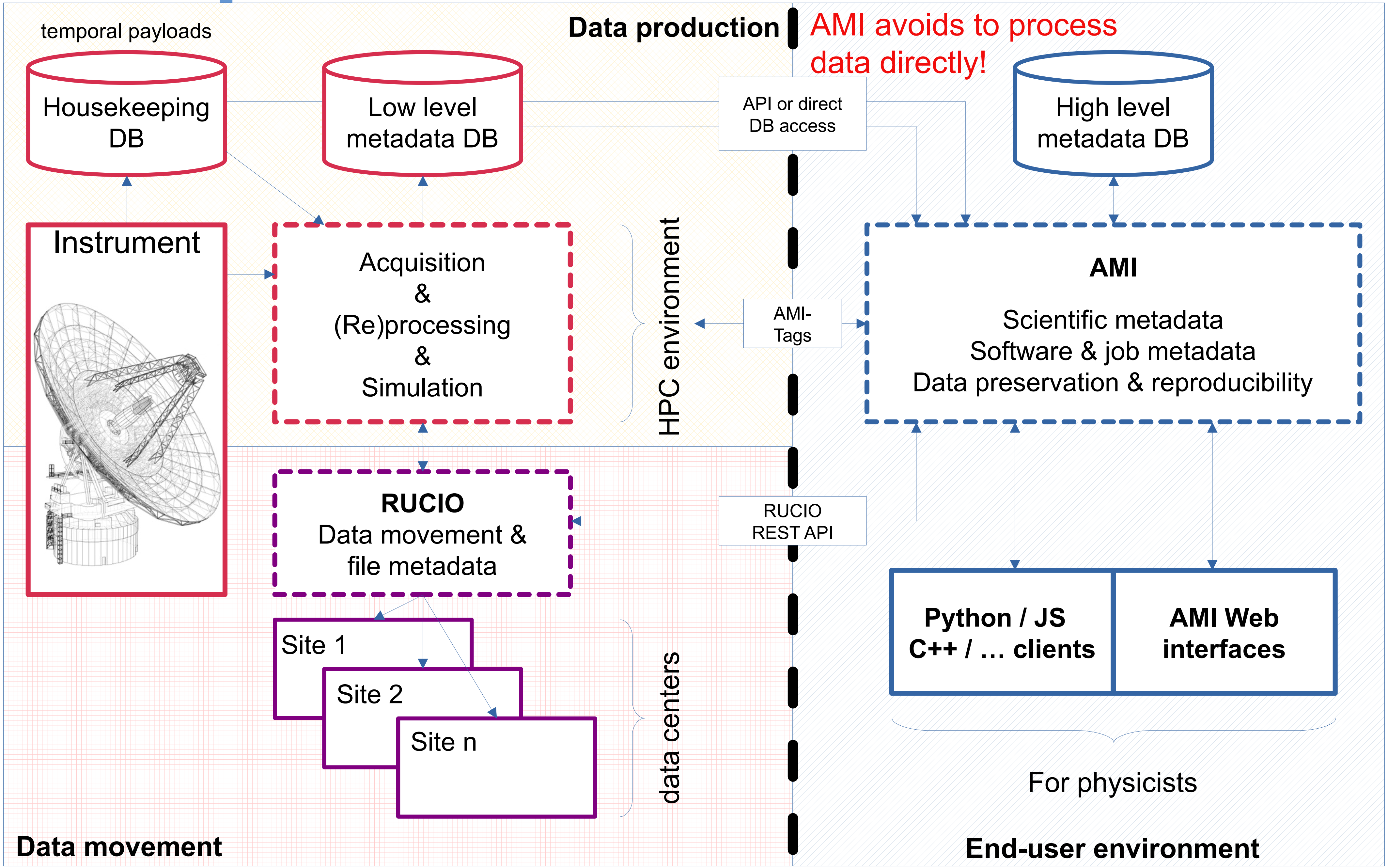
Data preservation & reproducibility

What is AMI?

# Typical usage

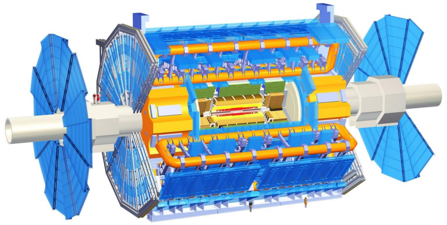
- **Aggregating scientific metadata from heterogeneous datasources:**
  - Condition / housekeeping metadata
  - Dataset and file metadata (from production and data movement systems)
  - End-user metadata (physics parameters, annotations, comments, …)
- **Searching scientific data by metadata criteria:**
  - Via Web interfaces and Python / JS / C++ / Java / ... scripts
  - 👍 This is the most important feature for end-users / physicists
- **Defining metadata for softwares (version, parameters), grid jobs, etc…**
  - In ATLAS, each grid job is defined from an AMI-Tag (= set of software parameters)
  - Each dataset is associated to one or many AMI-Tags
- **Associating datasets and papers:**
  - Data preservation
  - Reproducibility

# Example of workflow



What is AMI?

# AMI Ecosystem



### Detector

Initial data and metadata

1

2

### AMI TaskServer

Metadata aggregation, transformation and storing

### AMI Backend

Web services & high level metadata features

3

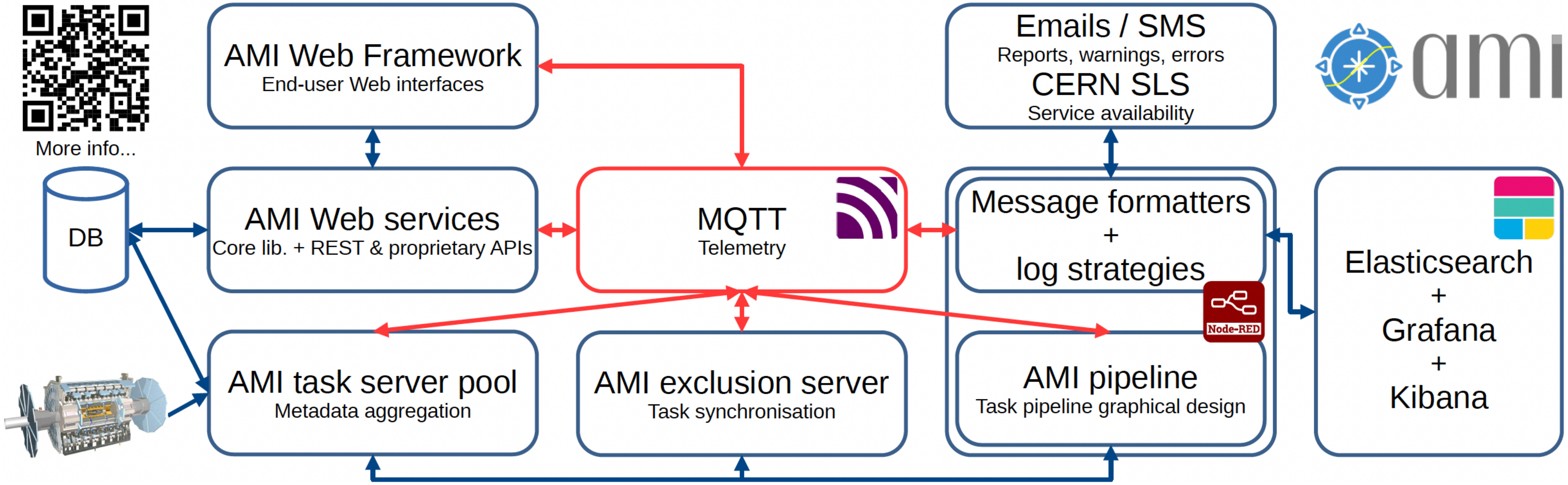
4

### AMI Frontend

Web interfaces & REST, Python, JS, Java, C++, ... clients

What is AMI?

# AMI Ecosystem – whole stack



More info...

DB

AMI Web Framework
End-user Web interfaces

AMI Web services
Core lib. + REST & proprietary APIs

MQTT
Telemetry

AMI task server pool
Metadata aggregation

AMI exclusion server
Task synchronisation

Emails / SMS
Reports, warnings, errors
CERN SLS
Service availability

Message formatters
+
log strategies

Node-RED

AMI pipeline
Task pipeline graphical design

Elasticsearch
+
Grafana
+
Kibana

- **Images on DockerHub and DockerCompose demo**
  - Easy deployment in a Docker Compose or Kubernetes environment
  - CHEP 2023 paper: https://cds.cern.ch/record/2868009/files/ATL-SOFT-PROC-2023-009.pdf
- **Each sub-system of the AMI ecosystem can connect to an optional MQTT broker for:**
  - Monitoring purpose (cpu usage, ram usage, disk usage, ...)
  - Control purpose (reload, restart, ...)

What is AMI?

**AMI can immediately connect to any existing database or web service**

**AMI meets the needs of both small and large scientific experiments**

👍

02

# AMI Backend

ami

# AMI Backend

- **AMI Backend**
  - Based on the AMI Java Core library
  - Control and monitoring capabilities via MQTT
  - Scalable Web service (REST and proprietary APIs) with authentications / authorizations
  - Heterogeneous datasource connectivity (as soon as a Java JDBC driver exists)
  - Command engine (= the way of talking with AMI)
    - Metadata queries (generic or more specific queries), experiment-specific commands, service administration, …
  - Metadata Query Language (= MQL, see next slide)
  - High-level primitives for data & metadata handling

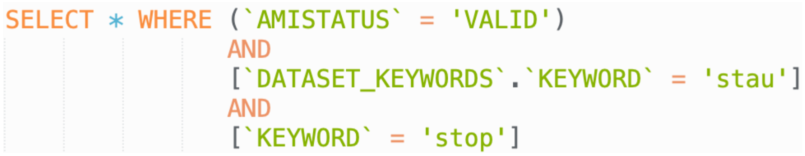
  - **See CHEP 2019 paper:**
    - https://www.epj-conferences.org/articles/epjconf/pdf/2019/19/epjconf_chep2018_05046.pdf

AMI Backend

# AMI Backend – MQL

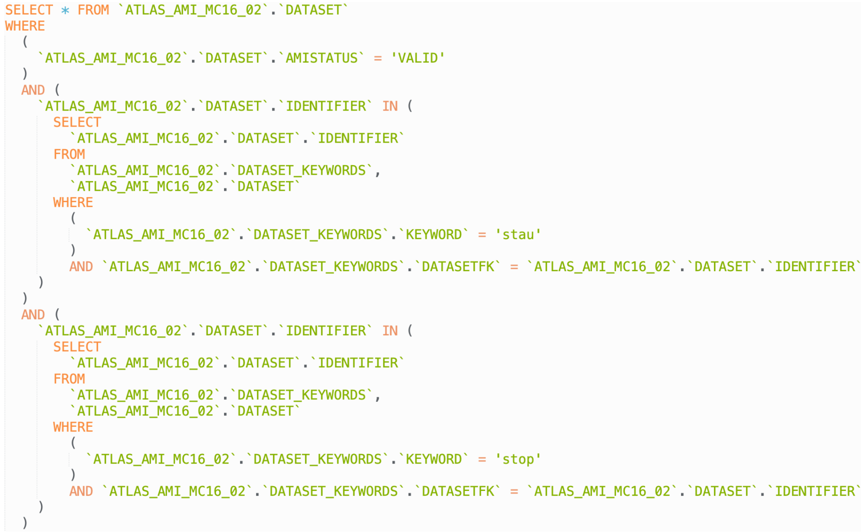
- **Metadata Query Language (MQL)**
  - Kind of SQL without  FROM clause nor join
  - It makes it possible to build queries without knowing table relations
  - Joins are automatically generated from the AMI reflexion sub-system
  - MQL turns the database-oriented perspective into a metadata-oriented perspective.

- **See CHEP 2019 paper:**
  - https://www.epj-conferences.org/articles/epjconf/pdf/2020/21/epjconf_chep2020_04044.pdf



```
SELECT * WHERE (`AMISTATUS` = 'VALID')
               AND
               [`DATASET_KEYWORDS`.`KEYWORD` = 'stau']
               AND
               [`KEYWORD` = 'stop']
```

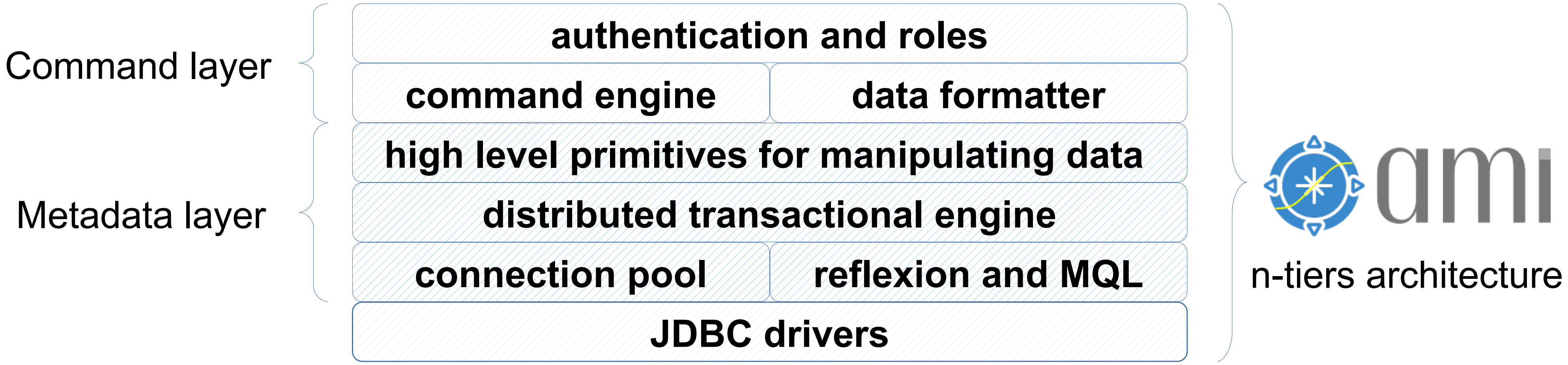
MQL to SQL



```
SELECT * FROM `ATLAS_AMI_MC16_02`.`DATASET`
WHERE
  (
    `ATLAS_AMI_MC16_02`.`DATASET`.`AMISTATUS` = 'VALID'
  )
  AND (
    `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER` IN (
      SELECT
        `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER`
      FROM
        `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`,
        `ATLAS_AMI_MC16_02`.`DATASET`
      WHERE
        (
          `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`.`KEYWORD` = 'stau'
        )
        AND `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`.`DATASETFK` = `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER`
    )
  )
  AND (
    `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER` IN (
      SELECT
        `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER`
      FROM
        `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`,
        `ATLAS_AMI_MC16_02`.`DATASET`
      WHERE
        (
          `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`.`KEYWORD` = 'stop'
        )
        AND `ATLAS_AMI_MC16_02`.`DATASET_KEYWORDS`.`DATASETFK` = `ATLAS_AMI_MC16_02`.`DATASET`.`IDENTIFIER`
    )
  )
```

AMI Backend

# AMI Backend – overview



Command layer

Metadata layer

authentication and roles

command engine | data formatter

high level primitives for manipulating data

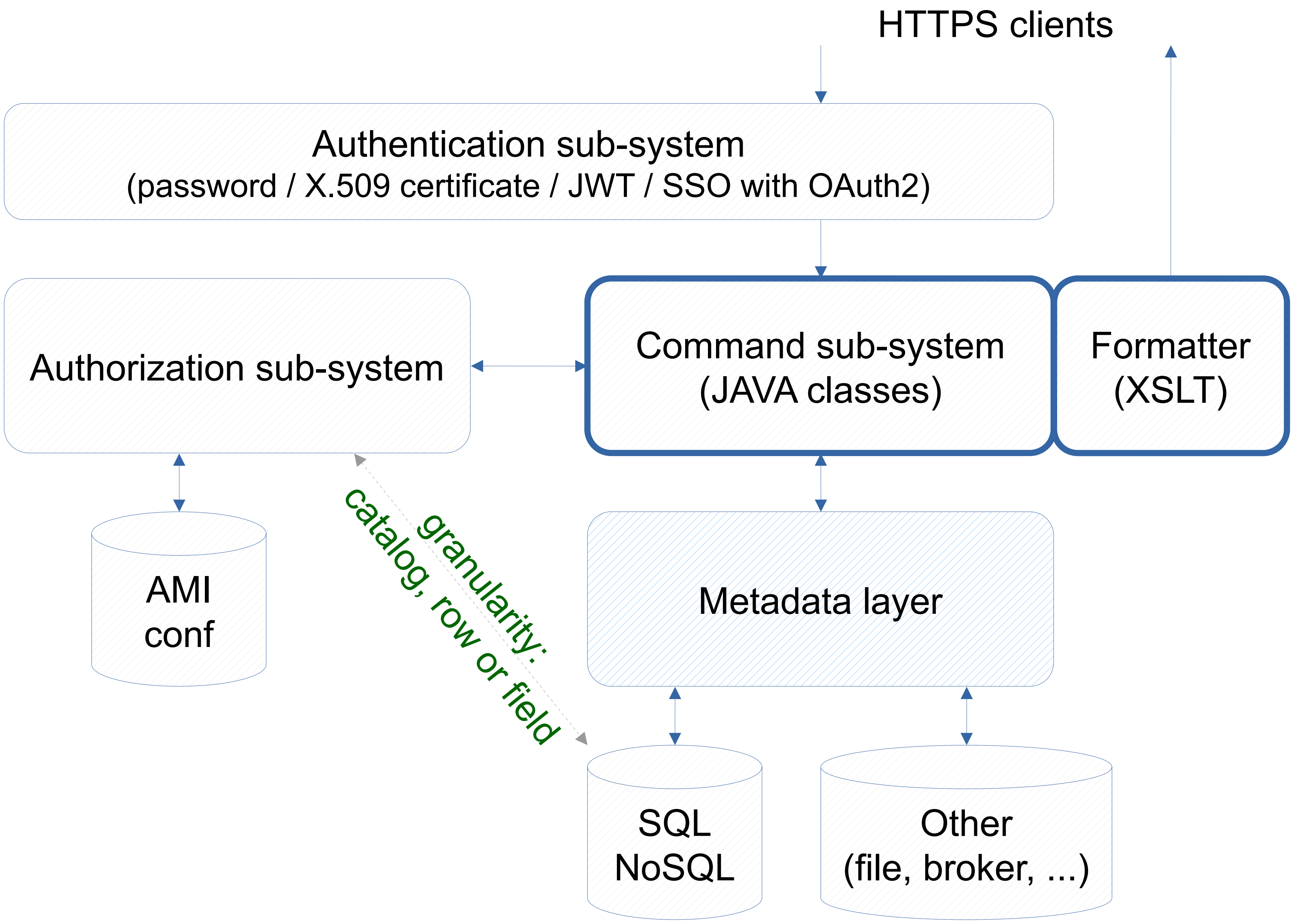
distributed transactional engine

connection pool | reflexion and MQL

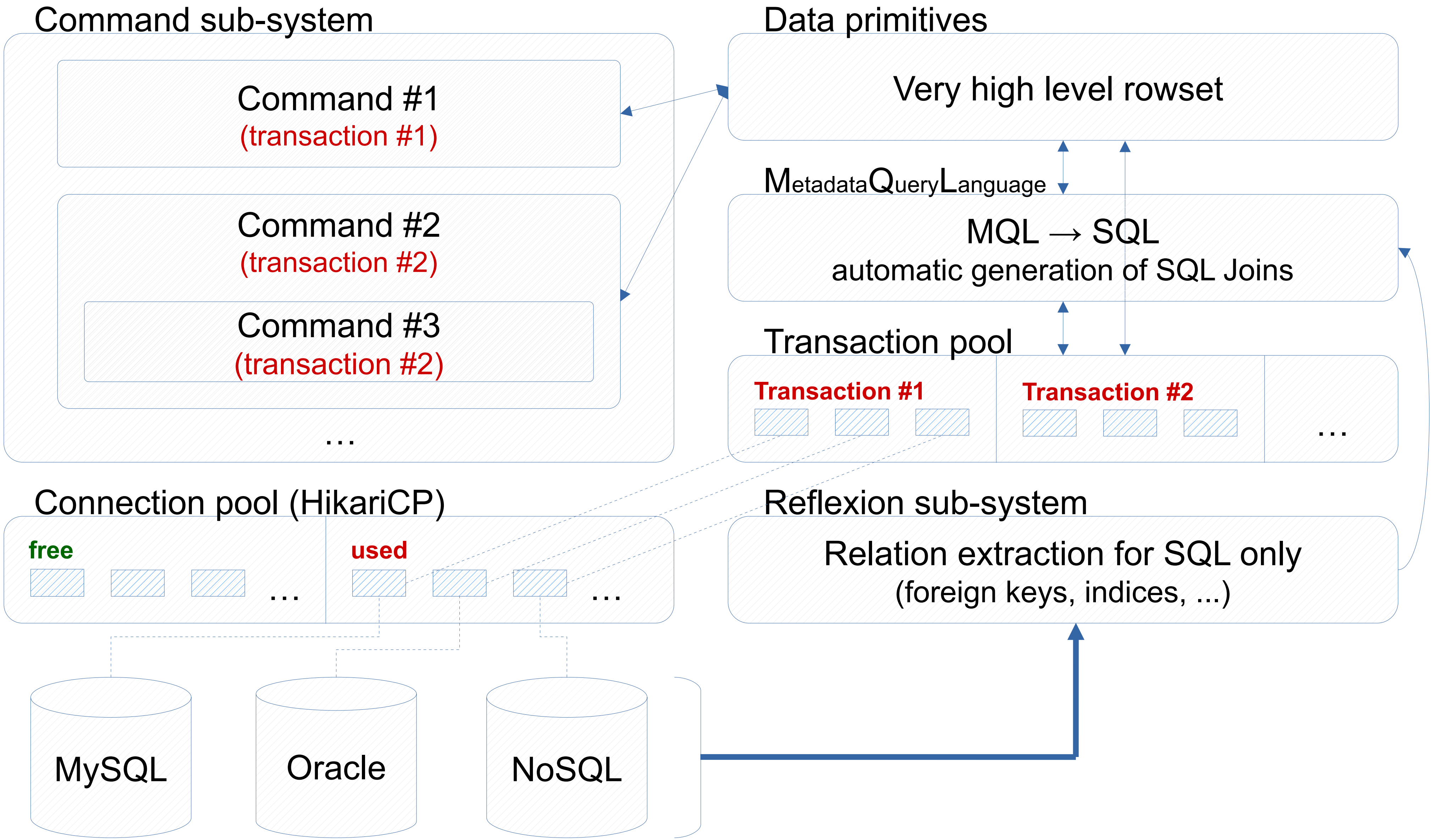
JDBC drivers

n-tiers architecture

# AMI Backend – command layer



# AMI Backend – metadata layer



Command sub-system

Command #1
(transaction #1)

Command #2
(transaction #2)

Command #3
(transaction #2)

…

Data primitives

Very high level rowset

MetadataQueryLanguage

MQL → SQL
automatic generation of SQL Joins

Transaction pool

Transaction #1

Transaction #2

…

Connection pool (HikariCP)

free

…

used

…

Reflexion sub-system

Relation extraction for SQL only
(foreign keys, indices, ...)

MySQL

Oracle

NoSQL

03

# AMI Frontend

ami

AMI Frontend

# AMI Web Framework

- **AMI Frontend** (aka AMI Web Framework (AWF))
  - Based on modern technologies (Webpack, Boostrap, TWIG)
  - Optimized for developing metadata-oriented JS Web applications
  - AMI provides a set of ready-to-use applications and controls (see next slide)
    - AMI controls are embeddable in external websites such as WIKIs, confluence, …
  - Patterns MVC (with AMI-Twig) or MVVM (with Vue.js 3)

- **See CHEP 2019 paper:**
  - https://www.epj-conferences.org/articles/epjconf/pdf/2019/19/epjconf_chep2018_04004.pdf

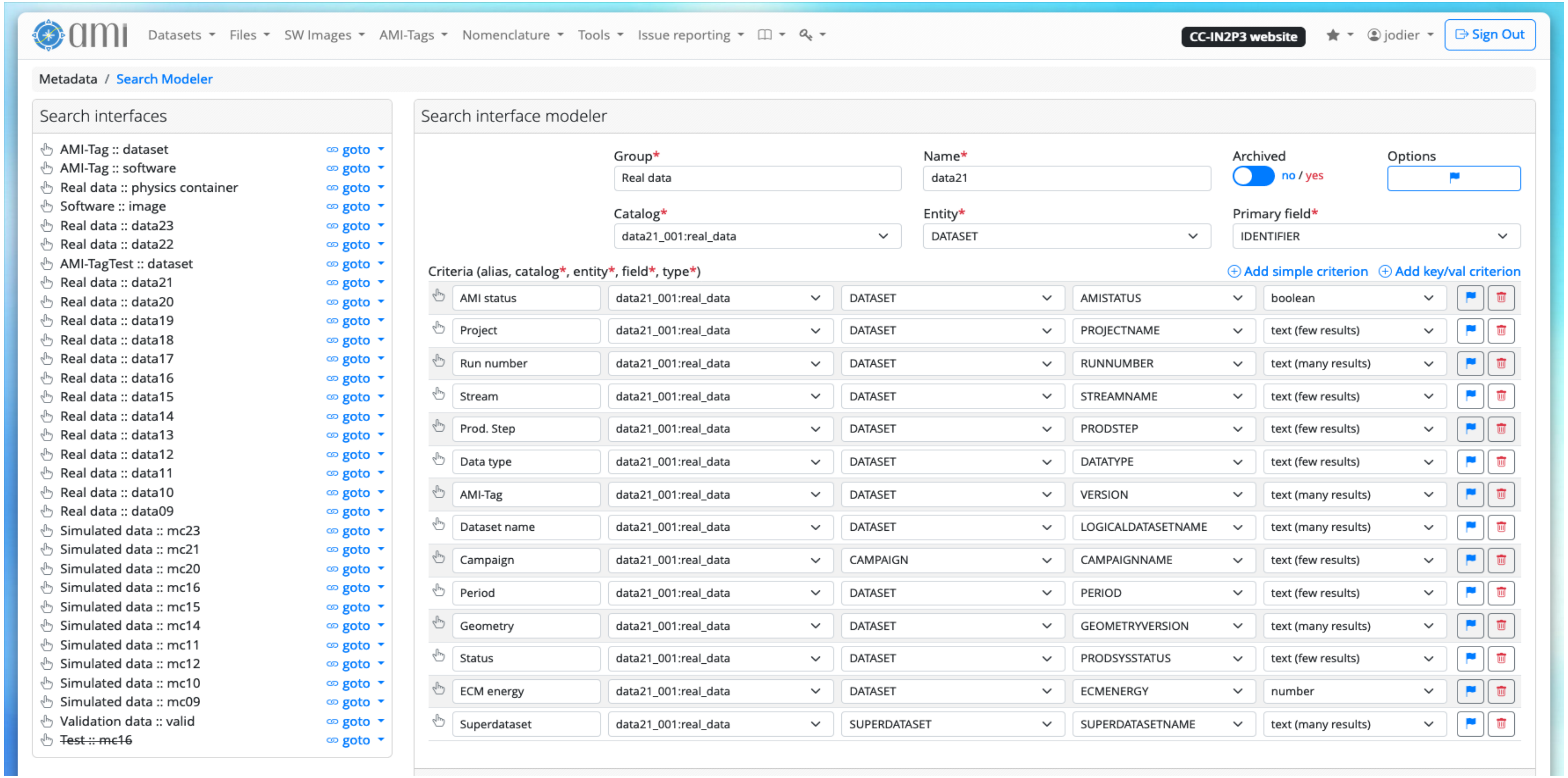
AMI Frontend

# AMI Web Framework

- **Applications are generally built by assembling controls**
- **Main available controls:**
  - Dialog boxes
  - Controls for annotating entities
  - Controls for searching (Simple Search, Criteria Search, …)
  - Controls for displaying (Schema Viewer, Tab, Table, Element Info, …)
- **Main available applications:**
  - Embedded CMS
  - AMI command interpreter
  - Admin Dashboard and Monitoring
  - Schema Viewer, Table Viewer, Simple Search, Criteria Search, Search Modeler, …

# Designing search interfaces (admin)

AMI Frontend



Metadata / Search Modeler

## Search interfaces

- AMI-Tag :: dataset
- AMI-Tag :: software
- Real data :: physics container
- Software :: image
- Real data :: data23
- Real data :: data22
- AMI-TagTest :: dataset
- Real data :: data21
- Real data :: data20
- Real data :: data19
- Real data :: data18
- Real data :: data17
- Real data :: data16
- Real data :: data15
- Real data :: data14
- Real data :: data13
- Real data :: data12
- Real data :: data11
- Real data :: data10
- Real data :: data09
- Simulated data :: mc23
- Simulated data :: mc21
- Simulated data :: mc20
- Simulated data :: mc16
- Simulated data :: mc15
- Simulated data :: mc14
- Simulated data :: mc11
- Simulated data :: mc12
- Simulated data :: mc10
- Simulated data :: mc09
- Validation data :: valid
- ~~Test :: mc16~~

## Search interface modeler

Group*: Real data

Name*: data21

Archived: no / yes

Options

Catalog*: data21_001:real_data

Entity*: DATASET

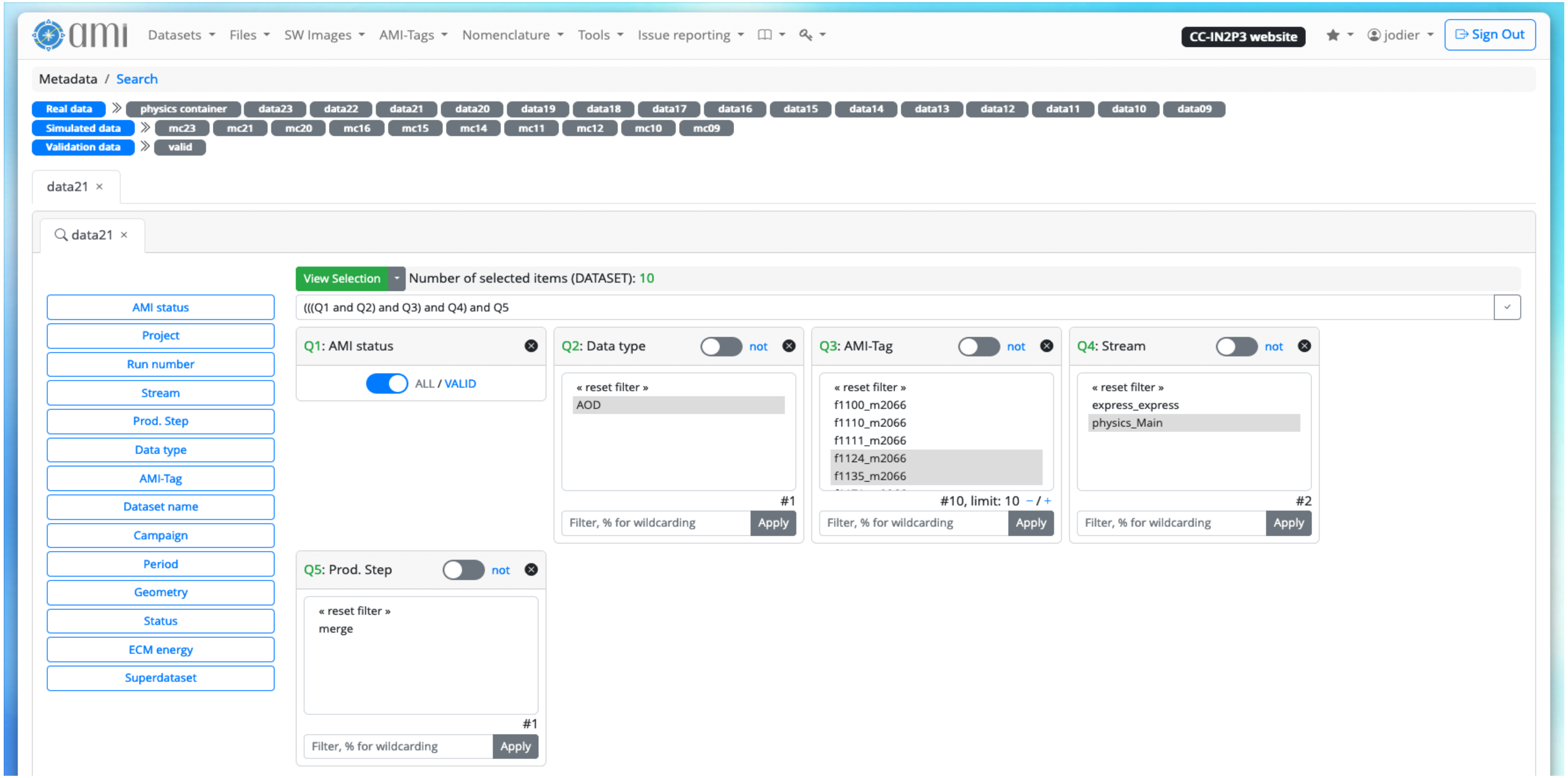
Primary field*: IDENTIFIER

Criteria (alias, catalog*, entity*, field*, type*)

| | | | | |
|---|---|---|---|---|
| AMI status | data21_001:real_data | DATASET | AMISTATUS | boolean |
| Project | data21_001:real_data | DATASET | PROJECTNAME | text (few results) |
| Run number | data21_001:real_data | DATASET | RUNNUMBER | text (many results) |
| Stream | data21_001:real_data | DATASET | STREAMNAME | text (few results) |
| Prod. Step | data21_001:real_data | DATASET | PRODSTEP | text (few results) |
| Data type | data21_001:real_data | DATASET | DATATYPE | text (few results) |
| AMI-Tag | data21_001:real_data | DATASET | VERSION | text (many results) |
| Dataset name | data21_001:real_data | DATASET | LOGICALDATASETNAME | text (many results) |
| Campaign | data21_001:real_data | CAMPAIGN | CAMPAIGNNAME | text (few results) |
| Period | data21_001:real_data | DATASET | PERIOD | text (few results) |
| Geometry | data21_001:real_data | DATASET | GEOMETRYVERSION | text (many results) |
| Status | data21_001:real_data | DATASET | PRODSYSSTATUS | text (few results) |
| ECM energy | data21_001:real_data | DATASET | ECMENERGY | number |
| Superdataset | data21_001:real_data | SUPERDATASET | SUPERDATASETNAME | text (many results) |

AMI Frontend

# Searching data by criteria



Metadata / Search

Number of selected items (DATASET): 10

(((Q1 and Q2) and Q3) and Q4) and Q5

Q1: AMI status

ALL / VALID

Q2: Data type — not

« reset filter »
AOD

#1

Filter, % for wildcarding

Q3: AMI-Tag — not

« reset filter »
f1100_m2066
f1110_m2066
f1111_m2066
f1124_m2066
f1135_m2066

#10, limit: 10 − / +

Filter, % for wildcarding

Q4: Stream — not

« reset filter »
express_express
physics_Main

#2

Filter, % for wildcarding

Q5: Prod. Step — not

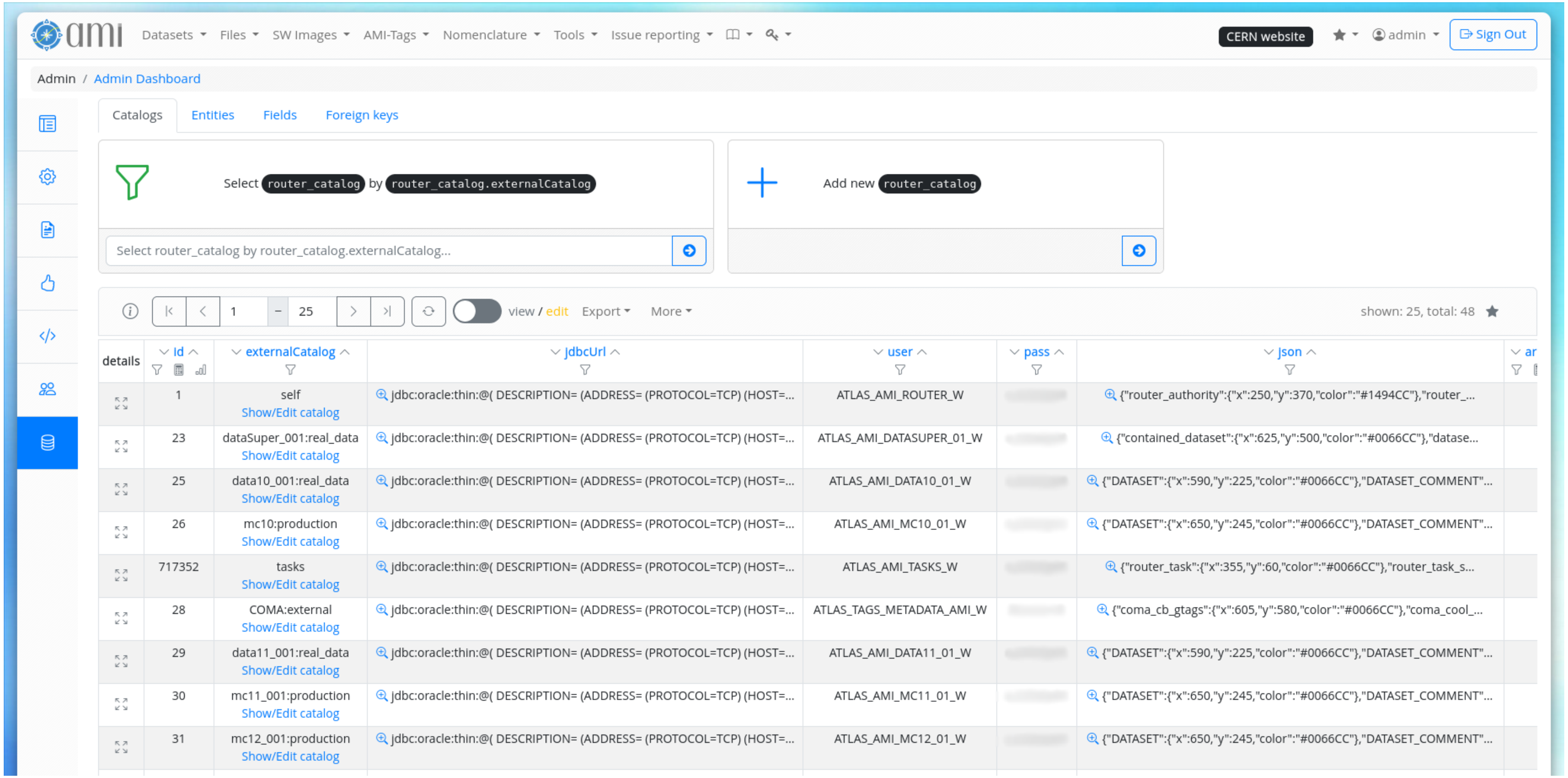
« reset filter »
merge

#1

Filter, % for wildcarding

# Search results

AMI Frontend



Admin / Admin Dashboard

Catalogs | Entities | Fields | Foreign keys

Select `router_catalog` by `router_catalog.externalCatalog`

Select router_catalog by router_catalog.externalCatalog...

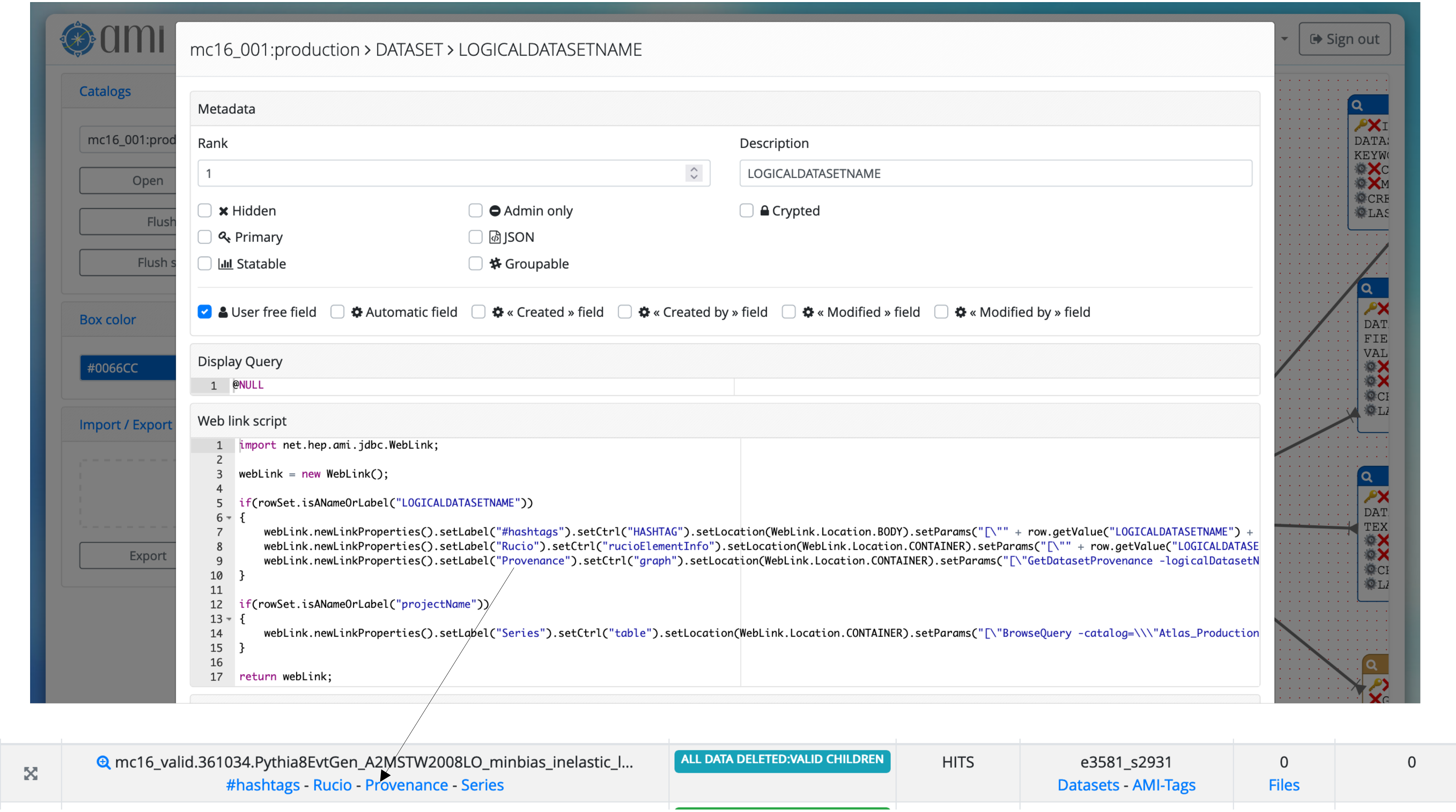
Add new `router_catalog`

1 – 25 view / edit Export ▾ More ▾ shown: 25, total: 48

| details | id | externalCatalog | jdbcUrl | user | pass | json | ar |
|---|---|---|---|---|---|---|---|
| | 1 | self<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_ROUTER_W | [redacted] | {"router_authority":{"x":250,"y":370,"color":"#1494CC"},"router_... | |
| | 23 | dataSuper_001:real_data<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_DATASUPER_01_W | [redacted] | {"contained_dataset":{"x":625,"y":500,"color":"#0066CC"},"datase... | |
| | 25 | data10_001:real_data<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_DATA10_01_W | [redacted] | {"DATASET":{"x":590,"y":225,"color":"#0066CC"},"DATASET_COMMENT"... | |
| | 26 | mc10:production<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_MC10_01_W | [redacted] | {"DATASET":{"x":650,"y":245,"color":"#0066CC"},"DATASET_COMMENT"... | |
| | 717352 | tasks<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_TASKS_W | [redacted] | {"router_task":{"x":355,"y":60,"color":"#0066CC"},"router_task_s... | |
| | 28 | COMA:external<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_TAGS_METADATA_AMI_W | [redacted] | {"coma_cb_gtags":{"x":605,"y":580,"color":"#0066CC"},"coma_cool_... | |
| | 29 | data11_001:real_data<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_DATA11_01_W | [redacted] | {"DATASET":{"x":590,"y":225,"color":"#0066CC"},"DATASET_COMMENT"... | |
| | 30 | mc11_001:production<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_MC11_01_W | [redacted] | {"DATASET":{"x":650,"y":245,"color":"#0066CC"},"DATASET_COMMENT"... | |
| | 31 | mc12_001:production<br>Show/Edit catalog | jdbc:oracle:thin:@( DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=... | ATLAS_AMI_MC12_01_W | [redacted] | {"DATASET":{"x":650,"y":245,"color":"#0066CC"},"DATASET_COMMENT"... | |

AMI Frontend

# Low code interface for customization (admin)



mc16_001:production > DATASET > LOGICALDATASETNAME

Metadata

Rank: 1

Description: LOGICALDATASETNAME

☐ Hidden ☐ Admin only ☐ Crypted
☐ Primary ☐ JSON
☐ Statable ☐ Groupable

☑ User free field ☐ Automatic field ☐ « Created » field ☐ « Created by » field ☐ « Modified » field ☐ « Modified by » field

Display Query

```
@NULL
```

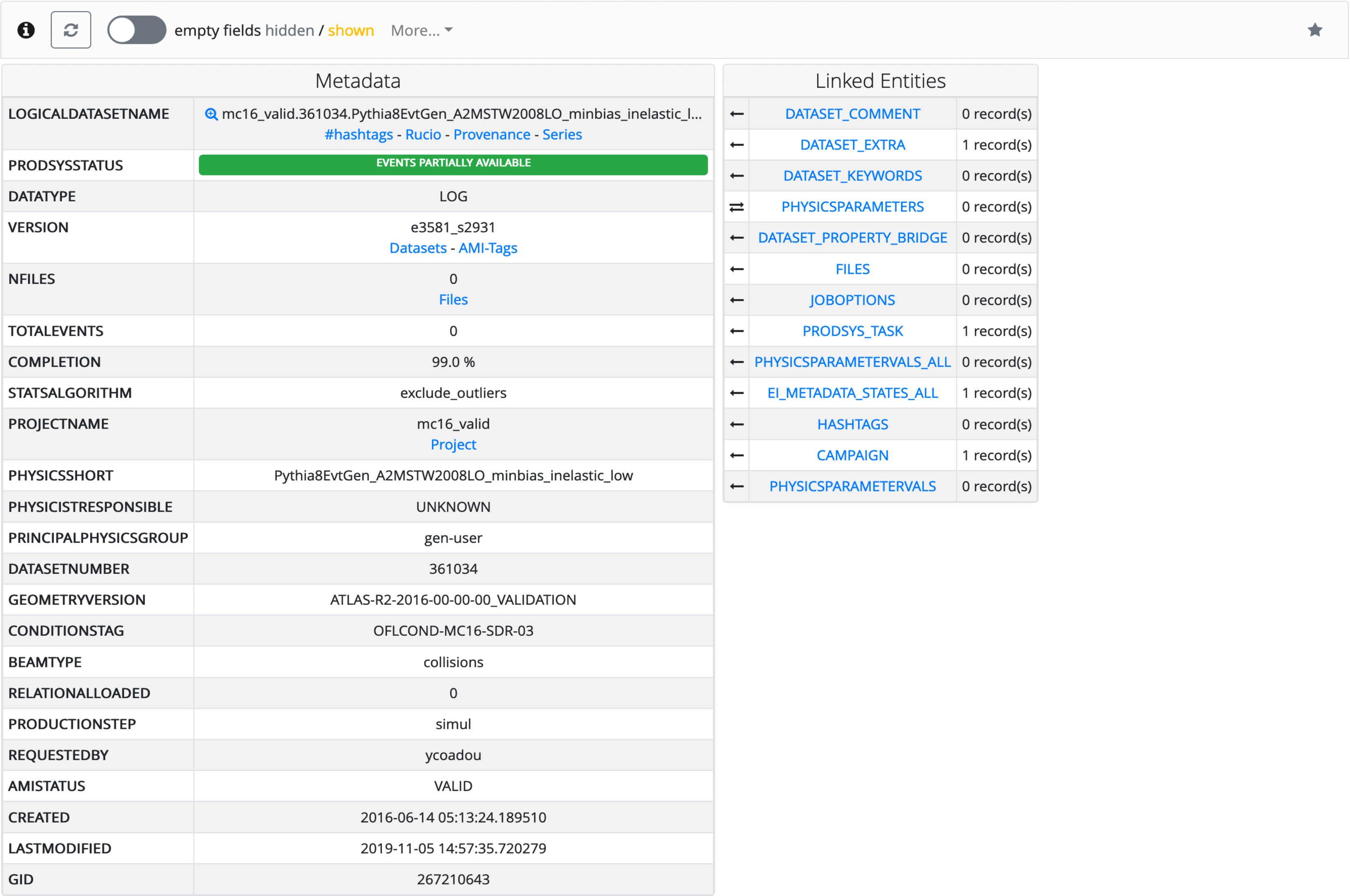
Web link script

```
import net.hep.ami.jdbc.WebLink;

webLink = new WebLink();

if(rowSet.isANameOrLabel("LOGICALDATASETNAME"))
{
    webLink.newLinkProperties().setLabel("#hashtags").setCtrl("HASHTAG").setLocation(WebLink.Location.BODY).setParams("[\"" + row.getValue("LOGICALDATASETNAME") +
    webLink.newLinkProperties().setLabel("Rucio").setCtrl("rucioElementInfo").setLocation(WebLink.Location.CONTAINER).setParams("[\"" + row.getValue("LOGICALDATASE
    webLink.newLinkProperties().setLabel("Provenance").setCtrl("graph").setLocation(WebLink.Location.CONTAINER).setParams("[\"GetDatasetProvenance -logicalDatasetN
}

if(rowSet.isANameOrLabel("projectName"))
{
    webLink.newLinkProperties().setLabel("Series").setCtrl("table").setLocation(WebLink.Location.CONTAINER).setParams("[\"BrowseQuery -catalog=\\\"Atlas_Production
}

return webLink;
```

| | | | | | | |
|---|---|---|---|---|---|---|
| | mc16_valid.361034.Pythia8EvtGen_A2MSTW2008LO_minbias_inelastic_l...<br>#hashtags - Rucio - Provenance - Series | ALL DATA DELETED:VALID CHILDREN | HITS | e3581_s2931<br>Datasets - AMI-Tags | 0<br>Files | 0 |

AMI Frontend

# Details and linked entities



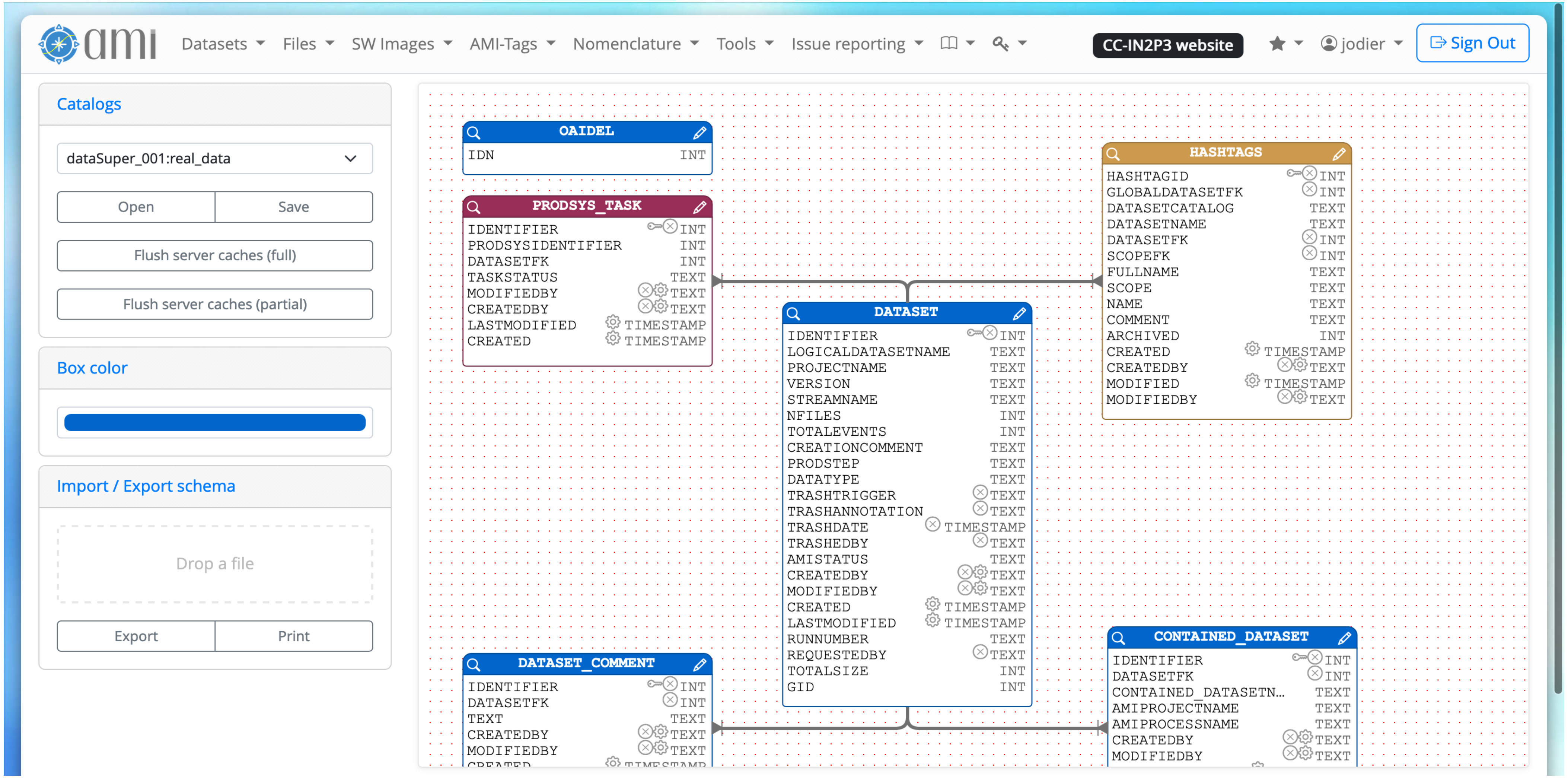
empty fields hidden / shown More...

| Metadata | |
|---|---|
| LOGICALDATASETNAME | mc16_valid.361034.Pythia8EvtGen_A2MSTW2008LO_minbias_inelastic_l...<br>#hashtags - Rucio - Provenance - Series |
| PRODSYSSTATUS | EVENTS PARTIALLY AVAILABLE |
| DATATYPE | LOG |
| VERSION | e3581_s2931<br>Datasets - AMI-Tags |
| NFILES | 0<br>Files |
| TOTALEVENTS | 0 |
| COMPLETION | 99.0 % |
| STATSALGORITHM | exclude_outliers |
| PROJECTNAME | mc16_valid<br>Project |
| PHYSICSSHORT | Pythia8EvtGen_A2MSTW2008LO_minbias_inelastic_low |
| PHYSICISTRESPONSIBLE | UNKNOWN |
| PRINCIPALPHYSICSGROUP | gen-user |
| DATASETNUMBER | 361034 |
| GEOMETRYVERSION | ATLAS-R2-2016-00-00-00_VALIDATION |
| CONDITIONSTAG | OFLCOND-MC16-SDR-03 |
| BEAMTYPE | collisions |
| RELATIONALLOADED | 0 |
| PRODUCTIONSTEP | simul |
| REQUESTEDBY | ycoadou |
| AMISTATUS | VALID |
| CREATED | 2016-06-14 05:13:24.189510 |
| LASTMODIFIED | 2019-11-05 14:57:35.720279 |
| GID | 267210643 |

| Linked Entities | | |
|---|---|---|
| ← | DATASET_COMMENT | 0 record(s) |
| ← | DATASET_EXTRA | 1 record(s) |
| ← | DATASET_KEYWORDS | 0 record(s) |
| ⇄ | PHYSICSPARAMETERS | 0 record(s) |
| ← | DATASET_PROPERTY_BRIDGE | 0 record(s) |
| ← | FILES | 0 record(s) |
| ← | JOBOPTIONS | 0 record(s) |
| ← | PRODSYS_TASK | 1 record(s) |
| ← | PHYSICSPARAMETERVALS_ALL | 0 record(s) |
| ← | EI_METADATA_STATES_ALL | 1 record(s) |
| ← | HASHTAGS | 0 record(s) |
| ← | CAMPAIGN | 1 record(s) |
| ← | PHYSICSPARAMETERVALS | 0 record(s) |

AMI Frontend

# DB visualization and browsing



Catalogs

dataSuper_001:real_data

Open | Save

Flush server caches (full)

Flush server caches (partial)

Box color

Import / Export schema

Drop a file

Export | Print

**OAIDEL**

| | |
|---|---|
| IDN | INT |

**PRODSYS_TASK**

| | |
|---|---|
| IDENTIFIER | INT |
| PRODSYSIDENTIFIER | INT |
| DATASETFK | INT |
| TASKSTATUS | TEXT |
| MODIFIEDBY | TEXT |
| CREATEDBY | TEXT |
| LASTMODIFIED | TIMESTAMP |
| CREATED | TIMESTAMP |

**DATASET**

| | |
|---|---|
| IDENTIFIER | INT |
| LOGICALDATASETNAME | TEXT |
| PROJECTNAME | TEXT |
| VERSION | TEXT |
| STREAMNAME | TEXT |
| NFILES | INT |
| TOTALEVENTS | INT |
| CREATIONCOMMENT | TEXT |
| PRODSTEP | TEXT |
| DATATYPE | TEXT |
| TRASHTRIGGER | TEXT |
| TRASHANNOTATION | TEXT |
| TRASHDATE | TIMESTAMP |
| TRASHEDBY | TEXT |
| AMISTATUS | TEXT |
| CREATEDBY | TEXT |
| MODIFIEDBY | TEXT |
| CREATED | TIMESTAMP |
| LASTMODIFIED | TIMESTAMP |
| RUNNUMBER | TEXT |
| REQUESTEDBY | TEXT |
| TOTALSIZE | INT |
| GID | INT |

**HASHTAGS**

| | |
|---|---|
| HASHTAGID | INT |
| GLOBALDATASETFK | INT |
| DATASETCATALOG | TEXT |
| DATASETNAME | TEXT |
| DATASETFK | INT |
| SCOPEFK | INT |
| FULLNAME | TEXT |
| SCOPE | TEXT |
| NAME | TEXT |
| COMMENT | TEXT |
| ARCHIVED | INT |
| CREATED | TIMESTAMP |
| CREATEDBY | TEXT |
| MODIFIED | TIMESTAMP |
| MODIFIEDBY | TEXT |

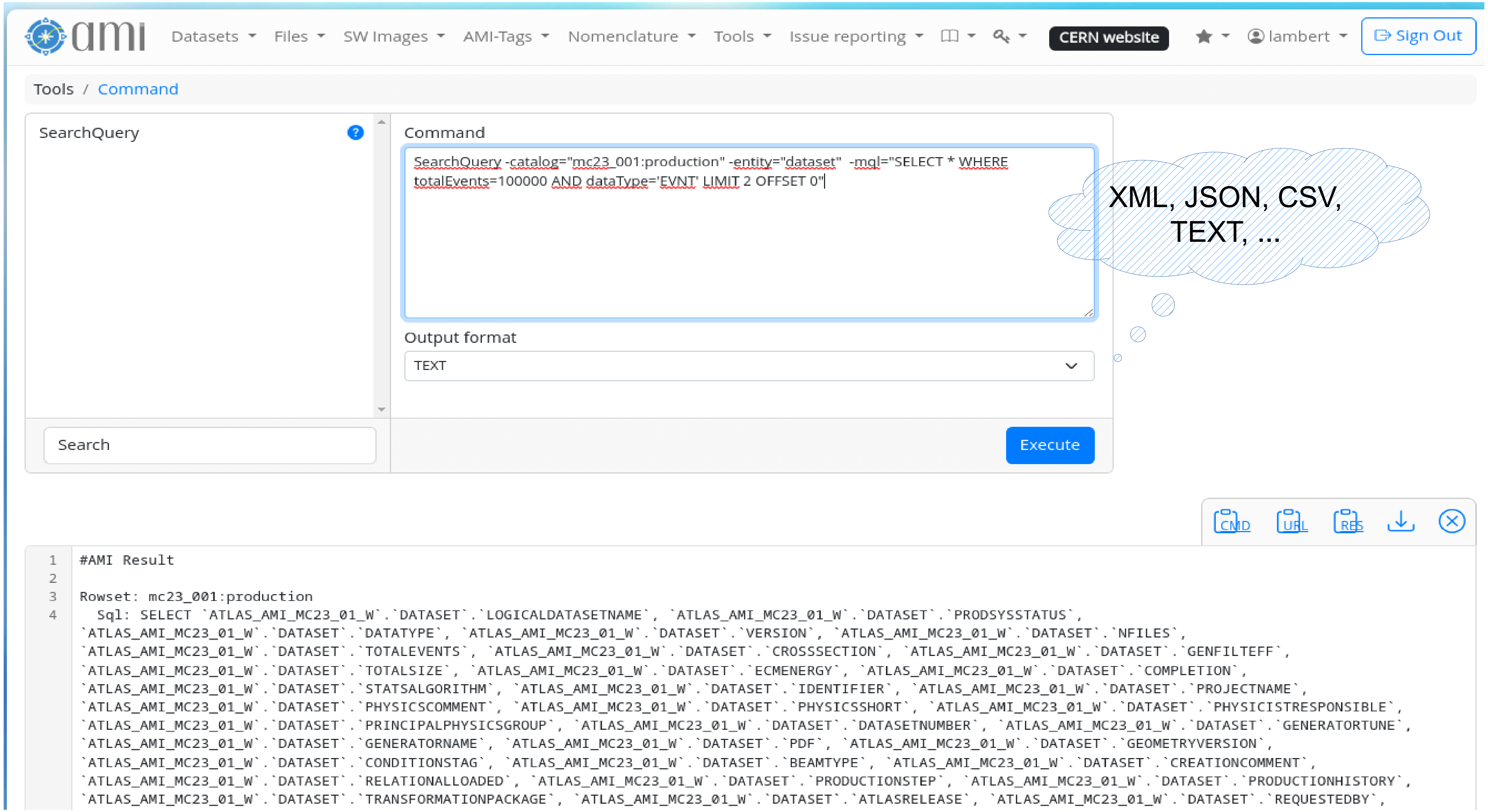
**DATASET_COMMENT**

| | |
|---|---|
| IDENTIFIER | INT |
| DATASETFK | INT |
| TEXT | TEXT |
| CREATEDBY | TEXT |
| MODIFIEDBY | TEXT |
| CREATED | TIMESTAMP |

**CONTAINED_DATASET**

| | |
|---|---|
| IDENTIFIER | INT |
| DATASETFK | INT |
| CONTAINED_DATASETN... | TEXT |
| AMIPROJECTNAME | TEXT |
| AMIPROCESSNAME | TEXT |
| CREATEDBY | TEXT |
| MODIFIEDBY | TEXT |

# Interacting with AMI from the Web



Command

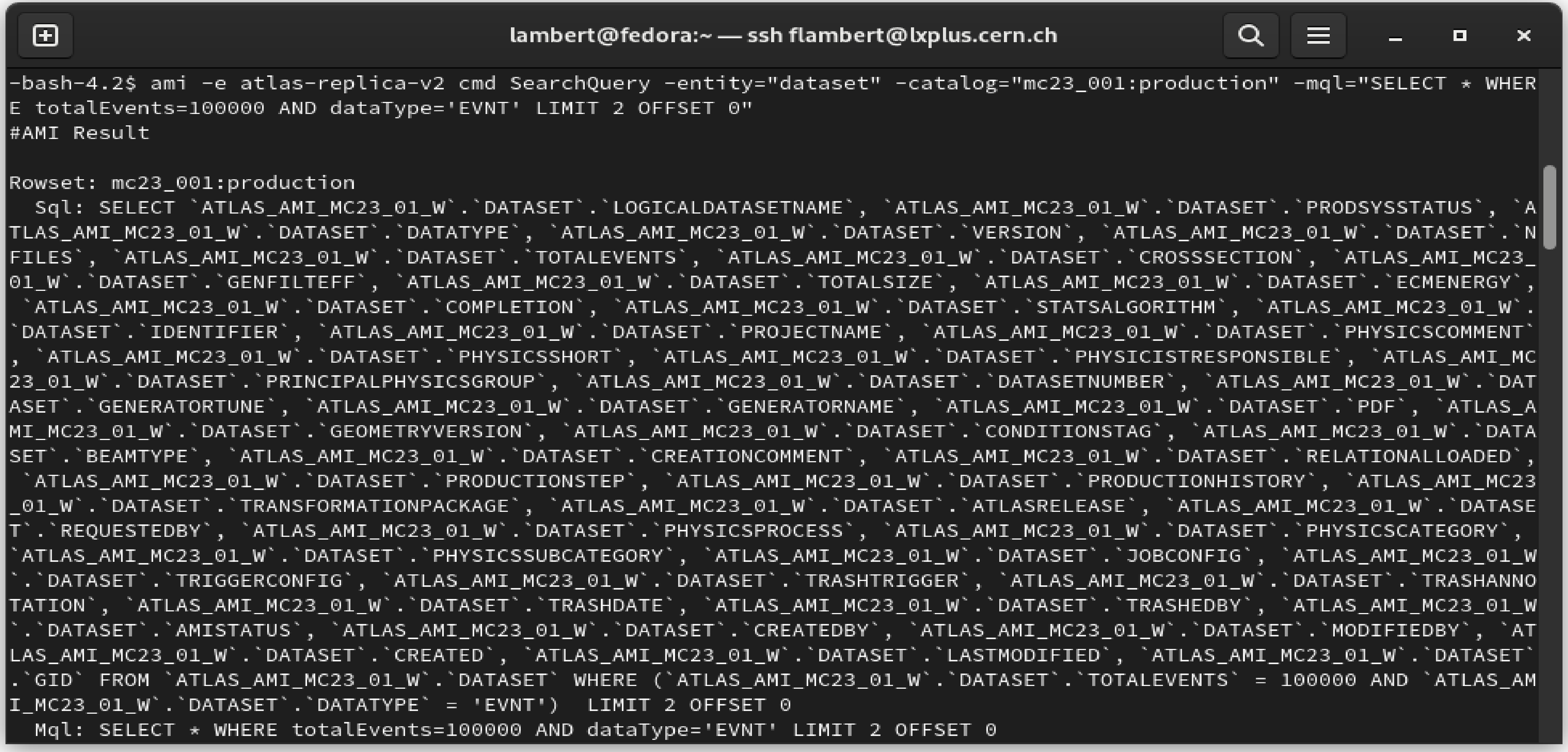
```
SearchQuery -catalog="mc23_001:production" -entity="dataset"  -mql="SELECT * WHERE totalEvents=100000 AND dataType='EVNT' LIMIT 2 OFFSET 0"
```

Output format: TEXT

```
#AMI Result

Rowset: mc23_001:production
  Sql: SELECT `ATLAS_AMI_MC23_01_W`.`DATASET`.`LOGICALDATASETNAME`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODSYSSTATUS`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`DATATYPE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`VERSION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`NFILES`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`TOTALEVENTS`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CROSSSECTION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`GENFILTEFF`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`TOTALSIZE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`ECMENERGY`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`COMPLETION`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`STATSALGORITHM`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`IDENTIFIER`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PROJECTNAME`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSCOMMENT`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSSHORT`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICISTRESPONSIBLE`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`PRINCIPALPHYSICSGROUP`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`DATASETNUMBER`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`GENERATORTUNE`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`GENERATORNAME`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PDF`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`GEOMETRYVERSION`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`CONDITIONSTAG`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`BEAMTYPE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CREATIONCOMMENT`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`RELATIONALLOADED`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODUCTIONSTEP`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODUCTIONHISTORY`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`TRANSFORMATIONPACKAGE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`ATLASRELEASE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`REQUESTEDBY`,
```

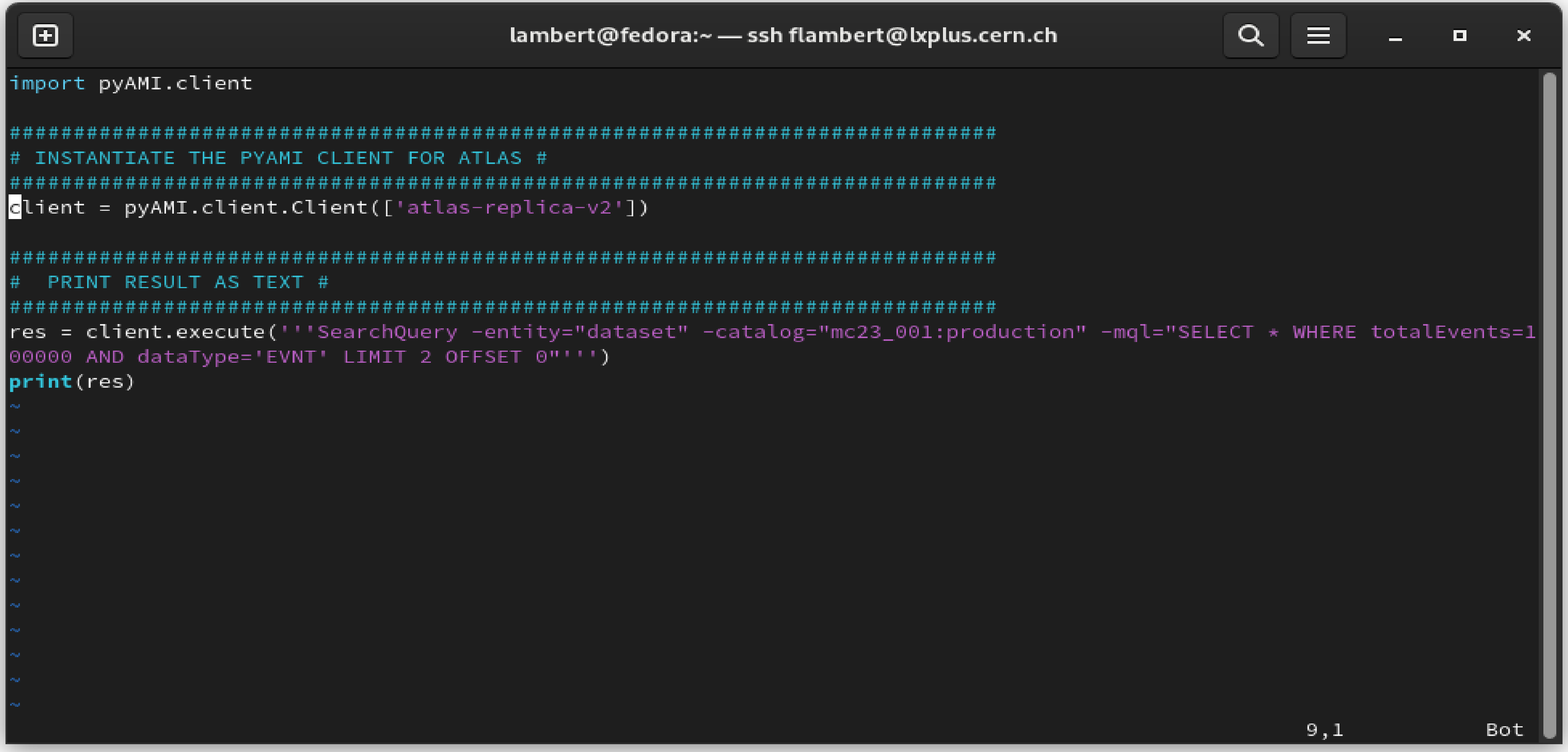
XML, JSON, CSV, TEXT, ...

# Interacting with AMI from a shell



```
-bash-4.2$ ami -e atlas-replica-v2 cmd SearchQuery -entity="dataset" -catalog="mc23_001:production" -mql="SELECT * WHER
E totalEvents=100000 AND dataType='EVNT' LIMIT 2 OFFSET 0"
#AMI Result

Rowset: mc23_001:production
  Sql: SELECT `ATLAS_AMI_MC23_01_W`.`DATASET`.`LOGICALDATASETNAME`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODSYSSTATUS`, `A
TLAS_AMI_MC23_01_W`.`DATASET`.`DATATYPE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`VERSION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`N
FILES`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TOTALEVENTS`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CROSSSECTION`, `ATLAS_AMI_MC23_
01_W`.`DATASET`.`GENFILTEFF`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TOTALSIZE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`ECMENERGY`,
 `ATLAS_AMI_MC23_01_W`.`DATASET`.`COMPLETION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`STATSALGORITHM`, `ATLAS_AMI_MC23_01_W`.
`DATASET`.`IDENTIFIER`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PROJECTNAME`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSCOMMENT`
, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSSHORT`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICISTRESPONSIBLE`, `ATLAS_AMI_MC
23_01_W`.`DATASET`.`PRINCIPALPHYSICSGROUP`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`DATASETNUMBER`, `ATLAS_AMI_MC23_01_W`.`DAT
ASET`.`GENERATORTUNE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`GENERATORNAME`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PDF`, `ATLAS_A
MI_MC23_01_W`.`DATASET`.`GEOMETRYVERSION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CONDITIONSTAG`, `ATLAS_AMI_MC23_01_W`.`DATA
SET`.`BEAMTYPE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CREATIONCOMMENT`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`RELATIONALLOADED`,
 `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODUCTIONSTEP`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PRODUCTIONHISTORY`, `ATLAS_AMI_MC23
_01_W`.`DATASET`.`TRANSFORMATIONPACKAGE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`ATLASRELEASE`, `ATLAS_AMI_MC23_01_W`.`DATASE
T`.`REQUESTEDBY`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSPROCESS`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSCATEGORY`,
`ATLAS_AMI_MC23_01_W`.`DATASET`.`PHYSICSSUBCATEGORY`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`JOBCONFIG`, `ATLAS_AMI_MC23_01_W
`.`DATASET`.`TRIGGERCONFIG`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TRASHTRIGGER`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TRASHANNO
TATION`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TRASHDATE`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`TRASHEDBY`, `ATLAS_AMI_MC23_01_W
`.`DATASET`.`AMISTATUS`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`CREATEDBY`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`MODIFIEDBY`, `AT
LAS_AMI_MC23_01_W`.`DATASET`.`CREATED`, `ATLAS_AMI_MC23_01_W`.`DATASET`.`LASTMODIFIED`, `ATLAS_AMI_MC23_01_W`.`DATASET`
.`GID` FROM `ATLAS_AMI_MC23_01_W`.`DATASET` WHERE (`ATLAS_AMI_MC23_01_W`.`DATASET`.`TOTALEVENTS` = 100000 AND `ATLAS_AM
I_MC23_01_W`.`DATASET`.`DATATYPE` = 'EVNT')  LIMIT 2 OFFSET 0
  Mql: SELECT * WHERE totalEvents=100000 AND dataType='EVNT' LIMIT 2 OFFSET 0
```

# Interacting with AMI from a python script



```
import pyAMI.client

#############################################################################
# INSTANTIATE THE PYAMI CLIENT FOR ATLAS #
#############################################################################
client = pyAMI.client.Client(['atlas-replica-v2'])

#############################################################################
#  PRINT RESULT AS TEXT #
#############################################################################
res = client.execute('''SearchQuery -entity="dataset" -catalog="mc23_001:production" -mql="SELECT * WHERE totalEvents=1
00000 AND dataType='EVNT' LIMIT 2 OFFSET 0"''')
print(res)
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                                                                          9,1           Bot
```

AMI Frontend

04

# Metadata aggregation

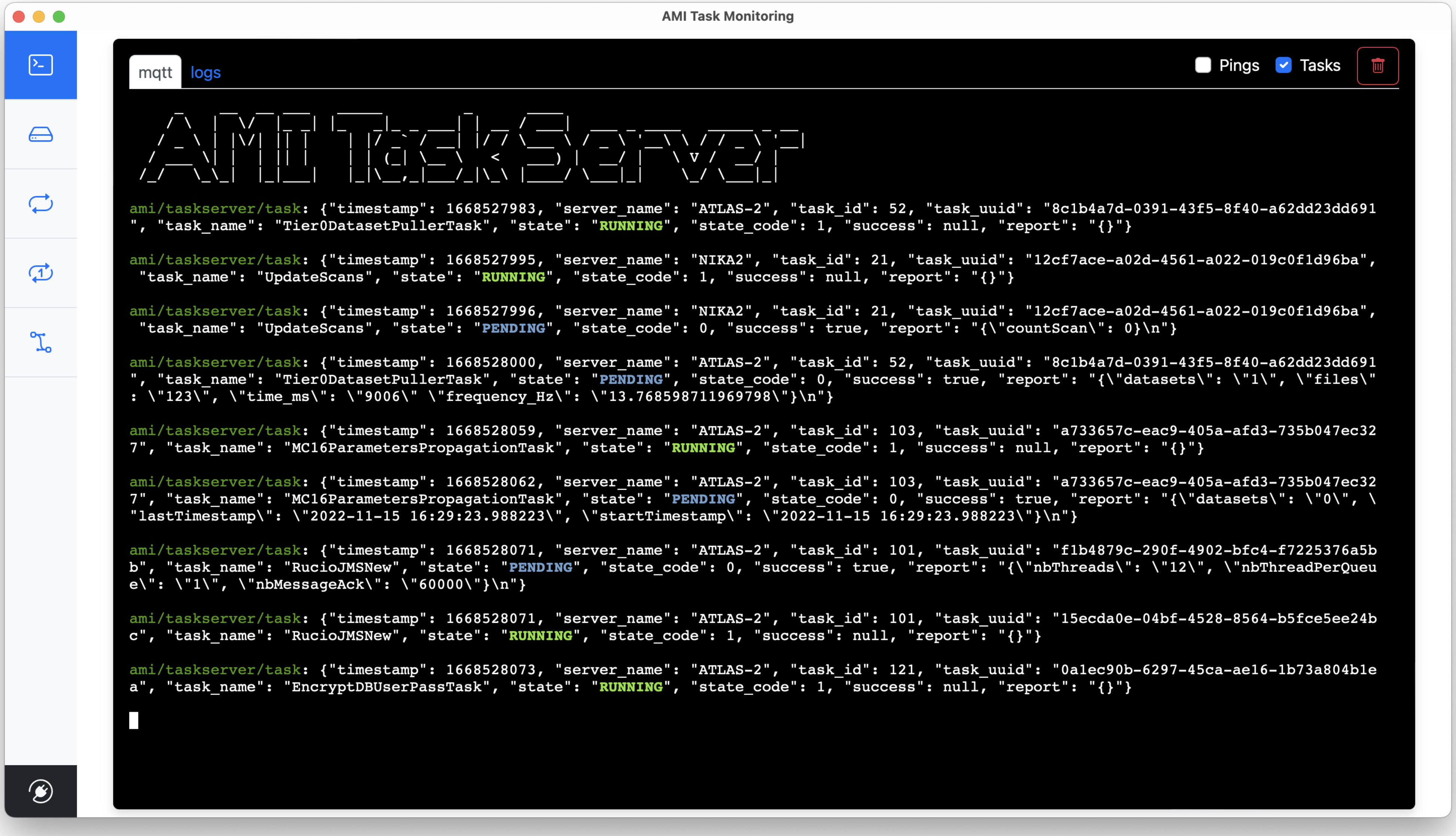
ami

# AMI Task Server

- **AMI Task server**
  - Extracting metadata from primary sources (pull mode)
  - (Re)Processing and storing metadata in AMI
- **It can run any kind of tasks (Shell, Python, JS, C++, Java, …)**
  - Optionally benefits from the AMI Java Core library
- **Main features:**
  - Kind of super CRON
  - The AMI Task Server is distributed
  - Control and monitoring capabilities via MQTT
  - Mutual exclusion mechanism between tasks (with the AMI Exclusion Server)
  - Priority lottery scheduler for avoiding starvation (not real time)
  - Pipelined tasks with execution report

Metadata Aggregation

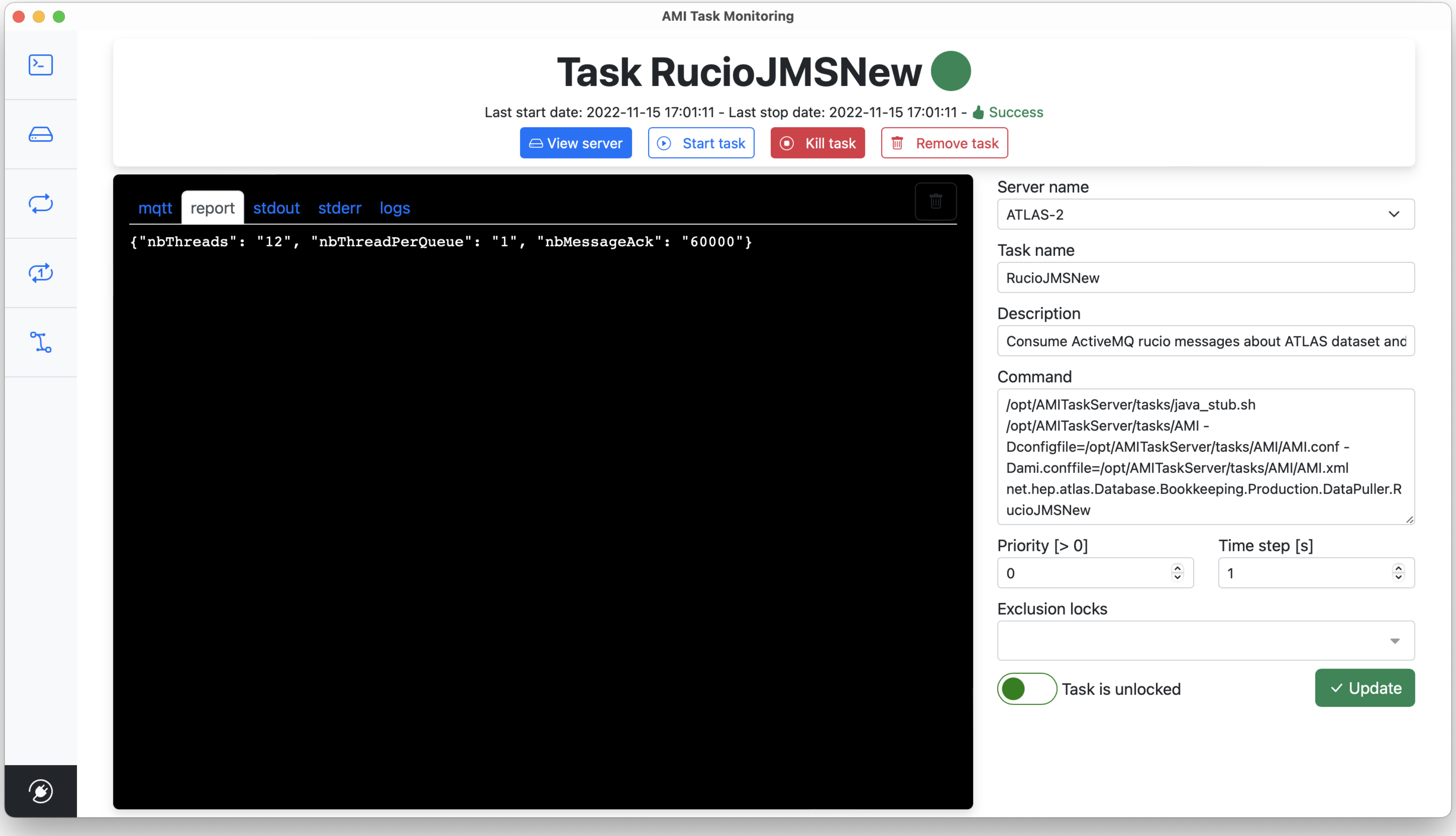
# AMI Task Server



□ Pings ☑ Tasks

```
ami/taskserver/task: {"timestamp": 1668527983, "server_name": "ATLAS-2", "task_id": 52, "task_uuid": "8c1b4a7d-0391-43f5-8f40-a62dd23dd691
", "task_name": "Tier0DatasetPullerTask", "state": "RUNNING", "state_code": 1, "success": null, "report": "{}"}

ami/taskserver/task: {"timestamp": 1668527995, "server_name": "NIKA2", "task_id": 21, "task_uuid": "12cf7ace-a02d-4561-a022-019c0f1d96ba",
 "task_name": "UpdateScans", "state": "RUNNING", "state_code": 1, "success": null, "report": "{}"}

ami/taskserver/task: {"timestamp": 1668527996, "server_name": "NIKA2", "task_id": 21, "task_uuid": "12cf7ace-a02d-4561-a022-019c0f1d96ba",
 "task_name": "UpdateScans", "state": "PENDING", "state_code": 0, "success": true, "report": "{\"countScan\": 0}\n"}

ami/taskserver/task: {"timestamp": 1668528000, "server_name": "ATLAS-2", "task_id": 52, "task_uuid": "8c1b4a7d-0391-43f5-8f40-a62dd23dd691
", "task_name": "Tier0DatasetPullerTask", "state": "PENDING", "state_code": 0, "success": true, "report": "{\"datasets\": \"1\", \"files\"
: \"123\", \"time_ms\": \"9006\" \"frequency_Hz\": \"13.768598711969798\"}\n"}

ami/taskserver/task: {"timestamp": 1668528059, "server_name": "ATLAS-2", "task_id": 103, "task_uuid": "a733657c-eac9-405a-afd3-735b047ec32
7", "task_name": "MC16ParametersPropagationTask", "state": "RUNNING", "state_code": 1, "success": null, "report": "{}"}

ami/taskserver/task: {"timestamp": 1668528062, "server_name": "ATLAS-2", "task_id": 103, "task_uuid": "a733657c-eac9-405a-afd3-735b047ec32
7", "task_name": "MC16ParametersPropagationTask", "state": "PENDING", "state_code": 0, "success": true, "report": "{\"datasets\": \"0\", \
"lastTimestamp\": \"2022-11-15 16:29:23.988223\", \"startTimestamp\": \"2022-11-15 16:29:23.988223\"}\n"}

ami/taskserver/task: {"timestamp": 1668528071, "server_name": "ATLAS-2", "task_id": 101, "task_uuid": "f1b4879c-290f-4902-bfc4-f7225376a5b
b", "task_name": "RucioJMSNew", "state": "PENDING", "state_code": 0, "success": true, "report": "{\"nbThreads\": \"12\", \"nbThreadPerQueu
e\": \"1\", \"nbMessageAck\": \"60000\"}\n"}

ami/taskserver/task: {"timestamp": 1668528071, "server_name": "ATLAS-2", "task_id": 101, "task_uuid": "15ecda0e-04bf-4528-8564-b5fce5ee24b
c", "task_name": "RucioJMSNew", "state": "RUNNING", "state_code": 1, "success": null, "report": "{}"}

ami/taskserver/task: {"timestamp": 1668528073, "server_name": "ATLAS-2", "task_id": 121, "task_uuid": "0a1ec90b-6297-45ca-ae16-1b73a804b1e
a", "task_name": "EncryptDBUserPassTask", "state": "RUNNING", "state_code": 1, "success": null, "report": "{}"}
```

# AMI Task Server



## Task RucioJMSNew ●

Last start date: 2022-11-15 17:01:11 - Last stop date: 2022-11-15 17:01:11 - Success

```
{"nbThreads": "12", "nbThreadPerQueue": "1", "nbMessageAck": "60000"}
```

Server name: ATLAS-2

Task name: RucioJMSNew

Description: Consume ActiveMQ rucio messages about ATLAS dataset and

Command:

```
/opt/AMITaskServer/tasks/java_stub.sh
/opt/AMITaskServer/tasks/AMI -
Dconfigfile=/opt/AMITaskServer/tasks/AMI/AMI.conf -
Dami.conffile=/opt/AMITaskServer/tasks/AMI/AMI.xml
net.hep.atlas.Database.Bookkeeping.Production.DataPuller.R
ucioJMSNew
```

Priority [> 0]: 0

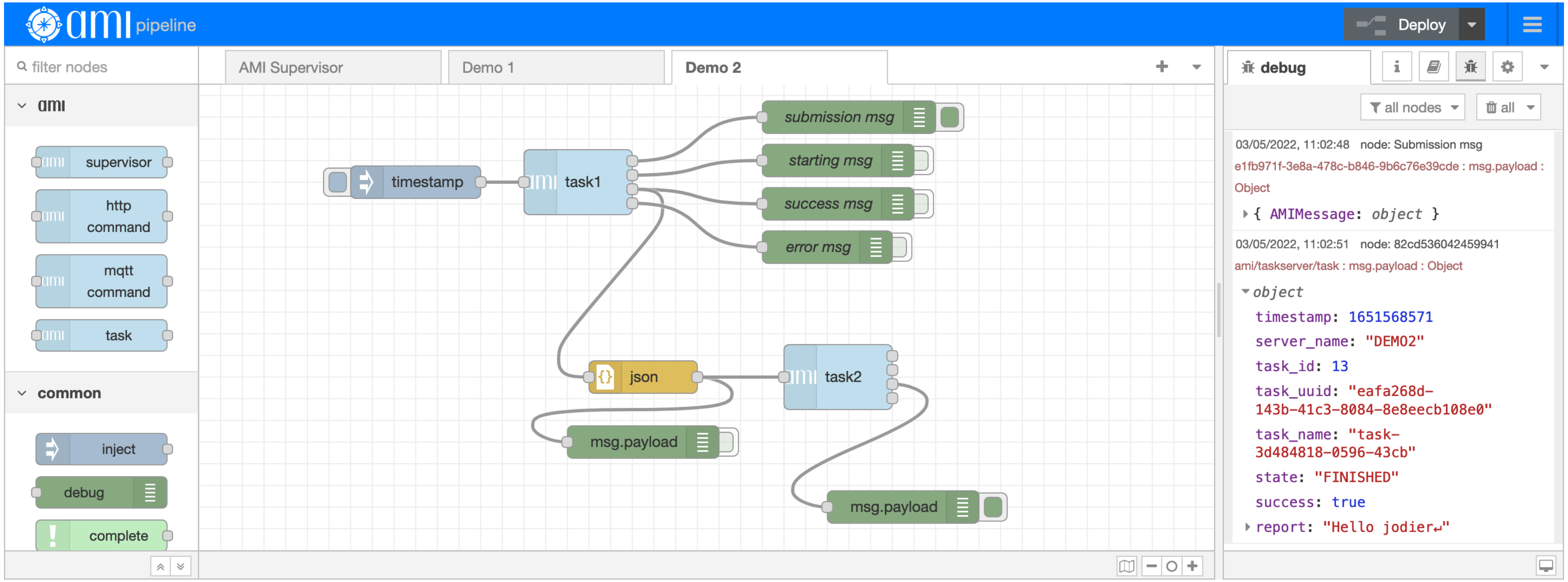
Time step [s]: 1

Exclusion locks:

Task is unlocked

# Pipelined tasks

- AMI provides a Node-RED-based interface for defining and monitoring pipelined tasks
- No single point of failure
- CHEP 2023 paper:
  - https://cds.cern.ch/record/2867330/files/ATL-SOFT-PROC-2023-006.pdf



```
03/05/2022, 11:02:48   node: Submission msg
e1fb971f-3e8a-478c-b846-9b6c76e39cde : msg.payload :
Object
 ▸{ AMIMessage: object }
03/05/2022, 11:02:51   node: 82cd536042459941
ami/taskserver/task : msg.payload : Object
 ▾object
   timestamp: 1651568571
   server_name: "DEMO2"
   task_id: 13
   task_uuid: "eafa268d-
   143b-41c3-8084-8e8eecb108e0"
   task_name: "task-
   3d484818-0596-43cb"
   state: "FINISHED"
   success: true
  ▸report: "Hello jodier↵"
```

05

# AMI
# live demo

AMI live demo

# AMI live demo

Run AMI on your laptop

https://github.com/ami-team/AMIDemo

## http://localhost:667/

# Conclusion

ami

# Conclusion

- **AMI is mature metadata ecosystem of more than 20 years of existence in the LHC context:**
  - AMI and a data movement tool (for example Rucio) are complementary tools with different purposes
  - AMI is easy to deploy and rescale (docker images), administrate (low code application design), and use
- **Benefits of using AMI in your experiment:**
  - AMI is pluggable to any existing production system (databases and Web services)
    - ➔ AMI is not intrusive
  - AMI aggregates low-level metadata into high-level metadata for physicists
    - Housekeeping, dataset and file, end-user, software, ... metadata
  - AMI provides Web services, interfaces, and clients (Python, JS, C++, Java) for easily select data by metadata criteria
  - AMI can associate data and papers for data preservation & reproducibility
- **Prerequisites: having well-defined low level metadata at data production level**

# Thank You

for your attention

LPSC - Grenoble

ami@lpsc.in2p3.fr
https://ami-ecosystem.in2p3.fr/

ami

ami