

FAIR Data Spaces Final Event

Demonstrator 4.2: Dataset Validation / Quality Assurance

Jonathan Hartman, *Data Scientist/Consultant*
RWTH Aachen University IT Center



Motivation

Datasets from diverse disciplines

- *Data generated by automated processes*
 - *Datasets generated by hand*
- *Data from imperfect processes*

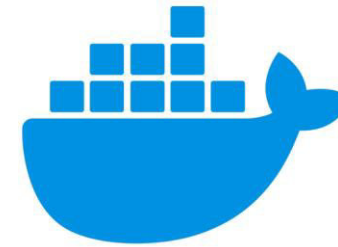
Automated Process for Validating Data

Goals

- Leverage existing technologies
 - GitLab
 - Open Telekom Cloud
- Access data in external storage
- Connect to HPC systems



GitLab



docker®



python™

Demonstrator Components

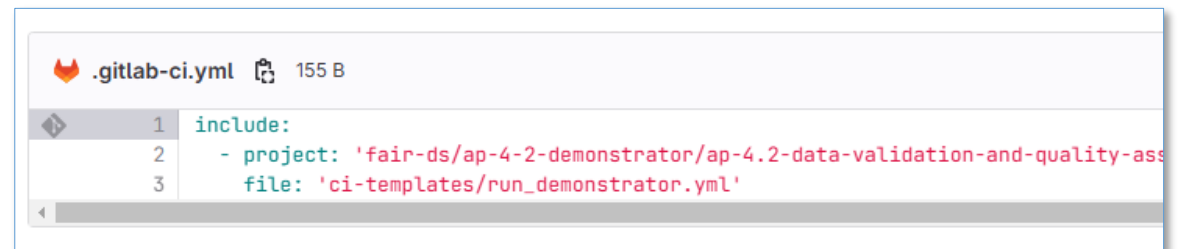
- Main demonstrator
 - Python library
 - Docker container
 - Example projects
- Workflow containers
 - Open for contribution
 - Currently 10 container projects
- HPC connector

Demonstrator Requirements - Setup

- GitLab repository
 - settings.toml
 - “Where is the data?”
 - “How should I load the data?”
 - Schemas
 - “What are your expectations of the data?”
 - Data files
 - “What am I looking at?”
 - .gitlab-ci.yml
 - “How do I run the project?”



```
settings.toml 362 B
1 [PROJECT]
2 name = "Example Project: New Zealand Environmental Studies"
3
4
5 [STORAGE]
6 [STORAGE.URL]
7 url = "https://www.stats.govt.nz/assets/Uploads/Greenhouse"
```



```
.gitlab-ci.yml 155 B
1 include:
2 - project: 'fair-ds/ap-4-2-demonstrator/ap-4.2-data-validation-and-quality-ass
3 file: 'ci-templates/run_demonstrator.yml'
```

Demonstrator Requirements – Data Files

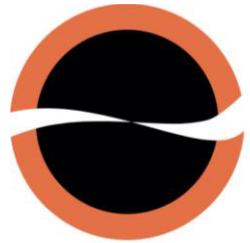
- Tabular data formats
 - CSV
 - Parquet
- Storage options:
 - Local
 - S3
 - URL
 - Coscine

Demonstrator Requirements - Settings

```
example_settings.toml 531 B
1 [PROJECT]
2 name = "Example Project"
3
4 [DIRECTORIES]
5 root = ""
6 data = "data"
7 schemas = "schemas"
8
9 [FILE_PATTERNS]
10 data_file_patterns = [".*\\.csv", ".*\\.parquet"]
11 schema_file_patterns = [".*\\.json"]
12
13 [S3]
14 bucket = ""
15
16 [GEODATA]
17 crs = "EPSG:4326"
18 shape_file = "naturalearth_lowres"
19
20 [FILE_DETAILS]
21 [[FILE_DETAILS.FILE]]
22     filename = "data/file1.csv"
23     delimiter = "\t"
24     encoding = "cp1252"
25 [[FILE_DETAILS.FILE]]
26     filename = "data/file2.csv"
27     latitude = "latitude_column_name2"
28     longitude = "longitude_column_name2"
29
```

- Extra details about the project to be included in the report
- Specific locations for data / schema files
- File specific directives

Demonstrator Requirements - Schemas



FRICITIONLESS
DATA

```
{  
  "fields": [  
    {  
      "name": "Survived",  
      "type": "boolean",  
      "trueValues": ["1"],  
      "falseValues": ["0"]  
    },  
    {  
      "name": "Pclass",  
      "type": "integer",  
      "constraints": {  
        "required": true,  
        "enum": [1, 2, 3]  
      }  
    }  
  ],  
}
```

- Based on the „Frictionless“ standard
- Matched with data files
 - Can also be specified
- Define types
- Specify constraints
- Refer to ontologies

Demonstrator Requirements - Schemas



FRICITIONLESS
DATA

```
{  
  "name": "Name",  
  "type": "string",  
  "constraints": {  
    "required": true,  
    "unique": true,  
    "minLength": 5,  
    "maxLength": 100,  
    "pattern": "^[A-Z][a-z]+\\. (.*)$"  
  }  
},
```

- Based on the „Frictionless“ standard
- Matched with data files
 - Can also be specified
- Define types
- Specify constraints
- Refer to ontologies

Demonstrator Requirements - Schemas



FRICITIONLESS
DATA

```
{  
  "name": "Sex",  
  "type": "string",  
  "ontology": "abcd::Sex",  
  "constraints": {  
    "required": true  
  }  
},
```

- Based on the „Frictionless“ standard
- Matched with data files
 - Can also be specified
- Define types
- Specify constraints
- Refer to ontologies

Demonstrator Requirements - Schemas

```
{  
  "name": "Sex",  
  "type": "string",  
  "ontology": "abcd::Sex",  
  "constraints": {  
    "required": true  
  }  
},
```

The screenshot shows the TIB Terminology Service interface. At the top, there's a search bar and navigation options like 'Exact match', 'Obsolete terms', and 'Advanced search'. Below that, the 'ABCD Base Ontology' is displayed with a class tree on the left and a detailed view of the 'Sex' term on the right. The detailed view includes fields for 'Label', 'Term ID', 'Description', 'fullIRI', and 'Instances'.

```
{  
  "name": "Sex",  
  "type": "string",  
  "ontology": "abcd::Sex",  
  "constraints": {  
    "required": true,  
    "enum": [  
      "Male",  
      "Female",  
      "SexUnknown",  
      "SexNotApplicable",  
      "MixedSex"  
    ]  
  }  
},
```

The screenshot shows the TIB Terminology Service homepage. It features a search bar at the top and a navigation menu with options like 'HOME', 'COLLECTIONS', 'ONTOLOGIES', 'HELP', 'API', and 'ABOUT'. Below the search bar, there's a section titled 'TIB Terminology Service' with a brief description. Further down, there's a 'Collections' section with three featured collections: NFDI4ing, NFDI4Chem, and CnFdi.

Running the Demonstrator

- Triggered as a CI/CD Script
 - Repository Changes
 - Scheduled
 - cURL via API Endpoint
 - Webhook

Workflow Containers

- Docker Containers
 - Making containers broadly available as a service
- Current Maintained Containers
 - AP4.2 Demonstrator
 - AP4.2 Demonstrator (QC and DV only)
 - Python/R/Julia w/ Frictionless
 - Emacs
- Open for Contributions
 - Only Maintainers can greenlight containers

HPC Connection

- Facilitated by AixCilenx CI Driver
 - by Adrian Schmitz and Felix Tomski
- Allows containers and scripts to be seamlessly executed on HPC infrastructure via GitLab CI/CD Scripts

```
1 Running with gitlab-runner 16.1.0 (b72e108d)
2   on RPDM 1MiH2arzS, system ID: s_cdfd3a24fed3
3   ✓ Resolving secrets
4   ✓ Preparing the "custom" executor
5   Using Custom executor with driver AixCilenx CI Driver 0.6.0...
6   ✓ Preparing environment
7   Running on n23m0055.hpc.itc.rwth-aachen.de via custom-hostname...
8   > Getting source from Git repository
25  > Downloading artifacts
38  ✓ Executing "step_script" stage of the job script
39  WARNING: Starting with version 17.0 the 'build_script' stage will be replaced with 'step_s
40  $ module load GCCcore/.13.3.0
41  [INFO] Module binutils/2.38 loaded.
42  [INFO] Module zlib/1.2.13 loaded.
43  [INFO] Module GCCcore/.13.3.0 loaded.
44  [INFO] Module UCX/1.17.0 loaded.
45  Due to MODULEPATH changes, the following have been reloaded:
46  1) UCX/1.17.0    2) binutils/2.38
47  The following have been reloaded with a version change:
48  1) GCCcore/.11.3.0 => GCCcore/.13.3.0    3) zlib/1.2.13 => zlib/1.3.1
49  2) numactl/2.0.16 => numactl/2.0.18
50  $ module load Python/3.12.3
51  [INFO] Module Python/3.12.3 loaded.
52  The following have been reloaded with a version change:
53  1) binutils/2.38 => binutils/2.42
```

Thank you

Please feel free to ask questions at the link below, or you can also contact me directly at *hartman@itc.rwth-aachen.de*



https://pad.otc.coscine.dev/2024-12-03_fair-ds-qa

<https://tinyurl.com/fairds1203>

Thank you for your interest!

