

# The SQAaaS platform

Pablo Orviz (IFCA, CSIC) - on behalf of EOSC-Synergy WP3

# SQAaaS goals



*Bring over novel quality practices close to researchers*

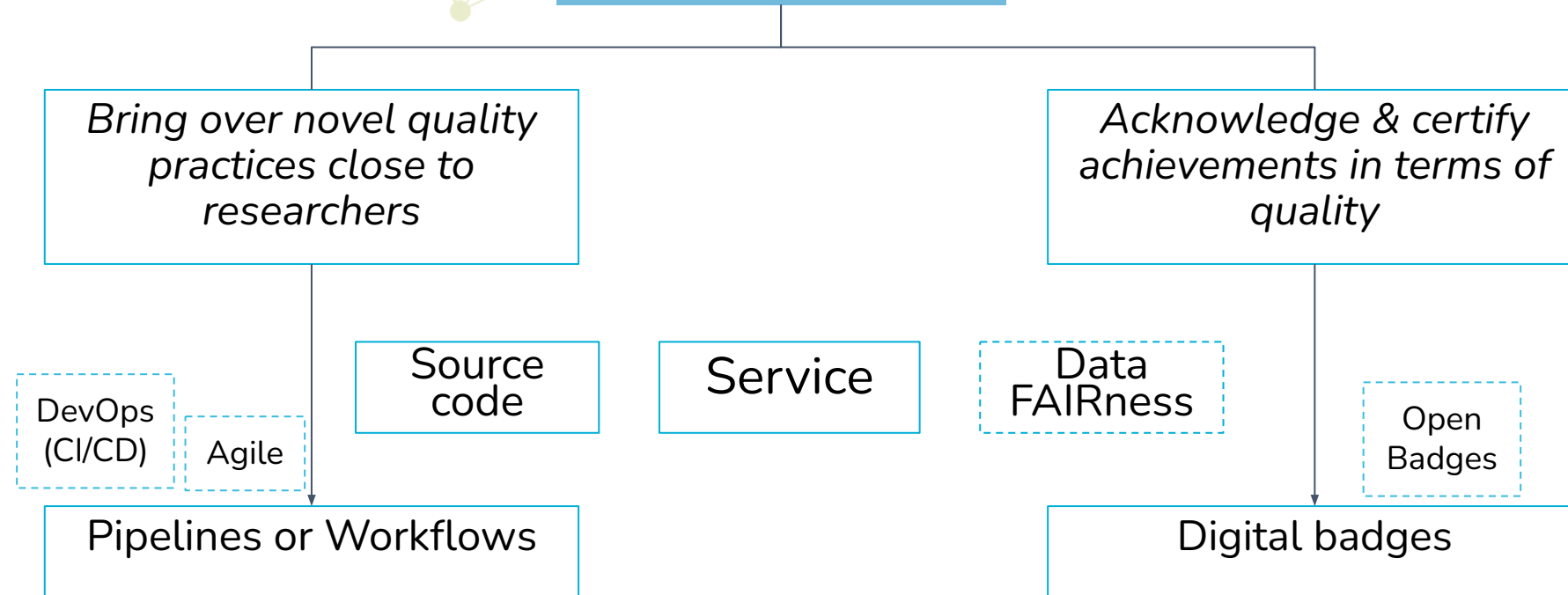
*Acknowledge & certify achievements in terms of quality*

Source code

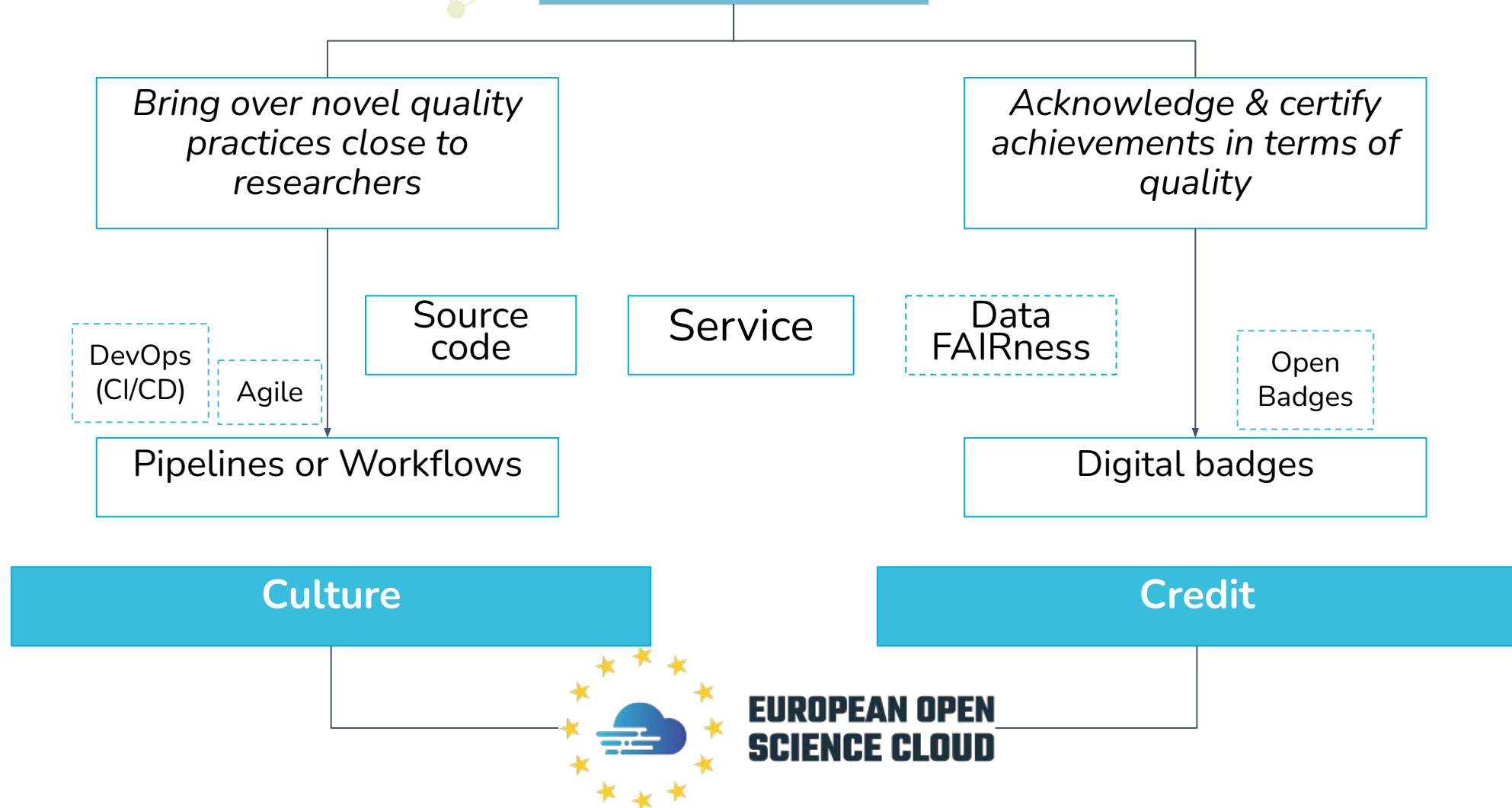
Service

Data FAIRness

# SQAaaS goals



# SQAaaS goals



# SQAaaS goals



*Bring over novel quality practices close to researchers*

*Acknowledge & certify achievements in terms of quality*

**Pipeline as a Service**

**Quality Assessment and Awarding**

DevOps (CI/CD)

Agile

Source code

Service

Data FAIRness

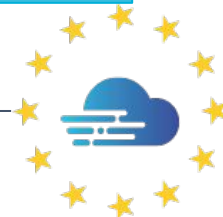
Open Badges

Pipelines or Workflows

Digital badges

**Culture**

**Credit**



**EUROPEAN OPEN  
SCIENCE CLOUD**

# Accessing SQAaaS platform

<https://sqaaas.eosc-synergy.eu>



EUROPEAN OPEN  
SCIENCE CLOUD  
EGI-Checkin

Straightforward to give it  
a try, open for every  
researcher

## SQAaaS module selection



### Pipeline as a Service

Compose customized CI/CD  
pipelines for your code  
repositories.



### Quality Assessment & Awarding

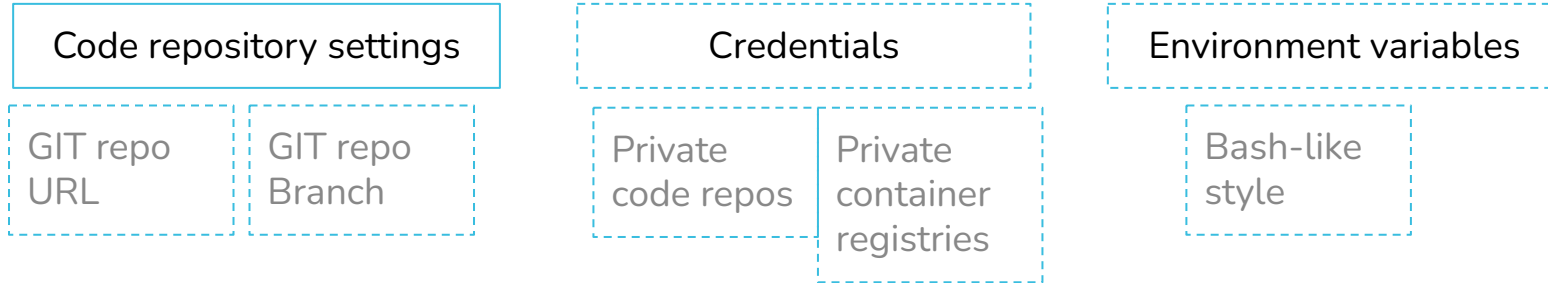
Take credit of the  
achievements in terms of  
software and service quality.

# SQAaaS: Pipeline as a Service

# The Pipeline as a Service

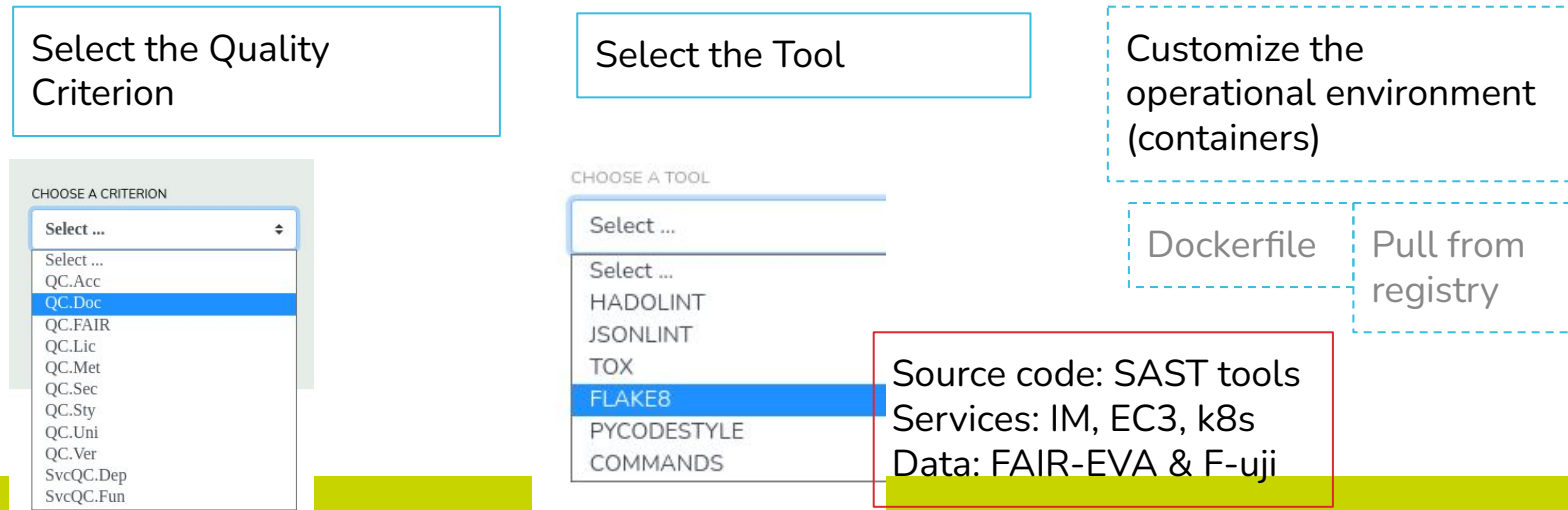
1

## Define the Repositories



2

## Add the Quality Criteria



Code & Service  
QA baselines

Features

Pipeline composition

JePL files

Pipeline sharing

Download  
& push to  
your code  
repo

PRs only  
for GitHub



Pipeline testing





# The Pipeline as a Service



SQAaaS

Software Quality Assurance as a Service

Pipeline as a Service

Compose and test your own customized quality pipelines

1 REPOS

2 SERVICES

3 CRITERIA

4 PIPELINE

< BACK

### Quality criteria define the CI/CD pipeline work

It is then the underpinning part where the pipeline's purpose takes shape. The associated properties for each criterion will be displayed once selected in the dropdown list below

CHOOSE A CRITERIA

QC.Sty

Select ...

QC.Sty

QC.Uni

QC.Fun

QC.Sec

QC.Doc

SELECT THE SERVICE

scipion-hadolint

Builder settings

According to the programming language in use, you can use for carrying out the work aligned with the given crit

CHOOSE A BUILDER TOOL

Select ...

### Your pipeline has been successfully created!

Download

Discover the additional features we provide

**Config summary**  
Provides a table-like view with the selections made when the pipeline was composed

**JePL files**  
Check out the files that drive the execution of the pipeline

**Pull request**  
Create a pull request to add the pipeline to your preferred repository  
[Github only](#)

**Try out**  
Execute the composed pipeline and check the results

www.eosc-synergy.eu - RIA 857647

8

# The JePL library



Pipeline customization

JePL

<https://github.com/indigo-dc/jenkins-pipeline-library>

SQAaaS does not provide a graphical way to modify existing pipelines

Fortunately, JePL simplifies the configuration of Jenkins pipelines

config.yml

```
config:
  node_agent: docker_compose
sqa_criteria:
  qc_style:
    repos:
      sqaaas-api-spec:
        container: node
        commands:
          - npm --prefix /sqaaas-api-spec install
          - npm --prefix /sqaaas-api-spec test
```

docker-compose.yml

```
version: "3.7"

services:
  node:
    image: node
    container_name: node
    hostname: "sqaaas-api-spec-node"
    volumes:
      - type: bind
        source: ./
        target: /sqaaas-api-spec
    command: sleep infinity
```

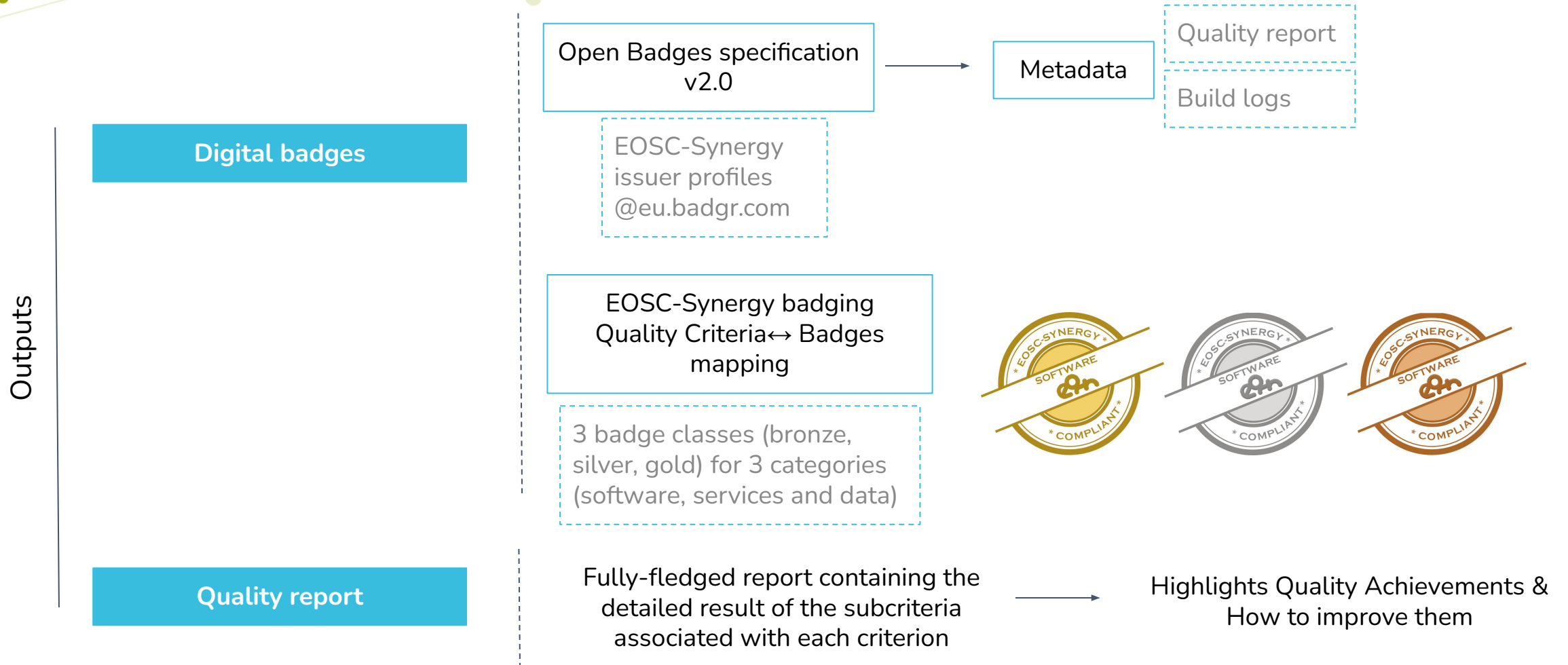
Jenkinsfile

```
pipeline {
  agent any

  stages {
    stage('OpenAPI linter') {
      steps {
        script {
          projectConfig = pipelineConfig(
            configFile: './.sqa/config_style.yml',
            scmConfigs: [ localBranch: true ]
          )
          buildStages(projectConfig)
        }
      }
    }
  }
}
```

# SQAaaS: Quality Assessment & Awarding

# The Quality Assessment & Awarding



# The Quality Assessment & Awarding

1

Trigger the Assessment

Source code

GIT code  
repo URL  
(& branch)

GIT doc  
repo URL  
(& branch)

Service (web)

GIT repo with  
deployment recipes (&  
branch)

Orchestration tool

Data

Persistent  
identifier

FAIRness  
tool

Matchmaking policies

2

Assessment results

Criteria breakdown

Subcriteria

ID

Evidence

Tools

Metrics



Documentation



QC.Doc06.1

Is the software scope outlined in the code repository?



A README file is present in the code repository

[More Info](#)



QC.Doc06.3

Does the project establish a code of conduct for its participants?



A CODE\_OF\_CONDUCT file is not present in the code repository

[More Info](#)



QC.Doc06.2

Is there a clear path or guidelines on how to contribute to the code?



A CONTRIBUTING file is not present in the code repository

[More Info](#)



QC.Doc02.X

Are docs following a style for the Markdown markup language?



Docs are not fully compliant with <markdownlint> standard

[More Info](#)

# The Quality Assessment & Awarding

Congratulations!!! the following badge/s have been awarded

[Learn more about the EOSC-Synergy badging approach](#)



[Verify](#)

[Go to Badgr's award page](#)

## Criteria Report



### Code Accessibility



**QC.Acc01** Is the source code managed with a Version Control System?

✓ Source code uses Git for version control

[More Info](#)



### Documentation



**QC.Doc06.1** Is the software scope outlined in the code repository?

✓ A README file is present in the code repository

[More Info](#)



**QC.Doc06.3** Does the project establish a code of conduct for its participants?

✓ A CODE\_OF\_CONDUCT file is present in the code repository

[More Info](#)

	Bronze	Silver	Gold
Accessibility ( QC.Acc )	✓	✓	✓
Code Management ( QC.Man )			✓
Code Metadata ( QC.Met )	✓	✓	✓
Code Style ( QC.Sty )		✓	✓
Code Workflow ( QC.Wor )			✓
Delivery ( QC.De l )			✓
Documentation ( QC.Doc )	✓	✓	✓
Licensing ( QC.Lic )	✓	✓	✓
Security Static Analysis ( QC.Sec )		✓	✓
Unit Testing ( QC.Uni )			✓
Versioning ( QC.Ver )		✓	✓



# Highlights

- Build culture: bring quality practices for software development into the research ecosystem
- Give credit: Proof of concept of a **quality certification tool** for software (incl. services) and data
  - **Digital badges**
    - Containing **metadata** about the assessment process
    - **Shared & Verifiable** (Open Badges v2.0)
- **Used by thematic services** (9) from EOSC-Synergy (specific presentation in this session)
  - ALL being delivered with a minimum quality (bronze) for code
  - AT LEAST 4 will achieve minimum quality (bronze) for services
- **Interest** from external research projects/communities:
  - InterTwin and DT-Geo EU-funded projects: V&V processes
  - EGI CSIRT: security testing
  - EOSC Infrastructures for Quality Research Software: QA baselines
- Be sure to try it out <https://sqaaas.eosc-synergy.eu/>
  - Docs at <https://docs.sqaaas.eosc-synergy.eu/>

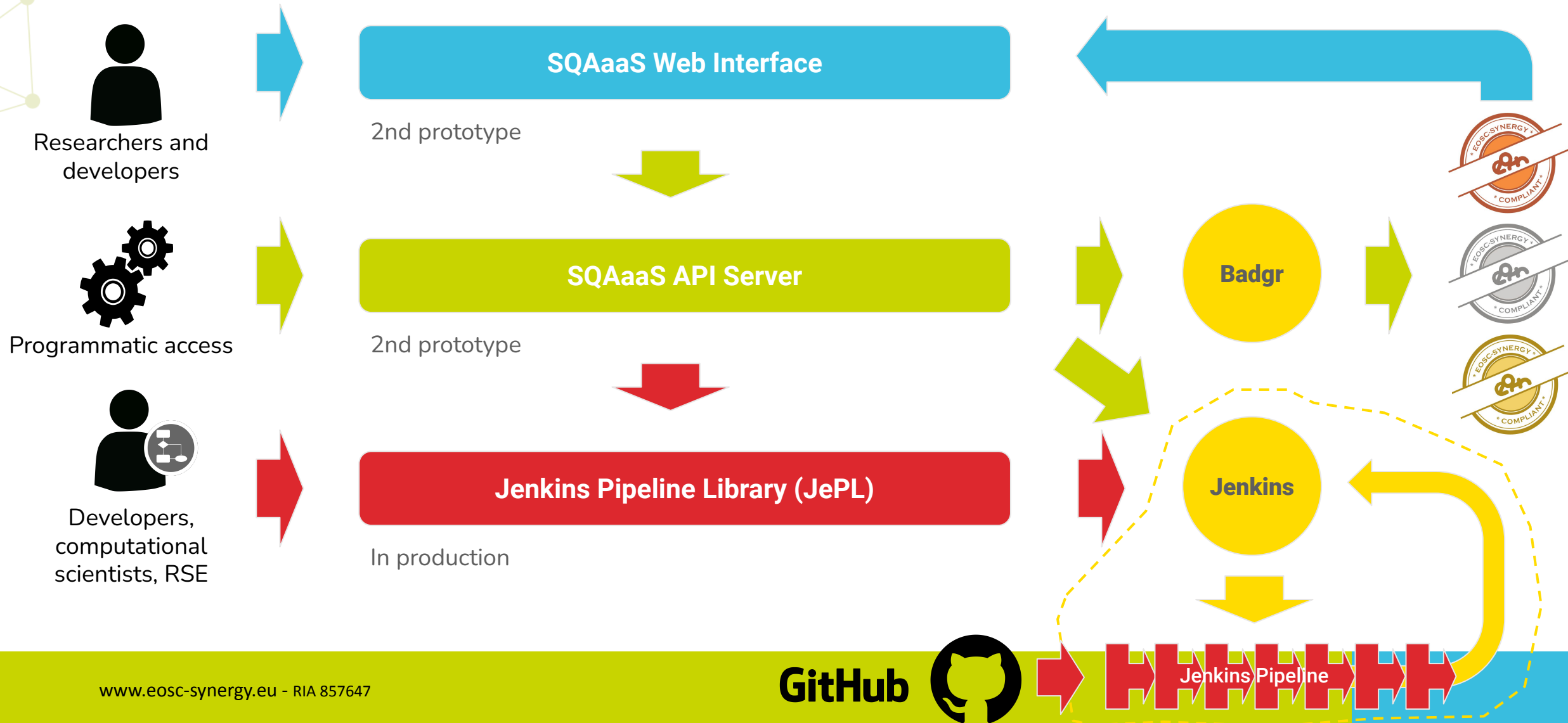
*Gracias !*  
*Obrigado !*  
*Danke !*  
*Dziękuję !*  
*Udaka !*  
*Dekuji !*  
*Bedankt !*  
*Merci !*  
*Thanks !*





# Backup slides

# SQAaaS: architecture and components



# SQAaaS breakdown: the library (JePL)



- SQAaaS core component
  - ⇒ **Implements the previous quality criteria**
    - Criterion-driven YAML config file (pipeline stage)
- Technology dependent
  - ⇒ Jenkins Pipeline as Code (PaC)
  - ⇒ Docker Compose for service orchestration

```
17 sqa_criteria:
18   qc_style:
19     repos:
20       o3api:
21         container: o3api-testing
22         tox:
23           tox_file: /o3api-testing/tox.ini
24           testenv:
25             - pep8
26   qc_coverage:
27     repos:
28       o3api:
29         container: o3api-testing
30         tox:
31           tox_file: /o3api-testing/tox.ini
32           testenv:
33             - unittest
34   qc_functional:
35     repos:
36       o3api:
```

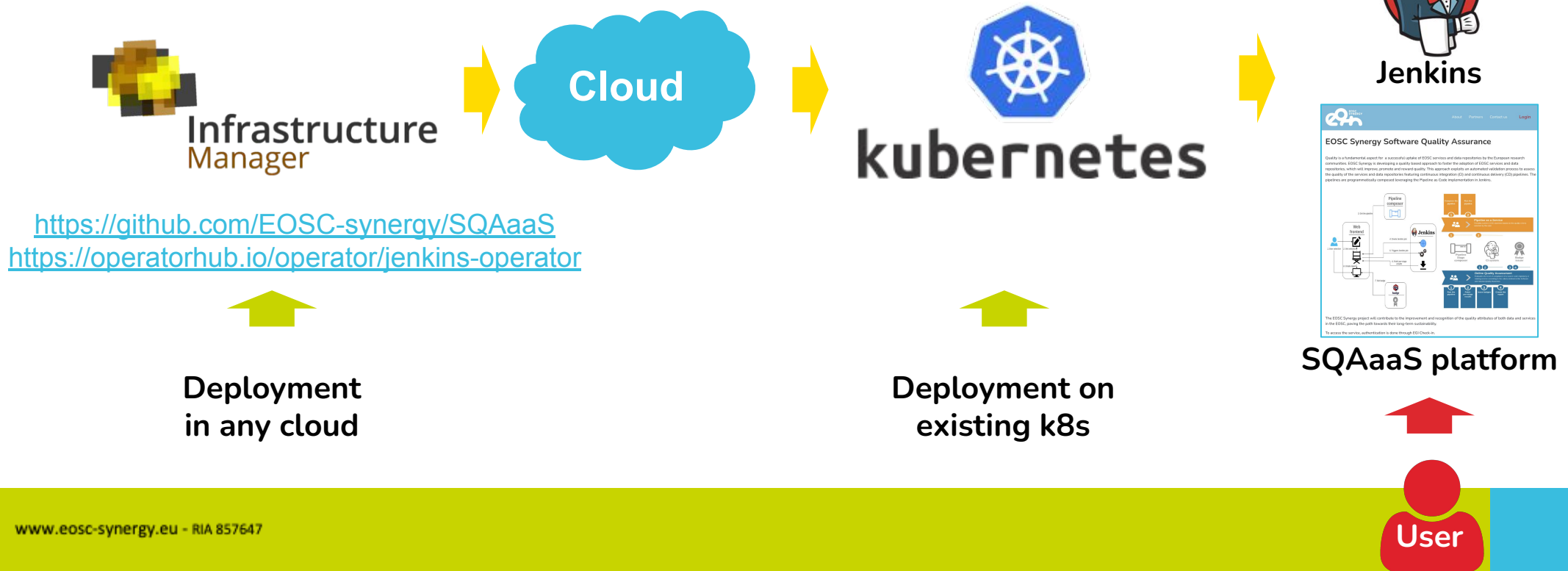
SQA baseline dynamic stages	Environment Setup	qc_style o3api	qc_coverage o3api	qc_functional o3api	qc_security o3api	qc_doc o3api	Push Images to Docker Registry	Docker Compose cleanup
14s	5s	1min 43s	23s	1min 50s	10s	1min 14s	7s	5s
14s	5s	1min 43s	23s	1min 50s	10s	1min 14s	7s	5s

Current release: 2.1.0

<https://indigo-dc.github.io/jenkins-pipeline-library/>

# SQAaaS: automated deployment under development

- Automated deployment of the complete SQAaaS platform
  - Facilitates SQAaaS production deployment, testing & promotes adoption
  - Also important for closed / private environments



# Thematic Services



## SCIPION



CryoEM data processing for Structural Biology



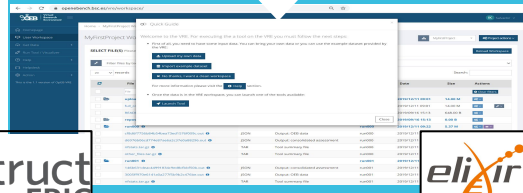
EIRENE



## OpenEBench



ELIXIR benchmarking and technical monitoring platform



## LAGO



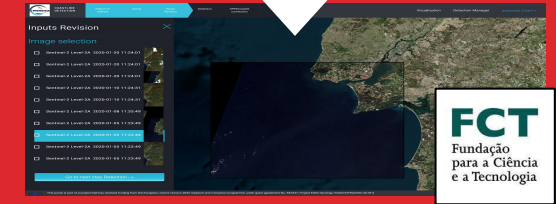
Latin American Giant cosmic ray Observatory



## WORSICA



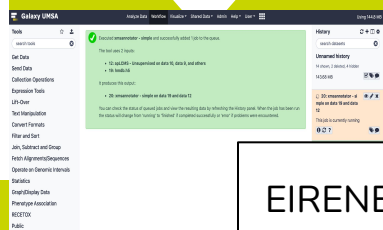
Water Monitoring Sentinel Cloud Platform



## UMSA



Untargeted Mass-Spectrometry Analysis

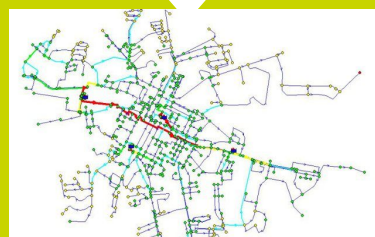


EIRENE

## MSWSS



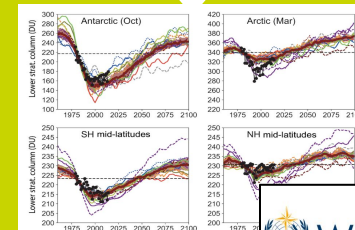
Water Supply Systems modeling and analysis



## O3AS



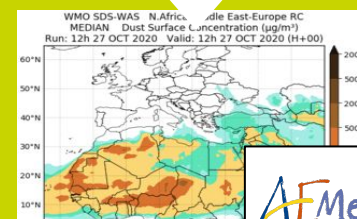
Ozone Analysis Service



## SDS-WAS



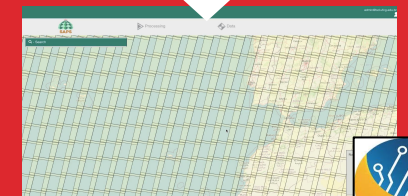
A Service related to mineral dust forecast



## SAPS



Surface Energy Balance Automated Processing Service

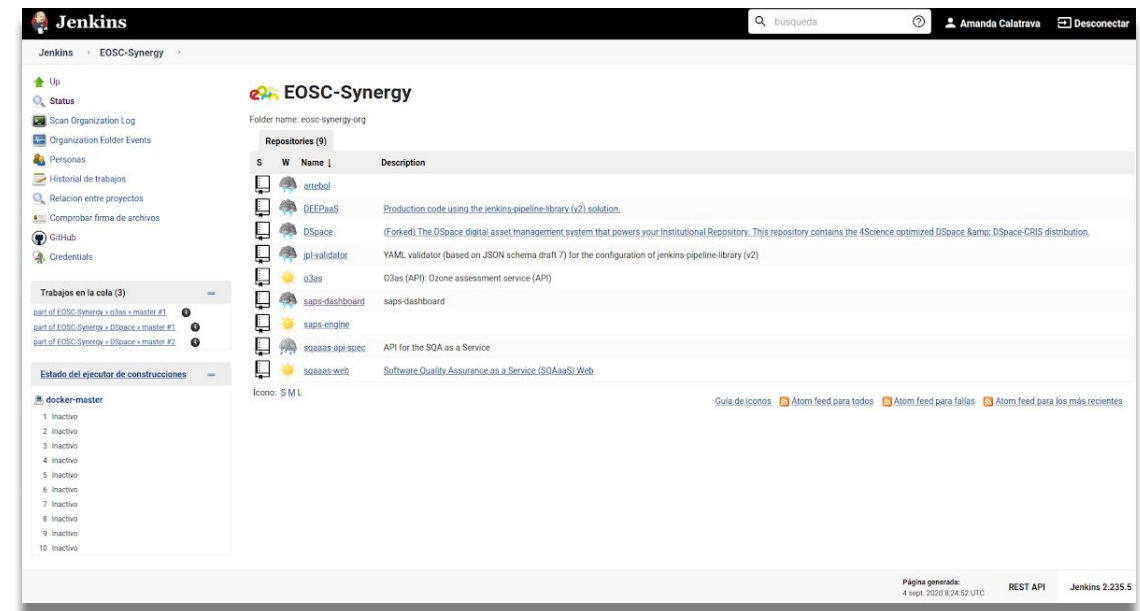




# SQAaaS: Jenkins instance for SQAaaS



- EOSC-Synergy Jenkins instance
- Checks automatically the projects in EOSC Synergy Github organization:
  - <https://github.com/EOSC-synergy>
- Jenkins Operator deployment to create your own Jenkins instances available:
  - A user deployable Jenkins with required plugins is almost ready for those that need on-premises solution

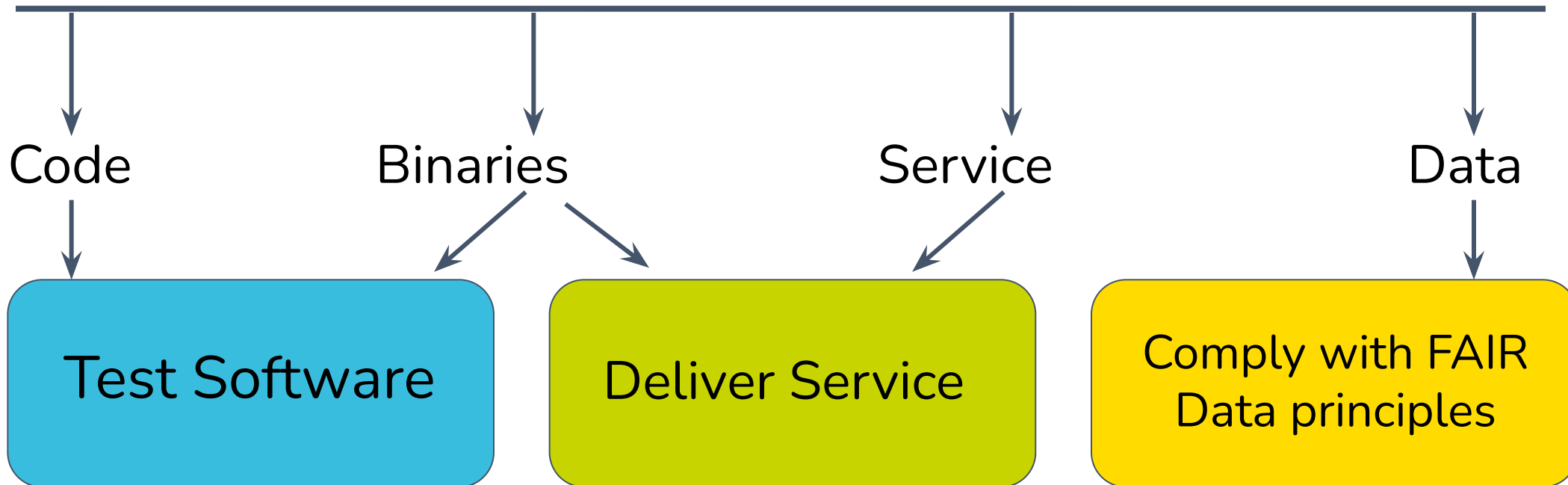


<https://jenkins.eosc-synergy.eu/job/eosc-synergy-org/>

# Generic Case in Earth Observation: Monitoring “XYZ” using Copernicus Data



## Monitoring XYZ using Copernicus data from Sentinel-2



# SQAaaS: Assessment of FAIR criteria



Testing Software  
and Services

- **Quality Assurance** baselines defined
- Automation tool integrated
- Deployment “*as a Service*”

Deliver Service

- **Build docker images** to automate the deployment
- **Deploy** virtual infrastructures automatically

Comply with FAIR  
Data principles

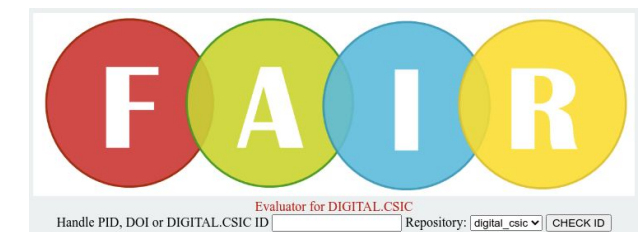
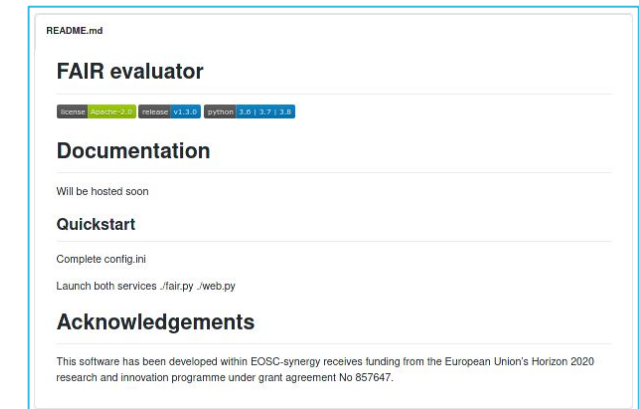
- **FAIR** principles **recommendations**.
- **Framework** to support FAIR best practices:  
implementation, validation, monitoring



# Technical Framework: FAIR-Evaluator

- Open Source tool for evaluating FAIR digital objects
- Oriented to:
  - Researchers and repository administrators.
  - To get feedback on FAIR compliance level of research data.
  - For institutional/multidisciplinary repositories.
- Provides FAIR assessment based on RDA indicators

## Stage View

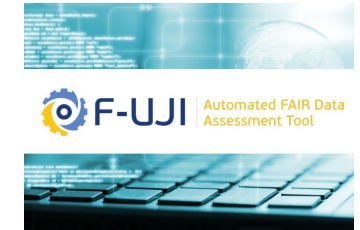


Leveraging pipelines for SQAaaS

[https://github.com/EOSC-synergy/FAIR\\_evaluator](https://github.com/EOSC-synergy/FAIR_evaluator)

# Technical Framework: F-UJI metrics in SQAaaS

- F-UJI is a FAIR assessment REST web service developed in FAIRsFAIR
- Enables validation of FAIRsFAIR Data Object Assessment Metrics
- Comparing alignment of F-UJI and FAIR validator metrics
- Looking at common interfaces for metrics
- Integrating F-UJI in the SQAaaS platform



Declarative: Checkout SCM	SQA baseline dynamic stages	Environment Setup	qc_doc fuji_app_repo	qc_style fuji_app_repo	Docker Compose cleanup
2s	12s	2s	16s	24s	16s
2s	12s	2s	16s	24s	16s

```
{
  "metric_specification": "https://doi.org/10.5281/zenodo.4081213",
  "metric_version": "metrics_v0.4.yaml",
  "request": {
    "callback_endpoint": "https://digital.eosc.eu/dspace-oai/request",
    "object_identifier": "https://hdl.handle.net/10261/153475",
    "test_debug": true,
    "use_datacite": null
  },
  "results": [
    {
      "id": 1,
      "metric_identifier": "Faf-FI-01D",
      "metric_name": "Data is assigned a globally unique identifier.",
      "metric_test": {
        "faf-fi-01d-1": {
          "metric_test_name": "Identifier follows an idutils defined unique identifier syntax",
          "metric_test_score": 1,
          "metric_test_status": "pass"
        }
      },
      "output": {
        "guid": "https://hdl.handle.net/10261/153475",
        "guid_scheme": "handle"
      },
      "score": {
        "earned": 1,
        "total": 1
      },
      "test_debug": {
        "INFO: Using idutils schemes",
        "SUCCESS: Unique identifier schemes found ['handle', 'url']",
        "INFO: Finalized unique identifier scheme - handle"
      },
      "test_status": "pass"
    },
    {
      "id": 2,
      "metric_identifier": "Faf-FI-02D",
      "metric_name": "Data is assigned a persistent identifier.",

```