AI Center of Excellence
Office of the CTO

Feedback about deployment of an intelligent app

DevConf US 2021

Francesco Murdaca
Senior Data Scientist

Red Hat

Agenda

- Project Thoth
- MLOps relation
- Why, What, When
- How to use it
- Pipeline and results

Project Thoth

Project Thoth

- Help developers in the selection of dependencies for their applications depending on their requirements
- Deliver optimized images for your applications
- Use bots to automate mundane work to offload humans work

How Thoth can help developers?

- Keep dependencies up to date.
- Maintain software stack secure avoiding CVE.
- Recommend the most performant software stacks for AI Apps.
- Integrate source metadata information related to the packages used in the software stack to give advice to users.
- Integrate Thoth in day-to-day developers/data scientists tools.
- ...

Thoth Integrations

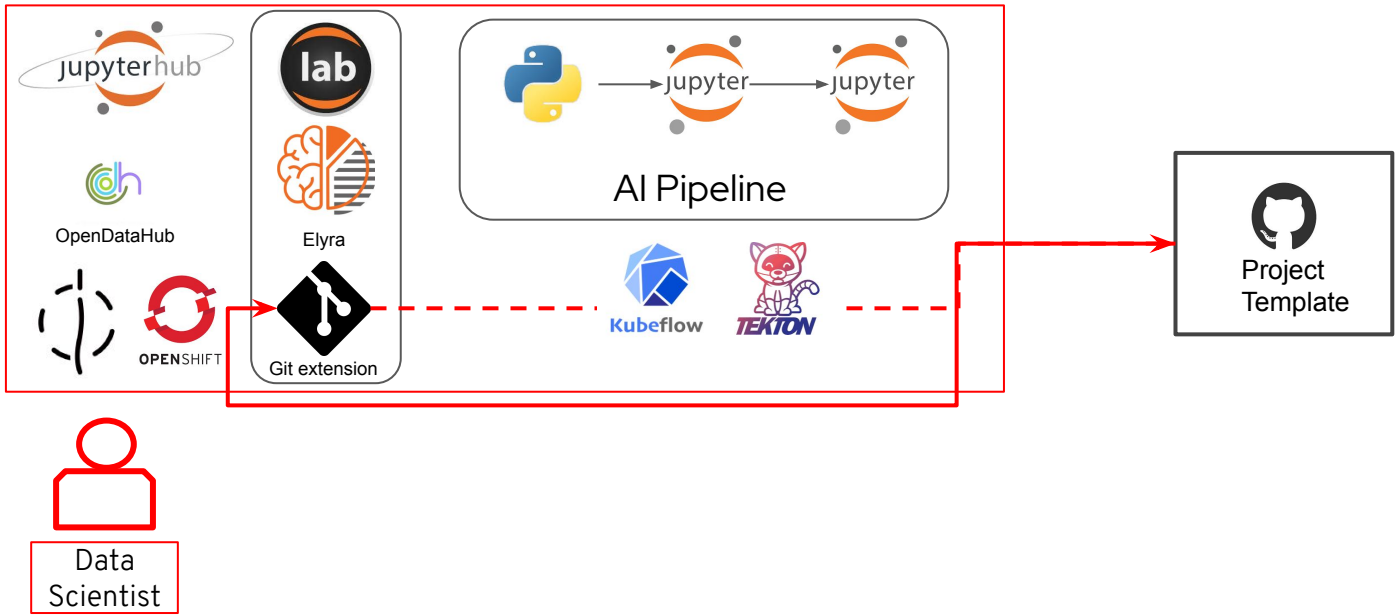
- Command line tool thamos (developer laptop)
- Jupyter Tools (data scientist browser)
- Cyborg Kebechet (pull request/issues creator)
- Source-to-Image (container builder)
- Optimizing Deployment Pipeline

```
host: {THOTH_SERVICE_HOST}
tls_verify: true
requirements_format: {requirements_format}

runtime_environments:
  - name: '{os_name}:{os_version}'
    operating_system:
      name: {os_name}
      version: '{os_version}'
    hardware:
      cpu_family: {cpu_family}
      cpu_model: {cpu_model}
    python_version: '{python_version}'
    cuda_version: {cuda_version}
    recommendation_type: stable
    platform: '{platform}'
```

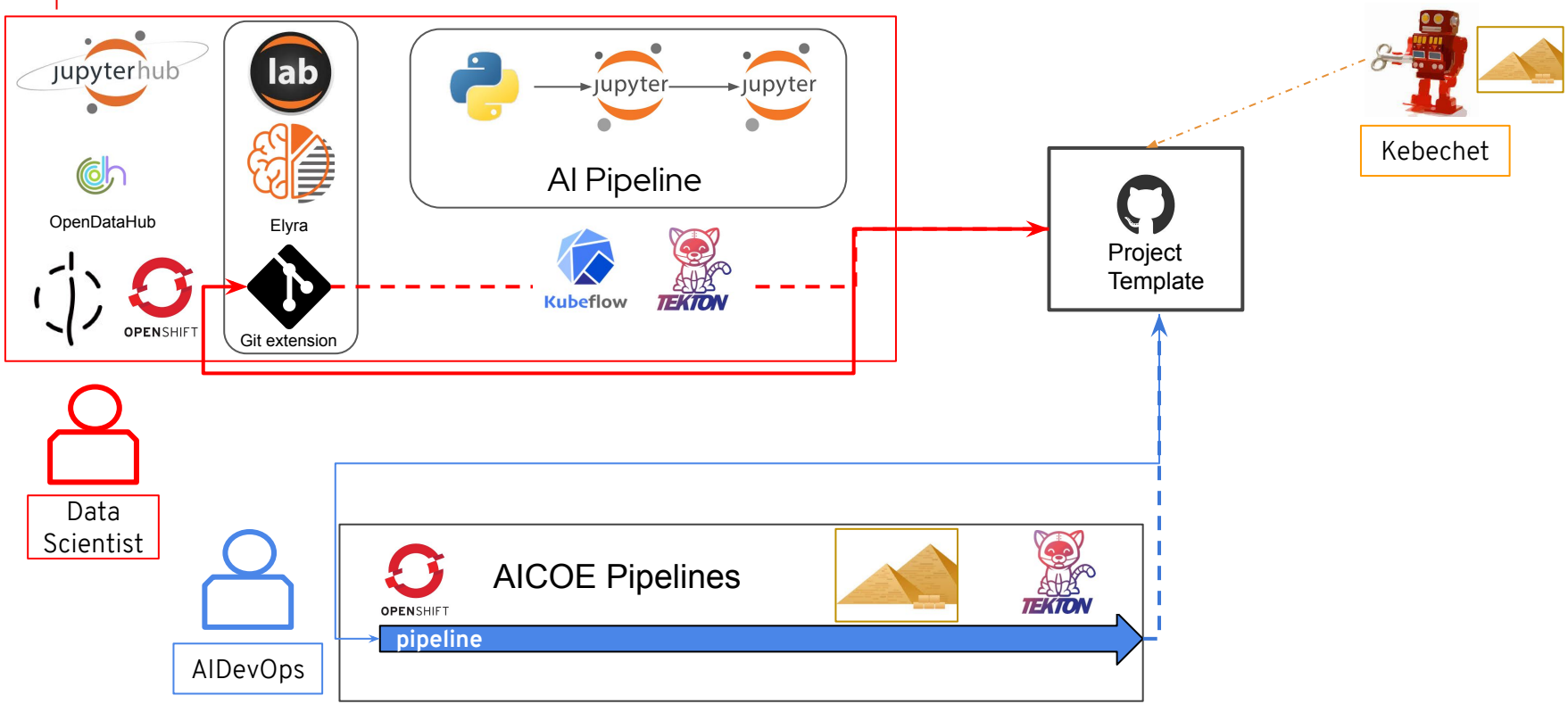
MLOps relation

Providing feedback about deployment of an intelligent app

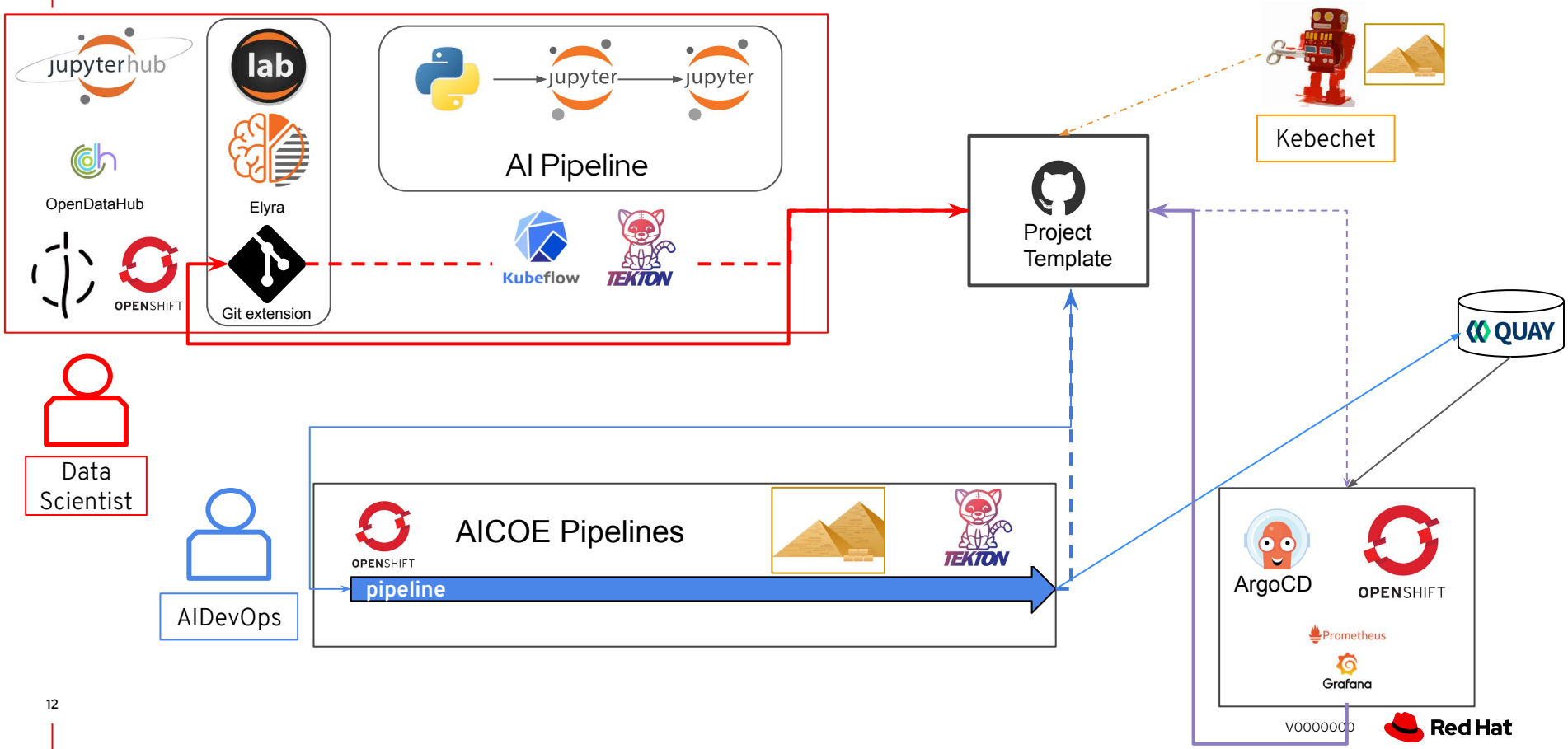
Providing feedback about deployment of an intelligent app



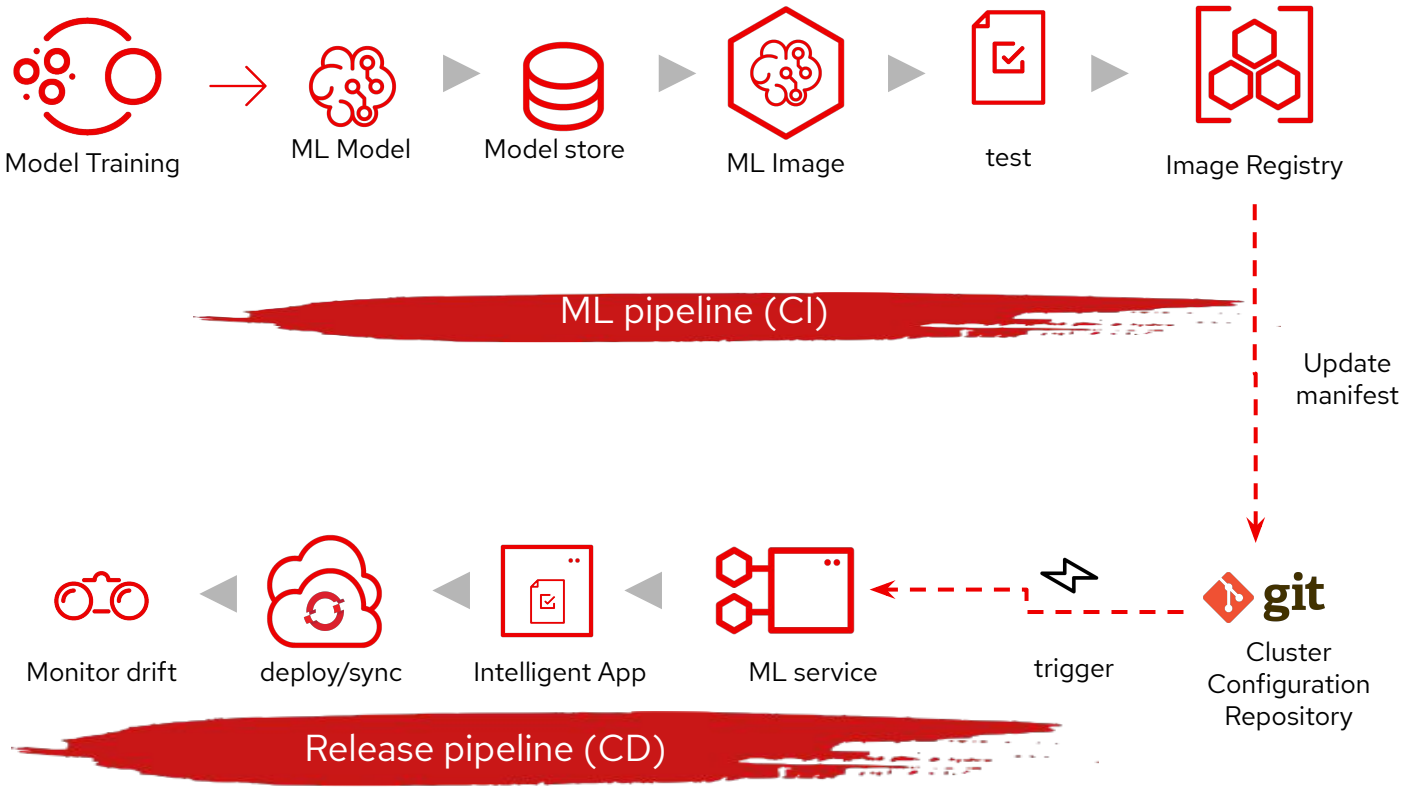
Providing feedback about deployment of an intelligent app



Providing feedback about deployment of an intelligent app

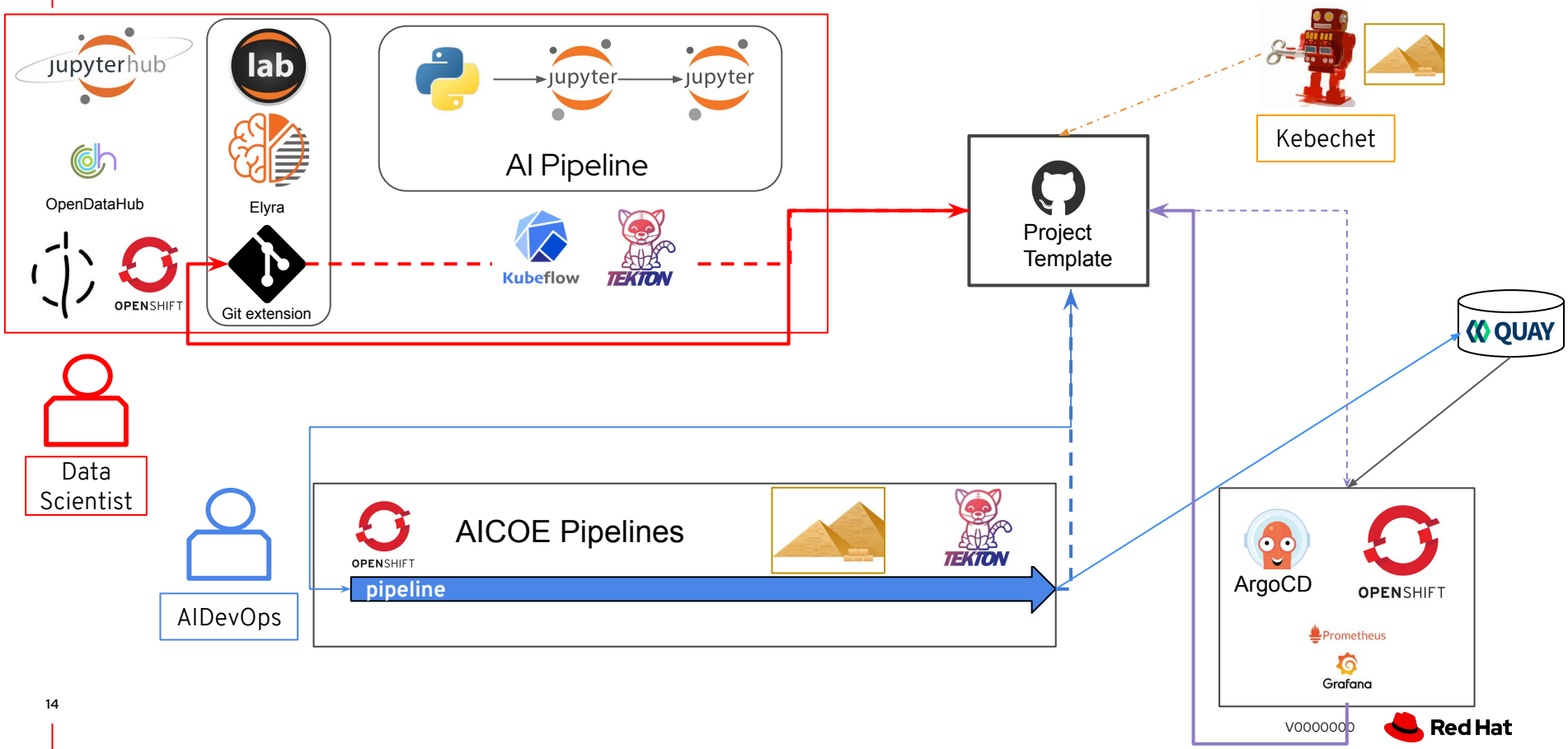


Providing feedback about deployment of an intelligent app

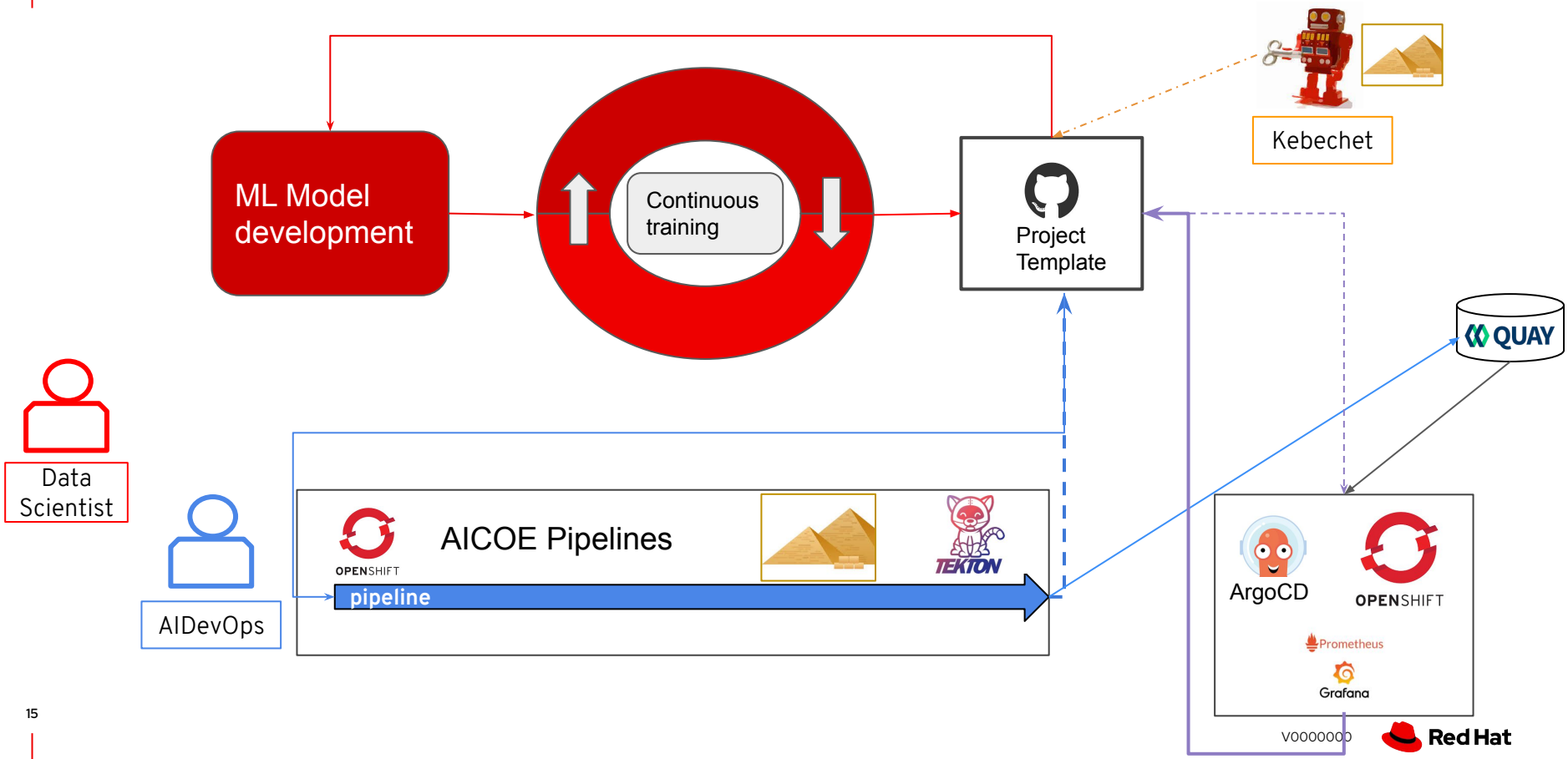


Red Hat

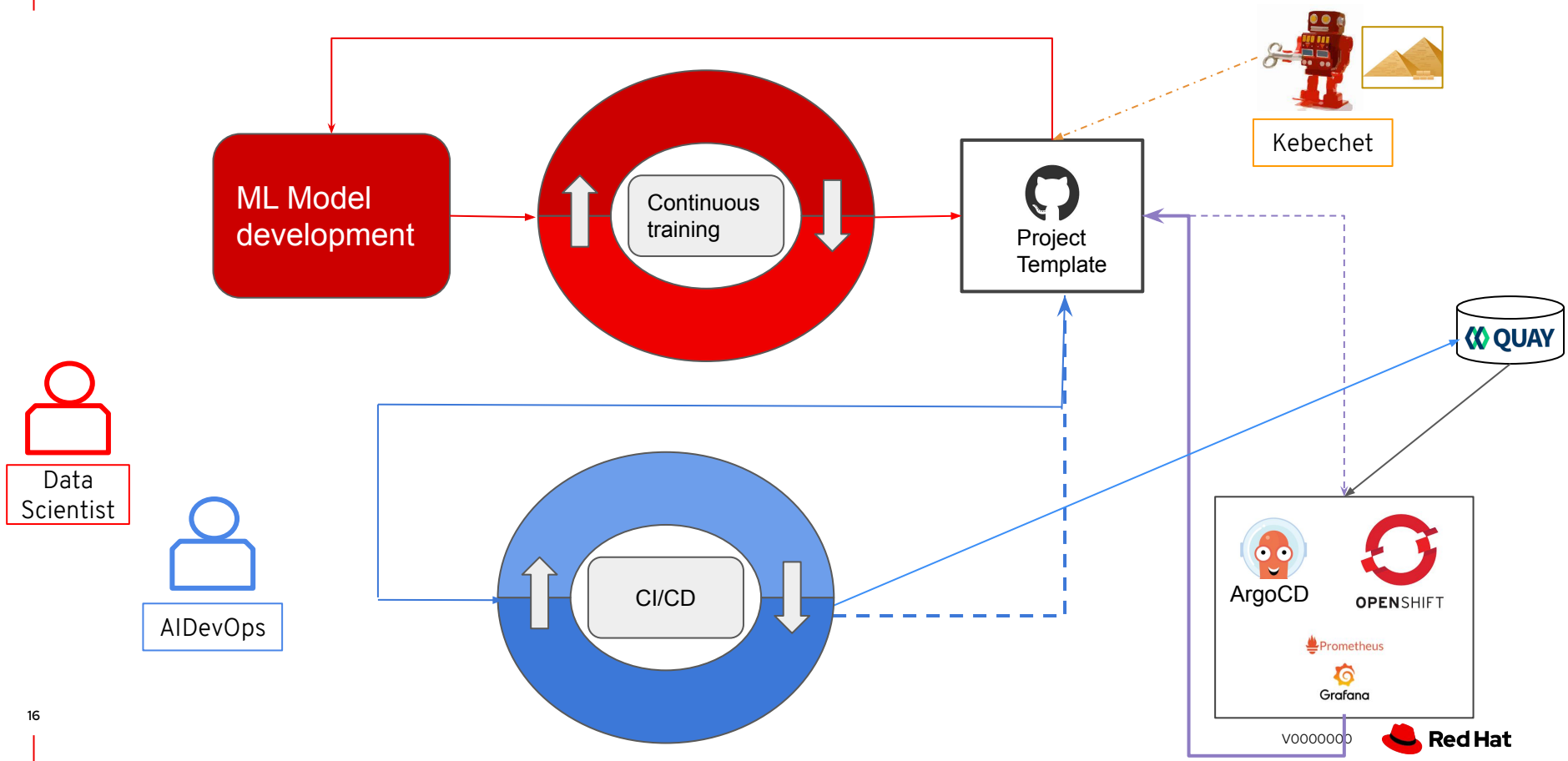
Providing feedback about deployment of an intelligent app



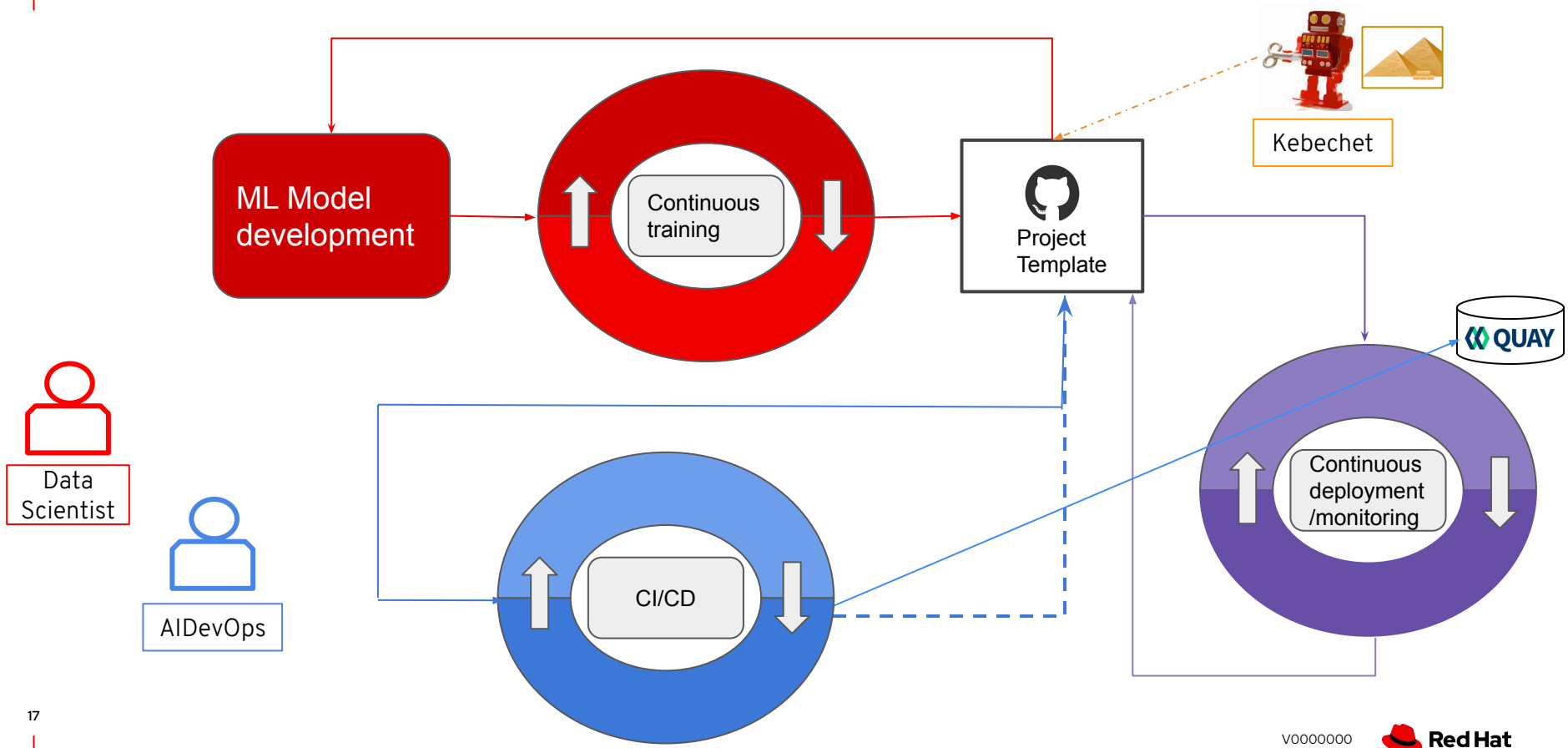
Providing feedback about deployment of an intelligent app



Providing feedback about deployment of an intelligent app

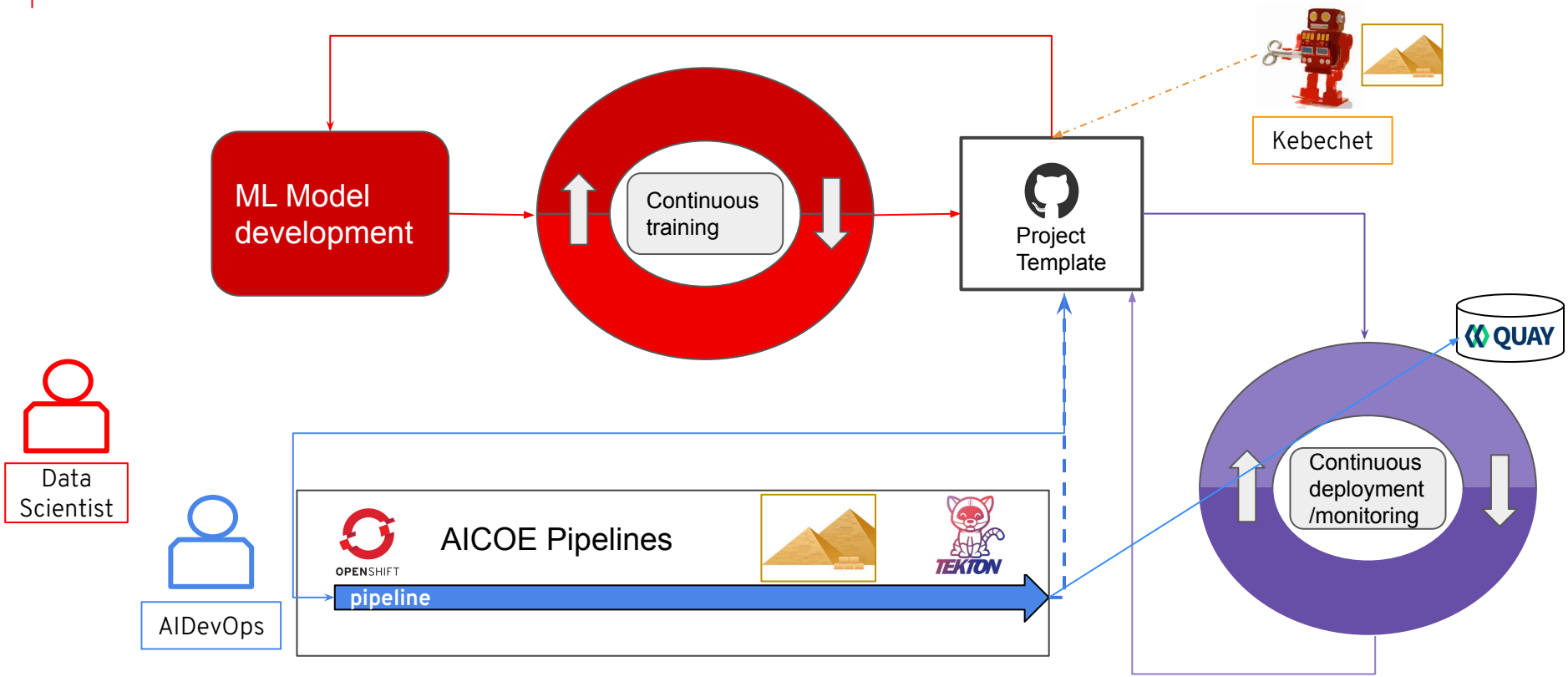


Providing feedback about deployment of an intelligent app

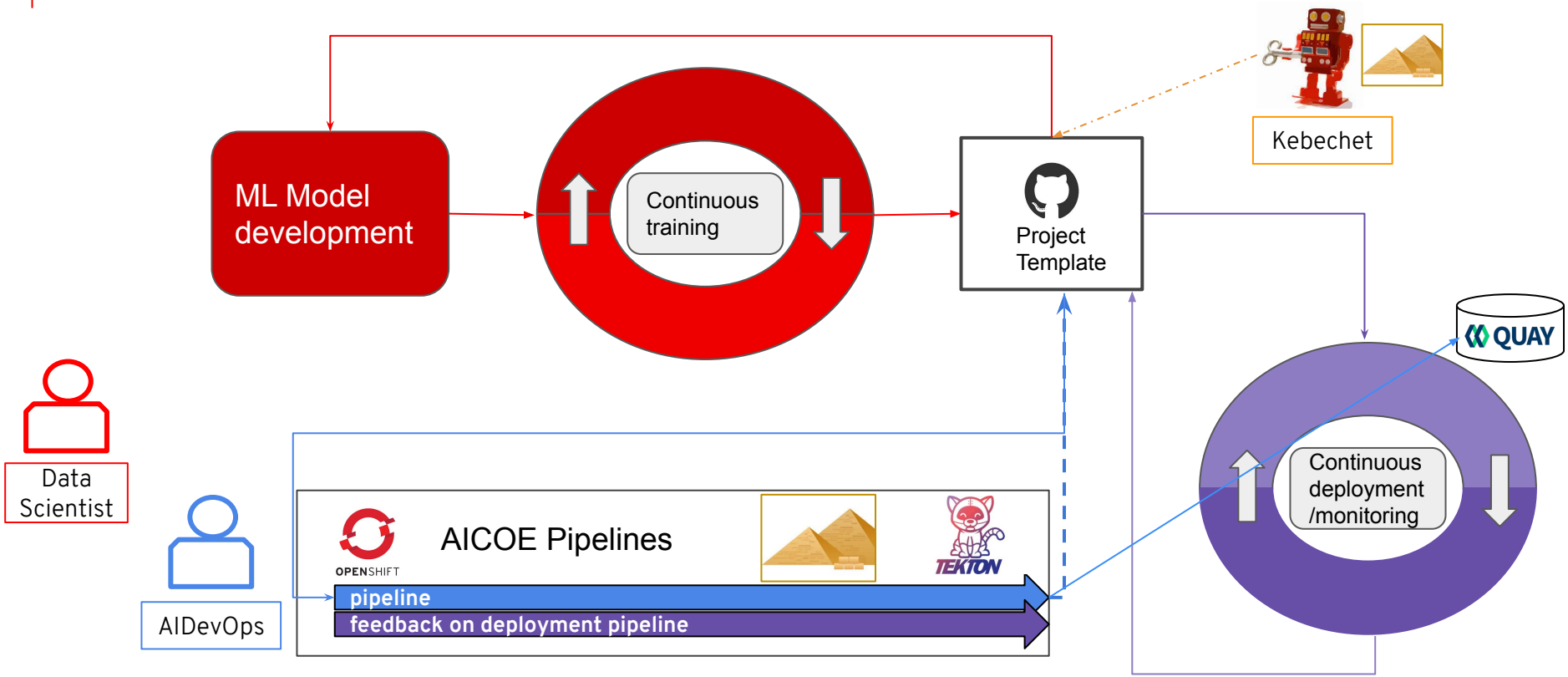


Red Hat

Providing feedback about deployment of an intelligent app



Providing feedback about deployment of an intelligent app



Red Hat

Why, What, When

Why feedback?

- App lifecycle is not static
- Speed up process
- Focus on modeling and data
- Reduce cost

What feedback?

- Model metrics
- Application metrics
- Platform Metrics

Personas that benefit from the feedback?

- AIDevOps wants to know operational information (latency, memory consumption, CPU usage)
- Data Scientist wants to verify model created is still performing well in production environment

When giving feedback?

- When a new software stack is created
- When a new model version is created
- When a new dataset version is created

How to use it?

What do you need?

- AICoE-CI
- Khebut Bot
- Model to be deployed
- Test to collect metrics

AICoE CI

GitHub App

aicoe-ci

This is the Continuous Integration Cyborg maintained by Thoth Station.

Configure

Manage your installation settings.

Developer

AICoE

Website

aicoe-ci is provided by a third-party and is governed by separate terms of service, privacy policy, and support documentation.

Report abuse

Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/build-images.md

AICoE CI

```
33
34    - name: inference
35      build:
36        base-image: "quay.io/thoth-station/s2i-thoth-ubi8-py38:v0.28.0"
37        build-stratergy: Source
38        registry: quay.io
39        registry-org: thoth-station
40        registry-project: elyra-aidevsecops-tutorial
41        registry-secret: thoth-station-thoth-pusher-secret
42      deploy:
43        project-org: "thoth-station"
44        project-name: "elyra-aidevsecops-tutorial"
45        image-name: "elyra-aidevsecops-tutorial"
46        overlay-contextpath: "manifests/overlays/test/imagestreamtag.yaml"
```

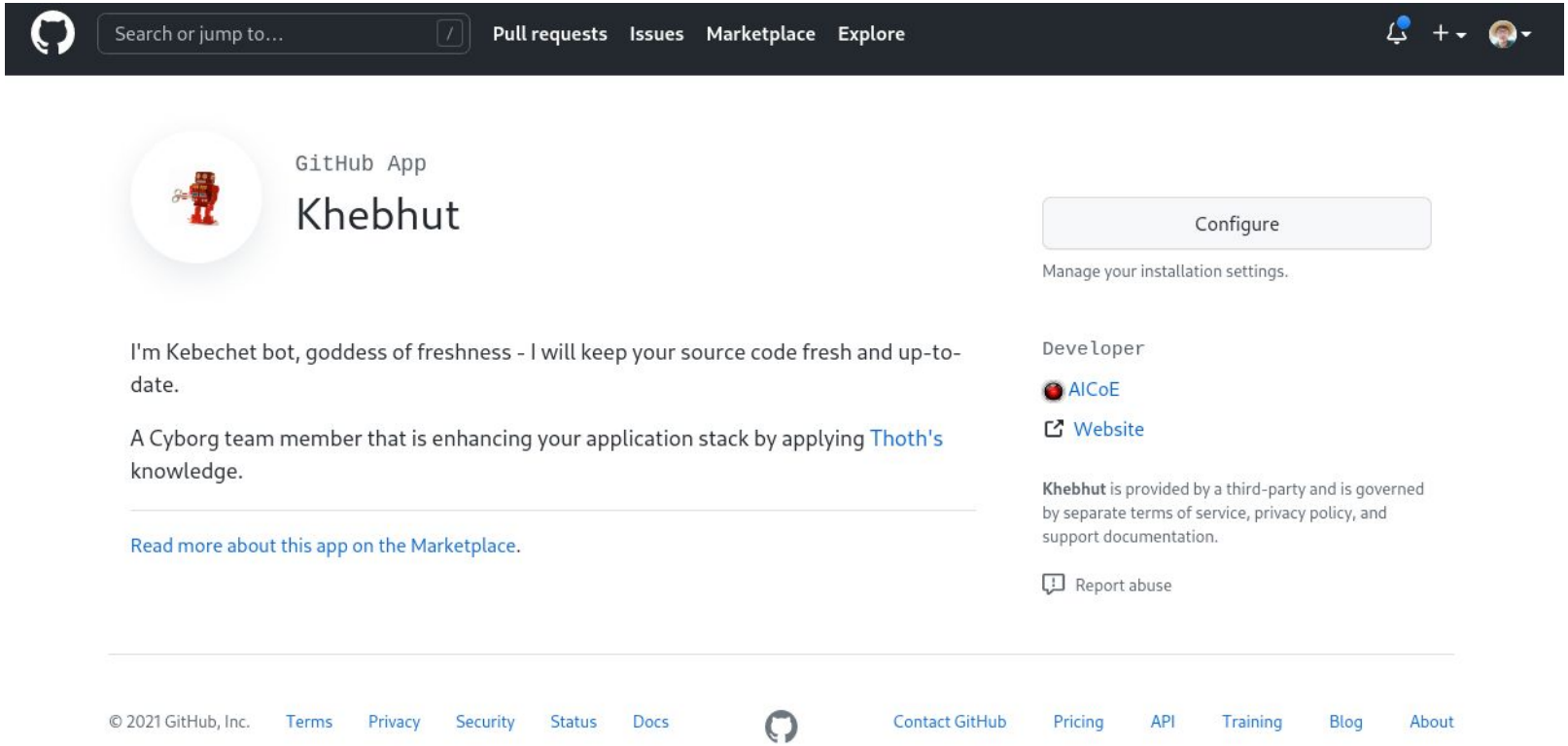

Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/build-images.md

Deployments

- **Flask Application**
- KFServing
- Seldon
- TensoRT

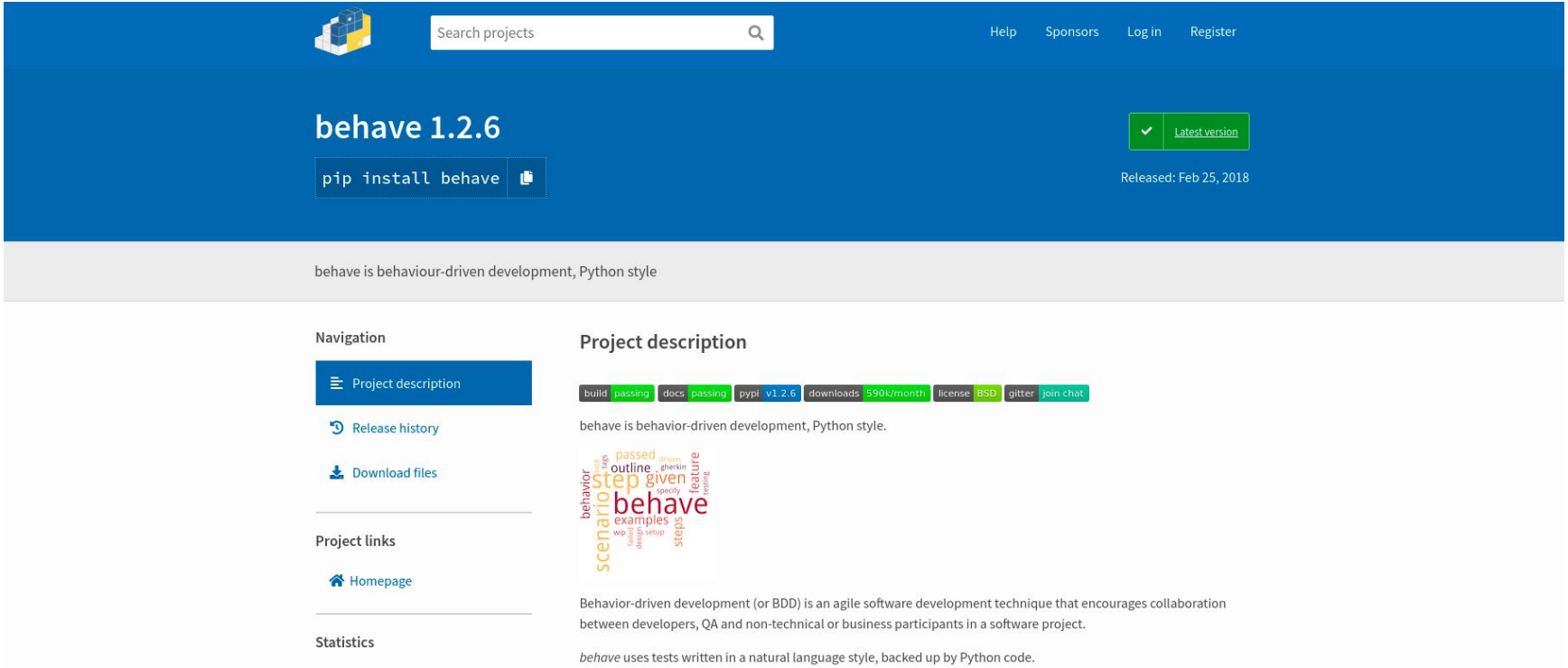
Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/deploy-model.md

Providing feedback about deployment of an intelligent app



Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/build-images.md

behave



behave uses tests written in a natural language style, backed up by Python code.

Providing feedback about deployment of an intelligent app

6 lines (6 sloc) | 274 Bytes

```
Feature: Gather metrics deployed model
    Scenario: Deployment metrics gathering
        Given dataset is available
        Given deployment is accessible
        When I run test to gather metrics on predict endpoint
        Then I should get model and application metrics
```


Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/test-model.md

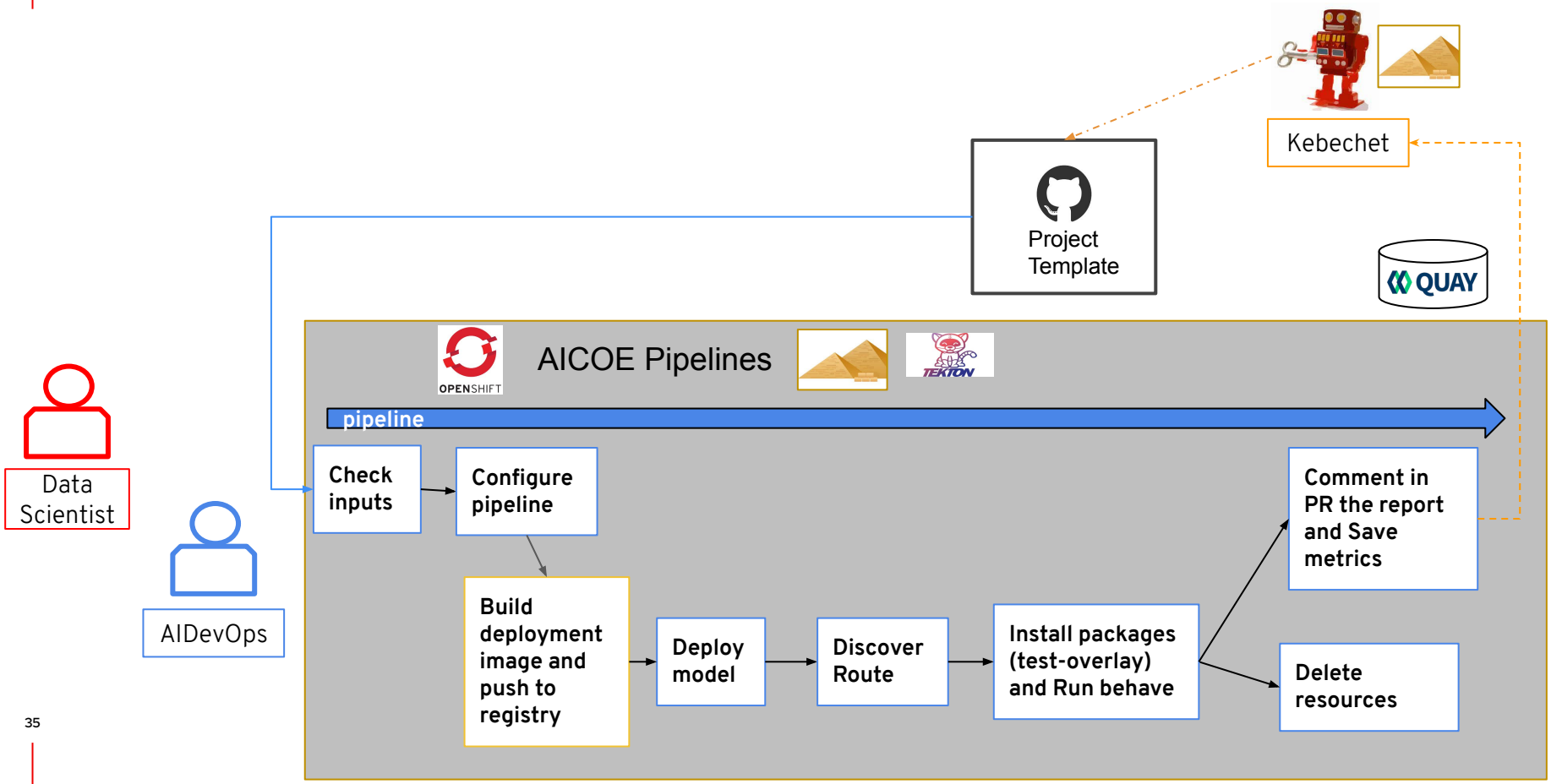
Providing feedback about deployment of an intelligent app

```python
@given("dataset is available")
def dataset_availability(context):
    """Check availability of dataset and retrieves it."""
    # Prepare MNIST data.
    _, (x_test, y_test) = tf_dataset.load_data()

    context.dataset = {
        "x_test": x_test,
        "y_test": y_test,
    }

    assert context.dataset


@given("deployment is accessible")
def deployment_accessible(context):
    """Check the deployment is accessible."""
    context.result = {}

    context.model_api_url = os.environ["DEPLOYED_MODEL_URL"]

    response = requests.get(f"{context.model_api_url}")

    assert (
        response.status_code == 200
    ), f"Invalid response when accessing {context.model_api_url}: {response.status_code!r}: {response.text}"

    assert response.text, f"Empty response from server for {context.model_api_url}"
```


Source: https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/master/docs/source/test-model.md

Pipeline and results

Providing feedback about deployment of an intelligent app



Providing feedback about deployment of an intelligent app

sesheta commented 1 hour ago — Member

AICoE CI results

Test inputs

The following table shows info about test used to gather metrics.

test URL	namespace deployment
https://github.com/AICoE/elyra-aidevsecops-tutorial/blob/e882fa5d4534aa0ec1909911665d1b05dab35736/features	aicoe-ci

Model and application metrics

The following table shows gathered metrics for model and application on your deployed models.

average_latency	average_error	average_probability	number_requests	model_version
0.118086	0	0.999916	21	pr-160

Platform metrics

The following table shows gathered metrics from platform on your deployed models.

CPU max usage	Memory max usage
1	384Mi

Providing feedback about deployment of an intelligent app

- **Thoth station**
  - https://thoth-station.ninja/
- **AICoE CI**
  - https://github.com/aicoe/aicoe-ci
- **Khebut**
  - https://github.com/apps/khebhut
- **Thoth YouTube**
  - https://www.youtube.com/channel/UCIUIDuq_hQ6vlzmqM59B2Lw
- **Operate First**
  - https://www.operate-first.cloud/

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make
Red Hat a trusted adviser to the Fortune 500.


linkedin.com/company/red-hat


youtube.com/user/RedHatVideos


facebook.com/redhatinc


twitter.com/RedHat

Red Hat