



Red Hat
Summit

Connect

Red Hat
Summit

ChatOCP - Intelligent application development on OpenShift

Juozas Auskalnis
Solution Architect

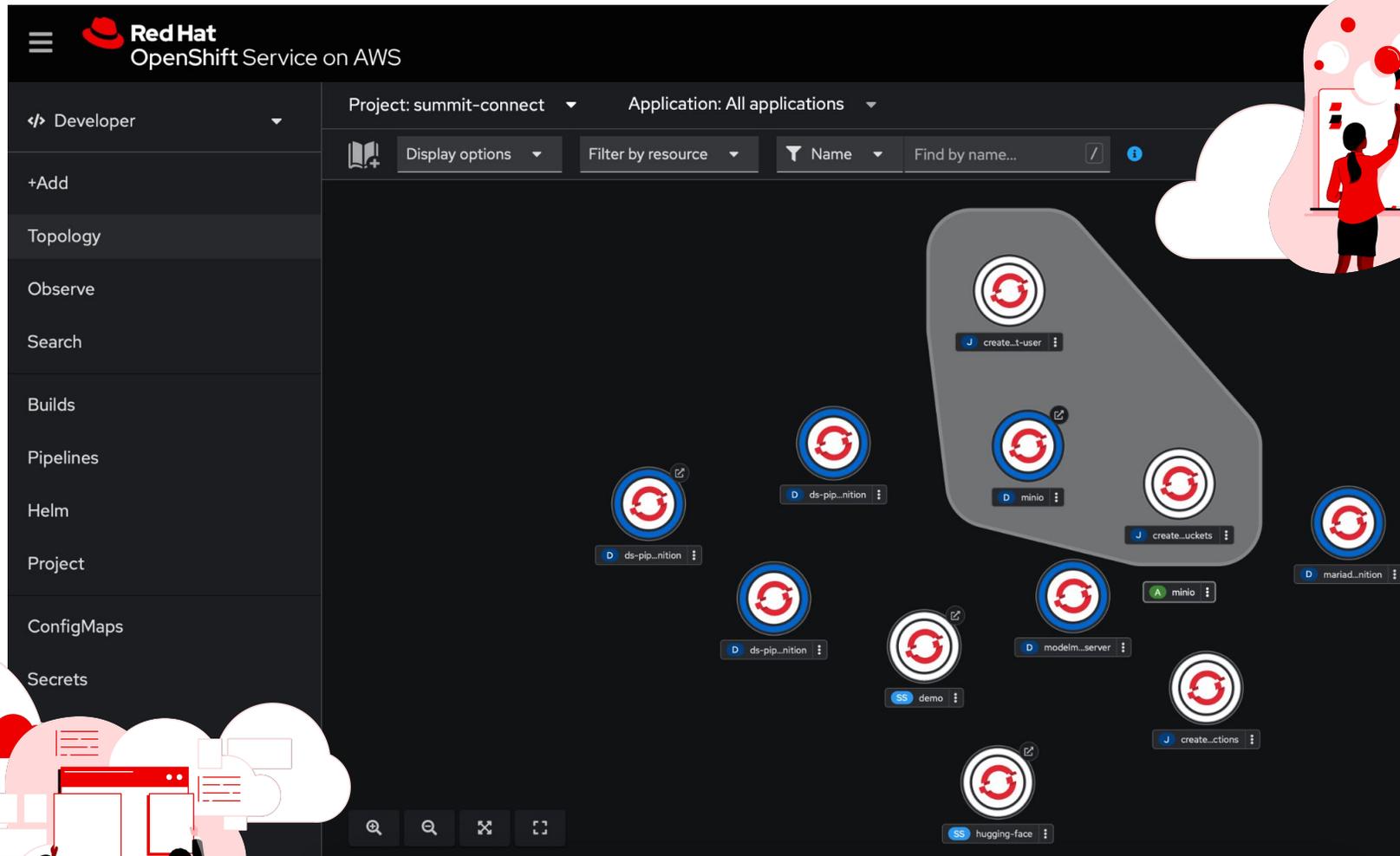


- Platform
- Development tools
- Storage
- OpenShift Data Science
- Security



- Disconnected operations across teams that create deployment delays.
- Increasing complexity of integrating AI solutions with in-place infrastructure.
- The need to continuously (re)train models, auditing requirements and monitoring of model performance.
- As generative AI has emerged, organizations are now facing new challenges related to pre-built models, including:
 - Licensing and ownership.
 - Regulatory and locality restrictions.

secure, supported & stable platform



secure, supported & stable platform

Start quickly, we manage it for you



Managed Red Hat OpenShift services



Red Hat OpenShift Service on AWS¹



Azure Red Hat OpenShift



Red Hat OpenShift on IBM Cloud¹



Red Hat OpenShift Dedicated²

You manage it, for control and flexibility



Self-managed Red Hat OpenShift

 **Red Hat OpenShift Platform Plus**

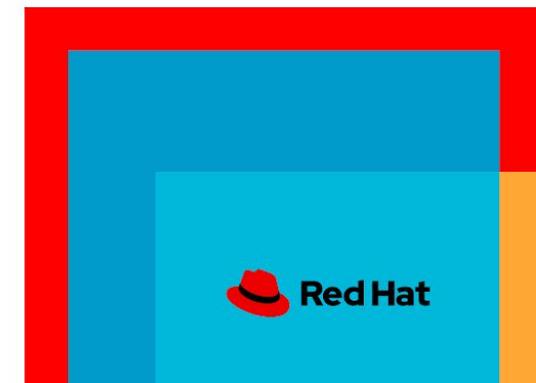
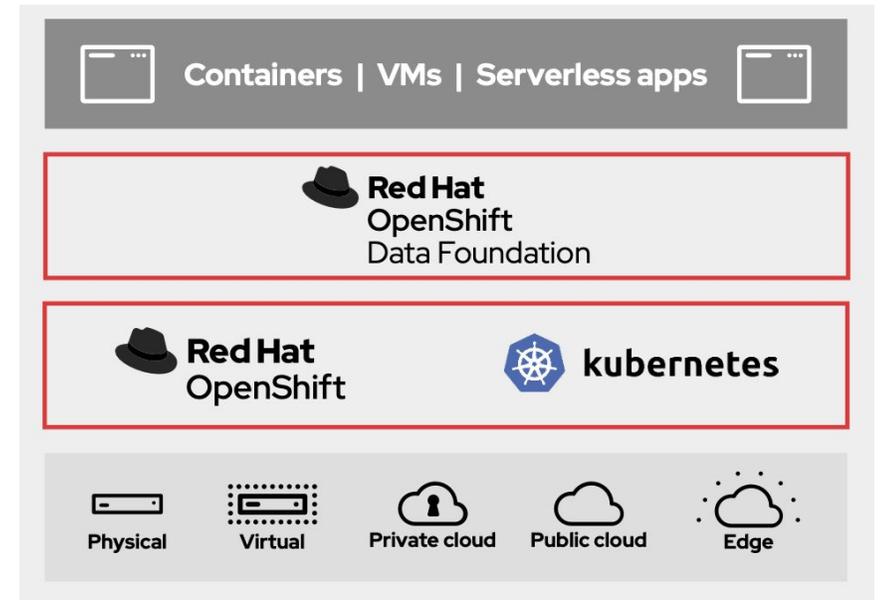
 **Red Hat OpenShift Container Platform**

 **Red Hat OpenShift Kubernetes Engine**

On public cloud, or on-premises on physical or virtual infrastructure³

object storage everywhere

- OpenShift Data Foundation provides the functionality that organizations need to support applications and workloads in one complete solution—fully integrated and delivered with Red Hat OpenShift.
- Multicloud object gateway exposes a consistent Amazon S3 compatible endpoint to developers and applications.
- OpenShift Data Foundation clusters support end-to-end encryption including encryption with user-provided keys.



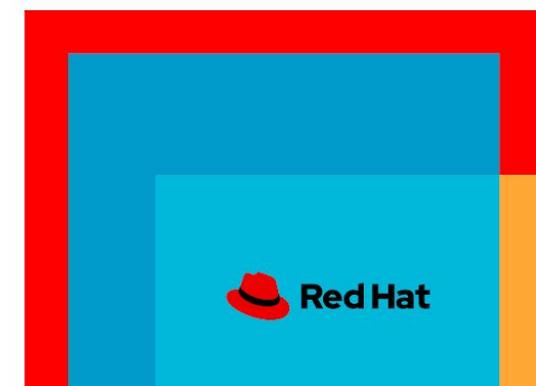
object storage everywhere

The screenshot displays the Red Hat OpenShift Service on AWS console. The left sidebar contains navigation options: Storage (expanded), Builds, Pipelines, Observe, Compute, and User Management. The main content area is divided into several sections:

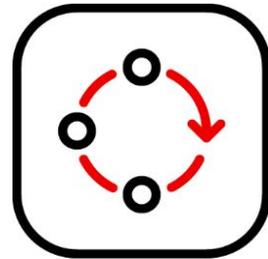
- Storage:** A list of storage-related items including Data Foundation, PersistentVolumes, PersistentVolumeClaims, StorageClasses, VolumeSnapshots, VolumeSnapshotClasses, VolumeSnapshotContents, Object Buckets, and Object Bucket Claims.
- Details:** Information about the service, including Service name (Data Foundation), System name (Multicloud Object Gateway), Provider (AWS), and Version (4.13.4-rhodf).
- Status:** Shows the status of Object Service and Data Resiliency, both marked as successful with green checkmarks.
- Capacity breakdown:** A section with a 'Projects' button and a message: 'Not enough usage data'.
- Storage efficiency:** Shows 'Compression ratio' and 'Savings', both as 'Not available'.
- Buckets:** Lists '1 NooBaa Bucket', '0 Object', '1 Object Bucket', and '1 Object Bucket Claim'.
- Resource providers:** Lists '1 AWS'.
- Performance:** A section with an 'I/O Operat' button and a bar chart showing performance metrics for 'AWS'.

This modal window shows the details for the 'OpenShift Data Foundation' operator. It includes the following information:

- Operator details:** Installed Operators > Operator details
- Operator icon:** Red Hat logo
- Operator name:** OpenShift Data Foundation
- Version:** 4.13.4-rhodf provided by Red Hat
- Navigation tabs:** Details (selected), YAML, Subscription, Events, Storage System
- Provided APIs:** A section titled 'Storage System' with a description: 'StorageSystem is the Schema for the storagesystems API' and a 'Create instance' button.



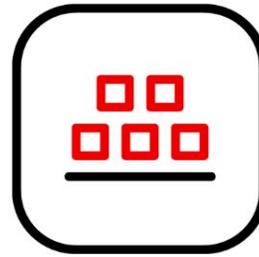
developer tools comes with OpenShift



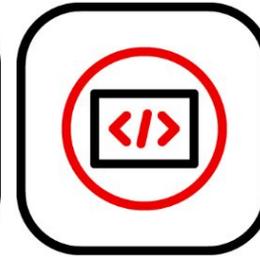
Red Hat OpenShift Pipelines



Red Hat Runtimes



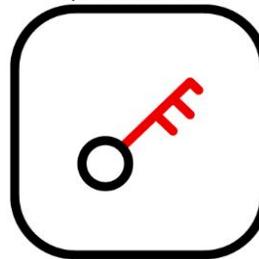
Red Hat OpenShift Serverless



Red Hat OpenShift Dev Spaces



Red Hat OpenShift GitOps



Single sign-on

dev spaces

```
1 package org.acme.getting.started.knative;
2
3 import javax.inject.Inject;
4 import javax.ws.rs.GET;
5 import javax.ws.rs.Path;
6 import javax.ws.rs.Produces;
7 import javax.ws.rs.core.MediaType;
8
9 @Path("/")
10 public class GreetingResource {
11
12     @Inject
13     GreetingService service;
14
15     @GET
16     @Produces(MediaType.TEXT_PLAIN)
17     @Path("/greeting/{name}")
18     public String greeting(String name) {
19         return service.greeting(name);
20     }
21
22     @GET
23     @Produces(MediaType.TEXT_PLAIN)
24     public String hello() {
25         return "hello";
26     }
27 }
```

Red Hat
OpenShift
Dev Spaces

Create Workspace

Workspaces (1)

RECENT WORKSPACES

- quarkus-quickstart

Select a Sample

Select a sample to create your first workspace.

java 5 items Temporary Storage Off

- Java 11 with Lombok**
Java stack with OpenJDK 11, Maven 3.6 and Lombok 1.18
- Java 11 with Quarkus**
Java stack with OpenJDK 11, Maven 3.6 and Quarkus Tools
- Java Lombok**
Java Stack with Lombok 1.18.18, OpenJDK 11 and Maven 3.6.0
- Java Spring Boot**
Java stack with OpenJDK 11 and Spring Boot Petclinic demo application
- Java with Spring Boot and MongoDB**
Java stack with OpenJDK 8, MongoDB and Spring Boot Guestbook demo application

what is ODS?

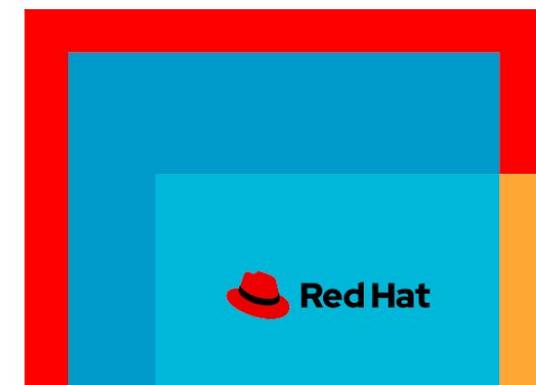
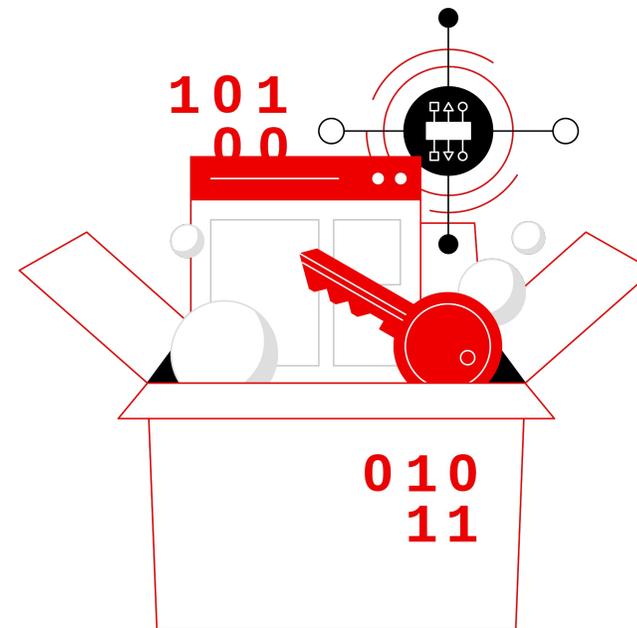
Red Hat OpenShift Data Science provides the ability to build, train, deploy and monitor models on premise, in the public cloud or at the edge.

Red Hat updates the platform and integrated AI tooling (like Jupyter Notebooks, PyTorch, and TensorFlow libraries).

Using the model serving and monitoring tools built into Red Hat OpenShift Data Science, models are container-ready, which makes it easier to integrate them into an intelligent app.

Models can be rebuilt and redeployed as part of a continuous integration/continuous development process based on changes to the source notebook.

Nvidia and Intel AI tools unlock high-performance training.



what is inside ODS?

The screenshot displays the Red Hat OpenShift Data Science console interface. On the left is a dark sidebar with navigation options: Applications (with a dropdown arrow), Enabled, Explore (highlighted), Data Science Projects, Data Science Pipelines (with a right arrow), Model Serving, Resources, and Settings (with a right arrow). The main content area is a grid of application cards, each with a logo, a management status badge, a title, a provider, and a brief description.

Application	Provider	Management Status	Description
Anaconda Professional	Anaconda	Partner managed	Anaconda Professional is a popular open source package distribution and management experience that is optimized for commercial use.
IBM Watson Studio	IBM	Self-managed	IBM Watson Studio is a platform for embedding AI and machine learning into your business and creating custom models with your own data.
Intel® oneAPI AI Analytics Toolkit Container	Intel®	Self-managed	The AI Kit is a set of AI software tools to accelerate end-to-end data science and analytics pipelines on Intel® architectures.
Jupyter	Jupyter	Red Hat managed	A multi-user version of the notebook designed for companies, classrooms and research labs.
NVIDIA GPU Add-on	NVIDIA	Self-managed	NVIDIA GPU Add-on prepares OpenShift to run GPU-accelerated workloads on a single node or on multiple nodes.
OpenShift API Management	Red Hat	Red Hat managed	OpenShift API Management is a service that accelerates time-to-value and reduces the cost of delivering API-first, microservices-based applications.
OpenVINO	Intel®	Self-managed	OpenVINO is an open source toolkit to help optimize deep learning performance and deploy using an inference engine onto Intel® hardware.
Pachyderm	Pachyderm	Self-managed	Pachyderm is the data foundation for machine learning. It provides industry-leading pipelines, data versioning, and lineage for data science teams to automate the machine learning lifecycle.
Starburst Galaxy	Starburst	Partner managed	Starburst Galaxy is a fully managed service to run high-performance queries across your various data sources using SQL.

how we use ODS?

Workbenches [Create workbench](#)

Name	Notebook image	Container size	Status	
> Demo	TensorFlow	Small	Stopped	Open
> Hugging Face	PyTorch	Medium	Running	Open

Cluster storage [Add cluster storage](#)

Name	Type	Connected workbenches
> Demo	Persistent storage	Demo
> Hugging Face	Persistent storage	Hugging Face

Data connections [Add data connection](#)

Name	Type	Connected workbenches	Provider
My Storage	Object storage	Hugging Face	AWS S3
Pipeline Artifacts	Object storage	No connections	AWS S3

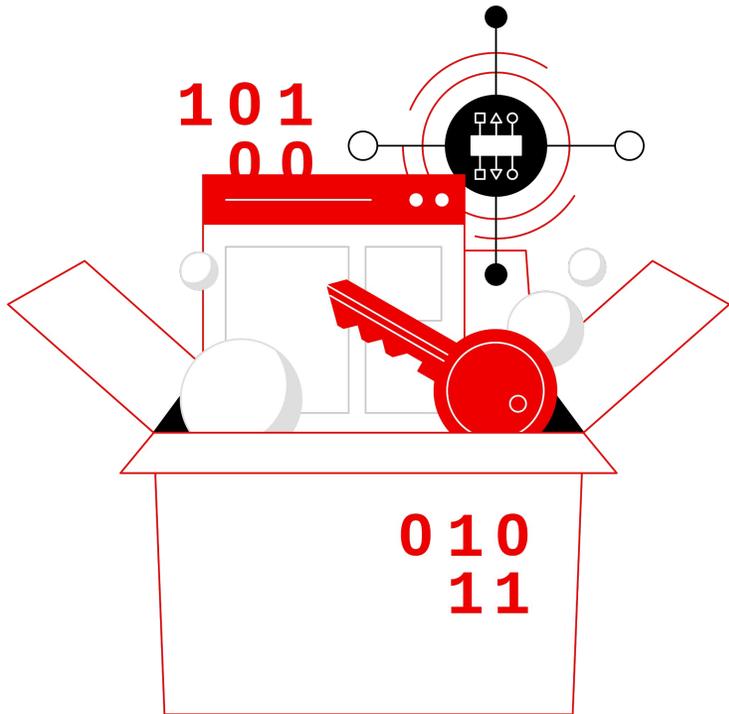
Pipelines [Import pipeline](#)

Pipeline name	Last run	Last run status	Last run time	Created
▼ Hugging Face Created with Elyra 3.15.0 pipeline editor using <code>untitled2.pipeline</code> .	Hugging Face-1104094346	Running	-	3 minutes ago
Runs Hugging Face-1104094346		Running	-	3 minutes ago

Models and model servers [Add server](#)

Model Server Name	Serving Runtime	Deployed models	Tokens
Custom Model Server	Triton runtime 23.01 - added on 20230815	4	Tokens disabled

workbenches



Notebook image

- Minimal Python 2023.1 [?](#)
Python v3.9
[› Versions](#)
- CUDA 2023.1 [?](#)
CUDA v11.8, Python v3.9
[› Versions](#)
- TensorFlow 2023.1 [?](#)
CUDA v11.8, Python v3.9, TensorFlow v2.11
[› Versions](#)
- Standard Data Science 2023.1 [?](#)
Python v3.9
[› Versions](#)
- PyTorch 2023.1 [?](#)
CUDA v11.8, Python v3.9, PyTorch v1.13
[› Versions](#)
- TrustyAI [?](#)
Python v3.9

storage and data connections



Red Hat OpenShift Data Science

Applications > Data Science Projects > Data Science Pipelines > Model Serving > Resources > Settings

Data science projects > Summit Connect

Summit Connect

Components Permissions

Jump to section

- Workbenches
- Cluster storage
- Data connections
- Pipelines
- Models and model servers

Workbenches

Create workbench

Name	Notebook image	Container size	Status	
Demo	TensorFlow	Small	Stopped	Open
Hugging Face	PyTorch	Medium	Running	Open

Cluster storage

Add cluster storage

Name	Type	Connected workbenches	
Demo	Persistent storage	Demo	
Hugging Face	Persistent storage	Hugging Face	

Data connections

Add data connection

Name	Type	Connected workbenches	Provider	
My Storage	Object storage	Hugging Face	AWS S3	
Pipeline Artifacts	Object storage	No connections	AWS S3	

Pipelines

Import pipeline

Pipeline name	Last run	Last run status	Last run time	Created	
Hugging Face Created with Elyra 3.15.0 pipeline editor using <code>untitled2.pipeline</code> .	Hugging Face-1104094346	Running	-	3 minutes ago	
Runs Hugging Face-1104094346		Running	-	3 minutes ago	

Models and model servers

Add server

Model Server Name	Serving Runtime	Deployed models	Tokens	
Custom Model Server	Triton runtime 23.01 - added on 20230815	4	Tokens disabled	Deploy model

pipelines

The image displays the Red Hat OpenShift Data Science interface, showing a pipeline workflow for text-to-image generation. The interface is divided into several sections:

- Top Bar:** Red Hat OpenShift Data Science logo and navigation menu.
- Left Sidebar:** Navigation menu with options: Applications, Data Science Projects, Data Science Pipelines, Model Serving, and Resources.
- Header:** Pipeline name: "Finetune Text to Image-0615194959" with a "Running" status indicator.
- Main Canvas:** A visual representation of the pipeline workflow. The workflow starts with "download-data", followed by "train". The "train" step branches into three parallel paths:
 - Path 1: "convert-text" followed by "upload".
 - Path 2: "convert-u-net" followed by "upload-2".
 - Path 3: "convert-vae" followed by "upload-3" and "upload-4".
- Bottom Panel:** A detailed view of the pipeline workflow, showing the sequence of steps: "Download Data" -> "Train" -> "Convert Text" -> "Upload Text", "Convert U-net" -> "Upload U-net", and "Convert VAE" -> "Upload VAE E..." -> "Upload VAE D...".

model serving



Red Hat OpenShift Data Science

Applications > Data Science Projects > Data Science Pipelines > Model Serving

Summit Connect

Components | Permissions

Jump to section

Workbenches [Create workbench](#)

Name	Notebook image	Container size	Status	
Demo	TensorFlow	Small	Stopped	Open
Hugging Face	PyTorch	Medium	Running	Open

Cluster storage [Add cluster storage](#)

Name	Type	Connected workbenches	
Demo	Persistent storage	Demo	
Hugging Face	Persistent storage	Hugging Face	

Data connections [Add data connection](#)

Name	Type	Connected workbenches	Provider	
My Storage	Object storage	Hugging Face	AWS S3	
Pipeline Artifacts	Object storage	No connections	AWS S3	

Pipelines [Import pipeline](#)

Pipeline name	Last run	Last run status	Last run time	Created	
Hugging Face Created with Elyra 3.15.0 pipeline editor using <code>untitled2.pipeline</code> .	Hugging Face-1104094346	Running	-	3 minutes ago	
Runs Hugging Face-1104094346		Running	-	3 minutes ago	

Models and model servers [Add server](#)

Model Server Name	Serving Runtime	Deployed models	Tokens	
Custom Model Server	Triton runtime 23.01 - added on 20230815	4	Tokens disabled	Deploy model

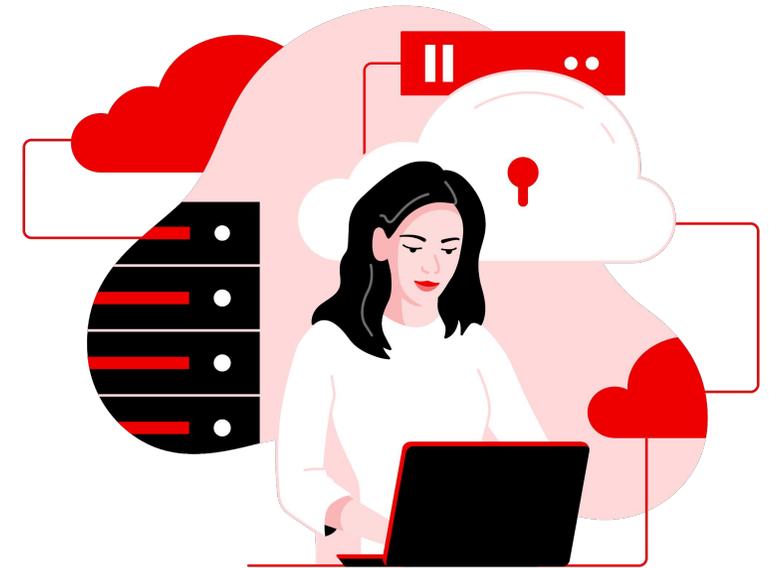
model serving

The screenshot shows the Red Hat OpenShift Dedicated console interface. The left sidebar contains navigation options: Developer, +Add, Topology, Observe, Search, Builds, Pipelines, Helm, Project, ConfigMaps, and Secrets. The main content area is titled 'Project: my-data-science-project' and 'Application: All applications'. It features several configuration sections:

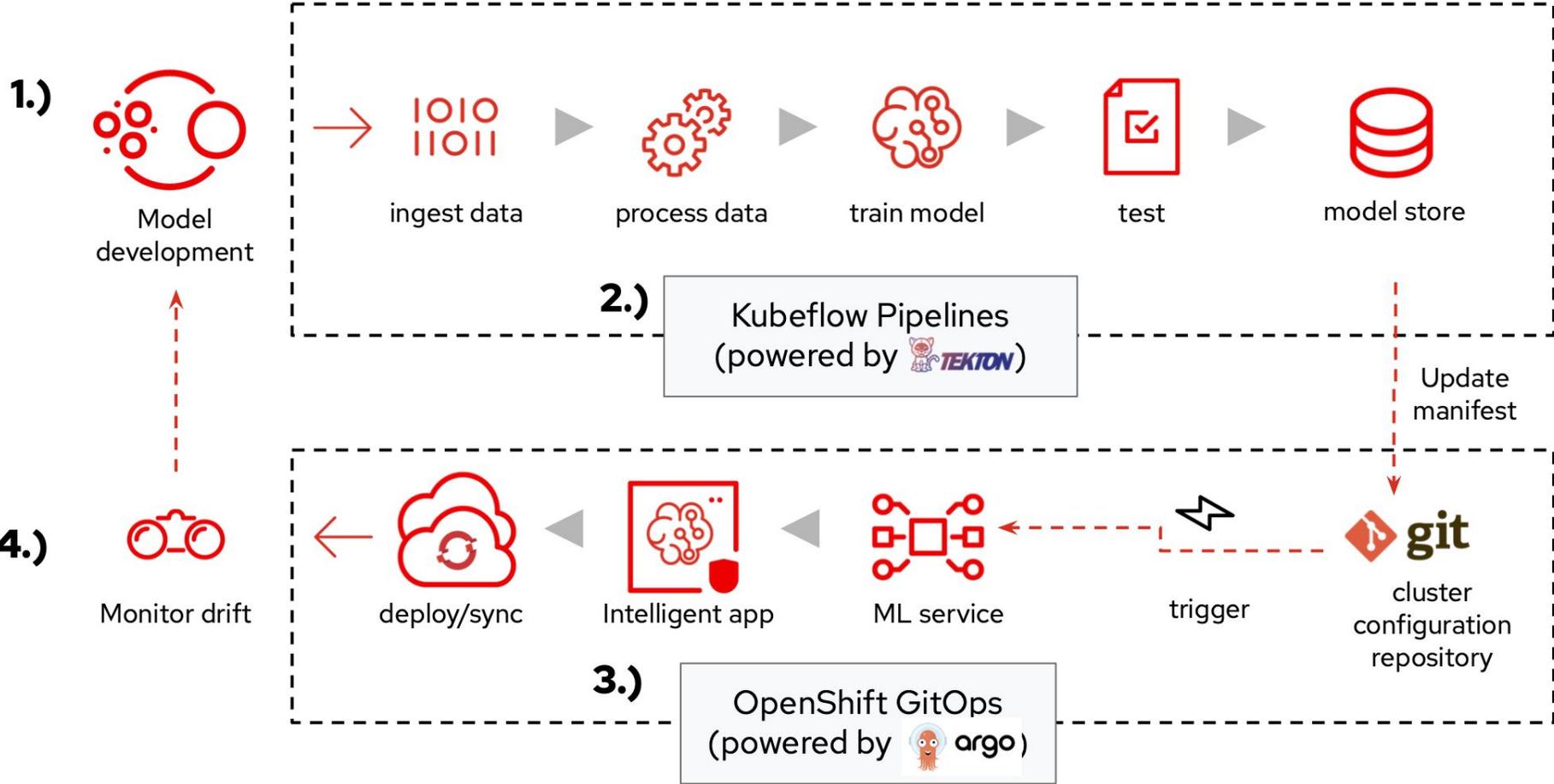
- Deployment Options:** A checked checkbox for 'Deploy image from an image stream tag' and another checked checkbox for 'Auto deploy when new Image is available'. A link for 'Show advanced image options' is also present.
- Image Stream Tag:** A form with three dropdown menus: 'Project' (my-data-science-pr...), 'Image Stream' (ai-photo-album), and 'Tag' (latest).
- Environment Variables:** A table listing variables for the 'ai-photo-album' container.

Name	Value
DEBUG_PORT	5858
PROJECTS_ROOT	/projects
PROJECT_SOURCE	/projects
MODEL_SERVER_GRPC	-serving.my-data-science-project:8033

At the bottom, there are buttons for '+ Add value', '+ Add from ConfigMap or Secret', 'Save', 'Reload', and 'Cancel'.

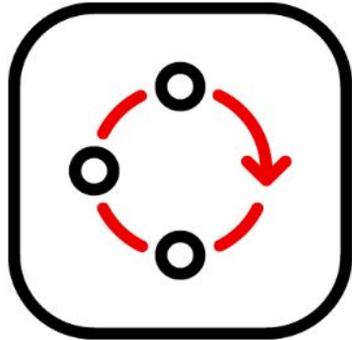


lifecycle



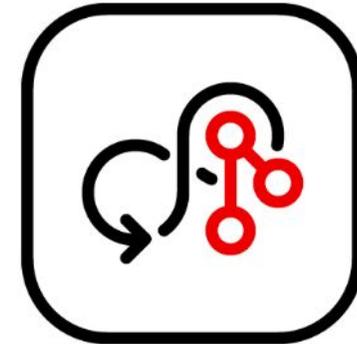


Red Hat
Advanced Cluster
Management
for Kubernetes



Red Hat OpenShift Pipelines

DevSecOps?



Red Hat OpenShift GitOps



Red Hat
Advanced Cluster
Security
for Kubernetes



Pipelines > Pipeline details

PL build-and-deploy Actions ▾

[Details](#) [Metrics](#) [YAML](#) [PipelineRuns](#) [Parameters](#) [Resources](#)

Pipeline details

```
graph LR; A[download-data] --> B[train]; B --> C[convert-text]; B --> D[convert-unet]; B --> E[convert-vae]; C --> F[upload-text]; D --> G[upload-unet]; E --> H[upload-vae]; F --> I[fetch-app-repository]; G --> I; H --> I; I --> J[build-image]; I --> K[scan-with-acs]; J --> L[update-manifests-to-git]; K --> L; L --> M[push-to-git]; I --> N[update-model-serv-config-to-git]; N --> O[push-configs-to-git];
```

🔍 🔍 ✂️ 🖼️

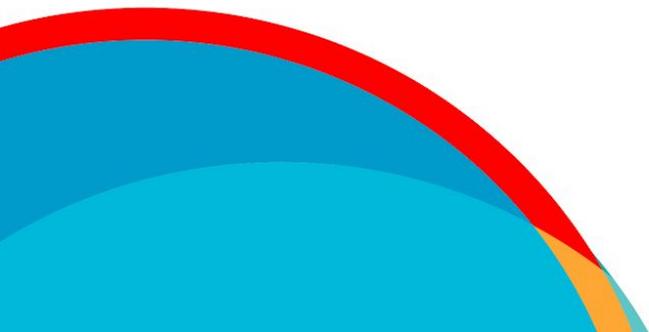
Summary



Red Hat OpenShift AI is an AI-focused portfolio that provides tools across the full lifecycle of AI/ML experiments and models and includes Red Hat OpenShift Data Science.

Red Hat OpenShift Data Science is an open source machine learning (ML) platform for the hybrid cloud.

By providing a fully supported environment to establish MLOps best practices, data scientists and developers can rapidly train, deploy and monitor ML workloads and models on-premise and in the public cloud.



<https://developers.redhat.com/learn/openshift-data-science>



Red Hat
Summit

Thank you



linkedin.com/company/red-hat



facebook.com/redhatinc



youtube.com/user/RedHatVideos



twitter.com/RedHat