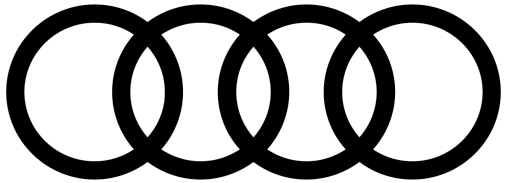


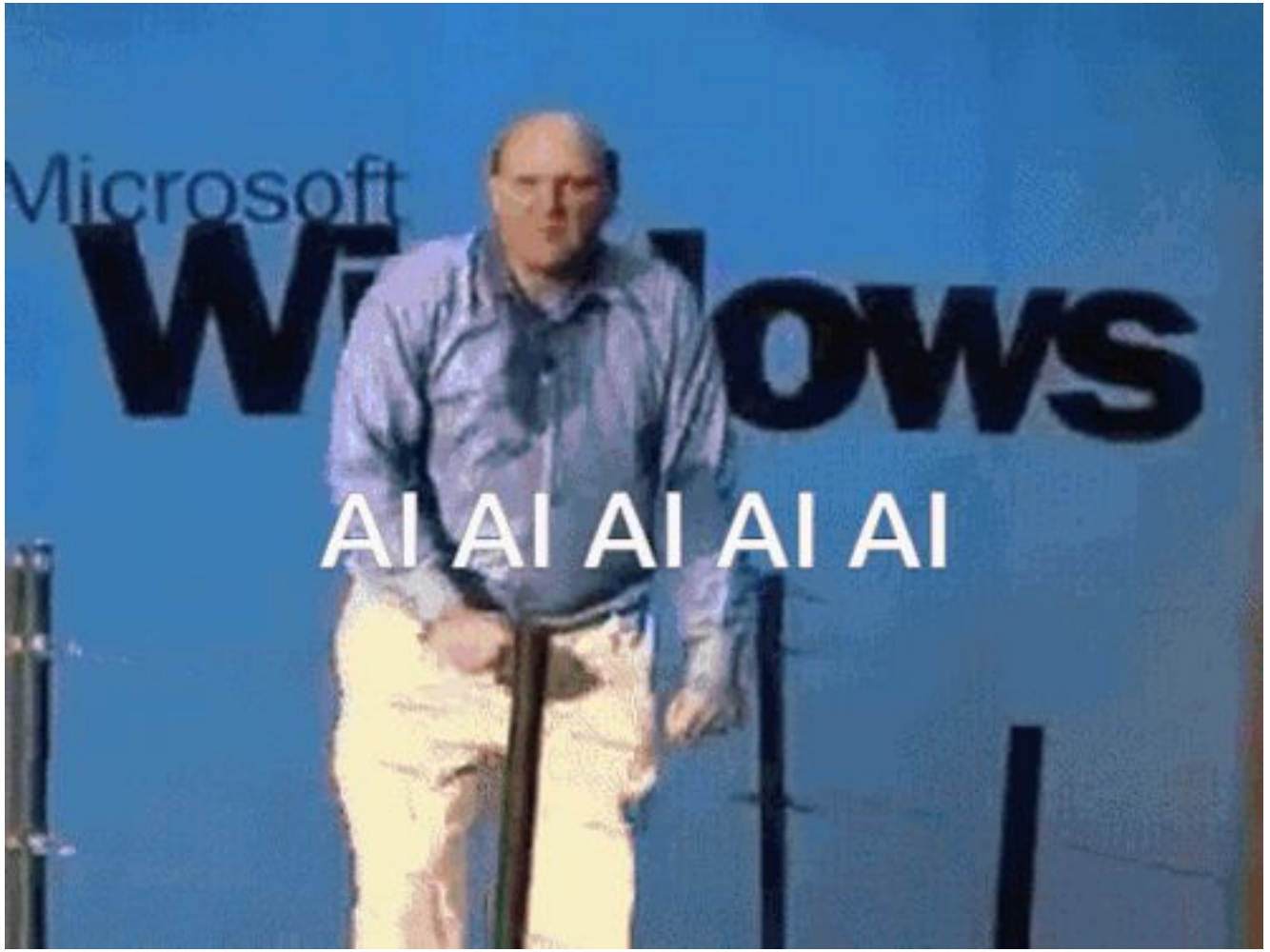
Audi

GroupAI

November 2025

What we'll discuss today

- Motivational Speech
- Red Hat AI Inference Server Overview
- PoC takeaways
- What is next
- Next Steps


Microsoft
AI AI AI AI AI

https://rhods-dashboard-redhat-ods-applications.apps.ocp.ocp-gm.de/projects/hello-world

https://maas.apps.prod.rhoai.rh-aiservices-bu.com/admin/applications/75149

What is what?

Little tour on the buzzwords

What is AI inference?

An AI inference server is the software that helps an AI model make the jump from training to operating. It uses machine learning to help the model apply what it's learned and put it into practice to generate inferences.

After successful training, the model can make inferences such as identifying a breed of dog, recognizing a cat's meow, or even delivering a warning around a spooked horse. Even though it has never seen these animals outside of an abstract data set before, the extensive data it was trained on allows it to make inferences in a new environment in real time.

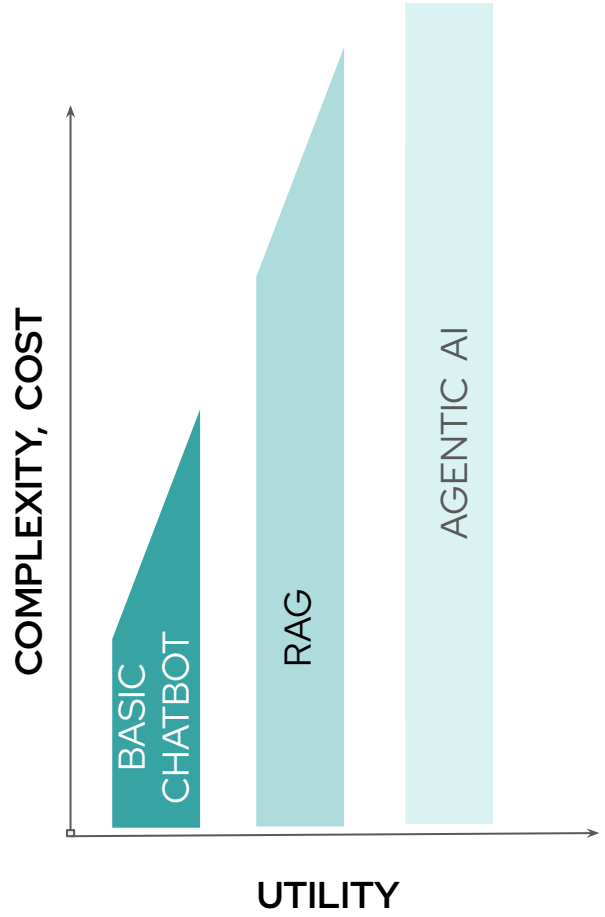
What is vLLM?

vLLM, which stands for virtual large language model, is a library of open source code maintained by the vLLM community. It helps large language models (LLMs) perform calculations more efficiently and at scale.

Specifically, vLLM is an inference server that speeds up the output of generative AI applications by making better use of the GPU memory.

The Hidden Cost of Generative AI

More Advanced Techniques Incur Higher Costs

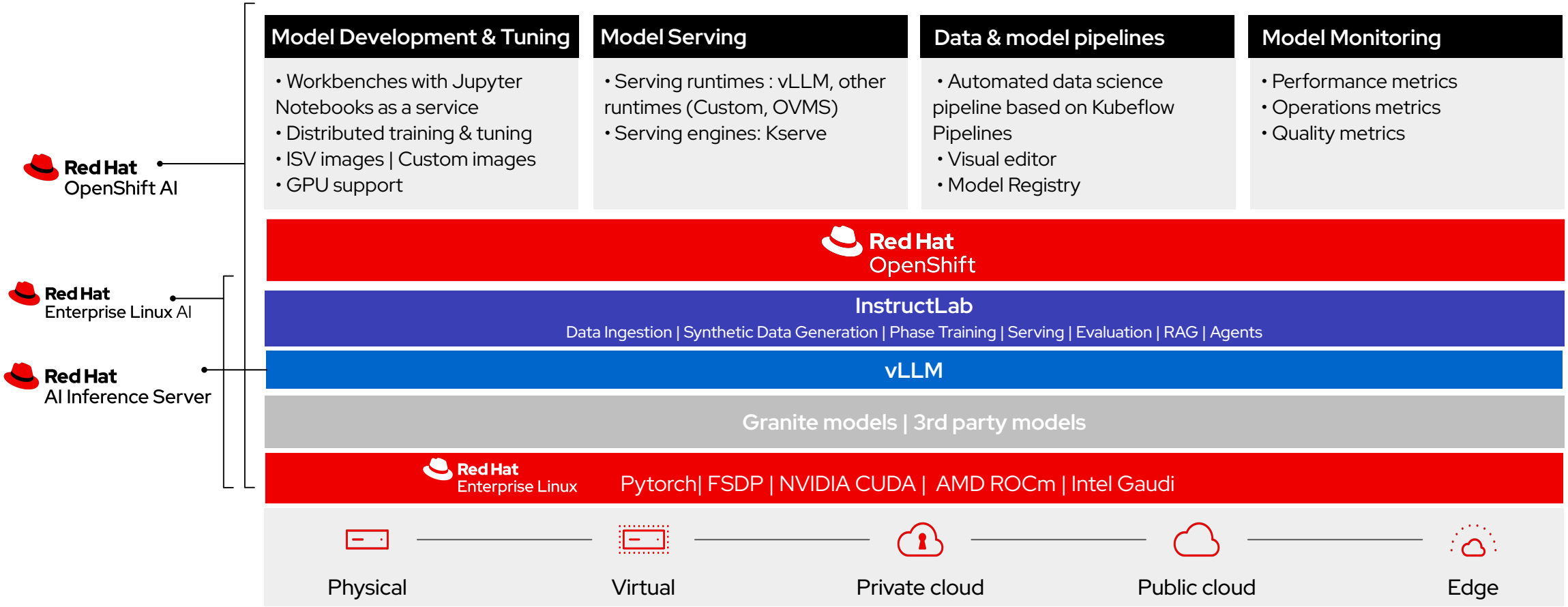


- **Basic chatbots** that provide direct LLM querying are relatively cheap to operate but offer limited utility.
- **Retrieval augmented generation (RAG)** allows tapping into proprietary knowledge. The cost per query multiplies.
- **Agentic AI** provides arbitrary integration and query iteration capabilities. One user query may generate many high-cost LLM queries.

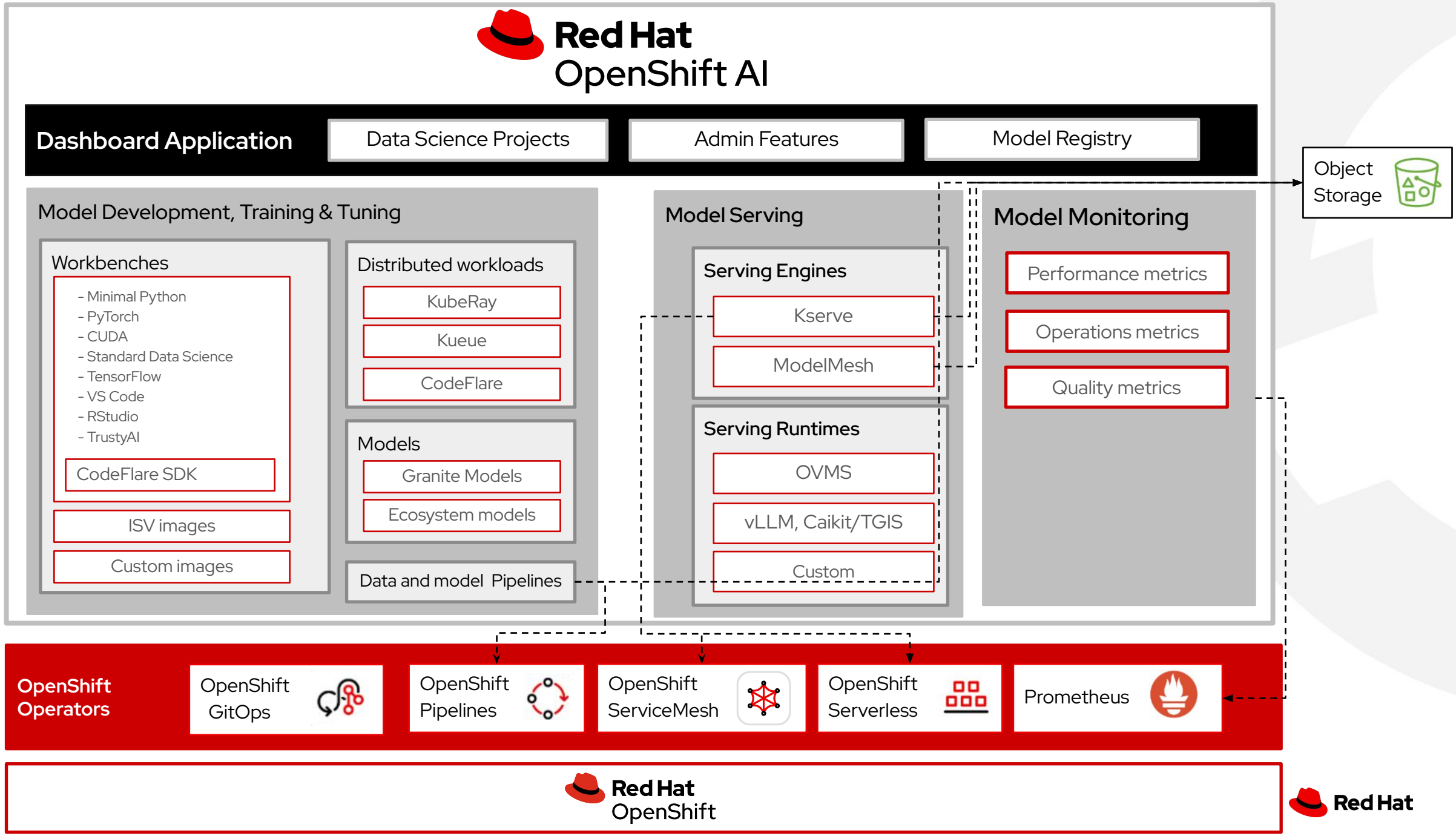
Red Hat

Red Hat AI platform

Generative AI, Predictive AI & MLOps capabilities for building flexible, trusted AI solutions at scale



Red Hat


Red Hat
OpenShift AI
Dashboard Application
Data Science Projects
Admin Features
Model Registry
Object Storage
Model Development, Training & Tuning
Workbenches
- Minimal Python
- PyTorch
- CUDA
- Standard Data Science
- TensorFlow
- VS Code
- RStudio
- TrustyAI
CodeFlare SDK
ISV images
Custom images
Distributed workloads
KubeRay
Kueue
CodeFlare
Models
Granite Models
Ecosystem models
Data and model Pipelines
Model Serving
Serving Engines
Kserve
ModelMesh
Serving Runtimes
OVMS
vLLM, Caikit/TGIS
Custom
Model Monitoring
Performance metrics
Operations metrics
Quality metrics
OpenShift Operators
OpenShift GitOps
OpenShift Pipelines
OpenShift ServiceMesh
OpenShift Serverless
Prometheus
Red Hat OpenShift

Red Hat

OpenShift AI Feature Overview

RHOAI v. 2.25

Retrain models

DataOps

- Connections
- S3 browser
- **Feature store**

Hardware

- Hardware profiles
- Accelerators
  - Nvidia
  - Intel
  - AMD
  - IBM AIU (serving)
  - IBM Spyre AI (serving)
- CPU Architectures
  - x86
  - IBM Power & Z (serving)
  - ARM

Model Development

- **Workbenches**
  - JupyterLab
  - VS Code
  - R Studio
  - LLM Compressor
- **Data Science Pipelines**
  - Experiment tracking
  - Pipeline versioning
  - Artifact tracking
- **Distributed Workloads**
  - Hyperparameter tuning
  - Kueue scheduling
  - KubeRay
  - Kubeflow Training Operator
  - LAB model customization

Integrate models in app dev

- **Model Serving Runtimes**
  - OpenVINO Model Server
  - vLLM
  - TGIS
  - Caikit
  - Nvidia Triton*
  - ML Server*
  - Nvidia NIM**
- **Model Serving Modes**
  - Single-model serving
  - Multi-model serving
  - Distributed serving
  - Disaggregated serving
  - Near edge deployments
  - Metrics-based autoscaling
- **Agentic AI**
  - Llama Stack
  - GenAI Studio

Model monitoring and management

- OCI model support
- **Model Registry**
- **Model Catalog**
- **Models-as-a-Service**
- **Monitoring**
  - Operational metrics
  - Runtime metrics
  - Data drift detection
  - Bias detection
  - LM evaluation
  - LM evaluation UI
  - LLM guardrails

Preview

Roadmap

This is an uncommitted roadmap and Red Hat reserves the right to change it.

* certified and tested runtime, self-support

** requires Nvidia license

Red Hat

Red Hat and kubiya.ai - better together

- Faster Time-to-Production
  Kubiya automates pipelines and workflows; Red Hat OpenShift AI provides scalable model serving — helping enterprises move from idea to deployment in minutes.
- Execution & Operations
  Kubiya adds an execution layer with RBAC, approvals, audit, and rollback; Red Hat handles model lifecycle and inference — making Day-2 operations safe and predictable.
- Leverage Existing Investments
  Kubiya integrates with existing infrastructure, processes, and tools; Red Hat provides the enterprise hybrid-cloud backbone — avoiding new silos and extra vendor costs.
- Enterprise Adoption
  Kubiya ensures secure, compliant workflows; Red Hat provides hardened infrastructure — reducing risk and cost at scale.
- Joint GTM & Velocity
  Kubiya drives usage through workflow automation; Red Hat amplifies reach with OpenShift — together accelerating adoption and market impact.

Together: Kubiya + Red Hat deliver production-ready AI on OpenShift — faster, safer, and with accelerated customer adoption. **With Red Hat OpenShift AI as a core component**

Flavors of RHOAI

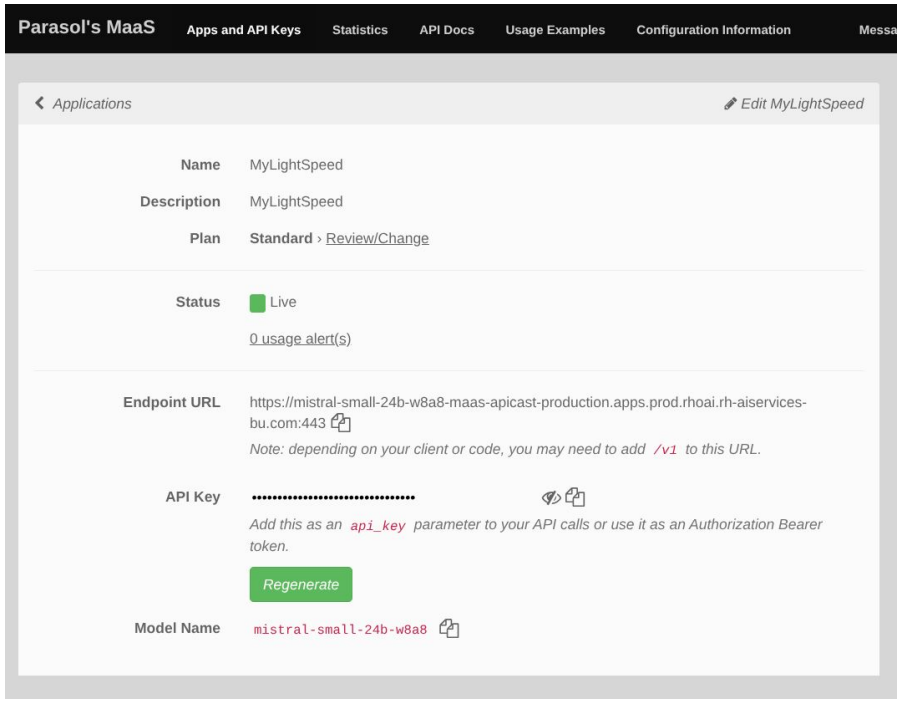
Supported deployment options		
Options available	**Self-managed RHOAI**	**Cloud Service RHOAI**
Bare metal	✓	
Virtual	✓	
Private cloud	✓	
Red Hat OpenShift on AWS (ROSA)	✓	✓
Azure Red Hat OpenShift (ARO)	✓	(future)
IBM Cloud	✓	
OSD-GCP/OSD-AWS	✓	✓
Edge (with SNO)	✓	

GroupAI

Red Hat

Example which we use ourselves

https://maas.apps.prod.rhoai.rh-aiservices-bu.com/admin/applications/75149



Red Hat

What we'll discuss today

- ~~Motivational Speech~~
- Red Hat AI Inference Server Overview
- PoC takeaways
- What is next
- Next Steps

What is what?

Little tour on the buzzwords

What are parameters

- Model Parameters: These are learned during training and include **weights and biases**. They are specific to the model's architecture and are adjusted based on the training data.
- Hyperparameters: These are set before training and influence the training process itself, such as the **learning rate or the number of training epochs**. Unlike model parameters, hyperparameters are **not learned from the data** but are manually configured.

[Data Poisoning bei LLMs: Feste Zahl Gift-Dokumente reicht für Angriff | heise online]

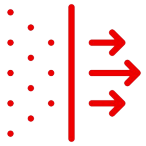
[...]
Sofern sich die Ergebnisse bestätigen, wäre die Ansicht, dass das Vergiften von KI-Daten wie "ins Meer pinkeln" sei, **wissenschaftlich widerlegt**.
Ein einzelner Akteur benötigt keine riesigen Ressourcen, um Schaden anzurichten
[...]


(Bild: busliq/Shutterstock.com)

The Operational Challenges in the Inference Era

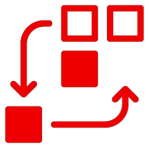
Infrastructure cost

Requires substantial compute power to deliver expected experience



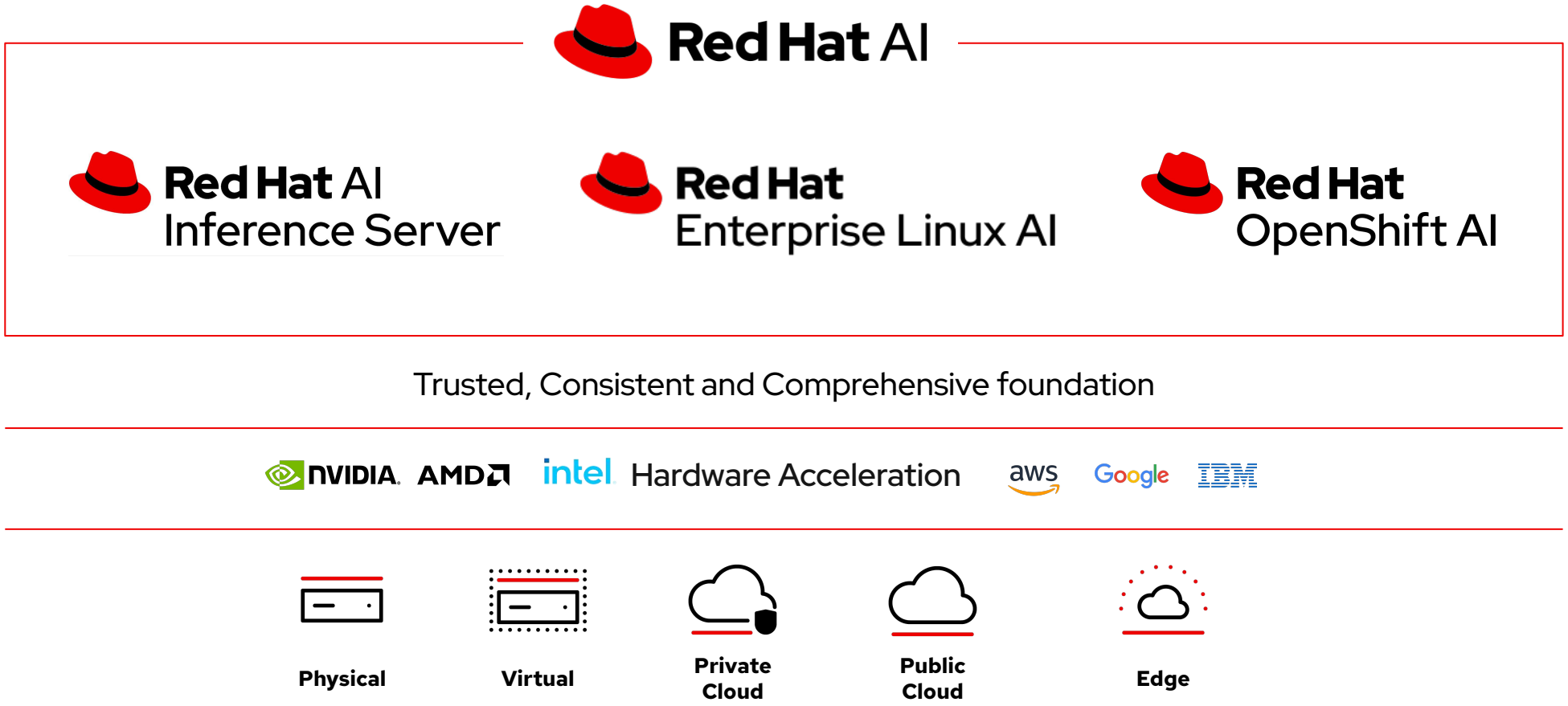
Operational complexities

Non- standardized approach creates inefficiencies



Deployment constraints

Inference across hybrid environments can lack flexibility



* NVIDIA GPUs fully supported in Red Hat AI. AMD Instruct $ Intel Gaudi in Preview, AWS Inferentia/Neuron, Google TPU, IBM AIU are on our roadmap

Fast

Cost-effective

Optimized

Red Hat AI
Inference Server

Any gen AI model

Any AI Accelerator

Any environment

Red Hat AI Inference Server

Gain consistent, fast and cost-effective inference at scale

Inference runtime for the hybrid cloud

Run your models of choice across any accelerator and any environment

Compress Models

Reduce compute and costs while preserving accuracy

Red Hat AI Hugging Face repository

Access a collection of third-party validated and optimized models ready for inference.

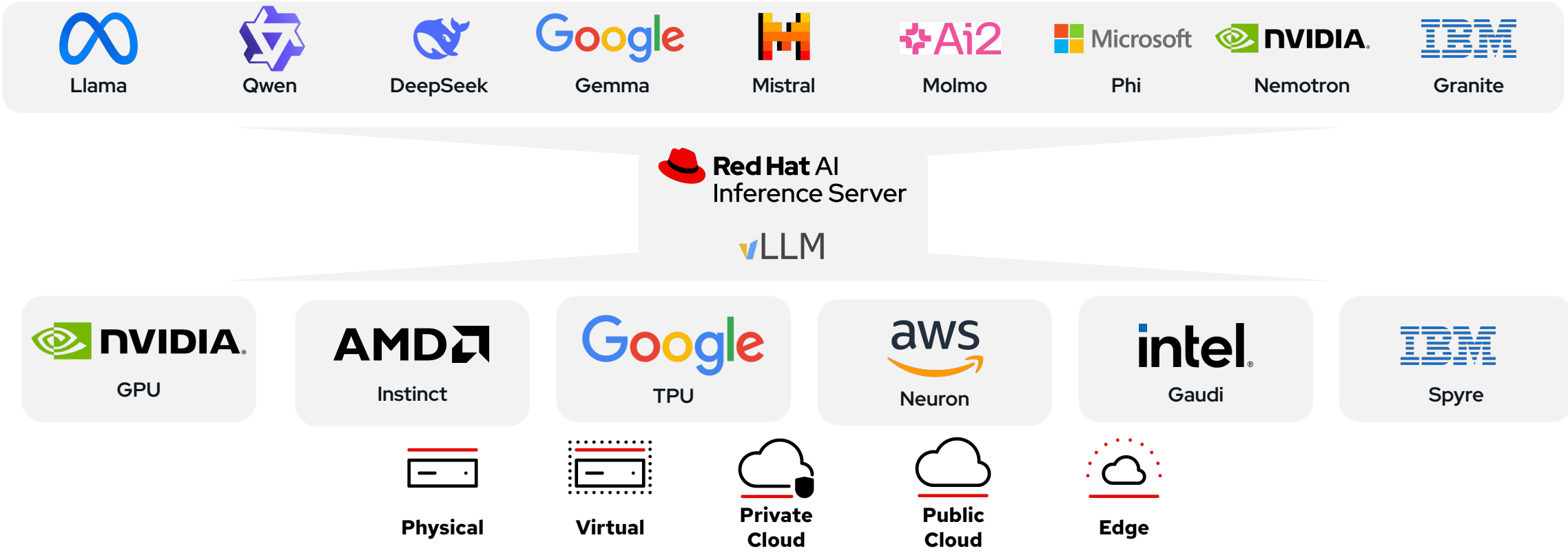
Certified for all Red Hat products

Deployable across non-Red Hat Linux and Kubernetes platforms

Red Hat AI Inference Server is included in OpenShift AI and RHEL AI

Red Hat AI Inference Server

vLLM connects model creators to accelerated hardware providers



Single platform to run any model, on any accelerator, on any cloud

Red Hat

Red Hat AI Inference Server

vLLM is emerging as the Linux of GenAI Inference

HIGH PERFORMANCE

- Advanced algorithms for high QPS serving
- Single server/GPU to distributed/multi GPU
- Already comparable to Nvidia (TRT-LLM)

EASY TO USE CAPABILITIES DRIVING DEVELOPER AND IT PRODUCTIVITY

- Native Hugging Face integration
- Simple APIs for online and offline inference
- OpenAI-compatible API protocol

Scalable inference across the hybrid cloud

Red Hat

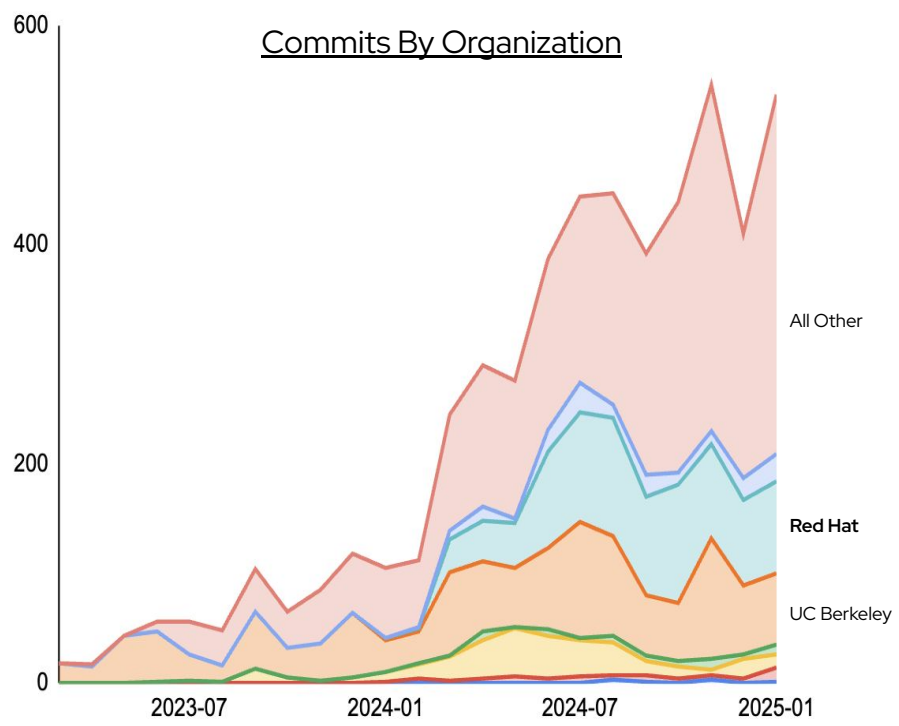
Red Hat: Leaders in OSS GenAI Inference

Expertise across high performance inference and SOTA model optimizations

Core Developers of vLLM

- HPC engineering team dedicated to vLLM, 7 core vLLM committers on staff
- Work on key subsystems, with a particular emphasis on fast model execution
- ML engineering team builds vLLM's optimization library llm-compressor
- ML research team create pre-optimized models for deployment with vLLM

Red Hat Community Contribution



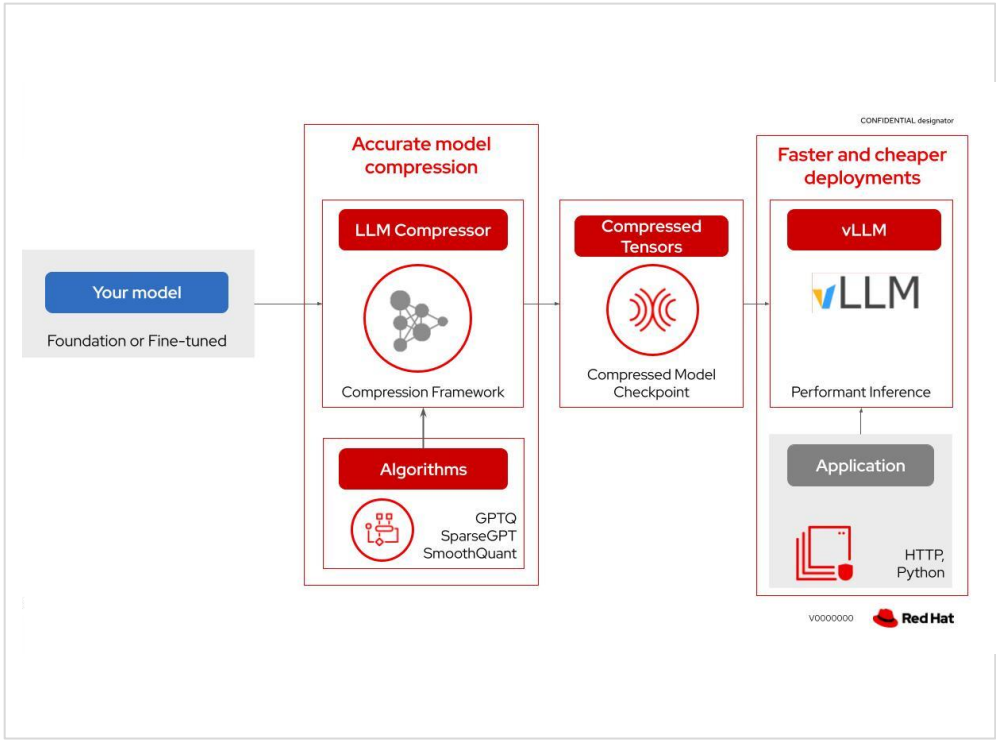
Red Hat: Leaders in OSS GenAI Inference

Red Hat has built a comprehensive set of model optimization capabilities to drive operational efficiencies

Third-party validated and optimized models

Llama
Qwen
Gemma
Mistral
DeepSeek
Phi
Molmo
Granite
Nemotron

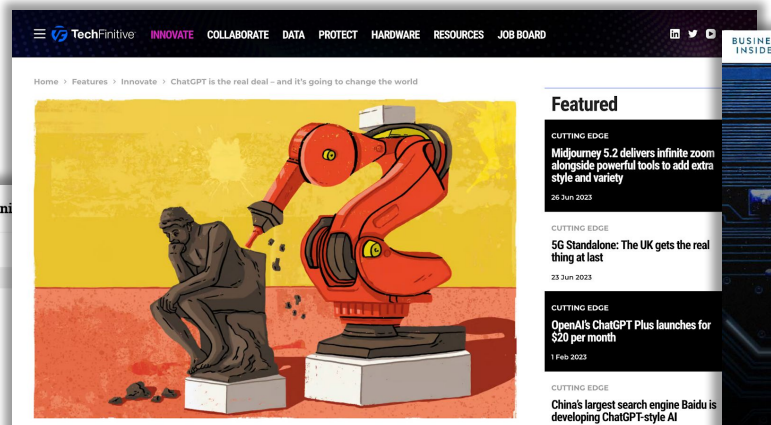
LLM Compression Tools

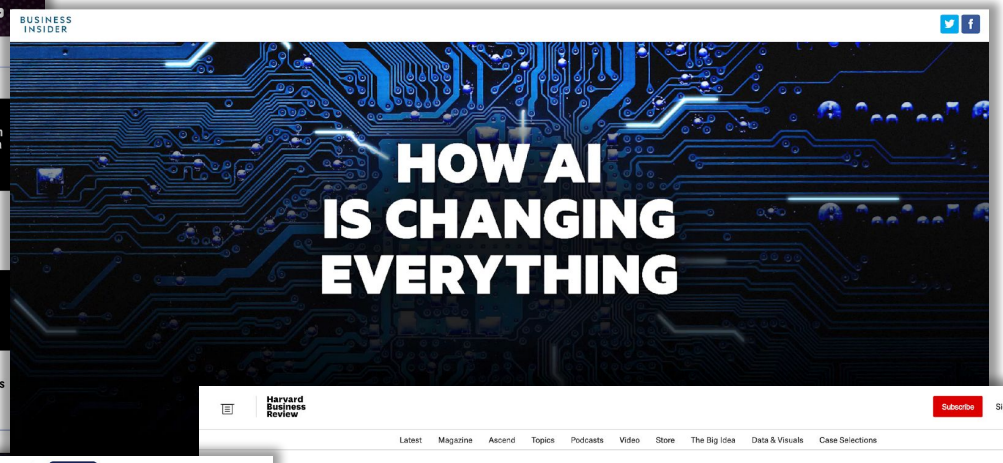


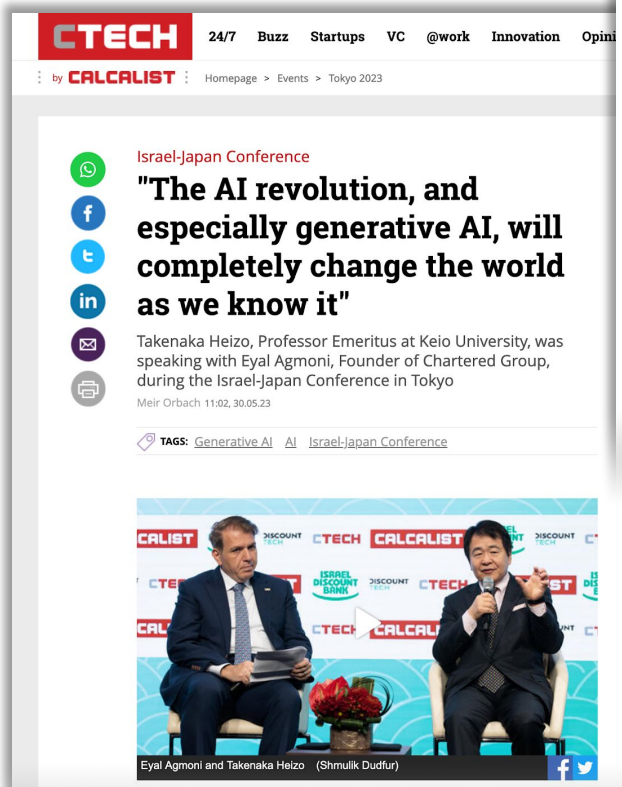
Hosted on the Red Hat AI repository on Hugging Face

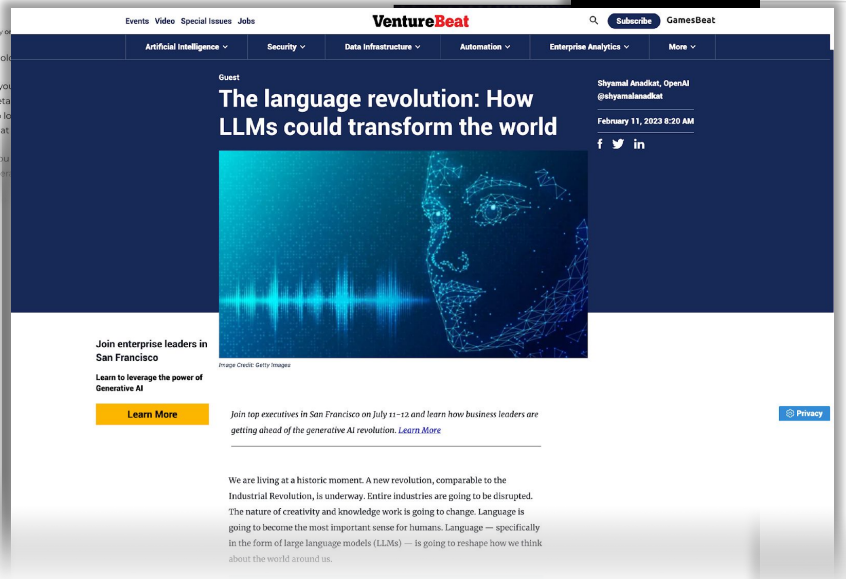
The World Changed in November 2022

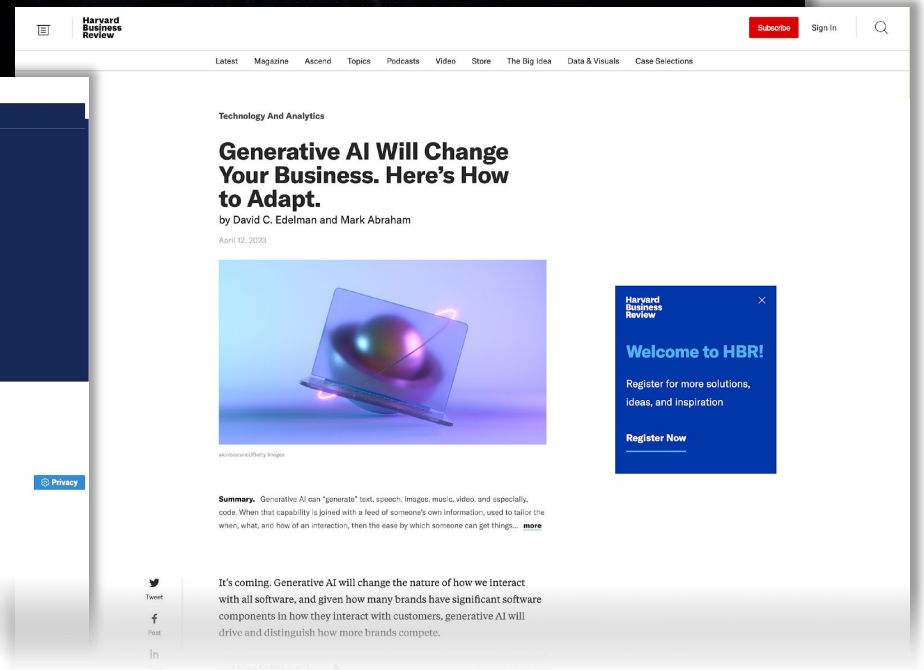
ChatGPT woke the world up to the power of generative AI







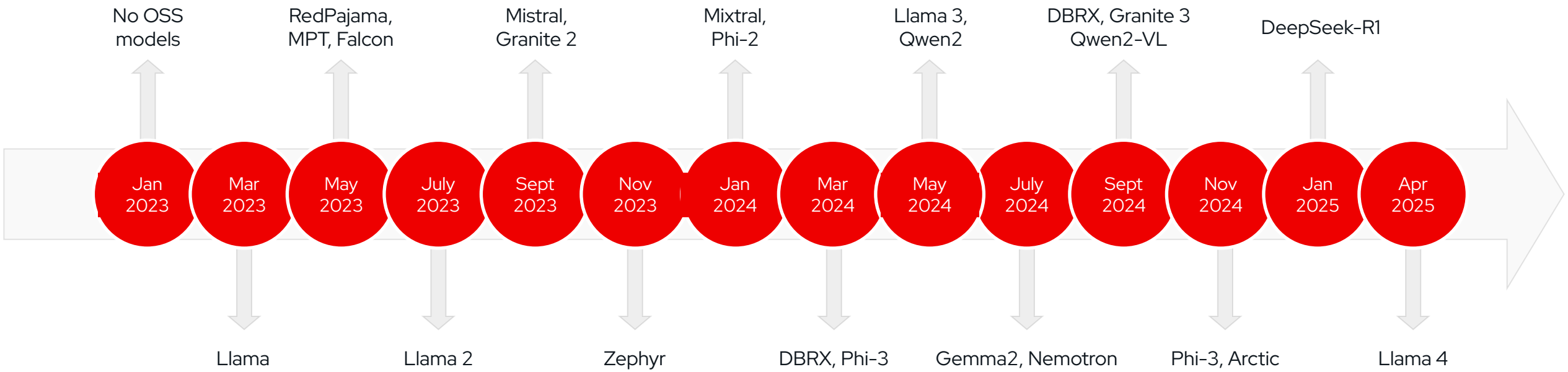




Red Hat

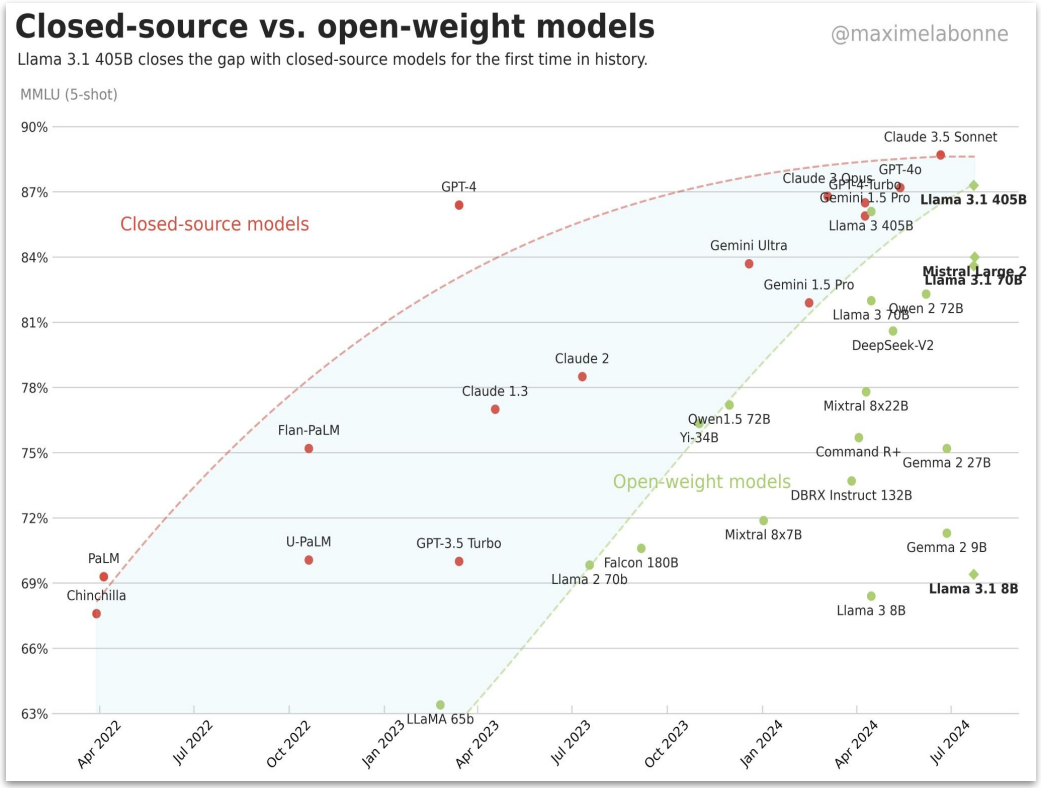
The Power of Open

There has been an explosion of capability from open-source over the last 2 years.



The Power of Open

Open models are deployment targets **today**. And the trend is not slowing down.



Headlines

Meta

Llama

- 650M downloads in 2024
- 85,000 Llama derivative models
- 1B, 3B, 8B, 70B, 405B variants
- Multilingual, Multimodal, Mobile

deepseek

R1

- First reasoning model on par in quality with OpenAI O1
- 1-70B parameter distilled versions
- Global market pandemonium?

Models are commoditizing → many options for diverse enterprise needs.

Advantages of Open Source Models

Open-source models play an important role in the Enterprise AI landscape

Customization

Improve accuracy and costs with task specific tuning

Security

Complete data privacy (no 3rd party APIs)

Control

Model lifecycle (no changes to the model in place) and Resources (no rate limits / API downtime)

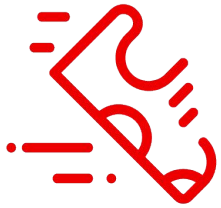
Cost

Self managed infrastructure. 1B-405B size - match task difficulty to model

Inference is becoming the gravity in AI because it is where the real world value happens

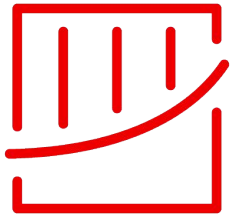
Always-on intelligence

Inference powers every user interaction 24/7, making inference the constant cost and performance driver



Latency defines experience

Low-latency and fast inference ensures optimal user experiences, essential for real-time applications and user retention

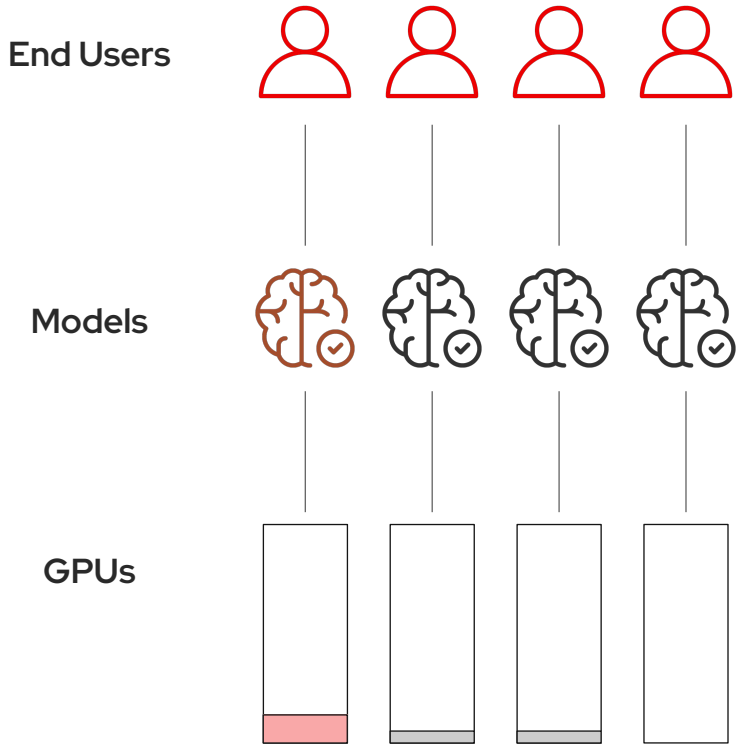


Exponential market momentum

The AI inference market is forecast to grow from USD 106.15 billion in 2025 to USD 254.98 billion by 2030, underscoring its role

Source:
What is AI Inference?, Jan 2025, Red Hat
AI Inference Market Size, Share & Trends, Feb 2025, Markets and Markets
Amazon Found Every 100ms of Latency Cost them 1% in Sales, July 2023, Gigaspaces

Infrastructure as a Service can be costly

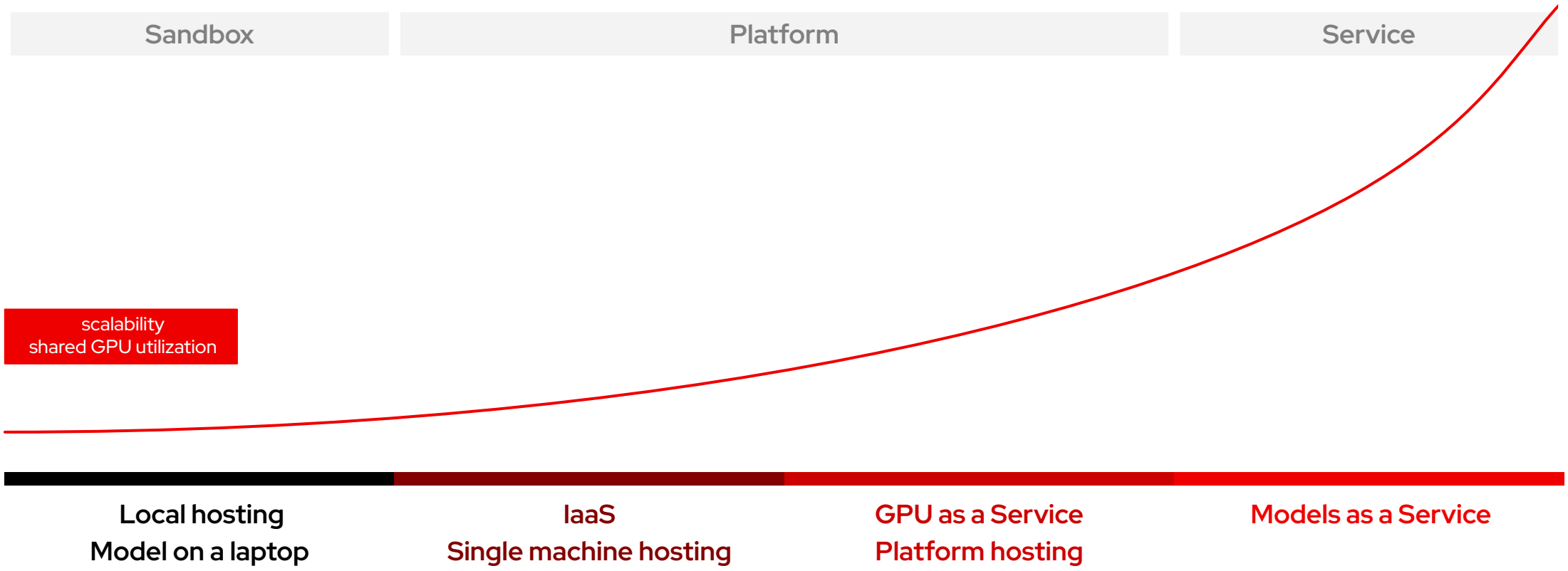


Self-Service AI could be good for small teams with ample resource, but can become risky and costly for wider audiences.

Most users primarily need an LLM endpoint, not simply direct GPU access.

Red Hat

Model Hosting Stages



Red Hat

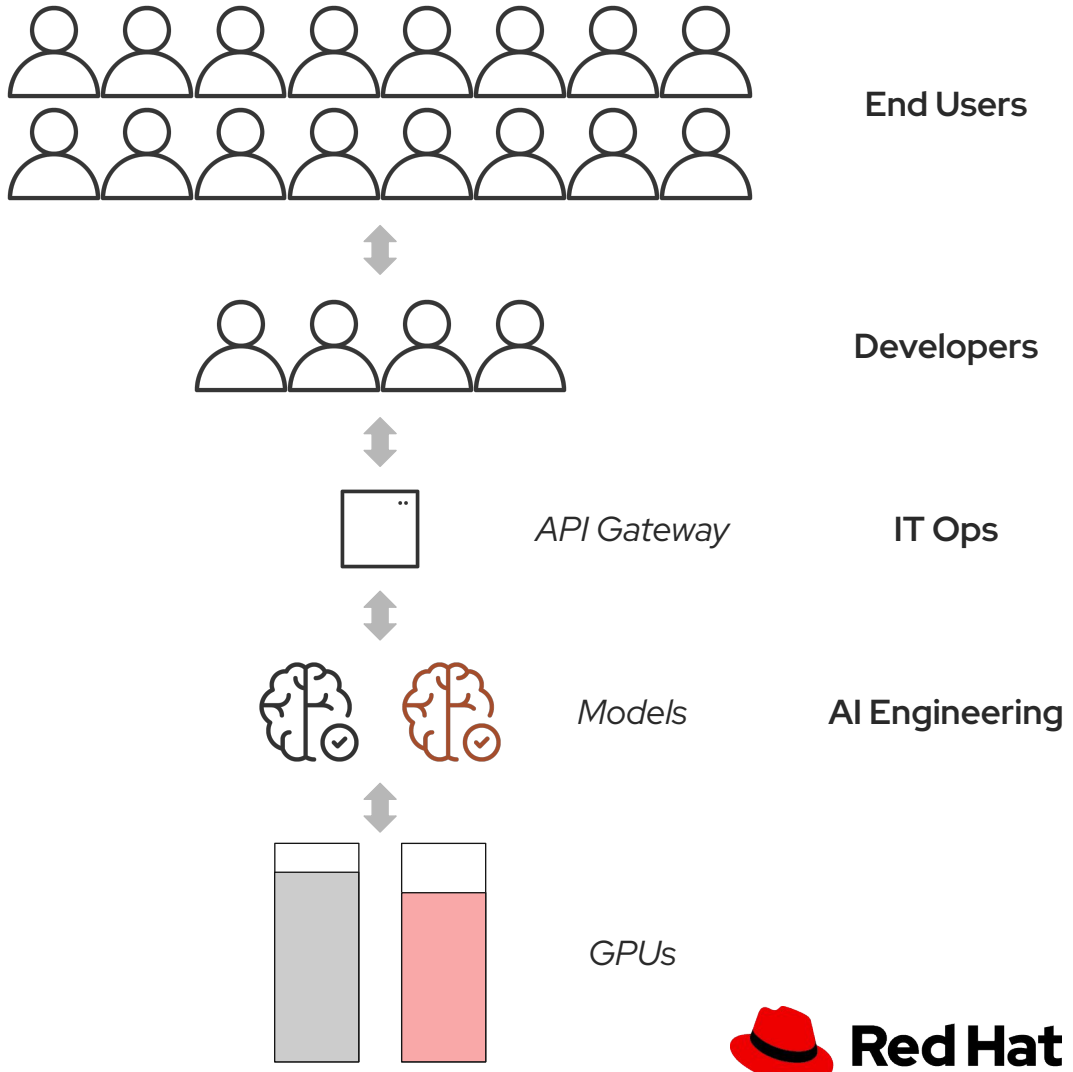
What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
- PoC takeaways
- What is next
- Next Steps

Introducing a Private-LLM-**as-a-Service** Pattern

Centralized AI Model Service for Broad Accessibility

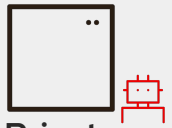
- IT centrally manages and governs common models, ensuring compliance and efficiency
- Models available through an API Gateway
- Developers consume models and build AI applications
- Shared Resources business model keeps costs down by optimizing GPU utilization



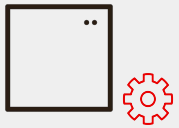
Red Hat

Become the **Private AI Provider** for your organization

Use cases



Private Assistant (RAG)



Embedding Application

Code Assistant

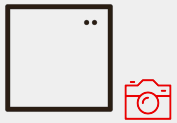
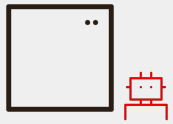


Image Generation



Agentic

Your next app

Red Hat can help

Red Hat OpenShift AI

Red Hat OpenShift

Red Hat Consulting

Red Hat 3scale API Management

\+ **API Gateway Partners**

Red Hat

Models as a Service – Requirements

As a **developer** I want to

- explore a catalog of available models with relevant descriptions and metadata,
- subscribe to individual models and receive credentials for programmatic access,
- rely on high quality of service.

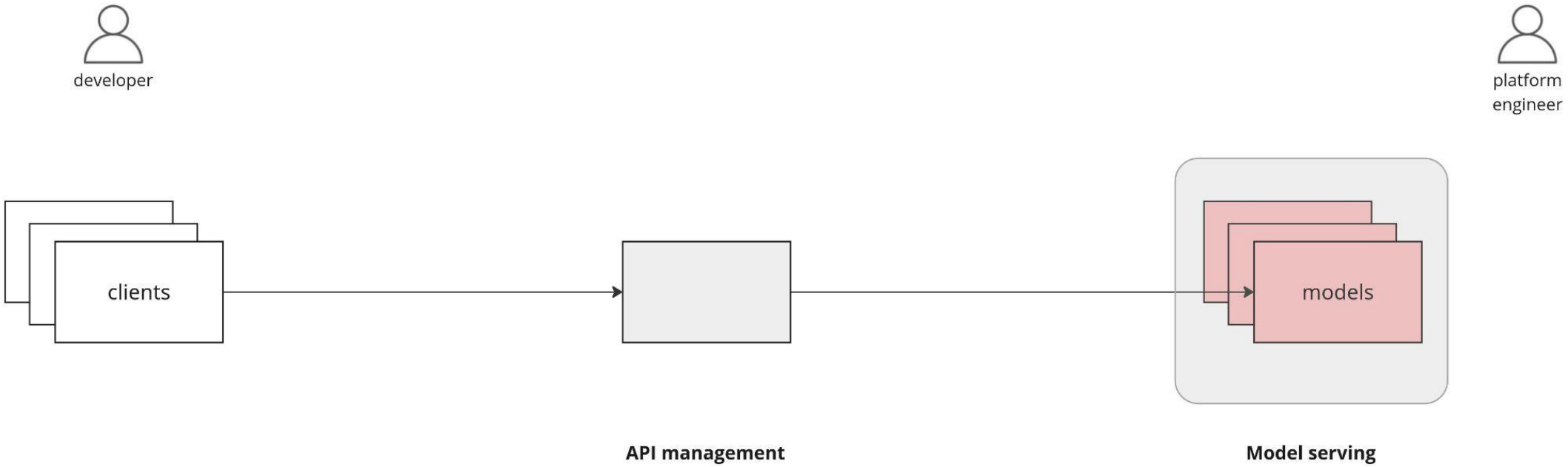
As a **platform engineer** I want to

- track usage of individual model endpoints and create reports for billing,
- enforce rate limits if needed,
- leverage existing identity providers for authentication.

Models as a Service – Architecture

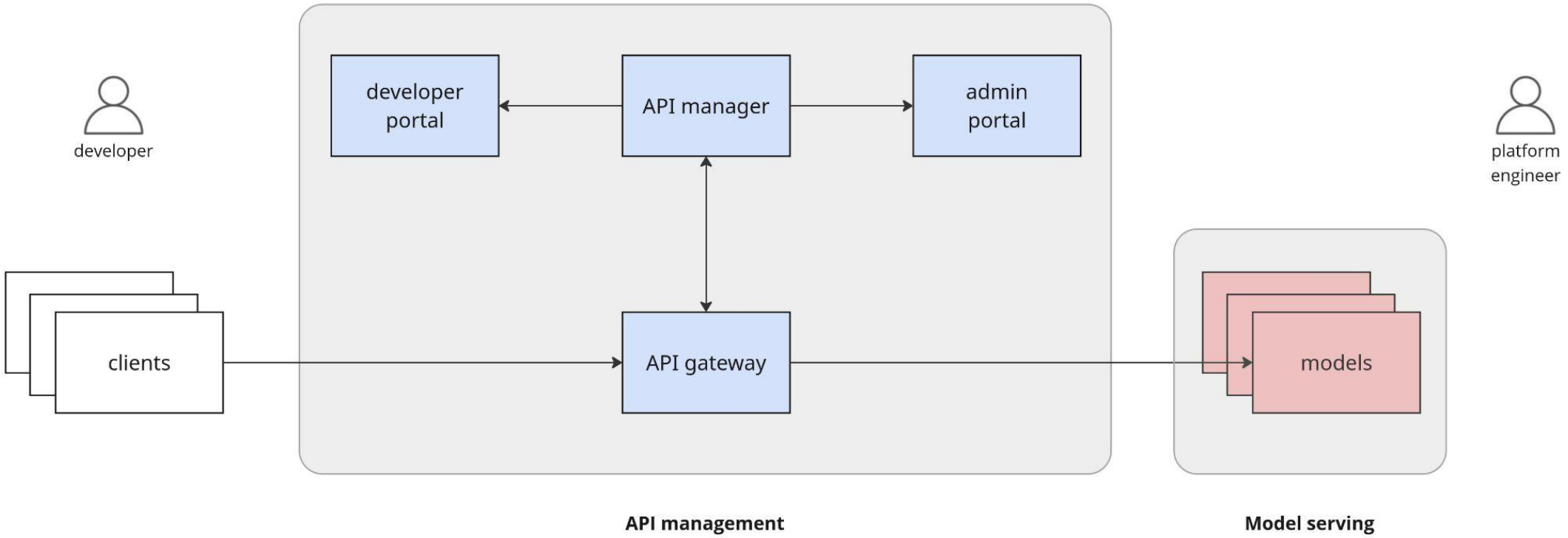
Red Hat

Models as a Service – Architecture



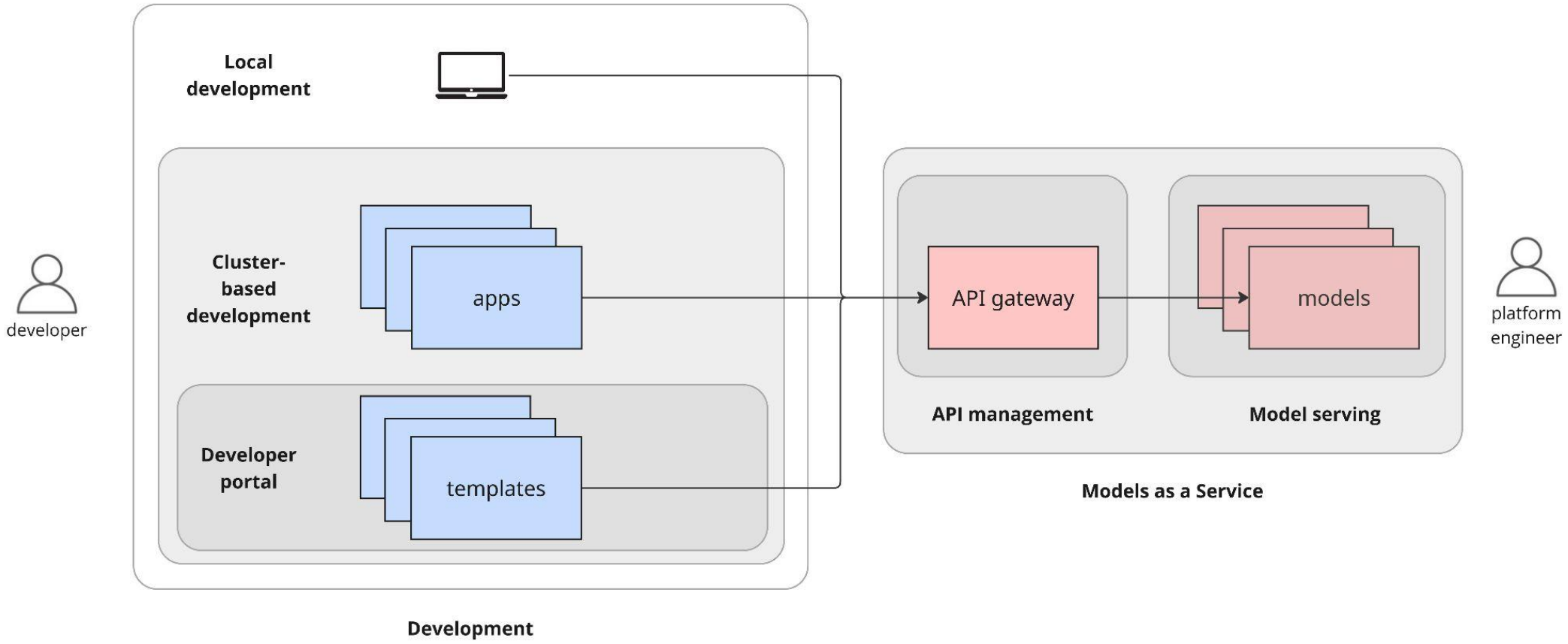
Red Hat

Models as a Service – Architecture



Red Hat

Models as a Service & Developers



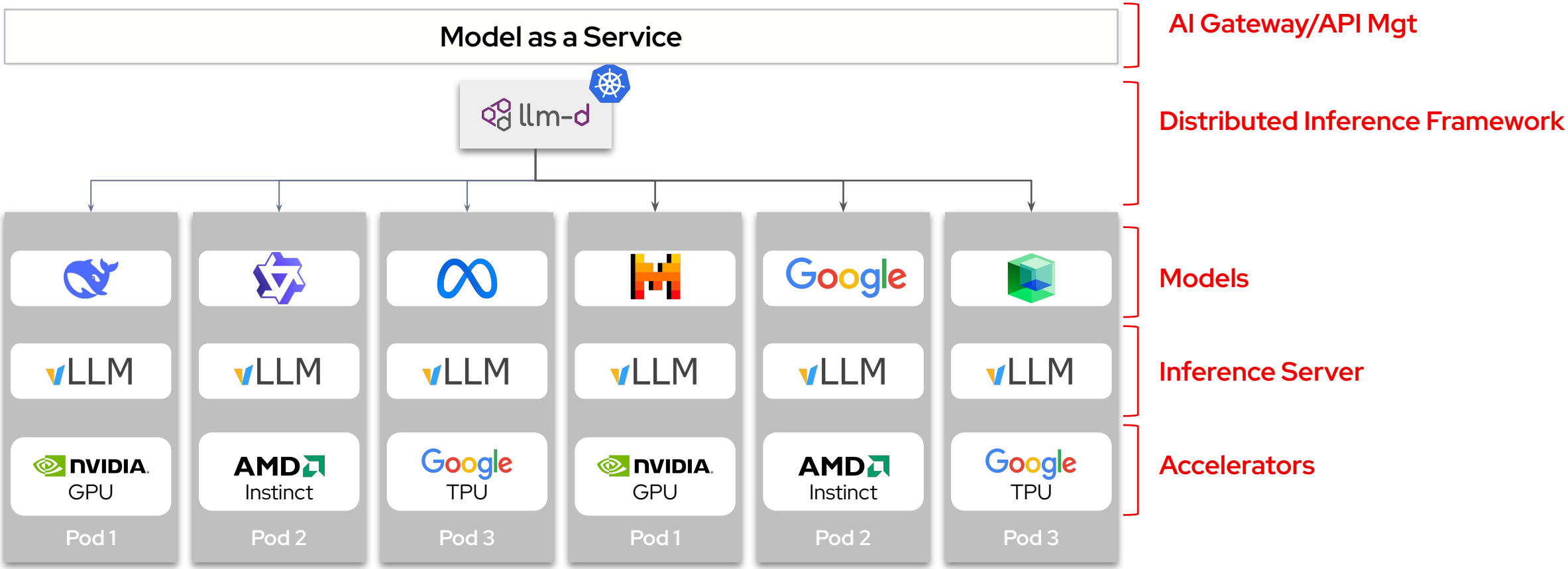
Red Hat

What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
- ~~PoC takeaways~~
- What is next
- Next Steps

Enterprise GenAI inference platform

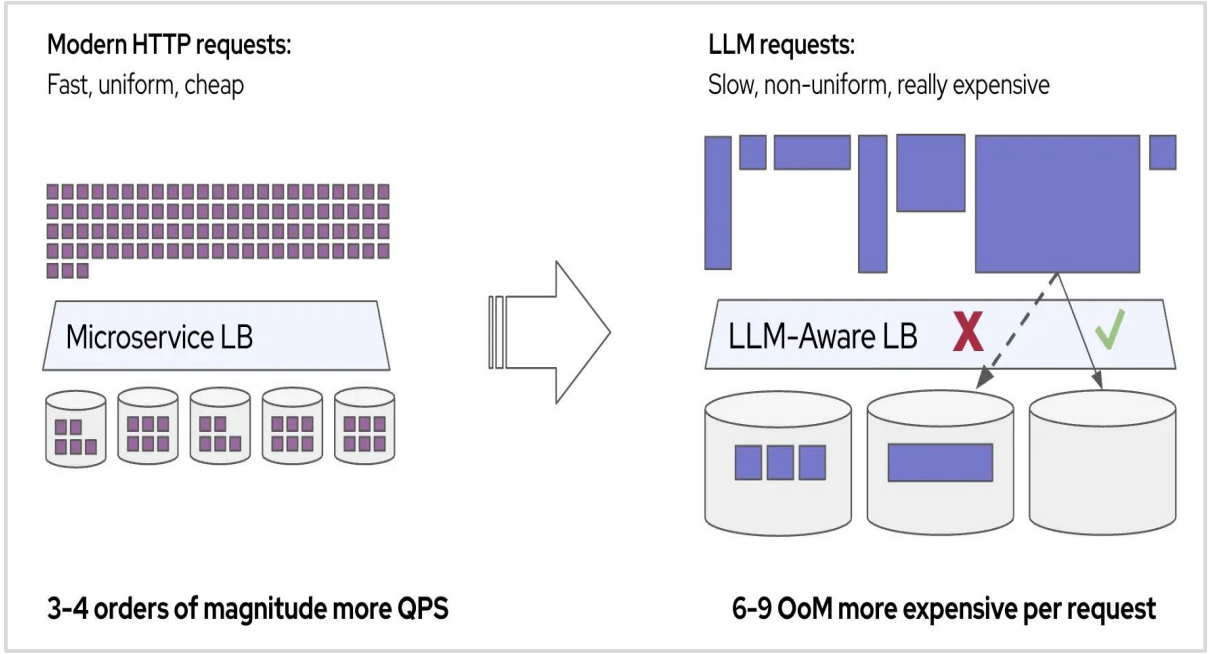
Holistic approach to optimize and operationalize deployment and scaling of open-source LLMs



Red Hat

Distributed Inference is essential, but introduces unique challenges

- LLM inference workloads break under traditional Kubernetes scaling
- **Ensuring SLO** (throughput, TTFT, latency) while **minimizing** resource utilization and operational complexity
- Leveraging and managing **heterogeneous hardware** for better cost-efficiency
- Low Inference efficiency

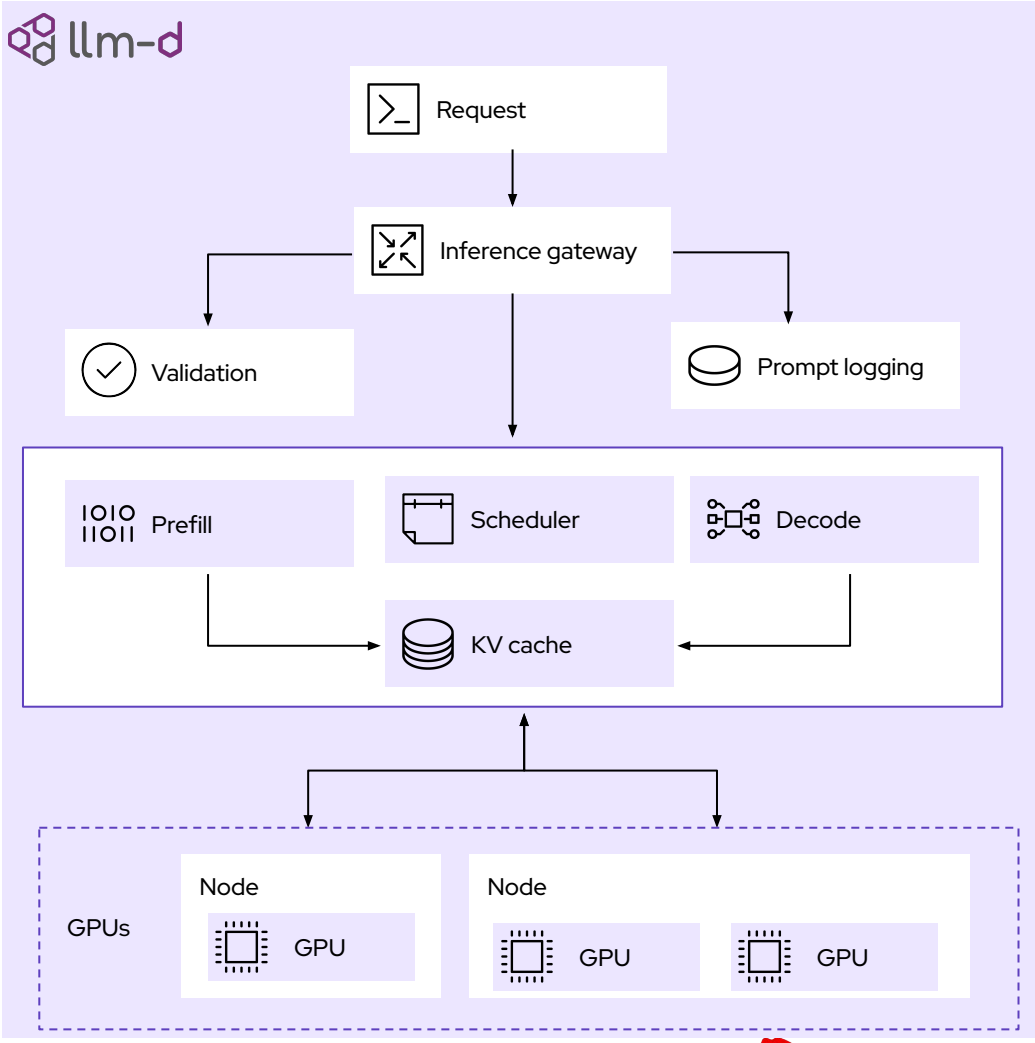


Red Hat

Distributed Inference with llm-d

Maximize GPU utilization and deliver on your SLOs with distributed inference

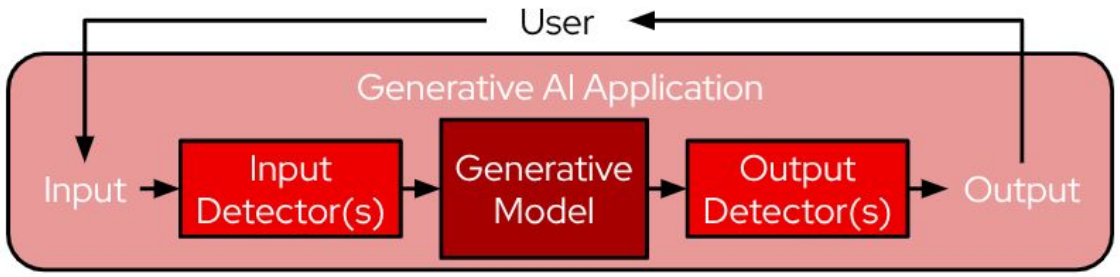
- Joint open source project by Red Hat, Google, NVIDIA, AMD, Hugging Face, and many more
- Kubernetes-Native Architecture for simple deployment and management of GenAI models
- Optimized GenAI Inference to accelerate LLM's and MoE
- Intelligent Resource Utilization to reduce inference costs
- High Performance and Scalability to meet demanding Service Level Objectives (SLOs).
- Supported on Heterogeneous Hardware like NVIDIA and AMD GPUs (and many more to come in the future)



Red Hat

Guardrails for Generative AI in Red Hat AI

- Ensure **secure, compliant, and efficient** AI operations with These key features:
  - **Customizable Input and Output Validators**: Tailor the AI's behavior to meet your business needs
  - **Request-Time Configuration**: Dynamically apply guardrails on a per-request basis
  - **Role-Specific Detection**: Design targeted validation pathways for different user groups



- **Protect customer's Brand**: Prevent mentions of competitors to maintain focus on your products.
- **Minimize Risk**: Restrict contract creation and negotiation to human oversight.
- **Enhance Customer Experience**: Provide role-specific, meaningful interactions for every user group.
- **Boost Efficiency**: Redirect technical queries to systems equipped with actionable insights for mechanics.

PS: signing and securing model artifacts is part of the Model Registry's OCI-compliance storage provided by RHOAI

What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
- ~~PoC takeaways~~
- ~~What is next~~
- Next Steps

Next steps

Some options to discuss

- Probe Red Hat OpenShift AI in ROSA
- Probe kubiya.ai on ROSA
- Probe Red Hat OpenShift AI on-prem with kubiya.ai to on-prem
- Probe ArgoCD in use cases of choice

Red Hat AI

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

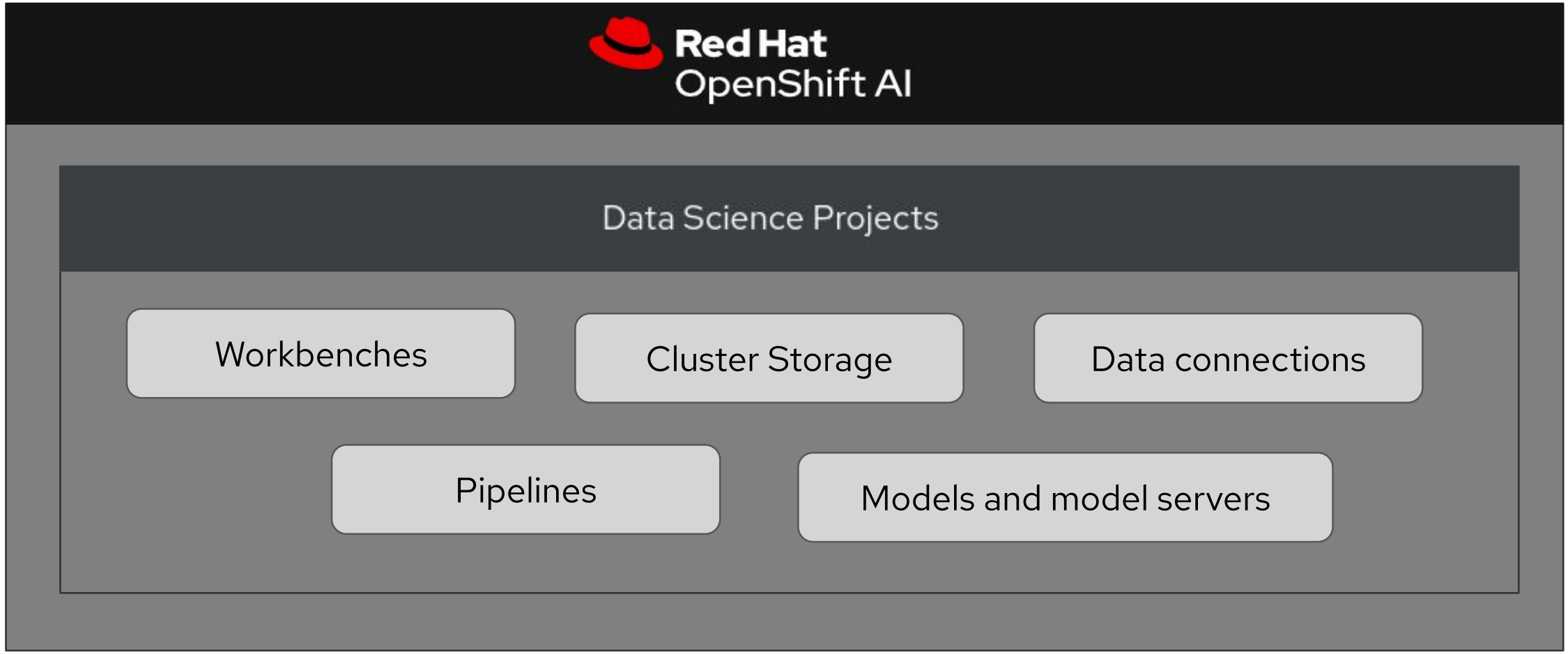

youtube.com/user/RedHatVideos


facebook.com/redhatinc

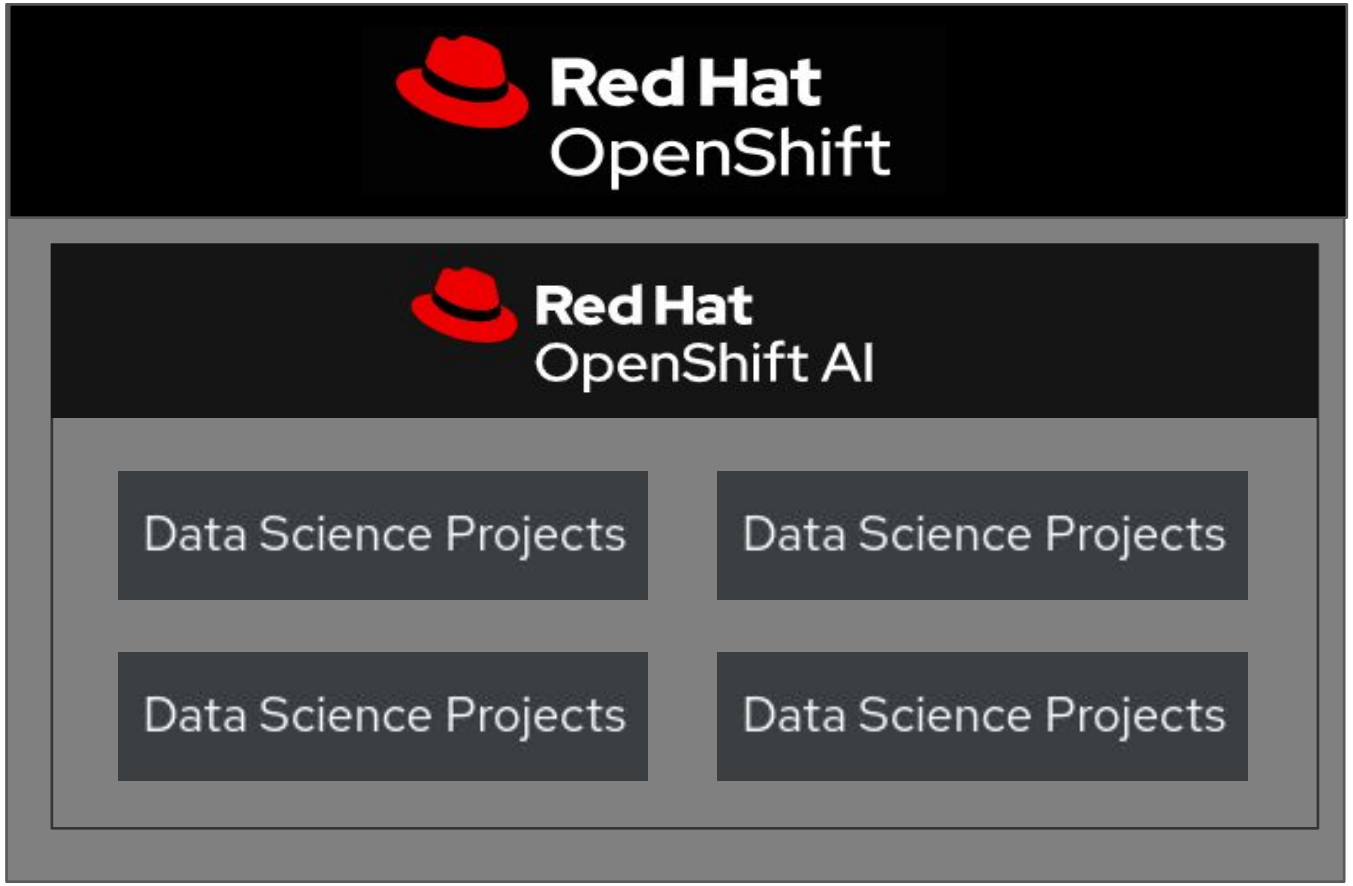

twitter.com/RedHat

Red Hat

Data Science Projects

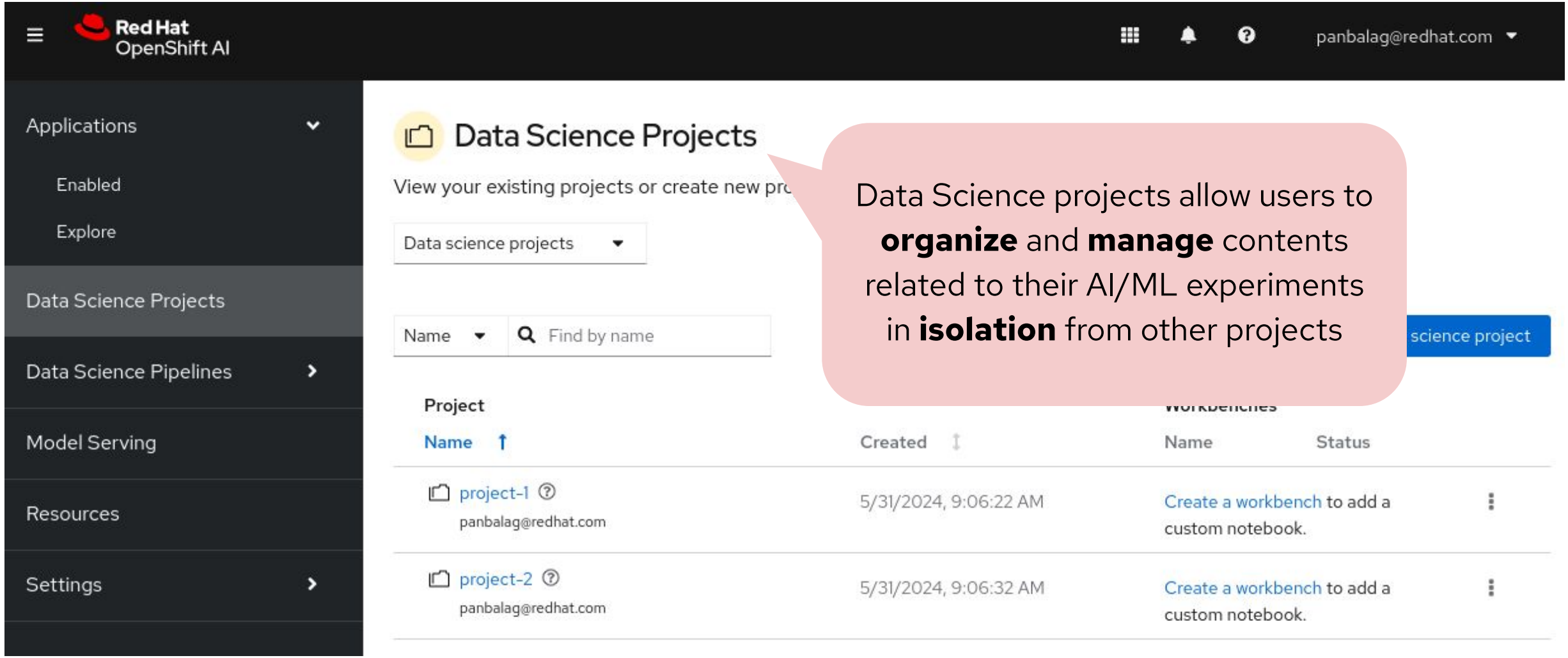


Data Science Projects



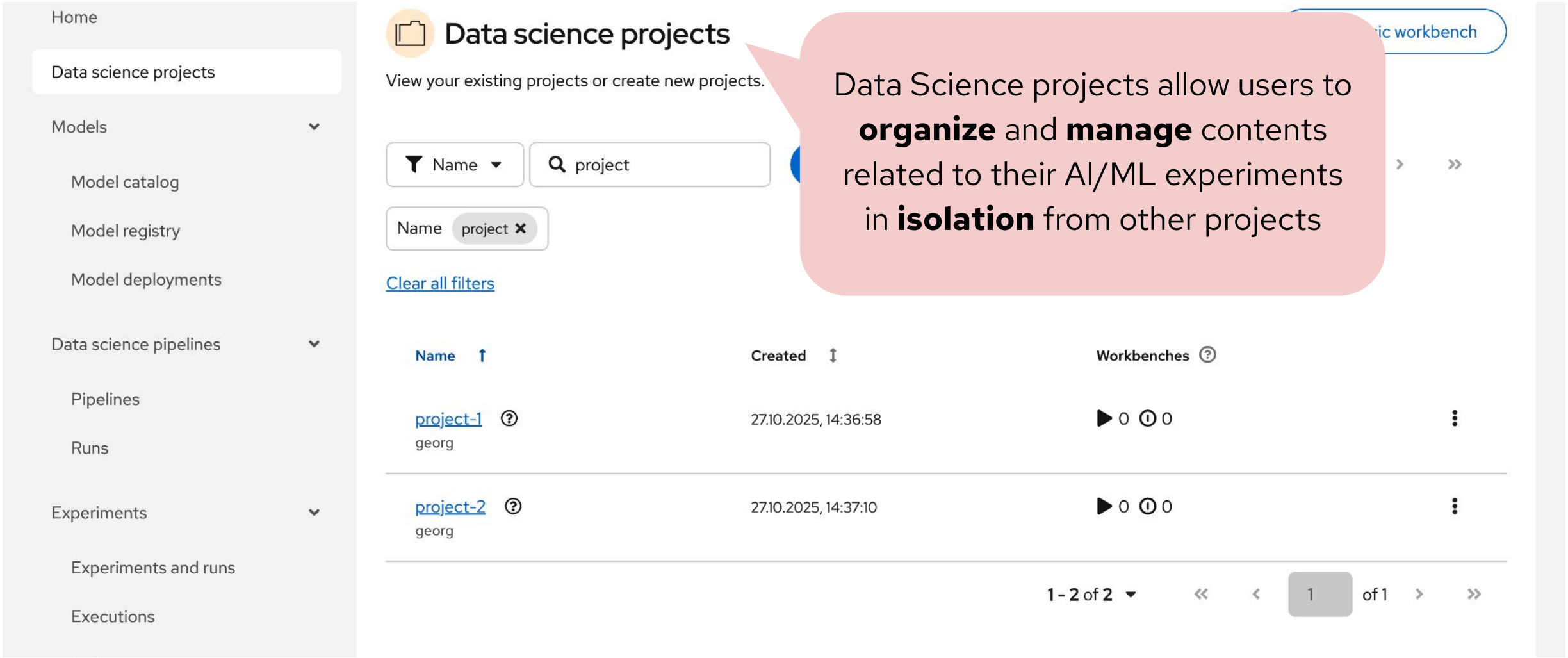
- Multiple data science projects.
- Isolation from other projects
- Created by admins or users
- User/Group access privileges

Data Science Projects



Red Hat

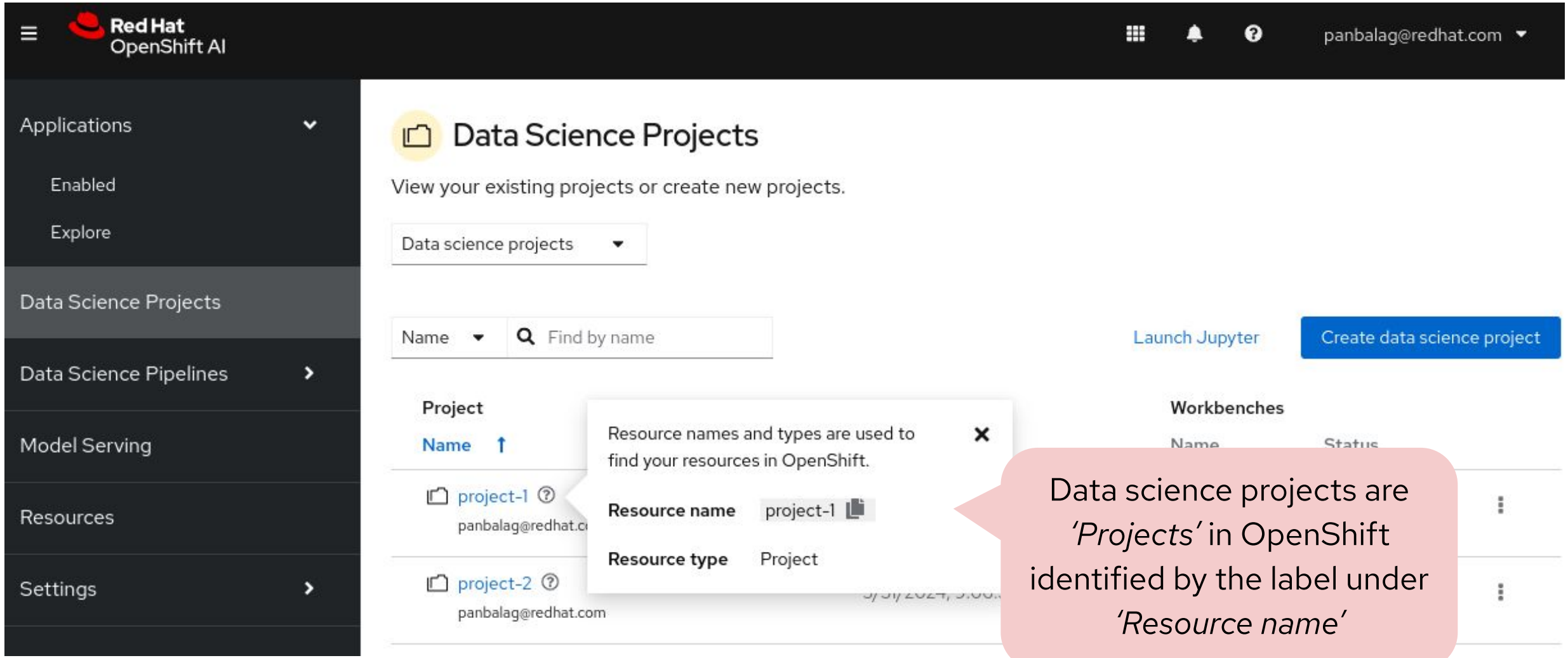
Data Science Projects



Data Science projects allow users to **organize** and **manage** contents related to their AI/ML experiments in **isolation** from other projects

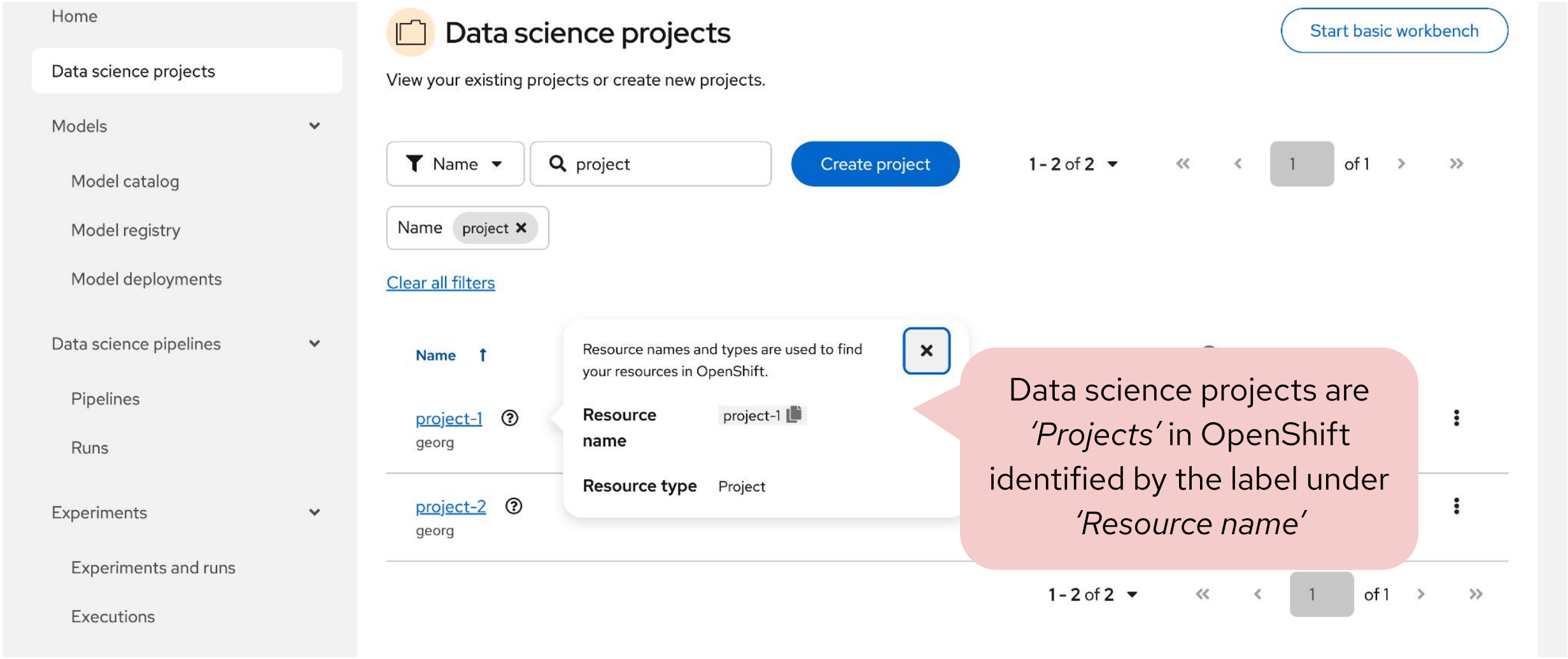
Red Hat

Data Science Projects



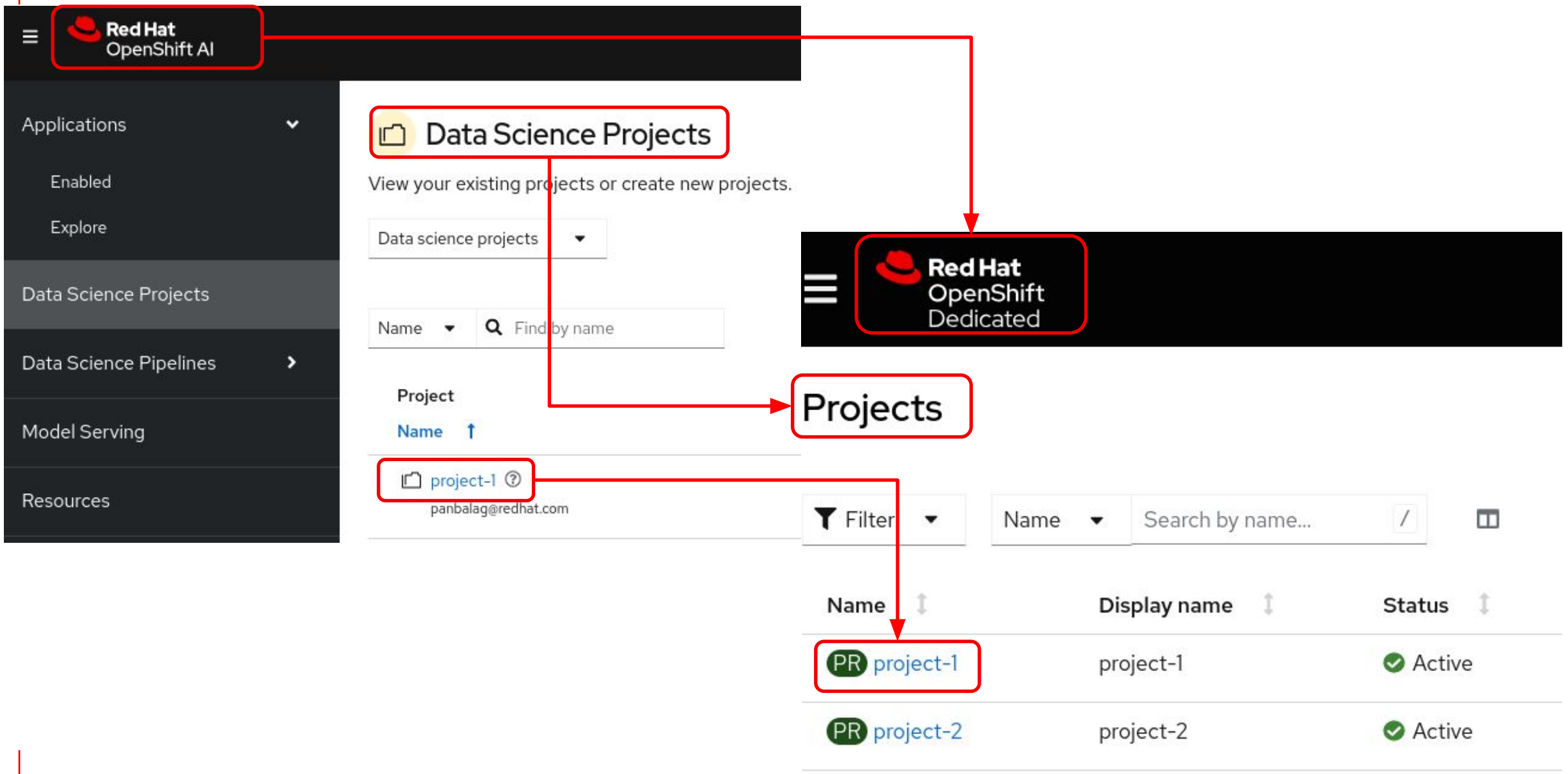
Red Hat

Data Science Projects

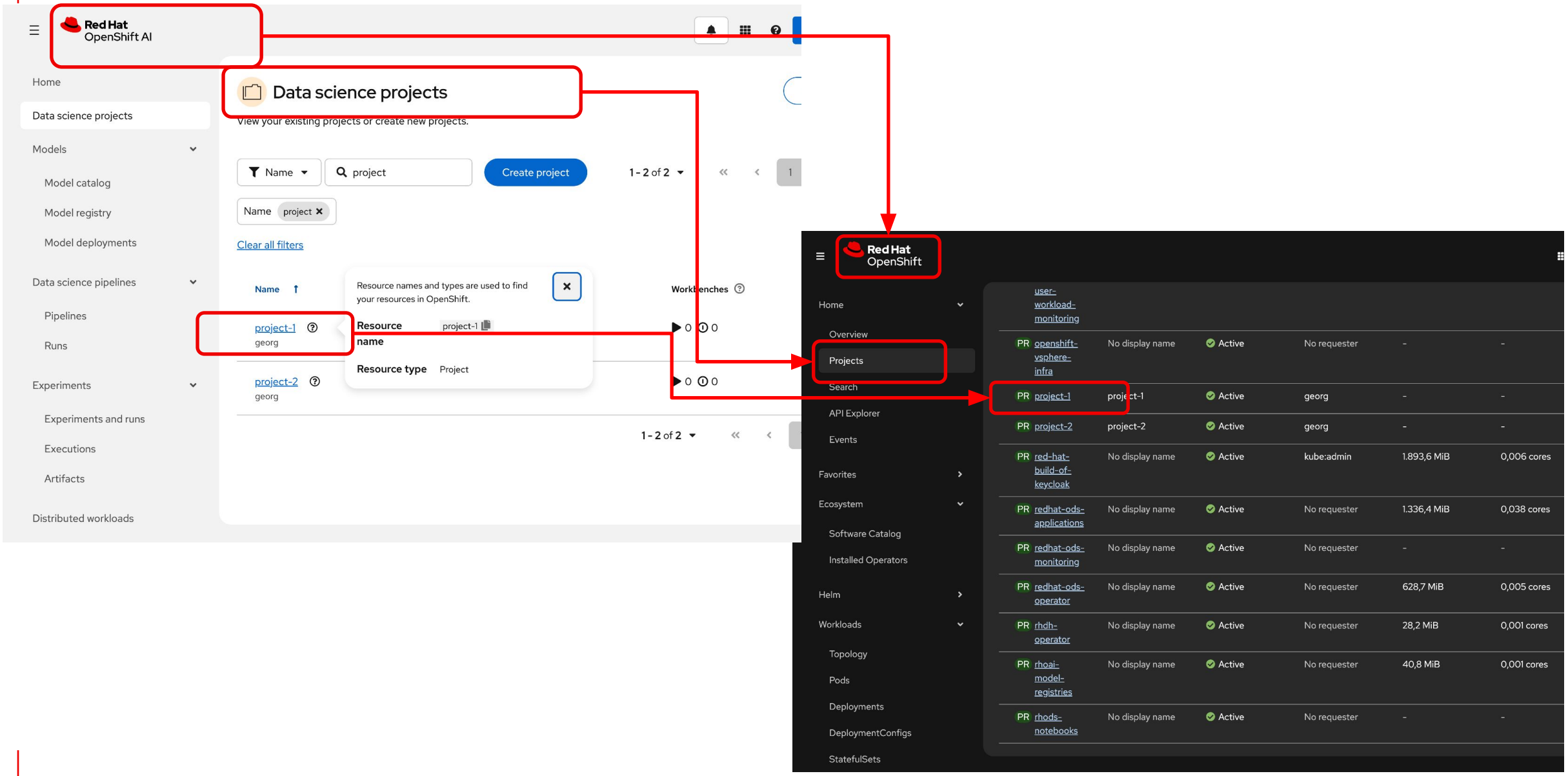


Red Hat

Data Science Projects



Data Science Projects



Data Science Projects

Collaborate within a project

- Users that create a data science project
  - become an admin of that project
  - can give access to a project to any user or group
- Users with access permissions can access all resources in the project, modify them, and create new ones.
- Limiting user level access to data science projects needs to be handled at an OpenShift level at the moment

Collaborate between projects

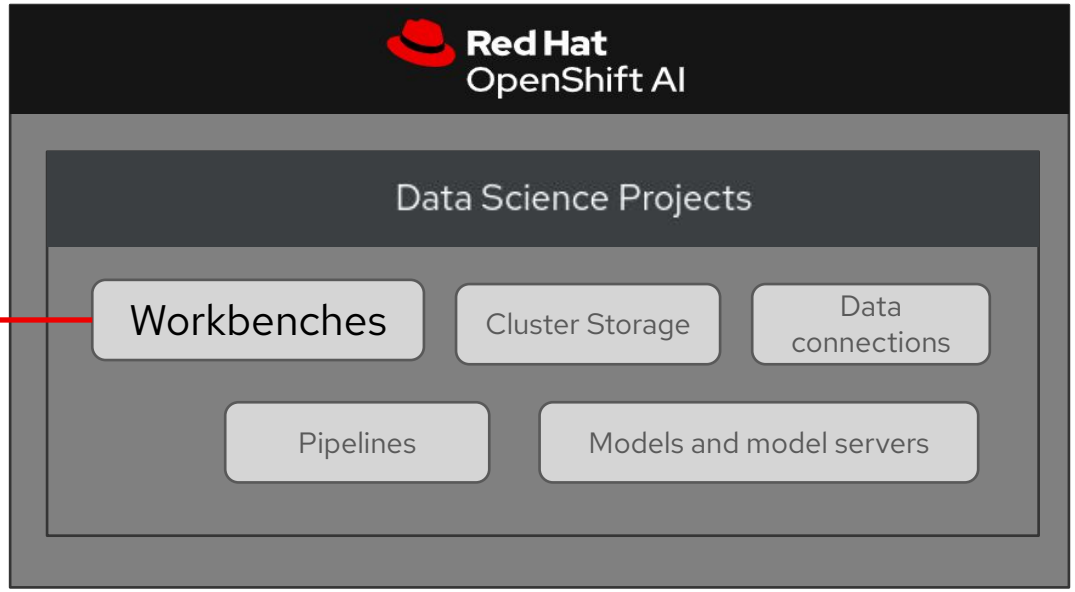
- Due to isolation of data science projects, resources need to be explicitly exposed in order to be shared between projects.
- A good way to do this is to have an external resource which the projects have access to.
  - Examples:
    - A git repository with shared code
    - An object storage with shared artifacts
    - A structured database with shared data

What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
  - ~~What are Data Science Projects~~
  - What are workbenches
  - Model Registry
- PoC takeaways
- Next Steps

Workbenches

- **Notebook Image**
  - **Development environment** in the form of a container image
    - combination of IDE like Jupyter Notebook, VSCode, etc., and choice of AI/ML framework like Tensorflow, PyTorch etc.,
  - **Custom notebook images**.
- **Deployment size**
  - Container size → **# CPUs** & **Memory** size
  - Accelerator → Choice of **Accelerators**/**GPUs**
- **Environment variables**
  - Config Map
  - Secret
- **Cluster Storage**
  - **PVC** connected to the development environment to store code & related artifacts.
- **Data connections**
  - **Object store** for hosting models as well as storing pipeline artifacts.



Red Hat

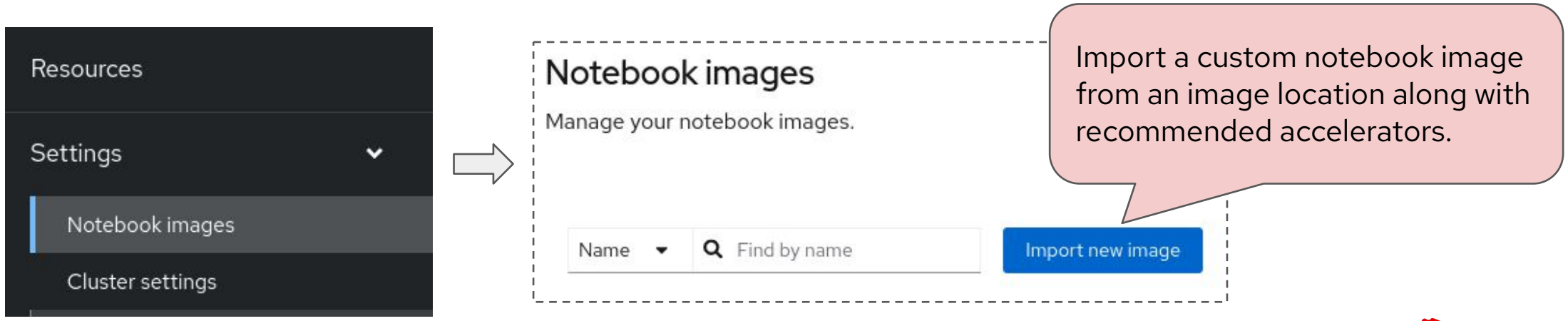
Default Notebook Images

Image	Description
CUDA	For compute-intensive data science models that require GPU support, the Compute Unified Device Architecture (CUDA) notebook image provides **access to the NVIDIA CUDA Toolkit** with GPU-accelerated libraries and optimization tools.
Standard Data Science	Contains **commonly used libraries** to assist you in developing your machine learning models.
TensorFlow	**TensorFlow**, a popular open source machine learning platform. TensorFlow provides advanced libraries, data visualization features that allows users to build, monitor and track models.
PyTorch	**PyTorch** is another open source machine learning library optimized for deep learning like computer vision or natural language processing models.
Minimal Python	A **minimal environment with JupyterLab** for basic exploration.
Trusty AI	For AI/ML work with **model explainability, tracing, and accountability**, & runtime monitoring
Habana AI	For high-performance optimization of deep learning training workloads and maximize training throughput and efficiency with **Habana Gaudi devices**.
code-server (Technology Preview)	Provides you with a **VSCode** environment, allowing you to customize the environment through **extensions**.

Default notebook images

Customizing Workbenches

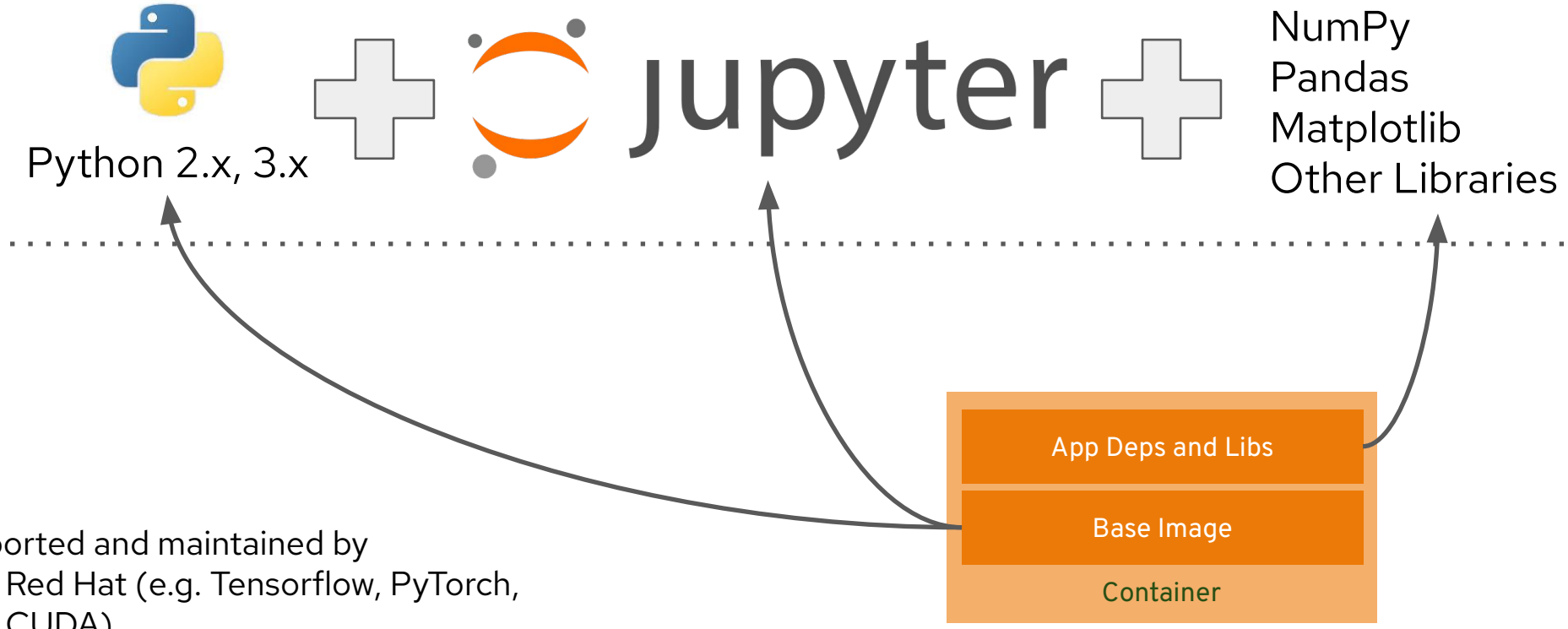
- To customize the workbench you can either:
  - Install dependencies on top of a workbench
  - Use a custom notebook image
- You can use package managers such as pip to add/remove dependencies in an existing workbench
  - Dependencies installed within the workbench are by default not saved to the persistent storage, this is by choice as restarting the workbench is an easy way to reset the environment if something caused an issue with the dependencies
- You can create and use custom notebook images to completely customize the environment



Customizing Workbenches

Base Notebook Images

Reproducible and shareable environments for building, training and serving



Supported and maintained by

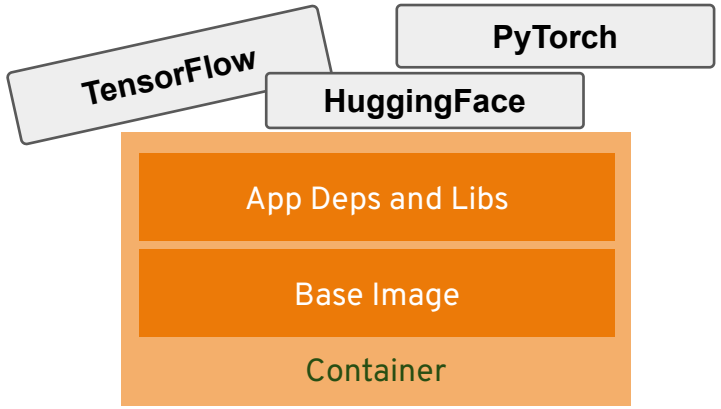
- Red Hat (e.g. Tensorflow, PyTorch, CUDA)
- partner (Anaconda, Intel)
- you (custom notebooks)

Red Hat

Customizing Workbenches

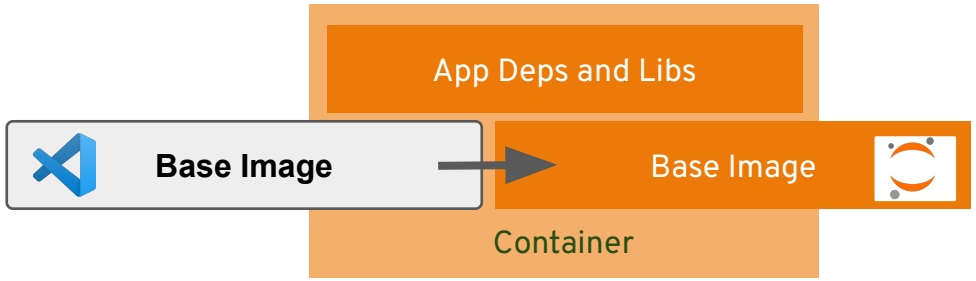
Customizing the workbench

Adding packages on top of a good image



Just remember that they are removed when restarting the workbench*

Creating your own custom image with all dependencies you need



You can now version and maintain it according to your preferences

* This is on purpose so that you can un-mess-up your environment easily if you get into dependency issues.

Red Hat

What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
  - ~~What are Data Science Projects~~
  - ~~What are workbenches~~
  - Model Registry
- PoC takeaways
- Next Steps

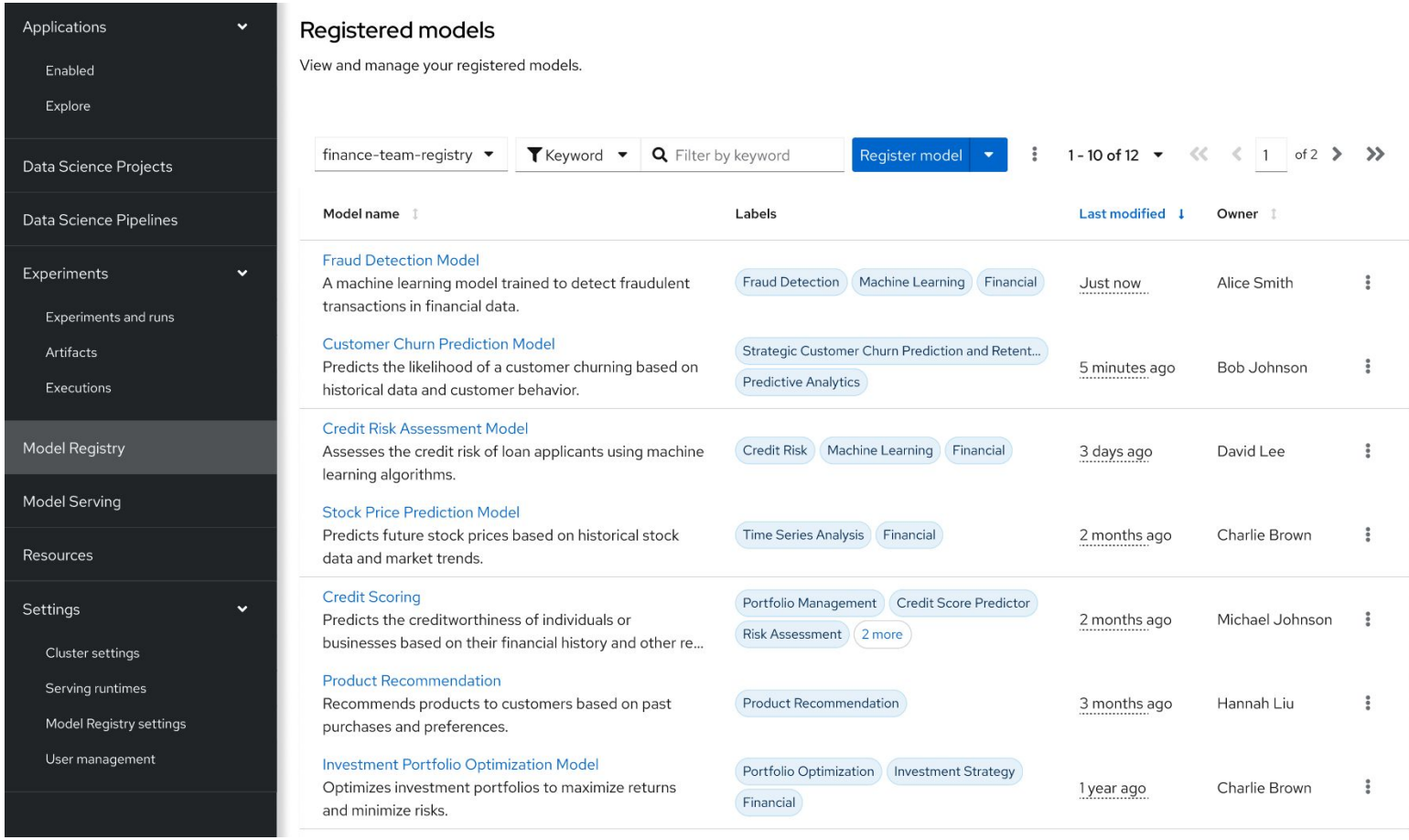
Model Registry Preview

How will it work?

- Can register a model along with properties such as name, tags, description, model type, dataset etc.
- Can edit the details of the model.
- Uses S3 as a default backend but can link to models in other storages as well, for example separate S3 or PVC.
- Can store artifacts such as generated files, sample data, text files, etc.

Model Registry Preview

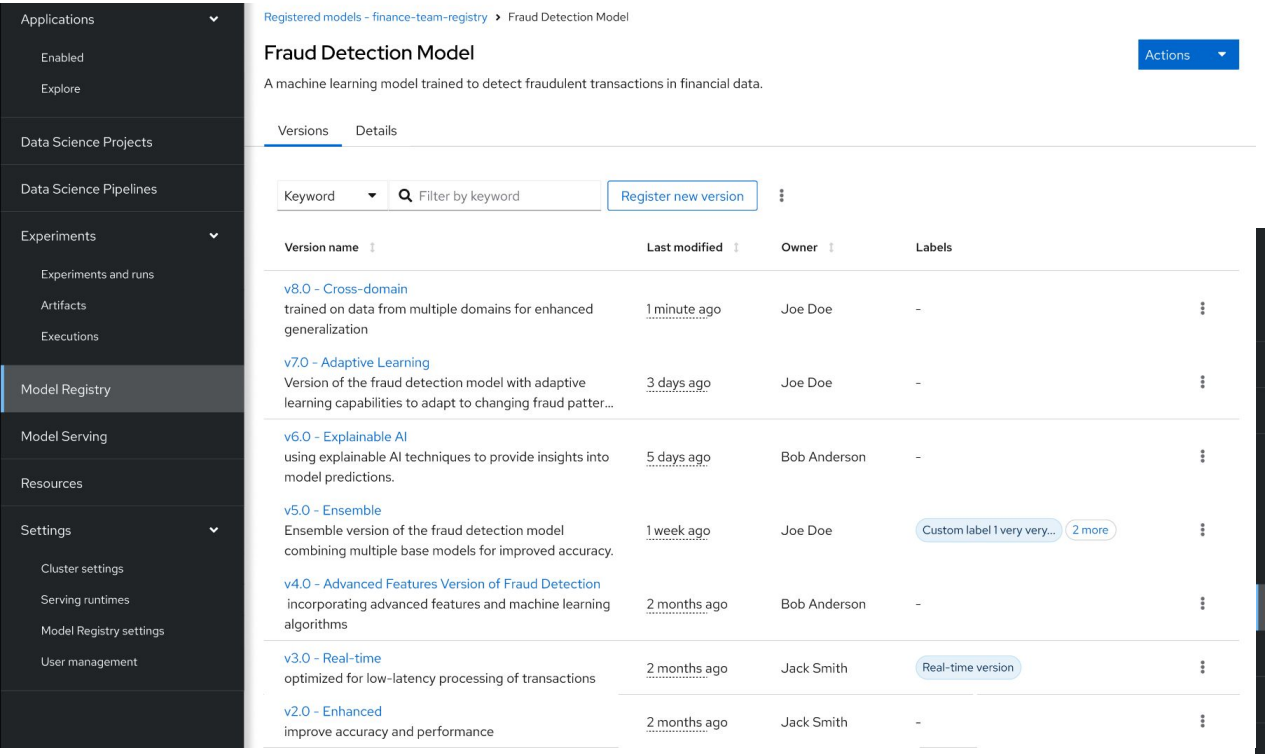
List models

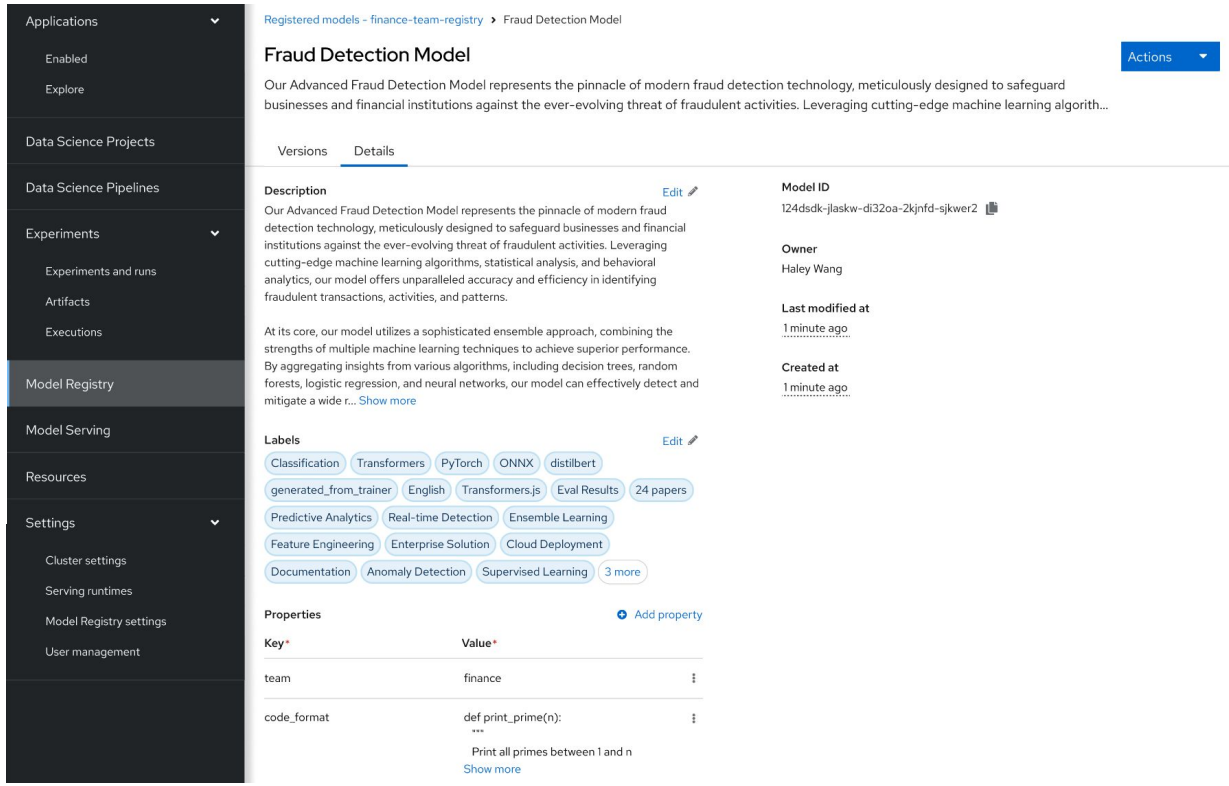


Red Hat

Model Registry Preview

Model details and versions

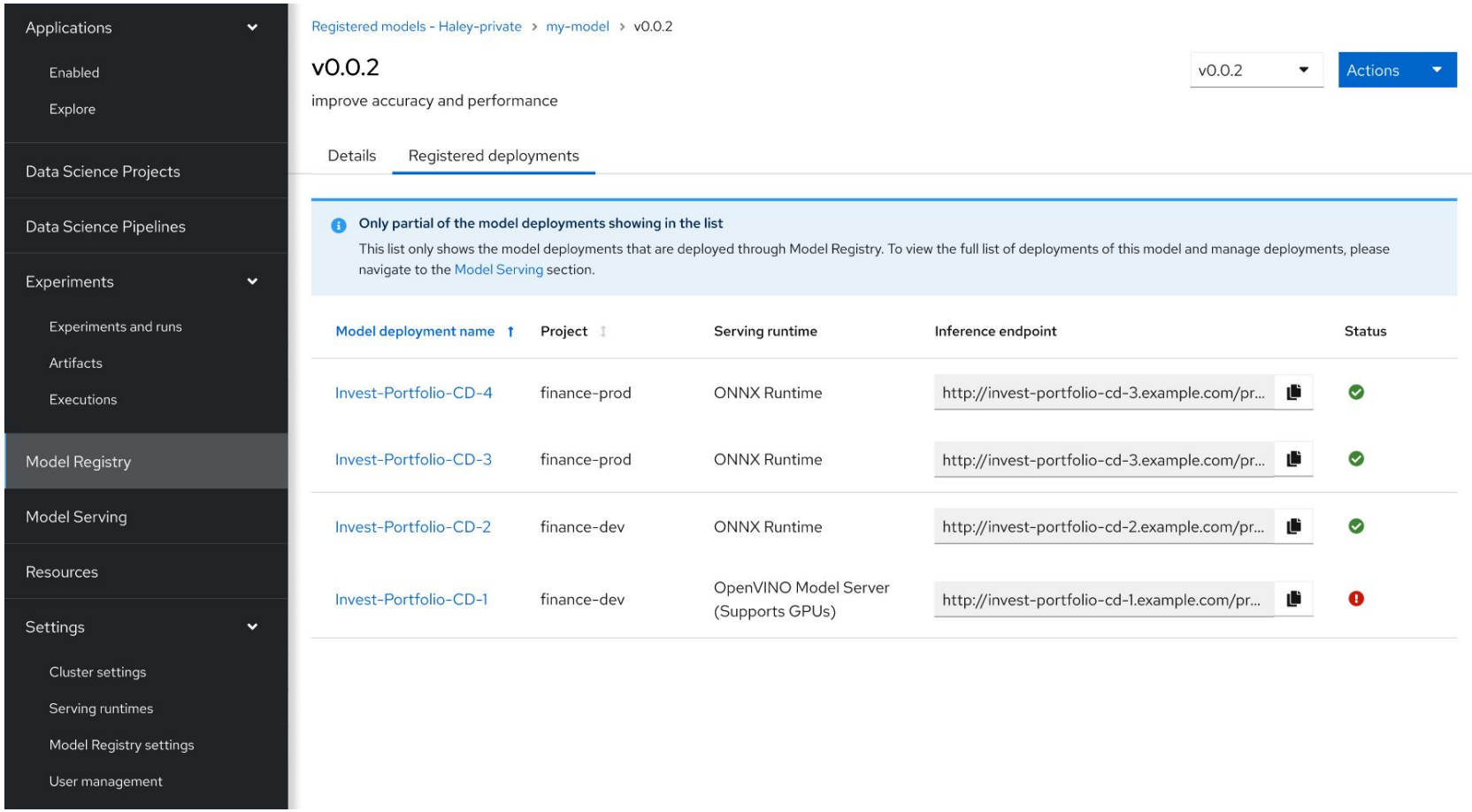




Red Hat

Model Registry Preview

Deploy and keep track

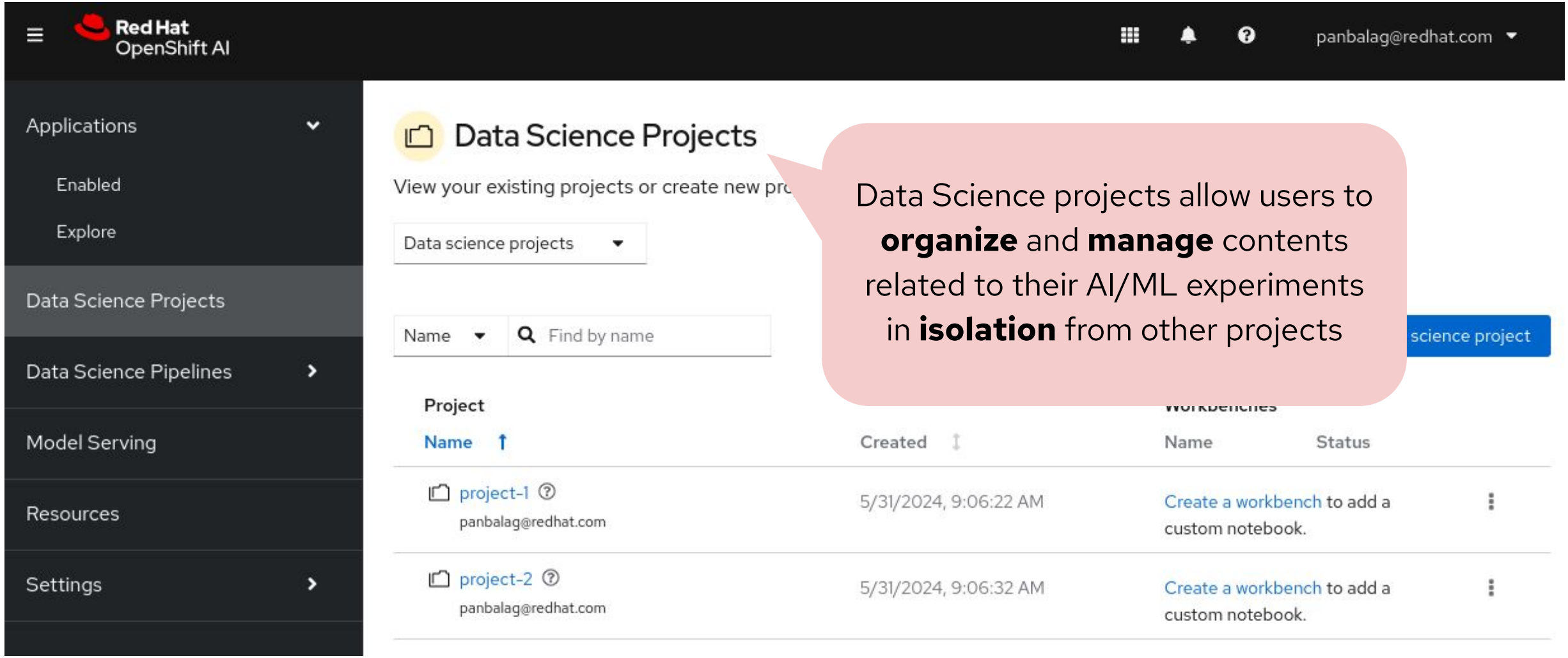


Red Hat

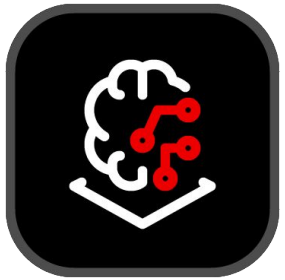
What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
  - ~~What are Data Science Projects~~
  - ~~What are workbenches~~
  - ~~Model Registry~~
- PoC takeaways
- Next Steps

Data Science Projects



Red Hat



Objective #1]

Provide a high level description of finding #1

[Objective #2]

Provide a high level description of finding #2

[Objective #3]

Provide a high level description of finding #2

These were our desired outcomes from this PoC.

[insert desired outcome]

Description

[insert desired outcome]

Description

[insert desired outcome]

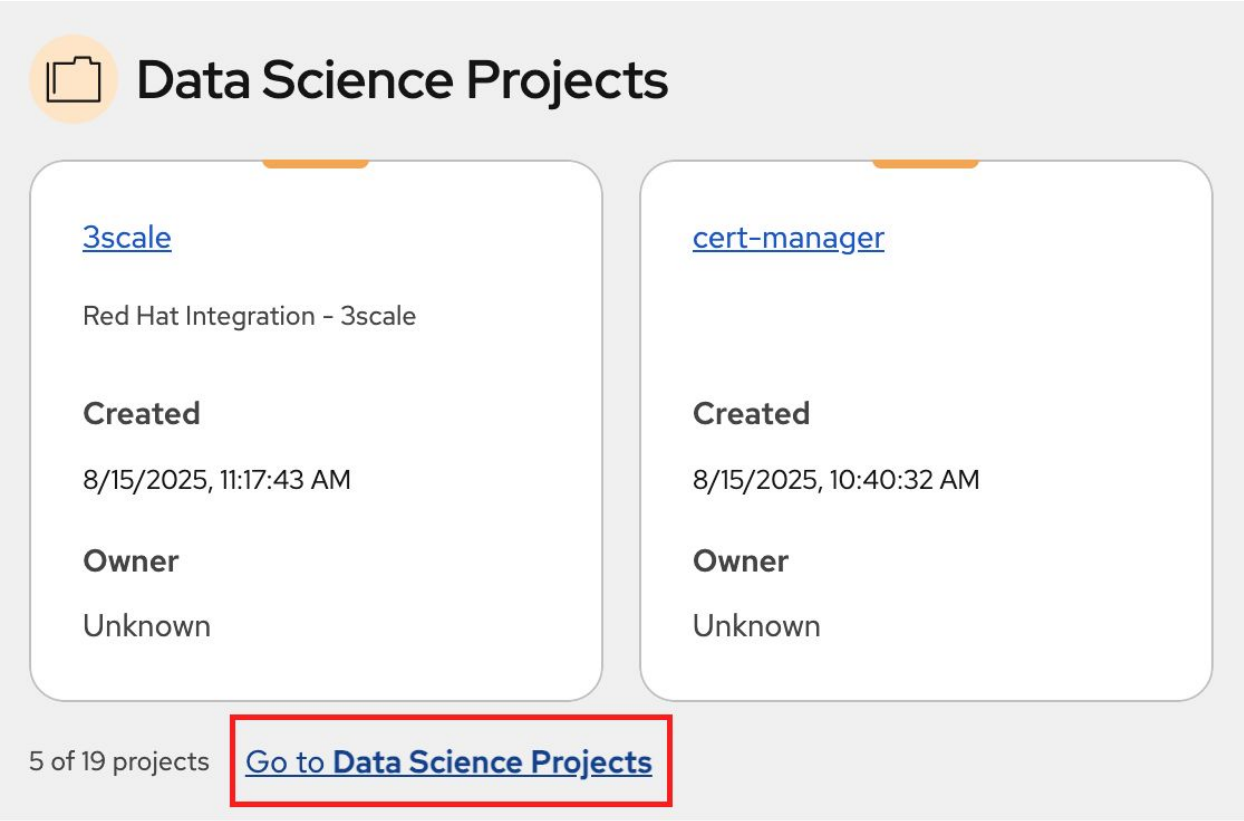
Description

What we'll discuss today

- ~~Motivational Speech~~
- ~~Red Hat AI Inference Server Overview~~
  - ~~What are Data Science Projects~~
  - ~~What are workbenches~~
  - ~~Model Registry~~
- ~~PoC takeaways~~
- Next Steps

Data Science projects

- Your “toolbox”
-



Red Hat

- Workbenches: Where you can create and manage various development environments like JupyterLab, VSCode, or other custom Workbenches. It provides a user-friendly interface for data scientists to work with notebooks, libraries, and datasets.
- Pipelines: You may use pipelines to automate the process of processing data or training and deploying machine learning models.
- Models: Where you can manage and deploy machine learning models. You can create, update, and delete models, as well as monitor their performance and usage.
- Cluster storage: Here you can manage the storage resources used by your models and workbenches. You can create, update, and delete storage resources, as well as monitor their usage.
- Connections: This is where you can manage the connections between your workbenches or model runtimes and other services, such as storage (S3), databases or APIs. You can create, update, and delete connections, as well as see which environment is using them.
- Permissions: This is where you can manage the permissions for project. You can create, update, and delete permissions, as well as see which users or groups have access to which resources.