Microsoft

Microsoft Partner Project Ready

Implement with Impact

Modern Data Platform with Azure Databricks

<Speaker name or subtitle>

<Date>

Day 3 of 3

Technofocus

Course Plan and Learning Objectives

Day 1

Module 1 - Introduction to Azure Databricks
- Azure Databricks: A Data Intelligent Platform
- Why Azure Databricks
- Decision guide: Azure Databricks vs. Microsoft Fabric

Module 2 - Migration to Azure Databricks
- Microsoft Cloud Adoption Framework for Azure
- Migration strategies
- Data landing zones
- Migration scenarios

Interactive Simulated Lab Experience
- End-to-End Streaming Pipeline with Lakeflow Declarative Pipelines in Azure Databricks

Day 2

Module 3 - Integration with Azure
- Seamless integration with Microsoft Azure services
- Connect to Azure Data Lake Storage (ADLS) Gen2 and Blob Storage
- Leverage Azure Databricks for Azure Cosmos DB Operations
- Secret management with Azure Key Vault
- Connect Azure Databricks to Azure Event Hubs

Module 4 - Integration with Microsoft Fabric and Power BI
- Data Intelligence with Azure Databricks and Microsoft Fabric
- Connect Power BI to Azure Databricks
- Integration with Azure Data Factory
- Mirroring Azure Databricks Unity Catalog

Interactive Simulated Lab Experience
- Setup and use Unity Catalog for Data Management in Azure Databricks
- Real-Time Streaming with Azure Databricks and Azure Event Hubs

Day 3

Module 5 - Integration with Azure AI Foundry
- Azure Databricks connector in Azure AI Foundry
- Mosaic AI and machine learning on Azure Databricks
- Query Generative AI model serving endpoints
- Databricks Assistant, AI/BI Genie and AI Functions on Azure Databricks
- Chat with LLMs and prototype GenAI apps using AI Playground
- Build and optimize agents on your data with Agent Bricks

Module 6 - Security and Governance
- Integrate Azure Databricks with Microsoft Purview
- Integration of Azure Databricks Unity Catalog with Microsoft Purview

Module 7 - Well-architected for Azure Databricks
- Lakehouse implementation: Principles and best practices
- Azure Databricks well-architected framework

Interactive Simulated Lab Experience
- Responsible AI with Large Language Models using Azure Databricks and Azure OpenAI
- Connect to and manage Azure Databricks in Microsoft Purview

05
Integration with Azure AI Foundry

Azure Databricks connector in Azure AI Foundry

Azure OpenAI

OpenAI

- gpt-4
- gpt-4.1
- gpt-4.1-nano
- gpt-4.1-mini
- gpt-4o

*All connections are zero-copy

Bring models to data

Lakeflow Ingest, ETL, Streaming

Databricks SQL Data Warehousing

AI/BI Business Intelligence

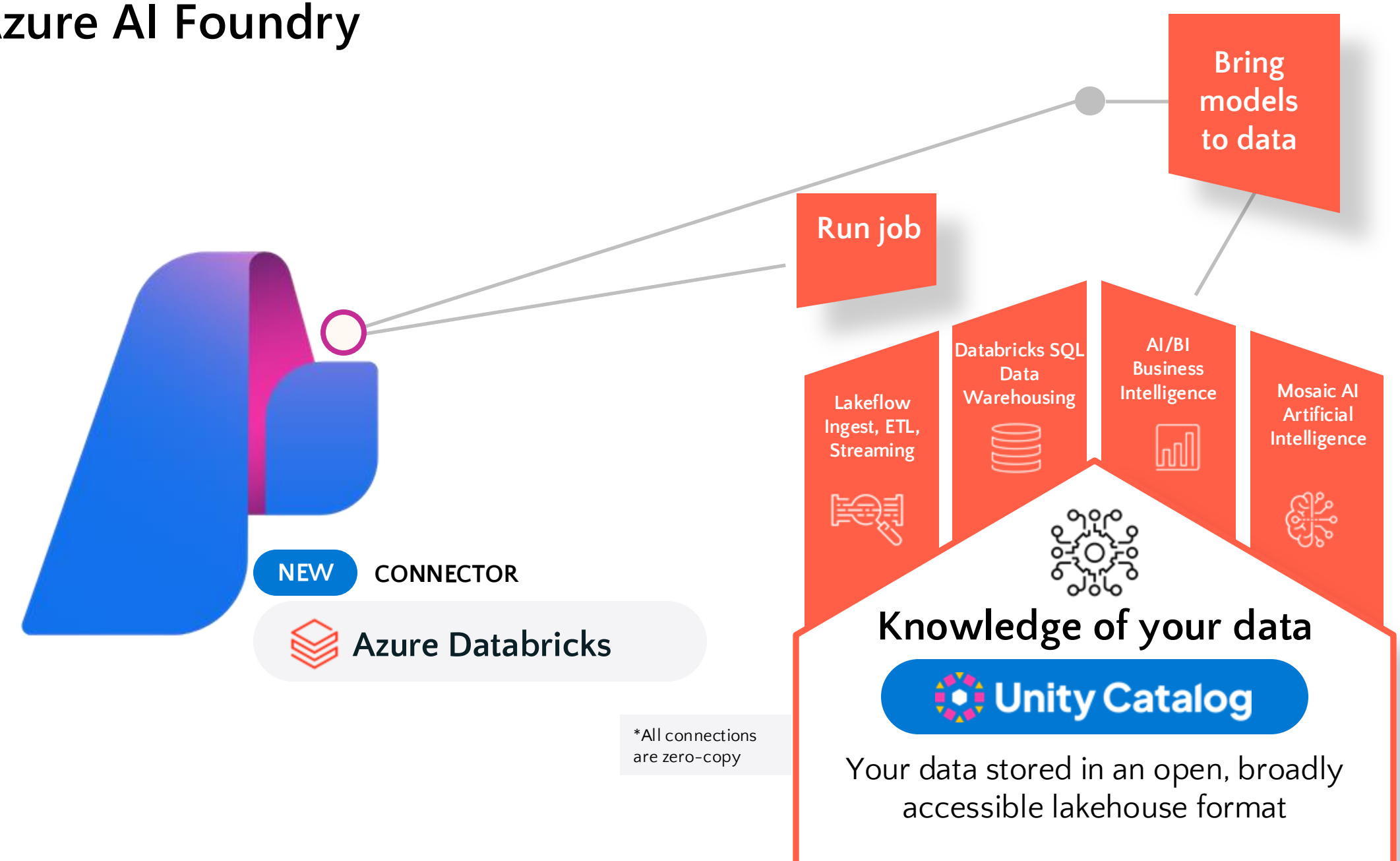
Mosaic AI Artificial Intelligence

Knowledge of your data

Unity Catalog

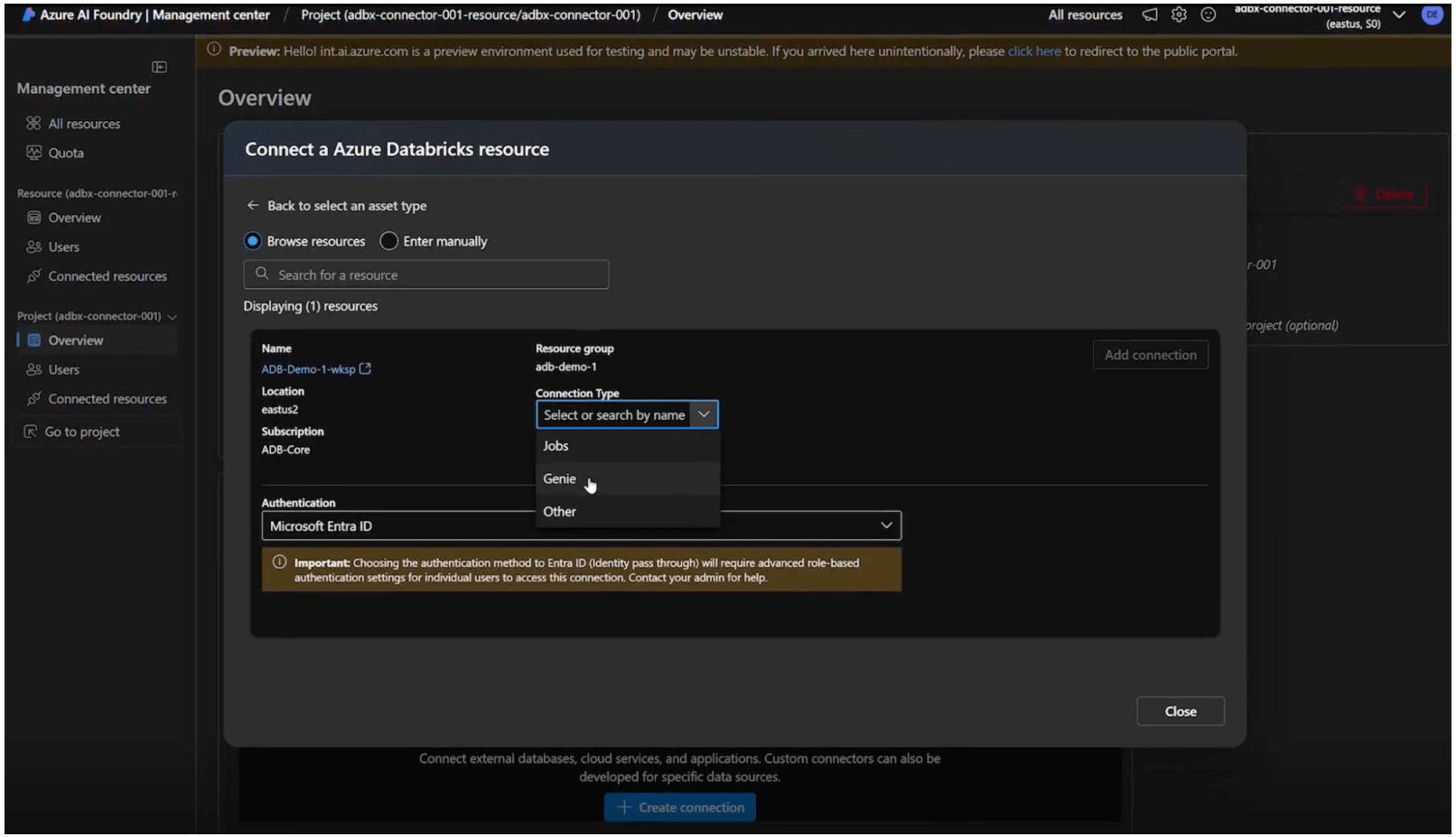
Your data stored in an open, broadly accessible lakehouse format

Azure AI Foundry



Azure Databricks connector in Azure AI Foundry

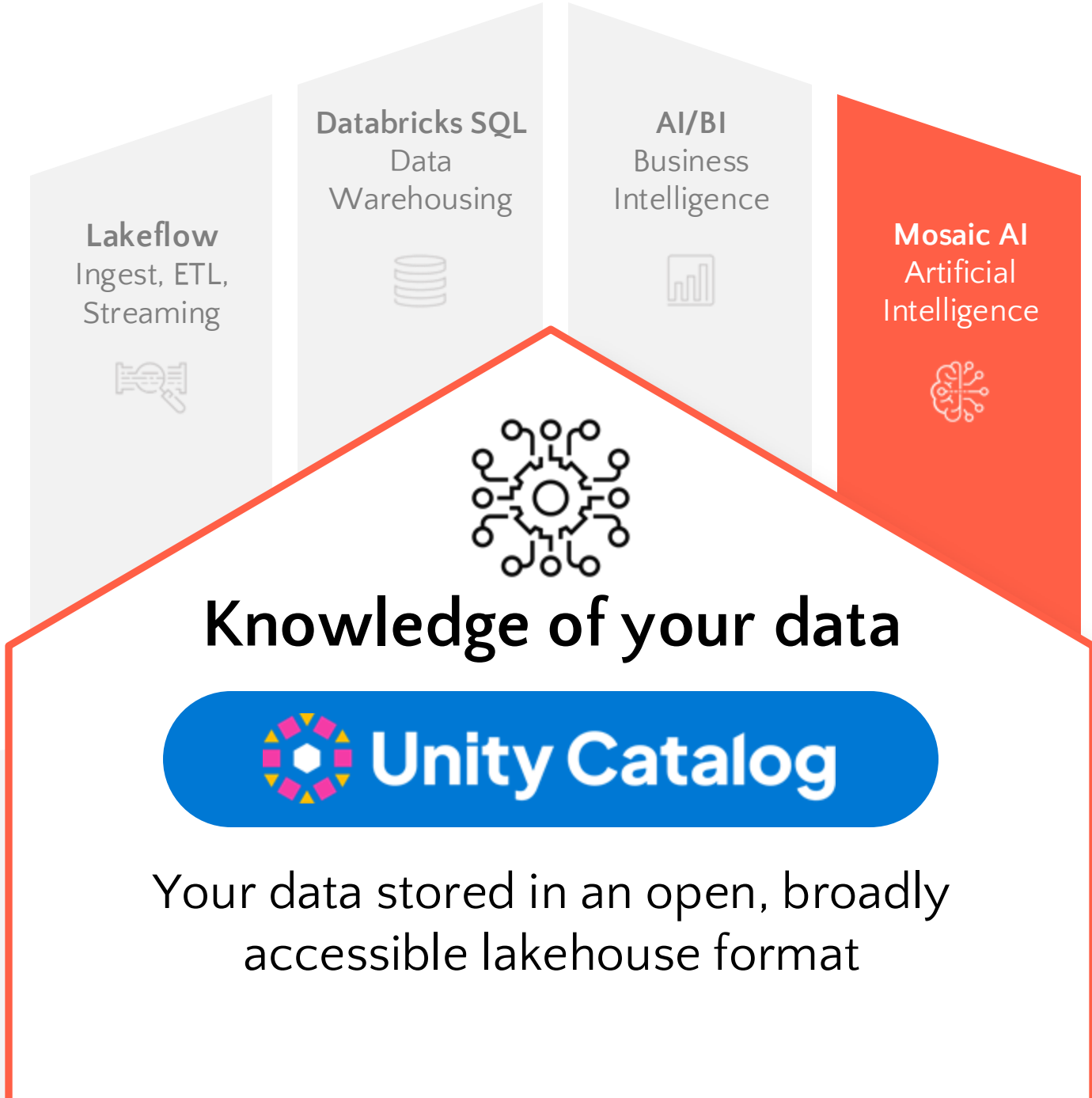
- Build enterprise-grade AI agents that reason over real-time Azure Databricks data while being governed by Unity Catalog
- These agents will also be enriched by the responsible AI capabilities of Azure AI Foundry



Mosaic AI and machine learning on Azure Databricks

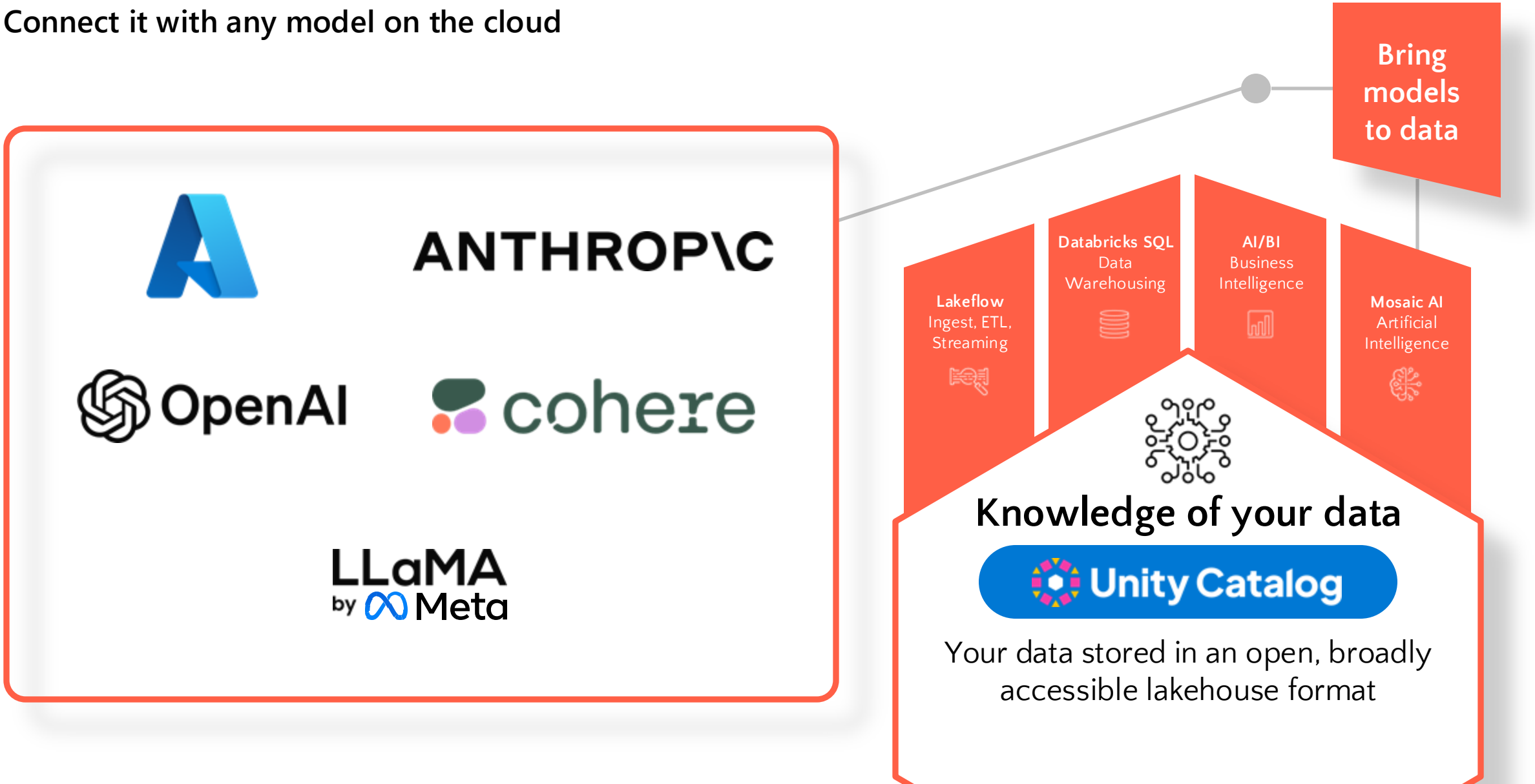
Mosaic AI

Create domain-specific agentic applications

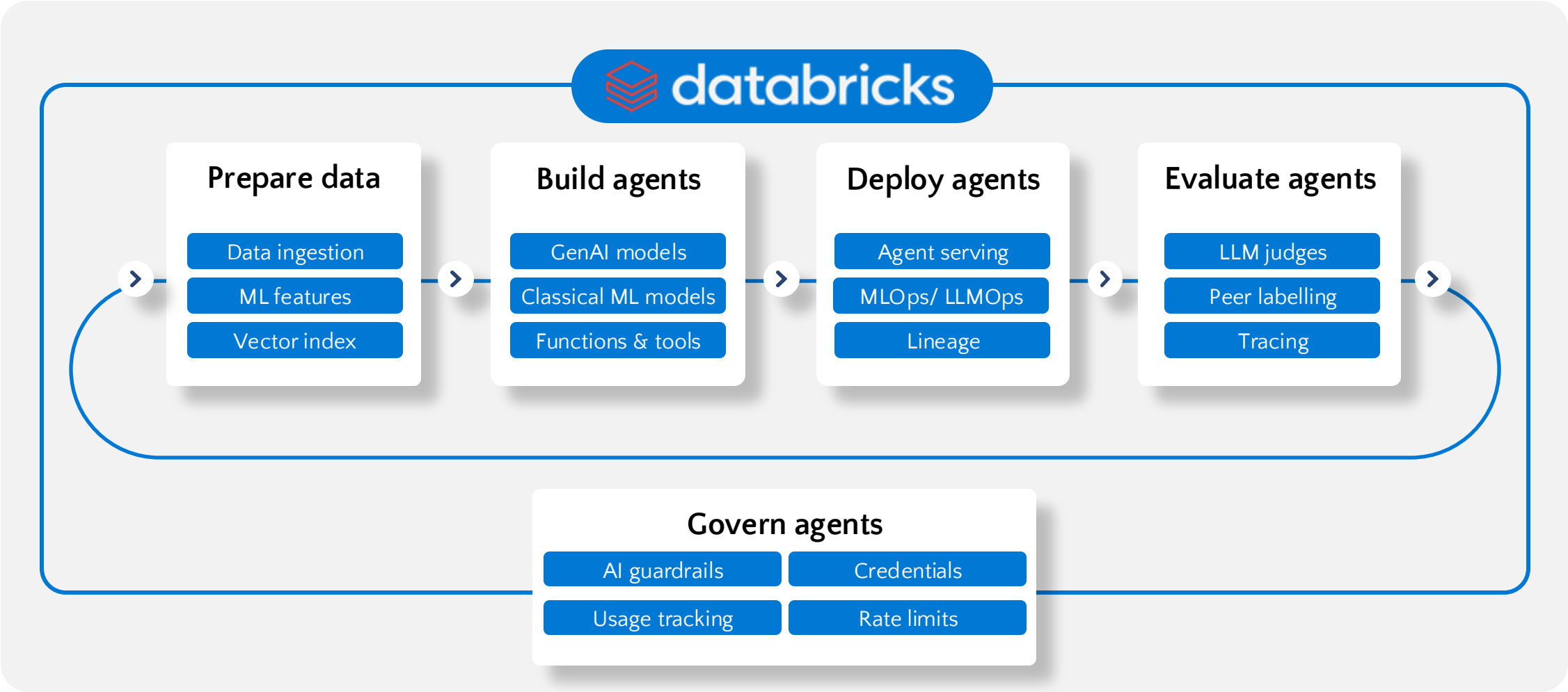


RAG, finetune, agentize your data

Connect it with any model on the cloud

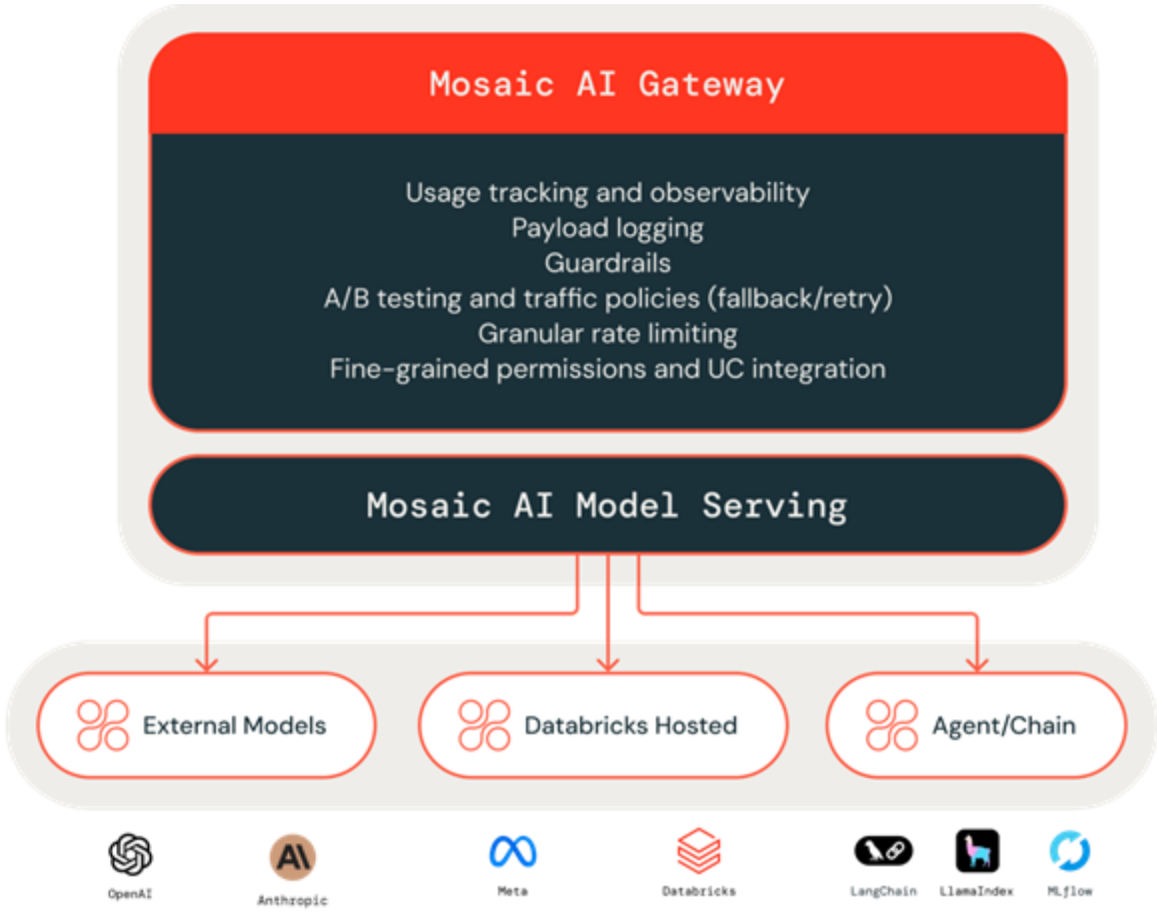


Unified platform to build AI apps & agents



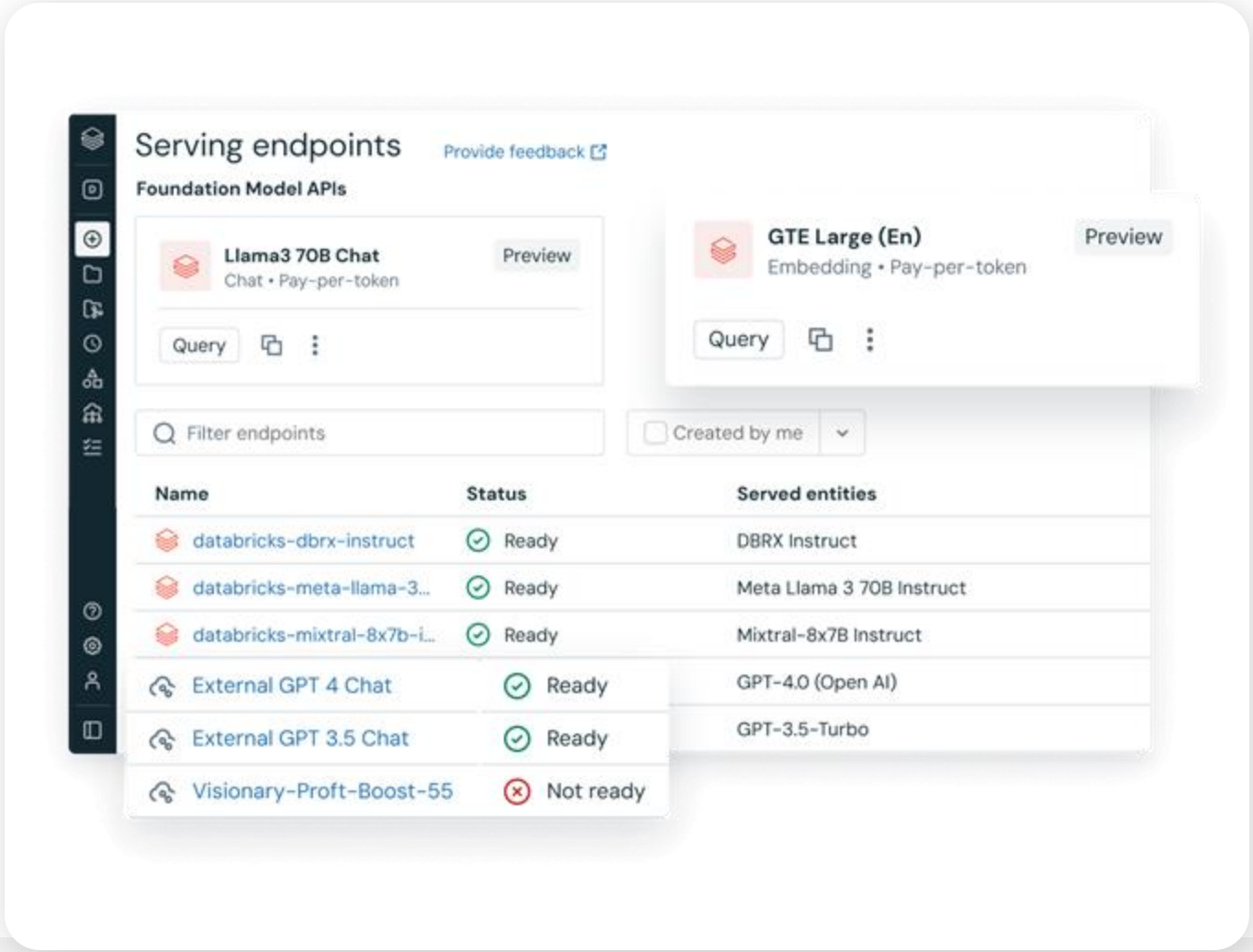
AI Gateway & Foundation Models

- Manage and govern all GenAI models, agents, and chains across the enterprise
- Support for Azure OpenAI, Anthropic
- Permission and rate limiting to control who has access and how much access



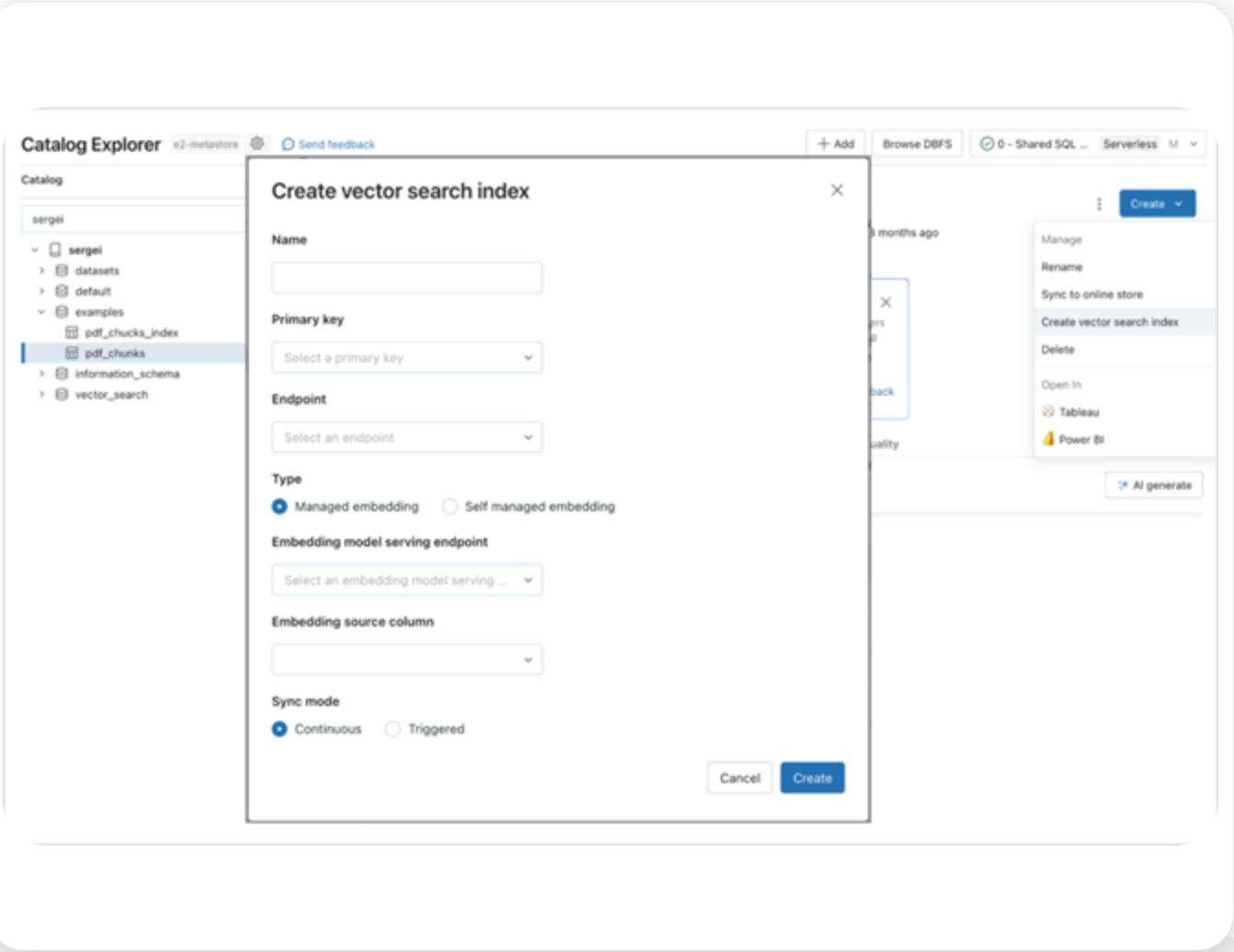
Batch and real-time AI serving

- Single solution to deploy, govern, and manage AI models and agents for batch and real-time workloads
- Deploy both custom and GenAI models as REST APIs
- Manage models hosted by other providers like Azure OpenAI (ChatGPT) and AWS (Anthropic)

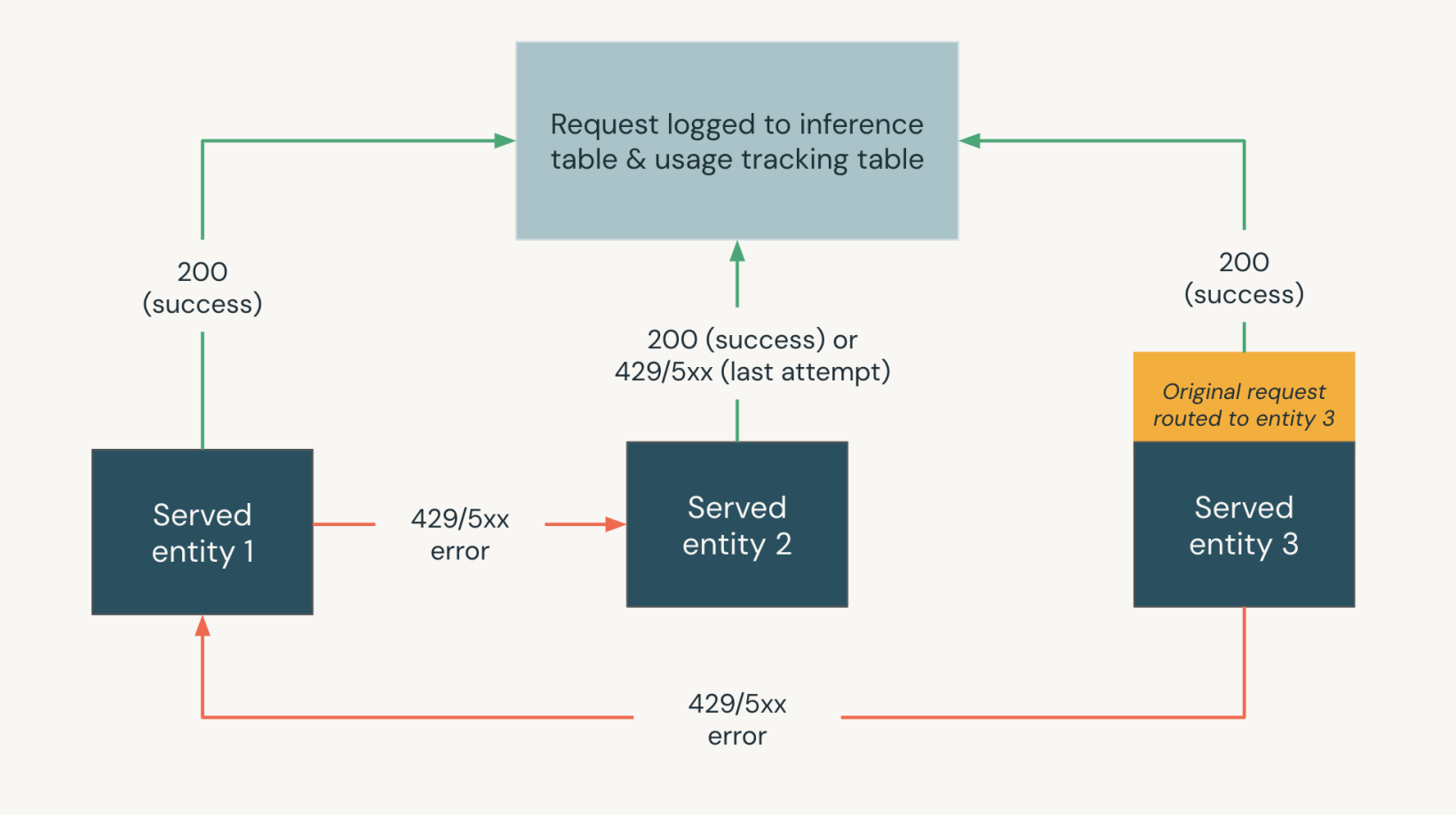


Vector Search and Feature Store

- Feature and Function Serving that serves structured data to RAG applications
- Vector Search to build RAG applications with unstructured data
- Automated real-time pipelines that synchronize with source data in the data lake without any intervention



Example | AI Gateway on model serving endpoints



Demo

Create and query a vector search index

External models in Mosaic AI Model Serving

- External models allow you to streamline the usage and management of various large language model (LLM) providers, such as

  OpenAI

  Anthropic

Python Copy

```python
import mlflow.deployments

client = mlflow.deployments.get_deploy_client("databricks")
client.create_endpoint(
    name="openai-completions-endpoint",
    config={
        "served_entities": [{
            "name": "openai-completions",
            "external_model": {
                "name": "gpt-3.5-turbo-instruct",
                "provider": "openai",
                "task": "llm/v1/completions",
                "openai_config": {
                    "openai_api_key": "{{secrets/my_openai_secret_scope/openai_api_key}}"
                }
            }
        }]
    }
)
```

Anthropic | Configure a serving endpoint

- To serve and query external models you need to configure a serving endpoint
- The *external_model* field defines the model to which this endpoint forwards requests

Python Copy

```python
import mlflow.deployments

client = mlflow.deployments.get_deploy_client("databricks")

client.create_endpoint(
    name="anthropic-completions-endpoint",
    config={
        "served_entities": [
            {
                "name": "test",
                "external_model": {
                    "name": "claude-2",
                    "provider": "anthropic",
                    "task": "llm/v1/completions",
                    "anthropic_config": {
                        "anthropic_api_key": "{{secrets/my_anthropic_secret_scope/anthropic_api_key}}
                    }
                }
            }
        ]
    }
)
```

Anthropic | Configure the provider for an endpoint

Configuration Parameter	Description	Required
anthropic_api_key	The Azure Databricks secret key reference for an Anthropic API key. If you prefer to paste your API key directly, see *anthropic_api_key_plaintext*	You must provide an API key using one of the following fields: *anthropic_api_key* or *anthropic_api_key_plaintext*
anthropic_api_key_plaintext	The Anthropic API key provided as a plaintext string. If you prefer to reference your key using Azure Databricks Secrets, see *anthropic_api_key*	You must provide an API key using one of the following fields: *anthropic_api_key* or *anthropic_api_key_plaintext*

Anthropic | Query a model endpoint

- You can send scoring requests to the endpoint using the OpenAI client, the REST API or the MLflow Deployments SDK
- The given example queries the claude-2 completions model hosted by Anthropic using the OpenAI client

Python

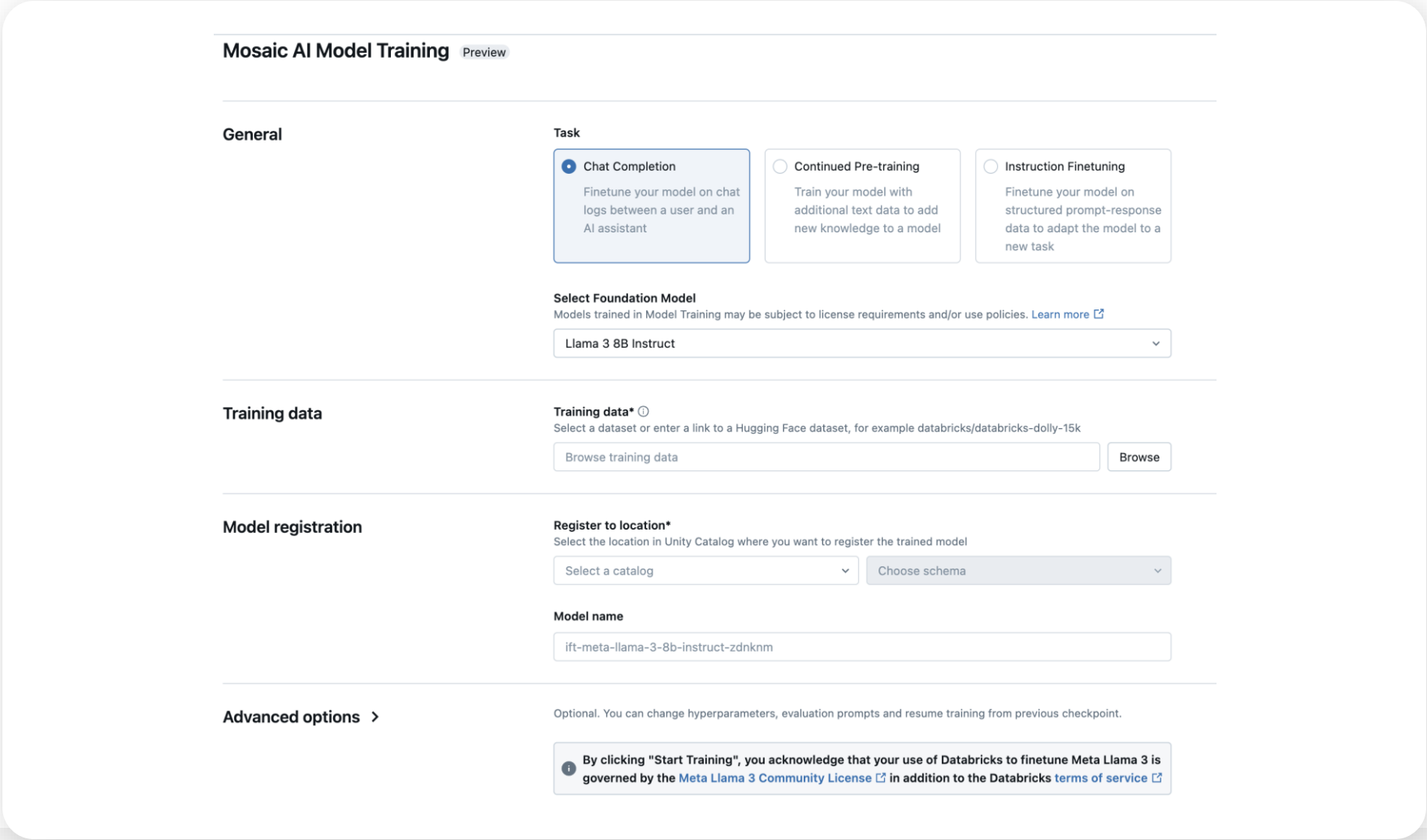
```python
import os
import openai
from openai import OpenAI

client = OpenAI(
    api_key="dapi-your-databricks-token",
    base_url="https://example.staging.cloud.databricks.com/serving-endpoints"
)

completion = client.completions.create(
  model="anthropic-completions-endpoint",
  prompt="what is databricks",
  temperature=1.0
)
print(completion)
```

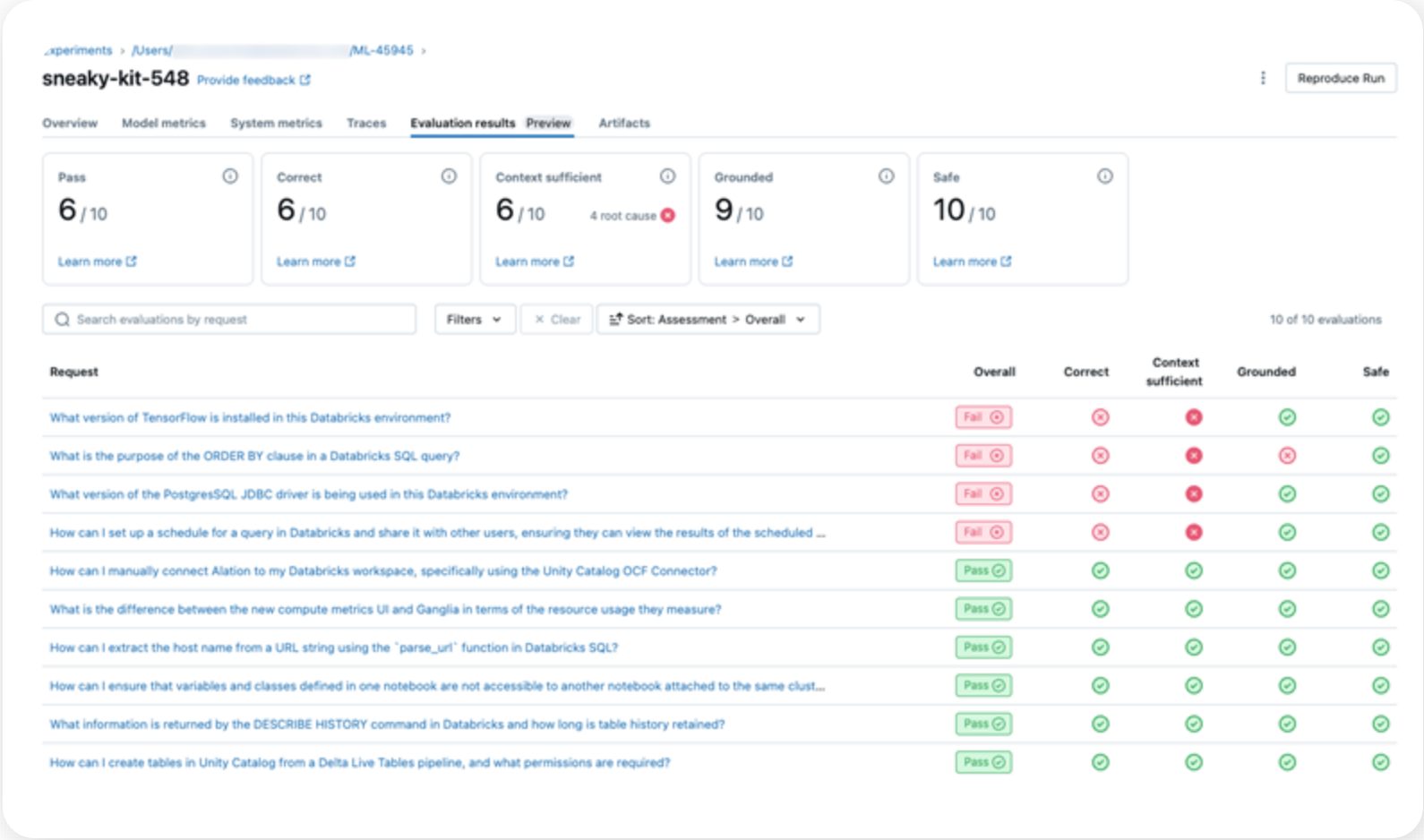
Mosaic AI Model Training

- Streamlines and unifies the process of training and deploying traditional ML models through
  - AutoML
  - Foundation Model Fine-tuning workloads

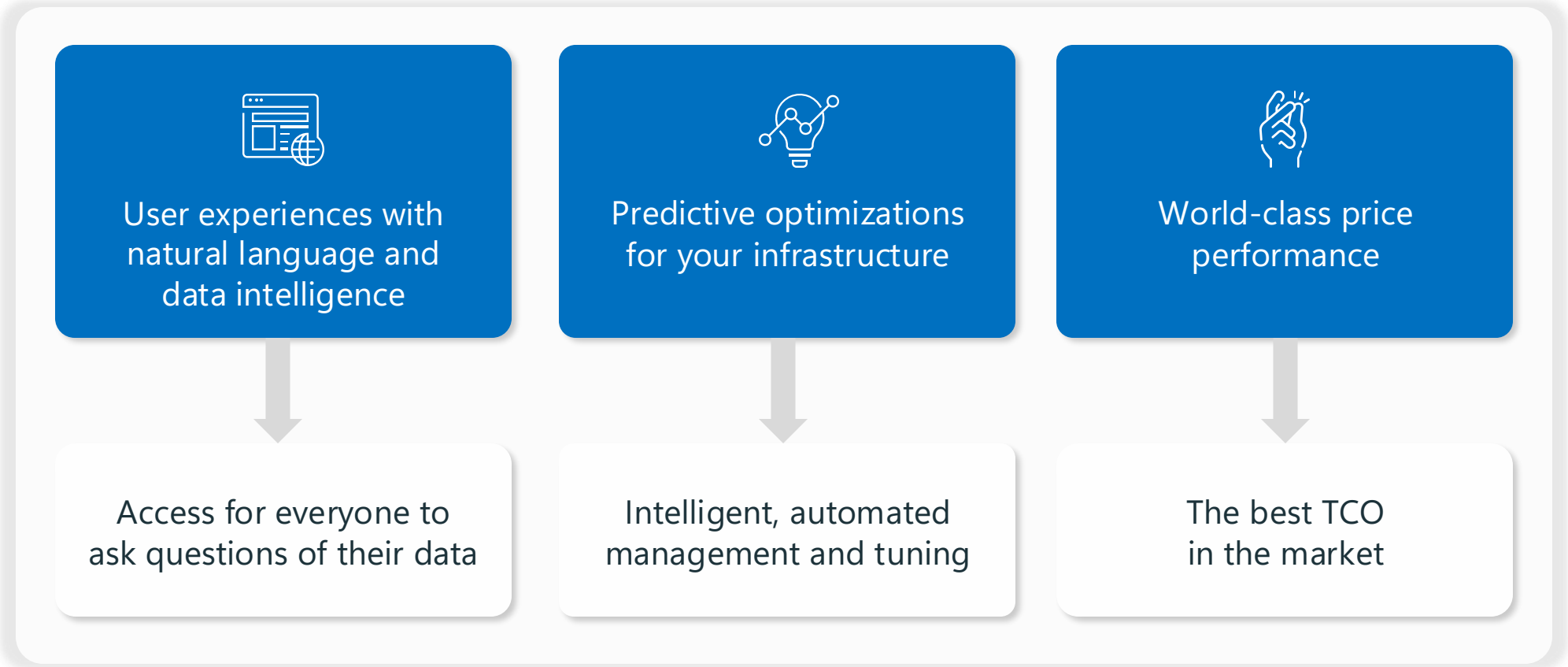


Agent evaluation

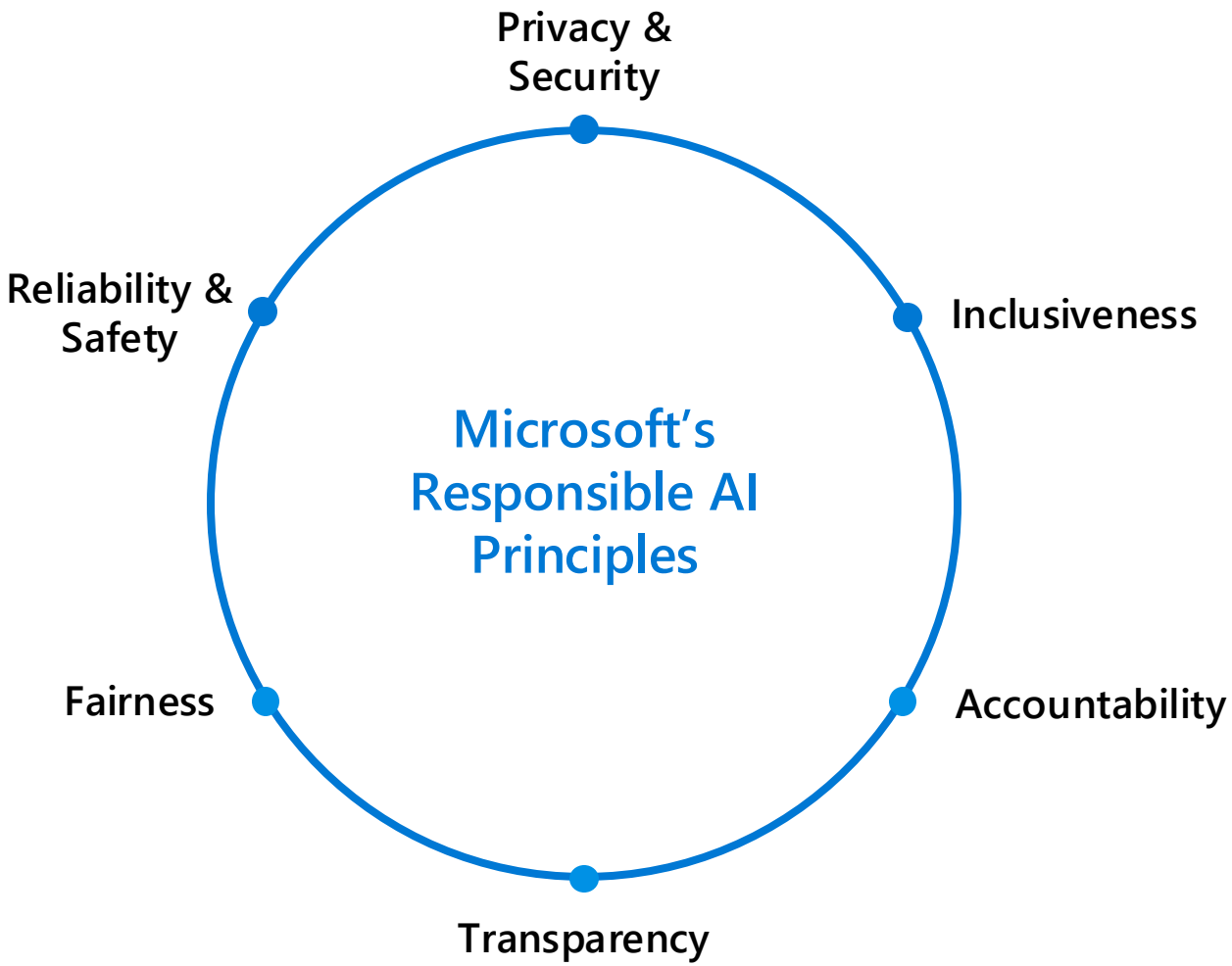
- Evaluate the quality, cost, and latency of Agents, including RAG applications and chains
- Human and machine evaluators which use advanced, research-backed techniques
- Synthetic data generation API creates high-quality evaluation data based on enterprise data



Intelligent Data Warehousing



Microsoft's Responsible AI Principles



Building blocks to enact principles

- Tools and processes
- Training and practices
- Rules
- Governance

Query Generative AI model serving endpoints

Query Generative AI models

- Mosaic AI Model Serving supports Foundation Models APIs and external models for accessing Generative AI models
- Model Serving uses a unified OpenAI-compatible API and SDK for querying them

Python

```python
from databricks.sdk import WorkspaceClient

w = WorkspaceClient()
openai_client = w.serving_endpoints.get_open_ai_client()

response = openai_client.chat.completions.create(
    model="databricks-dbrx-instruct",
    messages=[
      {
        "role": "system",
        "content": "You are a helpful assistant."
      },
      {
        "role": "user",
        "content": "What is a mixture of experts model?",
      }
    ],
    max_tokens=256
)
```

Model Serving | Options for sending scoring requests to endpoints

- OpenAI client
- Serving UI
- REST API
- MLflow Deployments SDK
- Databricks Python SDK
- SQL function

Demo

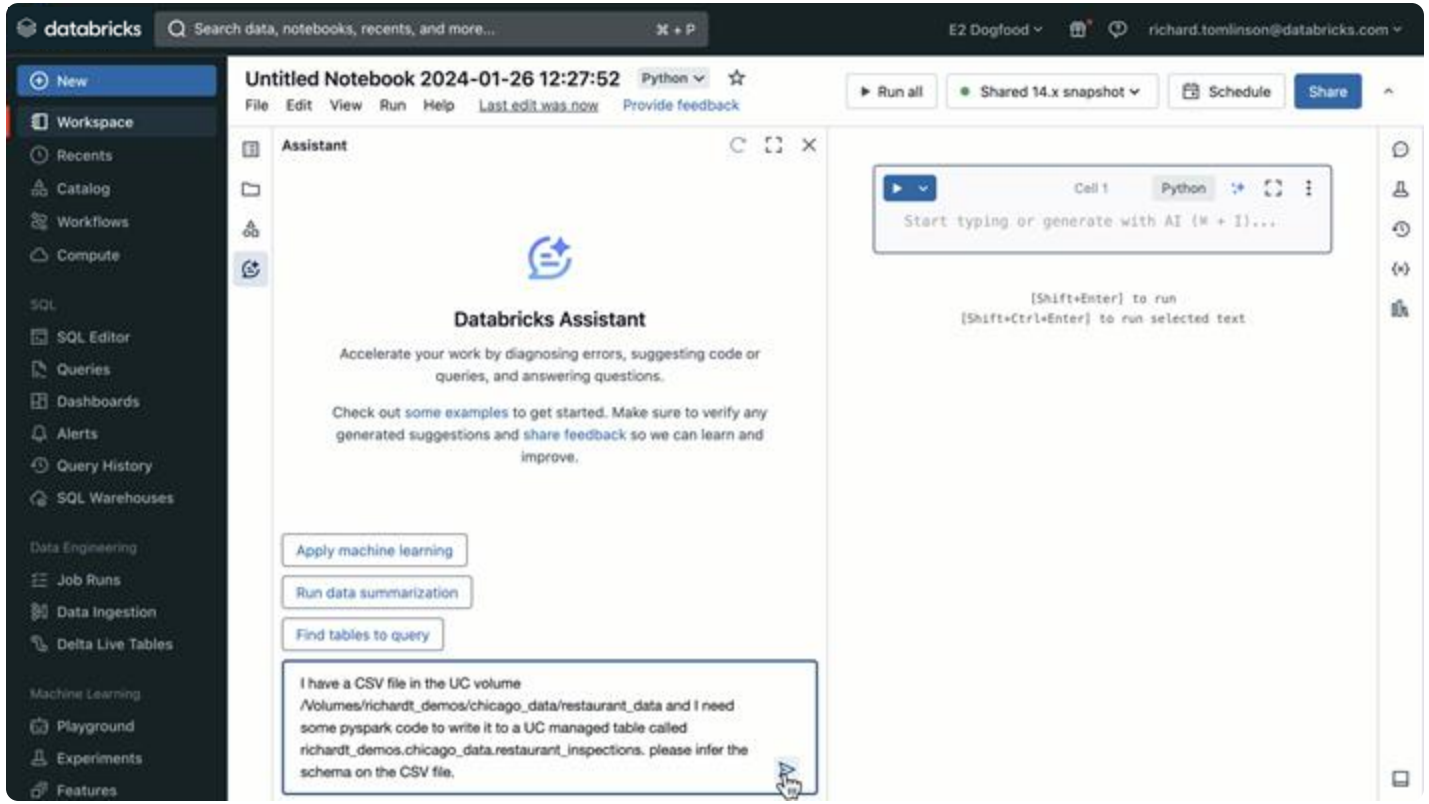
Query a chat completion model

Databricks Assistant, AI/BI Genie and AI Functions on Azure Databricks

Databricks Assistant

Boost productivity with your context-aware AI assistant

- Generate, fix and optimize code using natural language
- Leverages UC metadata, code cells, libraries, runtime, and more as context to improve accuracy
- Explains, diagnoses, and fixes issues from within a cell within seconds

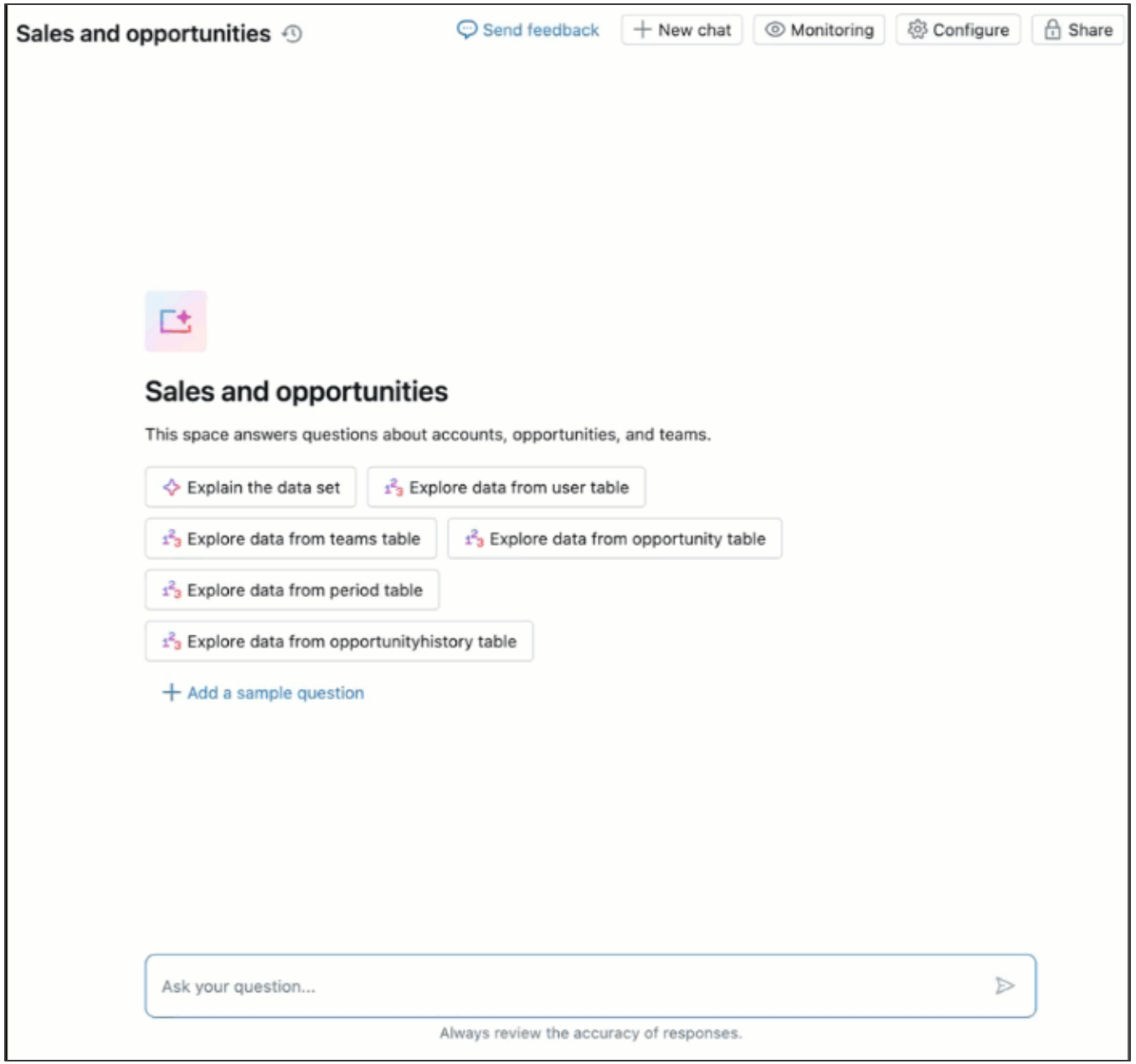


Databricks Assistant Quick Fix
has a 75% acceptance rate

🚀 Over 150K users assisted every month! 🚀

AI/BI Genie space

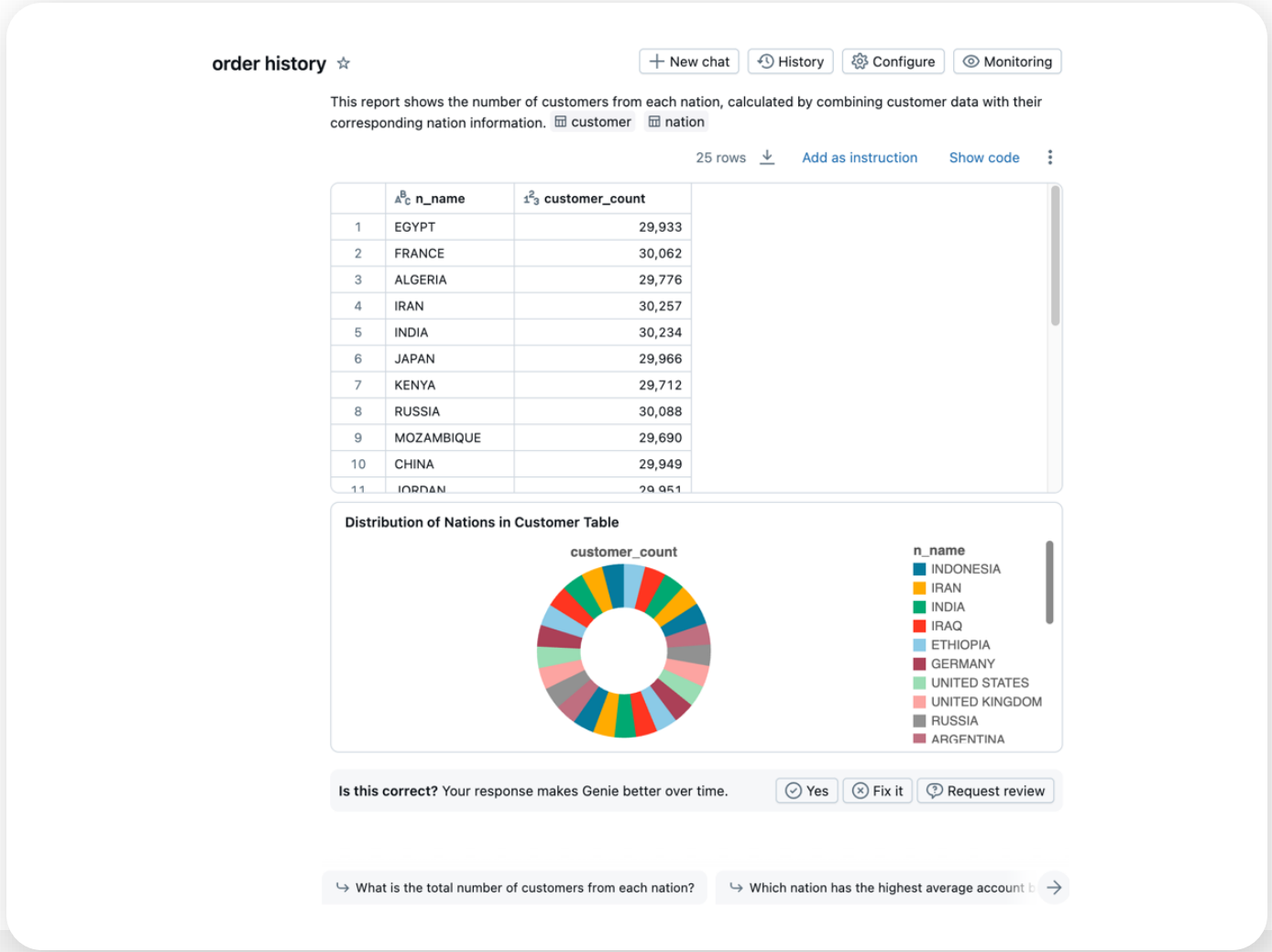
- A chat interface for business users to ask natural-language questions about their data
- Configure Genie spaces with datasets, sample queries, and text guidelines to help Genie translate business questions into analytical queries



Generate responses with Genie

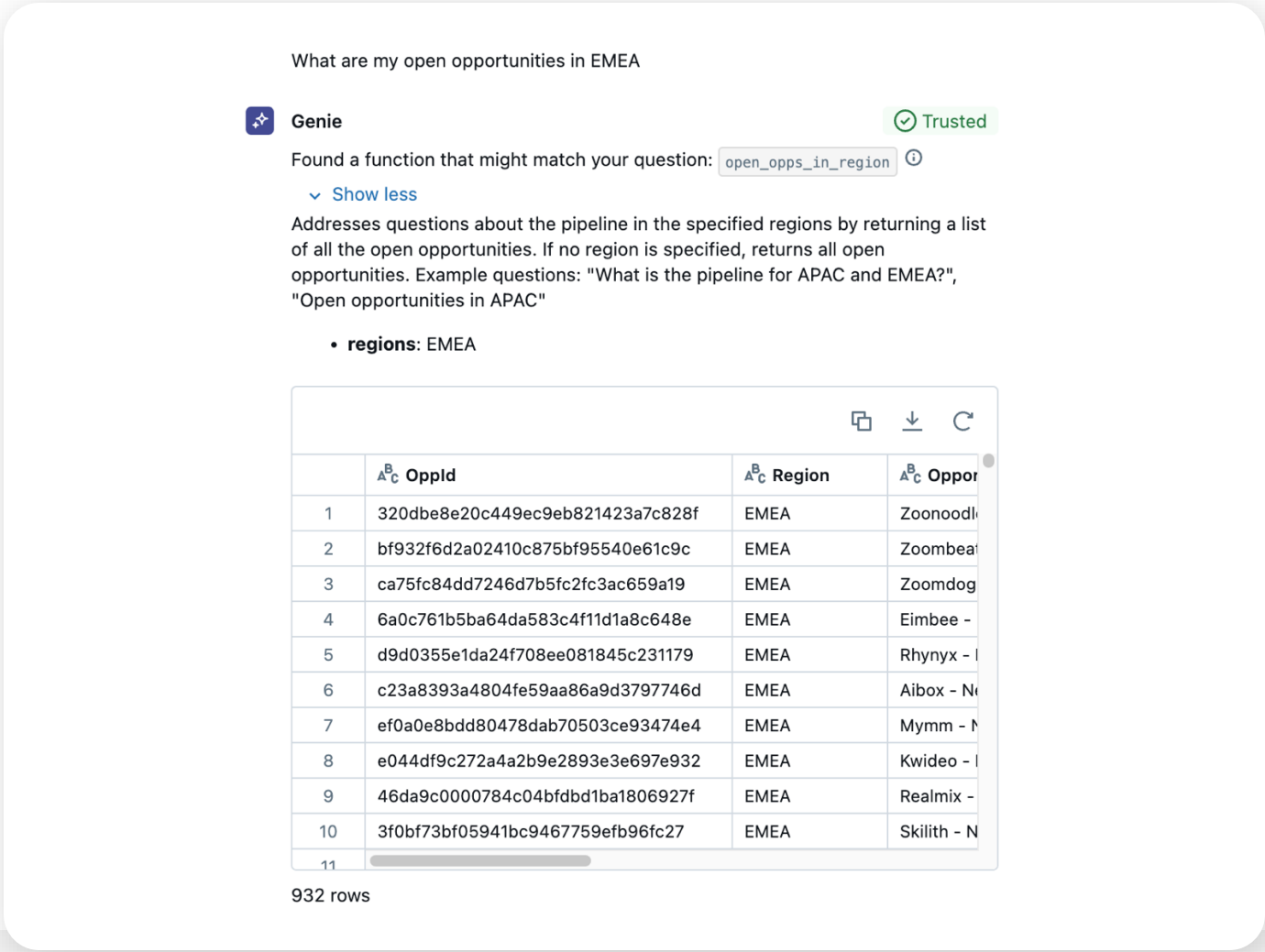
Genie uses the following components to generate responses:

- Table metadata
- Column names and descriptions
- Example SQL queries
- SQL functions
- Instructions
- Prompt and responses history



Trusted assets in Genie

- Trusted assets are predefined functions and example queries meant to provide verified answers to questions that you anticipate from users
- When a user submits a question that invokes a trusted asset, it's indicated in the response, adding an extra layer of assurance to the accuracy of the results



AI Functions on Azure Databricks

AI Functions invoke a state-of-the-art Generative AI model from Databricks Foundation Model APIs to perform tasks like,

- Sentiment analysis
- Classification
- Translation

SQL

```
> SELECT ai_analyze_sentiment('I am happy');
  positive

> SELECT ai_analyze_sentiment('I am sad');
  negative
```

SQL

```
> SELECT ai_translate('Hello, how are you?', 'es');
 "Hola, ¿cómo estás?"

> SELECT ai_translate('La vida es un hermoso viaje.', 'en');
 "Life is a beautiful journey."
```

Chat with LLMs and prototype GenAI apps using AI Playground

LangChain on Azure Databricks for LLM development

LangChain is a software framework designed to help create applications that utilize large language models (LLMs)

With LangChain integrations you can

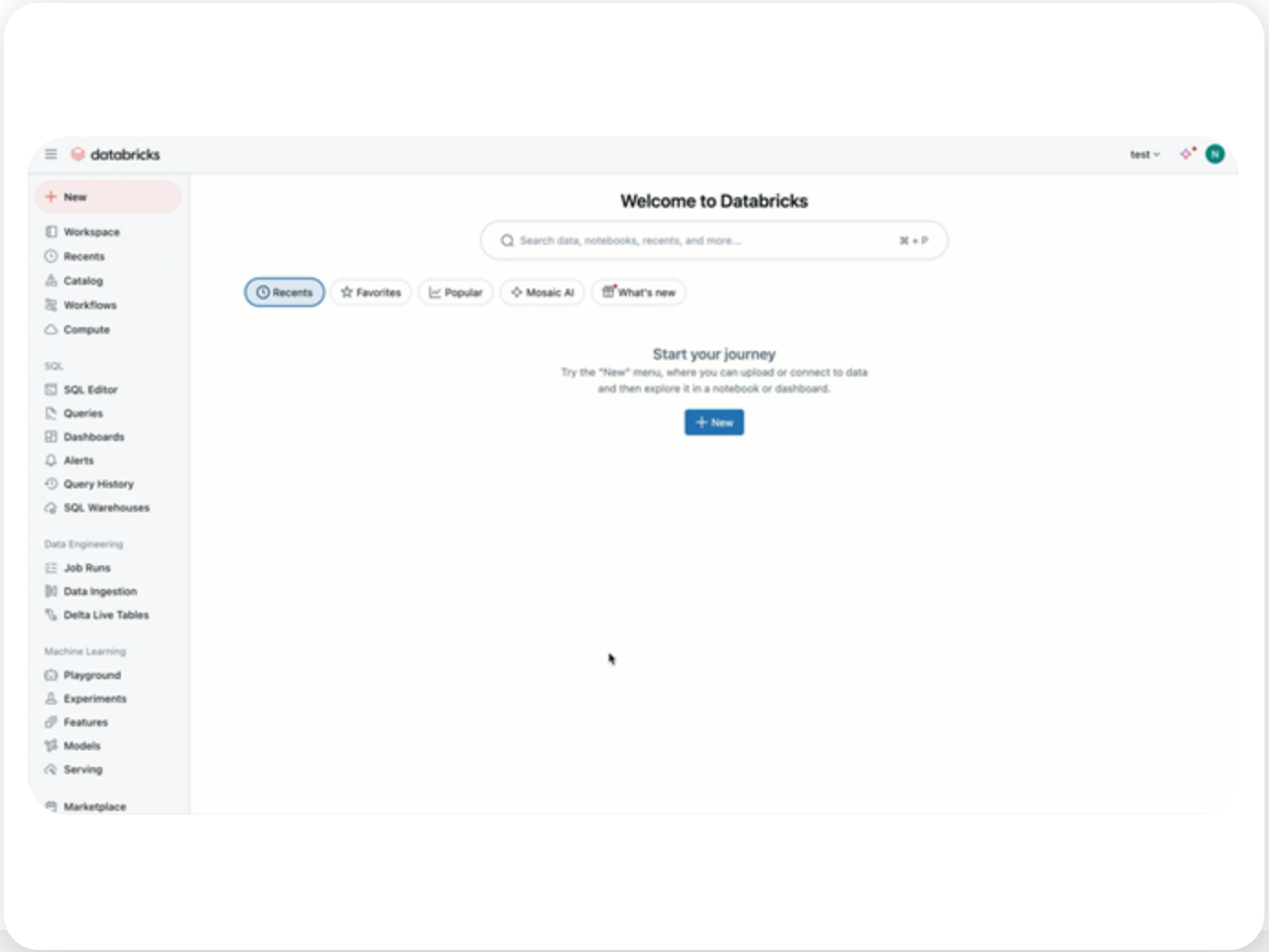
Seamlessly load data from a PySpark DataFrame with the PySpark DataFrame loader

Interactively query your data using natural language with the Spark DataFrame Agent or Databricks SQL Agent

Wrap your Azure Databricks served model as a large language model (LLM) in LangChain

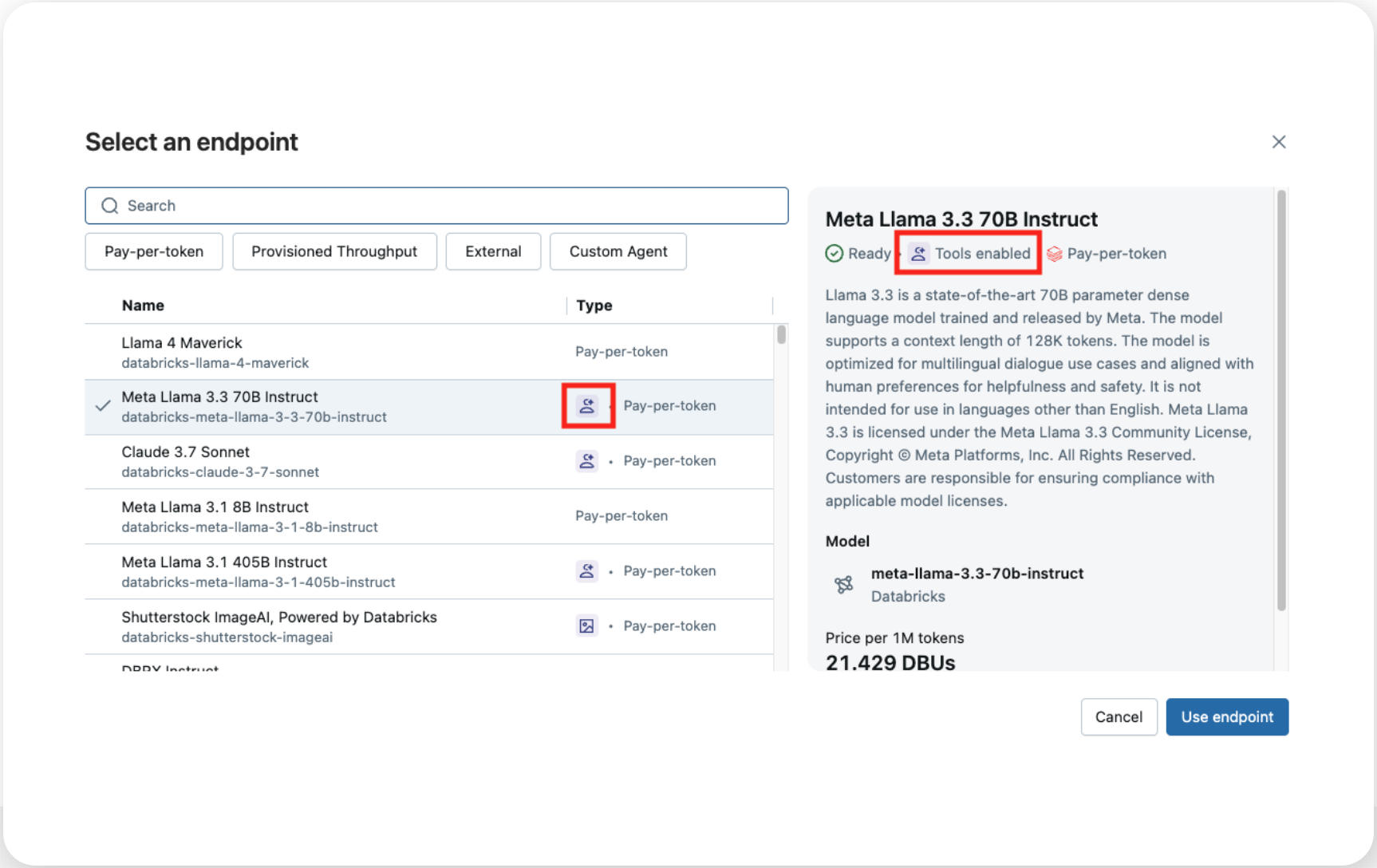
AI Playground

- Interactive environment where users can experiment with various models simultaneously and compare results
- Test any model endpoint hosted in Databricks or outside
- Automatically tracks prompt engineering experiments to build evaluation datasets



Prototype Generative AI agents in AI Playground

- Use the AI Playground to give the tool to an agent and interact with it to validate and test behavior
- The AI Playground provides a sandbox to prototype tool-calling agents



Build and optimize agents on your data with Agent Bricks

Building Agents today is challenging

Evaluation is difficult

Too many knobs

Trade-off cost versus quality

Agent Bricks

Powered by Databricks Mosaic AI Research

Best-of-N

Adaptive LLM judges

Test-time adaptive optimization

Constitutional AI

Deep research agents

MixAttention

Fine-tuning

Value-guided decoding

Critique filtering

RLHF

Prompt optimization

Synthetic data generation

Automated evaluation

Multi-agentic patterns

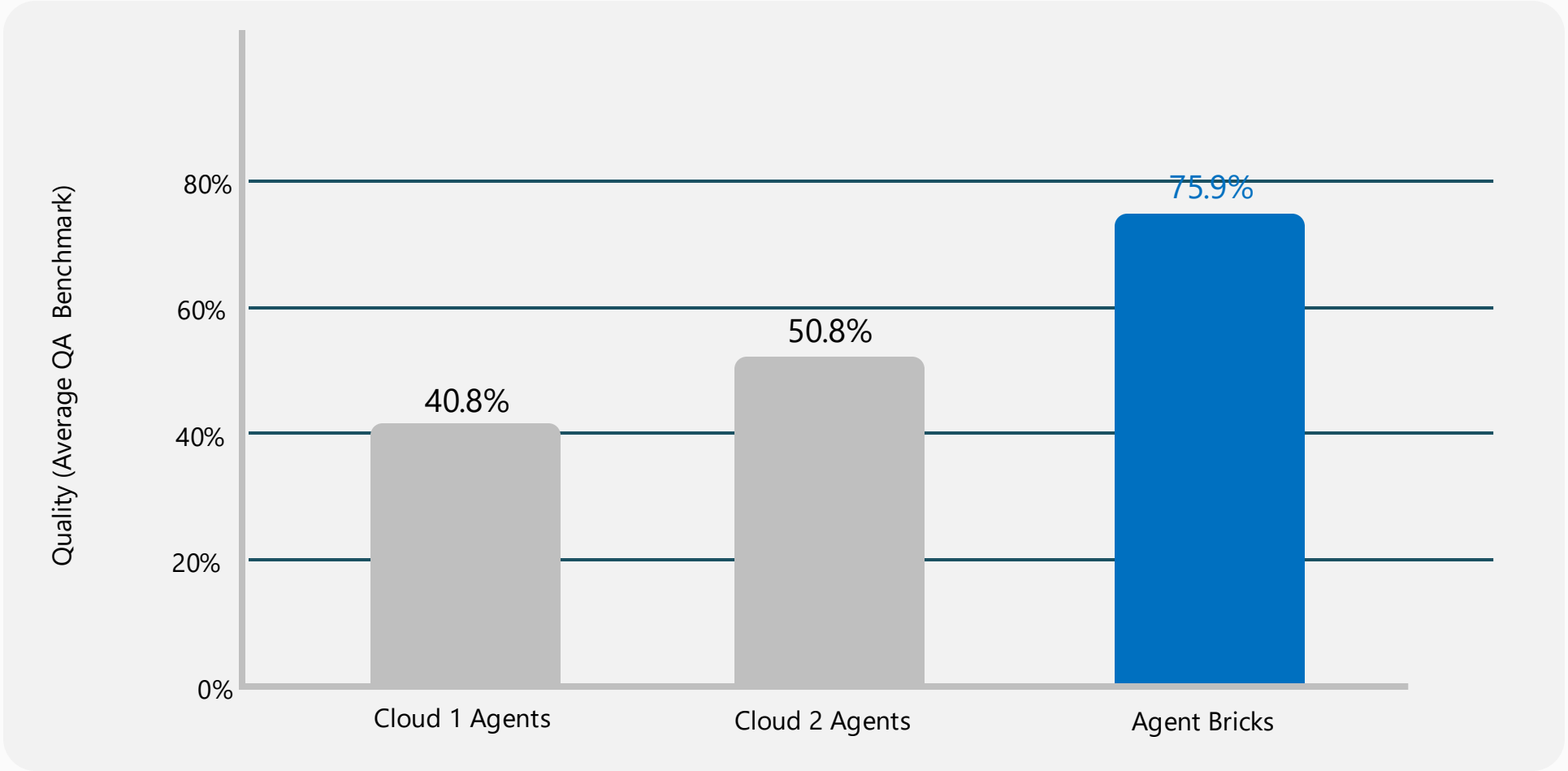
Agentic knowledge graphs

o1-style reasoning

Contextual retrieval

RLVR

High Quality Out of the Box



Use Cases

Agent Bricks

Information Extraction Agent

Problem: Data is buried

Value: Turn unstructured text into structured fields like names, dates, and entities—no labeled data needed.

Custom LLM Agent

Problem: Teams rely on generic models or manual effort to tailor text for their domain.

Value: Transform text for tasks like summarization, classification, or rewriting—optimized for your industry.

Knowledge Assistant Agent

Problem: Chatbots give vague or incorrect answers

Value: Turn unstructured text into structured fields like names, dates, and entities—no labeled data needed.

Multi-Agent Supervisor

Problem: Complex tasks break down when a single model tries to do everything at once.

Value: Deliver fast, accurate answers grounded in your enterprise data.

Information Extraction

Extract key information from documents to structured outputs

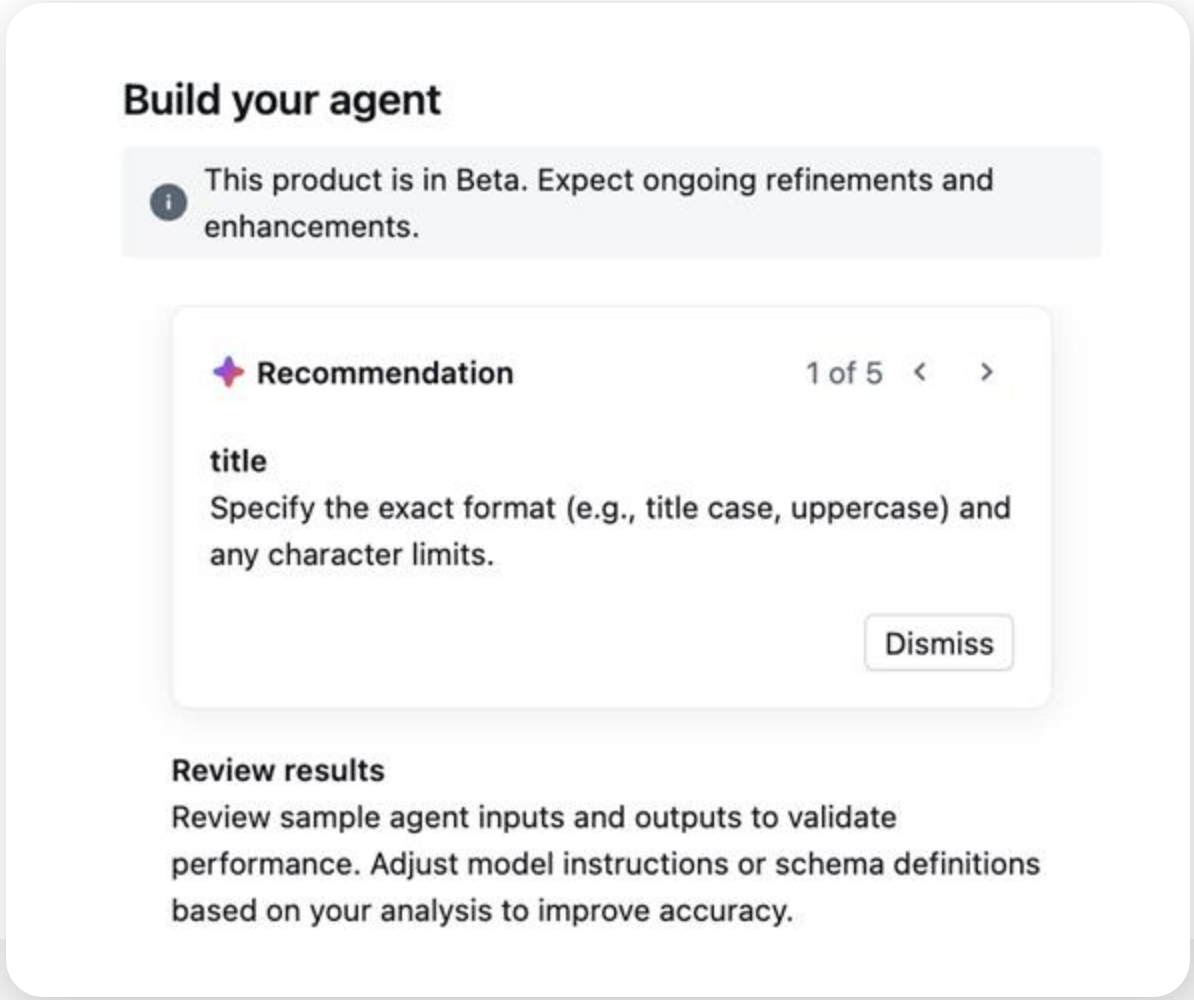
Example use cases:

Legal documents or contracts

Extract key details (parties, terms, dates) weekly to verify important discrepancies.

Customer reviews

Identify and categorize product names daily to analyze trending products and customer sentiment.



Custom LLM

Convert unstructured data into actionable insights

Example use cases:

Summarize customer service calls

Representatives can understand past customer touchpoints to improve interactions.

Generate reports

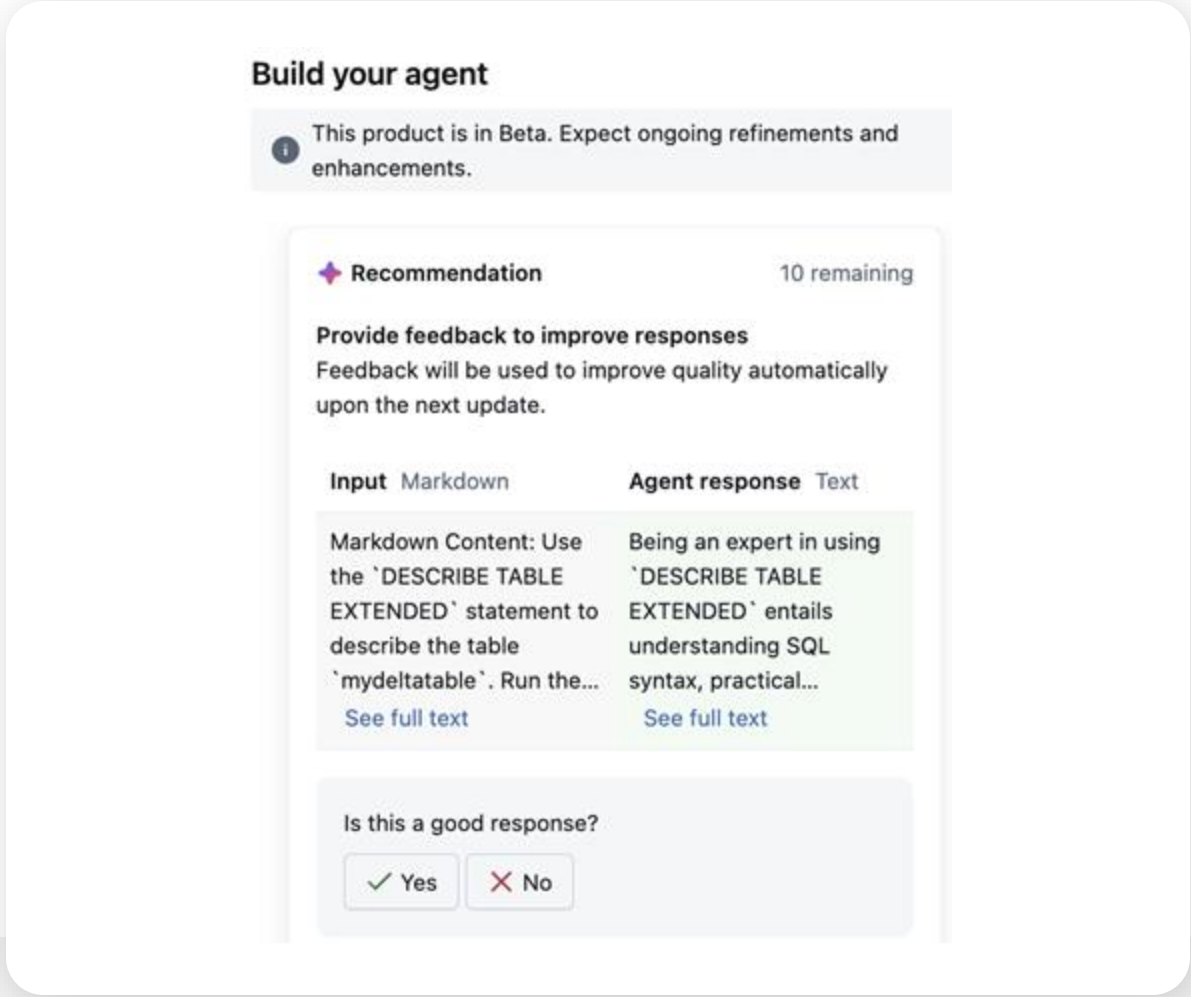
Distill key insights and save employee time.

Tailor a chat model to enterprise policies

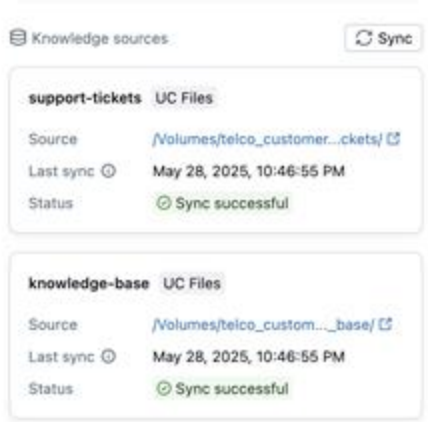
An LLM will adhere to enterprise guidelines when replying to chats.

Create marketing material

Generate engaging headlines aligned with company styles

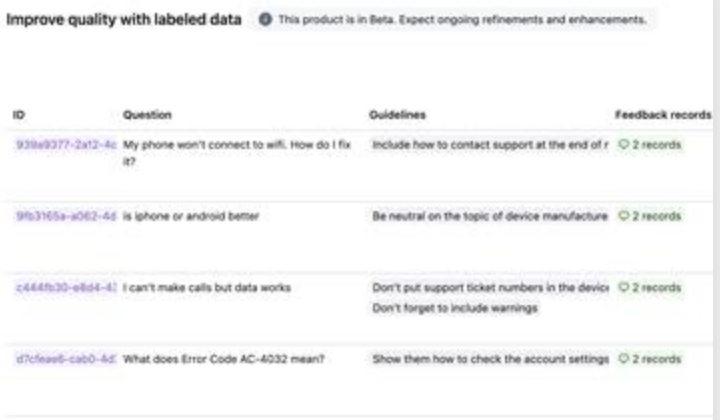


Use Knowledge Assistant for the answers you need



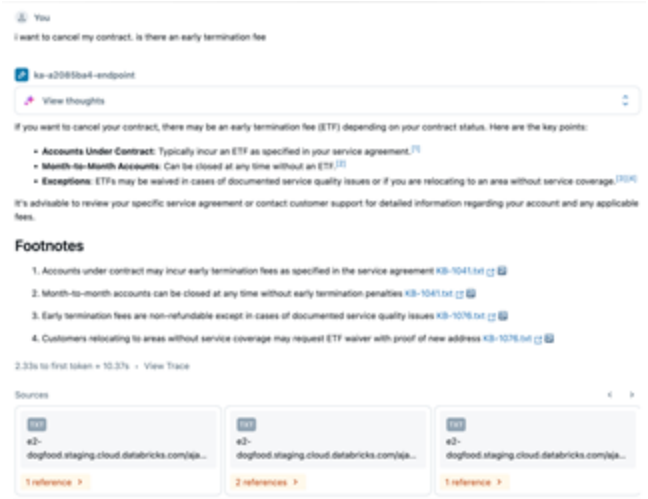
End-to-end ingestion handled for you

File parsing, ingestion, chunking, and embedding all handled for you



Agent learning & evaluation built-in

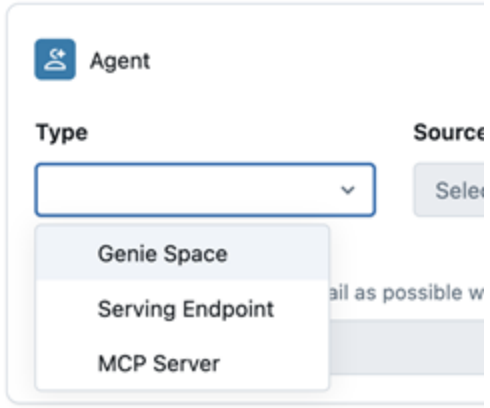
Gather feedback from SMEs & improve the model over time



Ready to scale endpoint out-of-the-box

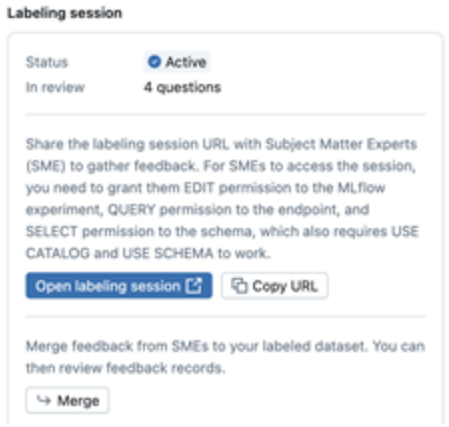
Easily embed your endpoint wherever you need it - including a Databricks App

Multi-Agent Supervisor makes the possibilities endless



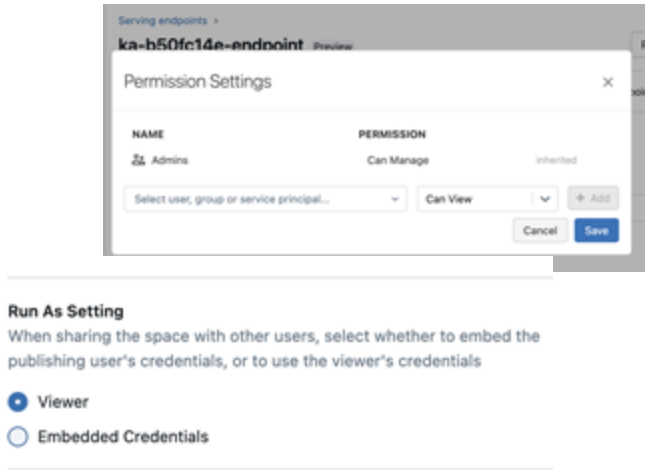
Flexible for your use case

Build agents that can reason over all your data – structured or unstructured



Agent Learning & evaluation built-in

Gather feedback from SMEs & improve the model over time



Governance & access built-in

Leverage existing permissions with each agent so you only expose data they have access to

Coming up next...

Day 1

Module 1 - Introduction to Azure Databricks
- Azure Databricks: A Data Intelligent Platform
- Why Azure Databricks
- Decision guide: Azure Databricks vs. Microsoft Fabric

Module 2 - Migration to Azure Databricks
- Microsoft Cloud Adoption Framework for Azure
- Migration strategies
- Data landing zones
- Migration scenarios

Interactive Simulated Lab Experience
- End-to-End Streaming Pipeline with Lakeflow Declarative Pipelines in Azure Databricks

Day 2

Module 3 - Integration with Azure
- Seamless integration with Microsoft Azure services
- Connect to Azure Data Lake Storage (ADLS) Gen2 and Blob Storage
- Leverage Azure Databricks for Azure Cosmos DB Operations
- Secret management with Azure Key Vault
- Connect Azure Databricks to Azure Event Hubs

Module 4 - Integration with Microsoft Fabric and Power BI
- Data Intelligence with Azure Databricks and Microsoft Fabric
- Connect Power BI to Azure Databricks
- Integration with Azure Data Factory
- Mirroring Azure Databricks Unity Catalog

Interactive Simulated Lab Experience
- Setup and use Unity Catalog for Data Management in Azure Databricks
- Real-Time Streaming with Azure Databricks and Azure Event Hubs

Day 3

Module 5 - Integration with Azure AI Foundry
- Azure Databricks connector in Azure AI Foundry
- Mosaic AI and machine learning on Azure Databricks
- Query Generative AI model serving endpoints
- Databricks Assistant, AI/BI Genie and AI Functions on Azure Databricks
- Chat with LLMs and prototype GenAI apps using AI Playground
- Build and optimize agents on your data with Agent Bricks

Module 6 - Security and Governance
- Integrate Azure Databricks with Microsoft Purview
- Integration of Azure Databricks Unity Catalog with Microsoft Purview

Module 7 - Well-architected for Azure Databricks
- Lakehouse implementation: Principles and best practices
- Azure Databricks well-architected framework

Interactive Simulated Lab Experience
- Responsible AI with Large Language Models using Azure Databricks and Azure OpenAI
- Connect to and manage Azure Databricks in Microsoft Purview

Microsoft

Technofocus

Thank You!