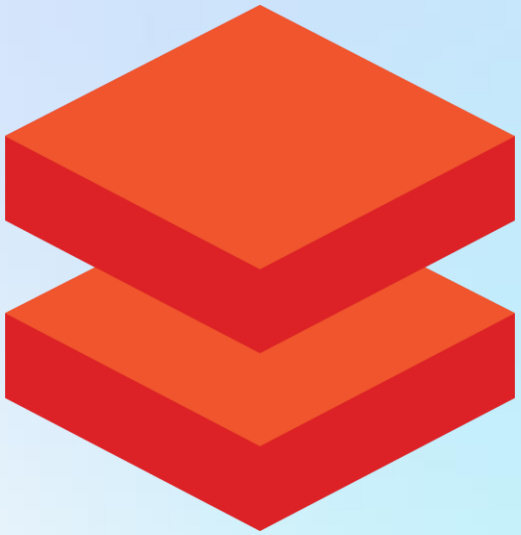
EPAM AI Factory

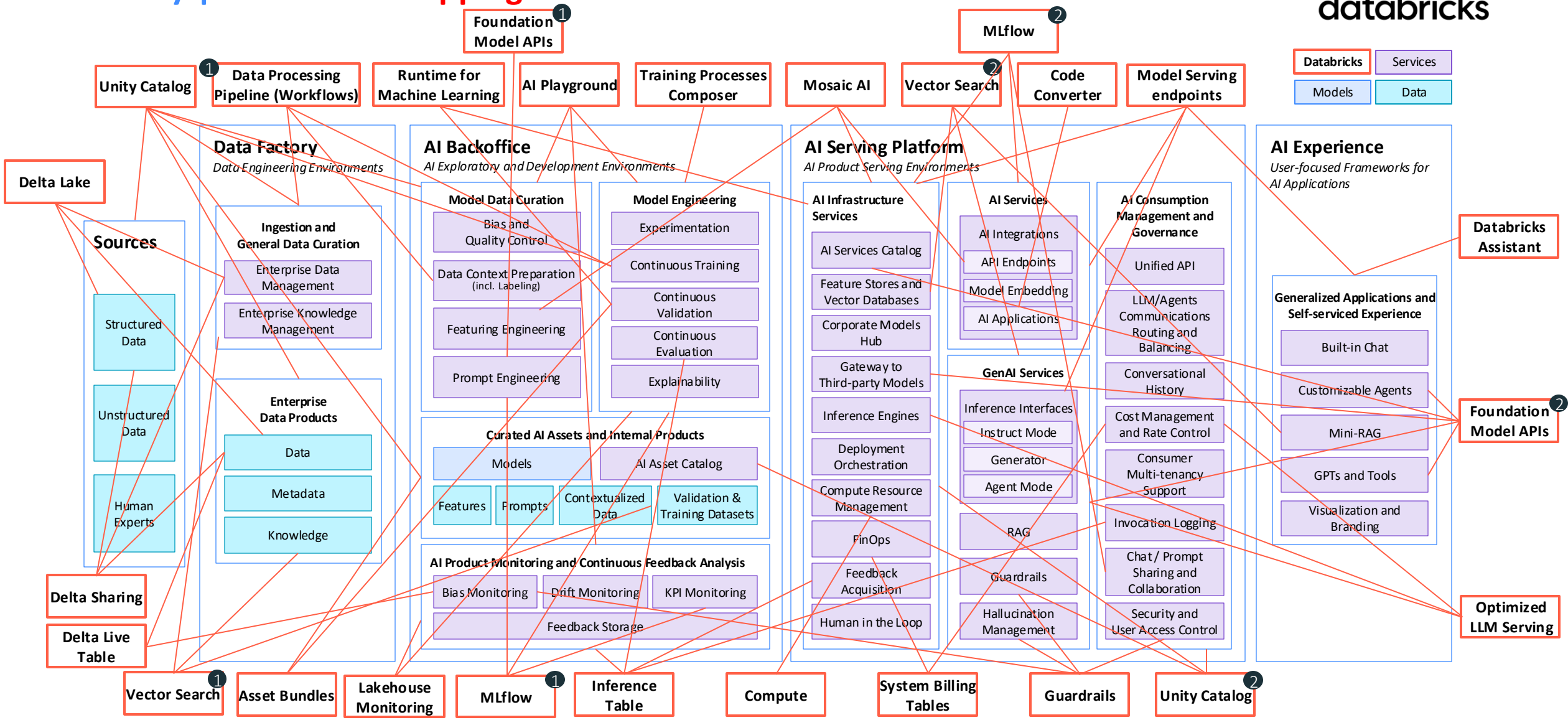
Technology Mapping Task for Architecture POV v0.2

2025

AI Factory Blueprint Databricks Mappings



AI Factory | Databricks Mapping



AI Factory | Databricks

Blueprint High-level, Sub-platforms and Major Capabilities Mapping – part 1

EPAM VISION		VISION	
SUB-PLATFORM	**MAJOR CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
N/A	*Sources*	Databricks integrates external structured sources—databases, data warehouses, object storage, and streaming platforms—via native connectors and Unity Catalog query federation, enabling secure, governed, read-only external tables without data replication.	• Connect to data sources • Unity Catalog: Data Governance • Lakehouse Federation Reference Architecture • Query Data by Path
Data Factory	*Ingestion and General Data Curation*	Databricks streamlines data ingestion via native connectors, Auto Loader, COPY INTO and Lakeflow Connect for SaaS and databases, supporting both batch and streaming workloads into Delta Lake tables.	• Efficient data ingestion • Ingest data into a Databricks lakehouse
	Enterprise Data Products	Enterprise Data Products in Databricks merge a Data-as-a-Product mindset—packaging datasets and services with clear schemas, SLAs, and APIs—with centralized governance and metadata via Unity Catalog, and shared Common Data Capabilities—cataloging, lineage, access controls, monitoring—to deliver high-quality, trusted data assets at scale.	• Guiding principles for the lakehouse • Building High-Quality and Trusted Data Products with Databricks • What is Databricks Marketplace?
	Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint	Databricks' Data Factory sub-platform excels at ingesting data (Auto Loader, Lakeflow) into Delta Lake and creating governed Enterprise Data Products via Unity Catalog (metadata, lineage, access). Key capabilities missing from the description include native Master/Reference Data Management, dedicated data modeling tools, advanced DQ remediation workflows, and explicit archiving features.	
AI Backoffice	*Model Data Curation*	Databricks model data curation centralizes feature versioning, serving and lineage in the Feature Store and automates schema enforcement and data-quality validation via Delta Live Tables expectations. Model versioning, staged deployments and governance are managed through the MLflow Model Registry integrated with Unity Catalog	• Feature engineering and serving • Manage data quality with pipeline expectations • Manage model lifecycle in Unity Catalog
	Model Engineering	Databricks offers an integrated environment for model engineering: collaborative notebooks, ML Runtimes, MLflow (tracking, packaging), Feature Store, and Model Registry for the end-to-end ML lifecycle.	• Machine Learning on Databricks • AI and machine learning on Databricks • MLflow Guide
	…	…	• …

AI Factory | Databricks

Blueprint High-level, Sub-platforms and Major Capabilities Mapping – part 2

EPAM VISION		VISION	
SUB-PLATFORM	**MAJOR CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Backoffice (continuation)	*Curated AI Assets and Internal Products*	Databricks enables curation and sharing of AI assets through Unity Catalog (discovery, governance), Feature Store (shared features), and MLflow Model Registry (versioned models), facilitating internal AI product creation.	• Unity Catalog • Feature engineering and serving • Manage model lifecycle in Unity Catalog
	AI Product Monitoring and Continuous Feedback Analysis	Databricks supports AI monitoring via Model Serving inference tables (automatic logging), Lakehouse Monitoring (drift, quality), and MLflow tracking, enabling analysis of performance and feedback data.	• Monitor models served with Databricks Model Serving • Introduction to Databricks Lakehouse Monitoring • Databricks AI Playground for interactive model exploration
	Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint	Databricks' AI Backoffice integrates model data curation, engineering (MLflow, notebooks), asset management (Unity Catalog, Registry), and monitoring (Lakehouse Monitoring). Key missing capabilities described include advanced XAI/bias tools, native data labeling, integrated Human-in-the-Loop workflows, and sophisticated pre-deployment simulation/testing frameworks.	
AI Serving Platform	*AI Infrastructure Services*	Databricks provides managed AI infrastructure including scalable compute (CPU/GPU), Lakehouse storage integration, optimized ML Runtimes, orchestration, and security, abstracting cloud complexities.	• Databricks Compute for AI workloads • Databricks Data Intelligence Platform • Compute configuration reference • Databricks Runtime for Machine Learning
	AI Services	Databricks primarily enables hosting custom AI models as scalable API services via Model Serving and supports integrating calls to external AI/ML services.	• Introduction to Databricks Model Serving • External Models in Databricks Model Serving • Databricks Machine Learning • MLflow Models
	…	…	• …

AI Factory | Databricks

Blueprint High-level, Sub-platforms and Major Capabilities Mapping – part 3

EPAM VISION		VISION	
SUB-PLATFORM	**MAJOR CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Serving Platform (continuation)	*GenAI Services*	Databricks provides specific GenAI services including Foundation Model APIs (pay-per-token), optimized Model Serving for LLMs, Vector Search for RAG, and tools supporting the LLM development lifecycle.	• Databricks LLM-optimized serving endpoints • Large Language Models (LLMs) on Databricks • Foundation Model APIs • Databricks Vector Search • Retrieval-Augmented Generation (RAG)
	AI Consumption Management and Governance	Databricks governs AI consumption via Unity Catalog (access control for models, features, data), endpoint permissions, cost tracking (system tables, tags), auditing, and lineage.	• Unity Catalog • Manage model lifecycle in Unity Catalog • Manage model serving endpoints • Monitor usage using system tables
	Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint	Databricks' AI Serving Platform provides managed infrastructure, hosts models via Model Serving and Foundation APIs (including GenAI/RAG), and governs consumption using Unity Catalog. It lack advanced deployment strategies (canary/shadow), integrated A/B testing frameworks, fine-grained endpoint tuning, built-in API feedback mechanisms, and service mesh features.	
AI Experience	*Generalized Applications and Self-serviced Experience*	Databricks enables AI experiences via Databricks Assistant (technical self-service), powering custom apps through Model Serving, BI tool integration, Lakeview Dashboards, and Solution Accelerators as starting points.	• Databricks Assistant • Databricks AI/BI • Databricks Model Serving • Solution Accelerators
	Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint	Databricks' AI Experience enables interaction via Assistant (devs), Model Serving APIs, BI/dashboards, and Solution Accelerators, emphasizing custom builds. Missing capabilities include a no-code/low-code builder, pre-built end-user AI apps, direct workflow integration, advanced UI customization frameworks, and dedicated non-technical collaboration features.	
Sub-platform missing in AI Factory Databricks Blueprint		Databricks AI Factory blueprint strongly covers the technical data/AI lifecycle but lacks native features for MDM/RDM, data labeling, HITL, advanced Responsible AI tooling (bias/XAI), and no-code app building. Pre-built apps, BPM integration, advanced A/B testing, explicit archiving, and dedicated prompt management are also missing.	

AI Factory | Databricks

Data Factory Major Capabilities and Capabilities Mapping

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
Ingestion and General Data Curation	*Enterprise Data Management*	Databricks provides EDM capabilities via Unity Catalog (governance, metadata, security, lineage) and Delta Live Tables (integration, quality), leveraging Spark for big data processing and Delta Lake for lifecycle aspects.	• Unity Catalog • Data Governance on Databricks • Delta Live Tables
	Enterprise Knowledge Management	Databricks supports EKM by providing a scalable Lakehouse repository and advanced search/retrieval via Vector Search (essential for RAG/LLMs) but lacks dedicated knowledge capture or comprehensive content management tools.	• Databricks Vector Search • Retrieval-Augmented Generation (RAG) • Large Language Models (LLMs) on Databricks • Work with Files on Databricks
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks offers strong EDM (Unity Catalog/DLT) and supports EKM (storage/Vector Search). Key missing capabilities include MDM/RDM, knowledge capture/content management, advanced DQ remediation, data modeling tools, and explicit archiving.	
Enterprise Data Products	*Data (Asset)*	Databricks treats data (raw, cleansed, aggregated) stored in Delta Lake tables within the Lakehouse as the core asset for data products. Unity Catalog governs access, while Delta ensures reliability, quality (via DLT), and versioning (time travel). Data assets can be securely shared via Delta Sharing.	• Delta Lake • Unity Catalog • Delta Sharing • Delta Live Tables
	Metadata (Asset)	Metadata (schema, lineage, descriptions, tags, access permissions) describing data and AI assets is managed as a critical asset by Unity Catalog. This enables discovery, understanding, trust, and governance of data products and their components.	• Unity Catalog • Data Discovery Using Unity Catalog • View Lineage with Unity Catalog • Add AI-generated comments to Data Objects
	Knowledge (Asset)	Knowledge assets within data products include ML models (managed by Model Registry in Unity Catalog), features (Feature Store in UC), analytical code (Notebooks, Repos), and insights derived from data (visualizations, dashboards). Vector Search enables finding knowledge within unstructured text assets.	• Databricks Vector Search • RAG on Databricks • Introduction to Databricks Notebooks • Unity Catalog
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks manages data, metadata (Unity Catalog), and knowledge assets (Vector Search) for governed data products via Delta Sharing. Missing: formal contracts, monetization, business glossary integration, semantic layer, business SLAs, advanced packaging.	
Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint		Databricks manages data/metadata/knowledge products via Unity Catalog/DLT/Vector Search. Missing capabilities include MDM/RDM, dedicated EKM features, data contracts, business glossary integration, semantic layer, monetization/SLAs, and advanced DQ remediation.	

AI Factory | Databricks

AI Backoffice Major Capabilities and Capabilities Mapping – part 1

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
Model Data Curation	*Bias and Quality Control*	Databricks facilitates bias detection and quality control in model data using Lakehouse Monitoring (statistical profiling, drift detection), Delta Live Tables (data quality expectations), and notebooks for custom analysis or integrating fairness libraries (e.g., Fairlearn).	• Introduction to Databricks Lakehouse Monitoring • Manage Data Quality with Delta Live Tables • Responsible Generative AI with Databricks
	Data Context Preparation (incl. Labeling)	Databricks provides powerful tools (Spark, notebooks, Delta Lake) for transforming and preparing data context for models. While lacking a native data labeling tool, it integrates with leading labeling partners and platforms via APIs, connectors, or Databricks Marketplace solutions.	• Databricks Data Processing Pipelines • Prepare Data for Machine Learning • Databricks Marketplace
	Featuring Engineering	Databricks provides robust feature engineering capabilities using Spark/Pandas APIs and centrally manages features through the Databricks Feature Store (integrated with Unity Catalog) for creation, storage, discovery, sharing, and ensuring consistency between training and serving.	• What is a Feature Store? • Create features in Unity Catalog • Training Models with Feature Store
	Prompt Engineering	Databricks supports prompt engineering via notebooks for development/testing, the AI Playground (within Foundation Model APIs) for interactive experimentation with models/prompts/parameters, and MLflow for tracking prompts/results. Custom prompt logic can be deployed via Model Serving.	• Databricks Foundation Model APIs • Introduction to Databricks Notebooks • MLflow Tracking • Building LLM Applications with Databricks
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks excels at feature engineering, quality control, data prep, and prompt engineering support. Missing: native labeling, integrated bias mitigation, advanced augmentation, dedicated prompt management, and synthetic data tools.	
Model Engineering	*Experimentation*	Databricks supports ML experimentation through collaborative notebooks, ML Runtimes, and centrally via MLflow Tracking, which automatically logs parameters, metrics, code versions, and artifacts for comparing and reproducing model development efforts. The AI Playground facilitates LLM prompt experimentation.	• Databricks AI Playground for model experimentation • MLflow Tracking • Organize training runs with MLflow Experiments
	Continuous Training	Databricks enables Continuous Training by automating ML training pipelines using Databricks Workflows (Jobs) to orchestrate notebooks or scripts that retrain models on new data. MLflow captures these runs, allowing comparison and promotion of newly trained models via the Model Registry.	• Introduction to Databricks Workflows • MLOps on Databricks • MLflow Guide • Manage Model Lifecycle in Unity Catalog
	…	…	• …

AI Factory | Databricks

AI Backoffice Major Capabilities and Capabilities Mapping – part 2

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
Model Engineering (continuation)	*Continuous Validation*	Databricks supports Continuous Validation (validating pipeline/code integrity) through Databricks Repos for Git integration (CI/CD triggers), Databricks Workflows to run validation jobs (e.g., unit/integration tests within notebooks), and standard testing libraries within the ML runtime.	• Git integration with Databricks Repos • Introduction to Databricks Workflows • Use Databricks Asset Bundles for CI/CD • Unit Testing Databricks Notebooks
	Continuous Evaluation	Databricks enables Continuous Evaluation (evaluating model performance) using MLflow to log evaluation metrics during training/retraining, Model Serving Inference Tables to capture prediction data, and Lakehouse Monitoring to track metrics/drift on inference data over time. Workflows can automate evaluation jobs.	• Monitor Models with Inference Tables • Databricks Lakehouse Monitoring Overview • MLflow Tracking • Evaluate LLMs with MLflow
	Explainability (XAI)	Databricks facilitates model explainability by allowing the integration and execution of standard XAI libraries (like SHAP, LIME) within notebooks. MLflow can store explainability artifacts (e.g., SHAP plots) alongside models. Lakehouse Monitoring can help track feature importance drift.	• SHAP plots for explainability on Databricks • Log models and artifacts with MLflow • Introduction to Databricks Notebooks • Introduction to Databricks Lakehouse Monitoring
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks Model Engineering covers experimentation (MLflow), CT/CV/CE (Workflows, Monitoring), supporting XAI integration. Missing: Deeply integrated XAI, dedicated model debugging/adversarial testing tools, a managed HPO service, and integrated code quality checks.	
Curated AI Assets and Internal Products	*AI Asset Catalog*	Databricks provides an AI Asset Catalog primarily through Unity Catalog, which acts as a centralized governance and discovery layer for all data and AI assets. This includes registering, organizing, and governing ML models (via Model Registry), features (via Feature Store), notebooks, jobs, and the underlying data tables.	• Unity Catalog • Manage model lifecycle in Unity Catalog • Feature Engineering in Unity Catalog • Data Discovery Using Unity Catalog
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks uses Unity Catalog to manage AI assets. Missing capabilities include AI product assembly/certification tools, pipeline-as-product management, product-level value tracking, and internal product templates/blueprints.	

AI Factory | Databricks

AI Backoffice Major Capabilities and Capabilities Mapping – part 3

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Product Monitoring and Continuous Feedback Analysis	*Bias Monitoring*	Databricks supports bias monitoring by using Lakehouse Monitoring to track statistical distributions of sensitive features in model inputs/outputs stored in Inference Tables. Custom bias metric calculations and analysis can be performed using notebooks on this logged data.	• Introduction to Databricks Lakehouse Monitoring • Monitor Models with Inference Tables • Databricks Guardrails for content safety
	Drift Monitoring	Databricks directly addresses drift monitoring through Lakehouse Monitoring, which calculates drift metrics (data and concept drift) between model input/output data (from Inference Tables) and a baseline dataset or previous time windows. Alerts can be configured based on drift thresholds.	• Introduction to Databricks Lakehouse Monitoring • Monitor Models with Inference Tables • Databricks Inference Tables for detecting model drift
	KPI Monitoring	Databricks enables KPI monitoring by allowing users to join model prediction data (from Inference Tables) with downstream business outcome data stored in Delta Lake. KPIs can then be calculated using SQL or notebooks, visualized using Lakeview Dashboards, and monitored via Alerts or scheduled Workflows.	• Databricks MLflow metrics tracking • Databricks Inference Tables for model performance tracking • Dashboards in Databricks (AI/BI)
	Feedback Storage	Databricks serves as a scalable repository for feedback data (e.g., user ratings, corrections) typically collected by external applications. Feedback can be ingested into Delta Lake tables governed by Unity Catalog, allowing it to be easily joined with model predictions (Inference Tables) for analysis, reporting, or retraining cycles.	• Ingest data into a Databricks lakehouse • Introduction to Databricks Notebooks
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks monitors bias/drift (Lakehouse Monitoring), KPIs, and stores feedback. Missing: native feedback acquisition, root cause analysis, automated remediation, structured feedback analysis, and comparative monitoring dashboards.	
Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint		Databricks' AI Backoffice integrates the ML lifecycle (curation, engineering, catalog, monitoring). Missing: Native labeling, advanced Responsible AI/XAI, HITL, feedback acquisition, prompt management, advanced testing/debugging, automated remediation, product assembly tools.	

AI Factory | Databricks

AI Serving Platform Major Capabilities and Capabilities Mapping – part 1

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Infrastructure Services	*AI Services Catalog*	Databricks catalogs AI services primarily through Unity Catalog, listing registered models (via Model Registry) that can be deployed. Deployed services appear as Model Serving Endpoints, and access to curated foundation models is provided via Foundation Model APIs.	• Unity Catalog • Manage model lifecycle in Unity Catalog • Manage model serving endpoints • Foundation Model APIs
	Feature Stores and Vector Databases	Databricks provides both natively: the Databricks Feature Store (integrated with Unity Catalog) for managing ML features, and Databricks Vector Search as a managed, serverless vector database for similarity search and RAG applications.	• What is a Feature Store? • Feature Engineering in Unity Catalog • Databricks Vector Search
	Corporate Models Hub	Databricks acts as a corporate models hub through the MLflow Model Registry within Unity Catalog, providing a central, governed repository for discovering, versioning, staging, and managing the lifecycle of an organization's machine learning models.	• Manage model lifecycle in Unity Catalog • Unity Catalog • MLflow Models • Data-centric MLOps and LLMOps
	Gateway to Third-party Models	Databricks provides access to third-party models via Foundation Model APIs (pay-per-token access to curated external models) and External Models within Model Serving (routing requests to external provider endpoints like Azure OpenAI, Anthropic, etc.).	• External Models in Databricks Model Serving • Foundation Model APIs • Query foundation models
	Inference Engines	Databricks provides optimized inference engines through Databricks Model Serving, offering scalable, low-latency hosting for ML models (MLflow, Python) and LLMs, with features like serverless compute, GPU support, and optimized LLM serving capabilities.	• Introduction to Databricks Model Serving • Databricks Optimized LLM Serving • GPU workload types • Serverless compute for notebooks
	Deployment Orchestration	Deployment orchestration is managed via the Model Serving API/UI for endpoint creation/updates, MLflow Model Registry for promoting models to deployment stages, and Databricks Workflows or external CI/CD tools (using Databricks CLI/API or Asset Bundles) for automating deployment pipelines.	• Create and manage model serving endpoints • CI/CD with Model Serving • Introduction to Databricks Workflows • Use Databricks Asset Bundles for CI/CD • Promote a model using the Model Registry UI
	…	…	• …

AI Factory | Databricks

AI Serving Platform Major Capabilities and Capabilities Mapping – part 2

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Infrastructure Services (continuation)	*Compute Resource Management*	Databricks manages compute resources through configurable Clusters (interactive/job, standard/high-concurrency, CPU/GPU, autoscaling policies), Serverless Compute options (for SQL, Notebooks, Model Serving), and Pools to manage instance availability and startup times.	• Databricks Compute resource management • Compute configuration reference • Serverless compute • Pool configuration reference
	FinOps	Databricks supports FinOps through detailed Usage System Tables for cost tracking/attribution, Cluster/Pool/Job Tags for cost allocation, Cluster Policies for cost control, Autoscaling features, Spot Instance options, and the predictable pricing models of Serverless Compute and Foundation Model APIs.	• Monitor usage using system tables • Monitor costs using tags • Optimize Databricks costs • Cluster policies
	Feedback Acquisition	Databricks does not provide native tools for acquiring user feedback (e.g., UI elements). It excels at storing (Delta Lake), processing (Spark, SQL), and analyzing feedback data once it has been collected by external applications and ingested into the platform.	• Ingest data into a Databricks lakehouse • Analyze inference tables
	Human in the Loop (HITL)	Databricks lacks native, built-in HITL workflow capabilities. However, it provides the infrastructure to build HITL systems: storing data requiring review (Delta Lake), triggering external review workflows (via Workflows, Webhooks, or API calls from notebooks), and ingesting human decisions back for analysis or retraining.	• Introduction to Databricks Workflows • Add notifications on a job • Introduction to Databricks Notebooks
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks AI infrastructure provides compute, storage, Feature Store/Vector DB, Model Hub, inference, orchestration, FinOps. Lacks native feedback acquisition/HITL, advanced deployment strategies, service mesh, edge support, and integrated A/B testing.	
Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint		-//-	

AI Factory | Databricks

AI Serving Platform Major Capabilities and Capabilities Mapping – part 1

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Services	*AI Integrations*	Databricks facilitates AI integrations by connecting its services internally (e.g., Model Serving accessing Feature Store), enabling calls to external AI providers (via External Models/Foundation Model APIs), integrating with BI tools for consuming insights, connecting to code repositories (Repos), and allowing custom integrations via its APIs/SDKs within notebooks and jobs.	• External Models in Databricks Model Serving • BI and Visualization Integrations • Use online tables for real-time model serving • Databricks REST API reference
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks integrates internal/external AI services, BI, repos via APIs. Missing: Dedicated AI event bus/workflow engine, integration monitoring, app connectors, security propagation, advanced integration gateway features.	
GenAI Services	Inference Interfaces	Databricks provides inference interfaces for GenAI models via Model Serving (hosting custom/fine-tuned/open-source LLMs as REST APIs), Foundation Model APIs (pay-per-token access to external models with a REST API and interactive AI Playground), and Notebooks for direct SDK/API calls during development.	• Databricks Model Serving • Databricks LLM-optimized serving interfaces • Foundation Model APIs • Query foundation models
	RAG	Databricks enables building RAG applications by providing integrated Vector Search (managed vector database), tools for data preparation/chunking/embedding (notebooks, Spark), and Model Serving or Foundation Model APIs to orchestrate the retrieval and generation steps.	• RAG on Databricks • Databricks Vector Search • Building High-Quality RAG Applications
	Guardrails	Databricks supports implementing GenAI guardrails through techniques like RAG (grounding responses), prompt engineering, output monitoring/filtering using Lakehouse Monitoring/Inference Tables, or by integrating external guardrail services via External Models or custom logic within Model Serving. It lacks a dedicated, built-in "Guardrails" service.	• Responsible Generative AI with Databricks • External Models in Databricks Model Serving • Introduction to Databricks Lakehouse Monitoring • RAG on Databricks
	Hallucination Management	Databricks helps manage hallucinations primarily by enabling RAG applications (grounding responses in factual data via Vector Search), providing LLM Evaluation tools (e.g., mlflow.evaluate) to assess metrics like groundedness/faithfulness, and supporting Monitoring of outputs to detect potential deviations.	• Evaluate LLMs with MLflow • RAG on Databricks • Databricks Vector Search • Introduction to Databricks Lakehouse Monitoring
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks provides GenAI inference, RAG (Vector Search), supporting guardrails/hallucination management. Lacks dedicated guardrails, managed fine-tuning, prompt management UI, advanced LLM evaluation, token optimization, or structured output validation.	

AI Factory | Databricks

AI Serving Platform Major Capabilities and Capabilities Mapping – part 2

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Consumption Management and Governance	*Unified API*	Databricks offers unified REST APIs for interacting with AI services via Model Serving (for custom/open-source models) and Foundation Model APIs (for pay-per-token models), providing consistent interfaces for invoking inference endpoints. Unity Catalog APIs govern access centrally.	• Query serving endpoints for custom models • Query foundation models • Databricks REST API reference
	LLM/Agents Communications Routing and Balancing	Model Serving automatically load balances requests across endpoint replicas. Routing between different models, versions (e.g., for A/B tests), or to external providers (via External Models) is typically configured within the endpoint settings or handled by upstream application logic calling specific endpoints.	• Create and manage model serving endpoints • External Models in Databricks Model Serving • Introduction to Databricks Model Serving
	Conversational History	Databricks does not provide a built-in service specifically for managing conversational history. Applications built on Databricks typically store this history in Delta Lake tables, allowing it to be queried, managed, and used as context for subsequent LLM calls within the application logic.	• Delta Lake • What is Databricks SQL? • RAG on Databricks
	Cost Management and Rate Control	Cost management for AI services is supported via Usage System Tables, Tags, and monitoring endpoint metrics. Rate limits can be configured on Model Serving endpoints. Consumption of Foundation Model APIs is tracked for billing based on tokens.	• Monitor usage using system tables • Databricks LLM-optimized serving for efficient inference • Create and manage model serving endpoints • Monitor costs using tags • Foundation Model APIs pricing
	Consumer Multi-tenancy Support	Multi-tenancy is supported through Unity Catalog (fine-grained ACLs on models, endpoints, data), workspace separation, Service Principals for application access, and cost attribution via Tags and Usage System Tables. Different teams can securely consume shared or dedicated AI resources.	• Unity Catalog • Manage privileges in Unity Catalog • Manage permissions on your model serving endpoint
	Invocation Logging	Model Serving automatically logs inference requests and responses to Inference Tables (Delta tables) for auditing and analysis. Platform API calls, including endpoint management, are logged in Audit Logs. Foundation Model API usage is tracked for billing and may appear in audit logs.	• Monitor Models with Inference Tables • Audit log reference • Analyze inference tables
	…	…	• …

AI Factory | Databricks

AI Serving Platform Major Capabilities and Capabilities Mapping – part 3

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
AI Consumption Management and Governance (continuation)	*Chat / Prompt Sharing and Collaboration*	Prompt sharing and collaboration primarily occur through Databricks Notebooks and Databricks Repos (Git integration). MLflow can track prompts associated with experiments. The AI Playground allows interactive prompt testing but isn't designed for persistent sharing of complex prompts.	• Introduction to Databricks Notebooks • Git integration with Databricks Repos • MLflow Tracking • Foundation Model APIs
	Security and User Access Control	Security for AI consumption is managed centrally via Unity Catalog (ACLs on models, data, features, functions), specific permissions on Model Serving endpoints, workspace-level access controls, and authentication methods (SSO, PATs, OAuth, Service Principals).	• Unity Catalog • Manage privileges in Unity Catalog • Manage permissions on your model serving endpoint • Access control overview • Security and compliance on Databricks
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks manages AI consumption via unified APIs, basic routing, cost/rate control, logging, security. Lacks native conversational history, dedicated prompt management, advanced routing, granular quotas, consumer API keys, governance workflows.	
Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint		Databricks' AI Serving Platform has unified interfaces, RAG, integrations, governance. Missing: native guardrails, conversational history, prompt management, advanced routing/evaluation/fine-tuning, granular quotas, dedicated integration engine, and integrated A/B testing.	

AI Factory | Databricks

AI Experience Major Capabilities and Capabilities Mapping

EPAM VISION		VISION	
MAJOR CAPABILITY	**CAPABILITY**	**ANALYSIS**	**SOURCES (LINKS)**
Generalized Applications and Self-serviced Experience	*Built-in Chat*	Databricks does not offer a general-purpose, built-in chat application for end-users. However, it provides the Databricks Assistant for AI-powered chat assistance within notebooks and editors for developers/analysts. It provides the backend infrastructure (Model Serving, RAG tools) to build custom chat applications.	• Databricks Assistant • Introduction to Databricks Model Serving • RAG on Databricks
	Customizable Agents	Databricks enables the development of customizable agents (LLMs combined with tools/functions) using notebooks for orchestration logic, Model Serving or Foundation Model APIs for LLM access, and integrating custom Python code or external API calls as tools. MLflow can track agent experiments.	• Introduction to Databricks Notebooks • Foundation Model APIs • External Models in Databricks Model Serving • Large Language Models (LLMs) on Databricks
	Mini-RAG	Databricks supports building RAG applications of any scale. Simple or "mini-RAG" implementations can be quickly prototyped using Vector Search, notebooks for embedding/querying, and Foundation Model APIs, often demonstrated in Solution Accelerators or documentation examples focusing on specific datasets.	• RAG on Databricks • Databricks Vector Search • Solution Accelerators (may contain examples)
	GPTs and Tools	Databricks allows combining LLMs (like GPT models via Foundation Model APIs or External Models, or other hosted models) with custom tools (Python functions, API calls implemented in notebooks or deployed code). Orchestration logic in notebooks or custom applications coordinates the interaction between the LLM and tools.	• Foundation Model APIs • External Models in Databricks Model Serving • Introduction to Databricks Notebooks • LangChain on Databricks
	Visualization and Branding	AI-derived insights can be visualized using Lakeview Dashboards within Databricks or by connecting external BI tools (Tableau, Power BI, etc.). Branding is typically applied within these dashboarding/BI tools or in custom web applications built consuming Databricks APIs, not natively within Databricks itself.	• Dashboards in Databricks (AI/BI) • BI and Visualization Integrations • Introduction to Databricks Model Serving
	Major capability summary and capabilities missing in AI Factory Databricks Blueprint	Databricks enables building AI experiences (Assistant, custom apps, RAG, viz). Missing: Pre-built apps, no-code builder, native end-user chat, workflow integration, deep branding, agent marketplace, and integrated app feedback.	
Sub-platform summary and major capabilities missing in AI Factory Databricks Blueprint		-//-	