Scale at Speed™

Whitepaper

Inference-first Platforms

Making Enterprise AI Economically Viable at Scale

Authors

Dr. Anshu Premchand

Sukumar Shanmugam

Executive Summary

Enterprise AI is at a critical juncture. It is no longer limited by model capabilities but rather by the economics and operational discipline of execution. As GenAI adoption scales from pilots to agentic and low-volume to high-volume workloads, enterprises are running into an 'inference wall.' Cost-per-token, latency, and reliability constraints are accelerating faster than the business value GenAI programs are delivering. Without deliberate architectural redesign, AI programs risk becoming financially unsustainable at scale.

This paper presents a case for an inference-first platform strategy built on compound AI patterns, including model routing, retrieval-augmented generation (RAG), and automated guardrails. By engineering for inference economics from the outset, the paper highlights how enterprises can ensure that AI remains powerful, trustworthy, and financially viable as adoption accelerates.

Table of Contents

Inference-first Platforms -
Making Enterprise AI
Economically Viable at Scale

Introduction 1

Strategic Design and Engineering Patterns for Inference-first Systems 1

Training Phase
Inference Phase

Key Features of AI Inference Systems 3

The Architectural Build of an Inference-first Platform 4

Economics of Intelligence 8

Primary Levers to Reduce Cost Per Inference (CPI) 8

Managed Inference, MLOps, and the Data Flywheel 9

Managed Inference
LLMOps (GenAI-specific MLOps)
The Inference Data Flywheel

Governance, Sovereignty, and Risk Management 11

Governance,
Sovereignty, and Risk Management

Implementation Roadmap 13

Phase 1: Setting Up RAG Foundation (0-3 Months)
Phase 2: Working on Cost Optimization (3-6 Months)
Phase 3: Scale for the Enterprise (6+ Months)

Future Outlook 15

Conclusion 16

Introduction

The advent of generative AI has unlocked significant transformational potential for enterprises. From intelligent automation and decision augmentation to customer engagement and knowledge discovery, GenAI is reshaping how organizations operate and compete. Early pilots have demonstrated productivity gains, improved customer experiences, and new pathways for innovation across functions, including marketing, customer service, finance, operations, and R&D.

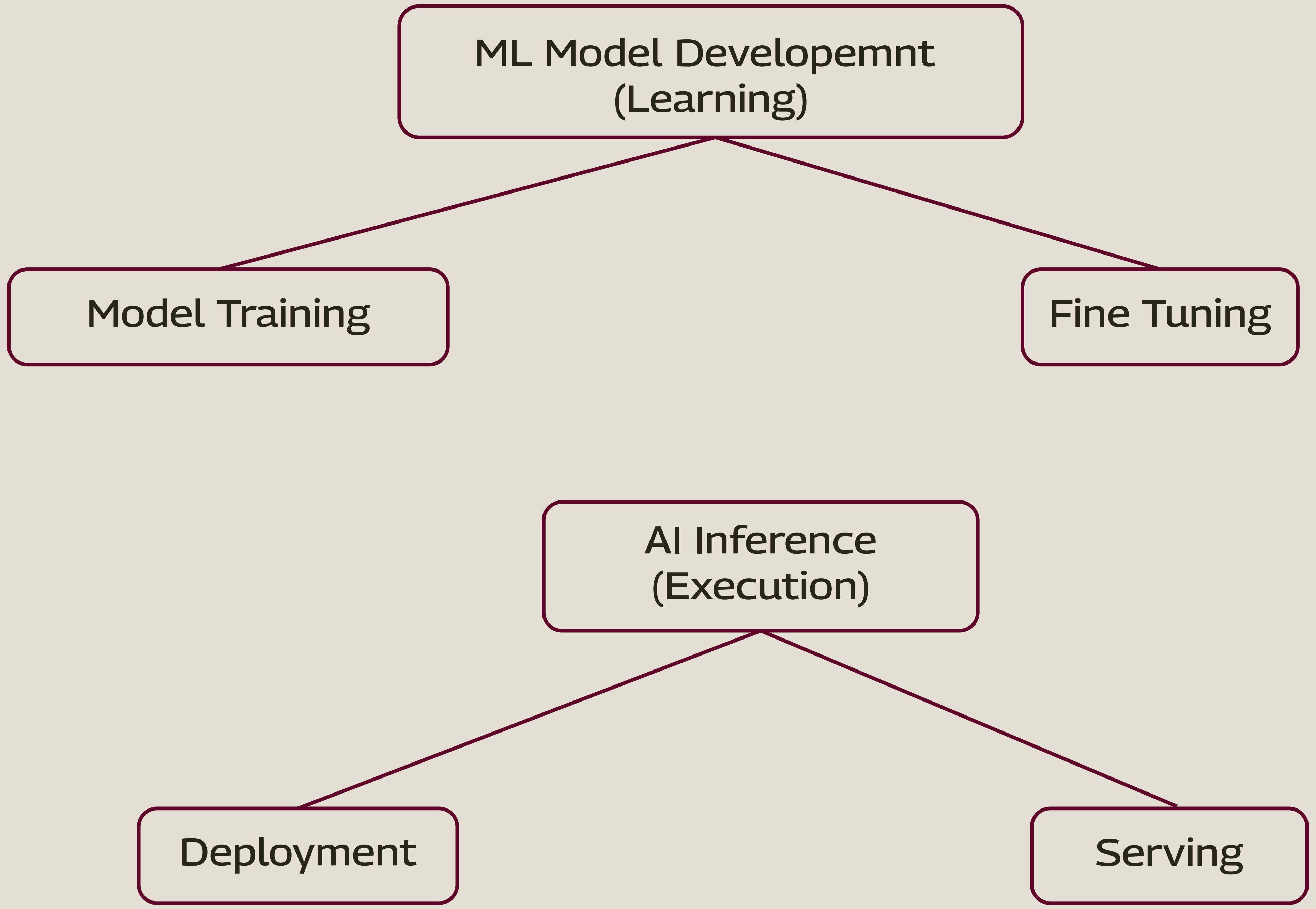
However, the transition from experimentation to enterprise-wide deployment introduces a new class of implementation and cost constraints, including compute and infrastructure maintenance, legacy system integrations, data quality and governance, risk management, and ROI demonstrations. The looming 'inference wall' unites these challenges, demanding critical intervention.

An inference-first AI platform is a purpose-built, cloud-native solution stack designed exclusively to tackle this challenge. Rather than optimizing for model training, it is designed to optimize hardware utilization, reduce operational overhead for real-time model serving, and enhance the cost-per-inference (CPI). Transitioning to this discipline-driven inference model aids enterprises in making economically sustainable and operationally resilient decisions.

Strategic Design and Engineering Patterns for Inference-first Systems

The life cycle of AI development involves training the model, deploying it, and enabling it to serve different real-world application use cases. It includes:

AI Life Cycle



Strategic Design and Engineering Patterns for Inference-first Systems

Training Phase:

Model Training

Operating as a foundational learning phase for building an ML model, this stage uses a developed model to analyze the data and learn from it based on its patterns and relationships. Model training requires powerful hardware like GPUs, TPUs, and massive datasets.

Model Fine-Tuning

The next step in the process involves using the pre-trained model and calibrating its key parameters to adapt for a specific task or domain with a smaller and specialized dataset.

Inference Phase:

Deployment

In the execution phase, the trained and fine-tuned models are used to handle real-world application requests. This process prioritizes specific requests from users or applications and applies the trained model to analyze the data and deliver an inference.

Serving

This phase is the end-user-facing stage with an API endpoint and often involves packing the model with required infrastructure to handle millions of requests.

To put it succinctly, AI inference is the process of applying a trained model to generate an output (i.e., predicting, generating, decision-making, etc.). It is categorized as:

Cloud Inference

Cloud-based AI inference offers hardware power and computing scale and is ideal for handling huge datasets and complex ML models. It can handle both API-based real-time inference requests as well as batch processing use cases.

Edge Inference

Running directly on the devices where data is generated, such as smartphones or sensors, edge inference processes data locally. Handling computations on the device, it reduces latency, lowers bandwidth consumption, and improves data privacy while enabling a faster and more responsive user experience.

While a single inference request is quick, the real challenge arises at scale. Such an endeavor requires an inference-first platform designed for low latency, high throughput, and optimized use of hardware and cloud infrastructure.

Key Features of AI Inference Systems

Inference-first AI platforms are built to deploy, run, and manage trained AI models for enterprises. They are equipped with internet-scale computing and data infrastructure to run pre-trained AI models efficiently, quickly, and at low cost. An AI inference system facilitates:

Model Optimization

The process of fine-tuning AI/ML models for accuracy and speed.

Hardware Acceleration

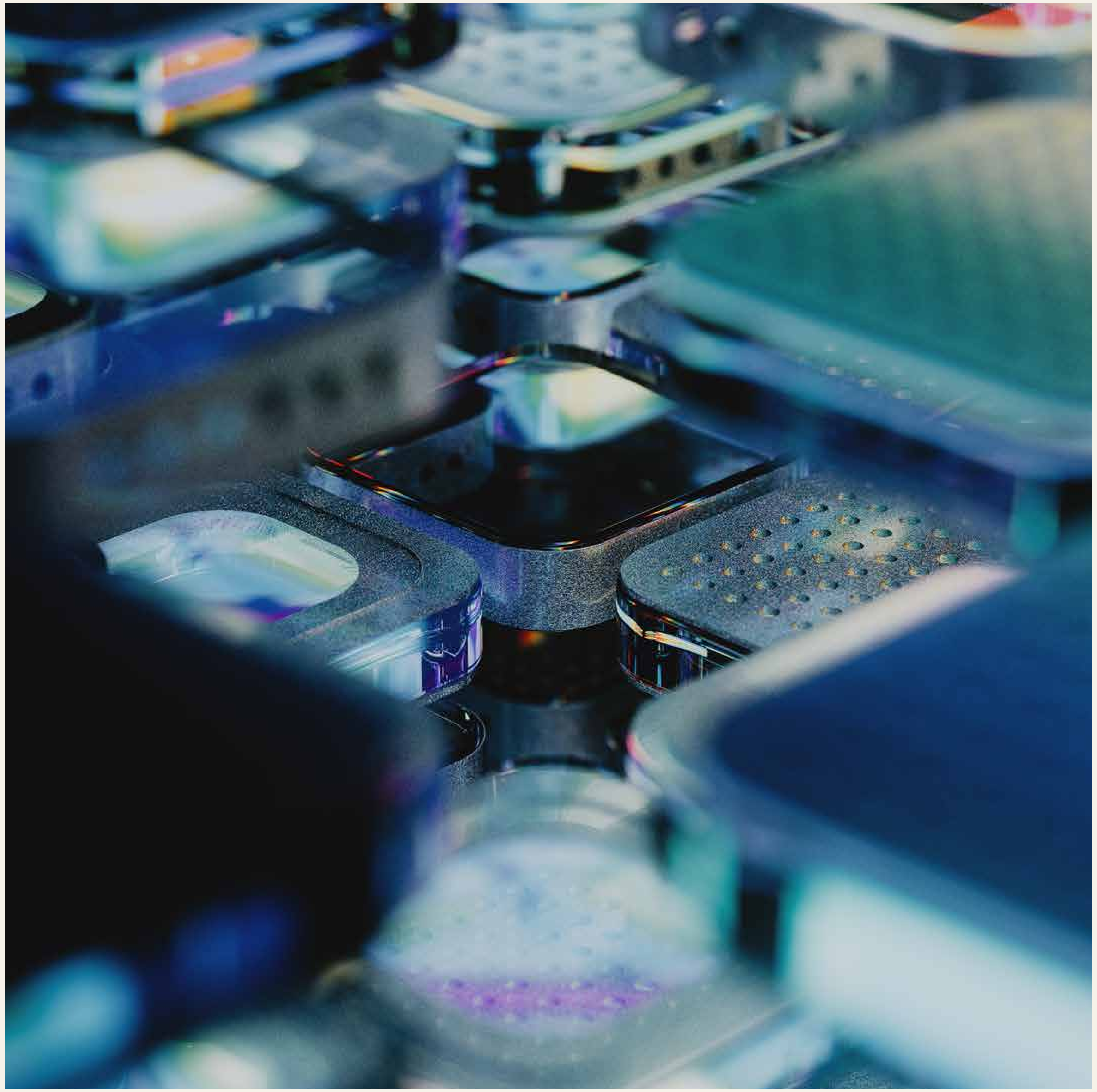
Utilization of specialized hardware like GPUs and AI accelerators (FPGAs) for specific tasks like video processing, generation, rendering, and high-demand AI computing workloads.

Low Latency Processing

Minimizing delays caused by I/O data transfer and network throughput constraints.

Platform Integration

Seamless integration with existing applications through APIs and supporting tools to enable easier implementation and scalable deployment.



The Architectural Build of an Inference-first Platform

A modular, inference-first architecture bridges the gap between AI models and enterprise users. The layers outlined below form a compound AI stack engineered for production-scale deployment.

Ingress and Identity Layer

This layer acts as platform's secure entry point. It manages multi-tenancy through robust authentication mechanisms like OIDC and SSO. The ingress and identity layer ensures stability and compliance by enforcing rate limits and quotas. It also maintains comprehensive audit logging for all incoming requests.

logging for all incoming requests.

Gateway and Router Layer

Acting as the primary layer, it authenticates requests and enforces rate limits and tenancy by routing prompts to the most cost-effective model.

Example: a smaller model could be used for routine tasks, whereas a frontier model could be leveraged for complex reasoning.

Context and Retrieval Layer (RAG)

Next, the RAG layer integrates vector databases and search capabilities with permission-aware retrieval to provide real-time domain context. Grounding responses in enterprise data, it reduces the need for repeated fine-tuning whenever business knowledge changes.

Inference Engine

This layer is optimized for serving with techniques such as continuous batching, KV-cache reuse, and hardware-aware scheduling, etc., to maximize throughput and minimize latency.

Guardrail Layer

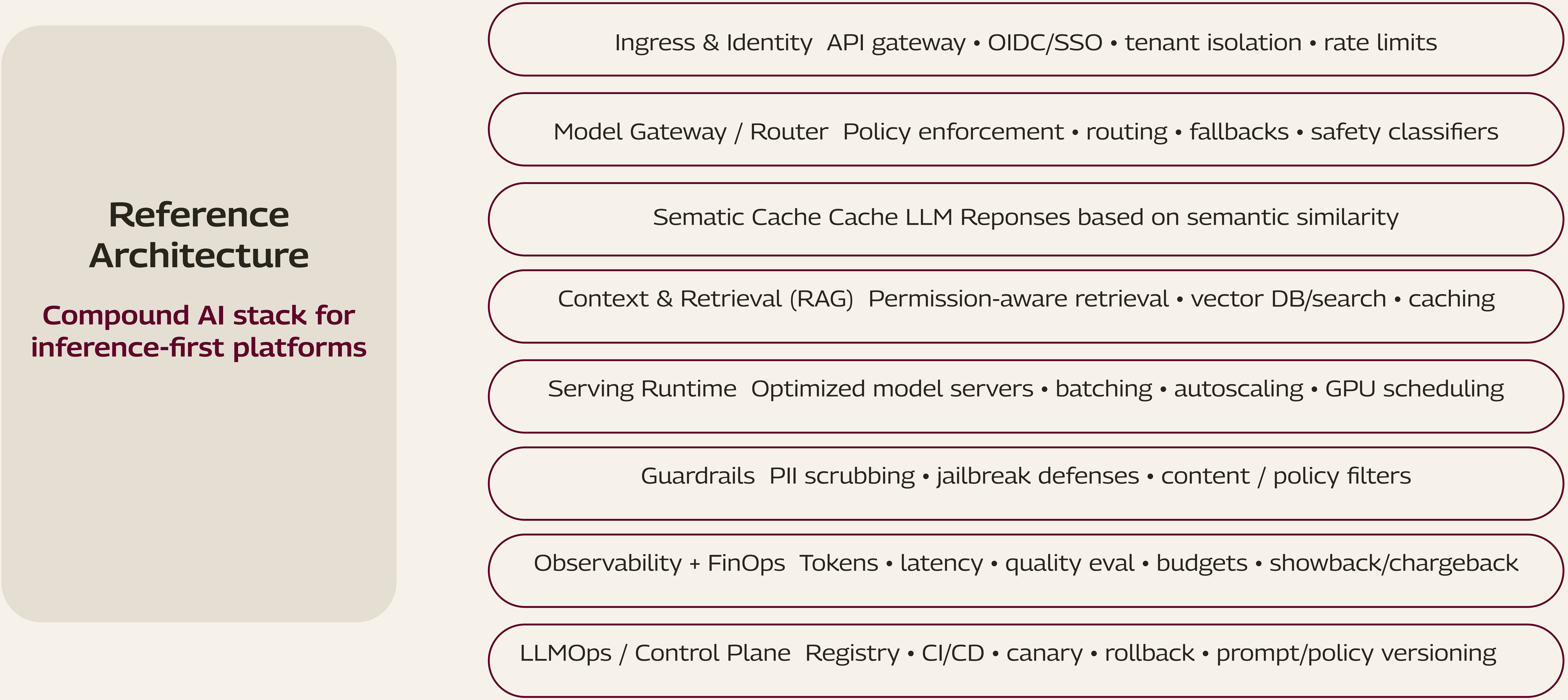
Focusing on risk management, this layer provides real-time protection for personally identifiable information (PII), policy violations, bias, compliance, and adversarial inputs before responses reach the requesting applications or agents.

Observability and Control Plane

Functioning as the insight and control layer, this plane manages metrics, traces, logs, quality evaluations, release management (canary/rollback), and cost governance.

The Architectural Build of an Inference-first Platform

Figure 2: Architectural Framework of an Inference-first Platform



Reference Architecture

Compound AI stack for inference-first platforms

Ingress & Identity API gateway • OIDC/SSO • tenant isolation • rate limits

Model Gateway / Router Policy enforcement • routing • fallbacks • safety classifiers

Sematic Cache Cache LLM Reponses based on semantic similarity

Context & Retrieval (RAG) Permission-aware retrieval • vector DB/search • caching

Serving Runtime Optimized model servers • batching • autoscaling • GPU scheduling

Guardrails PII scrubbing • jailbreak defenses • content / policy filters

Observability + FinOps Tokens • latency • quality eval • budgets • showback/chargeback

LLMOps / Control Plane Registry • CI/CD • canary • rollback • prompt/policy versioning

The Architectural Build of an Inference-first Platform

This table presents each layer's key responsibility and capabilities.

Layer	Primary Responsibility	Typical Capabilities
Ingress and Identity	• Secure entry point • Tenancy	• API gateway • OIDC/SSO • Rate limits • Quotas • Audit logging
Model Gateway and Router	• Best-fit model selection • policy enforcement	• Routing • Fallbacks • Safety classifiers • Prompt sanitation
Context and Retrieval	• Grounding with enterprise knowledge	• Vector DB and/or search • Permission checks • Caching • Document lineage
Serving Runtime	• Low-latency execution	• Optimized model servers • Batching • Autoscaling • GPU scheduling

The Architectural Build of an Inference-first Platform

This table presents each layer's key responsibility and capabilities.

Layer	Primary Responsibility	Typical Capabilities
Guardrails	• Trust • Safety controls	• PII scrubbing • Content filters • Jailbreak • Prompt injection defenses
Observability	• Reliability • Quality visibility	• Latency and throughput • Token usage • Eval metrics • Incident workflows
FinOps	• Unit economics control	• Cost-per-token dashboards • Budgets • Showback • Chargeback
LLMOps / MLOps	• Repeatable delivery • Governance	• Registry • CI/CD • Eval gates • Canary releases • Rollback

Economics of Intelligence

As enterprises adopt AI models and expand in usage, economic consideration becomes crucial. Central to the execution phase, AI inference will lead operations and capital management. Though inference costs have declined recently due to advancements in model optimization and more energy-efficient hardware, economic discipline remains essential as usage volumes grow.

Three economic metrics shape this discipline:

- Tokens reflect how much processing a request consumes.
- Throughput indicates how many requests a system can handle over a period of time.
- Latency measures how quickly responses are delivered.

Based on these metrics, enterprises can measure economic cost and work towards optimizing inference deployment strategies to achieve the desired commercial outcome. Additionally, enterprises can also implement various operational and architectural optimizations, such as caching, batching, and hybrid on-prem/cloud model usage, to manage inference cost.

Primary Levers to Reduce Cost Per Inference (CPI)

Cost per inference can be lowered through:

- **Model Routing and Cascading:** Reserves large models for complex cases and routes smaller ones for routine tasks.
- **Prompt and Context Discipline:** Minimizes unnecessary context length using selective retrieval.
- **Caching:** Uses response caching and retrieval caching for repetitive and stable queries.
- **Batching and Scheduling:** Increasing concurrency and throughput without breaking latency SLOs.
- **Optimization:** Applies techniques such as quantization or model compilation, along with hardware-aware serving configurations.

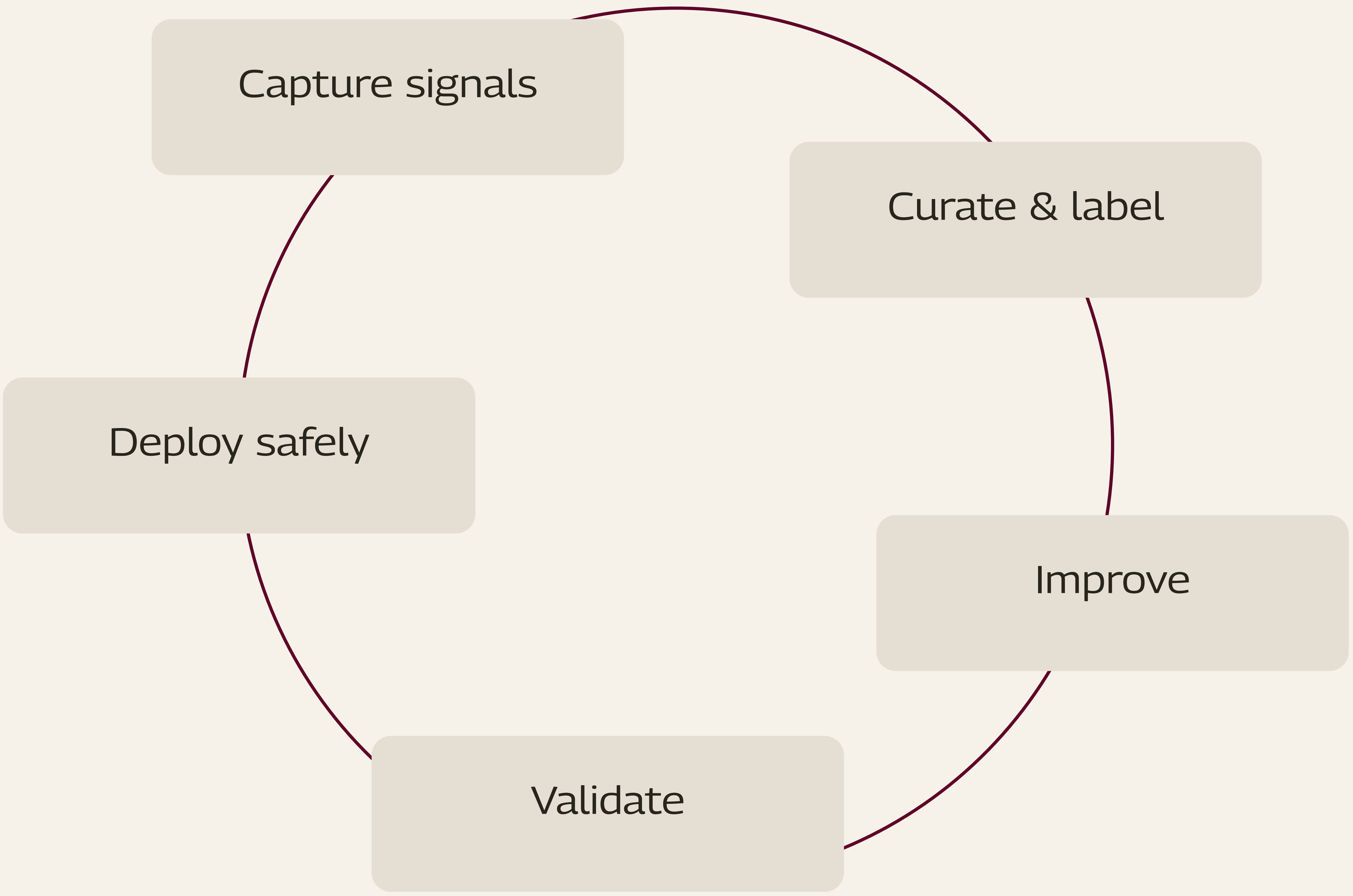
Managed Inference, MLOps, and the Data Flywheel

Enterprises need a closed-loop operating model to prevent model stagnation and uncontrolled spending. In this model, managed inference, LLMOps practices, and a data flywheel reinforce one another.

Figure 3: Closed-Loop Inference Operating Model

Managed Inference + LLMOps + Data Flywheel

Compounding value while controlling CPI



Managed Inference, MLOps, and the Data Flywheel

Managed Inference

- Runtime must serve as an SRE-grade service. It should enable standardized deployments, autoscaling, high availability, and incident response.
- Capacity pooling and multi-tenancy must be adopted with strong isolation to improve utilization.
- Cost controls must be implemented. These controls include quotas, budgets, and showback/chargeback by team or business unit.

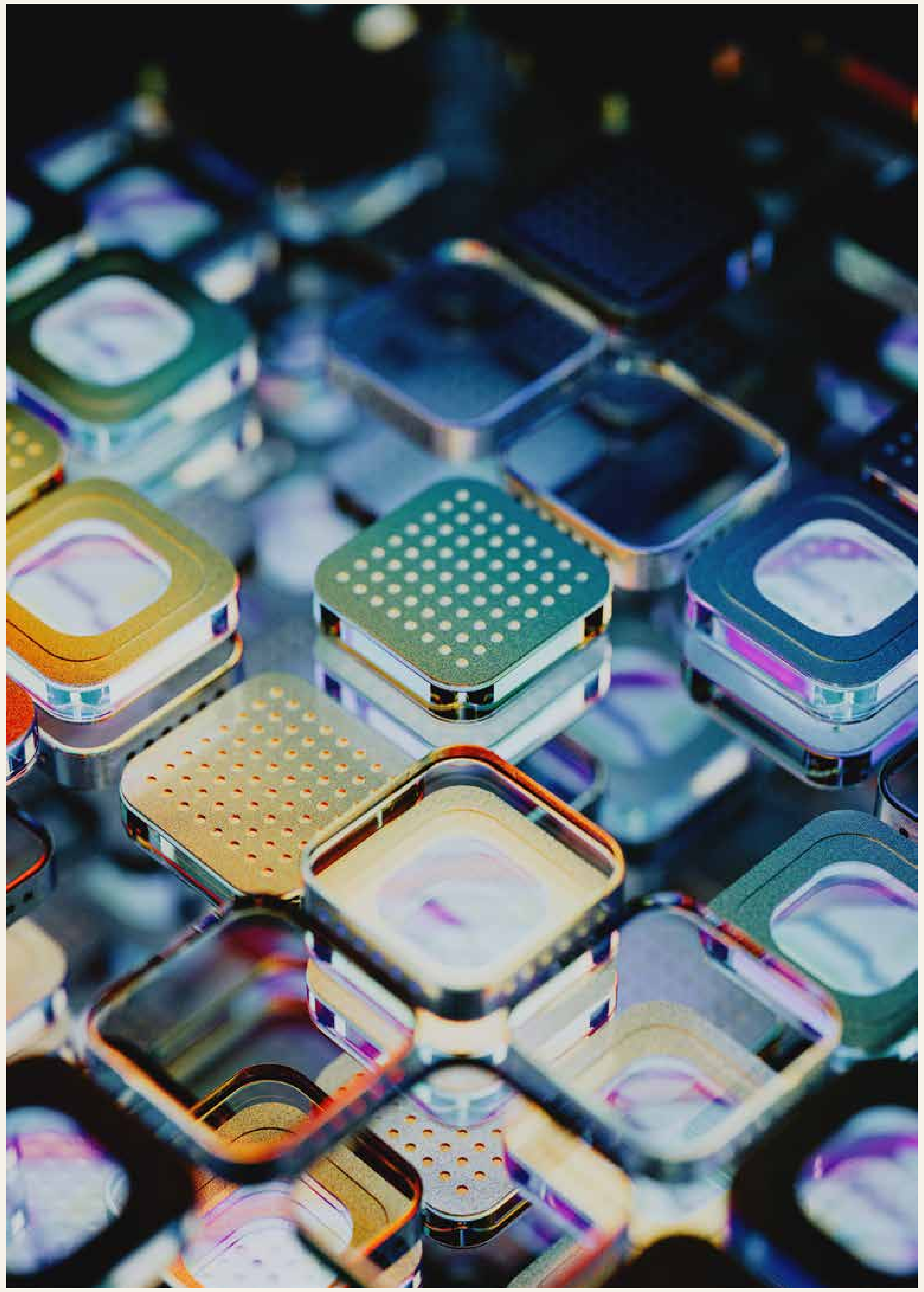
LLMOps (GenAI-specific MLOps)

- Inclusion of version and governance prompts, policies, adapters, and serving configuration along with the base model.
- Establishment of evaluation harnesses that include regression tests, adversarial prompts, quality thresholds for releases, etc.
- Maintenance of observability with redaction that captures token usage, latency, and failure modes while protecting sensitive data.

The Inference Data Flywheel

1. Capture Signals: Capture user feedback, corrections, tool outcomes, and escalation reasons.
2. Curate and Label: Convert production traces into high-quality datasets enabled with privacy controls.
3. Improve: Refine retrieval indexes, prompts, and routing policies for value derivation.
4. Validate: Always run evaluations, red-team tests and measure both quality and unit economics.
5. Deploy Safely: Do canary releases, monitor drift, and iterate.

Over time, this flywheel will enable the distillation of knowledge from larger models into smaller, cheaper, and specialized ones. This will automatically increase capability while lowering CPI.



Governance, Sovereignty, and Risk Management

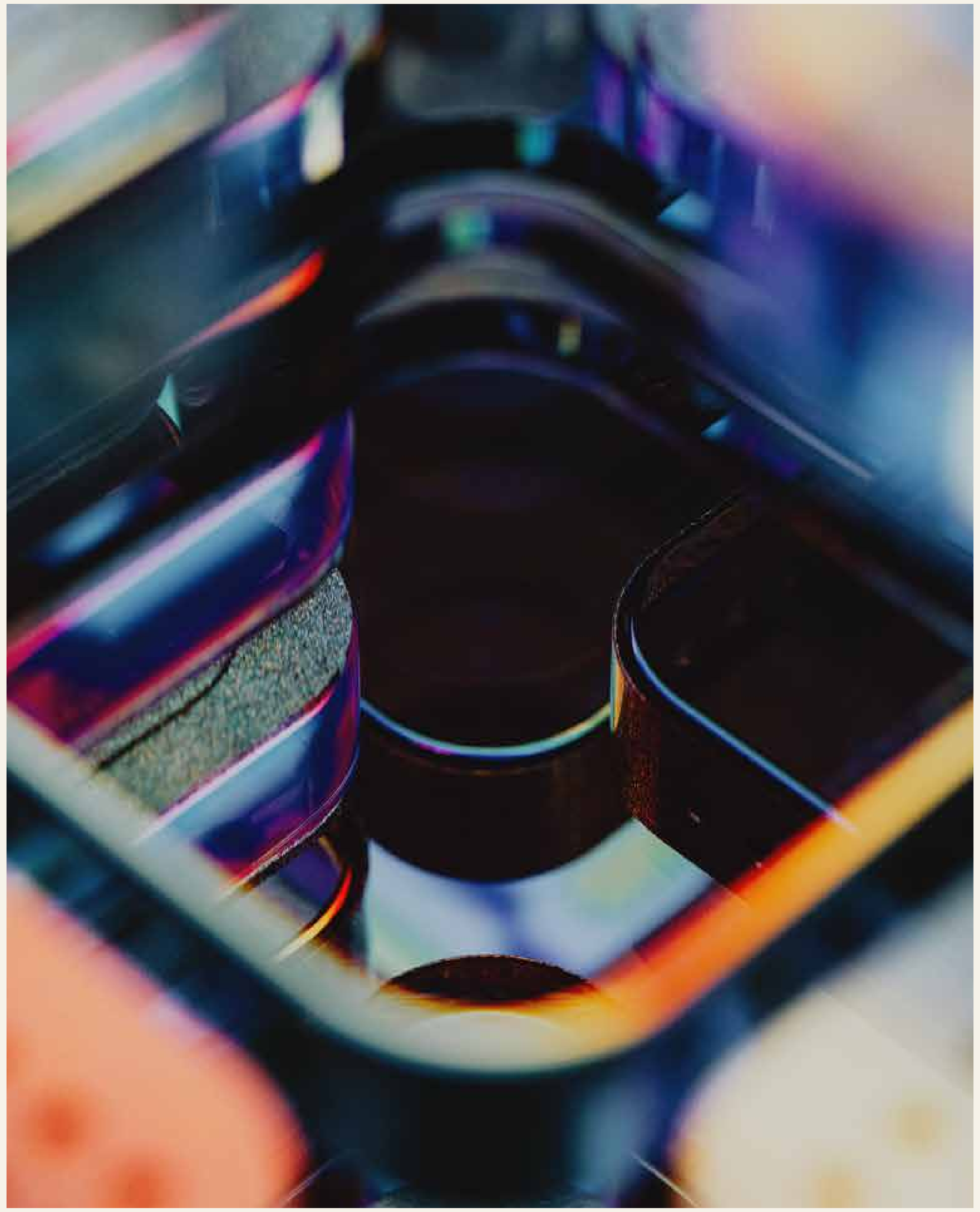
Once inference-first AI systems move into production, managing risk and enforcing policy become ongoing responsibilities.

Governance

AI governance in inference systems becomes critical once it begins operating at scale. Enterprises must enforce clear boundaries around policies, frameworks, and controls to ensure AI models operate ethically, safely, and legally. This process includes actively managing risks such as bias, lack of transparency, and hallucinations through configuration controls, continuous monitoring, and structured oversight.

Sovereignty and Risk Management

Organizations must address both jurisdictional control and operational risk, ensuring that data residency, cross-border processing and regulatory obligations are respected. This includes region-specific deployment, policy-based routing and controlled data movement aligned with regulatory mandates. They must also identify, assess, mitigate and continuously monitor potential negative impacts that may arise when using AI models in live operations.



Governance, Sovereignty, and Risk Management

Figure 4: Inference Risk and Mitigation Framework

Sovereignty cannot not an afterthought. We need to build "deploy anywhere" and policy-based routing into the platform from day one.

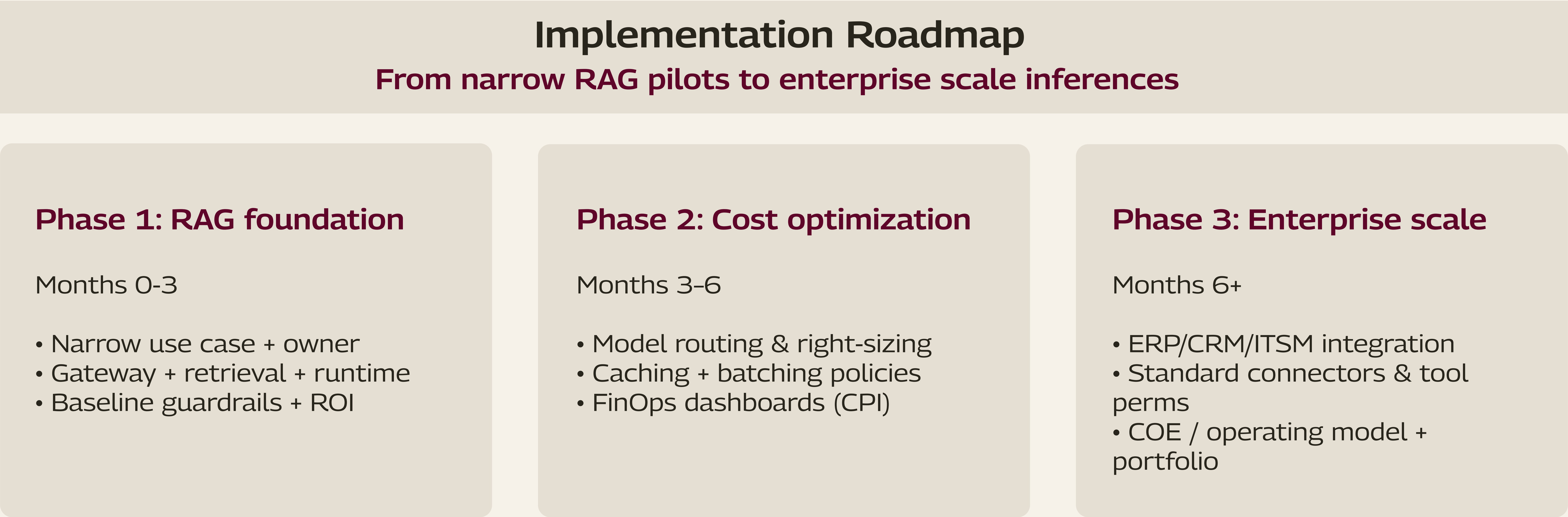
Risk category	Primary threat	Representative technical mitigation
Model risk	Hallucinations / drift / adversarial prompts	RAG grounding checks • continuous eval • canary + rollback
Data risk	PII leakage / confidential data exposure	PII scrubbing at gateway • permission-aware retrieval • audit trails
Ethical risk	Algorithmic bias / unsafe outputs	Adversarial testing • policy filters • human review for high risk workflows
Sovereignty	Jurisdictional compliance / residency	Regional cloud or on-prem inference • policy routing • BYOK/HSM

Risks in AI inference fall into the following categories:

- Model Risks: Adversarial attacks, model drift, and hallucinations are potential risks that enterprises encounter in real-world operations.
- Data Risks: AI systems handling personal and sensitive information during processing can be vulnerable to breaches and need strong protection.
- Operational Risks: Challenges regarding integration and legacy applications lead to malicious attacks and cyber threats.
- Ethical and Legal Risks: Algorithmic bias due to training data can result in discriminatory output and needs to be reviewed and controlled. Failure to adhere to emerging AI-specific regulations can result in non-compliance with significant fines and legal penalties.

Implementation Roadmap

Building an inference platform that offers high-value AI use cases while strengthening governance and unit economics requires a structured roadmap. The following outlines a three-phase approach.



Implementation Roadmap

From narrow RAG pilots to enterprise scale inferences

Phase 1: RAG foundation

Months 0-3

- Narrow use case + owner
- Gateway + retrieval + runtime
- Baseline guardrails + ROI

Phase 2: Cost optimization

Months 3-6

- Model routing & right-sizing
- Caching + batching policies
- FinOps dashboards (CPI)

Phase 3: Enterprise scale

Months 6+

- ERP/CRM/ITSM integration
- Standard connectors & tool perms
- COE / operating model + portfolio

Figure 5: Three-phased Implementation Roadmap

Implementation Roadmap

Phase 1: Setting Up RAG Foundation (0-3 Months).

- Deploy 1-2 narrow and high-confidence use cases with retrieval augmented generation
- Establishment of security, access controls, unit economics, and baseline ROI
- Implementation of the minimum platform
- Setting up the gateway, retrieval, serving runtime, observability, and basic guardrails

Phase 2: Working on Cost Optimization (3-6 Months)

- Introduction of model routing and right-sizing
- Transitioning common tasks to smaller specialized models, as appropriate
- Optimization of inference with caching and batching policies
- Setting up operational FinOps dashboards to reduce CPI

Phase 3: Scale for the Enterprise (6+ Months)

- Integration with internal systems such as ERP/CRM/ITSM
- Standardization of connectors
- Setting up tool permissions for agentic use cases
- Establishment of an AI Center of Excellence (CoE or equivalent operating model)
- Setting up governance gates, evaluation discipline, and portfolio management

Future Outlook

Inference-first architecture will power the next phase of enterprise AI, characterized by scale, economic control, and operational maturity. The following are the expected developments:

Omnipresent Inferences are the Future

Unit costs will continue to fall. At the same time, orchestration will emerge as the differentiator. The choice of model, the timing of its use, and the context and controls applied to it will become critical.

Right-sizing Models Will Become a Standard Practice

Smaller models will handle the majority of routine tasks, while frontier models will be prioritized for complex, high-risk workflows.

Agentic Workloads Will Reshape Capacity Planning

Tool calls, retries, and long-running tasks will demand stronger runtime control, and effective cost governance will be essential.

Evaluation Will be a Top Priority

Continuous safety and quality testing will reflect software regression testing, embedded directly into release cycles.

Sovereignty and Regulations will Lead Architectures

'Deploy anywhere' stacks with policy-based routing will become the baseline capability for enterprise operations.

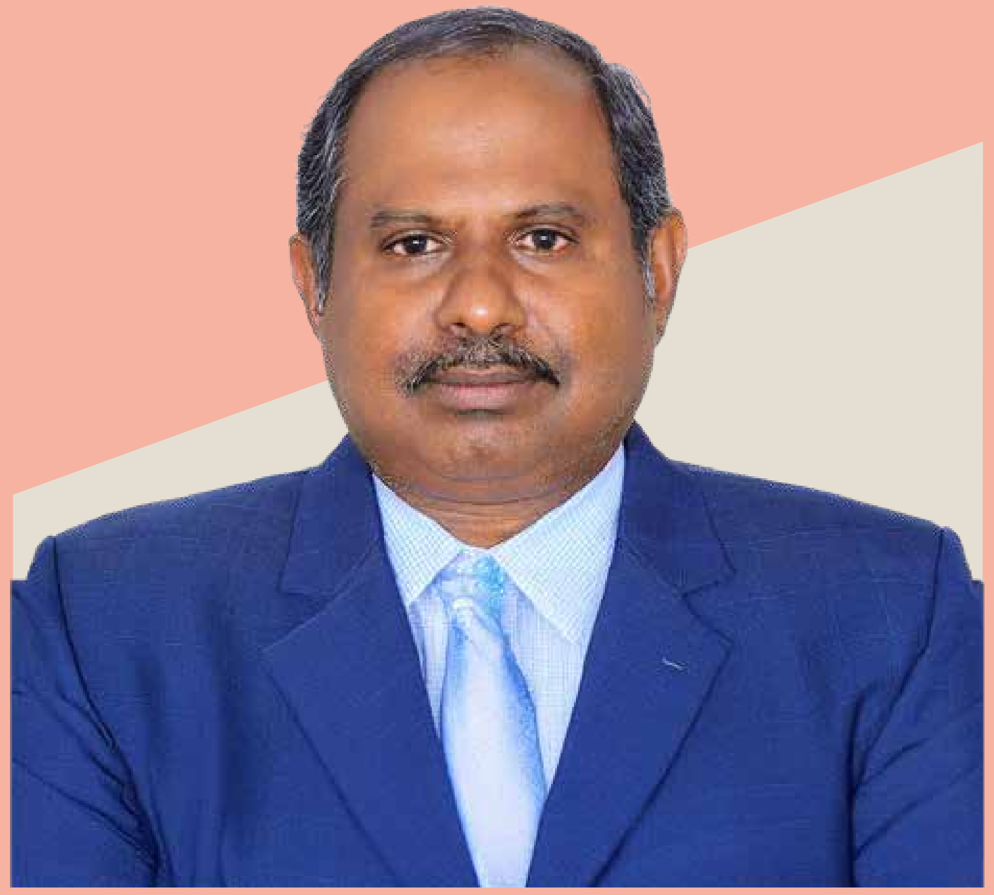
Conclusion

Enterprises implementing AI models cannot treat inference as an afterthought. As workloads scale and innovation accelerates, cost spikes, performance drift, and governance complexity become defining constraints. An inference-first AI platform brings control back to the execution layer, allowing organizations to expand and optimize operations while keeping orchestration, cost, and compliance firmly within enterprise control.

Authors

Dr. Anshu Premchand
Group Function Head
Multicloud and Digital Services,
Tech Mahindra

Dr. Anshu is a persuasive thought leader with more than 25 years of experience in digital and cloud services, technical solution architecture, research and innovation, agility and DevSecOps. She heads multicloud and digital services for the enterprise technologies unit of Tech Mahindra. In her last role, she was Global Head of Solutions and Architecture for Google Business Unit of Tata Consultancy Services, where she was responsible for programs across the GCP spectrum including data modernization, application and infrastructure modernization, and AI. She has extensive experience in designing large-scale cloud transformation programs and advising customers across domains in areas of breakthrough innovation. Anshu holds a PhD in Computer Science. She has special interest in simplification programs and has published several papers in international journals like IEEE, Springer, and ACM.



Sukumar Shanmugam
Delivery Head,
Tech Mahindra

Sukumar Shanmugam brings more than 20 years of expertise in AI, cloud, and data platforms. He leads strategic portfolios, builds client relationships, and scales high-performance teams. His experience spans telecom, banking, and hi-tech sectors, delivering large-scale technology programs with proven success.

About Tech Mahindra

Tech Mahindra (NSE: TECHM) offers technology consulting and digital solutions to global enterprises across industries, enabling transformative scale at unparalleled speed. With 152,000+ professionals across 90+ countries helping 1100+ clients, Tech Mahindra provides a full spectrum of services including consult-ing, information technology, enterprise applications, business process services, engineering services, network services, customer experience & design, AI & analytics, and cloud & infrastructure services. It is the first Indian company in the world to have been awarded the Sustainable Markets Initiative's Terra Carta Seal, which recognises global companies that are actively leading the charge to create a climate and nature-positive future. Tech Mahindra is part of the Mahindra Group, founded in 1945, one of the largest and most admired multinational federation of companies. For more information on how TechM can partner with you to meet your Scale at Speed™ imperatives, please visit https://www.techmahindra.com/.

TECH
mahindra

www.techmahindra.com
www.twitter.com/tech_mahindra
www.linkedin.com/company/tech-mahindra

Disclaimer: Brand names, logos, taglines, service marks, tradenames and trademarks used herein remain the property of their respective owners. Any unauthorized use or distribution of this content is strictly prohibited. The information in this document is provided on "as is" basis and Tech Mahindra Ltd. makes no representations or warranties, express or implied, as to the accuracy, completeness or reliability of the information provided in this document. This document is for general informational purposes only and is not intended to be a substitute for detailed research or professional advice and does not constitute an offer, solicitation, or recommendation to buy or sell any product, service or solution. Tech Mahindra Ltd. shall not be responsible for any loss whatsoever sustained by any person or entity by reason of access to, use of or reliance on, this material. Information in this document is subject to change without notice.