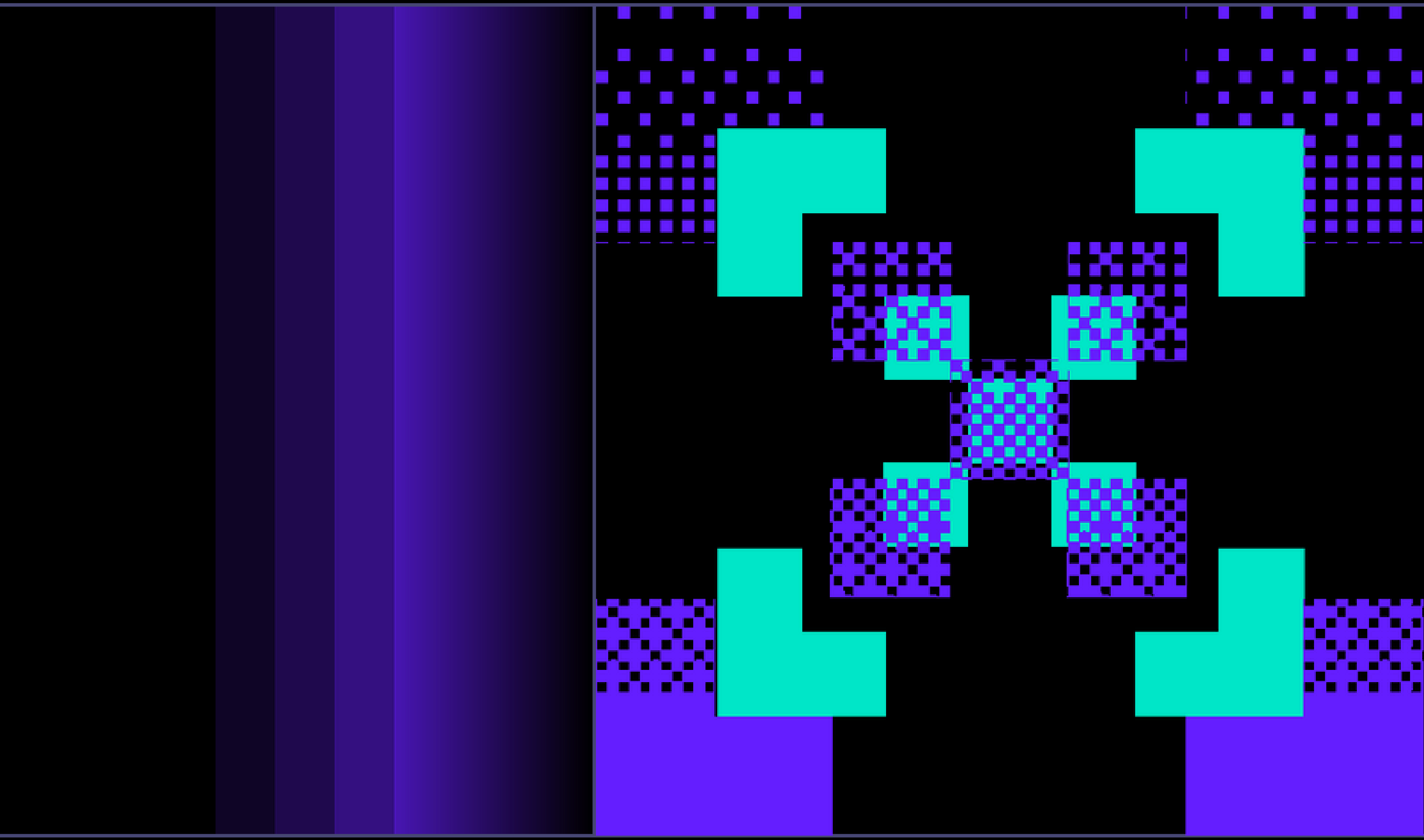
WHITEPAPER

Own Your AI

Enabling an Enterprise AI Playbook for Turning AI into Impact



Fireworks AI

WHITEPAPER

Own Your AI

Fireworks AI

Table of Contents

Executive Summary

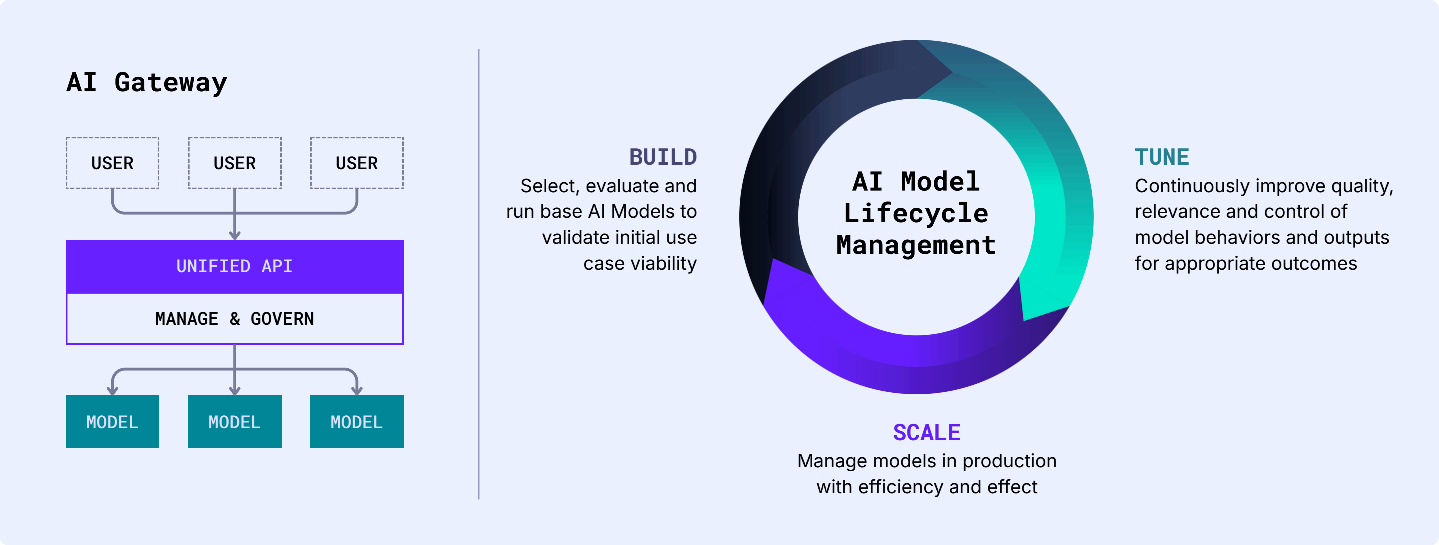
Enterprises now face a decisive moment. AI is no longer experimental; it is actively reshaping products, workflows, and competitive positioning across industries. Large language models, multimodal systems, and autonomous agents are enabling capabilities such as enterprise search, long-context summarization, intelligent coding, and complex workflow orchestration that were impossible just months ago. Companies that delay risk falling behind AI-native competitors who move faster, iterate smarter, and deliver insights at scale.

The challenge is not simply adopting AI but owning it strategically. Outputs must be accurate, domain-specific, fully-governed, and seamlessly integrated with product development. Achieving this requires a new enterprise-scale architecture that combines Product-Model Co-Design with robust AI infrastructure, including AI Gateways and Model Lifecycle Management. This architecture is powered by a continuous Run, Eval, Tune, Scale framework, ensuring models deliver measurable impact while evolving with the enterprise's needs.

This white paper presents a practical blueprint for capturing the full value of AI. It outlines how organizations can accelerate development, automate reasoning, extract insights from structured and unstructured data, and maintain strategic control over models and proprietary information. By adopting these patterns, enterprises move from ad-hoc experimentation to repeatable, high-impact AI deployments today while laying the foundation for long-term competitive advantage.

Key Takeaways

The enterprises that are best in class at harnessing the power of AI Models, are implementing two core patterns.



BUILD
Select, evaluate and run base AI Models to validate initial use case viability

TUNE
Continuously improve quality, relevance and control of model behaviors and outputs for appropriate outcomes

SCALE
Manage models in production with efficiency and effect

Owning Your AI can mean realizing the benefits of implementation rapidly and clearly:

SPEED
Development teams can quickly build at the edge.

EFFICIENCY
Models run, tune, and scale reliably for maximum impact and cost-effectiveness.

CONTROL
Proprietary data stays secure, and AI aligns with business objectives.

To discuss any of these topics, let us know how our engineering team can help at https://fireworks.ai/enterprise

The Rise of AI Model Use for Application Development

Artificial intelligence has moved from research labs into mainstream enterprise strategy. Very recently, large language models (LLMs) and other Generative AI systems have demonstrated remarkable capabilities in areas such as language understanding, code generation, and multimodal reasoning.

The result is a surge of interest in deploying AI-driven features, typically as agents — from intelligent chat assistants to autonomous coding tools - across every industry. These AI agents promise to handle complex, context-rich tasks autonomously, unlocking new levels of productivity.

Use Case Categories for AI Models

Code Assistance. From code generation to developer productivity, AI models excel at coding.

Conversational AI. Language understanding and reasoning is at the heart of LLM capabilities from chat bots to agents.

Agentic Workflows. Enabling function calling and multi-step reasoning has created powerful and flexible workflow possibilities.

Search & Understanding. AI Model context enables tailoring and tuning for enterprise search, comprehension, summarization, and Retrieval-Augmented Generation (RAG), ensuring accurate, actionable, and domain-specific insights.

Multimodal AI. Models that specialize in Audio and Vision unlock further capabilities such as transcription, and object identification.

See Use Cases & Solutions In Action for more »

Harnessing AI Capabilities: toward Product-Model Co-Development

AI Models upend the established method of software development in that they are probabilistic, vs deterministic. This means that they can provide unparalleled levels of capability for complex reasoning tasks previously only the domain of human operators, however at the same time need to be guided to required actions and outcomes, much like human operators. This, combined with the rapid improvements in general model capabilities, requires a new approach to general software development: Product-Model Co-Development.

In the mobile era, as the growth of web continued, and apps began to explode in volume and capability, the winners developed new practices around the concept of ‘Product Analytics’. Simply put, this was the process of analyzing how users would engage with a product or service. It enabled those product teams to track, visualize, and analyze user engagement and behavior data, which enabled optimization of those products and services.

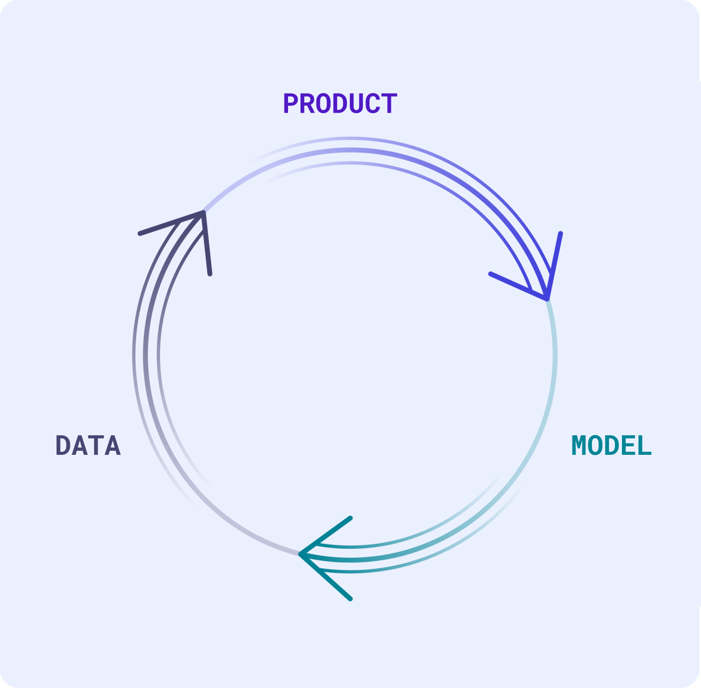
With AI Models, the concept is similar: by continuously evaluating and tuning AI models, product teams can very quickly harness new model improvements, while optimizing the user experience of products, and further, build a product in conjunction with the overall capabilities of the model. This synergy is driving rapid momentum in ‘AI Native’ startups, and producing powerful and novel new services such as Cursor (Code Assistance), ChatGPT (Conversational AI), Zapier (Agentic Workflows) and many more.

Building a Moat. The Value of the Data Flywheel.

Much like Product Analytics, in AI the key advantage is also data. AI Models thrive on continued data for training, and tuning. This is analogous to the very best products of the mobile app era: the winners were those who built atop product analytics as a flywheel for the right development for the right outcome.

In the AI era, this means using all enterprise knowledge to continuously tune AI Models to provide the highest quality, most effective experiences and outcomes.

As a consequence, and discussed later in the paper, enterprises must think about the longer term challenges of surrendering data outside of the perimeter, as it will most likely be used to tune something competitive at some time. This is a crucial element of emerging business strategy.



Challenges of LLM Integration

If AI is the answer then what is the question? Enthusiasm for AI does not automatically translate into successful integration. As LLMs' capabilities and promises rise, so do the practical challenges of weaving it into enterprise products and operations. The next section examines these challenges in people, technology, and risk domains.

Building AI-powered applications in an enterprise context is trivial to experiment with, but complex to take to production. Organizations often discover that after the excitement of AI proofs-of-concept, the road to reliable, scalable production AI is fraught with challenges. We categorize these essential challenges into three areas: people & process, technology implementation, and risk management.

People & Process Considerations

Adopting AI requires new skills and ways of working that many enterprises are still developing. Examples include:

- Building and maintaining the expertise to customize and maintain LLMs. Skills are scarce and in extreme high demand, tools and techniques for management and workflow are immature, and the knock on effects of LLM adoption such as GPU (instead of CPU) management and monitoring add to the burden.
- Existing product development teams may not have experience co-creating features with AI systems. This can lead to a gap between the core ML team and the rest of the engineering organization. Many organizations lack these collaborative processes and end up with siloed efforts.

Overcoming this requires a round of cultural change, training, and new workflows that integrate AI development into the regular software development lifecycle, alongside the adoption of tools to support.

Technology Considerations

Implementing AI at an enterprise scale introduces substantial technical complexity beginning with selection of models through to ensuring no regressions (known as divergences in LLMs) as new models are released.

Choosing the right model strategy is a primary challenge. There are essentially two core options:

Closed Models	Open Models
Start with external AI APIs (e.g. closed models from third-party vendors) in order to stay 'state of the art' easily and for speed, but this can backfire. Relying solely on closed models means your AI roadmap — which will be the core business logic for an organization — is at the mercy of someone else's API changes and limitations. Common pain points include: ● Lack of customization means no ability to fine-tune the model for your domain which can impact quality, speed, and cost. ● Inconsistent behavior (e.g. tone or accuracy changes when the provider updates their model) which can be difficult to spot and experientially damaging. ● Slow iteration cycles, and runaway costs as usage scales. ● In short, closed APIs can get things moving, but may stall adoption and limit agility, due to vendor lock-in, high costs, and below par quality.	**Adopting open-source models or building models to deliver the right customization and quality** also brings its own hurdles. There are significant potential advantages to doing so, however pain points include: ● Running state-of-the-art open models requires significant infrastructure (GPU clusters, high-performance inference engines) and deep ML ops expertise. ● Fine-tuning models on proprietary data can require managing large datasets, training jobs, and model versioning — tasks beyond the comfort zone of many engineering teams. ● Serving real-world workloads demands optimized, high-concurrency infrastructure to meet latency targets. For instance, delivering sub-second responses to millions of users is not achievable with naive model deployments. ● Evaluating and updating to the latest capabilities. New model versions and architectures emerge every few months, leading to "choice overload" with unclear evaluation methodologies. Without robust processes and tools, enterprises risk chasing a moving target or getting stuck with yesterday's AI technology.

In summary, the technical implementation challenge is about complexity: how to integrate AI models into products in a performant, maintainable, and future-proof way.

Risk Considerations

AI Model integration also introduces new risks that must be considered, understood, and managed. At the core is the choice of how to enable and manage key enterprise data is the focus of the associated risks, and as a result the risks are more easily mitigated in a fully owned environment (meaning through the use of custom Open Models), than through the use of third party vendor APIs.

For this reason, Open Models — such as DeepSeek, GPT OSS, QWEN, KIMI and a myriad of smaller specialized models — are increasingly attractive to organizations. While they may lack the pure capability of a closed model, they are tunable for better results with the right training, and do not expose what they know out of the perimeter.

They can be described as follows:

Data Safety. Data can be exposed to models in a controlled way through the use of typical and specific cloud storage with policies — for example zero data retention — to ensure the model is tuned with the appropriate information, but that the data is at risk of exposure. Models can be deployed into an enterprises established data center or cloud environment (known as Bring Your Own Cloud) to further ensure data safety.

Data Privacy. Similarly, by owning the model weights through the tuning of an Open Model on enterprise data, that data is not transmitted to a third party closed model API provider that may use it (deliberately or unintentionally) as part of a training data set for its future models. In other words, managing an open model ensures that the moat of enterprise data is retained.

Governance. AI outputs need to align with corporate values and guidelines, requiring mechanisms to filter or constrain responses. Continuous evaluation, customization, and monitoring are required, ideally across smaller specialized models where the knowledge and response surface area is more constrained. This is more manageable task with an owned model, vs any unintended side effects from a non-controlled model update.

Overall, assessing the strategic risk of data safety and privacy is a crucial first priority, and then tackling the operational risks through enhanced DevSecOps processes.

Patterns and Practices: Building Enterprise AI Advantage

Implementing AI across an enterprise is no longer just a technical challenge. It is a strategic opportunity. Early adopters are discovering that the right architectural patterns unlock efficiency gains, new capabilities and measurable business impact.

Core practices differentiate leading enterprises:

1. **Building an AI Gateway to centralize intelligence and decouple execution**: Enterprises move beyond fragmented AI experiments by consolidating model access, orchestration, and governance. Product teams can innovate without managing model complexity, while centralized tuning, deployment, and monitoring enable scaling AI across multiple products, departments, and regions, all while retaining full control over sensitive data.
2. **Implement AI Model Lifecycle Management to realize continuous value**: Organizations ensure models are continuously aligned with business outcomes by structuring how they run, tune, and scale. Integrating supervised fine-tuning, reinforcement fine-tuning, and automated evaluation pipelines transforms ad-hoc experiments into repeatable, high-impact deployments that drive consistent speed, efficiency, and control.

Building an AI Gateway

Enterprises want two things from AI adoptions: centralized control for governance and efficiency, and decentralized speed so product teams can build quickly. The AI Gateway pattern has emerged as the way to deliver both.

The AI Gateway serves as the intermediary through which all AI requests and responses flow. Instead of individual product teams calling a remote model API in an ad-hoc way, they interface with the gateway. The gateway abstracts the complexity of model selection, prompt formatting, fine-tuning, and scaling. Product teams gain a stable interface, while the enterprise retains flexibility to upgrade or swap models (open-source, proprietary, fine-tuned models, etc.) without changing front-end application code. The result is an architecture where product development and AI development happen in tandem, not in silos. And a central authority can monitor, assess, and manage the shared service offerings.

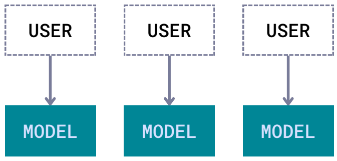
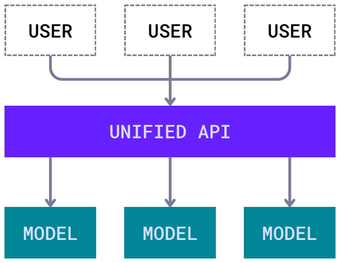
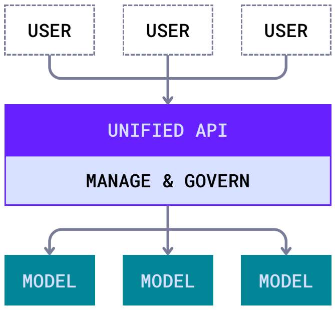
Core Capabilities of an AI Gateway:

- **Model Catalog and Orchestration** maintains a library of approved models for various tasks like language, vision, speech, or use case specific tasks such as coding assistance. It can orchestrate calls to the appropriate model or even chains of models/tools. For example, an AI assistant might use a language model for understanding a query, then a vision model for an image, then a tool API all coordinated by the gateway. This capability ensures teams consistently leverage the best model for each task without having to manage individual integrations.
- **Customization and Fine-Tuning Pipeline** provides a mechanism to fine-tune or customize models on enterprise data (through a range of techniques (see next section)). This ensures the AI output is tailored to the company's domain, terminology, and policies. The gateway supports both supervised fine-tuning (SFT) and reinforcement fine-tuning (RFT) using Fireworks' customization engine (FireOptimizer) enabling iterative improvements, automated evaluation, and versioning. This ensures models to achieve exact quality targets and align with business objectives, all while integrating seamlessly with existing applications and workflows without disrupting front-end applications.

- **Unified API for Developers** offers a consistent interface for product teams to access models through the gateway. This standardization speeds up development since teams don't need to reinvent integrations for each new model. It also enforces guardrails (for example, input/output validations or prompt templates) to maintain consistency and safety across applications. It may act as an intelligent broker selecting models that best fit the request in tandem with model catalog evaluation and performance characteristics.
- **Monitoring and Observability** logs all interactions and tracks key performance indicators such as latency, throughput, and quality metrics, as well as audit trails for content (important for compliance). An enterprise-ready platform will offer token-level logging, traceability, and controls like RBAC for model usage. Such monitoring helps detect issues (e.g. performance regressions or inappropriate outputs) early and provides transparency for audit requirements.
- **Scalability and Deployment Flexibility** allows the gateway to operate across cloud, on-premises, or hybrid environments to meet data governance needs. It manages the scaling of AI workloads — from provisioning GPUs to optimizing inference — so that applications get reliable, low-latency responses even as usage grows. A well-architected AI gateway can handle massive throughput (the Fireworks platform, for instance, processes over 5 trillion tokens per day in production environments) and support multi-region deployments for redundancy.

AI Gateway Development Model

Enterprises typically progress through phases of gateway adoption, from ad hoc experimentation to fully owned, enterprise control.

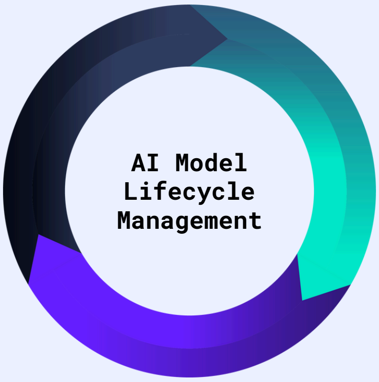
Phase	Ad Hoc	Unified	Complete
	 Initially, there is no control of AI model access, with teams and apps interacting with multiple models and vendors on an ad hoc basis.	 USER USER USER UNIFIED API MODEL MODEL MODEL By providing a Unified API, users and apps can now interact with a catalog of possible models in a consistent manner and select best fit models for use case.	 USER USER USER UNIFIED API MANAGE & GOVERN MODEL MODEL MODEL By providing a series of shared services such as logging, auditing, access controls, and routing, the shared service now provides optimization across the model catalog.
Unified API	No - per vendor	Yes	Yes. Seamless switching via brokering for the parallel evaluation.
Model Catalog	No	Yes. One model or series of models. Brokering based on available variables.	Yes. Many models, vendors, and specialized models for specific workloads. Brokering based on evaluated performance of model against use case.
Customization	Ad Hoc	Depends on vendor and open model availability.	Yes. Complete loop of evaluation, tuning, assessment to further improve and optimize.
Monitoring	Ad Hoc	Depends on vendor	
Deployment	Ad Hoc	Depends on vendor	Complete loop of evaluation, tuning, assessment to maximize efficient serving
Security & Compliance	Hosted (multiple)	Hosted	Hybrid: Hosted, BYOC

FIREWORKS PERSPECTIVE

Fireworks recommends that enterprises take ownership of an AI gateway to maximize control, flexibility, and performance. Owning the gateway enables orchestration of multiple models, fine-tuning to domain needs, efficient scaling of AI workloads, and full governance over data and operations. This approach positions enterprises to adapt rapidly as new models emerge, reduce vendor risk, and unlock the full potential of AI across the business.

AI Model Lifecycle Management

Behind the gateway is a Model Lifecycle Management process balancing ensuring maximum quality and throughput of requests, ensuring best cost and resource efficiency to achieve that. This process can be considered a loop of ‘Run, Eval, Tune, and Scale’ each with multiple elements of maturity:



BUILD
Select, evaluate, and run base AI Models to validate initial use case viability

TUNE
Continuously improve quality, relevance and control of model behaviors and outputs for appropriate outcomes

SCALE
Manage models in production with efficiency and effect

Behind the gateway is a Model Lifecycle Management process balancing ensuring maximum quality and throughput of requests, ensuring best cost and resource efficiency to achieve that. This process can be considered a loop of ‘Build, Tune, and Scale’ each with multiple elements of maturity:

BUILD

In this step, the focus is on selecting and running base AI models to validate use cases. To some extent initial tuning might be required to make that selection, but for simplicity we assume a general loop. Activities for Run include:

- **Integrating a pre-trained model to power a specific feature** - for example, adding a chat assistant using a generic LLM API, or using an off-the-shelf vision model for image recognition.
- **Creating and maintaining Evals** - for example, using https://evalprotocol.io, to ensure consistency of output from a model
- **Updates to latest versions** of the same model which implies the same evaluation mechanics.
- **Entering the model into catalog** for the AI Gateway for testing against enabled use-cases or workloads.

TUNE

In this step, organizations recognize the need to improve quality, relevance, and control — so they begin to tune the AI to their unique needs. This happens in pre-production, but also as continuous activity. Activities for Tune include:

- **Prompt Engineering** involves managing versioned prompt templates managed via the Fireworks console, with automated metrics for output relevance, hallucination rate, and token efficiency. Enterprise-grade tooling ensures consistent and scalable prompt management, while automated evaluation metrics provide real-time insights for proactive adjustments.

- **Supervised Fine-Tuning (SFT)** leverages FireOptimizer to train models on enterprise datasets, with integrated model versioning, automated evaluation, and logging to ensure domain-specific accuracy. Tailored model behavior improves task-specific performance, while comprehensive logging supports transparency and compliance.
- **Reinforcement Fine-Tuning (**RFT) optimizes models for business-aligned objectives such as speed, accuracy, or compliance. Continuous model refinement aligns AI behavior with evolving objectives, and automated deployment through the AI Gateway ensures governance without manual intervention.

SCALE

In this step the goal is to efficiently and effectively scale a given model across the organization and user base reliably. Here the concern is not just one feature or model, but a scalable, governed AI ecosystem. Activities for Scale include:

- **Deploying models** via hybrid AI Gateway architecture for flexible cloud, on-prem, or multi-region environments.
- **Monitoring latency, throughput, and quality** metrics to ensure consistent performance at scale..
- **Management of costs, access, authorization**, and audit trails for enterprise governance
- **Integrating tuned models into product workflows** through the unified API, enabling seamless adoption without changes to application code.

AI Model Lifecycle Management Maturity Model

Building Model Lifecycle Management capabilities is typically via increasing maturity in each step.

	* BASIC	** STANDARD	*** ADVANCED
Build	Deploy AI to validate business use cases using pre-trained open or third-party models via Fireworks public API	Ensure measurable, reliable AI performance with formal benchmarking using Fireworks evaluation suite; automated logging and observability	Optimize model selection for enterprise impact: AI Gateway orchestrates multiple vendors/specialized models per use case
Tune	Tailor AI outputs for relevance and quality with versioned prompt engineering using performance metrics tracked in Fireworks console	Improve accuracy, domain alignment, and consistency with FireOptimizer SFT with enterprise datasets, model versioning, and logging	Achieve fully aligned AI outputs with business goals using RFT with reward modeling, deployed via AI Gateway with real-time evaluation
Scale	Enable team-level or single-cloud deployment for quick experimentation	Govern multiple AI workflows reliably via centralized AI Gateway orchestrating multiple models with automated brokering, monitoring, and access control	Enterprise-grade global deployment: Multi-region, hybrid deployment with on-prem BYOC, high availability, and low-latency scaling

FIREWORKS PERSPECTIVE

Fireworks recommends that enterprises build processes to deliver consistent AI Model Lifecycle Management to maximize the performance and behaviors of individual models to achieve the desired technical and business outcomes. Adopting an MLM mindset enables teams to consistently deliver and continuously improve the models powering the next generation of user experiences, while also retaining the ability to take advantage of the latest innovations in model capabilities seamlessly.

Use Cases & Solutions in Action

Code Assistance

From code generation to developer productivity, Fireworks delivers context-aware code suggestions, inline fixes, and automated code review integrated directly into developer IDEs. Teams accelerate development, reduce repetitive work, and maintain high-quality code with Fireworks' low latency infrastructure (1000+ tokens/sec) and tuning capabilities via FireOptimizer, to ensure consistent, measurable results. Enterprises such as Cursor and Sourcegraph have reported measurable improvements in developer throughput and code quality.

Conversational AI

Language understanding and reasoning enable chatbots, virtual assistants, and AI agents to interact naturally and accurately. Fireworks ensures AI outputs are context-aware, aligned with business objectives, and ready for enterprise deployment.

Agentic Workflows

Autonomous AI agents built on the Fireworks platform execute complex, multi-step tasks from support workflows to operational decision-making. Leveraging the AI Gateway for orchestration and reinforcement fine-tuning, enterprises achieve outcomes that go beyond simple automation, delivering measurable efficiency gains while retaining full ocntrol over data and processes.

Search & Understanding

Fireworks enables unified search across structured and unstructured data while ensuring proprietary information remains secure. Tuned LLMs power enterprise search, Retrieval-Augmented Generation (RAG), and long-context summarization to improve retrieval accuracy, relevance, and actionable insights. The AI Gateway orchestrates retrieval and generation to ground outputs in trusted data, while continuous fine-tuning adapts outputs to domain and compliance needs. Teams find insights faster, make smarter decisions, and maintain full control over enterprise knowledge.

Multimedia

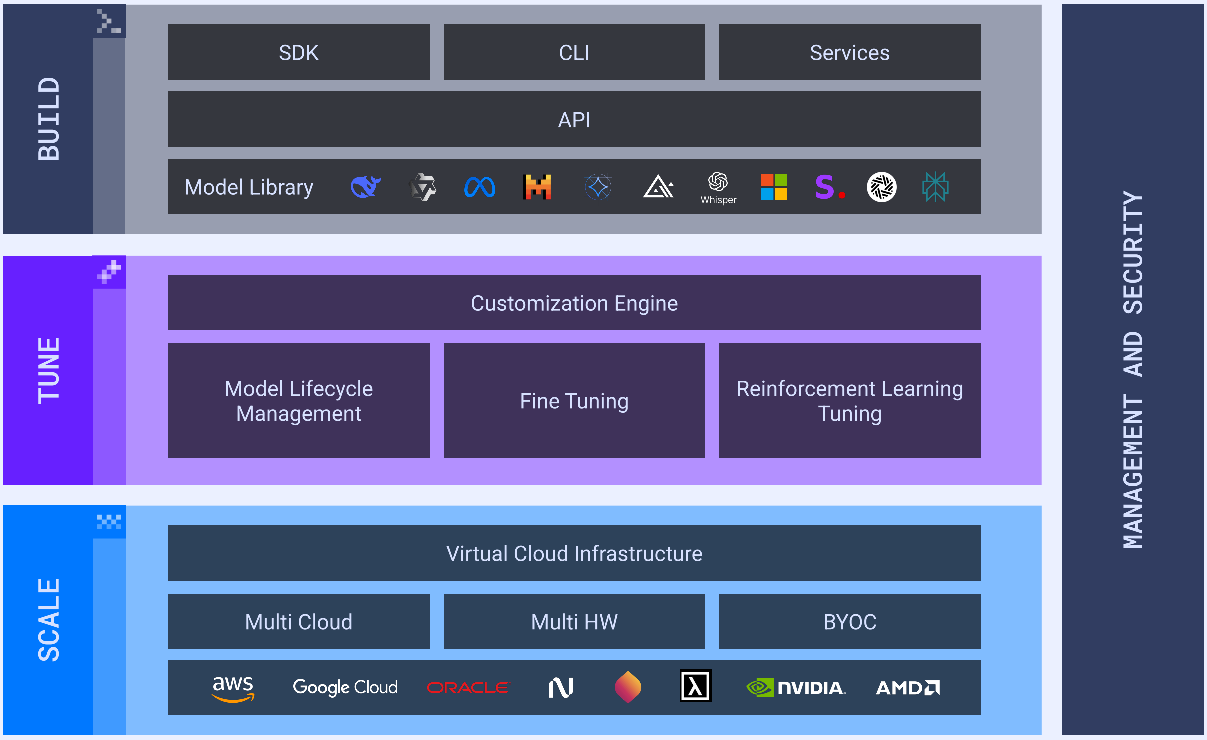
Audio, visual, and text models unlock advanced capabilities such as transcription, sentiment analysis, and multimedia recommendations, to streamline enterprise workflows. Optimized inference and flexible deployment accelerate cross-functional intelligence and automation while keeping enterprise data secure and under enterprise control.

Practical Applications

Fireworks translates these pillars into measurable enterprise impact across key use cases:

Use Cases	Fireworks Features	Key Metrics/Impact	Benefit
Code Assistance	IDE integrations FireOptimizer tuning Low-latency infrastructure	1000+ tokens/sec, Faster feature delivery Reduced errors	Teams build faster, maintain quality, reduce repetitive work
Conversational AI	Tuned LLMs Language understanding & reasoning	Context-aware responses, aligned to business objectives	Natural interactions, enterprise-ready chatbots and AI agents
Agentic Workflows	AI Gateway orchestration Reinforcement fine-tuning	Autonomous task execution Improved reasoning	High-impact workflows under full governance, measurable efficiency gains
Search & Understanding	Tuned LLMs Secure enterprise indexing AI Gateway orchestration Continuous fine-tuning Unified access to structured & unstructured enterprise data High-concurrency inference Extended token windows	Improved search relevance Immediate context-aware answers Continuously improve accuracy Reduced manual analysis Higher summary accuracy	Retain proprietary knowledge Faster insights Smarter decisions Efficient processing of large docs Reduce research effort Data stays under control
Multimedia	Audio/Text/Visual Analysis Optimized inference Hybrid deployment	Accelerated workflows Cross-functional insights	Streamlined intelligence across departments, secure data usage

The Fireworks Platform



Developer

Fireworks provides core API access to its platform to assist in building an AI Gateway, along with developer productivity through SDKs and Tools to enable improved model lifecycle management for engineering teams.

Capabilities

Fireworks offers multiple capabilities:

- Access to a broad array of the latest open weight AI models, pre-optimized for performance, and advanced tuning features to further ensure efficient performance.
- Finished services such as Audio Transcription that are pre-built and available for direct consumption.
- Enterprise services such as RBAC, Auditing and Logging to further assist Enterprise platform teams.

Infrastructure

The Fireworks virtual cloud infrastructure provides myriad deployment options across clouds, hardware, and regions to ensure the right deployment for any team, alongside a pre-optimized layer for performance and management.

How Cursor Built The Industry's Best Coding Assistant

CURSOR

From Startup to AI-Native IDE

Anysphere's Cursor launched in 2022 with a mission to transform software development by deeply integrating AI into the coding process. By mid-2025, Cursor established itself as one of the most advanced AI-native integrated development environments, with adoption across thousands of companies, including Amazon, Stripe, Instacart, and Shopify.

Enterprise Adoption: Used by thousands of companies including Amazon, Stripe, Instacart, Shopify

Amazon Internal Users: 1,500 employees onboarded to dedicated Slack channel

Key Feature Impact: "Fast Apply" enables single-click acceptance of AI-generated code suggestions

Developer Efficiency Gains: Context-aware code suggestions, smart refactoring, and natural language code generation

Early Challenges and Strategic Focus

In a crowded AI coding tools market, Cursor needed to differentiate. The team focused on delivering more than code suggestions, developing features that understood developer intent and context. Innovations such as natural language code generation and smart rewrites enabled more intuitive interactions with code.

Integration with Fireworks AI: A Turning Point

A key milestone in Cursor's evolution was integrating Fireworks AI's inference stack. This integration improved speed and accuracy, introduced the "Fast Apply" feature for one-click code acceptance, and streamlined developer workflows.

Scaling and Enterprise Adoption

Cursor's adoption grew rapidly among major tech companies with strong engagement from Amazon employees, including over 1,500 participants participating in a dedicated internal Slack Channel, highlighting its enterprise appeal.

Continuous Innovation and Developer-Centric Design

Cursor continues to enhance developer productivity with features that understand complex codebases, generate code from natural language prompts, and refactor intelligently. Its ongoing evolution demonstrates the benefits of product-model co-development and enterprise AI integration.

The Value of Owning Your AI

Enterprises adopt AI to gain competitive advantage, but true value comes from consistently balancing speed, efficiency, and control across applications, teams, and data. Fireworks enables organizations to deliver on all three dimensions consistently, transforming experiments into scalable, business-aligned AI deployments.

Impact Pillar	Benefit	How Fireworks Delivers
Speed	Development teams can quickly build at the edge.	Industry-leading low-latency infrastructure with continuous Day 0 support for new models. Offered on 8 cloud providers across 18 global regions.
Efficiency	Models run, tune, and scale reliably for maximum impact and cost-effectiveness.	Powerful tuning capabilities (SFT & RFT) and automated evaluation for optimal performance and selective brokering, Supporting datasets up to 100 million records. Its optimized infrastructure delivers up to 4X higher throughput per instance and up to 50% lower latency compared to open source solutions.
Control	Proprietary data stays secure, and AI aligns with business objectives.	Unified platform for consistent model lifecycle management, observability, and governance. Supports SOC 2 Type II and ISO 27001 compliance for enterprise deployments. Role-based access controls with audit logs covering 100% of model interactions.

Summary

AI is poised to become a core feature of virtually every enterprise application. To move swiftly from experimentation to high-impact deployment, organizations need a repeatable architectural blueprint that integrates AI into the fabric of product design and delivery.

By combining foundational patterns such as AI Gateway, Model Lifecycle Management, and Product-Model Co-Design with the Fireworks platform, enterprises gain:

- Tuned models for domain-specific performance
- Unified access via the AI Gateway
- AI Model Lifecycle Management maturity for continuous improvement

This approach ensures AI becomes a true strategic enabler rather than an ad-hoc experiment, delivering fast, reliable, and governed outcomes at scale. Fireworks' experience with AI-native startups, digital natives, and large enterprises enables organizations to leverage these patterns effectively, accelerating adoption and reducing risk.

Next Steps

Things are moving fast. We recommend the following steps:

Assess your current implementation strategy, and gaps. Fireworks can help with this as an early exploration with one of our architects.

Consider the maturity of essential skills and processes, of shared LLM services, and of essential LLM Lifecycle Management. Fireworks can assist with the next logical step, whether it is providing optimization and tuning to models, or embedding in the AI gateway as a model broker.

Establish a team to build on this position. Implementing and testing the patterns and practices to ensure they work for your team. Fireworks can assist in this early execution, and at later stages of optimization.

Get development teams started with Fireworks public platform. Development teams can get started with our self-service platform enabling them to run the best in open LLMs, tune them for effect, and scale them on the best low-latency infrastructure.