

Building an AI Agent-Ready Hybrid Cloud



PREPARATION COMMITTEE

Produced By	Huawei Cloud Computing Technologies Co., Ltd.
Editorial Advisors	Dr. Peter Zhou, Antony Gu, Fang Pu
Editorial Review Committee	Tu Yan, Feng Jilong, Li Yawei, Zhang Beibei
Chief Editors	Hao Lipeng, Huang Zhi, Li Yin, Li Xiao, Wang Can, Liu Yixin, Sun Junjie, Zhang Jindong, Bai Jun, Wang Fan, Wang Mengqing, Xiong Honghuai, Xu Jing, Wang Li, Shen Fangzhi, Yu Shanying, Chen Yong, Feng Chuhao, Wang Yijie, Zhai Xu
Executive Editors	Wang Rui, Shi Yonglong, Qin Yue, Fang Minjuan, Feng Xi, Ren Jiangjing

(Names are listed in no particular order.)

To prepare for the agentic era, CIOs need to act on seven priorities:

01

Embrace a hybrid cloud architecture

Adopt a multi-cloud hybrid architecture that supports cross-cloud Model-as-a-Service (MaaS), enabling the lowest total cost of ownership (TCO) while keeping pace with the rapid evolution of AI models and compute infrastructure.

02

Build a unified multimodal AI data lake

Establish a unified multimodal AI data lake as the data foundation for enterprise AI agents. Fuel enterprise AI with high-quality proprietary data and evolve your data architecture to support autonomous decision-making.

03

Adopt a "Stable On-Premises, Agile in the Cloud" deployment strategy

Deploy stable, industry-specific foundation models (such as those for meteorology) on premises, while accessing rapidly evolving general-purpose foundation models through cloud-based MaaS services. Leverage confidential computing to secure inference data when using cloud-based MaaS.



04

Build secure development and runtime environments for AI agents

Build a standardized operating environment encompassing secure development, compliance testing, runtime monitoring, and governance to ensure AI agents are auditable, traceable, explainable, and governable—providing a trusted foundation for large-scale deployment.

05

Break down AI application silos

Build a unified agent platform to eliminate AI application silos and black boxes, maximizing the value of enterprise data and AI assets.

06

Transform operational efficiency with AI

Build an intelligent cloud management platform where humans and AI agents work together to deliver enterprise-grade security and compliance, exceptional availability, and superior operational efficiency.

07

Shift to token- and agent-centric operations Optional

Move beyond traditional resource-centric cost management toward outcome-driven operations centered on tokens and AI agents, continuously optimizing cost, capacity, and business value.



CONTENTS

1

5-8

Building an AI Agent-Ready Enterprise: Trends and Hybrid Cloud Architecture

1.1 Trends and Challenges

1.2 Agent-Oriented Hybrid Cloud Deployment Models and Architecture

2

9-14

Agent-Native Enablement Layer

2.1 Key Trends Shaping AI Agents

2.2 Key Challenges in Scaling Enterprise AI Agents

2.3 Implementation Approach

3

15-23

AI-Native Enablement Layer

3.1 Model Gateway

3.2 Integrated Training & Inference Platform

3.3 AI Multimodal Data Lake

4

24-27

Agile Cloud Infrastructure

4.1 Trends and Requirements

4.2 Key Capabilities

4.3 Implementation Approach

5

28-31

Assured—Trusted Security

5.1 Trends and Requirements

5.2 Key Capabilities

5.3 Implementation Approach

6

32-35

Administrable—Intelligent Operations

6.1 Trends and Requirements

6.2 Key Capabilities

6.3 Implementation Approach





01

Building an AI Agent-Ready Enterprise: Trends and Hybrid Cloud Architecture

1.1 Trends and Challenges

Key Trends		
The exponential rise of enterprise AI agents As AI moves into production, enterprises are shifting from a model-centric approach to one centered on AI agents that execute business tasks and orchestrate workflows. At the same time, AI applications are rapidly evolving from standalone agents to collaborative multi-agent systems, unlocking increasingly autonomous and sophisticated business capabilities.	The widespread adoption of AI agents is accelerating AI-ready data architectures As AI-ready data architectures become the enterprise standard, enterprise data has entered the multimodal era. Data applications are evolving from human-driven analytics to autonomous decision-making by AI agents, and data management is evolving toward ontology-based knowledge governance. Meanwhile, continuous advances in AI agent memory are enabling data agents to evolve toward increasingly sophisticated forms of intelligence.	Security and compliance are non-negotiable prerequisites for enterprise AI adoption As AI agents transition from personal productivity tools to enterprise applications, selection criteria are shifting from functionality alone to security, trust, and compliance. Establishing a trusted AI foundation is becoming a prerequisite for deploying AI at scale.



Challenges Enterprises Must Address	
Limited cost elasticity and the lack of unified orchestration for general-purpose and AI compute - Traditional in-house built cloud infrastructures require significant upfront capital investment (CAPEX), making it difficult for enterprises to transition smoothly toward hybrid cloud architectures that combine on-premises infrastructure with cloud services under a more flexible operating expenditure (OPEX) model. - At the same time, the lack of unified scheduling across general-purpose compute, AI compute, and multi-cloud resources limits enterprises' ability to handle highly dynamic workloads generated by LLMs and large-scale AI agents.	Data silos and the lack of a unified multimodal AI data lake - In the era of Agentic AI, enterprise data platforms are evolving beyond traditional business intelligence and reporting to support foundation model training, incremental fine-tuning, retrieval-augmented generation (RAG), and agent inference and collaboration. - Traditional lakehouse architectures were designed primarily for structured data. Their data models, processing capabilities, compute architecture, and scalability are not sufficient to meet the end-to-end data requirements of modern AI workloads.
The lack of secure and flexible hybrid MaaS offerings - Modern AI is evolving at an unprecedented pace. Without mature hybrid MaaS offerings spanning on-premises and cloud environments, together with confidential computing, enterprises are often forced to choose between the security of isolated on-premises deployments and the flexibility of cloud-based AI services, albeit with higher security risks. - A more balanced approach would be to adopt a "Stable On-Premises, Agile in the Cloud" deployment strategy: Deploy stable, industry-specific foundation models (such as those for meteorology) on premises, while accessing rapidly evolving general-purpose foundation models through cloud-based MaaS services. When using cloud-based MaaS, leverage confidential computing to secure inference data.	Building trusted development and runtime environments for AI agents - AI agents introduce a new generation of security challenges, including expanded attack surfaces, cascading risks across interconnected agents, and increasingly complex accountability. - To address these challenges, enterprises need a standardized operating environment encompassing secure development, compliance testing, runtime monitoring, and governance to ensure AI agents are auditable, traceable, explainable, and governable. Such an environment provides the trusted foundation required for large-scale AI agent deployment.

1.2 Agent-Oriented Hybrid Cloud Deployment Models and Architecture

For enterprises deploying AI agents in on-premises data centers, hybrid cloud is emerging as the preferred architecture. It enables organizations to build an end-to-end environment for AI agent development, model training, and production deployment while ensuring that sensitive data remains on premises, security stays under enterprise control, and core AI capabilities remain in-house. As enterprises adopt AI agents at scale, we believe two deployment models will gradually emerge.

Model 1: Fully on-premises deployment. In the first model, LLMs, AI agent development, and AI agent runtime are all deployed and operated entirely on premises. This approach provides the highest level of security and control, but it also demands substantial capital investment, strong in-house AI engineering capabilities, and significant long-term operational resources.

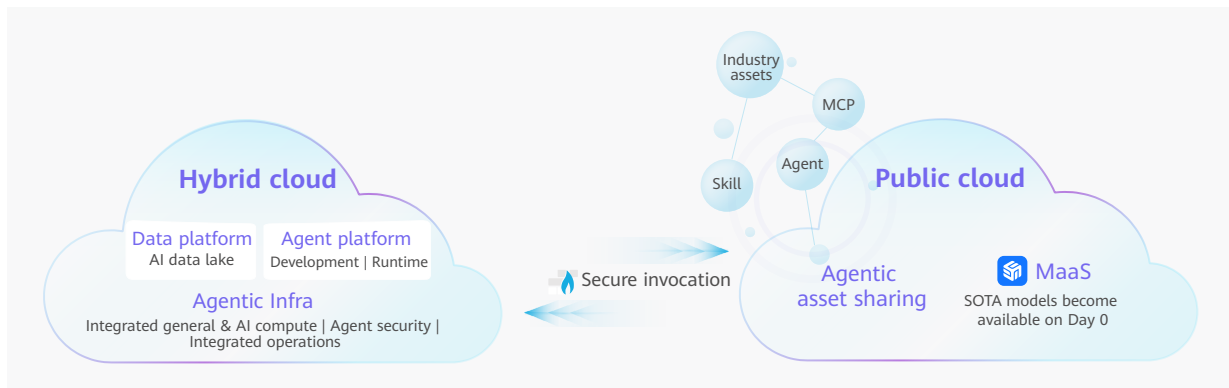
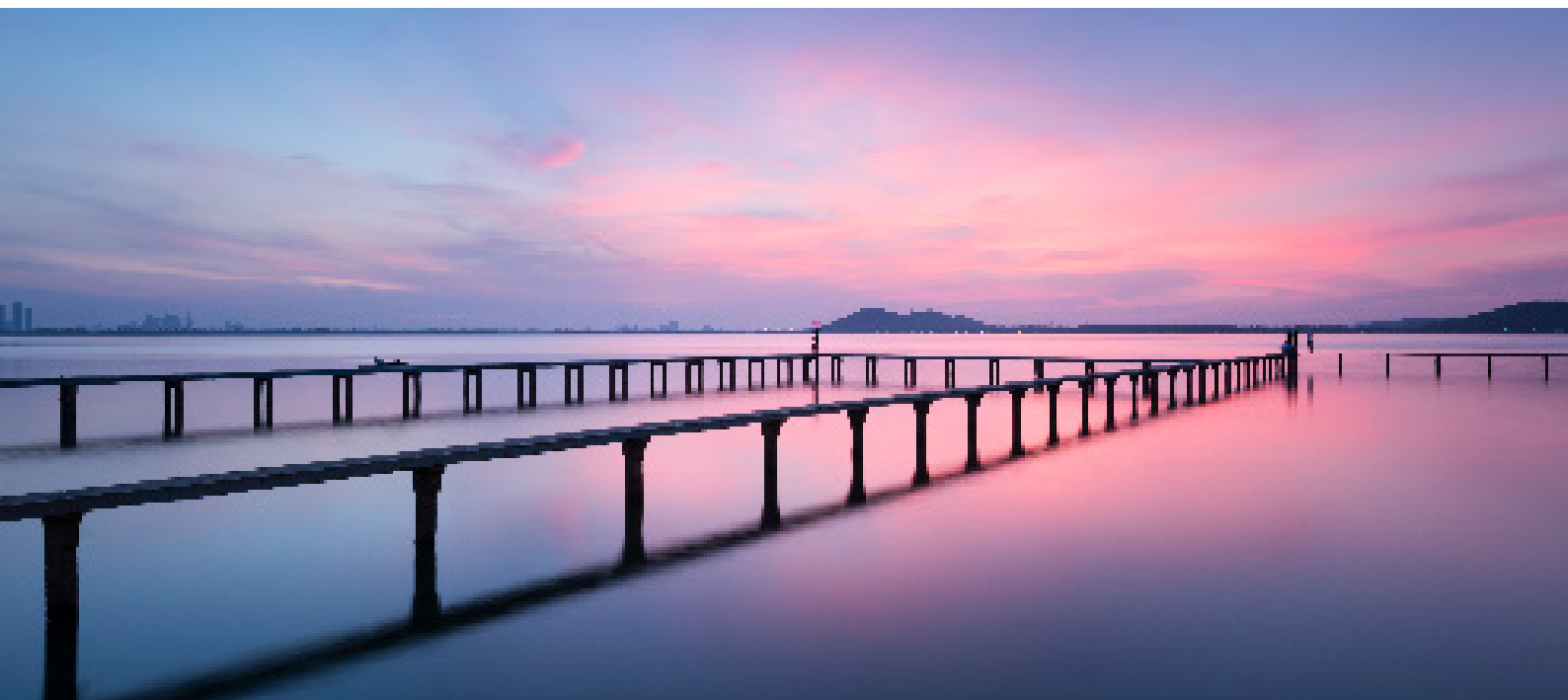


Figure 1.1 Hybrid cloud-based agent collaboration model

Model 2: Hybrid cloud-based agent collaboration. We believe the second model will become the dominant approach for enterprise AI agents. Under this model, enterprise data, core business workflows, and AI agents remain on premises, while advanced capabilities can be securely accessed from the public cloud through a Model Gateway, along with centralized operations management and security governance. This enables enterprises to securely access state-of-the-art MaaS services, MCP tools, and AI skills while maintaining control over sensitive data, access permissions, and enterprise AI assets.



Based on this approach, we propose the following hybrid cloud architecture for enterprise AI agents.

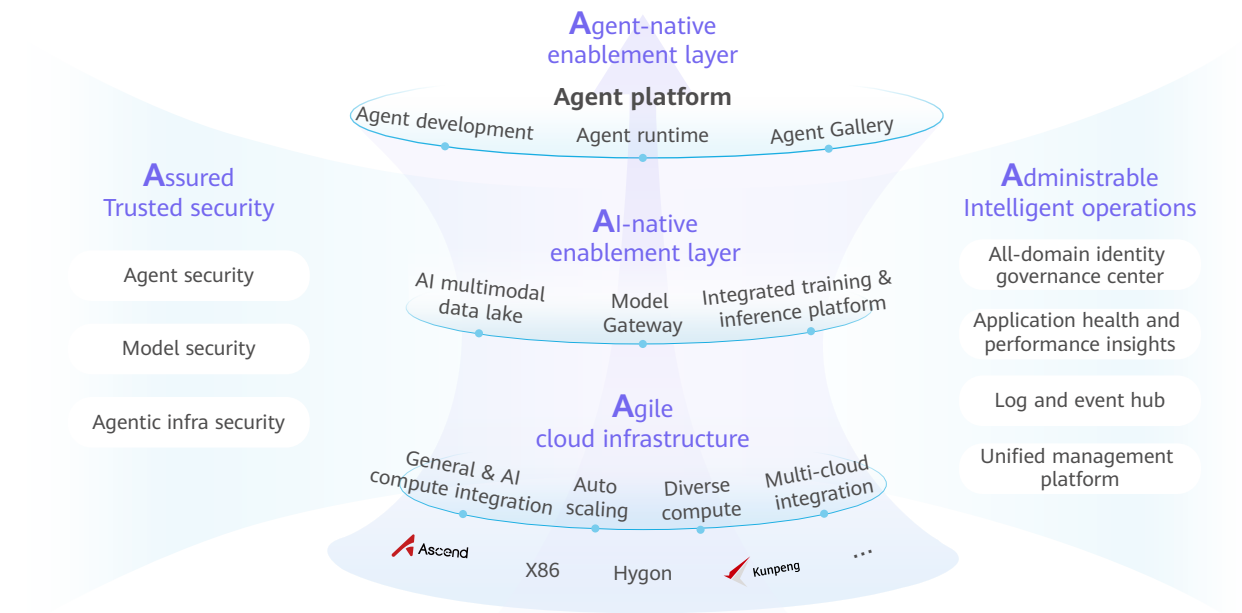


Figure 1.2 Hybrid cloud architecture for enterprise AI agents

Agile cloud infrastructure

An agile cloud infrastructure provides a unified hybrid cloud foundation centered on the enterprise's on-premises cloud while seamlessly integrating multiple clouds. It brings together general-purpose and AI compute, elastically scales with business demand, supports heterogeneous hardware and multi-generation AI processors, and delivers unified management across distributed cloud resources—ensuring a resilient and future-ready infrastructure.

AI-native enablement layer

The AI-native enablement layer serves as the enterprise's centralized AI capability hub. It establishes a unified multimodal AI data lake that integrates data lakes and data warehouses, enabling a closed-loop pipeline spanning data preparation, model training, inference, and continuous optimization. By combining high-quality proprietary enterprise data with reinforcement learning, it enables continuous model improvement while exposing standardized AI capabilities through token-based services, making AI securely accessible across the enterprise. This layer also incorporates a Model Gateway to securely connect to state-of-the-art MaaS services, MCP tools, and AI skills available in the public cloud.

Agent-native enablement layer

Built on Huawei's Harness engineering methodology and collaborative low-code and pro-code development, the agent-native enablement layer converts enterprise knowledge into AI-native assets through ontology modeling and supports intelligent business workflows across every enterprise scenario. It provides the foundation for large-scale concurrent AI agent execution, accelerating the adoption of AI across enterprise operations.

Administrable—Intelligent operations

The Administrable layer provides unified governance and operations for large-scale AI agents and token resources. Through fine-grained operational management and token resource optimization, it helps enterprises control costs, improve operational efficiency, and maximize the business value generated by AI.

Assured—Trusted security

The Assured layer establishes the security foundation for enterprise hybrid cloud deployments. It protects hybrid cloud networking and data transmission while introducing dedicated security capabilities for AI agents, ensuring end-to-end risk control, regulatory compliance, traceability, and trustworthy AI operations.

02

Agent-Native Enablement Layer



2.1 Key Trends Shaping AI Agents

A Major Shift in AI Capabilities: From Passive Reasoning to Proactive Execution and Continuous Self-Evolution

Traditional AI applications operate in a passive response mode. They rely on users to provide explicit, well-structured instructions and are primarily limited to static tasks such as question answering, content generation, and structured data retrieval. They lack the ability to plan autonomously, execute end-to-end workflows, and deliver value beyond isolated productivity improvements.

The new generation of enterprise AI agents represents a fundamental leap in capability by establishing a complete perception–memory–planning–execution–reflection loop. AI agents can interpret ambiguous business objectives, decompose complex tasks through Chain of Thought (CoT) and Tree of Thoughts (ToT) reasoning, and autonomously orchestrate APIs, databases, retrieval-augmented generation (RAG) knowledge bases, and third-party services to execute complex workflows. They continuously refine their execution strategies based on runtime feedback, enabling closed-loop execution of business workflows.

Compared with traditional automation tools, the defining advantage of AI agents lies in their ability to continuously evolve. Leveraging incremental learning, self-supervised fine-tuning, and reinforcement learning from human feedback (RLHF), AI agents continuously optimize task planning, tool calling, and scenario adaptation. Over time, they improve execution quality and operational efficiency while accumulating reusable enterprise AI assets.

Rapid Industry Adoption: From Personal Productivity to Enterprise-Scale Deployment

In 2026, the open-source AI agent ecosystem reached an inflection point with the rapid rise of two major open-source AI agent frameworks: OpenClaw and Hermes Agent. Powered by standardized agent architectures, low-code orchestration, multi-model compatibility, and rich tool ecosystems, these frameworks have become a foundation for enterprise AI agent deployment, moving beyond personal experimentation into production-scale enterprise applications.

Supported by standardized development components, enterprise-grade integration capabilities, and multi-agent systems (MAS), AI agents are rapidly expanding from personal productivity and content creation into mission-critical enterprise use cases across finance, manufacturing, government, and retail. They are increas-

ingly powering intelligent operations, risk management, production scheduling, customer service, and supply chain collaboration.

Today, AI agents have entered a new stage characterized by large-scale enterprise deployment, production workload support, and systematic adoption across business scenarios, positioning them as foundational infrastructure for enterprise digital transformation.

2.2 Key Challenges in Scaling Enterprise AI Agents

Scaling AI Agents Requires a New Generation of Development Toolchains

As AI agents move into large-scale commercial deployment, the number of deployed agents, system complexity, and task concurrency are all growing exponentially. Low-code/pro-code collaborative development, multi-agent collaboration, and dynamic orchestration across hybrid cloud environments—including public clouds, private clouds, and edge computing—are rapidly becoming the dominant deployment model. Traditional software architectures are no longer sufficient for this new paradigm.

First, insufficient support for low-code/pro-code collaboration. Developing enterprise AI agents requires multidisciplinary expertise spanning LLMs, software engineering, and business domains. Existing development environments lack lightweight visual tools that enable business users to participate effectively in AI agent development.

Second, poor standardization and reuse of AI assets. Core assets—including AI agent logic, prompts, and tool-calling policies—lack standardized packaging, lifecycle management, and centralized governance. Organizations frequently have to rebuild similar AI agents instead of reusing proven components, slowing large-scale adoption.

Third, complex integration with heterogeneous enterprise systems. AI agents must integrate with business applications, databases, external APIs, and third-party services. However, existing development tools often lack standardized connectors, unified gateways, and protocol adaptation capabilities, making integrations expensive to build and difficult to maintain.

Three-Fold Engineering Challenges for Enterprise AI Agents: Performance, Cost, and Reliability

The quality of AI agent outputs is highly sensitive to the

deployment scenario. Successful task execution depends on multiple factors, including the capabilities of the underlying AI model, context window management, the reliability of external tools, and the accuracy of RAG results. Weakness in any of these components can significantly impact the final result. Unlike traditional deterministic software, AI agents rely on dynamic reasoning, iterative decision-making, and autonomous orchestration, introducing a new generation of engineering challenges for production deployment.

In hybrid cloud environments, enterprises typically face three major challenges when deploying AI agents at scale:

Performance. Long-running, multi-step workflows are prone to context window exhaustion, inference latency, logic gaps, and hallucinated tool calls, making it difficult to satisfy the responsiveness and execution reliability required by mission-critical business workflows.

Cost. Large-scale inference, frequent tool invocation, and persistent state management place substantial demands on compute and infrastructure resources. Inference alone can account for more than 60% of the total cost of ownership (TCO) of enterprise AI agents.

Reliability. AI agent decision-making remains inherently a black box. Model hallucinations, reasoning failures, and poor adaptation to specific business use cases are compounded by the lack of unified observability, evaluation, and continuous optimization. As a result, enterprises struggle to quantify business value and continuously improve AI agent capabilities to meet enterprise reliability standards.

Security and Compliance: A Critical Challenge for Enterprise AI Agent Deployment

The autonomous nature of AI agents—including tool invocation, cross-system interaction, automated execution, and collaborative operation—fundamentally reshapes the enterprise security landscape and introduces a new generation of governance challenges.

These challenges are reflected in three key areas.

Expanded attack surfaces. AI agents interact with business systems, APIs, and external tools across complex execution paths, significantly expanding potential attack vectors and increasing the risk of prompt injection, unauthorized tool invocation, malicious instruction execution, and sensitive data leakage.

Cascading security risks. Security vulnerabilities affecting a single AI agent can rapidly propagate across interconnected systems through autonomous orchestration and cross-system collaboration, creating systemic risks for enterprise data and business operations.

Difficult accountability. Because AI agents make decisions, continuously evolve, and operate as black-box systems, tracing execution paths and assigning responsibility after a security incident becomes significantly more difficult, complicating enterprise compliance and governance.

Security and compliance are non-negotiable prerequisites for enterprise-scale AI agent deployment. Without a comprehensive security framework spanning the entire AI agent lifecycle, organizations face unacceptable business, security, and regulatory risks.

2.3 A Systematic Approach to Building Enterprise AI Agents

As enterprises enter the agentic era, application architectures are evolving from standalone systems to multi-agent ones. At the same time, growing demands for data sovereignty, business continuity, cost optimization, and agile innovation are making hybrid cloud the preferred infrastructure for enterprise AI agent deployment. To support this transformation, enterprises should establish an integrated AI agent platform spanning development, runtime, operations, security, and governance. Such a platform enables unified management and coordination across public clouds, private clouds, and edge environments, providing a consistent foundation for building, deploying, and operating AI agents at scale.

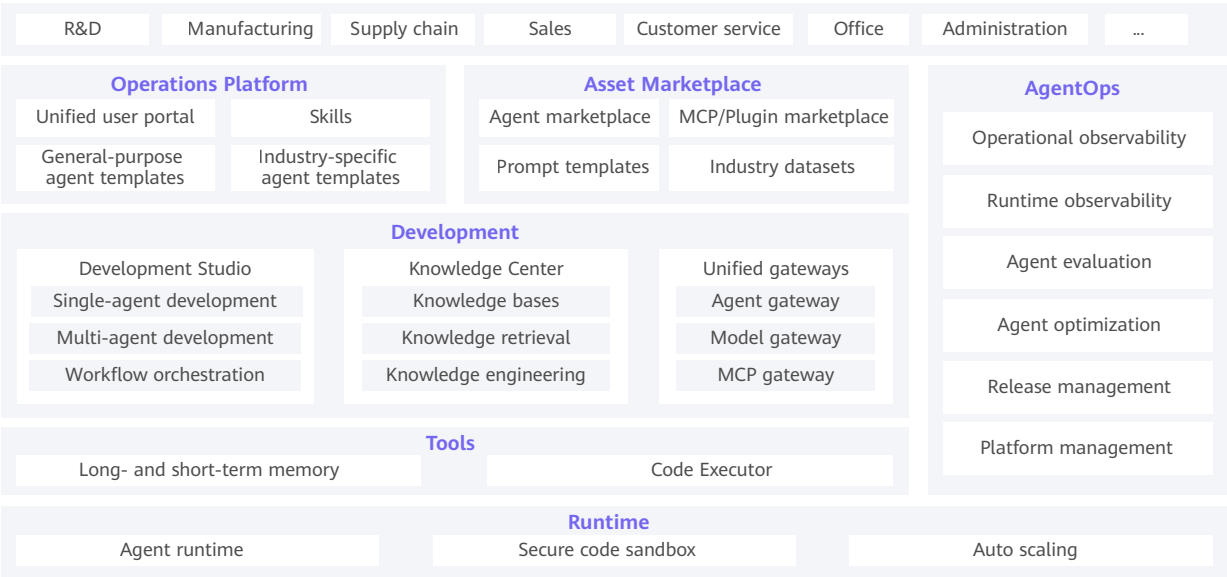


Figure 2.1 Overall architecture of the AI agent platform

Phase 1: AI agent engineering capabilities—Build a one-stop AI agent development platform with native harness integration and low-code/pro-code collaboration

To address the low development efficiency, high customization costs, and limited standardization of AI agent development in hybrid cloud environments, enterprises need a one-stop AI agent development platform covering the entire development lifecycle—from requirements definition and visual workflow orchestration to development, debugging, testing, release, and component reuse.

The platform should adopt a collaborative low-code/-pro-code development model that balances rapid delivery with deep customization.

In low-code mode, business users can quickly build standardized AI agents for common business use cases through visual drag-and-drop tools and reusable templates, significantly lowering the barrier to adoption. In pro-code mode, the platform exposes a full set of APIs and SDKs, enabling developers to build sophisticated business logic

for highly customized, mission-critical applications.

The platform should natively integrate harness engineering capabilities and provide standardized agent building blocks, reusable memory components, tool orchestration engines, and RAG enhancement modules. By standardizing development specifications and simplifying integration, it eliminates fragmented development workflows, poor interoperability, and slow iteration cycles, enabling enterprises to scale AI agent development while maximizing component reuse.

Phase 2: Elastic agent runtime—Build high-performance, highly elastic AI agent runtime environments

AI agent runtime environments are characterized by highly dynamic workloads, bursty traffic, and massive concurrency, placing stringent demands on elastic resource scheduling. The platform should support deployment across public clouds, private environments, and edge infrastructure, allowing enterprises to leverage their existing IT investments while meeting security and compliance requirements. It should also offer intelligent model routing

and a hybrid inference engine capable of orchestrating AI models with varying capabilities and deployment locations on demand. This enables enterprises to strike the optimal balance between performance, cost, and security while supporting diverse workloads.

The runtime platform adopts a dual-isolation architecture based on microVM and PicoOS to establish a multilayer security framework. At the session level, each agent session is provisioned with an isolated sandbox to ensure full isolation of session data. At the infrastructure level, virtual machine isolation establishes a hardware-enforced security boundary that effectively protects enterprise networks against malicious code and intrusions. At the same time, the platform employs an ultra-lightweight architecture that minimizes runtime overhead, maintaining both strong security and operational efficiency.

The platform further improves execution efficiency through physical cluster prewarming and end-to-end optimization backed by software-hardware synergy. Sandbox cold-start latency can be reduced to seconds and, after further optimization, to just a few hundred milliseconds. Combined with a serverless architecture, the platform delivers ultra-high concurrency, allowing hundreds of thousands of sandboxes and network connections to be created per second.

To accommodate diverse use cases, the runtime platform also supports customizable memory strategies, improving both execution efficiency and response accuracy. Short-term memory retention can be configured anywhere from one to 365 days, while long-term memory is securely persisted to preserve conversational continuity across sessions.

Phase 3: Efficient agent operations—Build an AgentOps platform for agent lifecycle management and rapid iteration

Unlike traditional software, AI agents operate as black-box systems, making their internal state difficult to observe, their performance difficult to quantify, and their optimization difficult to guide.

To address these challenges, enterprises need an AgentOps framework that provides fine-grained, end-to-end management across the entire agent lifecycle.

The platform should deliver comprehensive observability by continuously collecting telemetry from infrastructure resources, agent execution pipelines, model inference, tool invocations, decision logs, and runtime exceptions. By

reproducing the complete execution path of every task, it enables rapid fault isolation and root cause analysis. In hybrid cloud environments, the platform should further provide unified observability, evaluation, and operations across on-premises and cloud environments, preventing fragmented AI agent management from creating new operational silos.

The platform should also establish a quantitative evaluation framework with metrics covering technical stability, task completion rates, resource utilization, and employee productivity improvements. These metrics enable enterprises to measure, compare, and trace both agent performance and business value. At the same time, the platform should support continuous evolution by closing the optimization loop through operational telemetry and user feedback. AI agent capabilities can be continuously improved through incremental fine-tuning, prompt engineering, knowledge base updates, toolchain optimization, and model upgrades. Combined with automated testing and CI/CD pipelines, the platform shortens release cycles, responds rapidly to changing business requirements, and establishes a self-evolving Observe-Evaluate-Optimize-Release loop.

Beyond day-to-day operations, the platform should help enterprises transform AI agent development experience, domain knowledge, and best practices into reusable enterprise assets. By establishing a unified asset repository, organizations can systematically reuse proven capabilities and accelerate large-scale deployment.

The repository should adopt a layered architecture spanning the entire AI agent development lifecycle. At the foundation layer, it provides reusable building blocks such as prompt templates, tool connectors, APIs, and data processing components, enabling developers to rapidly assemble new AI agents. At the business layer, it stores industry-specific knowledge bases, business workflow templates, AI agent role definitions, and multi-agent collaboration strategies tailored for sectors such as finance, government, manufacturing, and retail. At the application layer, complete AI agent packages can be published, shared, and reused, allowing developers to rapidly build custom AI agents from existing ones, instead of building them from scratch.

The repository should also provide comprehensive version management, access control, search and discovery, and quality evaluation capabilities. These features support continuous asset iteration, fine-grained permission management, and efficient asset discovery while ensuring security, accuracy, and reliability. Ultimately, these capabilities enable proven AI assets to be rapidly reused across AI agents,

business use cases, and deployment environments.

Phase 4: AI agent security governance—Build an end-to-end, multi-layer security framework

Throughout the AI agent lifecycle in hybrid cloud environments, enterprises should establish an integrated security framework spanning development, deployment, runtime, and auditing to systematically mitigate emerging security risks.

During development, the platform should adopt a shift-left security approach by integrating code scanning, dependency analysis, and vulnerability detection into the development pipeline. This helps identify and eliminate configuration vulnerabilities, excessive privileges, and potential data leaks at the source.

During deployment, the platform should enforce the principle of least privilege through fine-grained access control, role-based isolation, and API allowlists, minimizing the attack surface before AI agents enter production.

During runtime, the platform should provide continuous behavior monitoring, intelligent anomaly detection, and automated risk mitigation. Built-in security guardrails, prompt injection protection, and data masking capabilities help prevent unauthorized operations, malicious tool invocation, and the cascading propagation of security incidents across interconnected AI agents.

An end-to-end security auditing framework should comprehensively record AI agent decisions, tool invocations, and operational activities, ensuring that every security incident is traceable, accountability can be clearly established, and enterprise compliance requirements are met. Where cross-cloud data movement and AI agent collaboration across heterogeneous environments are required, the platform should further provide unified identity management, access control, and data governance to enforce consistent security policies across the entire hybrid cloud environment.

Phase 5: Cloud-On-premises collaboration—Build a federated innovation framework

As enterprises scale AI agent adoption, they increasingly face a fundamental tradeoff: the pace of AI innovation versus the need to protect sensitive enterprise data.

This challenge is particularly acute in highly regulated industries such as finance and government, where stringent requirements for data security and compliance must be balanced against the rapid evolution of AI technologies.

On one hand, AI models continue to evolve at an unprecedented pace. To remain technologically competitive, industry-specific AI agents must continuously tap into the latest model capabilities, reusable AI assets, and expanding tool ecosystems. On the other hand, core business data, proprietary knowledge, and production systems must comply with strict requirements for data residency, auditability, and traceability. This means neither a purely on-premises deployment nor a cloud-only approach can fully satisfy these competing requirements.

A collaborative cloud-on-premises architecture provides a way out. By combining the security and data sovereignty of on-premises deployment with the scale, agility, and ecosystem advantages of the cloud, enterprises can establish a model of cloud-based innovation and on-premises execution. This approach effectively eliminates the traditional tradeoff between security and agility.

The architecture is built on three core mechanisms.

Cloud-based development, on-premises deployment.

Enterprises develop, test, and validate AI agents using a unified, cloud-based development platform and a shared asset repository, where they can access standardized resources such as prompt templates, tool connectors, industry knowledge bases, and pre-trained models. Mature AI agents are then deployed to on-premises environments, ensuring that sensitive business data never leaves enterprise premises.

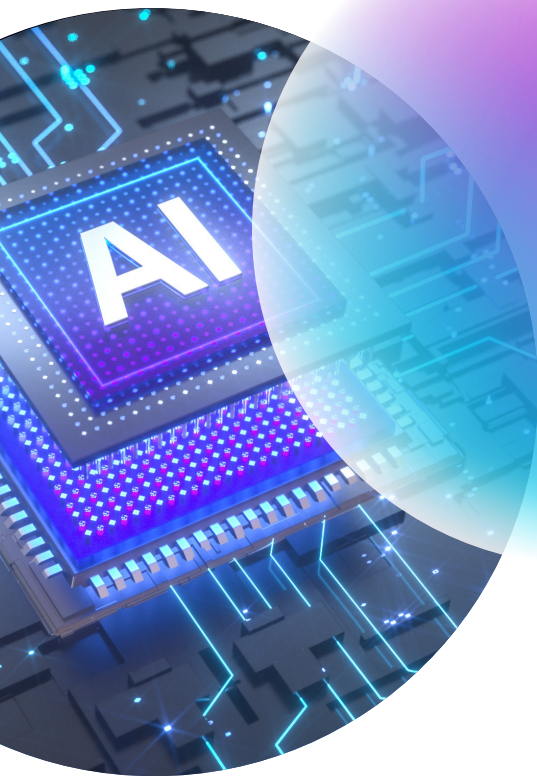
Cloud-based innovation, on-premises upgrades. The cloud offers access to the latest AI models, tool ecosystems, and algorithms. Through secure synchronization channels, these innovations are delivered to on-premises environments, enabling enterprises to keep their AI agents up to date without exposing proprietary data.

Cloud elasticity, on-premises operations. For temporary compute-intensive workloads, such as model training, large-scale inference, and simulation testing, enterprises can leverage on-demand cloud resources while keeping mission-critical production workloads on premises. This approach combines the scalability of the cloud with the security of on-premises environments, achieving an optimal balance between cost and performance.

By combining cloud-scale innovation with on-premises execution, enterprises can continuously tap into the latest advances in AI while maintaining data security and regulatory compliance. The result is an architecture that enables rapid AI agent development, continuous capability evolution, and large-scale deployment while achieving an optimal balance between security, efficiency, and cost.

03

AI-Native Enablement Layer



3.1 Model Gateway

3.1.1 Trends and Requirements

As enterprise AI agents become deeply embedded in enterprise operations, large language models have evolved into a core engine for driving operational efficiency and accelerating business innovation. This has led to an exponential surge in daily token consumption, with leading enterprises consuming hundreds of millions—and increasingly billions—of tokens every day.

Against this backdrop, enabling secure, efficient, and governed access to LLM APIs has become a critical requirement for enterprise AI adoption. Allowing employees or AI agents to invoke model provider APIs directly introduces significant security risks, uncontrolled costs, and compliance challenges. Enterprises therefore require a token management framework aligned with their organizational structure and business strategies to ensure that token resources are allocated efficiently and in accordance with governance requirements, so that every token counts.

The growing need for an enterprise Model Gateway is driven by three major trends.

Evolving application architectures: From single-model access to multi-agent collaboration

As the number of enterprise AI agents grows rapidly, multi-agent collaboration and multi-model orchestration are quickly becoming the new norm. This shift places new demands on API management, intelligent routing, circuit breaking, observability, and traffic governance, making the Model Gateway an essential component of enterprise AI infrastructure.

AI resource access: From IT departments to every employee

Traditionally, enterprise IT resources have been provisioned, managed, and governed primarily by IT organizations. In the AI era, however, tokens have emerged as a primary production resource consumed directly by business units, individual employees, and AI agents. Organizations therefore require fine-grained governance over token quotas, usage auditing, cost allocation, and access control, positioning the model gateway as the foundation for a Token-as-a-Resource management model.

Measuring AI value: From qualitative ROI to quantifiable business metrics

Enterprise AI investments are increasingly evaluated through measurable business outcomes rather than broad estimates of return on investment. A model gateway provides detailed visibility into cost and usage across models, business use cases, teams, and AI agents, enabling organizations to transform AI ROI from a subjective assessment into a set of measurable operational metrics.

3.1.2 Key Capabilities

An enterprise Model Gateway should provide three core capabilities to support secure, scalable, and cost-effective adoption of enterprise AI.

Unified access and intelligent routing

The Model Gateway abstracts multiple underlying model provider accounts into virtual API keys, providing a unified access layer for enterprise applications while eliminating uncontrolled "shadow AI" usage.

- **Model abstraction layer:** Abstracts provider-specific APIs behind a unified interface (such as an OpenAI-compatible API), reducing application development effort and minimizing vendor lock-in.
- **Intelligent routing:** Dynamically selects the optimal model based on task type, model capability, latency requirements, policy constraints, and cost budgets.
- **Fallback mechanism:** Automatically switches to backup models when the primary model becomes unavailable or exceeds latency thresholds, ensuring business continuity.
- **Multi-model orchestration:** Supports sequential, parallel, and conditional model execution patterns to accommodate complex enterprise AI workflows.

Multi-tenant identity and access management

- **Hierarchical multi-tenant architecture:** Supports platform-, enterprise-, department-, and user-level tenancy, accurately reflecting complex organizational structures.
- **Virtual API keys:** Issues independent virtual credentials to each tenant or AI agent, enabling strict permission isolation and precise usage attribution.
- **Workload identity:** Supports non-human identities such as Kubernetes Service Accounts and cloud instance roles, enabling secure authentication for enterprise applications and AI agents.
- **Multi-factor authentication (MFA) and device trust:** Enforces MFA or trusted-device verification for high-risk operations and access to sensitive data.

Cost and quota management

The Model Gateway provides comprehensive governance over token consumption through fine-grained allocation, budgeting, and cost attribution.

A multidimensional tagging framework—covering users, teams, projects, applications, and use cases—provides end-to-end visibility into token consumption, from enterprise-wide spending down to individual API requests.

- **Multidimensional tagging:** Supports automatic, mandatory, and manual tagging to ensure complete and accurate cost attribution.
- **Token plans:** Standardizes token delivery through subscription-based plans while supporting reserved quotas, prepaid, postpaid, and hybrid billing models.
- **Fine-grained quotas:** Defines token usage limits based on model type, user identity, business unit, application, and configurable time windows.
- **Dynamic budget control:** Supports multi-level budget alerts (for example, at 80%, 95%, and 100% utilization), automatically enforces policy actions when budgets are exceeded, and allows flexible budget reallocation.
- **Cost allocation and chargeback:** Aggregates usage by tags to support internal IT showback and chargeback across business units.

3.1.3 Implementation Approach

Phase 1: Centralized Token Management and Hierarchical Token Allocation

Unified Management of Model Providers

Enterprises should establish a standardized lifecycle management framework for AI model providers to ensure secure, reliable, and cost-effective access to foundation models.

- **Provider onboarding:** Establish a comprehensive model evaluation framework covering model capability, security, cost, and SLA performance to create an approved list of model providers.
- **Credential management:** Centrally manage and securely store provider API keys with automated key rotation and encrypted storage (for example, through KMS or Vault), while supporting load balancing across multiple provider accounts.
- **Health monitoring:** Continuously monitor provider

health, including latency, error rates, and traffic throttling events, and maintain dynamic health scores for each provider.

- **Provider routing:** Support provider-level failover, weighted traffic allocation for canary testing of new models, and intelligent routing policies optimized for cost and performance.
- **Lifecycle management:** Manage the complete provider lifecycle, including model version upgrades, compatibility testing, integration with contract and billing systems, and secure data cleanup when models are removed.

Differentiated Token Allocation for Employees and Applications

Enterprises should establish differentiated token allocation policies based on four dimensions: organizational roles, application scenarios, data sensitivity, and cost budgets. The following examples illustrate typical token allocation models. The figures are for illustrative purposes only.

Office Agents

Office agents are primarily used for document processing, schedule management, information retrieval, and lightweight data analysis. Assuming eight conversations per user per day, each user consumes approximately 1.2 million tokens per day.

- **Interaction characteristics:** Medium-complexity workflows involving one or two tool calls; enterprise knowledge is incorporated through Retrieval-Augmented Generation (RAG).
- **Token consumption breakdown:** A typical conversation consists of: system prompt (~1,500 tokens), RAG context (~2,000 tokens), multi-turn conversation history, and tool invocation results.

After accounting for the higher tokenization overhead of Chinese text (approximately 1.5×) and multi-step agent reasoning (approximately 2.5×), a complete conversation consumes roughly 150,000 tokens.

Coding Agents

Coding agents represent one of the most token-intensive AI workloads because they operate over large codebases while performing iterative reasoning and self-correction. Assuming four complex coding tasks per developer per day, a typical developer consumes approximately 6.8 million tokens per day.

- **Interaction characteristics:**
High-complexity workflows involving multiple reasoning steps and extensive tool usage; deep reasoning combined with large-scale code generation.
- **Token consumption breakdown:**
A typical coding task includes: Long-context codebase information (~8,000 tokens); multiple tool iterations (such as compilation, testing, and error analysis), typically around five iterations per task; and extensive Chain-of-Thought (CoT) reasoning.
Taking into account the higher token density of source code (approximately 1.8×) and the greater computational complexity (approximately 3×), a single coding task may consume as many as 1.7 million tokens.

Based on these AI agent usage patterns and token consumption models, enterprises can establish differentiated token allocation policies for different employee roles and use cases.

Level	Intended User and Use Case	Model Capability	Token Quota	Accessible Data	Approval Policy
L1 (Basic)	Common employees, basic internal Q&A	Basic lightweight model	Low (e.g., 100,000 tokens/day)	Only anonymized data	No approval required
L2 (Standard)	Key employees, daily document processing	Mainstream commercial model	Medium (e.g., 1 million tokens/day)	Internal, non-sensitive data	Approval by supervisor
L3 (Enhanced)	R&D team, code generation	Leading model + code-specific capabilities	High (e.g., 6 million tokens/day)	Internal data + code repository	Approval by project manager
L4 (Dedicated)	Core workloads, deep data analytics	Flagship model + ultra-long context	Custom	Core sensitive data	Security committee + business unit dual approval

Security and Compliance for External Model Access

Enterprises should establish a comprehensive security framework spanning the network, data, and application layers to protect enterprise AI workloads.

- **Network perimeter protection:** For highly sensitive workloads, access to LLM APIs should be established through dedicated connections or private virtual networks (VPCs) provided by cloud vendors, eliminating exposure of enterprise data over the public Internet.
- **PII masking:** Deploy an AI firewall that automatically detects and removes personally identifiable information (PII) before requests leave the enterprise network.
- **Access control and least privilege:** Implement role-based access control (RBAC) to define which models and capabilities may be accessed by each department, employee, and AI agent, ensuring strict enforcement of the principle of least privilege.

Phase 2: Fine-Grained Token Cost Management and Accurate ROI Measurement

Enterprises should establish an enterprise-wide token cost accounting framework to ensure that every token consumed is traceable, attributable, and measurable, enabling effective cost optimization and value realization.

- **Cost allocation models:** Support multiple cost

allocation methods, including allocation based on actual consumption, fixed allocation ratios, and business priorities, to meet the accuracy requirements of internal financial chargeback.

- **Multidimensional tagging:** Attach metadata such as user, department, project, agent, and use case to API keys, enabling drill-down analysis and multidimensional aggregation of token usage across all dimensions.
- **Real-time cost metering:** Calculate the cost of every request in real time using the following formula: $\text{Input Tokens} \times \text{Input Unit Price} + \text{Output Tokens} \times \text{Output Unit Price}$. This provides the data needed for real-time budget monitoring and alerting.
- **Cost baselines:** Maintain baseline cost profiles for different models across typical workloads (for example, the average cost of code generation using DeepSeek V4 Pro). These baselines enable rapid detection and interception of abnormal or unusually expensive requests.

3.2 Integrated Training & Inference Platform

3.2.1 Trends and Requirements

Enterprises typically consume foundation models in two ways: by invoking MaaS APIs or by deploying models

locally with continuous fine-tuning. These approaches are complementary rather than mutually exclusive and can be combined according to different business requirements. Today, local deployment with continuous fine-tuning remains the dominant approach in enterprise and public-sector environments, although MaaS is expected to play an increasingly important role over the long term. Under the local deployment model, enterprise AI is evolving from model-centric applications to agent-driven systems. This transition does not replace foundation models with AI agents; instead, it establishes a clear division of roles between them. AI agents are responsible for planning, memory, and action execution, while foundation models provide language understanding, reasoning, and response generation. As business requirements continue to evolve, enterprises need not only AI agents capable of reshaping business workflows, but also models continuously fine-tuned with domain expertise and industry-specific knowledge, ensuring that enterprises can continuously benefit from the latest AI capabilities.

To enable the co-evolution of models and AI agents, enterprises require a flywheel consisting of data processing → model fine-tuning → AI agent upgrades → business outcome evaluation, with evaluation results being continuously fed back into the next optimization cycle. Supporting this flywheel requires a unified, end-to-end platform that spans the entire lifecycle—from data preparation and model fine-tuning to deployment, monitoring, evaluation, and feedback. By standardizing and engineering the model development lifecycle, such a platform enables enterprises to continuously evolve their AI capabilities while responding rapidly to changing business requirements.

3.2.2 Key Capabilities

An integrated training and inference platform should provide the following core capabilities.

High-Performance Model Inference

AI agents rely on foundation models capable of delivering highly reliable, low-latency inference. To achieve this, the platform should provide the following capabilities:

- **High-performance inference:** Leverages inference acceleration techniques such as quantization, operator fusion, and speculative decoding, combined with distributed clusters and heterogeneous compute scheduling, to deliver low-latency, high-throughput, and highly reliable inference performance.

- **Elastic resource scaling:** Dynamically determines optimal resource allocation based on agent resource requirements and real-time resource utilization, enabling automatic scaling of compute resources to accommodate changing demand without service disruption.
- **Inference traffic management:** Dynamically balances request traffic across models based on real-time agent demands and traffic profiles, ensuring stable scheduling and consistent service quality under varying workload conditions.
- **Multi-version model management:** Manages the complete model version lifecycle, ensuring clear version lineage, controlled upgrades, and full traceability to historical releases.

Efficient Model Post-Training

The goal of post-training is to transform general-purpose foundation models into domain-specific AI engines that understand unique business contexts, master domain expertise, and execute complex policies, providing stronger support for enterprise AI agents. The platform should provide the following capabilities.

- **Supervised fine-tuning (SFT):** Fine-tunes foundation models using enterprise instruction datasets (such as customer service interactions, contract review examples, and code generation tasks) alongside domain-specific datasets. Combined with optimal dataset mixing and splitting policies, this helps align model outputs with enterprise-specific formats and quality standards, producing proprietary enterprise knowledge models.
- **Reinforcement learning:** Builds upon supervised fine-tuning using curated reasoning datasets to further improve domain understanding, analytical capabilities, and complex decision-making, ultimately producing enterprise-grade expert models.

High-Quality Dataset Production

Both enterprise model post-training and AI agent deployment depend on high-quality datasets. The platform should therefore provide the following capabilities.

- **Data governance:** Builds upon an enterprise multimodal AI data lake to provide unified governance for both structured data (such as tables) and unstructured data, including text, images, video, and audio. Intelligent operators and workflow orchestration automate feature extraction, storage, and reuse, ensuring a reliable supply of high-quality training data.

- **Data synthesis and augmentation:** Generates high-quality post-training and SFT datasets through AI-assisted sampling, data synthesis, and multi-stage data quality validation, effectively augmenting enterprise training datasets.

3.2.3 Implementation Approach

Enterprise AI adoption is an incremental journey. Rather than attempting enterprise-wide deployment from the outset, organizations should build capabilities progressively through a structured implementation roadmap. The recommended approach consists of three phases. The first phase focuses on building a unified platform that integrates data preparation, model development, model deployment, and business feedback into a single engineering workflow, preventing fragmented technology silos. The second phase introduces an iterative deployment strategy, progressing from pilot deployments to large-scale adoption while continuously refining governance practices. The third phase emphasizes scaling enterprise adoption reliably, overseen by a cross-functional AI governance team that defines technical standards, reviews models and AI agents, and creates a closed bidirectional feedback loop between business users and AI development teams. The following implementation roadmap is recommended.

Phase 1: Build a Unified Platform to Enable Technology Sharing

Establish a unified training and inference platform built on mainstream AI development frameworks compatible with diverse AI compute infrastructures. Standardizing the enterprise AI technology stack enables different business units to develop AI agents on a common foundation, promoting cross-team technology sharing and preventing architectural fragmentation.

The platform should also establish unified standards for transforming business data into model training corpora. Agents developed for similar use cases should share normalized, template-based datasets, creating high-quality data assets that can be reused across projects.

In addition, the platform should define a unified model evaluation framework so that AI agent development teams and model development teams share consistent performance objectives and evaluation criteria, preventing optimization efforts from diverging or creating negative feedback loops.

Phase 2: Launch Pilot Use Cases and Establish a Standard Architecture

Begin with pilot scenarios that have clearly defined workflows, well-understood decision logic, and short feedback cycles. These early deployments provide an opportunity to identify the optimal combination of foundation models, AI agents, and supporting tools for each business scenario, gradually establishing standardized Scenario-Model-Tool mappings that can evolve over time. At the same time, enterprises should standardize reusable development assets—including tool calling templates, prompt templates, memory interfaces, and other core AI agent components—so that all development teams follow a consistent architectural approach and model evolution roadmap.

Phase 3: Scale Enterprise Adoption and Establish a Closed Feedback Loop

As AI adoption expands, enterprises should establish a centralized AI governance team comprising business architects, data specialists, model engineers, AI agent specialists, and compliance experts. This team should be responsible for defining technical standards, maintaining the platform, reviewing models and AI agents, and overseeing enterprise-wide AI governance, ensuring that AI adoption scales in a standardized, secure, and well-governed manner. In the meantime, a bidirectional feedback mechanism should be established between business users and AI development teams. Business outcomes should continuously feed back into model optimization, enabling foundation models and AI agents to become progressively better aligned with evolving business requirements and domain knowledge.



3.3 AI Multimodal Data Lake

3.3.1 Trends and Requirements

As foundation models and AI agents are increasingly deployed at enterprise scale, the role of enterprise data platforms is undergoing a fundamental transformation. Traditionally, data platforms were designed primarily to support business intelligence, operational decision-making, and digital business processes. In the AI era, however, they are evolving into intelligent data infrastructures that power enterprise AI. At the heart of this transformation is a fundamental shift in how enterprise data is consumed. Rather than serving primarily BI reporting and analytical workloads, data platforms must now support model training, continuous fine-tuning, RAG, and AI agent reasoning and orchestration. Traditional lakehouse platforms, designed primarily for structured data and CPU-based processing, are no longer sufficient. Their limitations in supported data types, processing latency, compute architecture, and scalability prevent them from meeting the data requirements of modern AI pipelines.

Today, enterprises require a new generation of intelligent data platforms with the following capabilities:

- **Enterprise-wide multimodal data engineering:** Processes structured data together with text, images, audio, and video to produce standardized multimodal training datasets while supporting cross-modal retrieval.
- **Unified scheduling of heterogeneous compute resources:** Efficiently orchestrates heterogeneous, multi-generation compute resources to support AI workloads such as data preprocessing, feature extraction, and model inference.
- **Unified data management with decoupled storage and compute:** Combines storage-compute decoupling with unified metadata management to enable cost-effective storage of massive datasets while supporting enterprise-wide data governance.
- **Dedicated storage for AI agents:** Stores agent interaction state, memory, and contextual information, creating a feedback loop that continuously improves model performance and agent capabilities.
- **Hybrid cloud workload orchestration:** Dynamically schedules AI workloads across hybrid cloud environments by offloading compute-intensive but low-sensitivity tasks—such as dataset preprocessing and offline model training—to public clouds, while keeping sensitive enterprise data on premises to ensure security, compliance, and operational control.

3.3.2 Key Capabilities

To support three core objectives—high-quality AI dataset production, foundation model training, and AI agent inference—the next-generation AI multimodal data lake combines big data and AI technologies to deliver five core capabilities. Together, they provide an end-to-end data pipeline spanning data ingestion, data processing and governance, dataset construction, model training, and AI agent inference, enabling enterprise data platforms to evolve from supporting business analytics to powering intelligent content generation and autonomous decision-making.

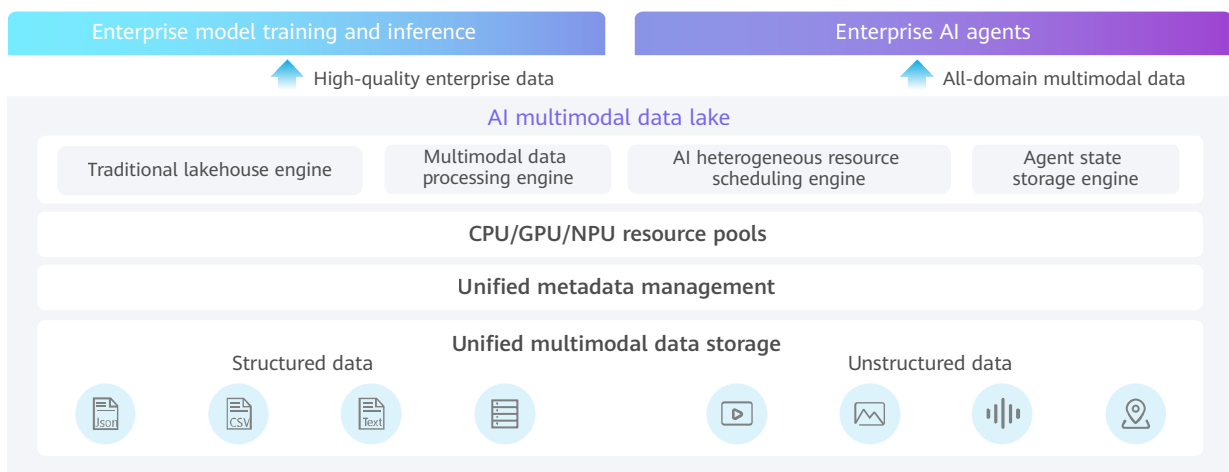


Figure 3.1 AI multimodal data lake architecture

Multimodal Data Processing and Cross-Modal Retrieval

The platform integrates a multimodal data processing engine capable of processing both structured data—including databases and tables—and unstructured data such as text, images, audio, video, and documents through a unified pipeline. It supports the complete data preparation workflow, including data cleaning, transformation, feature extraction, and data annotation, enabling large-scale production of standardized, high-quality multimodal training datasets.

Built on a unified vector data lake, the platform supports vectorized storage of multimodal data together with cross-modal retrieval, enabling AI engineers to efficiently discover training samples while allowing AI agents to rapidly retrieve semantically relevant data during inference.

To accommodate different development workflows, the platform supports multiple programming paradigms, including SQL, Python, and DataFrame APIs, enabling seamless collaboration between data engineers and AI engineers. Native compatibility with mainstream open-source AI ecosystems further reduces environment adaptation effort while improving the efficiency of dataset preparation, model development, and AI agent engineering.

Unified Scheduling of Heterogeneous Compute with a Distributed Framework

The platform employs Ray—or an AI scheduling framework with equivalent capabilities—as its distributed scheduling engine to build an enterprise-grade heterogeneous compute scheduling platform. CPU, GPU, and NPU resources are managed through a unified scheduling framework with elastic scaling, overcoming the limitations of single-node computing by distributing Python workloads, multimodal processing tasks, and model inference jobs seamlessly across cluster nodes.

The scheduler automatically assigns the most appropriate compute resources to each workload, routing general-purpose tasks to CPU pools while assigning AI workloads to GPU or NPU clusters. It also provides resource quotas, workload prioritization, and automatic failure recovery, improving overall resource utilization while ensuring the reliability of mission-critical AI workloads.

Cloud-Native Storage-Compute Decoupling and Unified Metadata Management

The multimodal data processing engine adopts a cloud-native storage-compute decoupled architecture to support

cost-effective storage and elastic processing of massive unstructured datasets.

Existing enterprise lakehouse platforms may be modernized to adopt the same architecture, enabling unified management of both legacy structured data and newly generated unstructured data. Object storage serves as the underlying storage layer for foundation model training datasets and raw multimodal data, providing petabyte-to-exabyte scalability while substantially reducing storage costs.

Compute resources are provisioned through a serverless architecture that enables second-level elastic scaling to accommodate temporary demand spikes during large-scale dataset generation and centralized model training.

A unified metadata service manages datasets, database tables, and vector indexes across the platform, enabling multiple compute engines to share data consistently while ensuring accurate cross-source data access for models and AI agents.

Dedicated Agent State and Memory Storage

The platform incorporates a dedicated AI agent state storage engine for persistent storage of conversational context, interactions, task state, and execution logs generated during agent-model interaction. The storage engine supports millisecond-level read/write latency, incremental updates, and semantic retrieval. Beyond maintaining execution continuity, accumulated interaction data can be continuously curated into valuable training datasets for subsequent model retraining, forming a closed feedback loop of AI agent execution → data accumulation → model optimization, thereby enabling continuous improvement of enterprise AI capabilities.

Hybrid Cloud Task Scheduling and Tiered Workload Execution

The platform supports cross-cloud workload scheduling based on data sensitivity and workload characteristics. Elastic workloads with relatively low security requirements—such as large-scale multimodal dataset preprocessing and offline foundation model training—can be dynamically offloaded to a public cloud to leverage on-demand compute resources, which are released immediately upon task completion. Security-critical workloads—including confidential dataset processing and real-time inference for mission-critical AI agents—remain strictly within on-premises infrastructure. The platform provides unified cross-cloud monitoring, state synchronization, and alerting

for both data preparation and model inference tasks, enabling enterprises to balance compute costs, operational elasticity, and data security through centralized workload governance.

3.3.3 Implementation Approach

The implementation of an AI multimodal data lake should follow three guiding principles: maximize reuse of existing infrastructure, evolve incrementally, and prioritize AI-native capabilities. Rather than replacing existing lakehouse platforms, organizations should progressively enhance them with AI-native capabilities, transforming traditional analytics platforms into dedicated data infrastructures for foundation model training and AI agent inference while protecting existing IT investments and ensuring uninterrupted business operations. The recommended implementation roadmap consists of three phases.

Phase 1: Build the Core AI Multimodal Data Lake

Phase 1 focuses on establishing the core capabilities of the AI multimodal data lake. Enterprises should introduce a multimodal data processing engine together with a heterogeneous AI scheduling engine to expand data processing beyond traditional structured datasets and unlock the value of vast amounts of previously underutilized unstructured enterprise data. The platform should support unified processing and correlation analysis across structured and unstructured data, while building automated multimodal data preparation pipelines capable of producing standardized AI training datasets at scale.

A unified vector data lake should also be established to support cross-modal retrieval.

At the infrastructure level, the heterogeneous scheduling engine should integrate CPU, GPU, and NPU resources into a unified compute pool capable of efficiently supporting multimodal parsing, feature vectorization, large-scale parallel dataset generation, and model inference workloads. Upon completion of this phase, the platform will be capable of producing high-quality multimodal datasets at scale, providing a solid data foundation for foundation model training and continuous model optimization.

Phase 2: Modernize the Data Infrastructure Through Storage-Compute Decoupling (Optional)

While AI capabilities are being introduced, existing production workloads must remain stable. Enterprises should therefore determine, based on their own business requirements, whether to modernize existing lakehouse platforms by migrating structured datasets to object storage and adopting a storage-compute decoupled architecture. Where this modernization is implemented, a unified metadata service provides unified management of both structured and unstructured data while enabling cross-source data discovery and querying. It also removes metadata silos between legacy lakehouse platforms and newly introduced AI data pipelines, allowing multiple compute engines to share data through unified governance while maintaining consistent dataset standards across the enterprise.

Phase 3: Close the Loop for Enterprise AI

The final phase establishes a closed-loop workflow spanning data production → dataset provisioning → foundation model training → AI agent inference.

A dedicated AI agent state and memory management engine stores agent states, conversational memory, and task execution logs, allowing valuable interaction data to accumulate as high-quality datasets for future model retraining.

At the same time, hybrid cloud workload scheduling offloads elastic, low-sensitivity workloads to a public cloud to optimize compute costs, while comprehensive data security, dynamic data masking, and end-to-end auditing ensure that core enterprise data remains protected throughout the AI lifecycle.

With these capabilities in place, the platform supports the complete AI data lifecycle—from data ingestion and governance to foundation model training and AI agent inference—continuously delivering high-quality datasets that drive ongoing model optimization and increasingly capable AI agents. The result is a smooth evolution of enterprise data platforms from traditional analytics architectures to AI-native multimodal data lakes, delivering a solid foundation for long-term enterprise AI innovation.

04

Agile Cloud Infrastructure



4.1 Trends and Requirements

As AI continues to advance rapidly, enterprise IT architectures are evolving from application-driven systems to AI-driven architectures, and from isolated model invocation to multi-agent collaboration. Traditional cloud infrastructure was designed primarily for transient, request-response workloads such as web applications and stateless microservices. While highly effective at service orchestration and resource management, it cannot support AI agents' defining characteristics: long-running tasks, persistent state, multi-turn interactions, and chained tool invocation. The lack of native state management, persistent contextual awareness, and cross-system orchestration capabilities makes traditional cloud platforms inadequate for reliable, production-scale AI agent deployment and governance. To address these challenges, enterprises need to build an agile, cloud-native infrastructure that natively supports long-running workflow orchestration, persistent state management, contextual memory, tool governance, and end-to-end observability. At the same time, the infrastructure must support unified scheduling of general-purpose and AI compute resources, allowing general-purpose compute to handle conventional application logic while AI compute accelerates model inference and intelligent decision-making. This transforms enterprise IT infrastructure from a conventional resource pool into an integrated compute foundation for both traditional and AI workloads.

Large-scale enterprise AI agent adoption is driving four major trends in infrastructure architecture:

The convergence of general-purpose and AI compute:

General-purpose and AI compute resources increasingly operate as tightly integrated components of a unified infrastructure, jointly supporting end-to-end multi-agent workflows.

Fine-grained elastic scaling for AI agents:

Infrastructure must dynamically allocate and auto-scale both general-purpose and AI compute resources to accommodate the highly variable, bursty compute demands of AI agents throughout different execution steps.

Unified management of heterogeneous compute:

Enterprises require a unified scheduling framework capable of orchestrating multi-generation, heterogeneous hardware platforms (CPUs, GPUs, and NPUs), eliminating resource silos while maximizing overall infrastructure utilization.

Multi-cloud collaboration as the enterprise norm:

Private clouds continue to host security-sensitive workloads, while public clouds provide access to frontier AI models and elastic compute capacity, enabling AI agents to operate seamlessly across hybrid cloud environments.

Overall, traditional static cloud architectures are no longer sufficient for AI-native production workloads. Enterprises must evolve toward an agile, cloud-native infrastructure that is ready for the agentic era.

4.2 Key Capabilities

The agentic era places fundamentally new demands on enterprise cloud infrastructure: more diverse compute, unified infrastructure management, more intelligent scheduling, and more elastic resource provisioning. By combining four core capabilities—integrated management of general-purpose and AI compute, elastic scaling across general-purpose and AI compute, unified scheduling of heterogeneous hardware, and unified multi-cloud compute management—an agile, cloud-native infrastructure provides a foundation purpose-built for AI agents. It abstracts away the complexity of heterogeneous hardware while maximizing the utilization of every available compute resource. AI agents are therefore no longer constrained by individual cloud environments, specific hardware platforms, or fixed infrastructure capacity. Instead, they can dynamically access the most appropriate compute resources for each workload. Within such an infrastructure, compute moves with demand, resources are allocated optimally, and AI agents execute at maximum efficiency.

Integrated Management of General-Purpose and AI Compute

AI agents do not rely exclusively on AI accelerators. Instead, they require close coordination between general-purpose compute and AI compute throughout the entire execution pipeline. Tasks such as model inference, vector retrieval, image recognition, and speech understanding depend on GPUs, NPUs, and other AI accelerators, while workflow orchestration, business logic execution, data preprocessing, API interactions, and application services continue to rely on CPUs, memory, storage, and networking resources.

An agile, cloud-native infrastructure therefore requires a unified resource management framework that manages CPUs, GPUs, NPUs, storage, and networking within a common scheduling architecture. Based on workload characteristics—including task type, execution priority,

latency requirements, and resource footprints—the platform automatically assembles the optimal combination of compute resources, enabling general-purpose and AI compute to operate as a unified, coordinated system. This enables AI agents to execute end-to-end workflows—from perception and understanding to reasoning and action—on a single integrated compute foundation.

Elastic Scaling Across General-Purpose and AI Compute

AI agent workloads are inherently dynamic and unpredictable. Compute demand can fluctuate dramatically depending on workload characteristics, user concurrency, and task complexity. Large-scale conversational AI, complex tool orchestration, multi-agent collaboration, and batch content generation can rapidly increase demand for both AI and general-purpose compute, while utilization may decline substantially during off-peak hours.

To address this variability, the infrastructure must support unified elastic scaling across both general-purpose and AI compute. The platform continuously monitors metrics such as task queue length, model invocation frequency, GPU and NPU utilization, CPU load, and response latency, and automatically scales infrastructure up or down as demand changes. Compute resources thus shift from fixed allocations to elastic, on-demand capacity that AI agents can acquire, scale, and release as workloads require. This enables higher concurrency, more complex workloads, and lower operational costs.

Unified Management of Multi-Generation

Heterogeneous Hardware

Rapid advances in AI compute have introduced multiple generations of computing hardware into enterprise environments. Next-generation AI supernode architectures, together with high-speed interconnect technologies, significantly improve communication bandwidth, reduce cross-node latency, and deliver substantially stronger distributed computing performance than traditional compute clusters. At the same time, most enterprises continue to operate a mixture of legacy CPU servers, earlier-generation GPU and NPU clusters, newly deployed AI accelerators, and next-generation AI supernodes. Differences in hardware generations, architectures, vendors,

software stacks, and scheduling mechanisms frequently result in fragmented resource pools and underutilized resources.

To address this challenge, the platform should abstract heterogeneous compute resources behind a unified resource management layer that accommodates multiple hardware generations within a common scheduling framework. Hardware capabilities—including compute performance, accelerator memory, instruction sets, supported software frameworks, and energy efficiency—are standardized as resource attributes that the scheduler can evaluate when assigning workloads. This approach allows enterprises to fully exploit the performance advantages of next-generation hardware while continuing to utilize existing infrastructure for suitable workloads, maximizing both resource utilization and return on infrastructure investment.

Unified Multi-Cloud Compute Management

Modern enterprise AI environments increasingly span private clouds, public clouds, and dedicated cloud infrastructures. To fully utilize these resources, enterprises require a unified management framework that eliminates resource fragmentation across cloud platforms. Through a unified resource view, common scheduling policies, and centralized permission control, the platform abstracts differences between underlying cloud providers and enables consistent discovery, allocation, and operation of compute resources across multiple cloud environments.

This unified approach allows enterprises to maximize utilization of existing infrastructure while avoiding isolated compute islands and idle capacity. When resources become constrained in one cloud environment, AI workloads can be automatically redirected to alternative resource pools. Likewise, workload placement can be optimized according to security, cost, or performance requirements by selecting the most appropriate cloud environment for execution. The result is a unified compute infrastructure in which multi-cloud resources are fully visible, centrally managed, and available on demand wherever workloads execute.



4.3 Implementation Approach

The implementation of an agile, cloud-native infrastructure should follow a progressive roadmap: integration → orchestration → scheduling → optimization. Rather than replacing existing infrastructure, organizations should incrementally evolve toward an AI agent-native compute architecture while maintaining compatibility with existing environments. This phased approach builds a scalable, resilient compute foundation for enterprise-wide AI agent deployment.

Phase 1: Build a Unified Pool of General-Purpose and AI Compute

Phase 1 focuses on consolidating enterprise compute resources, such as CPUs and NPUs, into a unified resource pool. Existing general-purpose infrastructure and AI accelerators should be integrated under a common resource management framework that abstracts heterogeneous compute resources into a single compute foundation. Through standardized service interfaces and resource abstraction, the platform shields applications from underlying hardware differences while providing AI agents with a unified, schedulable, and measurable compute foundation.

Phase 2: Enable Elastic Scaling Across General-Purpose and AI Compute

Phase 2 introduces intelligent scheduling policies driven by AI agent workload characteristics. Scheduling decisions should consider workload concurrency, execution duration, model invocation frequency, and workload priority to dynamically allocate both general-purpose and AI compute resources based on actual demand. With unified elastic scaling across both general-purpose and AI compute, resources can be provisioned on demand, expanded dynamically, reclaimed automatically, and continuously reused. This strengthens resource availability during workload peaks while minimizing idle capacity during off-peak periods, maximizing overall infrastructure utilization.

Phase 3: Unify Training and Inference Compute Scheduling

Phase 3 establishes a unified scheduling architecture that supports the complete AI agent lifecycle: foundation model fine-tuning, model evaluation, validation, and large-scale multi-agent online inference. General-purpose and AI compute resources spanning multiple hardware generations should be incorporated into a unified scheduling framework. Based on workload priority, latency requirements, resource utilization, and SLA requirements, the platform dynamically orchestrates training and inference workloads while enabling coordinated allocation and shared utilization of compute resources. Through resource isolation, priority management, and intelligent scheduling, the platform ensures that latency-sensitive online inference workloads maintain high responsiveness and availability while maximizing resource utilization for offline workloads such as model training and incremental fine-tuning. This approach minimizes resource fragmentation, workload contention, and duplicate infrastructure investments.

Phase 4: Integrate Multi-Cloud Resources and Continuously Optimize End-to-End Performance

The final phase extends the compute foundation across multiple cloud environments. Building upon a unified multi-cloud architecture, enterprises should integrate compute resources across private and public clouds while continuously optimizing inference pipelines, communication efficiency, caching mechanisms, and workload distribution. These optimizations improve responsiveness for demanding AI agent scenarios such as high-frequency interactions, multi-turn reasoning, and large-scale multi-agent collaboration. At the same time, workload-specific compute profiles can be created to enable tiered compute provisioning that matches infrastructure resources to workload characteristics. Through ongoing optimization of multi-cloud coordination, enterprises improve the performance, stability, and cost efficiency of the AI agent infrastructure while ensuring it continues to evolve alongside changing business requirements.



05

Assured—Trusted Security



5.1 Trends and Requirements

As enterprise AI architectures continue to evolve, deployment models are shifting from dedicated, privately owned compute infrastructure toward hybrid cloud environments that combine the elasticity of the public cloud with the security and control of the private cloud. While significantly reducing AI infrastructure costs and improving compute utilization, this approach also fundamentally dissolves the traditional physical security perimeter. Training datasets, model parameters, and inference requests now move frequently across private clouds, public clouds, and edge environments. Without proper safeguards, data is exposed to potential leakage or unauthorized access during transmission, storage, and computation.

Traditional security systems based on network isolation and access control are no longer sufficient in hybrid cloud environments. Differences in security standards, management models, and interfaces across cloud providers make it difficult to enforce consistent security policies. For highly regulated industries such as financial services, healthcare, and government, the primary challenge is no longer simply protecting sensitive information, but how to securely leverage public cloud AI capabilities while ensuring that confidential data never leaves trusted environments.

Today, enterprises typically have the following security concerns:

- Whether cloud service providers can access their customers' proprietary training data and AI models.
- Whether training and inference data remains fully under enterprise control throughout the AI lifecycle.
- Whether model weights can be stolen, copied, or otherwise compromised.
- Whether data exchange between public cloud AI platforms and private cloud or on-premises environments remains secure and fully controllable.
- Whether highly sensitive AI workloads can operate within fully isolated execution environments.
- Whether AI agents can bypass security controls and perform high-risk operations that could compromise enterprise systems or data.
- Whether abnormal AI agent behavior can be fully traced to its originating prompts, decisions, and execution path for investigation and accountability.

5.2 Key Capabilities

End-to-End Encrypted Data Flow with Customer-Held Encryption Keys

To address the security challenges introduced by hybrid cloud AI—including cross-domain data governance, inference privacy, and end-to-end trustworthiness—the recommended architecture adopts an end-to-end encrypted data flow with customer-held encryption keys. Designed for hybrid cloud deployments spanning public clouds, dedicated clouds, and edge environments, this security architecture combines five core capabilities: end-to-end encryption, Hold Your Own Key (HYOK), automatic data destruction upon leaving trusted environments, end-to-end data flow observability, and confidential inference.

Together, these capabilities establish a comprehensive security framework that protects data throughout its entire lifecycle and across every stage of its movement, while simultaneously preserving deployment flexibility, compute efficiency, and regulatory compliance.

Trusted AI Agents with Secure Model Inputs and Outputs

Enterprises should establish a trusted development and runtime infrastructure for AI agents that enforces security from the hardware layer through the application layer. Security policies enforced at the infrastructure level protect both hardware and system software, preventing unauthorized access to or tampering with AI agents, applications, and enterprise data. Based on each AI agent's operational role and applicable business policies, the platform should define fine-grained identities and permissions, continuously monitor for abnormal behavior, automatically block unauthorized operations, and maintain comprehensive security audit trails that support rapid investigation and forensic analysis when incidents occur.

Protection must also extend to the model input layer and its execution context. Coordinated input and context guardrails should continuously detect and block prompt injection, jailbreak attempts, multi-turn manipulation attacks, and attempts to bypass role restrictions. At the same time, contextual elements—including system prompts, tool outputs, and conversation history—should be validated for integrity and consistency before entering the inference pipeline. This prevents injected or contaminated context from influencing model reasoning, thereby preserving decision stability and behavioral reliability.

AI Infrastructure Security with Hardware-Rooted Trust and End-to-End Verifiability

A secure hybrid cloud AI infrastructure should be built around one security operations center supported by seven layers of defense, forming a comprehensive defense-in-depth architecture. The security operations

center provides capabilities such as a security data lake, intelligent security analytics, advanced threat detection, asset risk management, security posture monitoring, and security orchestration, automation, and response (SOAR). Together, these capabilities enable rapid, automated detection and response to security threats. The seven layers of defense are as follows: physical security, identity and authentication, network security, application security, host security, data security, and operations security. Working together with the centralized security operations center, these layers provide coordinated, end-to-end protection across enterprise AI environments.

As AI infrastructure evolves from isolated private deployments toward hybrid and multi-cloud architectures, the enterprise trust model must evolve as well. Trust can

no longer be confined to internally controlled environments but must extend across distributed infrastructure owned and operated by multiple parties. Once compute resources are no longer controlled by a single entity, every layer of the supply chain—from silicon and firmware, to network transport, virtualization platforms, and AI application containers—can become a potential attack surface.

Traditional trust models, which anchor trust primarily in the operating system or cloud management platform, are no longer sufficient for AI-native infrastructure. The industry is therefore moving toward a new trust architecture founded on hardware-rooted trust, coordinated security across software, hardware, and silicon, and end-to-end verifiable trust throughout the entire AI technology stack.

5.3 Implementation Approach

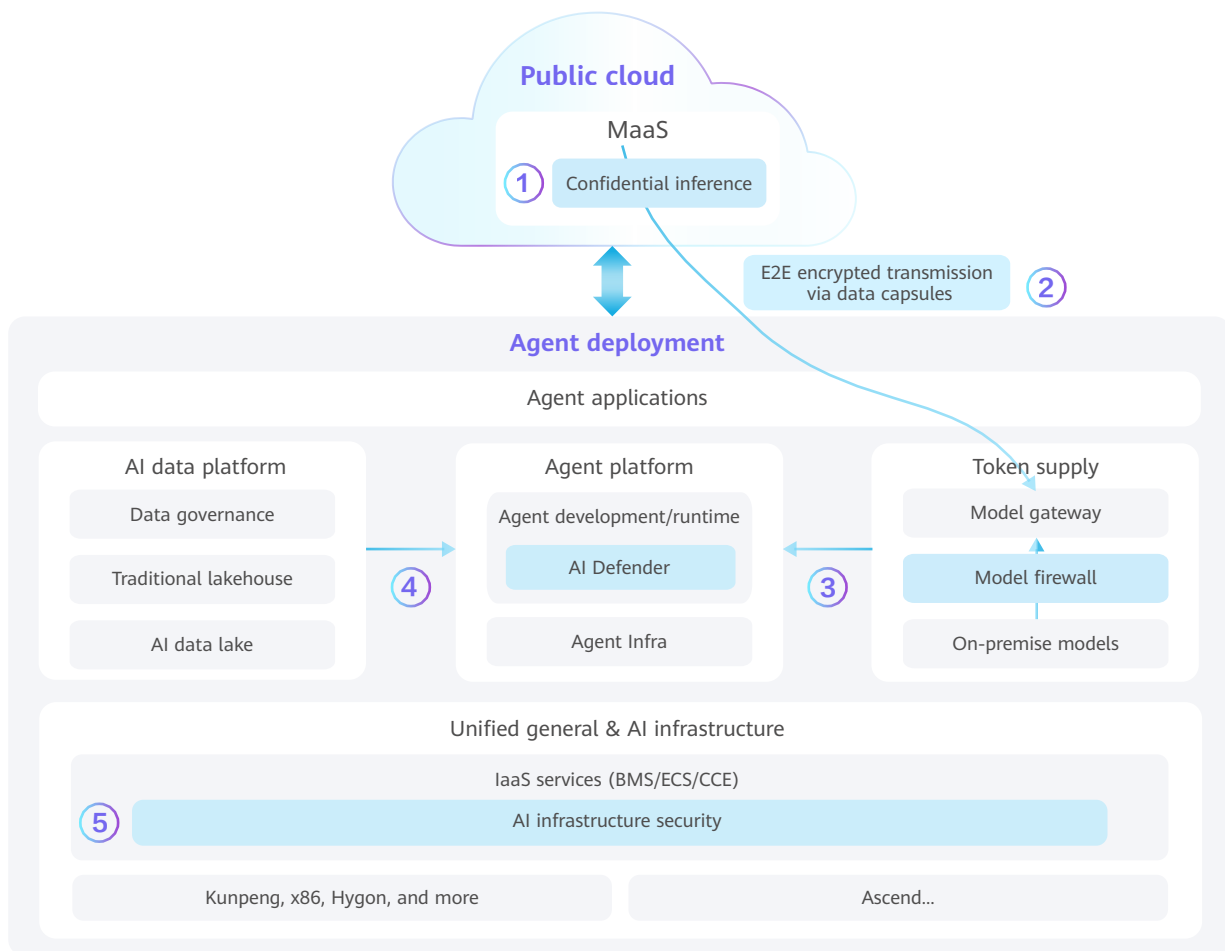


Figure 5.1 Security architecture of an AI agent-ready hybrid cloud

Phase 1: Secure the Integrated General-Purpose and AI Infrastructure

Phase 1 focuses on security for the integrated general-purpose and AI compute infrastructure. The security architecture should be designed in accordance with applicable cybersecurity requirements and built around one centralized security operations center supported by seven layers of defense. Together, these capabilities provide continuous security posture awareness while reinforcing defense-in-depth across the entire infrastructure.

This security foundation should be strengthened through three complementary capabilities: AI asset protection, trusted execution environments, and comprehensive threat protection.

A unified AI asset inventory continuously manages foundation models, datasets, and API endpoints while detecting backdoors, known vulnerabilities (CVEs), and other security risks. Integrated monitoring of data flows automatically blocks abnormal outbound data transfers.

Trusted execution environments combine hardware-rooted trust—including built-in Trusted Execution Environments (TEE) and secure boot chains—with lightweight sandboxes capable of millisecond-level startup to support low-latency AI agent execution.

Infrastructure security should further protect virtualization platforms and container environments against known vulnerabilities, defend AI applications against threats outlined in the OWASP Top 10 for Large Language Model Applications, and employ AI security agents for intelligent threat detection, forming a closed-loop protection system.

Phase 2: Secure Models, Data, and AI Agents

Phase 2 extends security to foundation models, enterprise data, and AI agents. To defend foundation models against prompt injection, malicious intent, and privilege escalation, enterprises should implement a three-layer defense architecture consisting of input filtering, behavioral analysis, and role-based access control.

Prompt protection combines prompt complexity analysis, rule-based detection, and machine learning techniques to identify both direct and indirect prompt injection attacks.

Intent-based threat detection should analyze complete AI agent execution sequences rather than individual requests,

enabling the platform to recognize suspicious combinations of actions—for example, reading confidential data, querying sensitive systems, and initiating financial transactions within the same workflow. Security checkpoints positioned at critical execution stages can automatically suspend suspicious workflows within isolated sandbox environments pending further verification.

AI agents should also operate according to the principle of least privilege. Office productivity agents should be restricted to administrative tasks such as scheduling, while software engineering agents should be prevented from directly deploying code into production environments. Behavioral models continuously detect operations that deviate from predefined role boundaries—such as an office assistant attempting to write production code or an engineering agent initiating financial transactions—and automatically reduce privileges or block execution.

Phase 3: Establish End-to-End Security Across Hybrid Cloud Environments

Phase 3 establishes an integrated security architecture spanning both on-premises and cloud environments through end-to-end encrypted data movement, customer-held encryption keys, automatic data destruction outside trusted domains, and security zone isolation.

End-to-end encryption ensures that data remains encrypted throughout its entire lifecycle. Using Hold Your Own Key (HYOK), enterprises retain exclusive control over encryption keys, while proprietary enterprise models can be encrypted and obfuscated before being deployed for cloud-based inference, mitigating security risks while leveraging the cost advantage of the cloud.

Sensitive datasets are encapsulated within secure data capsules governed by unified security policies. These datasets remain accessible only within trusted private cloud environments and designated trusted cloud zones. Once data leaves these trusted environments, access is automatically revoked, and temporary data is securely destroyed after task completion.

A security isolation zone positioned in front of AI workloads scans all foundation models, skills, and AI agent code before execution. AI security agents continuously monitor API interactions across both cloud and on-premises environments, ensuring controlled onboarding of AI assets while proactively detecting and blocking abnormal behavior across the hybrid cloud infrastructure.



06

Administrable—Intelligent Operations

6.1 Trends and Requirements

As AI agents move toward large-scale, cluster-based deployment, traditional enterprise operations practices are no longer sufficient to handle their unique operational characteristics. Conventional IT operations focus primarily on monitoring the health of the underlying infrastructure—including servers, networks, and container platforms. In contrast, the primary risks associated with AI agents stem from abnormal execution behavior rather than infrastructure failures. These include model hallucinations, improper tool usage, unauthorized actions beyond assigned permissions, context contamination, infinite execution loops, and erroneous decision-making. Because such failures often occur without any observable hardware or system faults, they are difficult to detect, diagnose, and remediate, significantly increasing operational risk.

Enterprise AI operations are shifting from traditional resource operations toward business operations and AI agent lifecycle management. Operational capabilities are no longer limited to monitoring infrastructure availability. Instead, they increasingly encompass the entire AI agent lifecycle—from development, deployment, and runtime operation to evaluation, optimization, version upgrades, and retirement. The emphasis shifts toward continuous observation of agent execution paths, behavioral states, task outcomes, and business impact, combined with intelligent diagnostics that reduce manual troubleshooting effort while improving the stability, reliability, and controllability of AI agents.

The operational characteristics of AI agents—including multi-turn reasoning and dynamic tool invocation—also introduce new governance challenges. Resource consumption becomes highly dynamic, while infrastructure costs may fluctuate dramatically over short periods. Traditional coarse-grained resource accounting is therefore no longer sufficient to support fine-grained usage measurement and cost attribution across individual AI agents, use cases, and organizational units.

To address these challenges, enterprise operational governance is evolving toward fine-grained cost management, scenario-based operations, and security and compliance governance. Capabilities such as precise usage metering, budget enforcement, cost attribution, and behavioral auditing enable enterprises to control AI compute consumption, govern AI agent behavior, and maintain full compliance and traceability throughout AI operations.

Overall, automation, intelligent operations, and end-to-end operational control have become the defining requirements for operating AI agents reliably at enterprise scale.

6.2 Key Capabilities

To support the operational and governance requirements of enterprise AI agents, organizations should establish an intelligent operations framework built around five core capabilities: end-to-end observability, intelligent self-healing, lifecycle management, fine-grained usage metering and cost management, and security and compliance governance. Together, they form a closed-loop governance framework that progresses from problem detection to automated remediation and ultimately continuous optimization.

End-to-End Observability

Organizations should establish a unified monitoring framework covering compute resources, AI agent processes, model inference, workflow execution, tool invocation, and network communication, providing real-time visibility into the operational state of AI agents. The platform continuously monitors infrastructure and application metrics—including CPU, GPU/NPU, memory, storage, networking, token consumption, inference latency, invocation success rate, and task execution status—to rapidly identify resource bottlenecks, performance degradation, abnormal agent behavior, and service failures. Combined with visualization dashboards, distributed tracing, and real-time alerting, these capabilities enable operations teams to quickly understand the health of AI agent clusters and significantly improve incident detection and response.

Intelligent Self-Healing

The platform should provide automated detection, rapid diagnosis, intelligent remediation, and post-incident analysis for AI agent failures, model service disruptions, infrastructure faults, and network anomalies. Through anomaly detection, root-cause analysis, alert correlation, and policy-driven orchestration, the platform automatically identifies issues such as infinite execution loops, inference failures, improper tool usage, service timeouts, and compute resource unavailability. Automated remediation actions—including task retries, service failover, workload migration, instance restart, traffic throttling, graceful degradation, and policy rollback—allow many failures to be resolved without human intervention. This closed-loop automation reduces recovery time, lowers operational costs, and improves the availability and reliability of AI agent applications.

Lifecycle Management

Organizations should establish standardized lifecycle management processes covering AI agent deployment, release, operation, monitoring, iteration, upgrade, rollback, and retirement, enabling unified management across multiple AI agents, foundation models, software versions, and deployment environments. Standardized deployment pipelines, version management, configuration management, canary releases, change auditing, and runtime evaluation ensure end-to-end reliability, from development and testing to production. For large-scale multi-agent deployments, centralized orchestration, unified monitoring, bulk operations, and policy management reduce operational complexity while improving consistency and automation.

Fine-Grained Usage Metering and Cost Management

Organizations should establish a fine-grained operational management framework for AI workloads that collects, aggregates, and analyzes operational data related to compute consumption, model invocation, token usage, tool utilization, task execution, and user access. Usage metering, cost allocation, resource utilization analysis, workload assessment, and cost attribution help identify expensive workloads, inefficient resource consumption, and abnormal usage patterns. Combined with budget management, quota enforcement, cost alerts, and operational reporting, these capabilities make resource utilization measurable, operational costs controllable, and business outcomes quantifiable, ultimately maximizing the value of enterprise AI infrastructure.

Security and Compliance Governance

Organizations should establish comprehensive security and compliance governance covering compute resource access, AI agent operations, model invocation workflows, and enterprise data movement. Identity authentication, role-based authorization, access control, operational auditing, log retention, and data lineage provide end-to-end governance throughout AI agent execution. For private-cloud deployments, additional capabilities—including sensitive data protection, least-privilege access control, anomaly detection, and compliance auditing—ensure that AI agents comply with enterprise security policies and regulatory requirements when invoking foundation models, accessing knowledge bases, using external tools, or processing business data. These capabilities ensure that AI agents operate within defined security boundaries, maintain full auditability, and comply

with regulatory requirements, so that enterprises can deploy and run AI agents with confidence.

6.3 Implementation Approach

The implementation of an intelligent operations framework should pursue two overarching objectives: fine-grained operational optimization and integrated operational governance. Organizations should progressively advance through four stages—unified observability, operational automation, quantitative operations management, and security and compliance—to establish a closed-loop management framework spanning infrastructure, foundation models, and AI agents.

Phase 1: Build a Unified Observability Platform

Phase 1 establishes a unified monitoring platform that integrates compute resource monitoring, model service monitoring, business workflow monitoring, and AI agent runtime monitoring into a single operational view. The platform should continuously monitor key metrics—including CPU, GPU/NPU, memory, storage, networking, model inference latency, token consumption, task execution status, and tool invocation—to provide end-to-end observability from the underlying infrastructure to business applications. Based on this foundation, a hierarchical alerting mechanism should detect resource bottlenecks, service failures, abnormal agent behavior, and cost anomalies in real time, providing the visibility needed for rapid incident diagnosis and informed operational decision-making.

Phase 2: Establish an Automated Operations Framework

Phase 2 introduces end-to-end automation across AI agent deployment, runtime operation, upgrades, and incident response. Core capabilities include automated fault detection, root-cause analysis, self-healing, canary releases, policy rollback, and dynamic resource scheduling. By combining standardized operational procedures with automated remediation policies, organizations can reduce human intervention, shorten recovery times, and improve the stability and availability of large-scale AI agent deployments. At the same time, compute resources are dynamically allocated according to workload characteristics and infrastructure utilization, ensuring that business-critical workloads receive priority during periods of resource contention.

Phase 3: Establish a Fine-Grained Agent Operations Management Platform

Phase 3 establishes a fine-grained operations management platform for AI agent workloads. The platform should collect and consolidate operational data—including compute consumption, token usage, model invocation, tool utilization, resource utilization, and business outcomes—to provide a comprehensive operational view. Through cost accounting, cost allocation, resource utilization analysis, workload optimization, and scenario-based operational assessment, organizations can identify expensive workloads, inefficient resource consumption, and abnormal usage patterns. These capabilities make resource utilization measurable, costs attributable, budgets controllable, and infrastructure continuously optimizable, enabling data-driven operations and continuous improvement.

Phase 4: Strengthen Security and Compliance Governance

Phase 4 establishes a comprehensive security and compliance governance framework for enterprise AI and agent deployments. Organizations should implement unified identity and access management, authentication, audit logging, data lineage tracking, risk monitoring, and incident response mechanisms to ensure that AI agents operate securely throughout model invocation, tool usage, knowledge access, and business execution. At the same time, standardized operational procedures should be established for monitoring, alerting, change management, release management, rollback, auditing, and incident response. Together, these governance mechanisms create a standardized, fully traceable operational framework that enables AI agents to operate securely, reliably, and in compliance with enterprise policies over the long term.





Scan for more info



Scan for an
electronic copy

Trademarks and Permissions



HUAWEI,  are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Disclaimer

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolios, and new technologies. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, the information in this document is for reference only and does not constitute any offer or commitment. Huawei is not liable for any behavior that you make based on this document. Huawei may change the information at any time without notice.

Copyright © Huawei Technologies Co., Ltd. 2026. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.