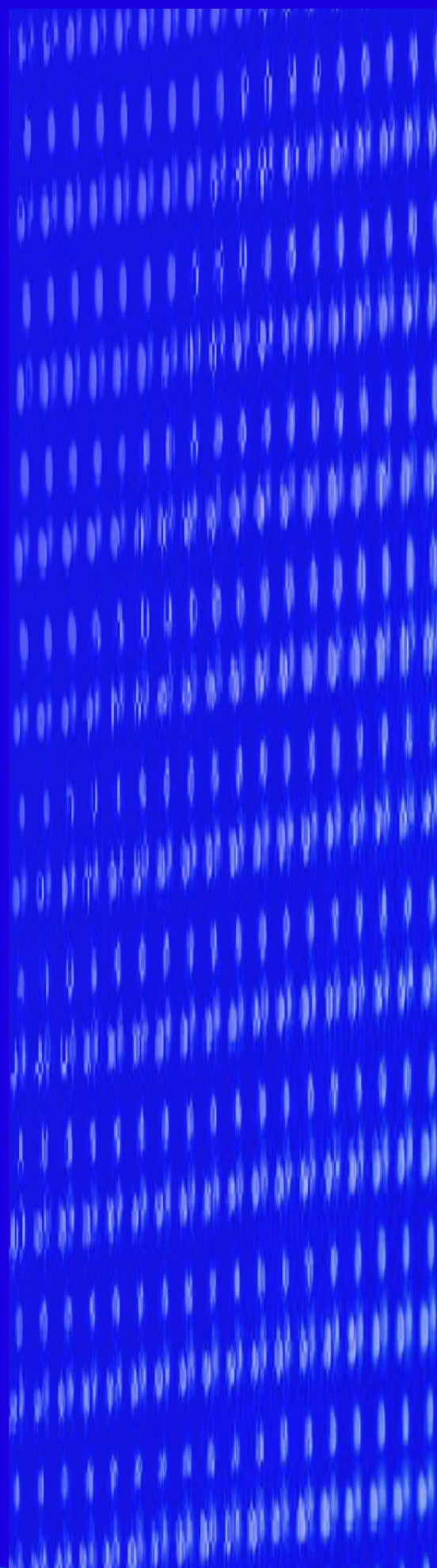


Select the Right AI Model

Use a practical decision path to shortlist, evaluate, route, and govern models for production agents and software with any model, any provider.



Contents

03

[Unlock the potential of today's transformative AI models](#)

05

[Start with your use case and explore](#)

09

[Map tasks to models](#)

12

[Benchmark, prototype, and optimize models](#)

21

[Design with the best models using Foundry](#)

23

[Next steps](#)



Unlock the potential of today's transformative AI models

Innovators everywhere are using agentic AI to prototype and design AI applications and agents faster than ever. They're transforming customer engagement, employee experiences, and business processes. They're building and combining AI agents that not only plan and execute steps in a workflow but also adapt to changes while keeping humans in the loop.

These breakthroughs are powered not just by a single class of AI models trained on massive datasets, but by an evolving stack. Teams are using fine-tuned models, domain specific models, multimodal models, open-source alternatives, and orchestration tools that adapt models to real-world workloads. Familiar flagship models like Claude, DeepSeek, and GPT-5 are part of a rapidly expanding ecosystem. Thousands of models exist now, each optimized for different tradeoffs in quality, latency, and cost. As this ecosystem expands across established providers and startups, choosing the right model for each use case is crucial.

A production model decision spans connected product roles. GitHub and Visual Studio Code carry code, tests, automation, collaboration, and delivery. Microsoft Foundry carries model discovery, workload evaluation, deployment, routing, and production monitoring. Copilot Studio carries agent creation, orchestration, governance, and business process automation across enterprise systems and knowledge. The products connect in the solution, but they do not share one model catalog, evaluation plane, or governance surface.

ABOUT THIS E-BOOK

WHO THIS IS FOR: AI architects and lead AI engineers evaluating models for a defined production workload across agents and software.

Trends and opportunities

Providers continue to rapidly upgrade models and release new ones, pushing the frontier and creating new opportunities. Today, Microsoft Foundry offers models across providers and deployment options, including those hosted and sold directly by Microsoft in addition to a broad choice from leading creators. Start across any model, any provider, then narrow the field using required capabilities, non-negotiable trust constraints, and dated evaluation thresholds. A catalog snapshot cannot carry a production decision. Record what was tested, under which conditions, and what change will trigger another evaluation

Organizations are moving from relying on a single general-purpose model to using portfolios of open, reasoning, multimodal, industry-specific, and lightweight models. This gives teams greater control over quality, cost, latency, customization, and deployment. Intelligent routing makes this diversity easier to manage by dynamically selecting models based on task complexity, performance requirements, cost thresholds, and workload conditions.

In addition, a shift toward right-sized AI models is also helping organizations balance capability and efficiency. For example, task-specific and industry-specific models are optimized for on-device execution in mobile, PC, and local environments. These are complemented by ultra-lightweight nano models designed for highly constrained hardware, such as microcontrollers, enabling low-latency, privacy-preserving AI in embedded and Internet of Things scenarios.

As AI models continue to advance, so do the ways you can apply them. Early AI efforts were often centered on deploying a single, general-purpose model capable of handling text, audio, and visual tasks. Today, teams are designing systems composed of multiple specialized and multimodal models that work together to deliver richer analysis and more immersive experiences. These systems increasingly combine agents, tools, and models that can adapt over time, enabling workflows that are autonomous, context-aware, and optimized continuously rather than fixed at deployment.

The answer is not a longer catalog. Use ordered gates to define the workload, constraints, shortlist, evaluation, operating pattern, and re-evaluation trigger.

The image displays a collection of AI model logos arranged in a grid. Two model cards are highlighted, providing detailed information about specific models.

Model Card 1: gpt-4.1 (Azure OpenAI)

Quick facts		Azure OpenAI
gpt-4.1		vision-preview
Fine-tuning supported		Chat completion
Last trained April 2025		
Pay-as-you-go (per 1K tokens)		
\$0.005	\$0.015	
Model input	Model output	
Benchmarks		
0.974	0.376	
AI quality	Cost	

Model Card 2: Phi - i-4k-inst... (Microsoft)

Quick facts		Microsoft
Phi - i-4k-inst...		
Fine-tuning supported		Chat completion
Last trained October 2023		

/01

Start with your use
case and explore



Evaluate, test, and monitor models in one place to make an informed selection

Define the workload and non-negotiable constraints before opening a model catalogue. Record the user and task, inputs and required outputs, measurable outcome, required capabilities, quality threshold, latency, efficiency, safety, privacy, compliance, residency, deployment, traffic, and operating environment. Any candidate that misses a non-negotiable requirement leaves the shortlist. Then decide whether the workload needs one fixed model, a bounded selection process, or runtime routing.

- * [High-stakes applications](#) such as healthcare diagnostics and fraud detection require models with accuracy, interpretability, and data availability.
- * [Real-time conversations](#) require fast AI models with acceptable precision that run affordably and integrate smoothly.
- * [Enterprise predictive analytics](#) require accurate, scalable AI models with cost-effective performance.



up to
35%

Improvement in developer
productivity using Foundry

Choose the operating pattern the workload supports

Fixed model

Use when one validated candidate meets the workload thresholds across the expected request set. Record the model, version, deployment, evaluation evidence, fallback behavior, and re-evaluation trigger.

Controlled model selection

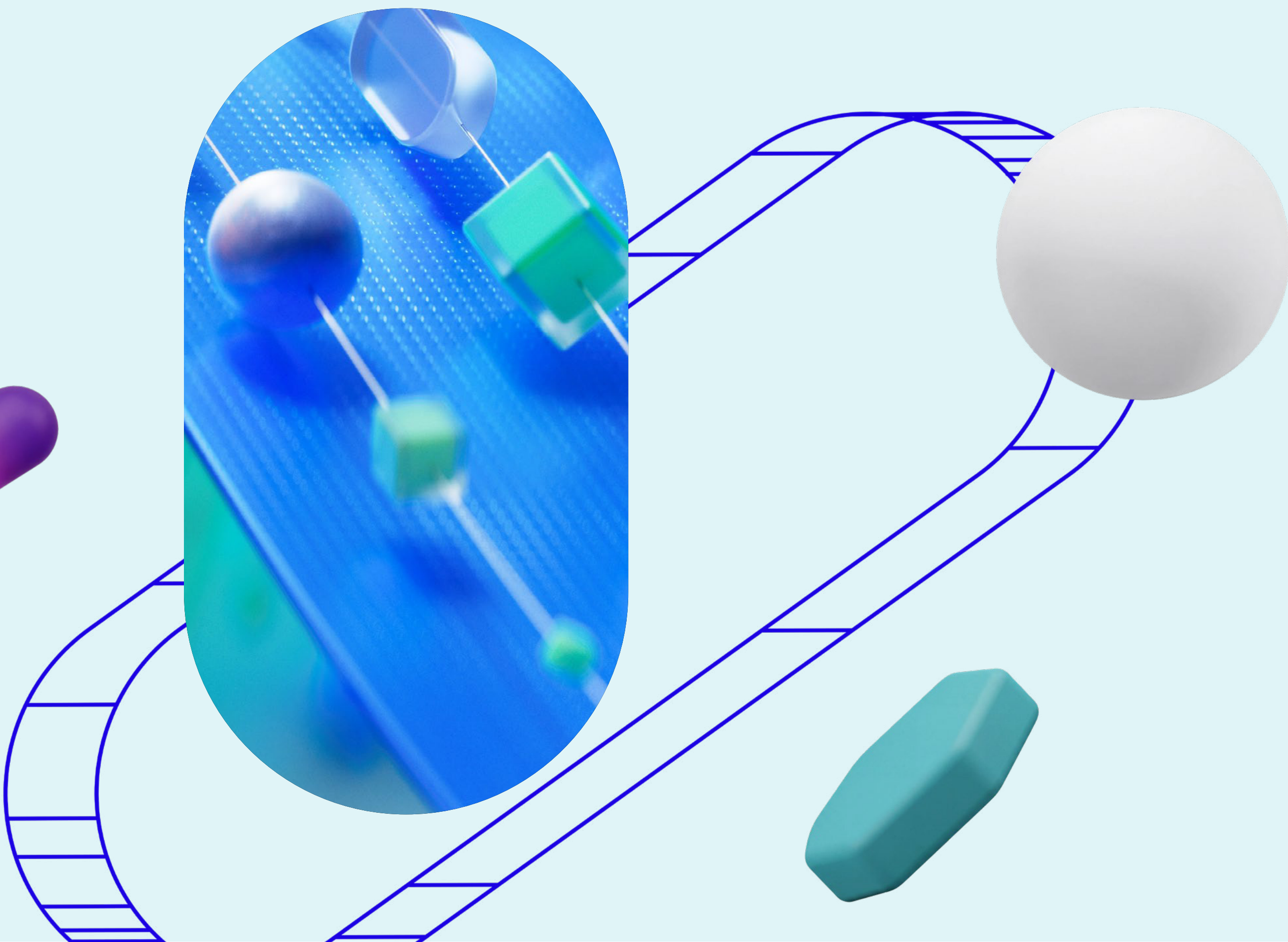
Use when development or evaluation teams need bounded choice before a production pattern is set. Define the approved candidate set, shared test harness, thresholds, logging, review owner, and promotion rule.

Dynamic routing

Use when request complexity or constraints vary enough to justify runtime selection. Define the allowed model pool, routing policy, fallback and failure behavior, observability, and re-evaluation trigger.

The pattern follows the evidence. Account for the policy, fallback, evaluation, and observability work before choosing a routed pattern.

Decision record output: defined workload, non-negotiable constraints, and selected operating pattern.





Transforming property marketing

Business need

Automate photo cleanup and create professional marketing materials efficiently.

Models

Azure OpenAI Models - AI FlyerGEN system - Flux. 1 Kontext [pro]

Products

Azure Blob Storage, Azure Data Factory, Azure Data Lake Storage, Azure Databricks, Azure OpenAI in Foundry Models, Microsoft Fabric, Microsoft Foundry, Power BI



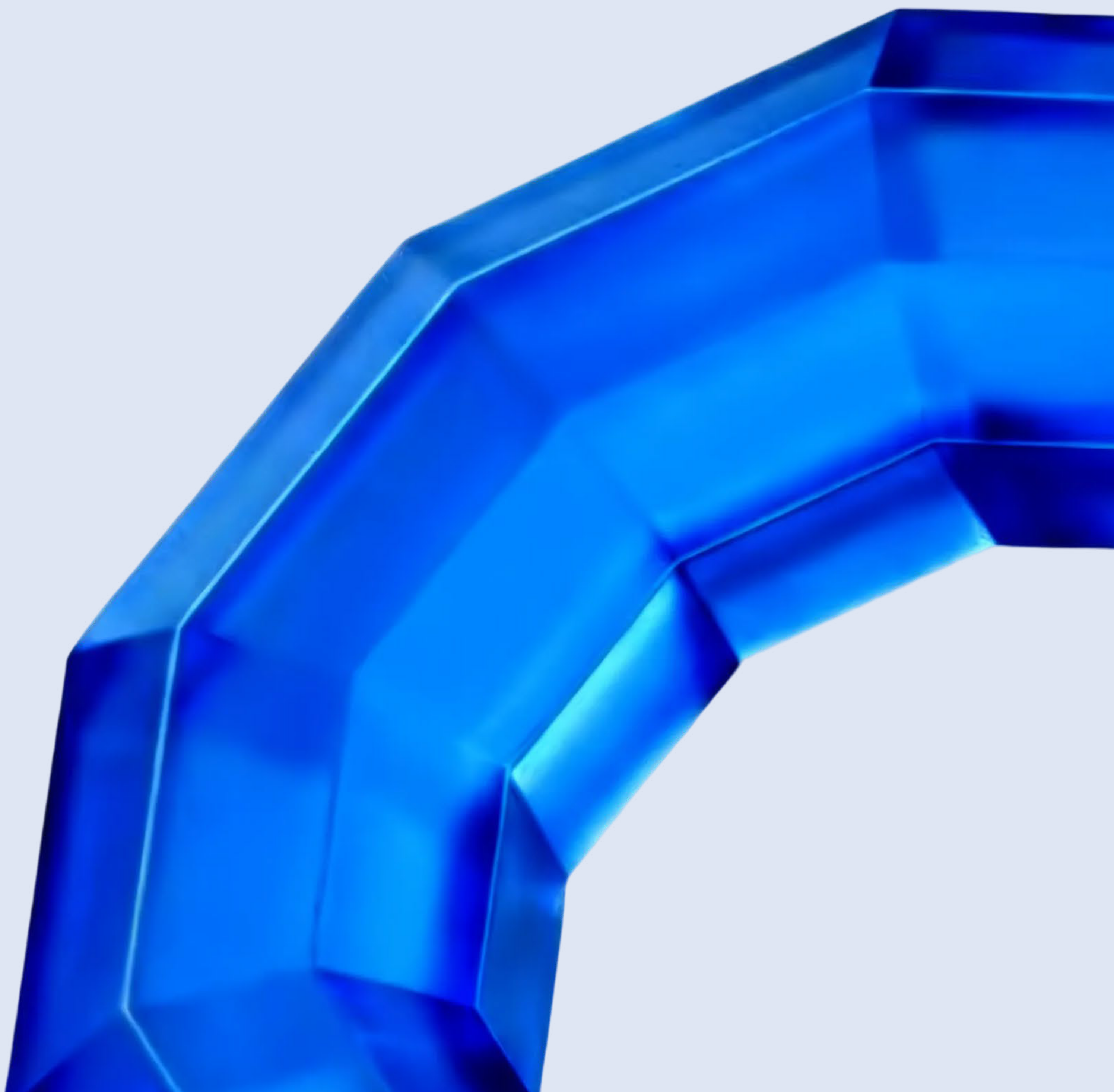
”

Creating customized flyers used to be time-consuming, but with this workflow, the entire process can be nearly fully automated. All agents need to do is upload data and click a few buttons. This allows them to focus on what truly matters: building relationships and closing deals.

ChuTing Hsu,
Project Manager,
Sinyi Realty

/02

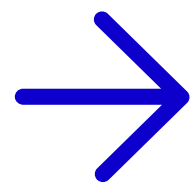
Map tasks to models



Can AI solve your use case?

Find any model you need - from OpenAI to Anthropic

ai.azure.com/explore/models



To find out, you must frame the question in terms of the capabilities you need. Try asking the following:

- * As input, does the model need to handle natural language processing (NLP), audio, computer vision, multimodal input, or something else? What output will it generate—text summaries, decisions, transcriptions, predictions?
- * How complex does the reasoning or interaction need to be? Do you need fine-tuning to match the required precision and domain knowledge, or do you need high performance on general knowledge across diverse fields?
- * What are your inference speed requirements?

Decision record output: required capabilities, non-negotiable eligibility constraints, and the shortlist that will enter workload evaluation.





Deeper customer connections through agentic workflows

Business need

Deliver hyper-personalized experiences and real-time insights to millions of subscribers.

Models

Azure OpenAI in Foundry models using GPT-4.1 and GPT-4.1 mini - Azure Speech-to-Text / Text-to-Speech models - Azure AI Content Safety models.

Products

Azure, Microsoft Teams, Foundry Models, Content Safety in Foundry Control Plane, Azure Speech in Foundry Tools Microsoft Foundry, Azure OpenAI in Foundry Models

TOP 10
IN SOUTH KOREA APP STORE FOR
THREE MONTHS

10 MILLION
SUBSCRIBERS AND MONTHLY
ACTIVE USERS

[READ THE FULL STORY](#)

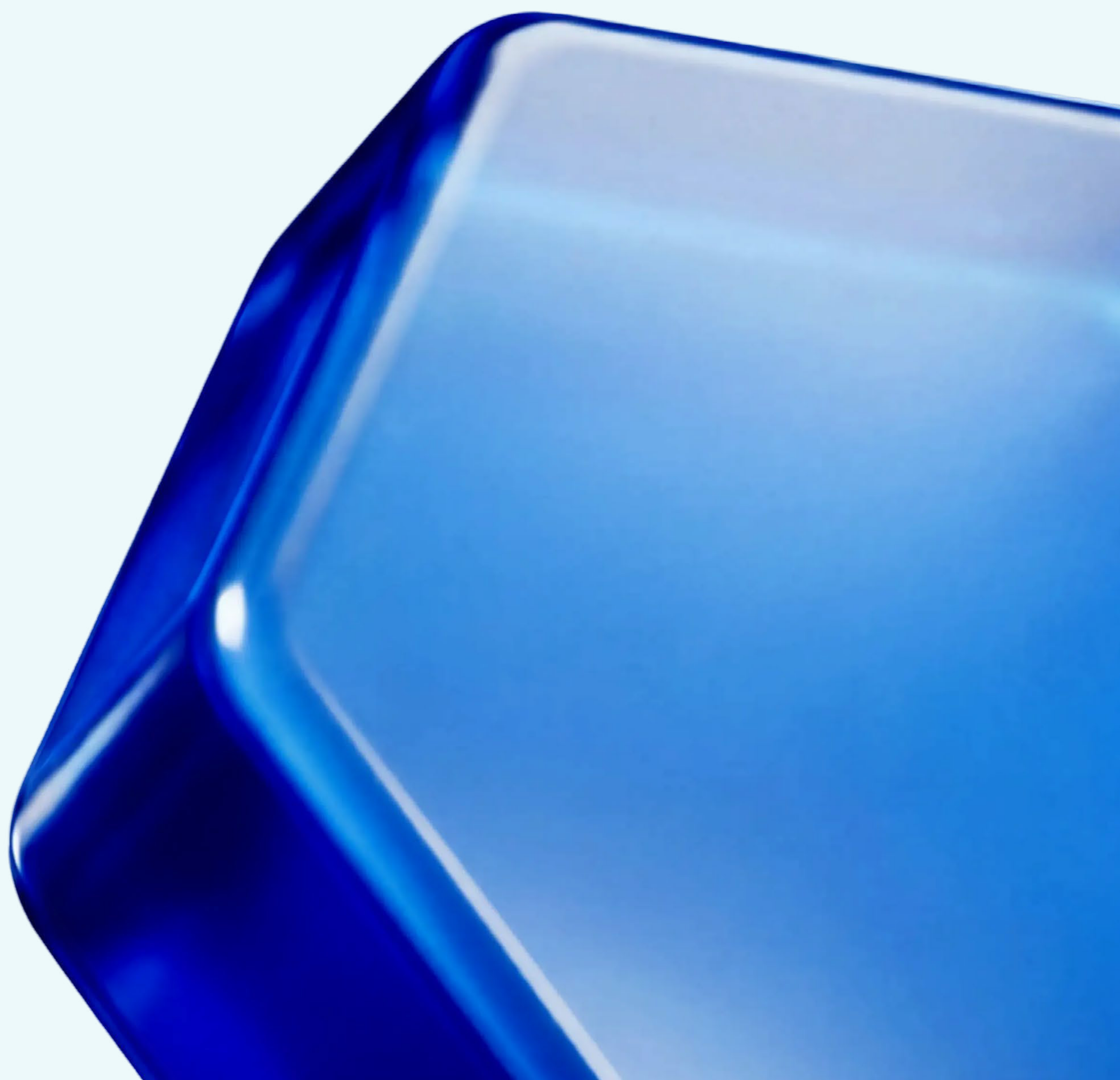
”

We're focused on developing very natural communication between humans and AI that agentic AI enables, and we plan to continue to use Microsoft Foundry to help us develop it.

*MinSeok Ko, Lead Planner of
Vertical Agents,
SK Telecom*

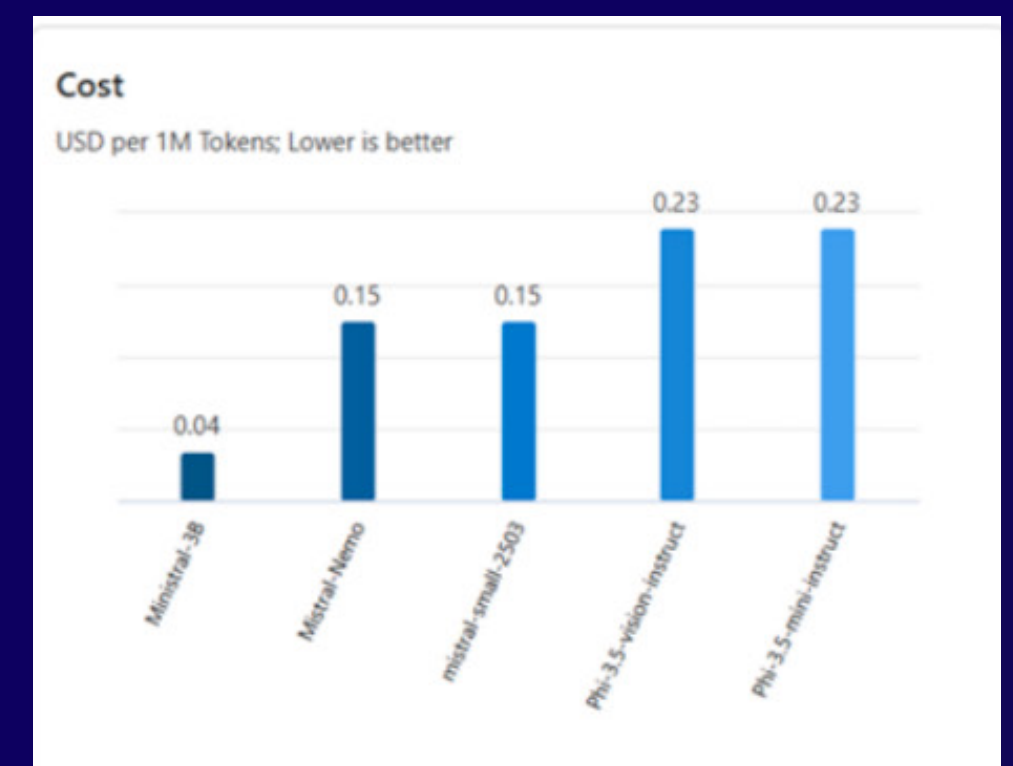
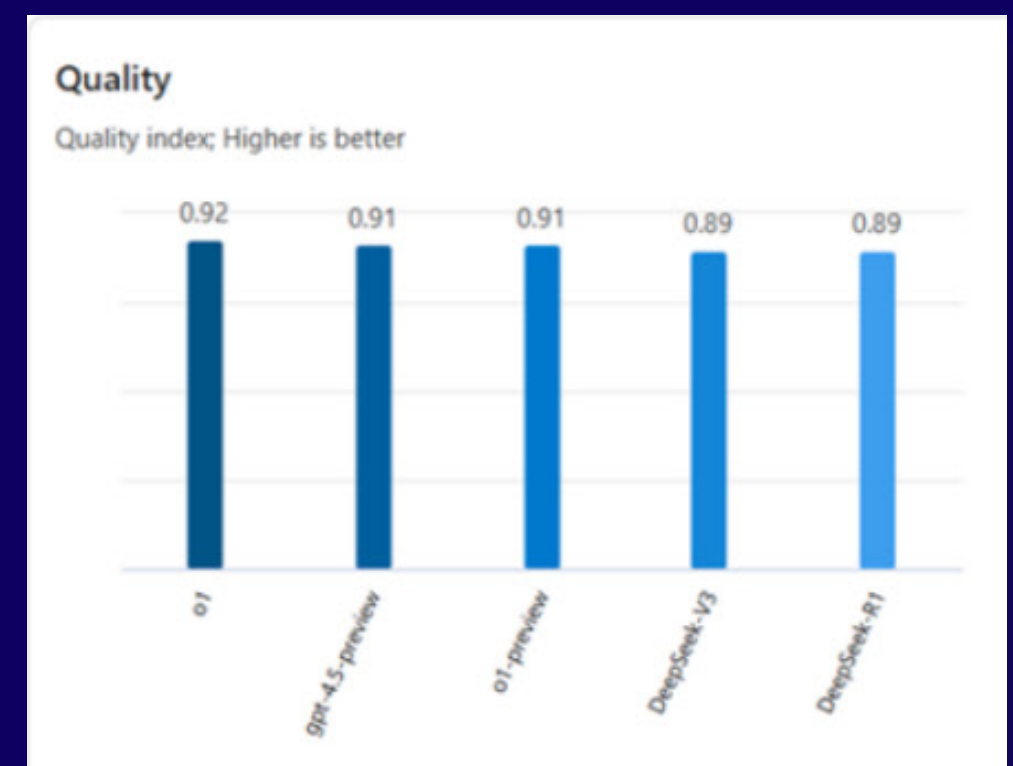
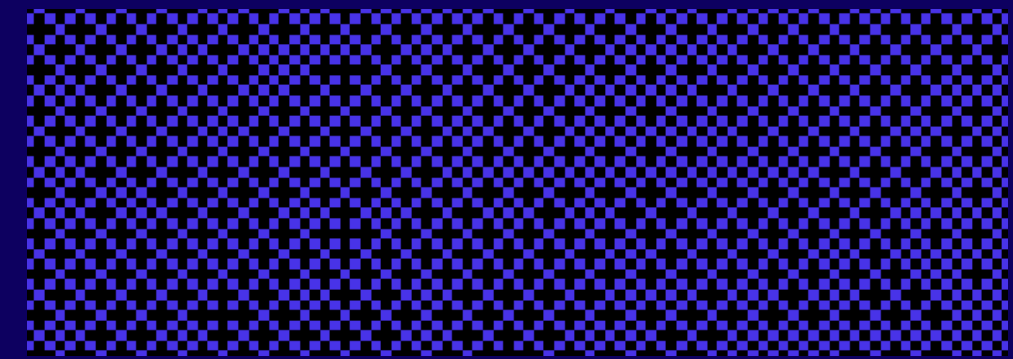
/03

Benchmark, prototype,
and optimize models



These days, it's easy to use benchmarks to compare AI models and identify potential candidates. Standardized industry benchmarks help you assess a model's general capabilities like intelligence, reasoning, or code generation. Application-specific benchmarks can tell you how well a model performs in a real-world context, such as healthcare, finance, ecommerce, or security.

However, raw benchmarks aren't everything. Two models may perform well against your benchmarks but behave differently in your app. The right model balances your priorities and constraints with your budget.



Leaderboards based on public datasets help you narrow down the choices. Quality is the most common criterion for model selection, followed by safety, cost, and performance. You can also use your own data to evaluate models in Foundry.

Performance and accuracy

Measures of performance help you make the most of a model's unique architecture and training methods. To align a model's strengths with the needs of your use case, you need both quantitative measures of performance and qualitative metrics of context handling and safety.

Evaluations are essential for understanding quality, safety, and reliability. Foundry supports model, retrieval, and agent evaluations for criteria such as relevance, groundedness, coherence, safety, and task completion. Trace-linked evaluation connects quality results to individual steps in an agent workflow, helping teams identify where a problem occurred and how to correct it.

BEST PRACTICES

- ✓ Define your acceptable thresholds for performance metrics, such as latency, throughput, accuracy, precision, and recall.
- ✓ Compare the top contenders based on your key metrics for quality, cost, safety, and throughput.
- ✓ Try browsing model leaderboards in Foundry to quickly surface leaders and run side-by-side comparisons.
- ✓ To assess the correctness of a model's output, use exact match (EM) scores, F1 scores, and recall and precision metrics.
- ✓ Measure the reliability of answers using risk and safety evaluators that detect bias, hallucinations, and fairness.
- ✓ To compare the semantic quality of models, use textual similarity evaluators—vital for text generation, summarization, and translation tasks.
- ✓ Perform load testing to make sure your AI model can handle the volume and variety of production-level data. You can use Foundry to simulate real-world workloads, stress-test your model, and make sure performance and accuracy are maintained.
- ✓ Use A/B testing to compare different model versions using your scenarios and data. Continuously test to see which models provide the right balance of impact, risk, and cost.
- ✓ Use model routing only when the workload benefits from runtime selection, with an approved model pool, explicit thresholds, fallback behavior, and monitoring.

Resource management

AI models use virtual and physical resources whether they run on an edge or local device, on-premises servers, or in the cloud. The constraints of your deployment environment determine the type and size of model you can run.

Cloud-based deployments can use large foundational models, while smaller nano models suit on-device deployments and ensure that no data leaves the device.



BEST PRACTICES

- ✓ Keep costs and performance in balance by verifying a model's compute requirements, memory footprint, and loading times.
- ✓ Make sure your infrastructure can support increases in demand without racking up costs or wasting resources. Foundry can help you monitor and optimize the computational resources required for training and deploying your models.
- ✓ Factor in any plug-ins or multiple-agent setups needed for models that reason, plan, or use tools.
- ✓ Use GitHub and Visual Studio Code for code, tests, automation, collaboration, and delivery. Use Microsoft Foundry for model discovery, workload evaluation, deployment routing, and production monitoring.
- ✓ For edge or offline scenarios, start with Foundry Local models like the Microsoft Phi family for on-device inference, then scale to Foundry for hybrid enterprise deployments with Azure Arc when you need centralized management and cloud integration.
- ✓ Use model subsets to limit routing to smaller, cost efficient models, then apply routing modes so model router dynamically selects the right model per request based on cost, performance, and complexity.

Costs and benefits

Budget always plays a role in assessing whether an AI model is the right fit for your needs. Freely available open-source models are popular, but a model's resource use and compute instances dramatically affect the overall runtime and maintenance costs—in addition to any other services you deploy. Plus, training a model using your own data usually means a bigger budget and infrastructure.

Model selection is also a deployment strategy that shapes how your AI systems scale, adapt, and operate. Good results in testing don't guarantee the model will work well in production, when traffic patterns, budget constraints, and governance requirements come into play. Using Foundry, you can make deployment a first-class consideration in your model strategy and ensure that performance, cost, and scale evolve together.

Here's a pragmatic way to think about cost tradeoffs:

\$ Start with open-source or smaller models to experiment and prototype quickly with minimal upfront cost. As your project evolves, you can switch to another model.

\$\$ Use pretrained foundation models from available providers for production workloads that need reliable performance, managed infrastructure, and predictable costs at scale.

\$\$\$ Use the most powerful models for high-stakes or complex tasks, choosing managed APIs or dedicated hardware based on performance and control requirements.

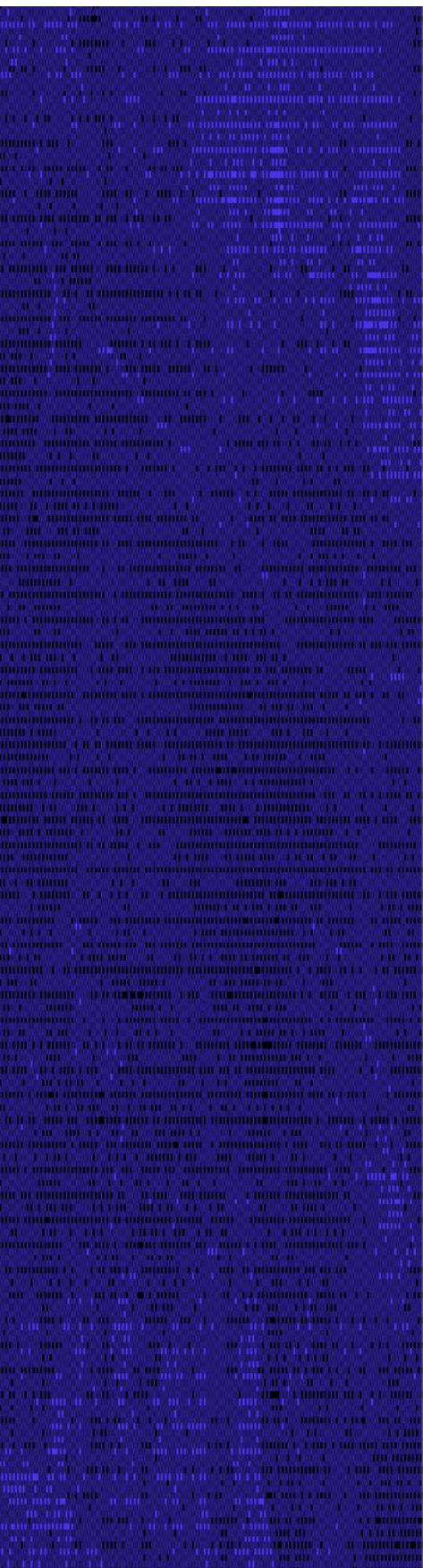
BEST PRACTICES

- ✓ Compare models on usage-based, per-token pricing but look further. Providers offer different pricing structures based on use case, deployment mode, and model type. For example, Foundry supports token-based Pay-As-You-Go pricing, provisioned throughput units (PTUs) for capacity reservation, global or data zone configurations for sovereignty needs, third-party model pricing, region-specific pricing, and potential discounts for certain usage patterns.
- ✓ In addition to inference costs per request, assess the cost per inference at scale should the model need to handle more users or data over time.
- ✓ Include the costs of customizing or fine-tuning a model. Work with providers that offer a low-cost window for experimentation and fine-tuning, like the Developer Tier in Foundry.
- ✓ To optimize overall costs, take advantage of serverless or autoscaling infrastructure in your applications and agents.
- ✓ Compare hosting options. For example, Foundry features Standard, Global Standard, and Regional provisioned throughput, each with its own pricing structure that may include hourly rates or token-based charges.
- ✓ For predictable workloads, compare current consumption and provisioned options using the region, deployment type, expected utilization, and current pricing basis. Record the source and review data before making a capacity decision.
- ✓ Monitor costs to help identify spending trends and potential overspending. For example, use the cost analysis tools in Microsoft Cost Management, available to anyone with access to a billing account, subscription, resource group, or management group.
- ✓ Use cost mode in model routing to prioritize cost-efficient models automatically. The router enforces cost and quality thresholds at runtime, reducing token spend and latency for high-volume workloads and eliminating per-prompt cost decisions.

Optimization branch: Close the measured gap

When a candidate misses a threshold, identify the failure before choosing a remedy. Test prompt changes or grounding when the gap concerns instruction or context. Consider fine-tuning, distillation, routing, or an architecture change only when the workload evidence supports that branch. Re-run the same evaluation after each change.

Many models support lightweight customization through RAG, low-rank adaptation, adapters, and prompt tuning. Some models include an API, command-line interface, or user interface for further fine-tuning. You can use Foundry to determine the level of customization a model supports and safely test models in evaluation playgrounds.



BEST PRACTICES

- ✔ To enhance precision in a specialized domain, choose a model that can be fine-tuned to match the specific knowledge needed.
- ✔ To improve output using RAG, look for models that you can connect to external data sources or that support knowledge retrieval.
- ✔ To improve performance, choose a model that can be quantized, pruned, or distilled without a noticeable drop in accuracy.
- ✔ Check how much expertise is required for tuning. Some models include simple interfaces that don't require a data scientist to use.
- ✔ Test whether your customization and fine-tuning efforts are, in fact, better. For example, use the Azure AI Evaluation SDK to evaluate model outputs using model-based graders, custom rubrics, and structured scoring—all from code.
- ✔ For development ease, use models that can be deployed as API endpoints.
- ✔ Make sure the model is compatible with other tools and systems in your stack.

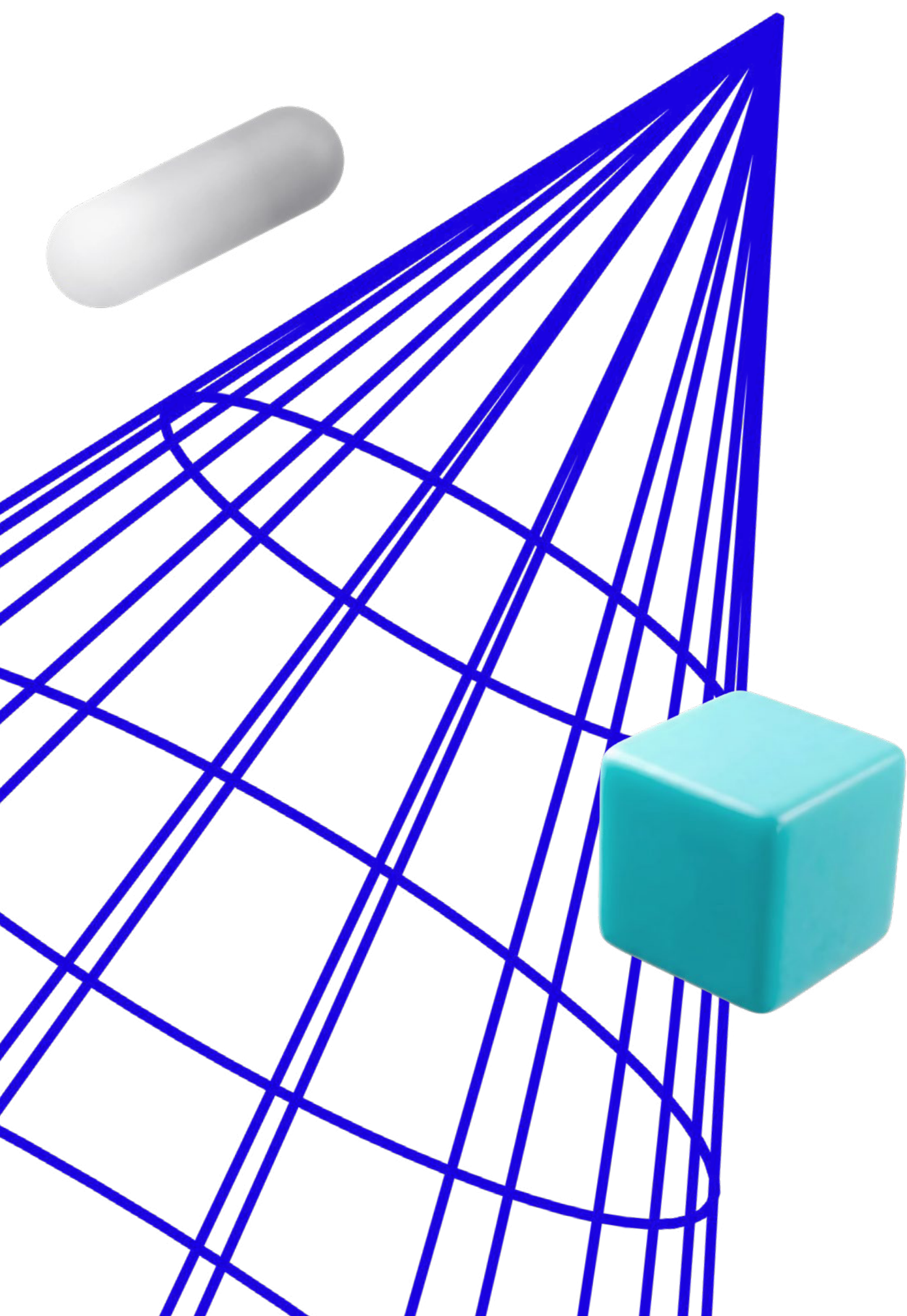
Safety and compliance

Before shipping an AI solution, organizations need to manage security, safety, privacy, and compliance across the complete application or agent—not only the selected model. Trust should be built into the platform rather than added separately to each workload. Add production monitoring for quality, latency, efficiency, safety, fallback use, version drift, and failed evaluations, with an owner and threshold that reopens the decision.

Foundry offers built-in security features and compliance certifications that help you follow regulatory and data protection standards, such as the General Data Protection Regulation and the Health Insurance Portability and Accountability Act.

BEST PRACTICES

- ✓ Choose a model that handles data appropriately given your regional and regulatory requirements.
- ✓ To preserve data privacy and sovereignty, choose open or licensed models that support local inferencing.
- ✓ Use Foundry and responsible AI tools to assess and configure how AI models handle sensitive or personal data, ensuring compliance with privacy and ethical standards.
- ✓ Extend secure deployment using a private infrastructure through compatible services like Azure Arc or Azure Kubernetes Service (AKS).
- ✓ Implement security, observability, and governance across the AI lifecycle—from development and testing through deployment and production. Foundry inventories agents and their owners, identities, permissions, tools, and data connections; traces model and tool activity; and monitors quality, safety, reliability, latency, token consumption, and cost.
- ✓ Teams can use these signals to find agents without accountable owners, flag excessive access or risky tool and data combinations, detect anomalous behavior and potential data leakage, enforce guardrails and scoped policies, and trigger alerts or remediation workflows when risk thresholds are exceeded.



Consistency and innovation

A model decision must remain inspectable as versions, providers, and workload conditions change. Record the update policy, fallback, rollback, monitoring owner, and event that triggers re-evaluation rather than treating provider change as automatic improvement.

However, you may need a model with long-term consistency to meet regulatory requirements or to support product stability. For example, customer support scripts, compliance tasks, and clinical decision-making rely on predictable behavior, so you don't want surprise changes in how a model works.



BEST PRACTICES

- ✓ Choose models that reflect responsible agentic AI practices. Foundry provides tools and resources to help you apply the Microsoft Responsible AI standard for security, safety, privacy, fairness, transparency, and accountability.³
- ✓ Determine how often a model is updated or retrained and the provider's version control and rollback policies.
- ✓ Consider vendor stability. Is the model provider likely to be around three to five years from now?
- ✓ For long-term consistency, use a model specifically designed for stability and reliability. Models sold directly by Azure typically offer enhanced integration, optimized performance, and direct support from Microsoft—factors that contribute to more predictable behavior.
- ✓ To handle change, connect the model decision to version control, delivery, monitoring, fallback, and rollback. Reopen the evaluation when the workload, model, provider, deployment, policy, or operating evidence changes.
- ✓ Decision record output: dated workload evaluation, selected model or routing policy, optimization branch, fallback, accountable owner, and re-evaluation trigger.

Bringing agronomic expertise to the edge

Business need

Fine-tuning on proprietary label data to ensure precise, compliant answers for advisors in the field

Models

Microsoft Phi model family

Products

Foundry, Foundry Models

5 TO 10%
PRODUCTIVITY
GAINS

30 SECONDS TO
ANSWER, NOT
DAYS

[READ THE FULL STORY](#)

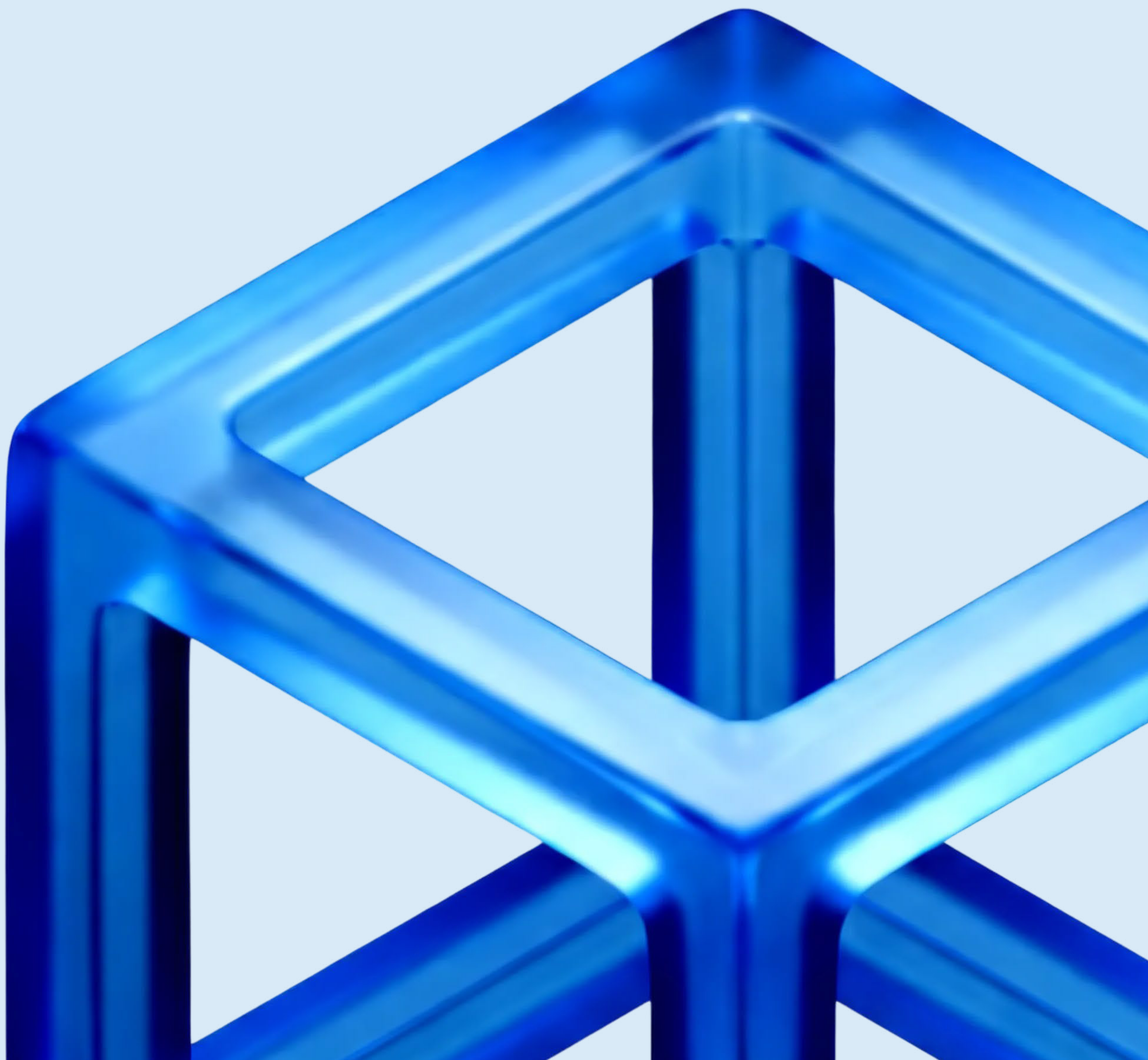
”

A wrong answer isn't just a missed opportunity—it could cause real harm to a farming operation. This model removes the guesswork. It also surfaces through the existing E.L.Y. Copilot tool, so advisors don't have to learn something new.”

*Balathasan “Giri” Giritharan,
Principal Data Science Architect,
Bayer*

/04

Design with the best
models using Foundry



Your starting point

Whether you're building AI agents, automating industry workflows, or launching retrieval-augmented systems, Foundry provides end-to-end support across the development lifecycle. Built for modern enterprise needs, Microsoft Foundry helps you evaluate candidates against workload, quality, latency, efficiency, safety, and deployment constraints, then carry the selected model or routing policy into production.

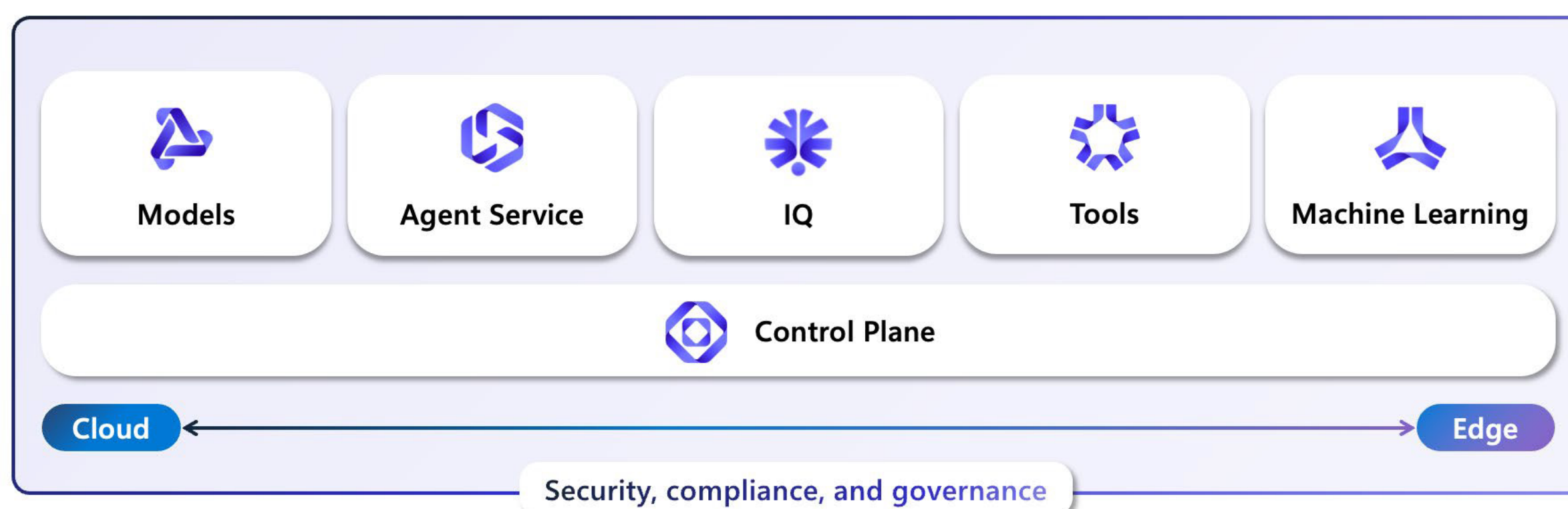
Foundry does not stand alone. GitHub carries code, test, security, automation, collaboration, and delivery. Microsoft Foundry carries model discovery, workload evaluation, deployment, routing, and production monitoring. Copilot Studio carries agent creation, orchestration, governance, and business process automation across enterprise systems and knowledge.

Carry the same workload thresholds and decision record across these handoffs without implying a shared catalog or control plane.

Experiment in a controlled environment

As AI workloads grow more complex, you need self-service development tools that streamline the end-to-end lifecycle of AI agents and applications.

- * [Work in on-demand, zero-setup Foundry playgrounds.](#)
Rapidly prototype, explore APIs, and validate feasibility.
- * [Build and deliver the software in GitHub or Visual Studio.](#)
Design, deploy, and govern agents in Microsoft Copilot Studio when the scenario requires automation of a business process or system, involving organization knowledge. Use Microsoft Foundry for model APIs, evaluation, deployment, routing, and production monitoring.
- * [Deploy and call models and the model router directly.](#) Both approaches use the Inference API, making it easy to introduce, test, and validate new model endpoints with minimal changes to existing integrations.
- * [Ship faster on standardized tools.](#) Use self-service Foundry resources and APIs to deploy apps and agents at scale with central governance.



Available through a unified portal, SDK, and API, Foundry helps you go from prototype to production faster and develop responsibly with safety, security, and privacy. Integration with popular development tools and the Microsoft security ecosystem helps you operationalize and monitor your workflows.

Next steps

Run a workload-specific model evaluation

Define success thresholds and non-negotiable constraints. Compare a shortlist across any model, any provider using representative data and production assumptions. Record the selected model or routing policy, fallback behavior, monitoring owner, and re-evaluation trigger.

Primary action

- Start a model evaluation in [Microsoft Foundry](#)
- [Supporting resources](#)
- [Microsoft Foundry model evaluation guidance](#)
- GitHub and Visual Studio Code delivery guidance
- Microsoft Copilot Studio workflow-led agent guidance

[Open Microsoft Foundry and start a model evaluation](#)

Supporting resources

- [Select, deploy, and evaluate Microsoft Foundry models](#)
- [GitHub Copilot delivery guidance for GitHub and IDEs](#)
- [Microsoft Copilot Studio solution and orchestration guidance](#)

Sources

¹ [The economics of enterprise AI: What the Forrester TEI study reveals about Microsoft Foundry](#), March 6, 2026. ² Azure Reservations ³ Responsible at Microsoft

©2026 Microsoft Corporation. All rights reserved. This document is provided "as-is." Information and views expressed in this document, including URL and other Internet website references, may change without notice. You bear the risk of using it. This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.