AI adoption

Article • 03/24/2025

The Cloud Adoption Framework (CAF) provides a structured process for adopting AI solutions in Azure. This framework outlines clear steps, many of which apply to Microsoft Copilot adoption.

The CAF AI adoption process supports organizations ranging from large enterprises to startups. You learn how to identify AI use cases, select appropriate AI solutions, and build effective AI workloads. The guidance also covers operational processes required for governance, management, and security of AI implementations.

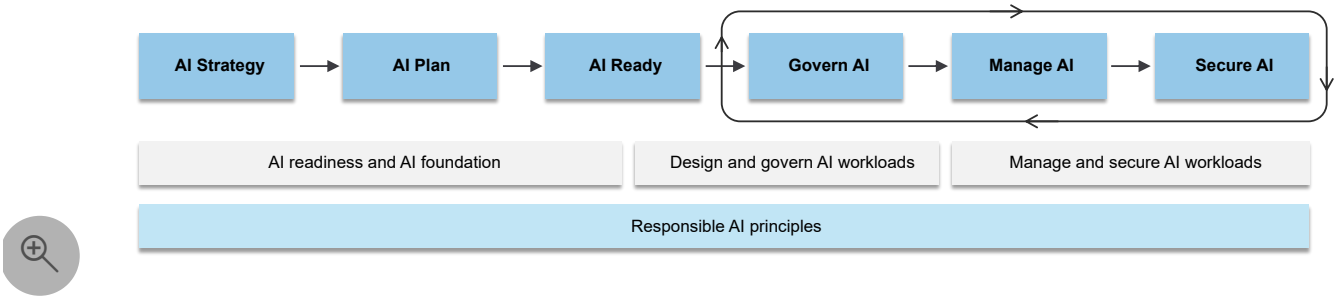


Figure 1. How to use the AI adoption guidance.

AI checklists

Use these AI checklists as a practical roadmap for adopting and managing AI. The enterprise checklist equips your organization to scale AI effectively. The startup checklist helps you quickly move toward production while adopting governance, management, and security best practices.

Expand table

AI adoption step	Applicable AI technology	Startup checklist	Enterprise checklist
AI Strategy	Copilots Azure	☐ Define an AI technology strategy	☐ Identify AI use cases ☐ Define an AI technology strategy ☐ Define an AI data strategy ☐ Define a responsible AI strategy
AI Plan	Copilots Azure	☐ Access AI resources ☐ Implement responsible AI	☐ Assess AI skills ☐ Acquire AI skills ☐ Access AI resources

AI adoption step	Applicable AI technology	Startup checklist	Enterprise checklist
			☐ Prioritize AI use cases ☐ Create an AI proof of concept ☐ Implement responsible AI ☐ Estimate delivery timelines
AI Ready	Azure	☐ Build an AI environment ☐ Choose an architecture ☐ Use AI design areas	☐ Establish AI reliability ☐ Establish AI governance ☐ Establish AI networking ☐ Establish an AI foundation ☐ Choose an architecture ☐ Use AI design areas
Govern AI	Copilots Azure	☐ Enforce AI governance policies	☐ Assess AI organizational risks ☐ Document AI governance policies ☐ Enforce AI policies ☐ Monitor AI organizational risks
Manage AI	Copilots Azure	☐ Manage AI models ☐ Manage AI costs	☐ Manage AI operations ☐ Manage AI deployment ☐ Manage AI endpoint sharing ☐ Manage AI models ☐ Manage AI costs ☐ Manage AI data ☐ Manage AI business continuity
Secure AI	Copilots Azure	☐ Implement AI security controls	☐ Assess AI security risks ☐ Implement AI security controls ☐ Maintain AI security controls

Next step

AI Strategy

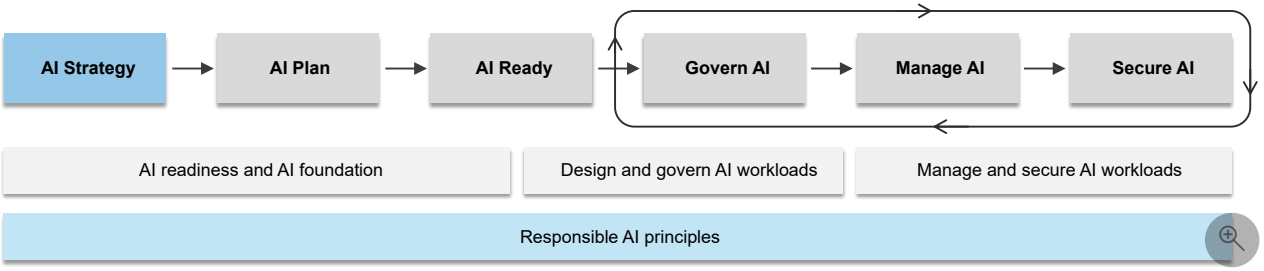
Feedback

Was this page helpful? Yes No

AI Strategy - Process to develop an AI strategy

Article • 03/07/2025

This article outlines the process to prepare your organization for AI adoption. You learn how to select the right AI solutions, prepare your data, and ground your approach in responsible AI principles. A well-planned AI strategy aligns with your business objectives and ensures that AI projects contribute to overall success.



Identify AI use cases

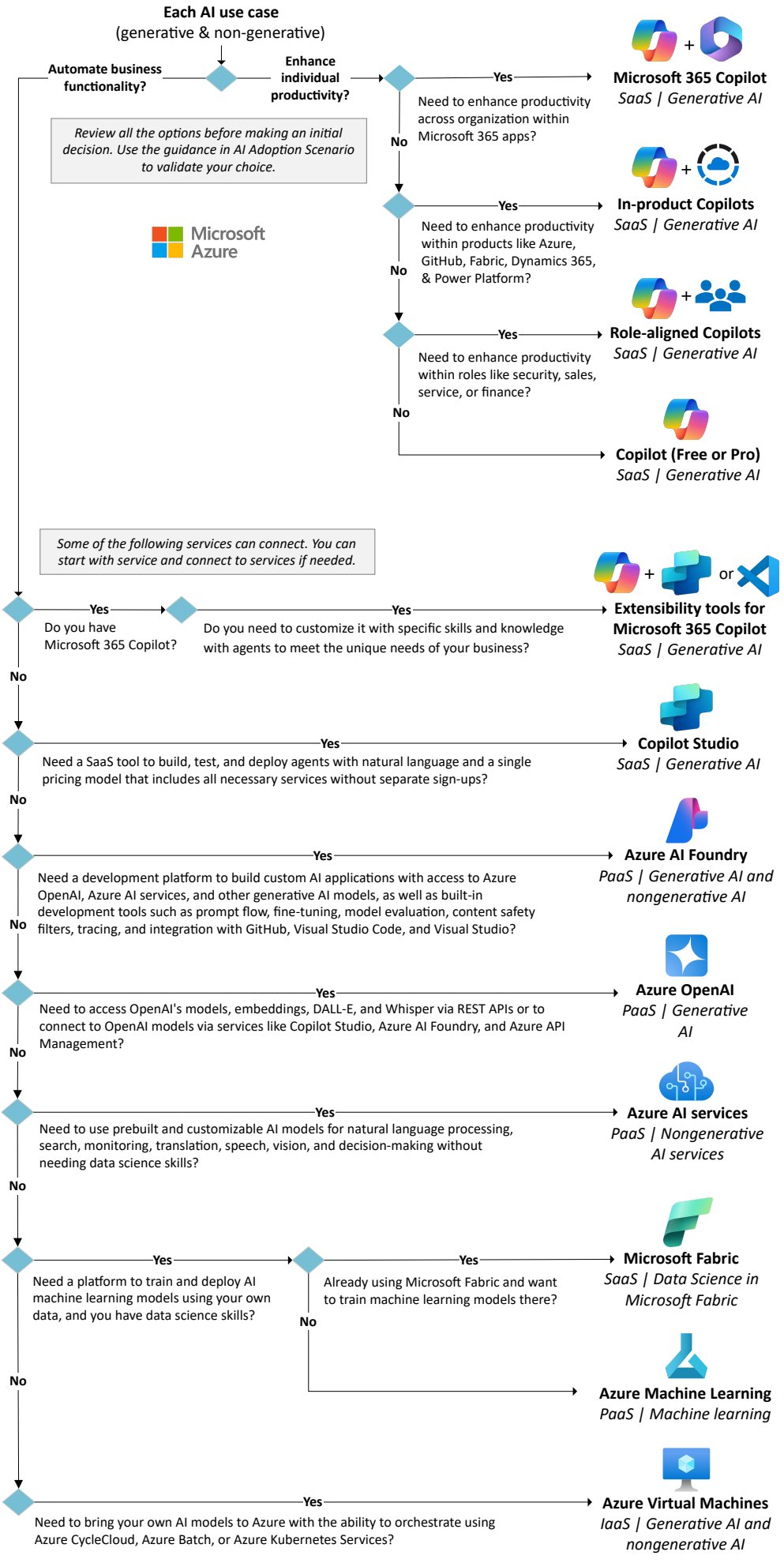
AI enhances individual efficiency and improves business processes. Generative AI fosters productivity and enhances customer experiences. Nongenerative AI, such as machine learning, is ideal for analyzing structured data and automating repetitive tasks. With this understanding, identify areas across your business where AI could add value. For more information, see example AI use cases.

- *Look for automation opportunities.* Identify processes suitable for automation to improve efficiency and reduce operational costs. Focus on repetitive tasks, data-heavy operations, or areas with high error rates where AI can have a significant effect.
- *Conduct an internal assessment.* Gather input from various departments to identify challenges and inefficiencies that AI could address. Document workflows and gather input from stakeholders to uncover opportunities for automation, insight generation, or improved decision-making.
- *Explore industry use cases.* Research how similar organizations or industries use AI to solve problems or enhance operations. Use tools like the AI architectures in the Azure Architecture Center for inspiration and to evaluate which approaches might suit your needs.
- *Set AI targets.* For each identified use case, clearly define the goal (general purpose), objective (desired outcome), and success metric (quantifiable measure). These elements serve as benchmarks to guide your AI adoption and measure its impact.

For more information, see example AI strategy.

Define an AI technology strategy

An AI technology strategy focuses on selecting the most suitable tools and platforms for your generative and nongenerative AI use cases. Microsoft offers a range of options, including software-as-a-service (SaaS), platform-as-a-service (PaaS), and infrastructure-as-a-service (IaaS), each with varying levels of customization and shared responsibility between you and Microsoft. To guide your decision, use the following AI decision tree. For each service, evaluate the skills, data, and budget required to be successful with that service. There's guidance in this article to help with this evaluation process.


Each AI use case
(generative & non-generative)
Automate business functionality?
Enhance individual productivity?
Review all the options before making an initial decision. Use the guidance in AI Adoption Scenario to validate your choice.
Microsoft Azure
Yes
Need to enhance productivity across organization within Microsoft 365 apps?
Microsoft 365 Copilot
SaaS | Generative AI
No
Yes
Need to enhance productivity within products like Azure, GitHub, Fabric, Dynamics 365, & Power Platform?
In-product Copilots
SaaS | Generative AI
No
Yes
Need to enhance productivity within roles like security, sales, service, or finance?
Role-aligned Copilots
SaaS | Generative AI
No
Copilot (Free or Pro)
SaaS | Generative AI
Some of the following services can connect. You can start with service and connect to services if needed.
Yes
Do you have Microsoft 365 Copilot?
Yes
Do you need to customize it with specific skills and knowledge with agents to meet the unique needs of your business?
or
Extensibility tools for Microsoft 365 Copilot
SaaS | Generative AI
No
Yes
Need a SaaS tool to build, test, and deploy agents with natural language and a single pricing model that includes all necessary services without separate sign-ups?
Copilot Studio
SaaS | Generative AI
No
Yes
Need a development platform to build custom AI applications with access to Azure OpenAI, Azure AI services, and other generative AI models, as well as built-in development tools such as prompt flow, fine-tuning, model evaluation, content safety filters, tracing, and integration with GitHub, Visual Studio Code, and Visual Studio?
Azure AI Foundry
PaaS | Generative AI and nongenerative AI
No
Yes
Need to access OpenAI's models, embeddings, DALL-E, and Whisper via REST APIs or to connect to OpenAI models via services like Copilot Studio, Azure AI Foundry, and Azure API Management?
Azure OpenAI
PaaS | Generative AI
No
Yes
Need to use prebuilt and customizable AI models for natural language processing, search, monitoring, translation, speech, vision, and decision-making without needing data science skills?
Azure AI services
PaaS | Nongenerative AI services
No
Yes
Need a platform to train and deploy AI machine learning models using your own data, and you have data science skills?
Yes
Already using Microsoft Fabric and want to train machine learning models there?
Microsoft Fabric
SaaS | Data Science in Microsoft Fabric
No
Azure Machine Learning
PaaS | Machine learning
No
Yes
Need to bring your own AI models to Azure with the ability to orchestrate using Azure CycleCloud, Azure Batch, or Azure Kubernetes Services?
Azure Virtual Machines
IaaS | Generative AI and nongenerative AI

Buy AI software services (SaaS)

Microsoft offers various Copilot generative AI services that enhance individual efficiency. These Copilots allow you to purchase software as a service (SaaS) for AI capabilities across your business or for specific users. SaaS products generally require minimal technical skills.

In terms of data needed, *Microsoft 365 Copilot* uses enterprise data in Microsoft Graph. You can categorize your data with sensitivity labels. *Role-based Copilots* have different data-connection and plug-in options to ingest data. Most *in-product Copilots* don't require extra data preparation. *Extending Microsoft 365 Copilot* allows you to add more data via Microsoft Graph or declarative agents that can pull from different data sources. *Copilot Studio* automates much of the data processing needed to create custom copilots for various business applications. Use the links in the following table for more information.

Expand table

Microsoft Copilots	Description	User	Data needed	Skills required	Main cost factors
Microsoft 365 Copilot	Use Microsoft 365 Copilot for an enterprise-wide solution that automates work in Microsoft 365 apps and provides an enhanced-security way to chat with business data in Microsoft Graph.	Business	Yes	General IT and data management	License
Role-based Copilots	Use Microsoft Copilot for Security and role-based agents for Microsoft 365 to enhance productivity for specific business roles. Role-based agents include Microsoft 365 Copilot for Sales, Microsoft 365 Copilot for Service, and Microsoft 365 Copilot for Finance.	Business	Yes	General IT and data management	Licenses or Security Compute Units (Copilot for Security)
In-product Copilots	Use Copilots to enhance productivity within Microsoft products. Products with in-product Copilots include GitHub, Power Apps, Power BI, Dynamics 365, Power Automate, and Azure.	Business and individual	Yes	None	Free or subscription
Copilot Free or Pro	Use the free version for browser-based access to Azure OpenAI models. Use Copilot Pro for better performance and more capacity.	Individual	No	None	None for Copilot Free or subscription for Copilot Pro
Extensibility tools for Microsoft 365 Copilot	Customize (extend) Microsoft 365 Copilot with more data (knowledge) via Microsoft Graph connectors or capabilities (skills) via declarative agents. To build declarative agents, use extensibility tools such as Copilot Studio (SaaS development), agent builder, Teams toolkit in VS Code (pro-code option), and Sharepoint.	Business and individual	Yes	Data management, general IT, or developer skills	Microsoft 365 Copilot license
Copilot Studio	Use Copilot Studio to build test, and deploy agents in a SaaS authoring environment.	Developer	Yes	Using a platform to connect data sources, map out prompts, and deploy copilots to various locations	License

Build AI workloads with Azure platforms (PaaS)

Microsoft provides various platform-as-a-service (PaaS) options for building AI workloads. The platform you choose depends on your AI goals, required skills, and data needs. Azure offers platforms suitable for different expertise levels, from beginner-friendly tools to advanced options for experienced developers and data scientists. Review the pricing pages and use the Azure pricing calculator to estimate costs.

Expand table

AI goal	Microsoft solution	Data needed	Skills required	Main cost factors
Build RAG applications with a code-first platform	Azure AI Foundry or Azure OpenAI	Yes	Selecting models, orchestrating dataflow, chunking data, enriching chunks, choosing indexing, understanding query types (full-text, vector, hybrid), understanding filters and facets, performing reranking, engineering prompt flow, deploying endpoints, and consuming endpoints in apps	Compute, number of tokens in and out, AI services consumed, storage, and data transfer
Fine-tune generative AI models	Azure AI Foundry	Yes	Preprocessing data, splitting data into training and validation data, validating models, configuring other parameters, improving models, deploying models, and consuming endpoints in apps	Compute, number of tokens in and out, AI services consumed, storage, and data transfer
Train and inference machine learning models by using your own data	Azure Machine Learning or Microsoft Fabric	Yes	Preprocessing data, training models by using code or automation, improving models, deploying machine learning models, and consuming endpoints in apps	Compute, storage, and data transfer
Consume nongenerative AI models in applications	Azure AI services	Yes	Picking the right AI model, securing endpoints, consuming endpoints in apps, and fine-tuning as needed	Use of model endpoints consumed, storage, data transfer, compute (if you train custom models)

Bring your own models with infrastructure services (IaaS)

For organizations needing more control and customization, Microsoft offers infrastructure-as-a-service (IaaS) solutions. While Azure platforms (PaaS) are preferred for AI workloads, Azure Virtual Machines through CycleCloud and Azure Kubernetes Service provides access to GPUs and CPUs for advanced AI needs. This setup allows you to bring your own models to Azure. Refer to the relevant pricing pages and the Azure pricing calculator.

Expand table

AI goal	Microsoft solution	Data needed	Skills required	Main cost factors
Train and inference your own AI models. Bring your own models to Azure.	Azure Virtual Machines or Azure Kubernetes Service	Yes	Infrastructure management, IT, program installation, model training, model benchmarking, orchestration, deploying endpoints, securing endpoints, and consuming endpoints in apps	Compute, compute node orchestrator, managed disks (optional), storage services, Azure Bastion, and other Azure services used

For more information, see example AI strategy.

Define an AI data strategy

For each AI use case, you should define an AI data strategy. The data strategy should outline data collection, storage, and usage practices aligning with regulatory, ethical, and operational standards. Tailor the strategy to each use case to ensure reliable AI outputs and promote data security and privacy. If needed, you can consolidate these individual strategies into a broader summary data strategy for your organization.

- *Establish data governance.* Specify how you collect, store, process, version, and retire data for each AI use case. Include retention and disposal policies, and use version control to maintain accuracy during updates.
- *Plan the data lifecycle.* Define guidelines for collecting, storing, processing, versioning, and retiring data. Include recommendations for retention and disposal policies, emphasizing version control to maintain data accuracy.
  - *Data collection:* Identify data sources such as databases, APIs, IoT devices, third-party data, or Azure Data Factory for ingestion.
  - *Data storage:* Recommend storage solutions appropriate to different types and volumes of data, including structured, unstructured, and real-time data
  - *Data process:* Use ETL (Extract, Transform, Load) or ELT pipelines to clean, transform, and prepare data. Tools such as Shortcuts or Mirroring in Microsoft Fabric can streamline these processes.

- *Set up AI fairness and bias controls.* Establish clear procedures to identify and mitigate bias in AI data. Use tools like Fairlearn to ensure models produce fair and equitable outcomes, particularly for sensitive data attributes.
- *Promote collaboration between AI and data teams.* Align AI development with data engineering efforts to ensure models are built using high-quality, well-managed data.
- *Prepare for data scalability.* Forecast the volume, velocity, and variety of data needed for this AI workload. Choose flexible architectures capable of scaling according to demand. Consider cloud-based infrastructure to manage resources efficiently.
- *Incorporate data management automation.* Utilize AI and machine learning for tasks such as tagging, cataloging, and conducting data quality checks. Automation enhances accuracy and allows teams to focus on strategic goals.
- *Plan for continuous monitoring and evaluation.* Establish regular audits of data and model outputs to ensure ongoing data quality, performance, and fairness. Monitor AI models and data pipelines to identify any shifts that might impact reliability or compliance. Implement automated data quality checks, including anomaly detection and validation rules. Regularly monitor data pipelines for failures or inconsistencies.

Define a responsible AI strategy

For each AI use case, you should define a responsible AI strategy that outlines your role in ensuring AI solutions remain trustworthy and beneficial for all users. Responsibilities might vary depending on the technology adopted in each case. If necessary, create a broader summary responsible AI strategy that encompasses overarching principles derived from individual use cases.

- *Establish AI accountability.* As AI technology and regulations advance, assign someone to monitor and govern these changes. It's typically a responsibility of the AI CoE or an AI lead.
- *Align with established responsible AI principles.* Microsoft follows six responsible AI principles that adhere to the NIST Artificial Intelligence Risk Management Framework (AI RMF). Use these principles as business goals to define success and govern your AI adoption in each use case.
- *Identify responsible AI tools.* Responsible AI tools ensure that your AI aligns with broader responsible AI practices. As part of your strategy, identify which Responsible AI tools and processes are relevant.
- *Understand legal and regulatory compliance requirements.* Legal and regulatory compliance influence how you build and manage AI workloads. Research and adhere to the requirements governing AI where you operate.

For more information, see example AI strategy.

Next step

AI Plan

Example AI use cases

These examples highlight various generative and nongenerative AI applications. While not exhaustive, they provide insights into how AI can be applied to different areas of your business.

Expand table

Generative AI	Nongenerative AI
Autonomous agents: Develop AI systems that perform tasks independently, such as virtual assistants managing schedules or customer inquiries.	Image recognition: Utilize AI to identify and classify objects within images or videos, useful in security or quality control systems.
Marketing: Automatically create social media posts and email newsletters.	Prediction: Forecast trends or optimize operations based on historical data.
E-commerce platforms: Generate personalized product recommendations and tailored shopping experiences.	Process automation: Automate routine tasks and workflows that don't require content generation, such as customer service bots.
Product design: Quickly create multiple variations of product prototypes or design elements.	Data analysis: Uncover patterns in structured data for insights and data-driven decisions.
Software development: Automate repetitive code generation, such as CRUD operations.	Model simulation: Simulate complex workloads (fluid dynamics, finite element analysis) to predict behavior and optimize designs or processes.

Generative AI	Nongenerative AI
Educational platforms: Generate personalized study materials for students.	Anomaly detection: Identify unusual patterns in data. You can use this strategy for fraud detection or equipment failure prediction, for example.
Customer service: Provide context-based responses through AI-driven chatbots.	Recommendation: Offer personalized recommendations based on user behavior, commonly used in e-commerce and streaming services.
Advertising agencies: Create targeted ad variations for different audience segments.	Optimization: Improve efficiency by solving complex problems (supply chain optimization, resource allocation).
Health and wellness apps: Generate customized workout routines and meal plans.	Sentiment analysis: Analyze text from social media or customer reviews to gauge public sentiment and enhance the customer experience.

Example AI strategy

This example AI strategy is based on a fictional company, Contoso. Contoso operates a customer-facing e-commerce platform and employs sales representatives who need tools to forecast business data. The company also manages product development and inventory for production. Its sales channels include both private companies and highly regulated public sector agencies.

Expand table

AI use case	Goals	Objectives	Success metrics	AI approach	Microsoft solution	Data needs	Skill needs	Cost factors	AI data strategy	Responsible AI strategy
E-commerce web application chat feature	Automate business process	Improve customer satisfaction	Increased customer retention rate	PaaS, generative AI, RAG	Azure AI Foundry	Item descriptions and pairings	RAG and cloud app development	Usage	Establish data governance for customer data and implement AI fairness controls.	Assign AI accountability to AI CoE and align with Responsible AI principles.
Internal app document-processing workflow	Automate business process	Reduce costs	Increased completion rate	Analytical AI, fine-tuning	Azure AI services - Document Intelligence	Standard documents	App development	Estimated usage	Define data governance for internal documents and plan data lifecycle policies.	Assign AI accountability and ensure compliance with data handling policies.
Inventory management and product purchasing	Automate business process	Reduce costs	Shorter shelf life of inventory	Machine learning, training models	Azure Machine Learning	Historical inventory and sales data	Machine learning and app development	Estimated usage	Establish governance for sales data and detect and address biases in data.	Assign AI accountability and comply with financial regulations.
Daily work across company	Enhance individual productivity	Improve employee experience	Increased employee satisfaction	SaaS generative AI	Microsoft 365 Copilot	OneDrive data	General IT	Subscription costs	Implement data governance for employee data and ensure data privacy.	Assign AI accountability and utilize built-in responsible AI features.
E-commerce app for regulated industry chat feature	Automate business process	Increase sales	Increased sales	IaaS generative AI model training	Azure Virtual Machines	Domain-specific training data	Cloud infrastructure and app development	Infrastructure and software	Define governance for regulated data and plan lifecycle with	Assign AI accountability and adhere to industry regulations.

AI use case	Goals	Objectives	Success metrics	AI approach	Microsoft solution	Data needs	Skill needs	Cost factors	AI data strategy	Responsible AI strategy
									compliance measures.	

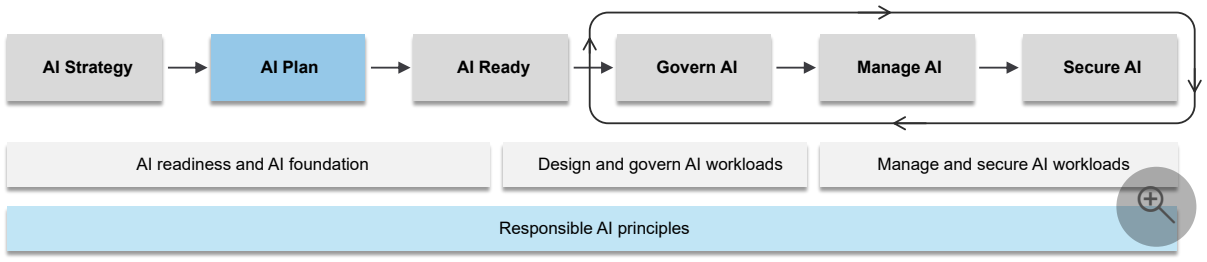
Feedback

Was this page helpful? Yes No

AI Plan - Process to plan for AI adoption

Article • 12/06/2024

This article outlines the organizational process for planning AI adoption. An AI adoption plan details the steps an organization must take to integrate AI into its operations. This plan ensures alignment between AI initiatives and business goals. It helps organizations allocate resources, develop skills, and deploy technology for effective AI adoption.



Assess AI skills

In your technology strategy, you identified AI uses cases and AI solutions for each. These solutions require certain AI skills to adopt. Evaluate your current AI skills and identify gaps to address before proceeding. An AI maturity assessment helps determine your readiness to implement AI. It also guides the selection of use cases that match your capabilities and expedites your success. Use the following table to assess your AI maturity level. For more information, see Technical Assessment for Generative AI in Azure.

Expand table

AI maturity level	Skills required	Data readiness	Feasible AI use cases
Level 1	▪ Basic understanding of AI concepts ▪ Ability to integrate data sources and map out prompts	▪ Minimal to zero data available ▪ Enterprise data available	▪ Azure quickstart (*see table*) ▪ Copilot Studio app
Level 2	▪ Experience with AI model selection ▪ Familiarity with AI deployment and endpoint management	▪ Minimal to zero data available ▪ Small, structured dataset ▪ Small amount of	▪ Any of the previous projects ▪ Custom analytical AI workload that uses Azure AI services ▪ Custom generative AI chat app without Retrieval Augmented Generation (RAG) in Azure AI

AI maturity level	Skills required	Data readiness	Feasible AI use cases
	▪ Experience with data cleaning and processing	domain-specific data available	Foundry ▪ Custom machine learning app with automated model training ▪ Fine-tuning a generative AI model
Level 3	▪ Proficiency in prompt engineering ▪ Proficiency in AI model selection, data chunking, and query processing ▪ Proficiency in data preprocessing, cleaning, splitting, and validating ▪ Grounding data for indexing	▪ Large amounts of historical business data available for machine learning ▪ Small amount of domain-specific data available	▪ Any of the previous projects ▪ Generative AI app with RAG in Azure AI Foundry (or Azure Machine Learning) ▪ Training and deploying a machine learning model in Machine Learning ▪ Training and running a small AI model on Azure Virtual Machines
Level 4	▪ Advanced AI / machine learning expertise, including infrastructure management ▪ Proficiency in handling complex AI model training workflows ▪ Experience with orchestration, model benchmarking, and performance optimization ▪ Strong skills in securing and managing AI endpoints	▪ Large amounts of data available for training	▪ Any of the previous projects ▪ Training and running a large generative or nongenerative AI app on Virtual Machines, Azure Kubernetes Service, or Azure Container Apps

Acquire AI skills

Acquiring AI skills requires organizations to assess their current talent pool and determine whether to upskill, recruit, or partner with external experts. Assess your current talent pool to identify needs for upskilling, recruiting, or external partnerships. Building a skilled AI team ensures you can adapt to challenges and handle various AI projects. AI constantly evolves, so maintaining a culture of continuous learning supports innovation and keeps skills current.

- *Learn AI skills.* Use the AI learning hub platform for free AI training, certifications, and product guidance. Set certification goals, such as earning Azure AI Fundamentals, Azure AI Engineer Associate, and Azure Data Scientist Associate certifications.

- *Recruit AI professionals.* For expertise beyond your internal capabilities, recruit AI professionals experienced in model development, generative AI, or AI ethics. These professionals are in high-demand. Consider collaborating with educational institutions to access fresh talent. Make sure to update job descriptions to reflect evolving AI needs, and offer competitive compensation. Create an attractive employer brand. Showcase your organization's commitment to innovation and technological advancement, making your brand appealing to AI professionals.

- *Use Microsoft partners to acquire AI skills.* Use the Microsoft partners marketplace to address skill shortages and meet time constraints. Microsoft partners provide AI, data, and Azure expertise across various industries.

Access AI resources

As a tactical step to developing AI solutions, you need to be able to access them. The goal is to provide a quick way to understand and access what you need to start using Microsoft AI solutions.

- *Access Microsoft 365 Copilot.* Most Microsoft SaaS Copilots require a license or an add-on subscription. Microsoft 365 Copilot requires a Microsoft 365 business or enterprise license to which you add on the Copilot license.

- *Access Microsoft Copilot Studio.* Microsoft Copilot Studio uses a standalone license or an add-on license.

- *Access in-product Copilots.* In-product Copilots have different access requirements for each, but access to the primary product is required. For more information on each, see GitHub, Power Apps, Power BI, Dynamics 365, Power Automate, and Azure.

- *Access role-based Copilots.* Role-based Copilots also have their own access requirements. For more information, see Role-based agents for Microsoft 365 Copilot and Microsoft Copilot for Security.

- *Access Azure AI resources.* Azure PaaS and IaaS solutions require an Azure account. These services include Azure OpenAI Service, Azure AI Foundry, Azure Machine Learning, Azure AI services, Azure Virtual Machines, and Azure CycleCloud.

Prioritize AI use cases

After assessing skills, resources, and AI maturity, prioritize AI use cases identified in your AI Strategy. This prioritization ensures you focus on projects that offer the greatest value, align with business goals, and match your current capabilities. Follow these steps:

- *Assess skills and resources*. After acquiring AI skills, review your current AI maturity, available data, and resource access. This assessment helps reset priorities based on what's possible.
- *Evaluate use cases*. Prioritize projects based on their feasibility and strategic value they add to your organization. Align AI use cases with your strategic objectives to ensure that efforts contribute to overall success.
- *Select top use cases*. Create a shortlist of high-priority AI use cases that form the basis for further exploration and testing.

Create an AI proof of concept

Developing an AI proof of concept (PoC) validates the feasibility and potential value of a prioritized use case on a smaller scale. The PoC process helps refine use case priority, reduce risk, and identify challenges before moving to full-scale deployment. This iterative approach lets you adjust your AI plan based on real-world insights.

- *Select the right opportunity*. From your shortlist of AI use cases, choose a high-value project that aligns with your AI maturity level. Ideally, start with an internal project, not customer facing. Internal projects minimize risk and provide a foundation for testing the workload. Use the PoC to validate the approach and refine it before expanding to production. Conduct A/B testing to establish what works and gather baseline data.
- *Start with an Azure quickstart guide*. Azure offers step-by-step guidance for creating basic applications using its AI platforms. These guides, called quickstarts, help you deploy an application and include instructions for deleting it afterward. Quickstarts provide a simple way to familiarize your organization with the technology.

Expand table

AI type	Azure AI quickstart guide
Generative AI	Azure AI Foundry, Azure OpenAI, Copilot Studio

AI type	Azure AI quickstart guide
Machine learning	Azure Machine Learning
Analytical AI	Azure AI services: Azure AI Content Safety, Azure AI Custom Vision, Document Intelligence Studio, Face service, *Azure AI Language, Azure AI Speech, *Azure AI Translator, Azure AI Vision. **Each feature of this AI service has its own quickstart guide.*

- *Reprioritize AI opportunities*. Use the insights gained from the PoC to refine your list of AI use cases. If the PoC presents unexpected challenges, adjust your priorities and focus on more feasible projects.

Implement responsible AI

Responsible AI adoption requires incorporating ethical frameworks and regulatory practices into your AI implementation plan. This approach ensures that AI initiatives align with organizational values, protect user rights, and comply with legal standards.

- *Use responsible AI planning tools*. To integrate responsible AI principles into your adoption process, use tools and frameworks that support ethical AI practices. Microsoft offers several resources.

Expand table

Responsible AI planning tool	Description
AI impact assessment template	Evaluate potential social, economic, and ethical impacts of AI initiatives.
Human-AI eXperience Toolkit	Design AI systems that prioritize user well-being and foster positive interactions.
Responsible AI Maturity Model	Assess and advance your organization's maturity in implementing responsible AI practices.
Responsible AI for workload teams	Recommendations for workload teams to implement responsible AI when building workloads in Azure.

- *Start the AI governance process*. Responsible AI adoption involves creating governance policies to guide AI projects and monitor AI system behaviors. Start by identifying organizational risks specific to your AI initiatives. Document governance

policies that outline responsibilities, compliance requirements, and ethical standards. See the article on Govern AI for details on this process.

Govern AI

- *Start the AI management process.* AI management frameworks, such as GenAIOps or MLOps, help ensure ongoing adherence to responsible AI principles as your AI systems evolve. These practices involve deployment management, continuous monitoring, and cost optimization for AI models in production. See the article on Manage AI for details on this process.

  Manage AI

- *Start the AI security process.* Security forms a critical part of responsible AI adoption. Regular security assessments help protect the confidentiality, integrity, and availability of your AI systems. Conduct risk assessments that address potential security threats specific to AI, such as adversarial attacks or data breaches. See the article on Secure AI for details on this process.

  Secure AI

Estimate delivery timelines

Estimating delivery timelines involves setting realistic schedules and milestones for AI project implementation. Clear timelines allow organizations to allocate resources effectively and manage stakeholder expectations, supporting a structured progression from proof of concept to production. By establishing specific milestones, organizations can measure their progress, identify potential delays, and make adjustments to keep projects on track and within budget.

Based on your PoC, assign a delivery timeline for your AI opportunities. Create a timeline with clear milestones and deliverables for implementing selected use cases. Assign teams, define roles, and secure necessary tools or partnerships. Microsoft AI SaaS solutions provide the shortest timelines for seeing a return on investment. Timelines for building AI apps on Azure PaaS and IaaS solutions depend on your use case and AI maturity. In most cases, it takes weeks or months before you have a production-ready AI workload.

Next step

To ***build*** AI workloads with Azure PaaS or IaaS services, follow the AI Ready guidance to establish your AI foundation. If you decided to ***buy*** a Microsoft Copilot SaaS solution, skip to the Govern AI guidance to establish organizational governance for AI.

AI Ready (only for AI PaaS and IaaS adoption)

Govern AI (for AI SaaS, PaaS, and IaaS adoption)

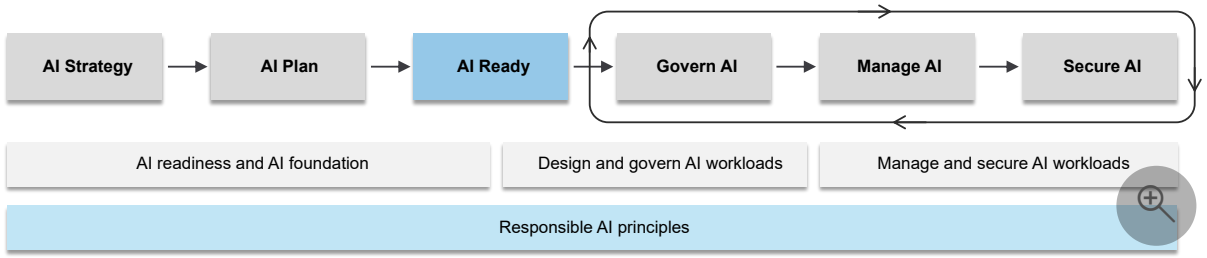
Feedback

Was this page helpful? Yes No

AI Ready – Recommendations for organizations building AI workloads in Azure

Article • 11/01/2024

This article outlines the organizational process for building AI workloads in Azure. The article provides recommendations for making key design and process decisions for adopting AI workloads at scale. It focuses on AI-specific guidance for region selection, resource organization, and networking.



Establish AI reliability

AI reliability involves selecting appropriate regions to host AI models to ensure consistent performance, compliance, and availability. Organizations must address redundancy, failover, and performance optimization to maintain reliable AI services.

- *Use multiple regions to host AI model endpoints.* For production workloads, host AI endpoints in at least two regions to provide redundancy and ensure high availability. Although generative AI models are stateless, hosting them in multiple regions ensures faster failover and recovery during regional failures. For Azure OpenAI Service models, you can use global deployments. These multiregion deployments can automatically and transparently route requests to a region that has enough capacity. If you choose a nonglobal deployment, also known as a regional deployment, use Azure API Management for load balancing API requests to AI endpoints.

- *Confirm service availability.* Before deployment, ensure that there's availability in the region for the AI resources that you need. Certain regions might not provide specific AI services or might have limited features, which can affect the functionality of your solution. This limitation can also affect the scalability of your deployment. For example, Azure OpenAI service availability can vary based on your deployment model. These deployment models include global standard, global

provisioned, regional standard, and regional provisioned. Check the AI service to confirm that you have access to the necessary resources.

- *Evaluate region quota and capacity.* Consider the quota or subscription limits in your chosen region as your AI workloads grow. Azure services have regional subscription limits. These limits can affect large-scale AI model deployments, such as large inference workloads. To prevent disruptions, contact Azure support in advance if you foresee a need for extra capacity.

- *Evaluate performance.* When you build applications that need to retrieve data, such as retrieval-augmented-generation (RAG) applications, it's important to consider data storage locations to optimize performance. You don't have to colocate data with models in RAG apps, but doing so can improve performance by reducing latency and ensuring efficient data retrieval.

- *Prepare for continuity of operations.* To ensure business continuity and disaster recovery, replicate critical assets such as fine-tuned models, RAG data, trained models, and training datasets in a secondary region. This redundancy enables faster recovery if there's an outage and ensures continued service availability.

Establish AI governance

AI governance encompasses organizing resources and applying policies to manage AI workloads and costs. It involves structuring management groups and subscriptions to ensure compliance and security across different workloads. Proper AI governance prevents unauthorized access, manages risks, and ensures that AI resources operate efficiently within the organization.

- *Separate internet facing and internal AI workloads*. At a minimum, use management groups to separate AI workloads into internet-facing ("online") and internal only ("corporate"). The distinction provides an important data governance boundary. It helps you keep internal separate from public data. You don't want external users to access sensitive business information required for internal work. This distinction between internet-facing and internal workloads aligns with Azure landing zone management groups.

- *Apply AI policies to each management group.* Start with baseline policies for each workload type, such as those policies used in Azure landing zones. Add more Azure Policy definitions to your baseline to drive uniform governance for Azure AI services, Azure AI Search, Azure Machine Learning, and Azure Virtual Machines.

- *Deploy AI resources in workload subscriptions.* AI resources need to inherit workload governance policies from the workload management group (internal or internet-facing). Keep them separate from platform resources. AI resources controlled by platform teams tend to create development bottlenecks. In the context of Azure landing zone, deploy AI workloads to application landing zone subscriptions.

Establish AI networking

AI networking refers to the design and implementation of network infrastructure for AI workloads, including security and connectivity. It involves using topologies like hub-and-spoke, applying security measures such as DDoS protection, and ensuring efficient data transfer. Effective AI networking is critical for secure and reliable communication, preventing network-based disruptions and maintaining performance.

- *Activate Azure DDoS Protection for internet-facing AI workloads.* Azure DDoS Protection safeguards your AI services from potential disruptions and downtime caused by distributed denial of service attacks. Enable Azure DDoS protection at the virtual network level to defend against traffic floods targeting internet-facing applications.

- *Connect to on-premises data.* For organizations transferring large amounts of data from on-premises sources to cloud environments, use a high-bandwidth connection.

    - *Consider Azure ExpressRoute.* Azure ExpressRoute is ideal for high data volumes, real-time processing, or workloads that require consistent performance. It has FastPath feature that improves data path performance.

    - *Consider Azure VPN Gateway.* Use Azure VPN Gateway for moderate data volumes, infrequent data transfer, or when public internet access is required. It's simpler to set up and cost-effective for smaller datasets than ExpressRoute. Use the correct topology and design for your AI workloads. Use site-to-site VPN for cross-premises and hybrid connectivity. Use a point-to-site VPN for secure device connectivity. For more information, see Connect an on-premises network to Azure.

- *Prepare domain name resolution services.* When you use private endpoints, integrate private endpoints with DNS for proper DNS resolution and successful private endpoint functionality. Deploy Azure DNS infrastructure as part of your Azure landing zone and configure conditional forwarders from existing DNS services for the appropriate zones. For more information, see Private Link and DNS integration at scale for Azure landing zones.

- *Configure network access controls.* Utilize network security groups (NSGs) to define and apply access policies that govern inbound and outbound traffic to and from AI workloads. These controls can be used to implement the principle of least privilege, ensuring that only essential communication is permitted.

- *Use network monitoring services.* Use services such as Azure Monitor Network Insights and Azure Network Watcher to gain visibility into network performance and health. Additionally, use Microsoft Sentinel for advanced threat detection and response across your Azure network.

- *Deploy Azure Firewall to inspect and secure outbound Azure workload traffic.* Azure Firewall enforces security policies for outgoing traffic before it reaches the internet. Use it to control and monitor outgoing traffic and enable SNAT to conceal internal IP addresses by translating private IPs to the firewall's public IP. It ensures secure and identifiable outbound traffic for better monitoring and security.

- *Use Azure Web Application Firewall (WAF) for internet-facing workloads.* Azure WAF helps protect your AI workloads from common web vulnerabilities, including SQL injections and cross-site scripting attacks. Configure Azure WAF on Application Gateway for workloads that require enhanced security against malicious web traffic.

Establish an AI foundation

An AI foundation provides the core infrastructure and resource hierarchy that support AI workloads in Azure. It includes setting up scalable, secure environments that align with governance and operational needs. A strong AI foundation enables efficient deployment and management of AI workloads. It also ensures security and flexibility for future growth.

Use Azure landing zone

An Azure landing zone is the recommended starting point that prepares your Azure environment. It provides a predefined setup for platform and application resources. Once the platform is in place, you can deploy AI workloads to dedicated application landing zones. Figure 2 below illustrates how AI workloads integrate within an Azure landing zone.

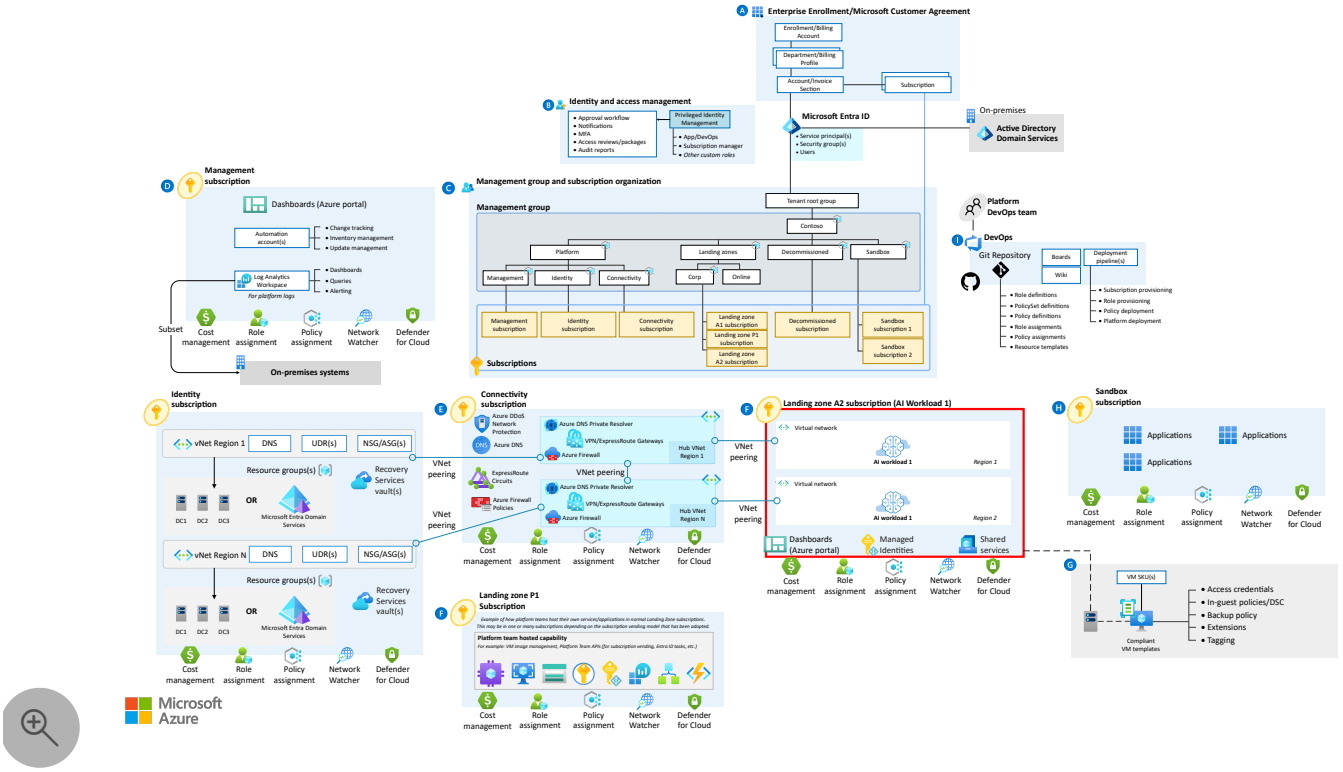


Figure 2. AI workload in an Azure landing zone.

Build an AI environment

If you don't use an Azure landing zone, follow the recommendations in this article to build your AI environment. The following diagram shows a baseline resource hierarchy. It segments internal AI workloads and internet-facing AI workloads, as described in establish AI governance. Internal workloads use policy to deny online access from customers. This separation safeguards internal data from exposure to external users. AI development uses a jumpbox to manage AI resources and data.

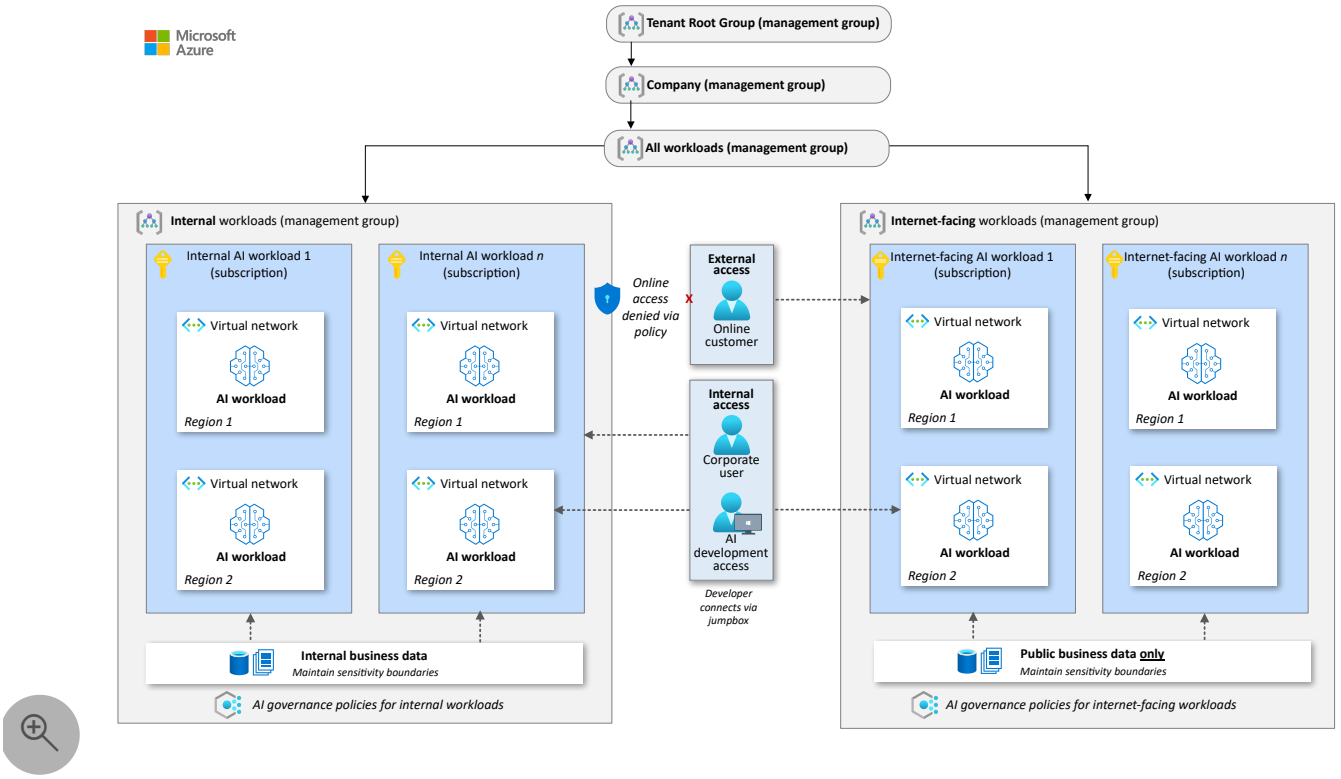


Figure 3. Baseline resource hierarchy for AI workloads.

Next steps

The next step is to build and deploy AI workloads to your AI environment. Use the following links to find the architecture guidance that meets your needs. Start with platform-as-a-service (PaaS) architectures. PaaS is Microsoft's recommended approach to adopting AI.

PaaS AI architecture guidance

IaaS AI architecture guidance

Feedback

Was this page helpful? Yes No

AI architecture guidance to build AI workloads on Azure

Article • 12/04/2024

This article offers architecture guidance for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

The Azure Architecture Center offers reference architectures and guides to help organizations build AI workloads efficiently and securely. These resources provide well-tested, structured frameworks for AI workload deployment. In AI Ready, you established a resource hierarchy that categorizes AI workloads into internal and internet-facing groups. Deploy AI workloads to subscriptions under the appropriate management groups (internal vs. internet-facing). The following tables list articles for building AI workloads.

> **Note**
>
> If you're using Azure landing zones, begin with the ***Baseline Azure OpenAI architecture in Azure landing zone*** and deploy it to an application landing zone subscription.

Generative AI architectures and guides

Expand table

Article	Article type	Target organization
Baseline Azure OpenAI architecture in an Azure landing zone	Architecture	Enterprise
Baseline Azure OpenAI reference architecture	Architecture	Any
Basic Azure OpenAI reference architecture	Architecture	Startup
GenAIOps	Guide	Any
Developing RAG solutions	Guides	Any

Article	Article type	Target organization
Proxy Azure OpenAI usage	Guide	Any
Application design	Design area	Any
Application platform	Design area	Any
Training data design	Design area	Any
Grounding data design	Design area	Any
Data platform	Design area	Any
MLOps and GenAIOps	Design area	Any
Operations	Design area	Any
Test and evaluate	Design area	Any
Responsible AI	Design area	Any

Nongenerative AI architectures and guides

Expand table

Article	Article type	Target organization
Document processing architectures	Architectures	Any
Video and image classification architecture	Architectures	Any
Audio processing architecture	Architecture	Any
Predictive analytics architecture	Architecture	Any
Azure Machine Learning	Guides	Any
MLOps	Guides	Any
Team Data Science Process	Guides	Any

Use the AI design areas as a framework

The AI design areas provide technology-specific framework to design AI workloads with Azure's AI platform-as-a-service (PaaS) solutions. It focuses on Azure AI Foundry, Azure

OpenAI, Azure Machine Learning, and Azure AI Services. Use them to establish standards and best practices related to these services:

- Resource selection
- Networking
- Governance
- Management
- Security

Each design area includes recommendations for both generative and nongenerative AI workloads on Azure, consolidating best practices that apply to all AI workloads using Azure PaaS AI platforms.

Next step

Resource selection

Feedback

Was this page helpful? Yes No

Resource selection recommendations for AI workloads on Azure

Article • 11/01/2024

This article offers resource selection recommendations for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

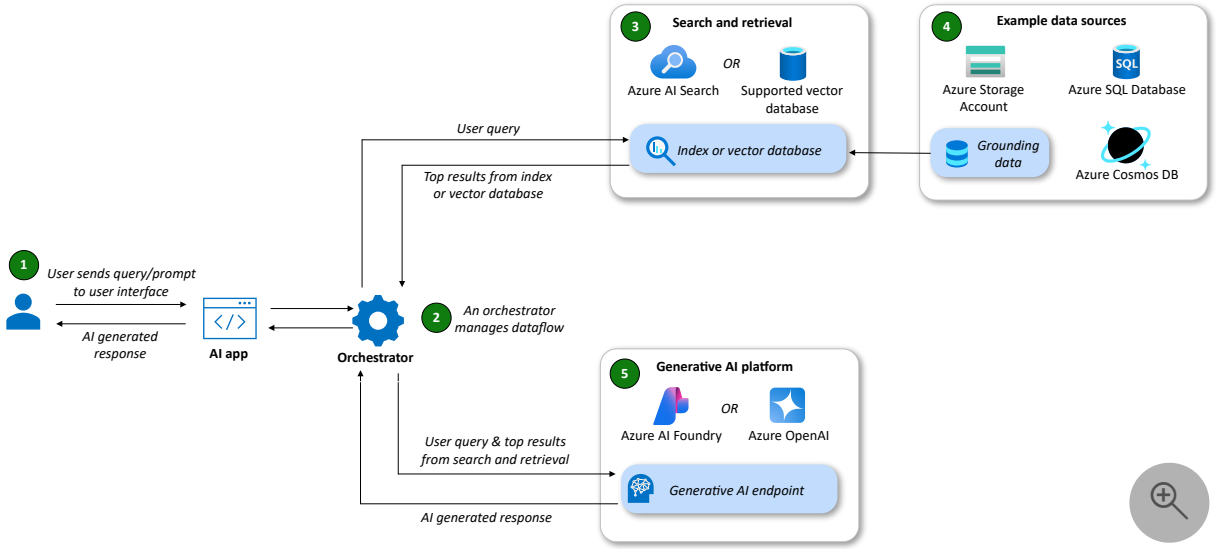
Making informed AI resource choices enables organizations to achieve better performance, scalability, and cost-effectiveness when managing AI workloads. The following table provides an overview of the primary Azure AI PaaS solutions and important decision criteria.

Expand table

AI platform	AI type	Description	Skills required
Azure OpenAI	Generative AI	Platform for accessing OpenAI models	Developer and data science skills
Azure AI Foundry	Generative AI	Platform for prompt engineering and deploy generative AI endpoints	Developer and data science skills
Azure AI services	Analytical AI	Platform for consuming prebuilt machine learning models	Developer skills
Azure Machine Learning	Machine learning	Platform for training and deploying machine learning models	Developer skills and advanced data science skills

Select resources for generative AI workloads

Generative AI requires the combination of different resources to process and generate meaningful outputs based on input data. Proper selection ensures that generative AI applications, such as those using retrieval augmented generation (RAG), deliver accurate by grounding AI models.



In a typical RAG workload, (1) the workload receives the user query. (2) An orchestrator, such as Prompt flow, Semantic Kernel, or LangChain, manages the data flow. (3) A search and retrieval mechanism finds the appropriate (4) grounding data to send to the generative AI endpoint. (5) A generative AI model endpoint generates a response based on the user query and grounding data. Use the following recommendations as framework to build generative RAG workloads.

- *Choose a generative AI platform.* Use Azure OpenAI or Azure AI Foundry to deploy and manage generative AI models. Azure OpenAI Service provides access to OpenAI models private networking, and content filtering. Azure AI Foundry offers a code-first platform for developing AI workloads. It has built-in tools for building and deploying applications. It also features a large model catalog, prompt flow, fine-tuning, content safety filters, and more.

- *Choose the appropriate AI compute type.* Azure AI Foundry requires compute instances for prompt flow, creating indexes, and opening Visual Studio Code (Web or Desktop) within the studio. Choose a compute type based on your performance and budget needs.

- *Pick an orchestrator.* Popular orchestrators for generative AI include Semantic Kernel, Prompt flow, and LangChain. Semantic Kernel integrates with Azure services. LangChain provides extensibility beyond Microsoft's ecosystem.

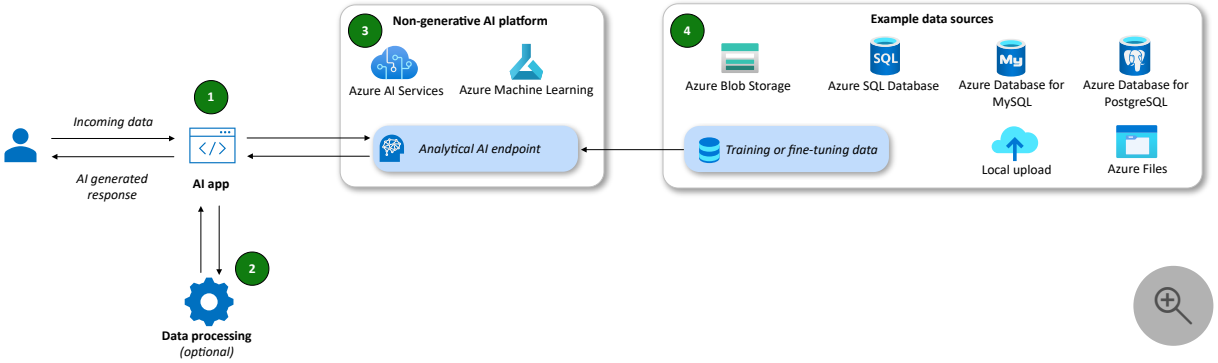
- *Pick a search and knowledge retrieval mechanism.* To ground generative AI models, create an index or vector database for relevant data retrieval. Use Azure AI Search to build traditional and vector indexes from various data sources, apply data chunking, and use multiple query types. If your data resides in structured databases, consider using Azure Cosmos DB, Azure Database for PostgreSQL, and Azure Cache for Redis.

- *Choose a data source for grounding data.* For images, audio, video, or large datasets, store grounding data in Azure Blob Storage. Alternatively, use databases supported by Azure AI Search or vector databases.

- *Pick a compute platform.* Use the Azure compute decision tree to pick the right platform for your workload.

Select resources for nongenerative AI workloads

Nongenerative AI workloads rely on platforms, compute resources, data sources, and data processing tools to support machine learning tasks. Selecting the right resources allows you to build AI workloads using both prebuilt and custom solutions.



In a nongenerative AI workload, (1) the workload ingests data. (2) An optional data processing mechanism extracts or manipulates incoming data. (3) An AI model endpoint analyzes the data. (4) Data supports training or fine-tuning of the AI models. Use the following recommendations as framework to build nongenerative AI workloads.

- *Choose a nongenerative AI platform.* Azure AI services offer prebuilt AI models that don't require data science skills. For guidance on selecting the right Azure AI service, see Choose an Azure AI services technology. Azure Machine Learning provides a platform to build machine learning models with your own data and consume those models in AI workloads.

- *Choose the appropriate AI compute.* For Azure Machine Learning, you need compute resources to run a job or host an endpoint. Use the compute type that meets your performance and budget needs. Azure AI services don't require compute resources.

- *Pick a data source.* For Azure Machine Learning, use one of the supported data sources to host your training data. For Azure AI services, many of the services don't

require fine-tuning data, and some, like Azure AI Custom Vision, provide an option to upload local files to a managed data storage solution.

- *Pick a compute platform.* Use the Azure compute decision tree to pick the right workload platform.
- *Pick a data processing service (optional).* Azure Functions is a common data processing choice because it provides a serverless option. Azure Event Grid is also a common trigger mechanism for kicking off a data processing pipeline.

Next step

Networking PaaS AI

Feedback

Was this page helpful? Yes No

Networking recommendations for AI workloads on Azure

Article • 12/03/2024

This article offers networking recommendations for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

Networking enables secure and efficient connectivity to critical AI resources and is foundational to data integrity and privacy. Effective networking strategies protect sensitive AI workloads from unauthorized access and help optimize performance for AI model training and deployment.

Configure virtual networks

Configuring virtual networks refers to setting up and managing private and secure networking environments for Azure AI platforms. Virtual networks allow organizations to isolate AI workloads and create secure communication channels. Proper configuration ensures that only authorized users and systems can access critical AI resources, and it minimizes exposure to the public internet.

Expand table

AI platform	Virtual network recommendations
Azure AI Foundry	Configure the managed virtual network and use private endpoints. If needed, connect the managed virtual network to on-premises resources.
Azure OpenAI	Restrict access to select virtual networks or use private endpoints.
Azure Machine Learning	Create a secure workspace with a virtual network. Plan for network isolation. Follow the security best practices for Azure Machine Learning.
Azure AI services	Restrict access to select virtual networks or use private endpoints.

Azure AI Foundry and Azure Machine Learning deploy to Microsoft-managed virtual networks and deploy required dependent services. The managed virtual networks use private endpoints to access supporting Azure services like Azure Storage, Azure Key Vault, and Azure Container Registry. Use the links to view the network architectures of these services so you can best configure your virtual network.

Secure virtual networks

Securing virtual networks involves using private endpoints, enforcing DNS zones, and enabling custom DNS servers to protect AI workloads. These strategies limit public internet exposure and prevent unauthorized access. Effective network security is essential for safeguarding sensitive AI models and ensuring privacy compliance.

- *Consider private endpoints.* No PaaS services or AI model endpoints should be accessible from the public internet. Private endpoints to provide private connectivity to Azure services within a virtual network. Private endpoints add complexity to deployments and operations, but the security benefit often outweighs the complexity.

- *Consider creating private endpoints for AI service portals.* Private endpoints provide secure, private access to PaaS portals like Azure AI Foundry and Azure Machine Learning studioF. Set up private endpoints for these global portals in a hub virtual network. This configuration provides secure access to public-facing portal interfaces directly from user devices.

- *Consider enforcing private DNS zones.* Private DNS zones centralize and secure DNS management for accessing PaaS services within your AI network. Set up Azure policies that enforce private DNS zones and require private endpoints to ensure secure, internal DNS resolutions. If you don't have central Private DNS Zones, the DNS forwarding doesn't work until you add conditional forwarding manually. For example, see using custom DNS with Azure AI Foundry hubs and Azure Machine Learning workspace.

- *Enable custom DNS servers and private endpoints for PaaS services.* Custom DNS servers manage PaaS connectivity within the network, bypassing public DNS. Configure private DNS zones in Azure to resolve PaaS service names securely and route all traffic through private networking channels.

Manage connectivity

Managing connectivity controls how AI resources interact with external systems. Techniques like using a jumpbox and limiting outbound traffic help protect AI workloads. Proper connectivity management minimizes security risks and ensures smooth, uninterrupted AI operations.

- *Use a jumpbox for access.* AI development access should use a jumpbox within the virtual network of the workload or through a connectivity hub virtual network. Use Azure Bastion to securely connect to virtual machines interacting with AI services.

Azure Bastion provides secure RDP/SSH connectivity without exposing VMs to the public internet. Enable Azure Bastion to ensure encrypted session data and protect access through TLS-based RDP/SSH connections.

- *Limit outbound traffic from your AI resources*. Limiting outbound traffic from your AI model endpoints helps protect sensitive data and maintain the integrity of your AI models. For minimizing data exfiltration risks, restrict outbound traffic to approved services or fully qualified domain names (FQDNs) and maintain a list of trusted sources. You should only allow unrestricted internet outbound traffic if you need access to public machine learning resources, but regularly monitor and update your systems. For more information, see Azure AI services, Azure AI Foundry, and Azure Machine Learning.

- *Consider a generative AI gateway*. Consider using Azure API Management (APIM) as a generative AI gateway within your virtual networks. A generative AI gateway sits between your front-end and the AI endpoints. Application Gateway, WAF policies, and APIM within the virtual network is an established architecture in generative AI solutions. For more information, see AI Hub architecture and Deploy Azure API Management instance to multiple Azure regions.

- *Use HTTPS for internet to Azure connectivity*. Secure connections using TLS protocols help protect data integrity and confidentiality for AI workloads connecting from the internet. Implement HTTPS through Azure Application Gateway or Azure Front Door. Both services provide encrypted, secure tunnels for internet-originating connections.

Next step

Governance PaaS AI

Feedback

Was this page helpful? Yes No

Governance recommendations for AI workloads on Azure

Article • 01/30/2025

This article offers governance recommendations for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

Effective governance supports the responsible use of AI. It enables businesses to optimize their AI investments while reducing risks associated with security, cost, and regulatory compliance.

Govern AI models

AI model governance refers to the processes for managing AI models to ensure they produce reliable, safe, and ethical outputs. Controls over model inputs and outputs help mitigate risks. These risks include harmful content and unintended AI use. Both could affect users and the organization. These processes support responsible AI deployment, and they safeguard against potential legal and ethical challenges.

- *Control the models you use*. Use Azure Policy to manage which specific models your teams are allowed to deploy from the Azure AI Foundry model catalog. You have the option to use a built-in policy or create a custom policy. Since this approach uses an allowlist, begin with an *audit* effect. The *audit* effect allows you to monitor the models your teams are using without restricting deployments. Only switch to the *deny* effect once you understand the AI development and experimentation needs of workload teams, so you don't hinder their progress unnecessarily. If you switch a policy to *deny*, it doesn't automatically remove noncompliant models that teams have already deployed. You must remediate those models manually.

- *Establish a process to detect AI risks*. Use tools like Defender for Cloud to discover generative AI workloads and explore risks to predeployment generative AI artifacts. Establish a policy to regularly red team generative AI models. Document identified risks and continuously update your AI governance policies to mitigate emerging issues.

- *Define baseline content filters for generative AI models*. Use Azure AI Content Safety to define a baseline content filter for your approved AI models. This safety system

runs both the prompt and completion for your model through a group of classification models. These classification models detect and help prevent the output of harmful content across a range of categories. Content Safety provides features like prompt shields, groundedness detection, and protected material text detection. It scans images and text. Create a process for application teams to communicate different governance needs.

- *Ground generative AI models.* Use system messages and the retrieval augmented generation (RAG) pattern to govern the output of generative AI models. Test the effectiveness of grounding by using tools like prompt flow or the open-source red teaming framework PyRIT.

Govern AI costs

AI cost governance involves managing expenses associated with AI workloads to maximize efficiency and reduce unnecessary spending. Effective cost control ensures that AI investments align with business objectives, which prevents unforeseen costs from over-provisioning or underutilization. These practices enable organizations to optimize their AI operations financially.

- *Use the right billing model.* If you have predictable workloads, use AI commitment tiers in Azure AI services. For Azure OpenAI models, use provisioned throughput units (PTUs), which can be less expensive than pay-as-you-go (consumption-based) pricing. It's common combine PTU endpoints and a consumption-based endpoints for cost optimization. Use PTUs on the AI model primary endpoint and a secondary, consumption-based AI endpoint for spillover. For more information, see Introduce a gateway for multiple Azure OpenAI instances.

- *Choose the right model for your use case.* Select the AI model that meets your needs without incurring excessive costs. Use less expensive models unless the use case demands a more expensive model. For fine-tuning, maximize time usage within each billing period to avoid extra charges. For more information, see Azure OpenAI models and pricing. Also see Azure AI Foundry model catalog and billing information for model deployments.

- *Set provisioning limits.* Allocate provisioning quotas for each model based on expected workloads to prevent unnecessary costs. Continuously monitor dynamic quotas to ensure that they match actual demand and adjust them accordingly to maintain optimal throughput without overspending.

- *Use the right deployment type.* Azure OpenAI models allow you to use different deployment types. Global deployment offers lower cost-per-token pricing on

certain OpenAI models.

- *Evaluate hosting options.* Choose the right hosting infrastructure, depending on your solution's needs. For example, for generative AI workloads, options include managed online endpoints, Azure Kubernetes Service (AKS), and Azure App Service, each with its own billing model. Select the option that provides the best balance between performance and cost for your specific requirements.

- *Control client behavior in consumption-based services.* Limit client access to your AI service by enforcing security protocols like network controls, keys, and role-based access control (RBAC). Ensure that clients use API constraints like max tokens and max completions. When possible, batch requests to optimize efficiency. Keep prompts concise, but provide necessary context to reduce token consumption.

- *Consider using a generative AI gateway.* A generative AI gateway allows you to track token usage, throttle token usage, apply circuit breakers, and route to different AI endpoints to control costs.

- *Create a policy to shut down compute instances.* Define and enforce a policy stating that AI resources must use the automatic shutdown feature on virtual machines and compute instances in Azure AI Foundry and Azure Machine Learning. Automatic shutdown is applicable to nonproduction environments and production workloads that you can take offline for certain periods of time.

For more cost management guidance, see Manage AI costs and cost optimization in the Azure OpenAI baseline architecture.

Govern AI platforms

AI platform governance includes applying policy controls to various AI services on Azure, such as Azure AI Foundry and Azure Machine Learning. Using platform-level governance enforces consistent security, compliance, and operational policies across the AI ecosystem. This alignment supports effective oversight, which strengthens overall AI management and reliability.

- *Use built-in governance policies.* Use Azure Policy to apply built-in policy definitions for each AI platform you're using. It includes Azure AI Foundry, Azure Machine Learning, Azure AI services, Azure AI Search, and others.

- *Enable Azure landing zone AI policies.* For Azure landing zone users, the deployment includes a curated set of recommended built-in policies for Azure AI platform services. Select the policy initiative you want to use under the *Workload Specific Compliance* category during an Azure landing zone deployment. The

policies sets include Azure OpenAI, Azure Machine Learning, and Azure AI Search, and Azure Bot services.

Govern AI security

AI security governance addresses the need to protect AI workloads from threats that could compromise data, models, or infrastructure. Robust security practices safeguard these systems against unauthorized access and data breaches. This protection ensures the integrity and reliability of AI solutions, which is essential for maintaining user trust and regulatory compliance.

- *Enable Defender for Cloud on every subscription.* Defender for Cloud provides a cost-effective approach for detecting configurations in your deployed resources that aren't secure. You should also enable AI threat protection.
- *Configure access control.* Grant least privilege user access to centralized AI resources. For example, start with the Reader Azure role, and elevate to the Contributor Azure role if the limited permissions slow down application development.
- *Use managed identities.* Use managed identity on all supported Azure services. Grant least privilege access to application resources that need to access AI model endpoints.
- *Use just-in-time access.* Use privileged identity management (PIM) for just-in-time access.

Govern AI operations

AI operations governance focuses on managing and maintaining stable AI services. These operations support long-term reliability and performance. Centralized oversight and continuity plans help organizations avoid downtime, which ensures the consistent business value of AI. These efforts contribute to efficient AI deployment and sustained operational effectiveness.

- *Review and manage AI models.* Develop a policy for managing model versioning, especially as models are upgraded or retired. You need to maintain compatibility with existing systems and ensure a smooth transition between model versions.
- *Define a business continuity and disaster recovery plan.* Establish a policy for business continuity and disaster recovery for your AI endpoints and AI data. Configure baseline disaster recovery for resources that host your AI model

endpoints. These resources include Azure AI Foundry, Azure Machine Learning, Azure OpenAI, or Azure AI services. All Azure data stores, such as Azure Blob Storage, Azure Cosmos DB, and Azure SQL Database, provide reliability and disaster recovery guidance that you should follow.

- *Define baseline metrics for AI resources.* Enable recommended alert rules to receive notifications of deviations that indicate a decline in workload health. For examples, see Azure AI Search, Azure Machine Learning, Azure AI Foundry prompt flow deployments, and guidance on individual Azure AI services.

Govern AI regulatory compliance

Regulatory compliance in AI requires organizations to follow industry standards and legal obligations, which reduce risks related to liabilities and build trust. Compliance measures help organizations avoid penalties and improve credibility with clients and regulators. Adhering to these standards establishes a solid foundation for responsible and compliant AI usage.

- *Automate compliance.* Use Microsoft Purview Compliance Manager to assess and manage compliance across cloud environments. Use the applicable regulatory compliance initiatives in Azure Policy for your industry. Apply other policies based on the AI services that you use, such as Azure AI Foundry and Azure Machine Learning.

- *Develop industry-specific compliance checklists.* Regulations and standards differ by industry and location. You need to know your regulatory requirements and compile checklists that reflect the regulatory demands that are relevant to your industry. Use standards, such as ISO/IEC 23053:2022 (Framework for Artificial Intelligence Systems Using Machine Learning), to audit policies that are applied to your AI workloads.

Govern AI data

AI data governance involves policies for ensuring that data feeding into AI models is appropriate, compliant, and secure. Data governance protects privacy and intellectual property, which enhances AI outputs' reliability and quality. These measures help mitigate risks related to data misuse, and they align with regulatory and ethical standards.

- *Establish a process for cataloging data.* Use a tool like Microsoft Purview to implement a unified data catalog and classification system across your

organization. Integrate these policies into your CI/CD pipelines for AI development.

- *Maintain data security boundaries.* Cataloging data helps ensure that you don't feed sensitive data into public-facing AI endpoints. When you create indexes from certain data sources, the indexing process can remove the security boundaries around data. Ensure that any data ingested into AI models is classified and vetted according to centralized standards.

- *Prevent copyright infringement.* Use a content filtering system like Protected material detection in Azure AI Content Safety to filter out copyrighted material. If you're grounding, training, or fine-tuning an AI model, ensure that you use legally obtained and properly licensed data and implement safeguards to prevent the model from infringing on copyrights. Regularly review outputs for intellectual property compliance.

- *Implement version control for grounding data.* Establish a version control process for grounding data, for example, in RAG. Versioning ensures that you can track any changes to the underlying data or its structure. You can revert the changes if necessary, helping maintain consistency across deployments.

Next step

Management PaaS AI

Feedback

Was this page helpful? Yes No

Management recommendations for AI workloads on Azure

Article • 12/04/2024

This article offers management recommendations for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

Effective management of AI workloads on Azure involves overseeing deployment, model performance, operations, data, and disaster recovery to support your AI workloads. Proper management helps ensure that AI workloads are reliable, trustworthy, and secure throughout their lifecycle.

Manage AI deployments

Managing AI deployments helps workload teams move from proof-of-concept stages to production environments with consistent configurations that improve security and compliance across teams. Azure offers tools like Azure AI Foundry hubs and projects to enforce governance and security. Azure Machine Learning has similar capabilities with its hub workspaces. For more information, see Manage AI deployments.

Manage AI models

Managing AI models includes monitoring their outputs, performance, and alignment with Responsible AI principles. AI models can drift over time due to changing data, user behaviors, or other external factors. These changes can lead to inaccurate results or ethical concerns if not addressed.

- *Monitor model outputs.* Implement a monitoring and testing process to ensure that these workloads remain aligned with your responsible AI targets.
    - *Monitor generative AI.* For generative AI workloads, use Azure AI Foundry's built-in evaluation and manual monitoring capabilities. If you're using prompt flow, monitor prompt flow deployments. Also consider using responsible AI tools to supplement model monitoring.
    - *Monitor nongenerative AI.* For nongenerative AI workloads, monitor data processing stages and model performance metrics to ensure predictions remain

accurate and reliable. Enable model monitoring in Azure Machine Learning. For Azure AI services, enable monitoring for each AI service you use.

- *Monitor model performance*. When a drop in performance or accuracy is detected, monitoring helps pinpoint the source of the issue. As with all workloads, use Azure Monitor and Application Insights to monitor the performance of AI workloads.
  - *Monitor generative AI performance*. In generative AI, monitor latency in response times or the accuracy of vector search results to enhance user experiences. In Azure AI Foundry, enable tracing to collect trace data for each request, aggregated metrics, and user feedback.
  - *Monitor nongenerative AI performance*. Capture performance metrics of models deployed in Azure Machine Learning. For Azure AI services, enable diagnostic logging for each Azure AI service.
- *Consider a generative AI gateway for monitoring*. A reverse proxy like Azure API Management allows you to implement logging and monitoring that aren't native to the platform. API Management allows you to collect source IPs, input text, and output text. For more information, see Implement logging and monitoring for Azure OpenAI Service language models.

Manage AI operations

AI operations management involves standardizing compute resources and monitoring platform resources for Azure AI workloads. It ensures that teams use the correct compute resources efficiently and capture metrics and logs from platform resources.

- *Monitor platform resources*. Use diagnostic settings to capture logs and metrics for all key services, such as Azure AI Foundry, Azure Machine Learning, and Azure AI services. Specific services should capture audit logs and relevant service-specific logs. Implement custom monitoring alerts based on your architecture's specific needs. Examples include alerts for container registries, Azure Machine Learning, and Azure OpenAI. Configure recommended monitoring alerts for each service in your AI architecture. For more information, see Azure Monitor Baseline Alerts.
- *Standardize compute management*. You need compute resources for certain actions like prompt flows and training models. A service like Machine Learning has different compute options, such as compute instances, clusters, and serverless options. Standardize the compute type, runtimes, and shutdown periods. For service-specific compute options, see Azure AI Foundry and Machine Learning.

Manage AI data

High-quality data is the foundation of accurate AI models. Tracking model drift helps maintain the relevance of AI predictions over time, and it allows organizations to adapt models as necessary to reflect current conditions.

- *Monitor data drift.* Track accuracy and data drift continuously in generative and nongenerative AI to ensure that models remain relevant. Monitoring can alert you when model predictions or large language model responses deviate from expected behavior. This deviation indicates a need for retraining or adjustment. Set up custom alerts to detect performance thresholds. This approach enables early intervention when problems arise. Use evaluations in Azure AI Foundry and metrics supported in Machine Learning.
- *Ensure quality data processing.* For machine learning, training data must be formatted, clean, and ready for model consumption. For generative AI, grounding data needs to be in the correct format, and likely chunked, enriched, and embedded for AI model consumption. For more information, see Guide to designing and developing a RAG solution.

Manage business continuity

Implement multi-region deployments to ensure high availability and resiliency for both generative and nongenerative AI systems For more information, see multi-region deployment in Azure AI Foundry, Azure Machine Learning, and Azure OpenAI.

Next step

Security PaaS AI

Feedback

Was this page helpful? Yes No

Security recommendations for AI workloads on Azure

Article • 11/01/2024

This article offers security recommendations for organizations running AI workloads on Azure. It focuses on Azure AI platform-as-a-service (PaaS) solutions, including Azure AI Foundry, Azure OpenAI, Azure Machine Learning, and Azure AI Services. It covers both generative and nongenerative AI workloads.

As AI becomes more integrated into business operations, protecting these resources from potential threats is crucial to maintain data integrity and compliance. Applying standardized security baselines and following well-architected frameworks helps organizations safeguard their AI infrastructure against vulnerabilities.

Secure AI resources

Securing AI resources means applying security baselines and best practices to protect the infrastructure used for AI workloads on Azure. This protection minimizes risks from external threats and ensures a consistent security posture across the organization.

Secure Azure AI platforms. Standardize the application of Azure security baselines for every AI resource. Follow the security recommendations in Azure Service Guides within the Azure Well-Architected Framework.

Expand table

Azure AI platform security baseline	Azure Well-Architected Framework service guide
Azure Machine Learning	Azure Machine Learning
Azure AI Foundry	
Azure OpenAI	Azure OpenAI

Secure the AI models

Securing AI models refers to implementing threat protection, monitoring for prompt injection risks, verifying model integrity, and centralizing governance. These practices ensure that AI models remain safe from malicious manipulation, maintain their reliability, and provide accurate results.

- *Implement threat protection for all AI models.* Use Microsoft Defender for Cloud to protect AI models from threats like prompt injection attacks and model manipulation. This tool provides continuous monitoring of AI workloads, helping to detect and prevent emerging threats. Implementing this protection across all workloads ensures a consistent security posture throughout the organization.

- *Monitor outputs and apply prompt shielding.* Regularly inspect the data returned by AI models to detect and mitigate risks associated with malicious or unpredictable user prompts. Implement Prompt Shields to scan text for the risk of a user input attack on generative AI models.

- *Ensure model verification.* Establish company-wide verification mechanisms to ensure all AI models in use are legitimate and secure. If you use open-source models, use model signatures or other verification processes to confirm the authenticity of AI models, preventing unauthorized or tampered models from being deployed.

- *Consider using an AI Gateway.* Azure API Management (APIM) can help ensure consistent security across AI workloads. Use its built-in policies for traffic control and security enforcement. Integrate APIM with Microsoft Entra ID to centralize authentication and authorization and ensure only authorized users or applications interact with your AI models. Ensure you configure least privilege access on the reverse proxy's managed identity. For more information, see AI authentication with APIM

Secure AI access

Securing AI access includes establishing authentication and authorization controls for both management plane and external access to AI resources. Proper access management restricts resource usage to only users with verified permissions. It reduces the chances of unauthorized interactions with AI models. Strong access controls, such as role-based access and conditional access policies, helps protect sensitive data and maintain compliance with security standards.

- *Organize resources and access controls.* Use distinct workspaces to organize and manage AI artifacts like datasets, models, and experiments. Workspaces centralize resource management and simplify access control. For example, use projects within Azure AI Foundry to manage resources and permissions efficiently, facilitating collaboration while maintaining security boundaries.

- *Use Microsoft Entra ID for authentication.* Wherever possible, eliminate static API keys in favor of Microsoft Entra ID for authentication. This step enhances security

through centralized identity management and reduces secret management overhead. Also limit the distribution of API keys. Instead, prefer identities in Microsoft Entra ID over API keys for authentication. Audit the list of individuals with API key access to ensure it's current. For authentication guidance, see Azure AI Foundry, Azure OpenAI, Azure AI services, Azure Machine Learning.

- *Configure authentication.* Enable multifactor authentication (MFA) and prefer secondary administrative accounts or just-in-time access with Privileged Identity Management (PIM) for sensitive accounts. Limit control plane access using services like Azure Bastion as secure entry points into private networks.

- *Use Conditional Access Policies.* Implement risk-based conditional access policies that respond to unusual sign-in activity or suspicious behavior. Use signals like user location, device state, and sign-in behavior to trigger extra verification steps. Require MFA for accessing critical AI resources to enhance security. Restrict access to AI infrastructure based on geographic locations or trusted IP ranges. Ensure that only compliant devices (those meeting security requirements) can access AI resources.

- *Configure least privilege access.* Configure least privilege access by implementing role-based access control (RBAC) to provide minimal access to data and services. Assign roles to users and groups based on their responsibilities. Use Azure RBAC to fine-tune access control for specific resources such as virtual machines and storage accounts. Ensure users have only the minimum level of access necessary to perform their tasks. Regularly review and adjust permissions to prevent privilege creep. For example,

Expand table

Role	Example permissions
Data scientists	Read/write access to data storage, permission to run training jobs, and access to model training environments.
AI developers	Access to development environments, deployment permissions, and the ability to modify AI applications.
IT administrators	Full access to manage infrastructure, network configurations, and security policies.

- *Secure Azure service-to-service interactions.* Use managed identity to allow Azure services to authenticate to each other without managing credentials.

- *Secure external access to AI model endpoints.* Require clients to authenticate using Microsoft Entra ID when accessing AI model endpoints. Consider using Azure API Management as an AI gateway in front of AI model endpoints to enforce access policies, control usage, and provide monitoring capabilities.

Secure AI execution

Securing AI execution involves safeguarding the processes by which AI agents, such as virtual assistants or autonomous agents, run code in response to user requests. Isolate the execution environments, conduct code reviews, and set resource limits. These measures help ensure that these executions don't compromise system stability or security. These practices prevent malicious activities and protect the integrity of the systems in which AI agents operate, allowing them to function reliably within a secure framework.

- *Implement isolation mechanisms.* Utilize dynamic session management, such as Dynamic Sessions in Azure Container Apps, to ensure each code execution occurs in a fresh, isolated environment that is destroyed after use.
- *Secure execution code.* Conduct thorough code reviews and testing before deploying scripts for execution by AI agents. This process helps identify and mitigate potential vulnerabilities. Use version control systems to manage code changes and ensure that only approved versions of scripts are executed.
- *Implement resource limits.* Set resource limits (CPU, memory, disk usage) for code execution environments to prevent any single execution from consuming excessive resources and potentially disrupting other services. Define execution timeouts to ensure that long-running or potentially stuck processes are terminated automatically.

For more information, see How to create Assistants with Azure OpenAI Service , How to use Azure OpenAI Assistants function calling , and Agent implementation.

Next step

Govern PaaS AI

Feedback

Was this page helpful? Yes No

Implementation option for AI on Azure infrastructure

Article • 11/01/2024

This article provides implementation recommendations for organizations running AI workloads on Azure infrastructure (IaaS). After deploying an Azure landing zone, you can set up the application landing zone using the CycleCloud Workspace for Slurm. Azure CycleCloud Workspace for Slurm offers several benefits for users who want to run AI workloads with the Slurm scheduler.

- *Easy and fast cluster creation.* Users can quickly create Slurm clusters on Azure through a simple GUI. They can choose from various Azure virtual machine (VM) sizes and types and customize cluster settings such as node count, network configuration, storage options (like Azure NetApp Files and Azure Managed Lustre Filesystem), and Slurm parameters.
- *Flexible and dynamic cluster management.* Azure CycleCloud scales Slurm clusters up or down automatically. Users can monitor cluster status, performance, and utilization, and view logs and metrics through the GUI. They can delete clusters when not needed and only pay for the resources they use.
- *Full control of the infrastructure.* Users have full control over the deployed infrastructure, allowing them to bring their own code, libraries, and packages, and to use resources on demand.

Design guidelines

The following articles provide guidelines for AI workloads on Azure infrastructure (IaaS):

- Compute
- Storage
- Networking
- Governance
- Management
- Security

Architecture

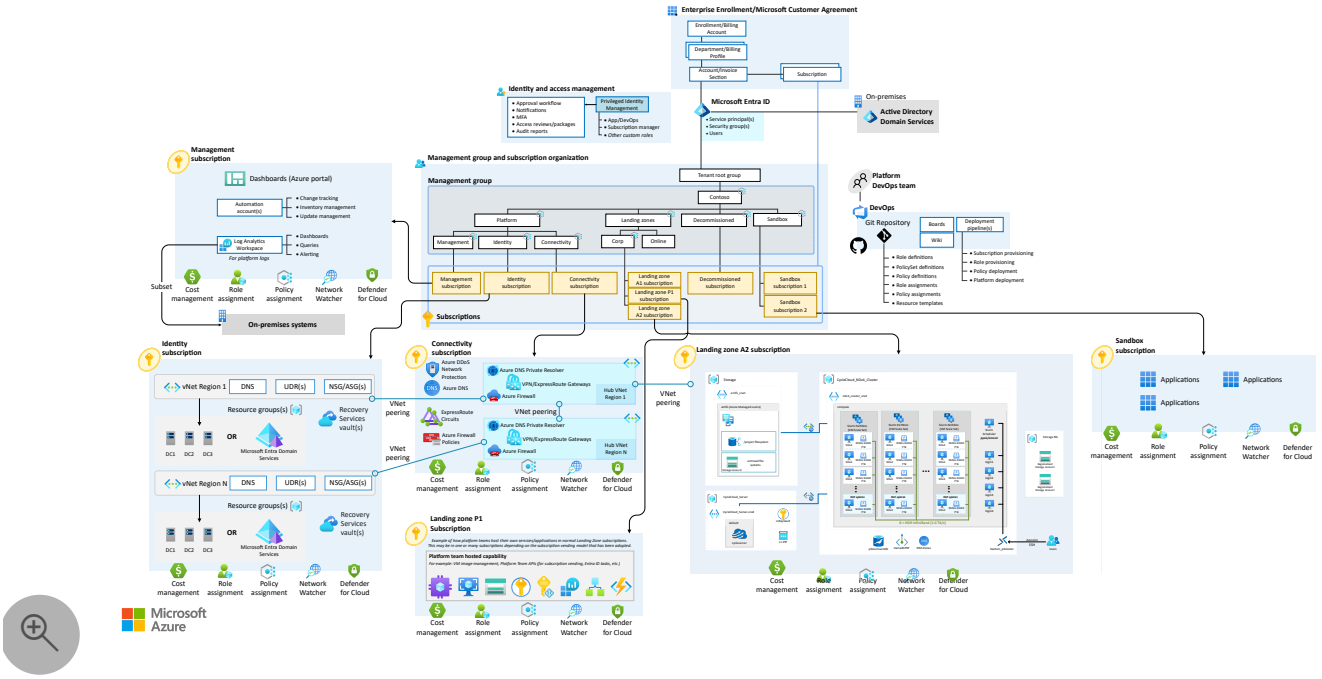


Figure 1. AI application on Azure infrastructure in Azure landing zone.

Deploy CycleCloud Workspace for Slurm

The CycleCloud Workspace for Slurm can be used as the initial deployment in the enterprise environment. You can develop and customize the code to expand its functionality and/or adapt it to your Azure landing zone environment. Then, follow the guidance to fine-tune a diffusion model from Hugging Face using Azure CycleCloud Workspace for Slurm.

Next step

Compute IaaS AI

Feedback

Was this page helpful? Yes No

Compute recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 10/25/2024

This article provides compute recommendations for organizations running AI workloads on Azure infrastructure (IaaS). The preferred approach is to start your AI adoption with Azure AI platform-as-a-service (PaaS) solutions. However, if you have access to Azure GPUs, follow this guidance to run AI workloads on Azure IaaS.

AI workloads require specialized virtual machines (VMs) to handle high computational demands and large-scale data processing. Choosing the right VMs optimizes resource use and accelerates AI model development and deployment. The following table provides an overview of recommended compute options.

Expand table

AI phase	Virtual Machine Image	Generative AI	Nongenerative AI (complex models)	Nongenerative AI (small models)
Training AI models	Data Science Virtual Machines	GPU (prefer ND-family. Alternatively use NC family with ethernet-interconnected VMs)	GPU (prefer ND-family. Alternatively use NC family with ethernet-interconnected VMs)	Memory-optimized (CPU)
Inferencing AI models	Data Science Virtual Machines	GPU (NC or ND family)	GPU (NC or ND family)	Compute-optimized (CPU)

Pick the right virtual machine image

Choose a suitable virtual machine image, such as the Data Science Virtual Machines, to access preconfigured tools for AI workloads quickly. This choice saves time and resources while providing the software necessary for efficient AI processing

- *Start with the Data Science Virtual Machines images.* The Data Science Virtual Machine image offers preconfigured access to data science tools. These tools include PyTorch, TensorFlow, scikit-learn, Jupyter, Visual Studio Code, Azure CLI, and PySpark. When used with GPUs, the image also includes Nvidia drivers, CUDA Toolkit, and cuDNN. These images serve as your baseline image. If you need more

software, add it via a script at boot time or embed into a custom image. They maintain compatibility with your orchestration solutions.

- *Find alternative images as needed.* If the Data Science Virtual Machine image doesn't meet your needs, use the Azure Marketplace or other search methods to find alternate images. For example, with GPUs, you might need Linux images that include InfiniBand drivers, NVIDIA drivers, communication libraries, MPI libraries, and monitoring tools.

Pick a virtual machine size

Selecting an appropriate virtual machine size aligns with your AI model complexity, data size, and cost constraints. Matching hardware to training or inferencing needs maximizes efficiency and prevents underutilization or overload.

- *Narrow your virtual machine options.* Choose the latest virtual machine SKUs for optimal training and inference times. For training, select SKUs that support RDMA and GPU interconnects for high-speed data transfer between GPUs. For inference, avoid SKUs with InfiniBand, which is unnecessary. Examples include the ND MI300X v5 series, ND H100 v5 series, NDm A100 v4-series, and ND A100 v4-series.

- *Check virtual machine pricing.* Use the Linux and Windows VM pricing pages for a general cost overview. For a detailed estimate, use the Azure Pricing Calculator.

- *Consider spot instances.* Spot instances are cost-effective for inference scenarios with minimal data loss risk. Spot instances offer significant savings by utilizing unused datacenter capacity at a discount. However, this capacity can be reclaimed at any time, so spot instances are best for workloads that can handle interruptions. Regularly checkpoint data to minimize loss when evicted. For information, see Using Spot VMs in Azure CycleCloud.

Choose a compute orchestration solution

Compute orchestration solutions facilitate the management of AI tasks across virtual machine clusters. Even for simple deployments, an orchestrator can help reduce costs and ensure an environment is reproducible. Orchestrators help ensure you only use the compute that you need for a specific amount of time. Select an orchestration tool based on your scheduling, containerization, and scaling needs to improve operations and scalability.

- *Use Azure CycleCloud for open-source schedulers.* Azure CycleCloud is ideal for open-source schedulers like Slurm, Grid Engine, or Torque/PBS. It provides flexible cluster management, customizable configurations, and advanced scheduling capabilities. Virtual machines within the cluster need configuration for AI workload execution. Virtual machines for CycleCloud and Batch are non-persistent. The orchestrator creates and removes VMs when needed to help with cost savings. For more information, see Azure CycleCloud Workspace for Slum.

- *Use Azure Batch for built-in scheduling.* Azure Batch offers built-in scheduling features with no need for extra software installation or management. It has a consumption pricing model and no licensing fees. It also supports containerized tasks natively. For deployment best practices, see Azure Batch Accelerator.

- *Use Azure Kubernetes Service (AKS) for container scaling.* AKS is a managed service for deploying, scaling, and managing containers across a cluster. It's suitable for running AI workloads in containers at scale. For more information, see Use Azure Kubernetes Service to host GPU-based workloads.

- *Manually orchestrate jobs for simpler tasks.* If orchestration needs are minimal, manage AI resources manually. Consider the following steps for small-scale workloads:
  - *Define your workflow.* Understand your workflow end-to-end, including dependencies and job sequence. Consider how to handle failures at any step.
  - *Log and monitor jobs.* Implement clear logging and monitoring frameworks for your jobs.
  - *Validate prerequisites.* Ensure your environment meets all workflow requirements, including necessary libraries and frameworks.
  - *Use version control.* Track and manage changes using version control.
  - *Automate tasks.* Use scripts to automate data preprocessing, training, and evaluation.

Consider containers

Containers provide a consistent, reproducible environment that scales efficiently. Containers streamline transitions between environments, making them essential for scalable AI solutions.

- *Install drivers.* Ensure the necessary drivers are installed to enable container functionality in various scenarios. For cluster configurations, tools like Pyxis and Enroot are often required.

- *Use NVIDIA Container Toolkit*. This toolkit enables GPU resources within containers. Install all required drivers, such as CUDA and GPU drivers, and use your preferred container runtime and engine for AI workload execution.

Next step

Storage IaaS AI

Feedback

Was this page helpful? Yes No

Storage recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 11/01/2024

This article provides storage recommendations for organizations running AI workloads on Azure infrastructure (IaaS). A storage solution for AI workloads on Azure infrastructure must be capable of managing the demands of data storage, access, and transfer that are inherent to AI model training and inferencing.

AI workloads require high throughput and low latency for efficient data retrieval and processing. They also need mechanisms for data versioning and consistency to guarantee accurate and reproducible outcomes across distributed environments. When selecting the appropriate storage solution, consider factors such as data transfer times, latency, performance requirements, and compatibility with existing systems.

- *Use a file system for active data*. Implement a file system to store "job-specific/hot" data actively used or generated by AI jobs. This solution is ideal for real-time data processing due to its low latency and high throughput capabilities. These capabilities are critical for optimizing the performance of AI workflows. Azure has three principal file system solutions to support training and inferencing AI models on Azure infrastructure. To choose the right file system, follow these recommendations:

    - *Use Azure Managed Lustre for lowest data transfer times and minimized latency.* Azure Managed Lustre provides high performance with parallel file system capabilities and simplifies management with Azure integration. It's cost-effective, with usage-based storage costs, and allows selective data import from Blob Storage, optimizing data handling.

    - *Use Azure NetApp Files when you need enterprise-grade features and performance for AI workloads.* Azure NetApp Files offer high reliability and performance, ideal for mission-critical applications. Azure NetApp Files is beneficial if you have existing investments in NetApp infrastructure. It's beneficial for hybrid cloud capabilities and when you need to customize and fine-tune storage configurations.

    - *Use local NVMe/SSD file systems when performance is the top priority.* It aggregates the local NVMe of compute (worker nodes) using a job-dedicated parallel file system like BeeGFS On Demand (BeeOND). They operate directly on the compute nodes to create a temporary, high-performance file system during the job. These systems offer ultra-low latency and high throughput, making

them ideal for I/O-intensive applications like deep learning training or real-time inferencing.

- *Transfer inactive data to Azure Blob Storage.* After completing a job, transfer inactive job data from Azure Managed Lustre to Azure Blob Storage for long-term, cost-effective storage. Blob storage provides scalable options with different access tiers, ensuring efficient storage of inactive or infrequently accessed data, while keeping it readily available when needed.
- *Implement checkpointing for model training.* Set up a checkpointing mechanism that saves the model's state, including training weights and parameters, at regular intervals such as every 500 iterations. Store this checkpoint data in Azure Managed Lustre to allow restarting the model training from a previously saved state, improving the flexibility and resilience of your AI workflows.
- *Automate data migration to lower-cost storage tiers.* Configure Azure Blob Storage lifecycle management policies to automatically migrate older, infrequently accessed data to lower-cost storage tiers, such as the Cool or Archive tiers. This approach optimizes storage costs while ensuring that important data remains accessible when needed.
- *Ensure data consistency across distributed environments.* Ensure data consistency across distributed AI workloads by setting up synchronization between Azure Managed Lustre and Azure Blob Storage. This synchronization ensures that all nodes accessing the data are working with the same, consistent version, preventing errors and discrepancies in distributed environments.
- *Enable data versioning for reproducibility.* Activate versioning in Azure Blob Storage to track changes to datasets and models over time. This feature facilitates rollback, enhances reproducibility, and supports collaboration. It maintains a detailed history of modifications to data and models and allows you to compare and restore previous versions as needed.

Next step

Networking IaaS AI

Feedback

Networking recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 10/25/2024

This article provides networking recommendations for organizations running AI workloads on Azure infrastructure (IaaS). Designing a well-optimized network can enhance data processing speed, reduce latency, and ensure the network infrastructure scales alongside growing AI demands.

Ensure sufficient bandwidth

Sufficient bandwidth refers to the capacity of a network to handle large volumes of data without delays or interruptions. High bandwidth ensures fast, uninterrupted data transfer between on-premises systems and Azure, supporting rapid AI model training and reducing downtime in the pipeline. For organizations transferring large datasets from on-premises to the cloud for AI model training, a high-bandwidth connection is essential. Use Azure ExpressRoute to establish a dedicated, secure, and reliable high-speed connection between your on-premises network and Azure.

Minimize latency

Minimizing latency involves reducing delays in data transfer between networked resources. Lower latency provides quicker data processing, enabling real-time insights, and improving the performance of latency-sensitive workloads.

- *Optimize resource placement.* To minimize latency for AI workloads, such as data preprocessing, model training, and inference, deploy virtual machines (VMs) within the same Azure region or availability zone. Colocating resources reduces physical distance, thus improving network performance.

- *Use proximity placement groups (PPGs).* For latency-sensitive workloads requiring real-time processing or fast inter-process communication, utilize PPGs to physically colocate resources within an Azure datacenter. PPGs ensure that compute, storage, and networking resources remain close together, minimizing latency for demanding workloads. Orchestration solutions and InfiniBand handle node proximity automatically.

- *Use preconfigured Linux OS images.* Simplify cluster deployment by selecting Linux OS images from the Azure Marketplace prepackaged with InfiniBand drivers,

NVIDIA drivers, communication libraries, and monitoring tools. These images are optimized for performance and can be deployed with Azure CycleCloud for fast, efficient cluster creation.

Implement high-performance networking

High-performance networking utilizes advanced networking features to support large-scale, intensive AI computations, particularly for GPU-accelerated tasks. High-performance networks ensure rapid, efficient data exchanges between GPUs, which optimizes model training and accelerates AI development cycles.

- *Utilize InfiniBand for GPU workloads.* For workloads dependent on GPU acceleration and distributed training across multiple GPUs, use Azure's InfiniBand network. InfiniBand's GPUDirect remote direct memory access (RDMA) capability supports direct GPU-to-GPU communication. It improves data transfer speed and model training efficiency. Orchestration solutions like Azure CycleCloud and Azure Batch handle InfiniBand network configuration when you use the appropriate VM SKUs.

- *Choose Azure's GPU-optimized VMs.* Select VMs that use InfiniBand, such as ND-series VMs, which are designed for high-bandwidth, low-latency inter-GPU communication. This configuration is essential for scalable distributed training and inference, allowing faster data exchange between GPUs.

Optimize for large-scale data processing

Optimizing for large-scale data processing involves strategies to manage extensive data transfers and high computational loads. By using data and model parallelism, you can scale your AI workloads and enhance processing speed. Use Azure's GPU-optimized virtual machines to handle complex, data-intensive AI workloads.

- *Apply data or model parallelism techniques.* To manage extensive data transfers across multiple GPUs, implement data parallelism or model parallelism depending on your AI workload needs. Ensure the use of High Bandwidth Memory (HBM), which is ideal for high-performance workloads due to its high bandwidth, low power consumption, and compact design. HBM supports fast data processing, essential for AI workloads that require processing large datasets.

- *Use advanced GPU networking features.* For demanding AI scenarios, choose Azure VMs like NDH100v5 and NDMI300Xv5. Azure configures these VMs with dedicated 400 Gb/s NVIDIA Quantum-2 CX7 InfiniBand connections within virtual machine

scale sets. These connections support GPU Direct RDMA, enabling direct GPU-to-GPU data transfers that reduce latency and enhance overall system performance.

Next step

Security IaaS AI

Feedback

Was this page helpful? Yes No

Governance recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 11/01/2024

This article provides governance recommendations for organizations running AI workloads on Azure infrastructure (IaaS). These recommendations help organizations establish a structured framework for resource management, cost control, security, and operational efficiency. By following these practices, you can scale your AI workloads responsibly and ensure they meet compliance, security, and financial goals.

Resource governance

Resource governance establishes rules and standards for managing Azure resources. By enforcing governance policies, organizations can ensure compliance, standardize resource use, and control costs, which support the responsible scaling of AI operations.

- *Enforce tag usage.* Use Azure Policy to enforce rules like resource location, allowed SKUs, and mandatory tags. For example, create policies to restrict the deployment of certain high-cost VMs, helping to manage budgets effectively.
- *Apply governance policies to ensure compliance and standardization.* Use Azure Policy to enforce rules such as resource location, allowed SKUs, and mandatory tags. For example, create policies to restrict the deployment of certain high-cost VMs to control the budget.
- *Use resource groups for lifecycle management.* Deploy AI resources within resource groups that share a common lifecycle. Resource groups allow you to deploy, configure, and delete resources collectively. They also provide extra governance (policy), security (RBAC), and cost (budget) boundaries.
- *Standardize naming conventions.* Implement a standardized naming convention for AI resources. This practice improves tracking and management. Use the naming rules and restrictions for each Azure resource and follow the recommended abbreviations, as many resources often have name-length restrictions.
- *Govern infrastructure as code.* Use Microsoft Defender for Cloud to monitor and enforce IaC security. This tool helps detect IaC misconfigurations and ensures secure deployments.

Cost management

Cost management monitors and controls expenses related to AI workloads on Azure. Effective cost management enables organizations to set budgets, track spending, and maintain financial sustainability for AI projects.

- *Use tags to allocate costs*. Configure an Azure Policy definition to enforce tagging on resources. Use tags to categorize resources by project, cost center, environment, and owner for better management and billing.
- *Use tag inheritance*. Use tag inheritance in Cost Management to apply billing, resource group, and subscription tags to child resource usage records.
- *Manage billing accounts*. Use Microsoft Billing to oversee billing accounts and handle invoices. Assign a billing account to each AI project or team to facilitate accurate expense tracking.
- *Monitor costs*. Use Microsoft Cost Management to set budget alerts, cost anomaly alerts, and scheduled alerts. Monitoring costs in this way helps organizations maintain financial discipline.
- *View spending patterns*. Use the Azure Cost analysis to tool to regularly review spending patterns. This process identifies trends and reveals areas for potential savings, especially in VM usage.
- *Allow specific virtual machine SKUs*. Use Azure policy to allow only the virtual machines SKUs that align with your AI budget. The built-in policy definition *Allowed virtual machine SKUs* can enforce this control.
- *Consider autoscaling*. Use a virtual machine scale set to dynamically adjust VM counts based on demand, optimizing costs.
- *Configure VM autoshutdown*. Use the autoshutdown feature to schedule VMs to shut down during off-hours, reducing unnecessary costs.

Security governance

Security governance addresses the need for robust protection measures across AI workloads. By implementing security policies and access controls, organizations can protect sensitive data and resources. It reduces risk and supports a secure AI environment on Azure.

- *Integrate with Microsoft Entra ID*. Use Microsoft Entra ID for centralized identity management and single sign-on (SSO) capabilities across AI workloads.

- *Implement distinct access controls for each environment.* Limit each deployment pipeline's identity to its designated environment, reducing the risk of accidental deployments.

- *Enable Azure Defender.* Activate Azure Defender for advanced threat protection. Azure Defender enhances security for workloads, including virtual machines, storage accounts, and databases, promoting a robust security posture for AI workloads.

Operational governance

Operational governance ensures consistent monitoring and management of AI workloads. By using tools for monitoring, alerting, and automated deployments, organizations can maintain system health, detect issues early, and improve operational efficiency, contributing to reliable and stable AI operations.

- *Deploy monitoring agents.* Ensure that Azure Monitor agents are deployed by default for virtual machines, Azure Virtual Machine Scale Sets, and Azure Arc connected servers. Connect them to a central Log Analytics workspace within the management subscription.

- *Configure alerts.* Enable recommended alert rules to receive notifications of metric deviations.

- *Use a CI/CD pipeline.* Implement continuous integration and continuous delivery (CI/CD) to automate code testing and deployment to different environments.

Next step

Management IaaS AI

Feedback

Was this page helpful? Yes No

Management recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 11/01/2024

This article provides management recommendations for organizations running AI workloads on Azure infrastructure (IaaS). Effective management of AI workloads on Azure requires continuous monitoring, optimization practices, and a strong backup and recovery strategy. These efforts minimize downtime and ensure reliability in AI operations.

Monitor AI infrastructure

Monitoring AI infrastructure involves tracking and evaluating the performance, health, and availability of all components in an AI deployment on Azure IaaS. Proactive monitoring allows organizations to detect and resolve potential issues before they affect operations.

- *Ensure monitoring by default.* Deploy the required Azure Monitor agents for virtual machines and Azure Virtual Machine Scale Sets, including Azure Arc connected servers. Connect them to the central Log Analytics workspace in the management subscription. Consider using Azure Monitor Baseline Alerts (AMBA).

- *Use Azure Update Manager.* You can monitor Windows and Linux update compliance across your machines in Azure and on-premises/on other cloud platforms (connected by Azure Arc) from a single pane of management. You can also use Update Manager to make real-time updates or schedule them within a defined maintenance window.

- *Monitor virtual machines.* Monitor virtual machine (VM) host data (physical host) and VM guest data (operating system and application). Consider using VM Insights to simplify the onboarding, access predefined performance charts, and utilize dependency mapping. Track Spot VM evictions and maintenance events to manage interruptions effectively. Learn more about scheduled events.

- *Monitor networks.* Monitor and diagnose networking issues without logging into your VMs. Get real-time performance information at the packet level. Troubleshoot performance issues with the Performance Diagnostics tool. Track topology, health, and metrics for all deployed network resources.

- *Monitor storage*. Monitor the performance of storage, such as local SSDs, attached disks, file shares, and Azure storage accounts.

- *Use orchestrator monitoring capabilities (if applicable)*. Consider using the built-in monitoring capabilities of orchestrators like Azure CycleCloud, Azure Batch, and Azure Kubernetes Service (AKS). Follow the guidance for the orchestrator you chose:

  - *Azure CycleCloud or Azure CycleCloud Workspace for Slurm:* Track CPU, disk, and network metrics. Store data from Azure CycleCloud clusters to Log Analytics and create custom metrics dashboards. For more information, see Monitoring Azure CycleCloud. Node Health Checks are a set of automated tests to ensure that your HPC/AI hardware is healthy. You can run this check in Azure CycleCloud as part of cluster deployment or separately using the GitHub repo instructions. Ensure that you pay attention to the compatibility matrix in the documentation. Run where appropriate to ensure that you identify any unhealthy nodes before running your AI workloads.

  - *Azure Batch:* Collect job and task metrics such as active tasks, task duration, job start time, duration, task start time. Also collect pool metrics, such as idle nodes, running nodes, CPU usage, Disk I/O. For more information, see Azure Batch monitoring.

  - *Azure Kubernetes Service*. Use Azure Monitor for containers. Monitor pod performance, node health, and resource utilization. Set up alerts and custom dashboards.

Manage business continuity and disaster recovery

Managing business continuity and disaster recovery for AI applications on Azure ensures that organizations can recover quickly from disruptions. By implementing strategies such as real-time replication, automated recovery, and regular backups, organizations safeguard their AI infrastructure against data loss and operational downtime.

- *Use Azure Site Recovery.* Site Recovery uses real-time replication and recovery automation to replicate workloads across regions. Built-in platform capabilities for VM workloads meet low RPO and RTO requirements. You can use Site Recovery to run recovery drills without affecting production workloads. You can also use Azure Policy to enable replication and to audit VM protection.

- *Use orchestrator capabilities (if applicable).* Use your orchestrator to recover failed compute nodes. For example, configure Azure Batch to automatically retry tasks if there's failure.
- *Schedule backups.* Determine if you need to backup incremental changes to datasets and models daily or weekly. Backups could also include databases or entire datasets.
- *Ensure data compliance.* Make sure your backup strategy complies with data protection regulations. Comply with data residency requirements and store backups in appropriate geographic locations.
- *Create snapshots.* You can use the capabilities of your scheduler to take snapshots. For example, CycleCloud can take point-in-time snapshots of the underlying application datastore as recovery points.

Next step

Secure IaaS AI

Feedback

Was this page helpful? Yes No

Security recommendations for AI workloads on Azure infrastructure (IaaS)

Article • 01/17/2025

This article provides security recommendations for organizations running AI workloads on Azure infrastructure (IaaS). Security for AI on Azure infrastructure involves protecting data, compute, and networking resources that support AI workloads. Securing these components ensures that sensitive information remains safe, minimizes exposure to potential threats, and ensures a stable operational environment for AI models and applications.

Secure Azure services

Azure service security requires configuring each Azure service used in an AI architecture to meet specific security standards and benchmarks.

- *Harden Azure services.* To apply secure configurations to Azure services, use the Azure security baselines for each service in your architecture. Common Azure services in AI workloads on Azure infrastructure include: Windows virtual machines, Linux virtual machines, Azure CycleCloud, and Key Vault.

- *Consider secure compute options.* Secure the boot process and integrity of your VMs using trusted launch. Depending on your industry and use case, consider using confidential AI. Confidential AI is for cryptographically verifiable protection for AI data and models during training, fine-tuning, and inferencing.

Secure networks

Securing networks involves setting up private endpoints, Network Security Groups (NSGs), and firewalls to manage and control data flow within Azure. This step limits exposure to external threats and protects sensitive data as it moves between services within the Azure infrastructure.

- *Use private endpoints.* Use private endpoints available in Azure Private Link for any PaaS solution in your architecture, such as your storage or filesystem.

- *Use encrypted virtual network connections for Azure to Azure connectivity.* Encrypted connections between virtual machines or virtual machine scale sets in the same or peered virtual networks prevent unauthorized access and eavesdropping. Establish

these secure connections by configuring encryption options in Azure Virtual Network for virtual machine communication.

- *Implement Network Security Groups (NSGs).* NSGs can be complex. Ensure you have a clear understanding of the NSG rules and their implications when setting up your Azure infrastructure for AI workloads.
- *Use Application Security Groups*. If you need to label traffic at a greater granularity than what virtual networks provide, consider using Application Security Groups.
- *Understand NSG prioritization rules*. NSG rules have a priority order. Understand this order to avoid conflicts and ensure the smooth running of your AI workloads.
- *Use a network firewall.* If you're using a hub-spoke topology, deploy a network firewall to inspect and filter network traffic between the spokes.
- *Close unused ports.* Limit internet exposure by exposing only services intended for external-facing use cases and using private connectivity for other services.

Secure data

Securing data includes encrypting data at rest and in transit, along with protecting sensitive information such as keys and passwords. These measures ensure that data remains private and inaccessible to unauthorized users, reducing the risk of data breaches and unauthorized access to sensitive information.

- *Encrypt data*: Encrypt data at rest and in transit using strong encryption technologies between each service in the architecture.
- *Protect secrets*: Protect secrets by storing them in a key vault or a hardware security module and routinely rotate them.

Secure access

Securing access means configuring authentication and access control mechanisms to enforce strict access permissions and verify user identities. By restricting access based on roles, policies, and multifactor authentication, organizations can limit exposure to unauthorized access and protect critical AI resources.

- *Configure authentication*: Enable multifactor authentication (MFA) and prefer secondary administrative accounts or just-in-time access for sensitive accounts. Limit control plane access using services like Azure Bastion as secure entry points into private networks.

- *Use Conditional Access Policies.* Require MFA for accessing critical AI resources to enhance security. Restrict access to AI infrastructure based on geographic locations or trusted IP ranges. Ensure that only compliant devices (those meeting security requirements) can access AI resources. Implement risk-based conditional access policies that respond to unusual sign-in activity or suspicious behavior. Use signals like user location, device state, and sign-in behavior to trigger extra verification steps.

- *Configure least privilege access.* Configure least privilege access by implementing role-based access control (RBAC) to provide minimal access to data and services. Assign roles to users and groups based on their responsibilities. Use Azure RBAC to fine-tune access control for specific resources such as virtual machines and storage accounts. Ensure users have only the minimum level of access necessary to perform their tasks. Regularly review and adjust permissions to prevent privilege creep.

Prepare for incident response

Preparing for incident response involves collecting logs and integrating them with a Security Information and Event Management (SIEM) system. This proactive approach enables organizations to detect and respond to security incidents quickly, reducing potential damage, and minimizing downtime for AI systems.

Secure operating systems

Securing operating systems requires keeping virtual machines and container images up-to-date with the latest patches and running antimalware software. These practices protect AI infrastructure from vulnerabilities, malware, and other security threats. They help maintain a secure and reliable environment for AI operations.

- *Patch virtual machine guests.* Regularly apply patches to virtual machines and container images. Consider enabling automatic guest patching for your virtual machines and scale sets.

- *Use antimalware.* Use Microsoft Antimalware for Azure on your virtual machines to protect them from malicious files, adware, and other threats.

Next step

Govern AI

Feedback

Was this page helpful? Yes No

Well-architected considerations for AI workloads on Azure infrastructure (IaaS)

Article • 03/11/2025

Well-architected considerations for AI on Azure infrastructure involve best practices that optimize the reliability, security, operational efficiency, cost management, and performance of AI solutions. These principles ensure robust deployment, secure data handling, efficient model operation, and scalable infrastructure on Azure's IaaS platform. Applying these principles allows organizations to build resilient, secure, and cost-effective AI models that meet business needs.

Reliability

Reliability involves minimizing downtime and ensuring consistent performance for AI applications on Azure infrastructure. Ensuring reliable operations across distributed virtual machines (VMs) and maintaining performance during infrastructure changes prevents service interruptions. These practices are important because they guarantee continuous model availability and improve user experience.

- *Distribute VMs across Availability Zones.* Minimize downtime from hardware failures or maintenance events by using Availability Zones. They distribute VMs across fault and update domains to ensure continued application operation.
- *Set up health monitoring with Azure Monitor.* Track CPU, memory, and network performance on your VMs using Azure Monitor and configure alerts to notify you of performance degradation or failures in the infrastructure supporting your models. For more information, see Azure Monitor VM Insights.
- *Automate patching and updates with rolling instances.* Use Azure Update Management to apply patches in a rolling manner, allowing one instance to be updated while others continue to serve traffic, preventing downtime during maintenance.
- *Design for graceful degradation during partial failures.* Ensure core functionality remains available by serving less complex AI models or limiting specific features when some VMs become unavailable, allowing users access to essential services even during outages.
- *Implement regular backups for key assets.* Regularly back up model data, training datasets, and configurations to enable quick restoration if there was a failure,

safeguarding valuable progress and data.

Security

Security covers protective measures to safeguard AI models, data, and infrastructure against unauthorized access and threats. Implement updates, monitor model integrity, and control access to prevent vulnerabilities that could compromise sensitive information. These steps are essential to maintain data privacy and trustworthiness of AI solutions in production environments.

- *Schedule updates for Azure resources.* Use maintenance configurations to set specific update schedules for VMs and extensions, reducing vulnerability windows.
- *Patch virtual machines and container images regularly.* Enable automatic guest patching for VMs and scale sets to maintain security against new threats. For more information, see Guest updates and host maintenance overview.
- *Monitor for model drift and ensure integrity.* Ensure model integrity by implementing security mechanisms such as digital signatures or hash verifications for model files to prevent unauthorized modifications. Use Azure Monitor to track key performance metrics and identify model drift, which could indicate potential security vulnerabilities or data shifts. You can define custom metrics (accuracy, F1-score, and data distribution on your models) in your code by using the Azure Monitor Metrics SDK. Azure Monitor Metrics SDK allows you to send your model's performance statistics and data drift measurements to Azure Monitor. Monitoring for performance changes over time can help detect when a model's behavior deviates, potentially signaling an attack or a need for retraining. This proactive approach helps safeguard model integrity and maintain security compliance.
- *Implement auditing and access logs.* Use Azure Monitor and Log Analytics to log access to models and VMs, helping to identify unauthorized access or unusual usage patterns. For more information, see Activity logs in Azure Monitor.
- *Use version control for model files.* Store model files in Azure Storage (Blob, File, or Disk) with versioning to track changes, ensuring a clear audit trail for identifying and rolling back harmful modifications. Using Azure DevOps for version control enhances security by managing access to code changes and enforcing best practices in code reviews. This layered approach mitigates risks of unauthorized changes and provides accountability. For more information, see Blob Versioning in Azure Storage.

- *Set up anomaly detection for model outputs.* Use Azure Monitor to track the output metrics of your models and set up alerts for unusual behavior. For example, monitoring API responses from your model can help detect abnormal output. You can set anomaly detection on a metric like prediction accuracy to automatically detect when it drops outside of an expected range. For more information, see Create and Manage Metric Alerts with Dynamic Thresholds.

- *Enforce model access policies.* Use Azure role-based access control (RBAC) and Microsoft Entra ID to secure access to VMs and model files, limiting access to authorized users only.

- *Regularly revalidate models against updated data.* Implementing periodic revalidation of your model using automated scripts or workflows on your VMs ensures that the model remains accurate and effective against current datasets, mitigating risks from outdated or inaccurate predictions. By scheduling these tasks with Azure Automation or Azure Logic Apps, you can maintain compliance with data standards and enhance overall model security. This proactive approach helps identify vulnerabilities early, ensuring continuous improvement and safeguarding against potential threats. You can schedule your automation workflows to periodically trigger revalidation tasks. Start with an Azure Automation runbook, run in the virtual machine, create an appropriate schedule to get validation results.

- *Track data lineage and model file changes.* Enable versioning in Azure Blob Storage and track data used in training and inference, ensuring no unauthorized data affects model outcomes.

- *Apply resource quotas and rate limits.* Implement rate limits and quotas for your model APIs through Azure API Management to prevent overuse or abuse, which can lead to system vulnerabilities or service outages. This strategy ensures the system remains responsive during high traffic and mitigates risks associated with denial-of-service attacks. By controlling access, you can maintain performance and protect sensitive data from potential exploitation API Management Quotas and Limits.

- *Conduct regular vulnerability scans.* Use Microsoft Defender Vulnerability Scanning to conduct vulnerability assessments of your VMs and related resources. Regularly check for any security issues or misconfigurations in your VM setup that could expose your models. Microsoft Defender Vulnerability Scanning.

Cost optimization

Cost optimization involves aligning resource usage with workload requirements to avoid unnecessary expenses. Right-sizing VMs, committing to reserved instances, and setting up autoscaling help manage costs without compromising performance. Controlling costs on Azure infrastructure is vital for long-term sustainability and scalability of AI deployments.

- *Commit to Reserved Instances*. Save on virtual machine (VM) costs by committing to a one- or three-year term, which offers discounted rates.
- *Use Azure Virtual Machine Scale Sets for automatic scaling.* Automatically scale VM instances based on metrics like CPU usage, paying only for what you need and preventing over-provisioning.
- *Set automatic shutdowns for idle instances.* Avoid costs from unused resources by enabling automatic shutdown, especially for development and test environments.
- *Use Azure Savings Plans for predictable usage.* Reduce costs compared to pay-as-you-go pricing by committing to consistent usage across VM sizes and regions.
- *Use Azure Spot instances for fault-tolerant workloads.* Get substantial discounts on spare capacity for workloads that can tolerate interruptions.
- *Select the right storage solution.* Balance cost and performance based on workload needs. Choose Azure Managed Lustre (AMLFS) for high-throughput, large-scale applications, and Azure NetApp Files (ANF) for advanced data management and reliability.

Operational excellence

Operational excellence involves optimizing the configuration and management of Azure resources to improve the functionality of AI applications. Efficient resource allocation, performance tuning, and distributed training support smooth operation and adaptability to varying demands. This focus on operational efficiency ensures AI models perform as intended, without excessive resource use.

- *Optimize resource allocation.* Regularly review Azure VM sizes and configurations based on actual resource usage to match workload needs. Use Azure Advisor for recommendations on optimal sizing and scaling.
- *Configure autoscaling for efficiency.* Set up autoscaling for VMs or containers to handle workload demands without over-provisioning. Use Azure Virtual Machine Scale Sets to adjust resources dynamically based on demand. For more information, see Azure Virtual Machine Scale Sets.

- *Conduct regular performance tuning.* Continuously profile the application to identify and resolve performance bottlenecks. Use Application Insights Profiler to analyze model code and resource usage.

- *Implement distributed training for efficiency.* Use distributed training techniques, if applicable, to reduce training time by using multiple VMs. Frameworks like Horovod and PyTorch support distributed training on Azure.

- *Save checkpoints in Azure Blob Storage.* Save model states, weights, and configurations periodically to Azure Blob Storage. You can use Azure SDKs or libraries available in the programming language you're using for the LLM. Store the checkpoints in a structured format, like TensorFlow SavedModel or PyTorch checkpoint files. Modify your training or inference code to include checkpoint logic. Start with setting intervals (after every epoch or some iterations) to save the model's state. Use a consistent naming convention for checkpoint files to track the most recent state easily.

- *Design for state recovery.* Ensure your application can recover from a saved checkpoint. Implement logic to load the model's state from Azure Blob Storage when the application starts. It includes, checking for existing checkpoints, and loading the most recent checkpoint if available, allowing the application to resume without losing progress.

Performance efficiency

Performance efficiency refers to maximizing the processing power of Azure infrastructure to meet AI model demands. You should tune GPU settings, optimize input/output (I/O) processes, and run benchmarking tests to improve computational speed and responsiveness. Ensuring high performance supports the execution of complex AI models at scale, which enhances user satisfaction and reduces latency.

GPU tuning

Increase the clock rate of a graphics processing unit (GPU) to improve performance, especially for tasks requiring high graphical processing or complex computations. Higher clock speeds allow the GPU to execute more operations in a given time period, enhancing overall efficiency. Use this GPU-optimization script to set the GPU clock frequencies to their maximum values.

- *Enable Accelerated Networking.* Accelerated Networking is a hardware acceleration technology that allows virtual machines to use single root I/O virtualization (SR-IOV) on supported virtual machines types. It provides lower latency, reduced jitter,

and decreased CPU utilization. Enable accelerated Networking offers substantial enhancements in front-end network performance.

I/O tuning

- *Optimize scratch storage*. Scratch needs to have high throughput and low latency. The training job requires reading data, processing it, and using this storage location as scratch space while the job runs. Ideally, you would use the local SSD directly on each VM. If you need a shared filesystem for scratch, combining all NVMe SSDs to create a Parallel File System (PFS) might be a great option in terms of cost and performance, assuming it has sufficient capacity. One method is to use Azure Managed Lustre. If Azure Managed Lustre isn't suitable, you can explore storage options like Azure NetApp Files or Azure Native Qumulo.

- *Implement checkpoint storage*. Large deep learning training jobs can run for weeks, depending on the number of VMs used. Just like any HPC cluster, you can encounter failures, such as InfiniBand issues, dual in-line memory module (DIMM) failures, error-correcting ode (ECC) errors in GPU memory. It's critical to have a checkpointing strategy. Know the checkpoint interval (when data is saved). Understand how much data is transferred each time. Have a storage solution that meets capacity and performance requirements. Use Blob Storage, if it meets the performance needs.

Benchmarking tests

Benchmarking tests help evaluate and improve distributed deep learning training performance on GPUs, especially for large-scale models. These tests measure the efficiency of GPU communication across nodes, aiming to reduce data transfer bottlenecks during distributed training. The three tests discussed include:

- *Megatron framework*: Supports large-scale language models by improving distributed training efficiency.
- *The NVIDIA Collective Communications Library (NCCL) and ROCm Communication Collectives Library (RCCL) tests*: Evaluate performance and accuracy in multi-GPU communication using NCCL or RCCL libraries, testing patterns like all-reduce and scatter.

These tests ensure scalability and optimal performance for LLMs, with Megatron focusing on model training and NCCL/RCCL on GPU communication.

NVIDIA Megatron-LM test

NVIDIA Megatron-LM is an open-source framework for training large language models. It allows developers to create massive neural networks for NLP tasks, with features including:

- *Parallelism*: Supports model, data, and pipeline parallelism for billion-parameter models.
- *Scalability*: Scales across multiple GPUs and nodes for efficient large model training.
- *Flexibility*: Allows configuration of model architecture, data loading, and training strategies.
- *Optimizations*: Uses NVIDIA GPU optimizations for performance gains.

Megatron-LM deploys on Azure HPC infrastructure, and it uses Azure's scalability for large language models without requiring on-premises hardware.

Megatron-LM test set up

Deploying Megatron-LM requires specific software and hardware.

- *Pick the right deployment options*. Use the CycleCloud Workspace for Slurm to simplify deployment. Choose NC-series or ND-series SKUs for the GPU partition. For multi-node training, ND-series SKUs are recommended for RDMA support. Azure's HPC marketplace images generally include these drivers and libraries. If customization is needed, the azhpc-images repository can ensure compatibility.

- *Use the right image*. The software requirements for the project include a Linux-based operating system, typically Ubuntu. For multi-GPU and multi-node communication, it's essential to have communication libraries such as NCCL and MPI. Additionally, appropriate NVIDIA drivers must be installed to ensure GPU acceleration. Azure's HPC marketplace images come with these drivers and libraries preinstalled. However, if customization is necessary, the azhpc-images repository can be used to ensure compatibility.

Megatron-LM test use

You should run Megatron-LM using the latest release of NGC's PyTorch container. To run the container against a traditional Slurm-based HPC cluster, you need to install and configure these other components in your cluster:

- enroot: a tool that allows users to run containerized applications on HPC clusters without requiring root privileges or modifying the host system.

- pyxis: a plugin for Slurm that enables seamless integration of enroot with Slurm, allowing users to submit containerized jobs to Slurm queues and run them on HPC nodes.

Both of these components are included in CycleCloud Workspace for Slurm but are currently not included in Slurm clusters that are built via CycleCloud. You can introduce these extra components via cluster-init with CycleCloud projects. With these requirements met, you can use Megatron-LM for LLM training by:

- *Verifying the performance of your cluster*: Identify any potential hardware issues before running your workload with Node Health Checks. Use NCCL tests to verify the distributed all-reduce performance of the cluster.
- *Selecting your training data*: Use the codeParrot model as a starting point to validate your workflow.
- *Preprocessing your data*: Use the preprocess_data.py script within the Megatron-LM repository to convert your data to a format that is compatible with Megatron-LM.
- *Training with Megatron-LM*: Use the examples within Megatron-LM as a reference to configure Megatron for training.

This setup ensures efficient deployment and training of large language models on Azure's infrastructure.

NCCL bandwidth test

To verify and optimize GPU communication across nodes, run the NCCL bandwidth test. The NCCL bandwidth test is specialized tool within NCCL, a library that facilitates high-speed communication between GPUs. NCCL supports collective operations, including all-reduce, all-gather, reduce, broadcast, and reduce-scatter, across single or multi-GPU nodes, and achieves optimal performance on platforms with PCIe, NVLink, NVswitch, or networking setups like InfiniBand or TCP/IP. For more information, see NVIDIA/NCCL tests.

NCCL performance metrics

Use the NCCL bandwidth test to assess key metrics, including time and bandwidth. "Time" (in milliseconds) measures the overhead or latency in operations, making it useful for evaluating operations with small data sizes. "Bandwidth" (in GB/s) evaluates point-to-point operation efficiency, such as Send/Receive. "Bus bandwidth" reflects hardware usage efficiency by accounting for inter-GPU communication speed and bottlenecks in components like NVLink or PCI. Calculations for various collective

operations are provided, such as AllReduce, ReduceScatter, AllGather, Broadcast, and Reduce.

NCCL test initiation

To initiate these tests within a CycleCloud deployment, connect to the scheduler node via SSH and access a GPU-equipped compute node. Clone the Git repository for NCCL tests, navigate to the `nccl-tests` directory, and create a host file listing the nodes for testing. Obtain the scheduler node's IP address from CycleCloud's web app.

NCCL test arguments

Before running tests, specify different arguments like the number of GPUs per thread (`-g`), data size range (`-b` for minimum bytes and `-e` for maximum bytes), step increment (`-i` or `-f`), reduction operation type (`-o`), datatype (`-d`), root device (`-r`), iteration count (`-n`), warmup count (`-w`), and CUDA graph settings (`-G`). Refer to the NCCL test documentation for a full list of adjustable parameters.

RCCL tests

ROCm Communication Collectives Library (RCCL) is a specialized library designed for efficient communication between AMD GPUs. It provides collective operations such as all-reduce, all-gather, broadcast, and reduce, supporting both intra- and inter-node GPU communication. Optimized for platforms using PCIe and networking technologies like InfiniBand, RCCL ensures scalable data transfer in multi-GPU environments. It supports integration into both single- and multi-process workflows, such as those using MPI. For more information, see ROCm Communication Collectives Library

- *Set Up Environment*. Install ROCm and ensure RCCL is properly installed on all nodes.
- *Build RCCL Tests*. Clone the repository, navigate to the rccl-tests directory, and compile the tests.
- *Run Bandwidth Tests*. Use the provided test executables (rccl-tests), specifying communication operations like all-reduce.
- *Analyze Performance*. Compare bandwidth and latency results across nodes and GPUs.

Next step

Govern AI

Feedback

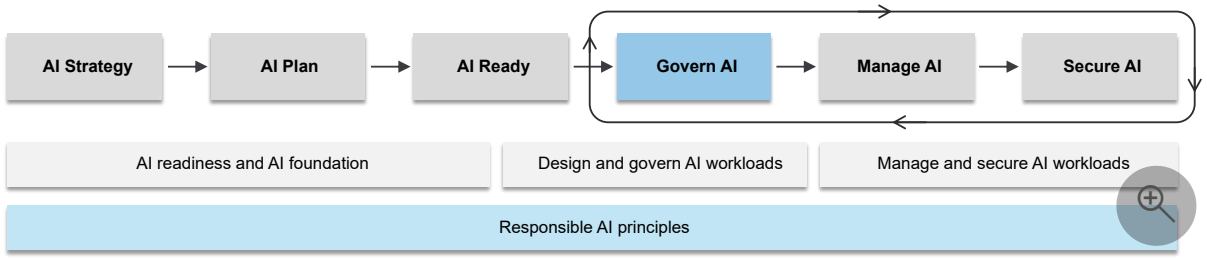
Was this page helpful? Yes No

Govern AI – Process to govern AI

Article • 11/01/2024

This article outlines the organizational process for governing AI. It follows the NIST Artificial Intelligence Risk Management Framework (AI RMF) and NIST AI RMF Playbook. It also aligns with the framework in CAF Govern.

This guidance aims to help you integrate AI risk management into broader risk management strategies. This integration provides more cohesive handling of AI, cybersecurity, and privacy risks for a unified governance approach.



Assess AI organizational risks

AI risk assessment identifies and addresses potential risks introduced by AI technologies. This process builds trust in AI systems and reduces unintended consequences. Addressing organizational risks ensures that AI deployments align with the organization's values, risk tolerance, and operational goals.

- *Understand the AI workloads.* To mitigate AI risks, you must understand your AI workloads. By clarifying the scope and purpose of each AI workload, you can map associated risks. This clarification should include any assumptions and limitations related to the AI workload.
- *Use Responsible AI principles to identify risks.* These principles provide a framework for assessing AI risks. Use the following table to identify and mitigate risks through a structured assessment of AI principles.

Expand table

Responsible AI principle	Definition	Risk assessment question
AI Privacy and Security	AI workloads should respect privacy and be secure.	How might AI workloads handle sensitive data or become vulnerable to security breaches?

Responsible AI principle	Definition	Risk assessment question
Reliability and Safety	AI workloads should perform safely and reliably.	In what situations could AI workloads fail to operate safely or produce unreliable outcomes?
Fairness	AI workloads should treat people equitably.	How could AI workloads lead to unequal treatment or unintended bias in decision-making?
Inclusiveness	AI workloads should be inclusive and empowering.	How might certain groups be excluded or disadvantaged in the design or deployment of AI workloads?
Transparency	AI workloads should be understandable.	What aspects of AI decision-making could be difficult for users to understand or explain?
Accountability	People should be accountable for AI workloads.	Where could accountability be unclear or difficult to establish in the development or use of AI?

- *Identify AI risks.* Start by evaluating the security risks of AI workloads, including potential data breaches, unauthorized access, or misuse. Consult stakeholders to uncover less visible risks, and assess both qualitative and quantitative impacts, including reputational risks, to determine the organization's risk tolerance.
- *Identify risks from external dependencies.* Assess risks related to third-party data sources, software, and integrations. Address issues like security vulnerabilities, bias, and intellectual property risks by establishing policies that ensure alignment with organizational privacy and compliance standards.
- *Assess integration risks.* Evaluate AI workloads integrate with existing workloads and processes. Document potential risks, such as dependency on other workloads, increased complexity, or incompatibilities that could impact functionality.

Document AI governance policies

AI governance policies provide a structured framework for responsible AI usage. These policies align AI activities with ethical standards, regulatory requirements, and business objectives. Documenting policies ensures clear guidelines for managing AI models, data, and operations.

Expand table

AI governance policy area	AI governance policy recommendations
Define policies for selecting and onboarding models	▪ *Establish policies for selecting AI models.* Policies should outline criteria for choosing models that meet organizational values, capabilities, and cost constraints. Review potential models for alignment with risk tolerance and intended task requirements. ▪ *Onboard new models with structured policies.* A formal process for model onboarding maintains consistency in model justification, validation, and approval. Use sandbox environments for initial experiments, then validate and review models in the production catalog to avoid duplication.
Define policies for using third-party tools and data	▪ *Set controls for third-party tools.* A vetting process for third-party tools safeguards against security, compliance, and alignment risks. Policies should include guidelines for data privacy, security, and ethical standards when using external datasets. ▪ *Define data sensitivity standards.* Keeping sensitive and public data separate is essential for mitigating AI risks. Create policies around data handling and separation. ▪ *Define data quality standards.* A "golden dataset" provides a reliable benchmark for AI model testing and evaluation. Establish clear policies for data consistency and quality to ensure high performance and trustworthy outputs.
Define policies for maintaining and monitoring models	▪ *Specify retraining frequency by use case.* Frequent retraining supports accuracy for high-risk AI workloads. Define guidelines that consider the use case and risk level of each model, especially for sectors like healthcare and finance. ▪ *Monitor for performance degradation.* Monitoring model performance over time helps detect issues before they affect outcomes. Document benchmarks, and if a model's performance declines, initiate a retraining or review process.
Define policies for regulatory compliance	▪ *Comply with regional legal requirements.* Understanding regional laws ensures AI operations remain compliant across locations. Research applicable regulations for each deployment area, such as data privacy laws, ethical standards, and industry regulations. ▪ *Develop region-specific policies.* Tailoring AI policies to regional considerations supports compliance with local standards. Policies might include language support, data storage protocols, and cultural adaptations. ▪ *Adapt AI for regional variability.* Flexibility in AI workloads allows for location-specific functionality adjustments. For global operations,

AI governance policy area	AI governance policy recommendations
	document region-specific adaptations like localized training data and feature restrictions.
Define policies for user conduct	▪ *Define risk mitigation strategies for misuse.* Misuse prevention policies help protect against intentional or unintentional harms. Outline possible misuse scenarios and incorporate controls, such as restricted functionalities or misuse detection features. ▪ *Set user conduct guidelines.* User agreements clarify acceptable behaviors when interacting with the AI workload, reducing the risk of misuse. Draft clear terms of use to communicate standards and support responsible AI interaction.
Define policies for AI integration and replacement	▪ *Outline integration policies.* Integration guidelines ensure AI workloads maintain data integrity and security during workload interfacing. Specify technical requirements, data-sharing protocols, and security measures. ▪ *Plan for transition and replacement.* Transition policies provide structure when replacing old processes with AI workloads. Outline steps for phasing out legacy processes, training staff, and monitoring performance throughout the change.

Enforce AI governance policies

Enforcing AI governance policies ensures consistent and ethical AI practices within an organization. Automated tools and manual interventions support policy adherence across deployments. Proper enforcement helps maintain compliance and minimizes human error.

- *Automate policy enforcement where possible* Use platforms like Azure Policy and Microsoft Purview to enforce policies automatically across AI deployments, reducing human error. Regularly assess areas where automation can improve policy adherence.
- *Manually enforce AI policies.* Provide AI risk and compliance training for employees to ensure they understand their role in AI governance. Regular workshops keep staff updated on AI policies, and periodic audits help monitor adherence and identify areas for improvement.
- *Use workload specific governance guidance.* Detailed security guidance is available for AI workloads on Azure platform services (PaaS) and Azure infrastructure (IaaS). Use this guidance to govern AI models, resources, and data within these workload types.

Governance for Azure platforms (PaaS)

Governance for Azure infrastructure (IaaS)

Monitor AI organizational risks

Monitoring AI risks enables organizations to identify emerging risks and address them promptly. Regular evaluations ensure AI workloads operate as intended. Consistent monitoring helps organizations adapt to evolving conditions and prevent negative impacts from AI systems.

- *Establish procedures for ongoing risk evaluation.* Set up regular reviews to identify new risks, engaging stakeholders to assess the broader impacts of AI. Develop a response plan for issues that arise to allow for risk reassessment and necessary adjustments.
- *Develop a measurement plan.* A clear measurement plan ensures consistent data collection and analysis. Define data collection methods, such as automated logging for operational metrics and surveys for qualitative feedback. Establish the frequency and scope of measurements, focusing on high-risk areas, and create feedback loops to refine risk assessments based on stakeholder input.
- *Quantify and qualify AI risks.* Choose quantitative metrics (error rates, accuracy) and qualitative indicators (user feedback, ethical concerns) that align with the workload's purpose. Benchmark performance against industry standards to track the AI's impacts, trustworthiness, and performance.
- *Document and report measurement outcomes.* Regular documentation and reports enhance transparency and accountability. Create standardized reports that summarize metrics, findings, and any anomalies to guide decision-making. Share these insights with stakeholders, using them to refine risk mitigation strategies and improve future deployments.
- *Establish independent review processes.* Regular independent reviews provide objective assessments of AI risks and compliance, using external or uninvolved internal reviewers. Use findings to strengthen risk assessments and refine governance policies.

Next step

Manage AI

Example AI risk mitigations

The following table lists some common AI risks and provides a mitigation strategy and a sample policy for each one. The table doesn't list a complete set of risks.

Expand table

Risk ID	AI risk	Mitigation	Policy
R001	Noncompliance with data protection laws	Use Microsoft Purview Compliance Manager to evaluate data compliance.	The Security Development Lifecycle must be implemented to ensure that all AI development and deployment complies with data protection laws.
R005	Lack of transparency in AI decision making	Apply a standardized framework and language to improve transparency in AI processes and decision making.	The NIST AI Risk Management Framework must be adopted and all AI models must be thoroughly documented to maintain transparency of all AI models.
R006	Inaccurate predictions	Use Azure API Management to track AI model metrics to ensure accuracy and reliability.	Continuous performance monitoring and human feedback must be used to ensure that AI model predictions are accurate.
R007	Adversarial attack	Use PyRIT to test AI workloads for vulnerabilities and strengthen defenses.	The Security Development Lifecycle and AI red team testing must be used to secure AI workloads against adversarial attacks.
R008	Insider threats	Use Microsoft Entra ID to enforce strict access controls that are based on roles and group memberships to limit insider access to sensitive data.	Strict identity and access management and continuous monitoring must be used to mitigate insider threats.
R009	Unexpected costs	Use Microsoft Cost Management to track CPU, GPU, memory, and storage usage to ensure efficient resource utilization and prevent cost spikes.	Monitoring and optimization of resource usage and automated detection of cost overruns must be used to manage unexpected costs.

Risk ID	AI risk	Mitigation	Policy
R010	Underutilization of AI resources	Monitor AI service metrics, like request rates and response times, to optimize usage.	Performance metrics and automated scalability must be used to optimize AI resource utilization.

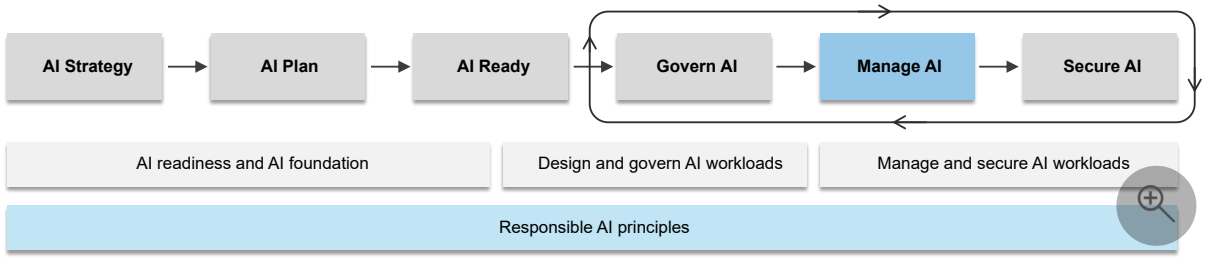
Feedback

Was this page helpful? Yes No

Manage AI – Process to manage AI

Article • 11/01/2024

This article outlines the organizational process for managing AI workloads. It provides recommendations for managing AI workloads from development, deployment, and operations. Effective AI management requires a structured approach from development through deployment and ongoing operations. Businesses need standardized practices and regular monitoring to prevent issues such as data and model drift, ensuring AI remains accurate and reliable over time.



Manage AI operations

Managing AI operations ensures visibility and consistency throughout the AI lifecycle. By adopting operational frameworks like MLOps, creating sandbox environments, and establishing CI/CD pipelines, you can oversee development, testing, and deployment.

- *Adopt an AI operational framework.* Implement MLOps (Machine learning operations) frameworks for traditional machine learning workflows and GenAIOps for generative AI workloads. These operational frameworks organize the end-to-end cycle for AI development. Each framework affects the workload team's approach and tooling. For more information, see MLOps and GenAIOps.

- *Standardize AI development tools.* Define and standardize the use of SDKs and APIs for consistency across development teams. Tools like Azure SDK for AI workloads provide libraries and APIs that are optimized for scaling AI models and integrating them into applications. For generative AI, standardize your AI platform and orchestrators, such as Semantic Kernel, LangChain, and Prompt Flow.

- *Use a sandbox environment for AI experimentation.* Use a sandbox environment for AI model experimenting. You want to consistency across dev, test, and prod environments. So, the sandbox environment should be distinct from dev, test, and production environments in the AI development lifecycle. If you change deployment and governance models between dev, test, and prod environments, it can hide and introduce breaking changes.

- *Establish continuous integration and continuous delivery pipelines for deployment.* Ensure that your data pipelines cover code quality checks, including linting and static analysis. Data pipelines should also include unit and integration tests, as well as experimentation and evaluation flows. Finally, incorporate production deployment steps, such as promoting releases to test and production environments following manual approvals. Maintain separation between models, prompt flows, and the client user interface to ensure updates to one component don't affect others. Each flow should have its own lifecycle for independent promotion.

Manage AI deployment

AI deployment management is about defining who can deploy AI resources and who governs these endpoints. A structured approach, led by an AI center of excellence, helps businesses decide whether workload teams or a central team should manage resources, balancing development speed with governance requirements. The AI CoE should lead the effort to determine the best approach.

- *Use workload-team management of AI resources for faster development.* When workload teams manage AI resources, they have the autonomy to deploy and manage AI resources within the confines of your governance policies. Use Azure Policy to enforce governance consistently across all workload environments. Create and communicate AI policies that the workload teams must follow to address any governance gaps. For example, create generative AI policies to enforce content filter settings and prevent the use of disallowed models. Make these policies clearly known to workload teams and audit regularly.

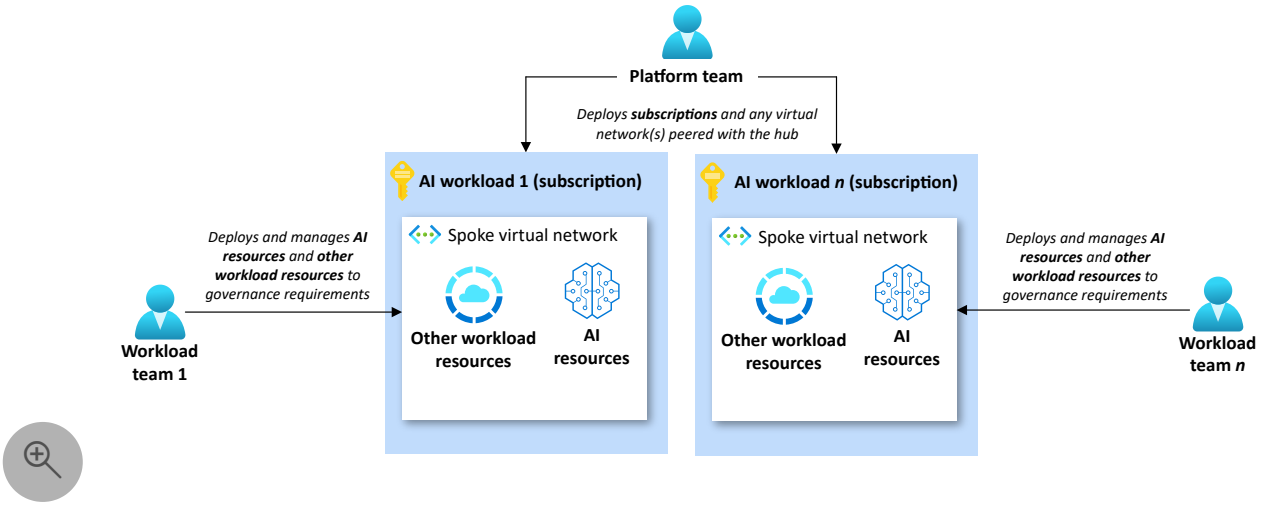


Figure 1. Workload-team management of AI resources.

- *Use a shared management of AI resources increased AI governance.* In a shared AI management approach, a central team manages AI resources for all AI workloads.

This team deploys core AI resources and configures security and governance that all workload teams use. Use this approach if you want a single team to control AI deployments and governance across your workloads.

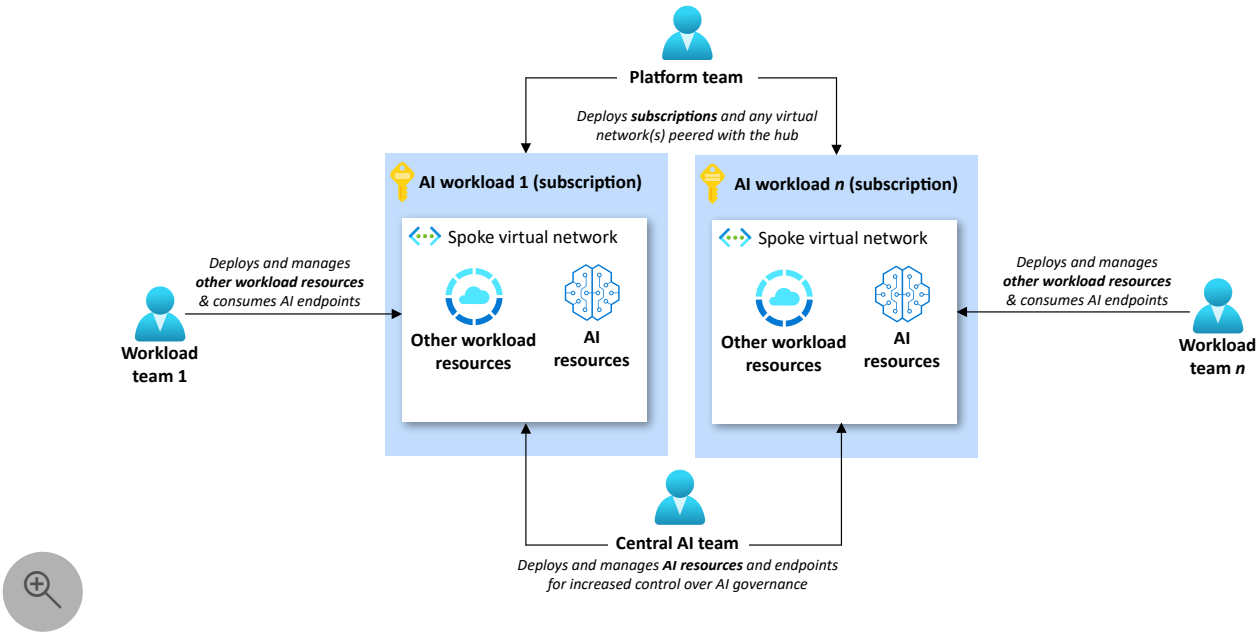


Figure 2. Central AI team management of AI resources.

Manage AI endpoint sharing

Sharing AI endpoints across workloads can streamline management, but it requires careful consideration of governance and model requirements. Businesses should only share endpoints within a single workload with consistent needs, as shared usage across differing needs can complicate governance and increase costs.

- *Avoid sharing AI endpoints when governance and model needs vary.* Workloads that require different content filter settings, such as governance on input and output, shouldn't share an endpoint. Also, don't share a single AI endpoint if a different AI model would provide a more cost-effective way to meet workload requirements.

- *Share AI endpoints only within a single workload.* Sharing an AI endpoint works best when a workload team has multiple applications as part of the same workload. AI endpoint sharing provides the least amount of management overhead and simplifies deployment. These applications must share the same governance needs and AI model needs. Sharing endpoints can cause you to hit rate limits and quota limitations. Most Azure services have limits per subscription. Within a subscription, each region has quota limits.

Manage AI models

AI model management involves setting governance structures, continuous monitoring, and retraining to maintain performance over time. This process helps businesses align models with ethical standards, track model performance, and ensure that AI systems remain effective and aligned with business objectives.

- *Establish a governance structure for AI oversight.* Create an AI center of excellence (AI CoE) or appoint an AI lead. They should ensure adherence to responsible AI standards. They should make decisions on whether systems need to be adjusted based on these reports. Use the Responsible AI dashboard to generate reports around model outputs.

- *Define an AI measurement baseline.* Establish a measurement baseline to ensure that AI models align with business goals and ethical standards. Use KPIs that are related to responsible AI principles like fairness, transparency, and accuracy. Map these KPIs to AI workloads. For example, in a customer service chatbot, measure fairness by evaluating how well the model performs across different demographic groups. To take these measurements, start with the tools used in the Responsible AI dashboard.

- *Implement continuous monitoring.* AI workloads can change over time due to evolving data, model updates, or shifts in user behavior. Monitor AI models, AI resources, AI data to ensure that these workloads remain aligned with KPIs. Conduct audits to assess AI systems against the defined responsible AI principles and metrics.

- *Identify root causes of performance issues.* Pinpoint the source of the issue when a drop in performance or accuracy is detected by monitoring the AI. Ensure that you have visibility into each stage of the interaction to isolate the problem and implement corrective actions more quickly. For example, if a customer service chatbot generates inaccurate responses, monitoring should help you determine whether the error is in the prompt crafting or the model's understanding of context. Use built-in tools like Azure Monitor and Application Insights to proactively identify performance bottlenecks and anomalies.

- *Track model retirement.* Track retirement for pretrained models to prevent performance issues as vendor support ends. For instance, a generative AI model might be deprecated, so you'd need to update it to maintain functionality. Studio shows the model retirement date for all deployments.

- *Retrain AI models as needed.* Account for models degrading over time because of changes in data. Schedule regular retraining based on model performance or business needs to ensure that the AI system stays relevant. Retraining can be expensive, so assess the initial training cost and use that cost to evaluate how

frequently you should retrain AI models. Maintain version control for models and ensure a rollback mechanism for underperforming versions.

- *Establish model promotion process.* Use quality gates to promote trained, fine-tuned, and retrained models to higher environments based on performance criteria. The performance criteria are unique to each application.

Manage AI costs

Managing AI costs requires a clear understanding of expenses related to resources like compute, storage, and token processing. You should implement cost management best practices, monitor usage, and set up automated alerts to avoid unexpected expenses and optimize resource efficiency.

- *Follow cost management best practices for each service.* Each Azure service has specific features and best practices that maximize cost optimization. Familiarize yourself with following guidance for planning and managing cost in Azure AI Foundry, Azure OpenAI Service, and Azure Machine Learning.

- *Monitor and maximize billing efficiency.* Understand cost breakpoints to avoid unnecessary charges. Examples include making full use of fixed-price thresholds for image generation or hourly fine-tuning. Track your usage patterns, including tokens per minute (TPM) and requests per minute (RPM), and adjust models and architecture accordingly. Consider a commitment-based billing model for consistent usage patterns.

- *Set up automated cost alerts.* Use budget alerts notify you of unexpected charges and establish budgeting strategies to control and predict your AI expenses.

For generative AI applications using Azure OpenAI, see these cost optimization recommendations.

Manage AI data

Effective AI data management focuses on maintaining data accuracy, integrity, and sensitivity throughout the AI lifecycle. When you curate high-quality datasets and securing data pipelines, your organization can ensure that data remains reliable and compliant with changing regulatory requirements.

- *Maintain data accuracy and curate golden datasets.* Develop an authoritative set of data used for regular testing and validation across both AI types. Continuously curate this dataset to ensure it reflects up-to-date, accurate information.

- *Ensure data pipeline integrity.* Develop and maintain custom data pipelines to ensure data integrity from data collection to preprocessing and storage. Each step of the pipeline must be secure to maintain performance and reliability in both types of AI applications.

- *Manage data sensitivity changes.* Understand that the sensitivity classification of data can change over time. You might want to reclassify low sensitivity data as highly sensitive because of business or regulatory changes. Develop processes for removing or replacing sensitive data in downstream systems. Microsoft Defender for Cloud and Microsoft Purview can help you label and manage sensitive data. This process starts with a good data catalog before AI ingestion. When changes occur, identify all models or systems that use the sensitive data. If possible, retrain AI models by using datasets that exclude the reclassified sensitive data.

Manage AI business continuity

Business continuity and disaster recovery for AI involve creating multi-region deployments and regularly testing recovery plans. These strategies help ensure AI systems remain operational during disruptions and minimize the risk of prolonged outages or data loss.

- *Use multiregion deployments for AI.* Implement multiregion deployments to ensure high availability and resiliency for both generative and nongenerative AI systems. These strategies minimize downtime and ensure that critical AI applications remain operational during regional outages or infrastructure failures. Make sure to implement the necessary redundancy for trained and fine-tuned models to avoid the need for retraining during an outage.

- *Test and validate disaster recovery plans regularly.* Perform regular tests of disaster recovery plans to verify that you can restore generative and nongenerative AI systems effectively. Include testing of data restoration processes and validation procedures to ensure that all AI components are functioning properly after recovery. Validating regularly ensures that the organization is prepared for real-world incidents and minimizes the risk of failures during recovery.

- *Manage and track changes to AI systems.* Ensure that all changes to models, data, and configurations are managed through version control systems such as Git. Doing so is critical for tracking modifications and ensuring the ability to restore previous versions during recovery. For generative and nongenerative AI, automated auditing of model and system changes should be in place so that you can quickly identify and revert unplanned alterations.

Next step

Secure AI

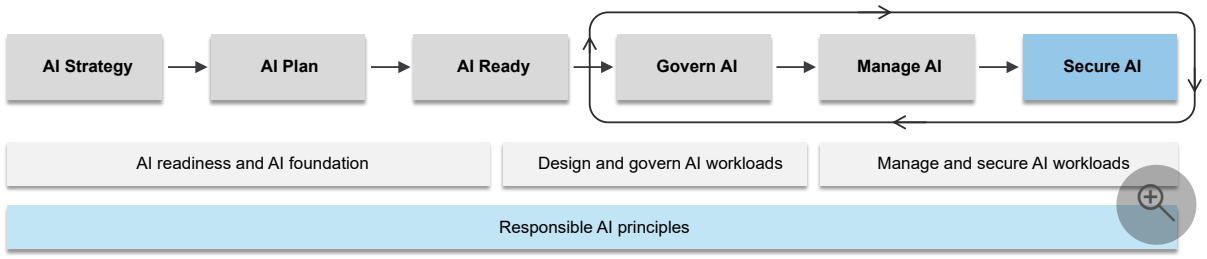
Feedback

Was this page helpful? Yes No

Secure AI – Process to secure AI

Article • 01/30/2025

This article outlines the organizational process for securing AI workloads. It focuses on the confidentiality, integrity, and availability (CIA) of AI. Effective security practices reduce the risk of compromise by safeguarding the confidentiality, integrity, and availability of AI models and data. A secure AI environment also aligns with business security goals and enhances trust in AI-driven processes.



Assess AI security risks

Assessing AI security risks involves identifying and evaluating potential vulnerabilities that might affect AI workloads. Proactively addressing these risks helps prevent breaches, manipulation, and misuse, which strengthens the reliability of AI applications. This approach also supports organizational goals by protecting sensitive data and maintaining stakeholder trust.

- *Identify common AI security risks.* Use recognized resources like MITRE ATLAS and OWASP Generative AI risk to regularly evaluate risks across all AI workloads.
- *Manage your AI security posture.* For ongoing security posture management, consider using AI security tools like AI security posture management in Microsoft Defender for Cloud. These tools can automate the detection and remediation of generative AI risks.
- *Identify data risks.* Use enterprise-wide tools like Microsoft Purview Insider Risk Management to assess insider risk and maintain data security throughout the business. Across all AI workloads, classify and prioritize risks based on the sensitivity of the data that they process, store, or transmit.
- *Red team AI models.* Conduct red team testing against generative AI models and nongenerative models to assess their vulnerability to attacks. Follow these recommendations for red teaming AI:

- *Assess system capabilities and application context.* Identify what the AI system can do and where it is applied to target real-world vulnerabilities effectively. Work backward from potential impacts to design meaningful attack strategies.
- *Use simple attack techniques first.* Exploit basic prompt engineering and system weaknesses before attempting complex adversarial attacks. Many real-world breaches rely on low-resource techniques.
- *Distinguish red teaming from benchmarking.* AI red teaming uncovers unknown risks. Benchmarking assesses known vulnerabilities. Focus on testing AI in real-world scenarios rather than relying solely on predefined evaluation metrics.
- *Automate to expand risk coverage.* Use tools like PyRIT to test AI systems at scale but maintain human oversight.
- *Prioritize human judgment in AI red teaming.* Automation aids testing, but humans provide necessary context for evaluating nuanced risks like bias, emotional responses, and cultural implications.
- *Develop reliable methods to measure responsible AI failures.* Responsible AI failures occur when AI systems violate the principles of responsible AI. Unlike security vulnerabilities, these failures are harder to define and measure due to their subjective, social, and ethical implications. Use structured guidelines and scenario-based assessments to evaluate and mitigate harmful outputs.
- *Secure both traditional and AI-specific threats.* Address conventional security vulnerabilities alongside AI risks like prompt injections and data exfiltration. Strengthen both system-level and model-specific defenses.

For more information, see Lessons from red teaming 100 generative AI products.

Implement AI security controls

Implementing AI security controls means establishing policies, procedures, and tools that safeguard AI resources and data. These controls help ensure compliance with regulatory requirements and protect against unauthorized access, supporting continuous operation and data privacy. When you apply consistent controls across AI workloads, you can manage security more effectively.

Secure AI resources

Securing AI resources includes managing and protecting the systems, models, and infrastructure that support AI applications. This step reduces the likelihood of

unauthorized access and helps standardize security practices across the organization. A comprehensive resource inventory allows consistent application of security policies and strengthens overall control of AI assets.

- *Establish a centralized AI asset inventory.* Maintaining a detailed and up-to-date inventory of your AI workload resources ensures you can apply security policies uniformly to all AI workloads. Compile a company-wide inventory of all AI systems, models, datasets, and infrastructure across Azure. Utilize tools like Azure Resource Graph Explorer and Microsoft Defender for Cloud to automate the discovery process. Microsoft Defender for Cloud can discover generative AI workloads and in predeployment generative AI artifacts.

- *Secure Azure AI platforms.* Standardize the application of Azure security baselines for every AI resource. Follow the security recommendations in Azure Service Guides.

- *Use workload specific governance guidance.* Detailed security guidance is available for AI workloads on Azure platform services (PaaS) and Azure infrastructure (IaaS). Use this guidance to secure AI models, resources, and data within these workload types.

  Security for Azure platforms (PaaS)

  Security for Azure infrastructure (IaaS)

Secure AI data

Securing AI data involves protecting the data that AI models use and generate. Effective data security practices help prevent unauthorized access, data leaks, and compliance breaches. Controlling data access and maintaining a detailed catalog also support informed decision-making and reduce the risk of exposing sensitive information.

- *Define and maintain data boundaries.* Ensure AI workloads use data appropriate for their access level. AI applications accessible to all employees should only process data suitable for all employees. Internet-facing AI applications must use data appropriate for public consumption. Use separate datasets or environments for different AI applications to prevent inadvertent data access. Consider using Microsoft Purview's suite of data security tools to secure your data.

- *Implement strict data access controls.* Ensure applications verify that end-users are authorized to access the data involved in their queries. Avoid broad system

permissions for user actions. Operate under the principle that if the AI can access certain information, the user should be authorized to access it directly.

- *Maintain a data catalog.* Keep an up-to-date catalog of all data connected to and consumed by AI systems, including storage locations and access details. Regularly scan and label data to track sensitivity levels and suitability, aiding in analytics and risk identification. Consider using Microsoft Purview Data Catalog to map and govern your data.

- *Create a data sensitivity change management plan.* Track data sensitivity levels as they can change over time. Use your data catalog to monitor information used in AI workloads. Implement a process to find and remove sensitive data from AI workloads.

- *Secure AI artifacts.* Recognize AI models and datasets as valuable intellectual property and implement measures to protect them accordingly. Store AI models and datasets behind private endpoints and in secure environments such as Azure Blob Storage and dedicated workspaces. Apply strict access policies and encryption to safeguard AI artifacts against unauthorized access or theft to prevent data poisoning.

- *Safeguard sensitive data.* When the original data source is unsuitable for direct use, use duplicates, local copies, or subsets that contain only the necessary information. Process sensitive data within controlled environments that feature network isolation and rigorous access controls to prevent unauthorized access or data leaks. Additionally, implement comprehensive safeguards such as encryption, continuous monitoring, and intrusion detection systems to protect against data breaches during processing.

Maintain AI security controls

Maintaining AI security controls includes ongoing monitoring, testing, and updating of security measures to address evolving threats. Regularly reviewing security controls ensures that AI workloads remain protected and that the organization can adapt to new risks. Proactive maintenance helps prevent breaches and maintains trust in AI systems over time.

- *Implement testing for data leakage and coercion in AI systems*. Conduct rigorous tests to determine if sensitive data can be leaked or coerced through AI systems. Perform data loss prevention (DLP) tests and simulate AI-specific attack scenarios. Simulate model inversion or adversarial attacks to evaluate the resilience of data protection measures. Ensuring that AI models and data handling processes are

secure against unauthorized access and manipulation is critical for maintaining data integrity and trust in AI applications.

- *Provide AI-focused employee training and awareness.* Provide training programs for all employees involved in AI projects. Emphasize the importance of data security and best practices that are specific to AI development and deployment. Educate staff on how to handle sensitive data that's used in training and recognize threats like model inversion or data poisoning attacks. Regular training ensures that team members are knowledgeable about the latest AI security protocols and understand their role in maintaining the integrity of AI workloads.
- *Develop and maintain an incident response plan for AI security incidents.* Create an incident response strategy tailored to AI systems to address potential data breaches or security incidents. The plan should outline clear procedures for detecting, reporting, and mitigating security incidents that might affect AI models, data, or infrastructure. Conduct regular drills and simulations focused on AI-specific scenarios to ensure that the response team is prepared to handle real-world AI security incidents efficiently.
- *Conduct periodic risk assessments.* Evaluate emerging threats and vulnerabilities specific to AI regularly through risk assessments and impact analyses. These evaluations help identify new risks that are associated with AI models, data handling processes, and deployment environments. Evaluations also assess the potential effects of security breaches on AI systems.

Next steps

Govern AI, Manage AI, and Secure AI are continuous processes you must iterate through regularly. Revisit each AI Strategy, AI Plan, and AI Ready as needed. Use the AI adoption checklists to determine what your next step should be.

AI checklists

Feedback

Establish an AI Center of Excellence

Article • 11/01/2024

An AI Center of Excellence (AI CoE) is a dedicated team or organizational structure that centralizes AI expertise, resources, and governance. It serves as the nerve center for AI initiatives, ensuring that your organization effectively uses AI to achieve business objectives. This guide provides a step-by-step approach to building a practical and impactful AI CoE.

What is an AI CoE?

An AI CoE serves as a centralized hub for AI initiatives. It provides a structured approach to AI adoption and aligns AI workloads with business goals. The AI CoE also establishes development standards, oversees compliance and ethical concerns, and promotes an AI-driven mindset across the organization.

Why is an AI CoE important?

An AI CoE facilitates AI adoption by streamlining initiatives, reducing duplication, and focusing on projects with significant business results. It establishes governance structures to manage ethical and compliance issues, fosters collaboration, and enables knowledge sharing.

Define the AI CoE function

The first step in building an AI CoE is to clearly define its role and objectives. The CoE should focus on operationalizing the following areas.

The first step involves defining the AI CoE's role and objectives. Focus on operationalizing key areas:

- *Business strategy:* Identify business goals that AI can support, prioritize use cases, and establish measurable KPIs to track success. Develop a roadmap to guide employee engagement with AI and foster skill development.
- *Technology strategy:* Design an AI-ready platform and data architecture. Create a decision framework for building or purchasing AI tools and plan for scalable storage, compute, and application hosting.

- *AI development:* Develop customer-centric solutions and implement a process for building, testing, and deploying AI models across various business units. Ensure each model aligns with business needs and delivers tangible value.
- *Cultural integration:* Establish a formal operating model to guide AI activities. Secure executive sponsorship to promote organizational commitment. Develop structured learning pathways to upskill employees and create governance policies that ensure ethical AI use and data security.
- *Governance:* Implement controls and accountability structures to monitor AI ethics, data privacy, and security. Establish a governance model that enforces responsible AI use across the organization.

Build a cross-functional team

An AI CoE requires a diverse set of skills and expertise. Assemble a cross-functional team by assigning clear roles and responsibilities:

Expand table

Role	Responsibilities	Key deliverables
AI CoE Lead	Sets the strategic direction of the CoE	AI roadmap, leadership for AI initiatives
AI Strategist	Aligns AI strategy with business objectives	AI strategy document, prioritized AI projects
Business Analyst	Integrates AI solutions into business workflows	Business case documentation, process improvement plans
Data Scientist	Develops and tests AI models	AI models, data insights, and actionable recommendations
Data Engineer	Manages data pipelines and infrastructure	Data integration plan, data quality assurance reports
AI Engineer	Deploys and maintains AI systems	AI system architecture, deployment schedules, and maintenance logs
Chief Ethics Officer	Monitors AI ethical standards and compliance	AI ethics review processes, risk assessment reports
Compliance Officer	Ensures AI compliance with regulations	Compliance documentation, regulatory reports
MLOps	Oversees AI model lifecycle	AI model pipeline, continuous improvement

Role	Responsibilities	Key deliverables
Specialist	management	processes

Define structure and operations

Determine whether the AI CoE operates as an extension of an existing Cloud CoE or functions as a standalone team. Define workflows to ensure that AI projects align with business goals.

- *Identify strategic opportunities:* Collaborate with business leaders to uncover AI use cases. Prioritize use cases with high business value and feasibility.
- *Create an implementation roadmap:* Develop a timeline for AI adoption, specifying the necessary infrastructure, tools, and personnel.
- *Enable professional and citizen developers:* Provide resources, training, and self-service tools. Set up a support system for ongoing learning and troubleshooting.
- *Foster an AI-driven culture:* Develop a change management plan, encourage collaboration between teams, and recognize innovative AI-driven outcomes.
- *Implement AI governance:* Set up frameworks to monitor ethical AI use, review models for bias and transparency, and regularly audit systems for data security and compliance.

Implement, monitor, and evolve

After establishing the AI CoE, continuously monitor performance, make adjustments, and scale AI initiatives as needed:

- *Monitor AI performance:* Track KPIs and business metrics related to AI initiatives. Use feedback loops to improve model accuracy.
- *Iterate and scale:* Optimize AI processes based on lessons learned from pilot projects, and expand successful solutions to other business units or regions.
- *Maintain compliance and ethics:* Conduct regular audits to ensure adherence to ethical standards and regulatory requirements. Update governance frameworks as necessary.
- *Foster continuous learning:* Provide ongoing training programs and encourage experimentation to keep employees up-to-date on AI advancements.

Next step

Use the AI adoption checklists to determine what your next step should be.

AI adoption checklists

Feedback

Was this page helpful? Yes No