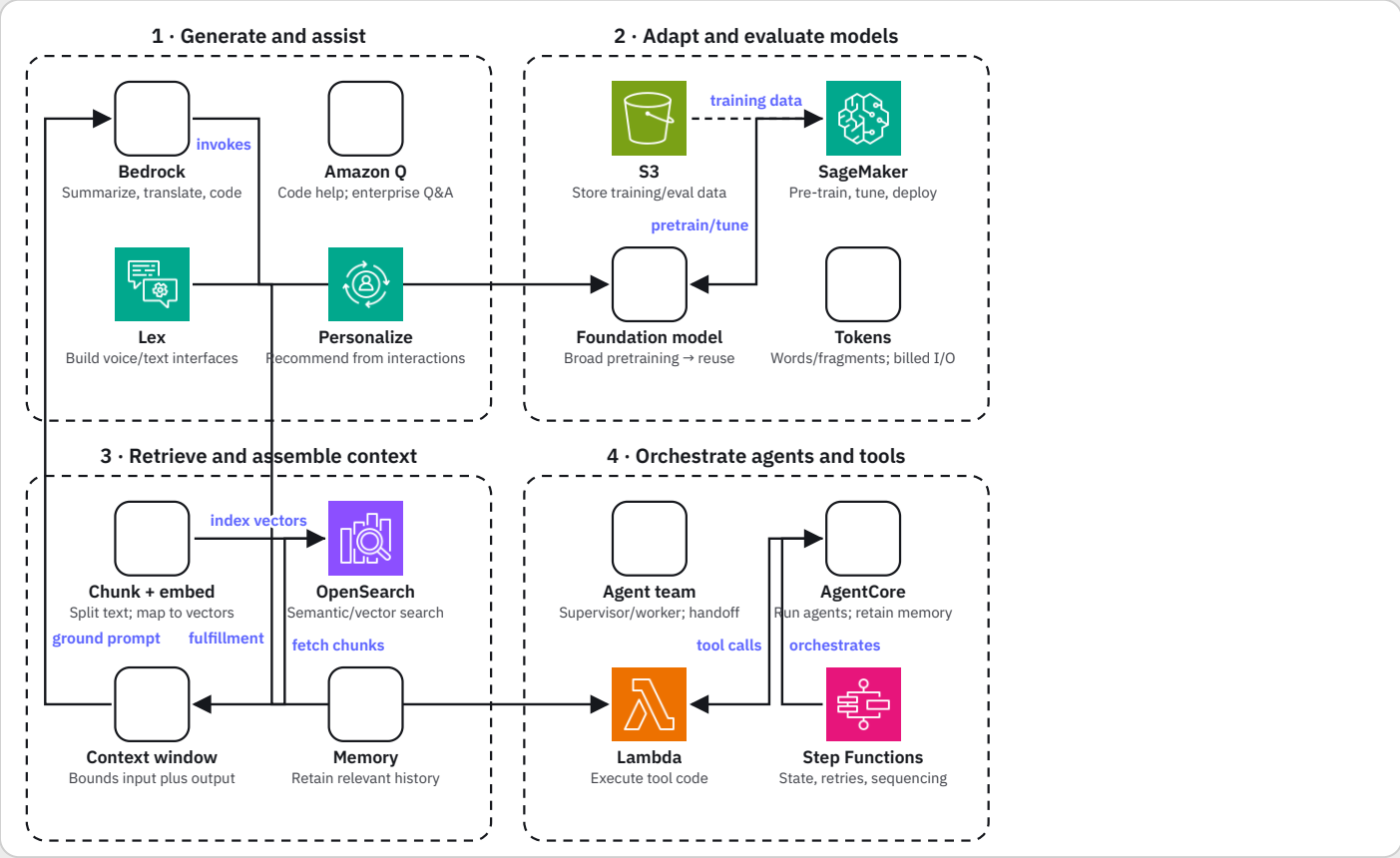
DOMAIN 2 · FUNDAMENTALS OF GENAI · 24% OF THE EXAM · TASK STATEMENT 2.1

Explain the basic concepts of generative AI (GenAI)

Generate, adapt, ground, orchestrate



1 Generate and assist

Model choice determines capabilities: transformer LLMs use attention to generate language; diffusion models iteratively denoise. Multimodal models accept or generate multiple media types: text, images, audio, or video.

2 Adapt and evaluate models

Fine-tuning changes pretrained weights; pre-training learns general patterns. Select data/model → train or adapt → evaluate quality, safety, cost, and latency → deploy → collect feedback and re-evaluate.

3 Retrieve and assemble context

Context engineering selects instructions, history, retrieved chunks, and tool results within the window. Similar embeddings enable semantic retrieval. Prompt engineering shapes instructions; memory must be selected and supplied.

4 Orchestrate agents and tools

MCP standardizes access to external tools/data, not agent-to-agent messaging. Agents delegate or hand off tasks; orchestration controls communication and state. Applications authorize, validate, and execute tool calls.

Exam traps

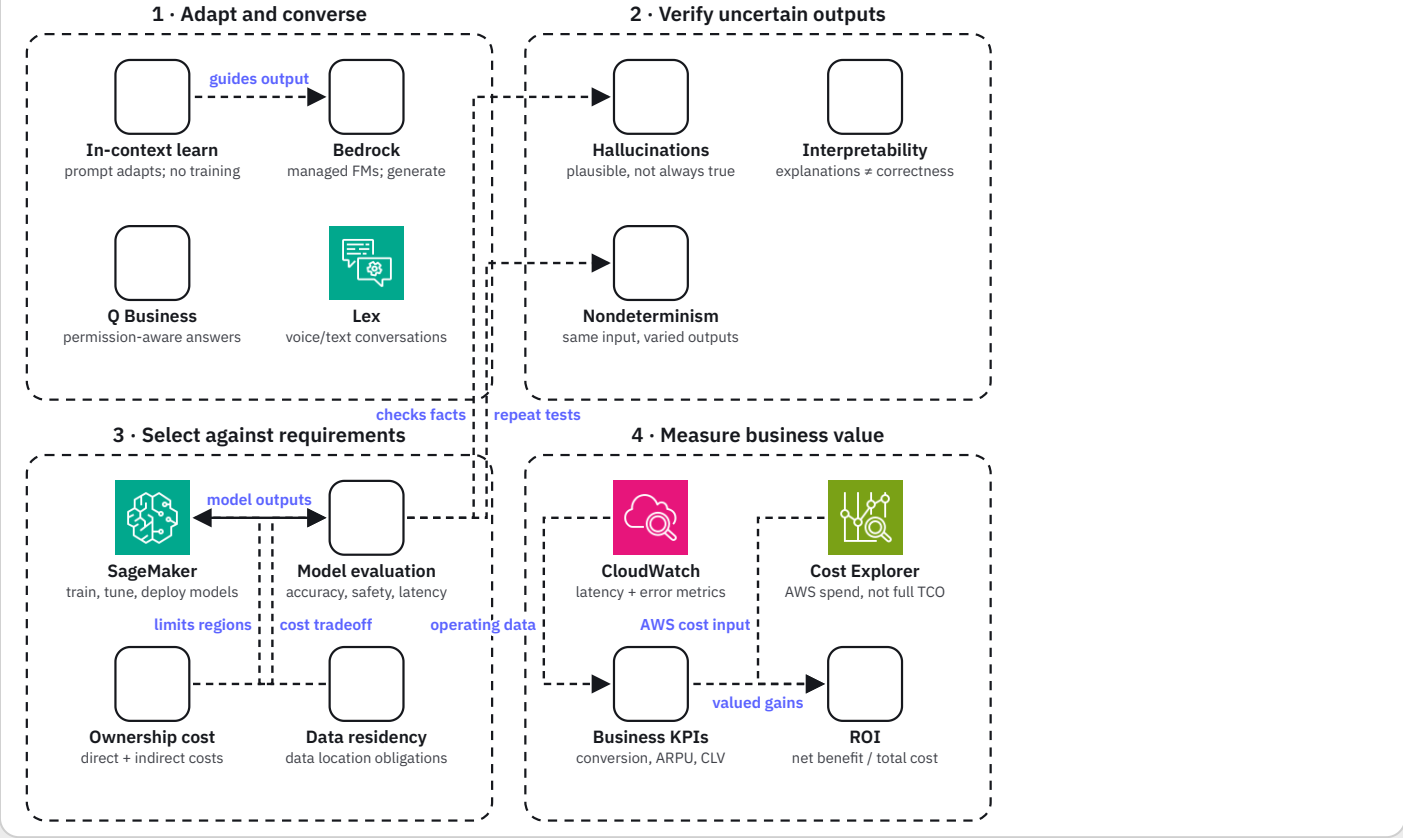
- **Tokens ≠ words**. Token-metered inference cost ≈ input tokens × input rate + output tokens × output rate; rates vary by model. Longer inputs/outputs generally increase latency.
- **RAG ≠ fine-tuning**: retrieval supplies external evidence without changing model weights; fine-tuning changes weights. Neither guarantees factual answers.
- **Conversational ≠ generative**: Amazon Lex builds voice/text interfaces; Amazon Personalize predicts recommendations from interactions. Neither inherently requires a generative FM.

Get practice exams and lessons for AWS Certified AI Practitioner

DOMAIN 2 · FUNDAMENTALS OF GENAI · TASK STATEMENT 2.2

Understand the capabilities and limitations of GenAI for solving business problems

Adapt, verify, select, measure



1 Adapt and converse

In-context learning adapts responses without changing model weights. Bedrock generates content without managing model servers; Q Business retrieves enterprise knowledge; Lex manages conversational interfaces.

2 Verify uncertain outputs

Verification matters when false claims cause harm. Grounding and human review reduce risk, not guarantee accuracy. Repeated tests expose output variability; explanations support debugging, not proof.

3 Select against requirements

Representative evaluation compares model types and capabilities across task domains. Selection balances quality, latency, complexity, compliance and TCO. Residency includes replicas, backups, logs and downstream systems.

4 Measure business value

Business KPIs need a baseline, target and review period. Track efficiency, conversion, average revenue per user, accuracy and customer lifetime value. ROI connects monetized benefits to total costs, not model scores alone.

Exam traps

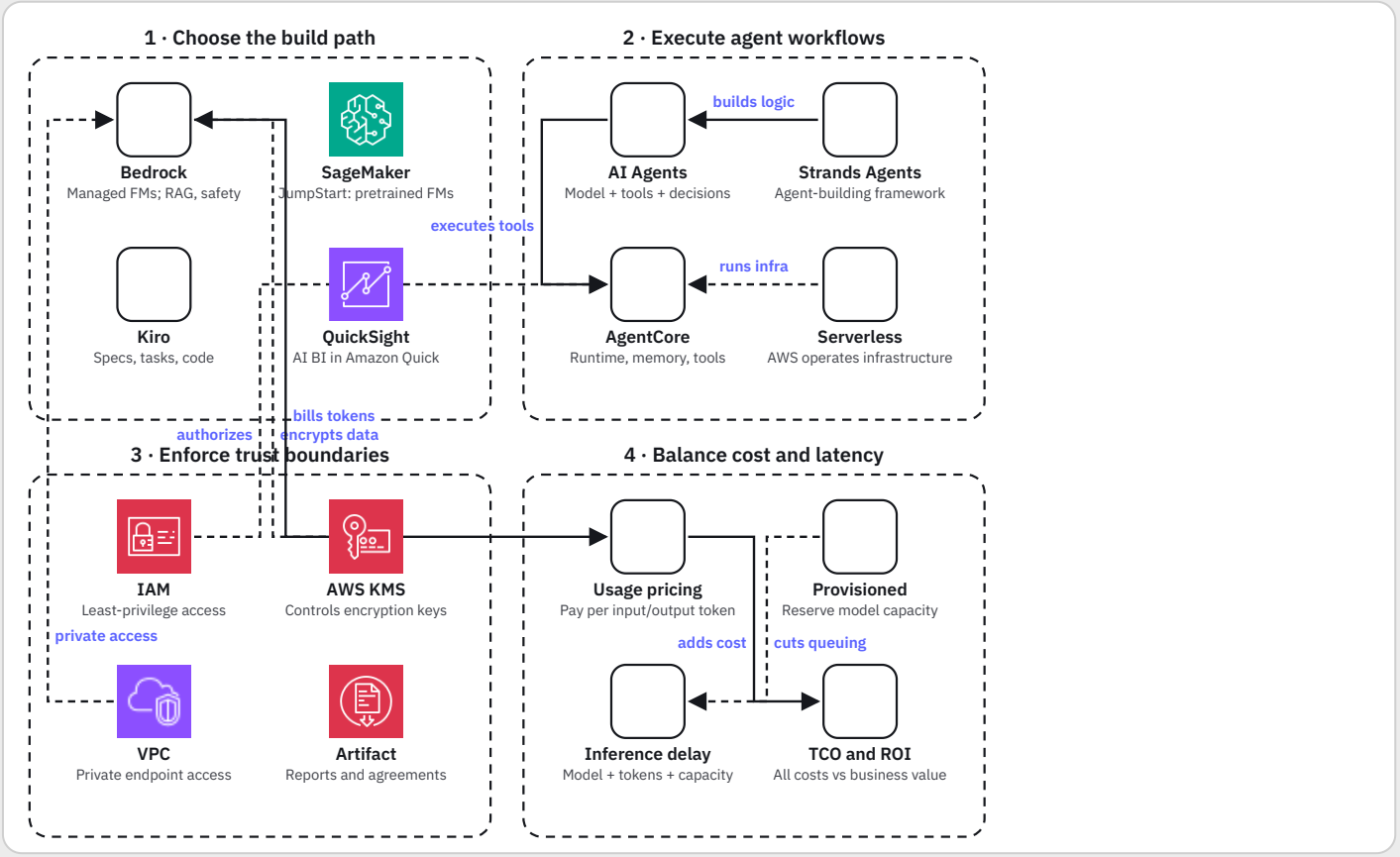
- **Conversation ≠ generation**: Amazon Lex supports voice/text dialogue without requiring generative AI; Amazon Q Business answers from enterprise sources with configured permissions.
- **Grounding reduces, not eliminates, hallucinations**. Low temperature does not guarantee identical outputs, and interpretability does not prove accuracy or fairness.
- **Lowest token price ≠ lowest TCO**: include retrieval, hosting, evaluation and human review. Choosing a Region alone does not establish data-residency compliance.

Get practice exams and lessons for AWS Certified AI Practitioner

DOMAIN 2 · FUNDAMENTALS OF GENAI · TASK STATEMENT 2.3

Describe AWS infrastructure and technologies for building GenAI applications

Build, execute, secure, optimize



1 Choose the build path

Bedrock offers managed model APIs for faster delivery; SageMaker supports deeper training and deployment control. Kiro builds code; QuickSight delivers AI-powered BI.

2 Execute agent workflows

AgentCore supplies agent infrastructure; Strands builds agent logic. AgentCore supports different frameworks and models. Serverless reduces operational work, not customer responsibility or necessarily total cost.

3 Enforce trust boundaries

Shared responsibility: AWS secures infrastructure; customers govern data, access and configurations. Artifact supplies compliance evidence, not automatic compliance; service and Region scope must match requirements.

4 Balance cost and latency

Provisioned throughput reserves capacity for predictable demand; on-demand suits variable use. Custom models, redundancy and regional availability change cost and deployment choices.

Exam traps

- **Managed does not mean hands-off**: Bedrock safety controls do not replace customer access policies, data governance or output validation.
- **Token pricing is not universal**: Bedrock on-demand commonly bills input/output tokens; SageMaker hosting commonly bills instance time. Provisioned capacity can incur cost when idle.
- **Lowest unit price is not lowest TCO**: Longer outputs add token cost and latency; operations and redundancy add costs. ROI measures business value, not model accuracy.

Get practice exams and lessons for AWS Certified AI Practitioner

DOMAIN 2 · FUNDAMENTALS OF GENAI · WHAT TO REMEMBER

Concepts and services of Fundamentals of GenAI

Concepts to know

Tokenization

Tokenization splits content into model-specific units, not necessarily words. Input and output token counts affect context capacity, inference cost, and latency; **shorter relevant context** can reduce processing and cost.

Retrieval-Augmented Generation

RAG supplies retrieved external information to a model without changing its weights. Choose **RAG for current or private knowledge**; fine-tuning adapts model behavior through training. Retrieval does not eliminate hallucinations.

Model Evaluation

Model evaluation compares outputs against representative tasks and defined criteria. Choose models using **quality, safety, cost, and latency**, not size alone; business KPIs establish whether technical performance delivers value.

Core services

Amazon Bedrock

Choose **managed foundation model access** to build GenAI applications without operating model-serving infrastructure, with capabilities for retrieval, customization, agents, and safety controls.

Amazon SageMaker

Choose **greater control over the ML lifecycle** for preparing data, training, tuning, evaluating, and deploying models; managed tooling reduces operational work but does not remove design decisions.

Amazon Bedrock AgentCore

Choose **managed agent infrastructure** for execution, memory, identity, tool access, and observability; its components work with different agent frameworks and foundation models.