AWS Certified Generative AI Developer - Professional Cheat Sheet

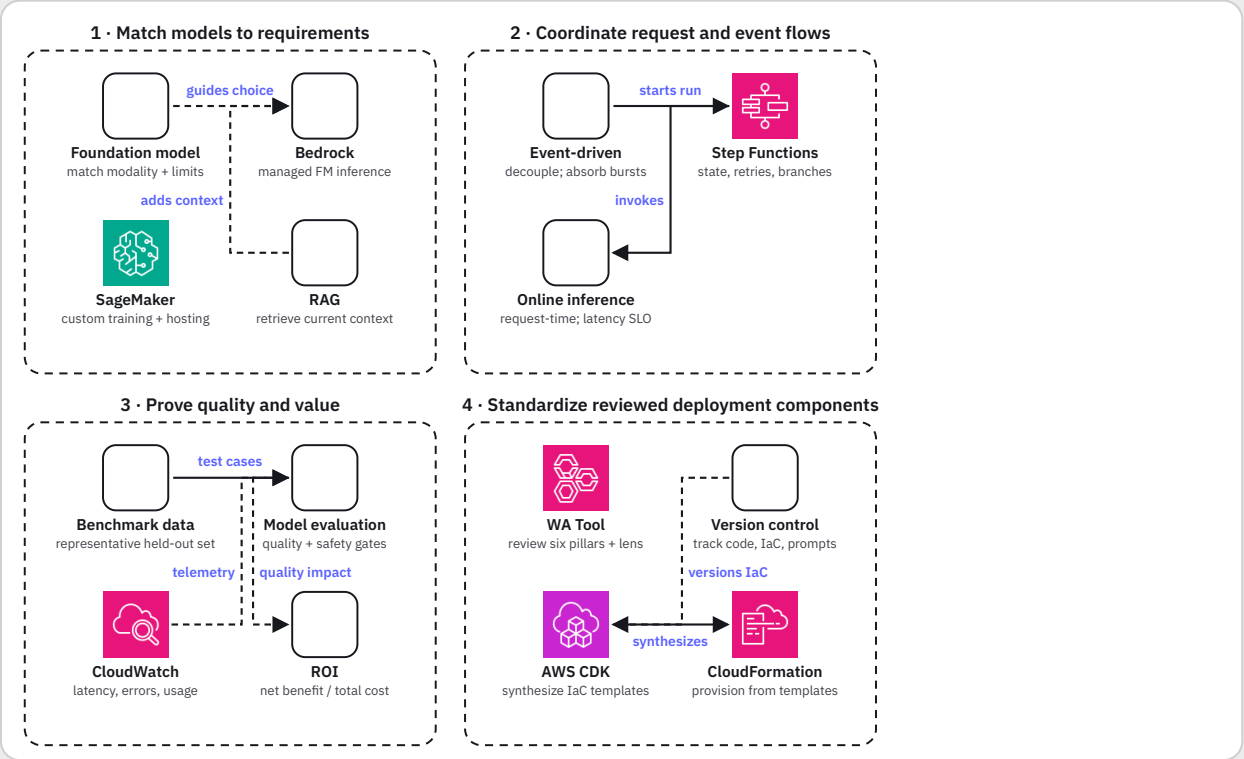
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · 31% OF THE EXAM · TASK STATEMENT 1.1

Analyze requirements and design GenAI solutions

Select, integrate, validate, standardize



1 Match models to requirements

Choose **managed FM access** with Bedrock; choose SageMaker for custom training and hosting control. Match modality, context limits, latency, and budget. RAG supplies external knowledge without retraining.

2 Coordinate request and event flows

Online inference serves current requests under latency constraints; event-driven processing decouples producers and consumers. Step Functions coordinates multi-step execution, retries, and branching.

3 Prove quality and value

A **proof of concept** uses representative tasks, human review, and baseline comparisons. Quality, safety, latency, and cost gates determine feasibility; ROI compares measurable financial benefits with total costs.

4 Standardize reviewed deployment components

The **Generative AI Lens** extends AWS Well-Architected Framework reviews; it does not deploy controls. Version reusable CDK constructs and templates for consistent, traceable infrastructure as code deployments.

Exam traps

- **RAG is not fine-tuning**: retrieval supplies external facts at inference time; fine-tuning changes model weights. Neither guarantees factual answers.
- **Best benchmark score is not best fit**: generic rankings omit workload-specific quality, safety, latency, and cost constraints. CloudWatch telemetry alone does not establish answer correctness.
- **Review is not enforcement**: the Well-Architected Tool identifies risks; CDK and CloudFormation provision declared resources. Neither a review nor IaC automatically makes a workload compliant.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

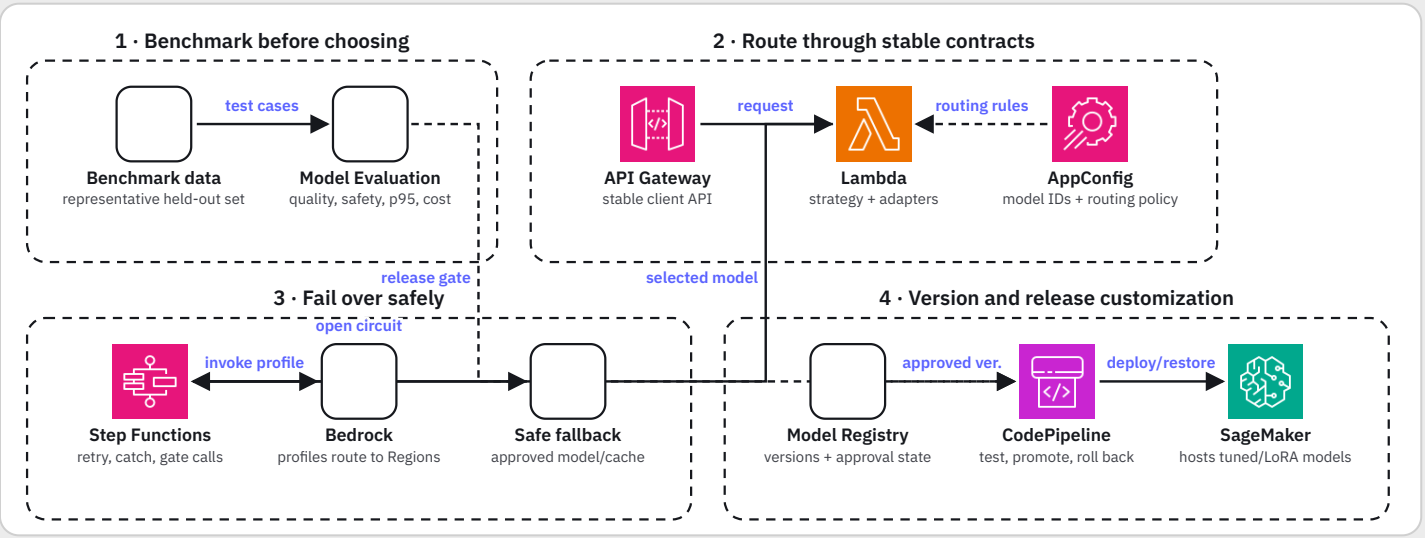
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · TASK STATEMENT 1.2

Select and configure FMs

Evaluate, route, recover, release



1 Benchmark before choosing

Choose by **workload benchmarks**, not leaderboard rank: verify modality, context limits, quality, safety, p95 latency, and token cost under expected load.

2 Route through stable contracts

Use **configuration-driven routing** to switch supported models without code changes. Strategy selects the implementation; adapters normalize requests, responses, and errors. AppConfig supports alarm-triggered rollback.

3 Fail over safely

Use **cross-Region inference** for supported models and residency-approved destinations. Implement persisted closed/open/half-open breaker state in the workflow; retries alone are not a circuit breaker.

4 Version and release customization

Use **approved model versions** for repeatable releases. LoRA reduces trainable parameters. Configure evaluation gates and rollback to preserved artifacts; retire old versions after replacements pass validation.

Exam traps

- **Configuration is not an adapter**: AppConfig can select implemented integrations; an unsupported provider still needs request, response, authentication, and error handling code.
- **Cross-Region inference is not arbitrary failover**: profiles route only supported models to defined destinations. SageMaker cross-Region endpoints and failover require explicit deployment and routing.
- **LoRA is not automatic deployment**: smaller trainable updates still require compatible serving, evaluation, and versioned artifacts. Reverting source code alone does not roll back a deployed model.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

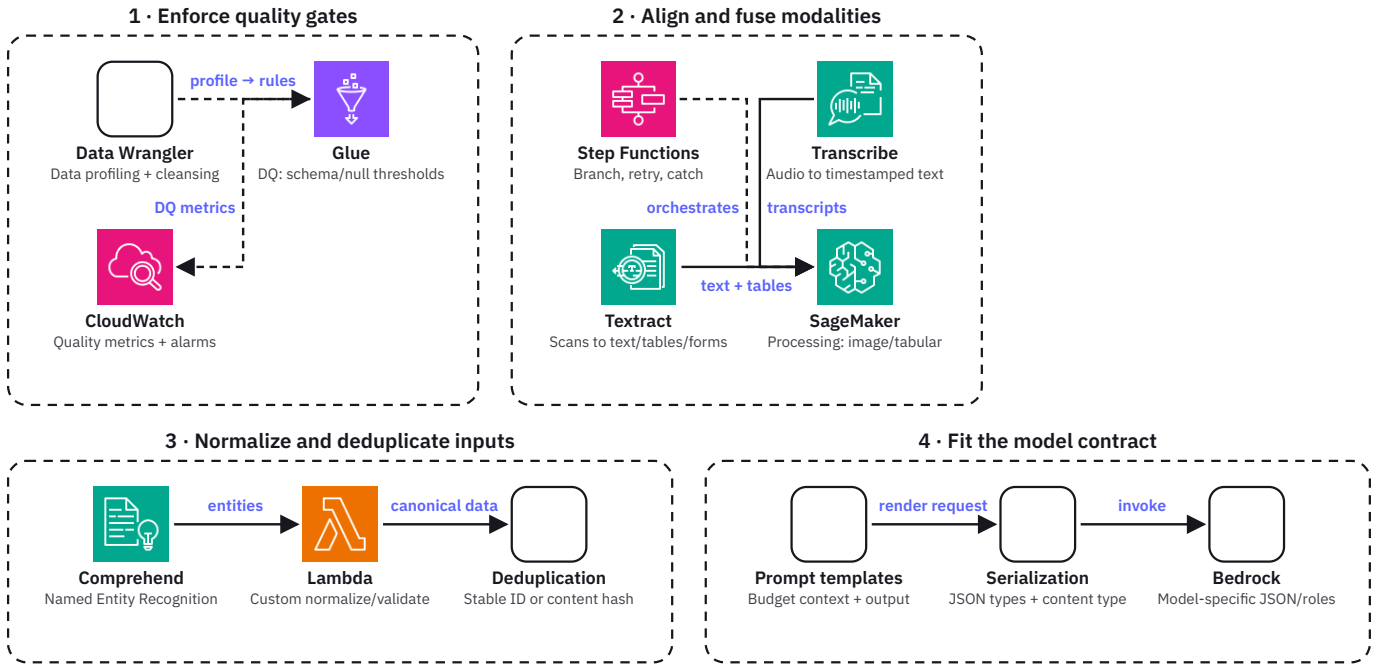
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · TASK STATEMENT 1.3

Implement data validation and processing pipelines for FM consumption

Validate, process, enrich, format



1 Enforce quality gates

Schema validation checks types and required fields at ingestion and API boundaries. Profiling informs Glue Data Quality rules; workflow logic explicitly routes failed records.

2 Align and fuse modalities

Multimodal fusion requires modality-specific preparation and alignment by timestamps or document IDs. Step Functions coordinates branches; SageMaker Processing runs custom transformations before fusion.

3 Normalize and deduplicate inputs

Text normalization must preserve meaning. Comprehend extracts, not verifies, entities; Bedrock can reformat text but may alter facts. IDs/hashes catch exact duplicates; semantic duplicates need similarity.

4 Fit the model contract

Model contract determines fields, roles, and modalities. Converse standardizes requests for supported models; InvokeModel bodies are model-specific. SageMaker endpoint serialization must match the inference handler.

Exam traps

- **Monitoring is not enforcement**: CloudWatch alarms report quality failures; they do not automatically quarantine records. Rejection, quarantine, and remediation require explicit workflow logic.
- **Extraction is modality-specific**: Transcribe converts speech to text; Textract extracts document text and structure. A Bedrock model accepts images or audio only when that model and API support them.
- **Context window** budgeting includes instructions, history, retrieved content, and output allowance. Valid JSON does not prevent overflow; chunking or summarization must preserve task-relevant context.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

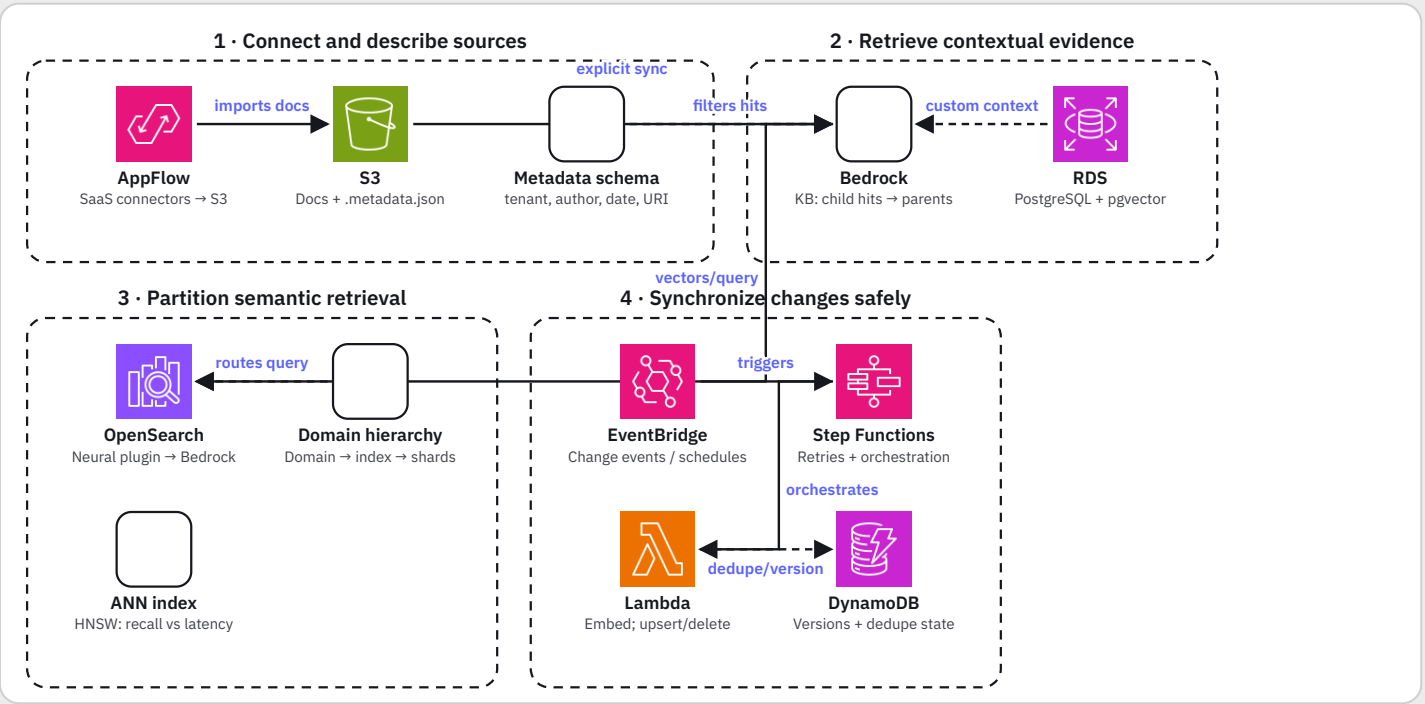
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · TASK STATEMENT 1.4

Design and implement vector store solutions

Connect, filter, partition, synchronize



1 Connect and describe sources

Metadata filters require populated, typed attributes and explicit retrieval conditions. S3 object metadata does not automatically become Knowledge Bases filter attributes; source URI/version preserves provenance.

2 Retrieve contextual evidence

Hierarchical chunking searches child chunks and returns parent context. Knowledge Bases manages ingestion and retrieval with supported stores; RDS PostgreSQL with pgvector also supports custom SQL-based RAG.

3 Partition semantic retrieval

Domain routing reduces shard fan-out; overly narrow routing misses relevant domains. Balance shards and benchmark ANN recall, latency and filtered queries. More shards add coordination overhead, not guaranteed speed.

4 Synchronize changes safely

Incremental CDC needs stable IDs, version checks and deletion handling. Validate changed content, then checkpoint after successful indexing. Knowledge Bases source changes require ingestion; retries must be idempotent.

Exam traps

- **Embedding compatibility** requires matching model/version and dimensions for document and query vectors. Changing embedding models requires re-embedding the corpus, not only changing query settings.
- **Store support is specific**: DynamoDB metadata/state does not imply native vector search. RDS PostgreSQL with pgvector supports custom retrieval; generic RDS is not a managed Knowledge Bases target.
- **Filtering is not authorization**: enforce trusted tenant/ACL constraints. Filtering only after ANN top-k can leave too few results; supported in-search filtering changes candidate selection.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

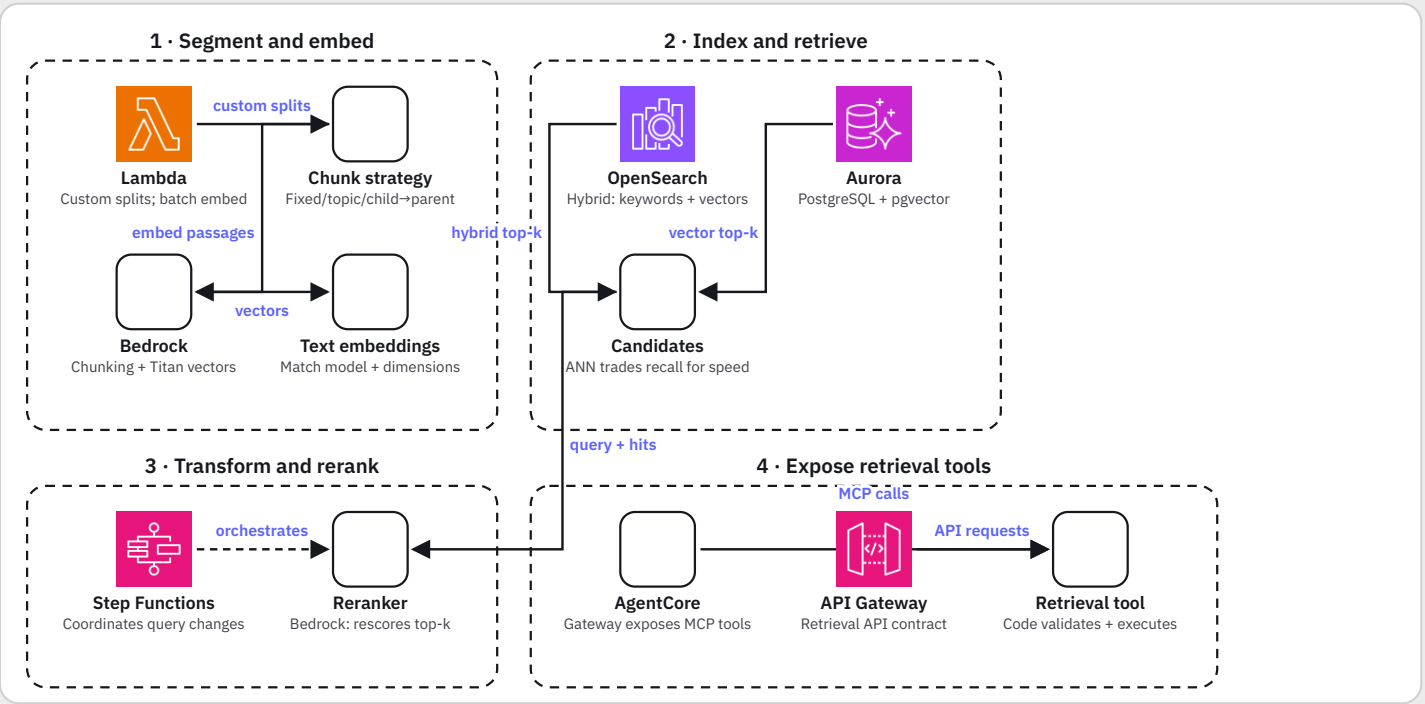
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · TASK STATEMENT 1.5

Design retrieval mechanisms for FM augmentation

Segment, embed, retrieve, refine



1 Segment and embed

Choose **chunk boundaries**: fixed tokens + overlap for predictability; semantic for topic shifts; hierarchical child matches return parents. Evaluate embedding dimensions/domain fit; batch offline work for throughput.

2 Index and retrieve

Choose **hybrid search** for exact identifiers plus meaning; Aurora PostgreSQL + pgvector fits relational data. Bedrock Knowledge Bases manages ingestion and retrieval over a supported vector store.

3 Transform and rerank

Query rewriting resolves references; expansion adds synonyms; decomposition retrieves for subquestions. Step Functions orchestrates Bedrock/Lambda transformations and retrieval; reranking selects stronger context.

4 Expose retrieval tools

MCP standardizes tool discovery and calls; API Gateway exposes retrieval APIs. Bedrock tool calling produces requests, not execution; application code authorizes calls, validates arguments, and runs vector queries.

Exam traps

- **Equal dimensions are insufficient**: different embedding models can produce incompatible spaces. Changing the embedding model requires re-embedding indexed documents, not just new query vectors.
- **ANN is approximate**, not guaranteed exact nearest-neighbor retrieval. More search effort can improve recall but increases latency; larger vectors do not automatically improve domain relevance.
- **Reranking cannot recover missing evidence**: it only reorders retrieved candidates and adds cost/latency. Improve chunking, query handling, or candidate recall when relevant passages never reach it.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

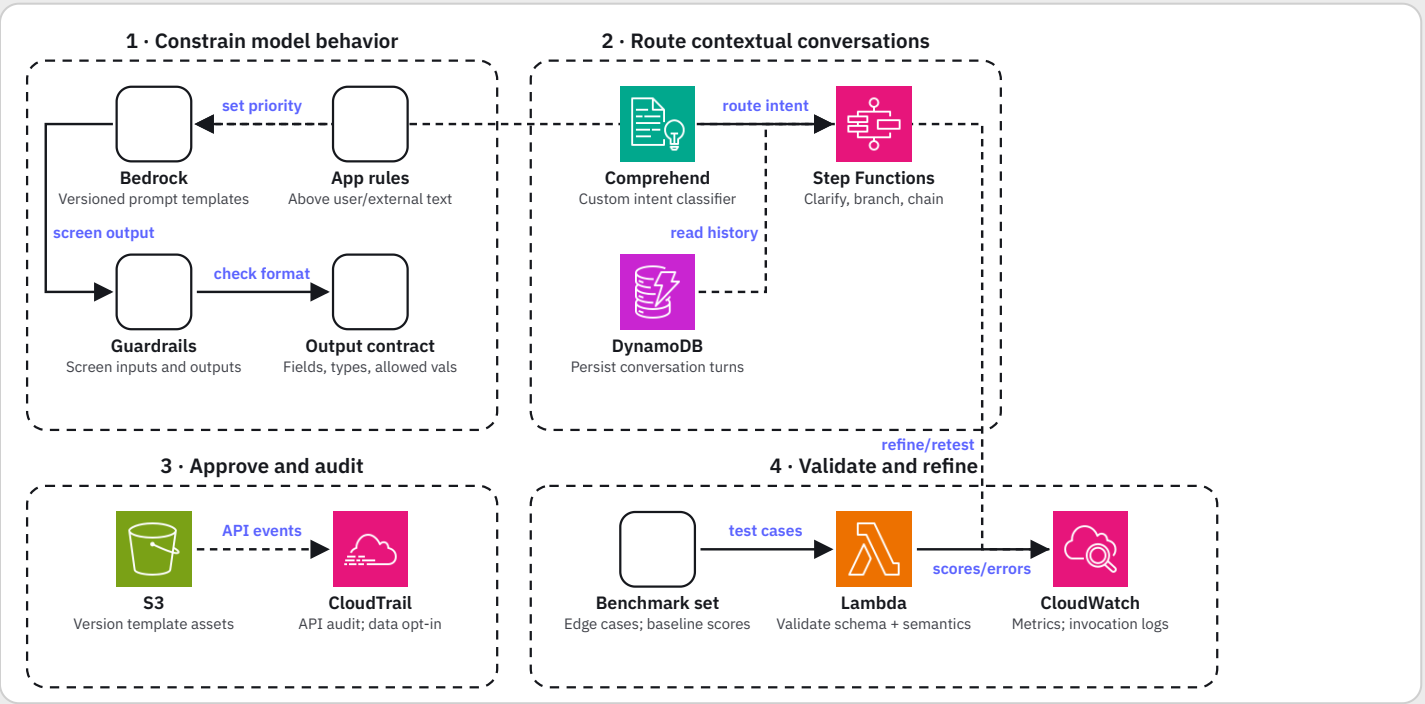
AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · TASK STATEMENT 1.6

Implement prompt engineering strategies and governance for FM interactions

Constrain, contextualize, govern, refine



1 Constrain model behavior

Use **Prompt Management** for parameterized, versioned templates with roles, context boundaries, and examples. Instruction hierarchy guides behavior; configured Guardrails independently screen content, not tool permissions.

2 Route contextual conversations

Use **clarification** for ambiguous intent and Step Functions for workflow branches. Bedrock Prompt Flows chains reusable prompts. Select or summarize stored history; input plus output must fit the context window.

3 Approve and audit

Use an **approval workflow** before promoting a pinned prompt version. Prompt Management versions and S3 Versioning preserve revisions; neither automatically approves changes. CloudTrail records supported API activity.

4 Validate and refine

Use **regression tests** after prompt or model changes; Step Functions runs cases and CloudWatch tracks scores. Human feedback guides revisions. Multistep examples can help; evaluate results, not claimed reasoning.

Exam traps

- **Prompt instructions are not a security boundary.** Hierarchy depends on model/API behavior; injection resistance also requires independent checks and least-privilege tool permissions.
- **Valid JSON is not a correct answer.** Schema validation checks structure; semantic checks test facts, allowed actions, and business rules. Guardrails do not replace either.
- **CloudTrail is not a prompt transcript.** Enable Bedrock model invocation logging to CloudWatch Logs or S3 for supported request/response capture; configure S3 object data events separately.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional

AWS Certified Generative AI Developer - Professional Cheat Sheet

AIP-C01 · Domain 1 of 5

BlowStack

DOMAIN 1 · FOUNDATION MODEL INTEGRATION, DATA MANAGEMENT, AND COMPLIANCE · WHAT TO REMEMBER

Concepts and services of Foundation Model Integration, Data Management, and Compliance

Concepts to know

Retrieval-Augmented Generation

RAG supplies external evidence at inference time. Choose **retrieval for current or private knowledge**, rather than fine-tuning to memorize facts; chunking, metadata filters, hybrid search, and reranking determine evidence quality.

Model Evaluation

Evaluation compares models and application changes on representative benchmarks. Choose using **quality, safety, latency, and cost together**; test the complete retrieval and prompt pipeline rather than relying on generic model rankings.

AI Guardrails

Guardrails screen inputs and outputs against configured safety policies. Use **independent controls rather than prompt instructions alone**; authorization, retrieval permissions, and output validation remain application responsibilities.

Core services

Amazon Bedrock

Choose **managed foundation model integration** for model access, Knowledge Bases retrieval, prompt versioning, and Guardrails; use supported inference profiles for cross-Region routing.

Amazon SageMaker

Choose **custom training and deployment control** for fine-tuned models, managed endpoints, and processing jobs; use Model Registry for versioned model packages and approval status.

Amazon OpenSearch Service

Choose **hybrid keyword and vector retrieval** when exact terms and semantic similarity both matter; configure metadata filters, indexes, and shards for retrieval precision and scale.

Get practice exams and lessons for AWS Certified Generative AI Developer - Professional