AWS Certified Machine Learning Engineer - Associate Cheat Sheet

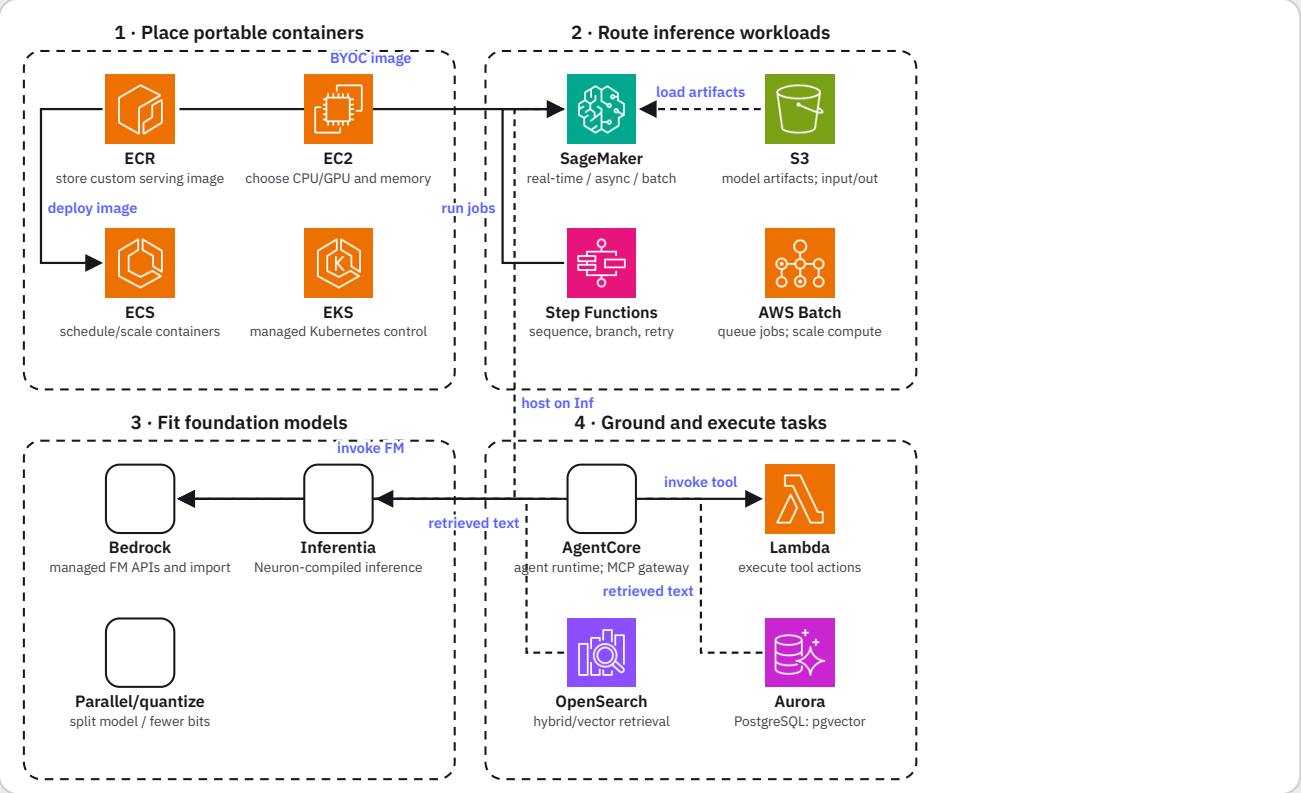
MLA-C02 (beta) · Domain 3 of 4

BlowStack

DOMAIN 3 · DEPLOYMENT AND ORCHESTRATION OF ML AND AI WORKFLOWS · 24% OF THE EXAM · TASK STATEMENT 3.1

Manage deployment infrastructure for ML and AI model types

Place, route, size, ground



1 Place portable containers

Choose **Kubernetes** for EKS; ECS for managed container scheduling; EC2 for host control. Right-size instance types from measured CPU/GPU, memory and throughput needs; containers do not provision compute themselves.

2 Route inference workloads

Choose **asynchronous** for queued, long-running requests with delayed S3 results; real-time for immediate responses; Batch Transform for offline datasets. Step Functions coordinates workflows; AWS Batch schedules compute jobs.

3 Fit foundation models

Choose **hosting control** with SageMaker for custom/open-source FMs; Bedrock manages serving. Split models across devices when memory is insufficient; quantize to reduce memory, then validate accuracy and latency.

4 Ground and execute tasks

RAG retrieves candidates, reranks, then augments the prompt; it does not retrain the FM. Agents invoke authorized tools; MCP standardizes tool discovery/access, not agent-to-agent messaging.

Exam traps

- **Multi-model** endpoints dynamically load models onto shared serving resources; serial multi-container inference pipelines chain processing stages. Multiple models do not imply a pipeline.
- **Custom Model Import** accepts supported model architectures and formats, not arbitrary serving containers. External models can use SageMaker with an ECR inference image and S3 artifacts.
- **Reranking** reorders retrieved candidates; it cannot recover documents absent from that set. Increase retrieval recall first when relevant evidence is missing; larger top-k also adds cost and latency.

Get practice exams and lessons for AWS Certified Machine Learning Engineer - Associate

AWS Certified Machine Learning Engineer - Associate Cheat Sheet

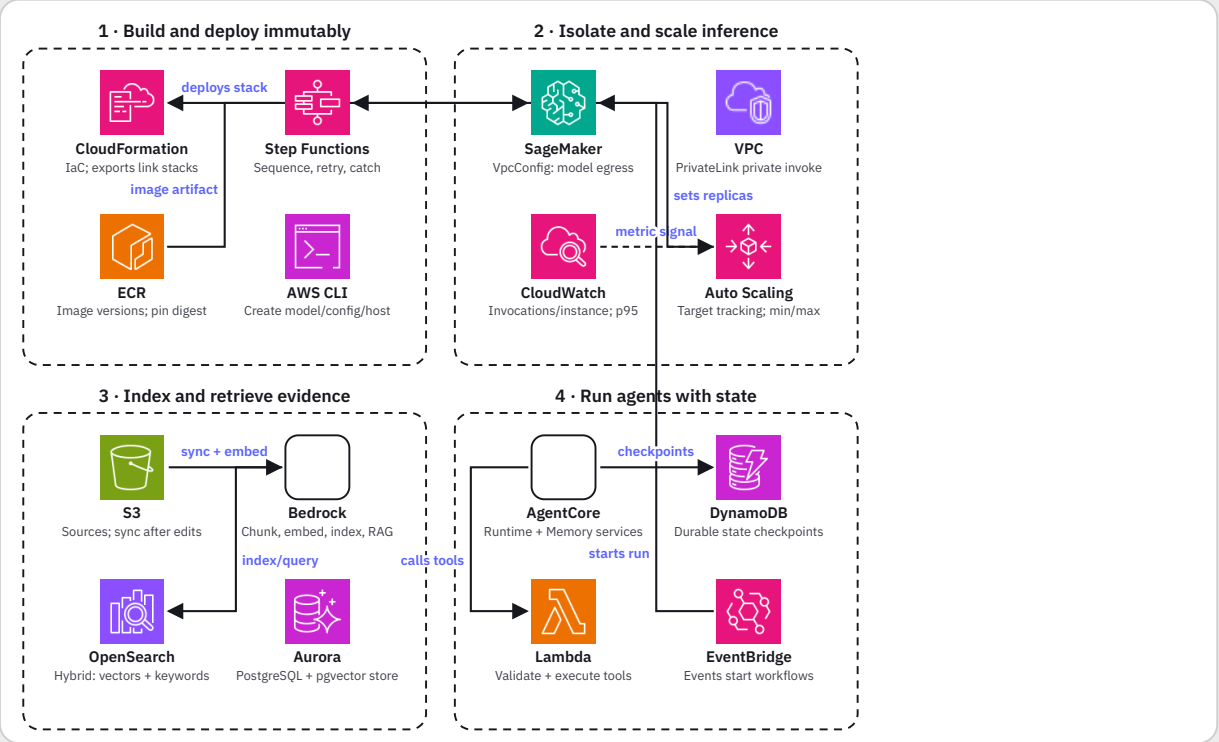
MLA-C02 (beta) · Domain 3 of 4

BlowStack

DOMAIN 3 · DEPLOYMENT AND ORCHESTRATION OF ML AND AI WORKFLOWS · TASK STATEMENT 3.2

Provision and configure resources for ML and AI workloads based on existing architecture and requirements

Provision, isolate, scale, retrieve



1 Build and deploy immutably

Immutable: AWS CodeBuild can extend AWS Deep Learning Containers and push tested images to ECR. Deploy a pinned digest; replace rather than patch containers. CLI, Boto3, and SageMaker Python SDK automate deployment.

2 Isolate and scale inference

Scale target: SageMaker uses Application Auto Scaling. Use invocations per instance or concurrency; track GPU utilization and latency. Right-size GPU memory; allow model-loading warmup before scaling.

3 Index and retrieve evidence

Retrieval: preserve context when chunking; match embedding dimensions to the index. Choose a supported vector store. Semantic search finds meaning; supported hybrid search adds exact terms. RetrieveAndGenerate adds generation.

4 Run agents with state

Recovery: persist tenant/session-scoped checkpoints in DynamoDB; conversation memory supplies context, not workflow progress. Validate and authorize tool calls. Step Functions coordinates steps, retries, and agent handoffs.

Exam traps

- **Predictable demand** favors supported Bedrock Provisioned Throughput; variable usage favors on-demand. Savings Plans discount usage, not capacity; Spot requires interruption tolerance.
- **VpcConfig is not private invocation**: it connects model containers to VPC resources. AWS PrivateLink provides private runtime access; network isolation separately blocks container network calls.
- **Memory is not a checkpoint**: conversation history does not automatically restore agent execution. Persist progress explicitly; cache-only state can disappear on eviction or expiry.

Get practice exams and lessons for AWS Certified Machine Learning Engineer - Associate

AWS Certified Machine Learning Engineer - Associate Cheat Sheet

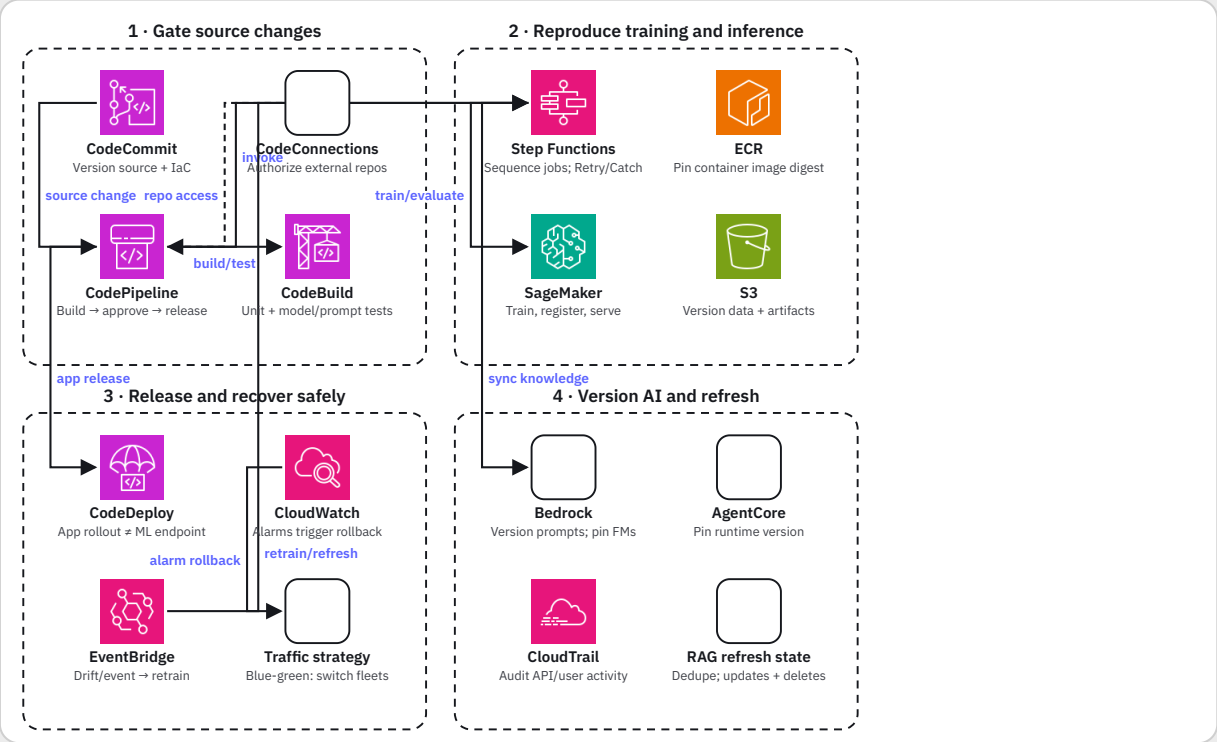
MLA-C02 (beta) · Domain 3 of 4

BlowStack

DOMAIN 3 · DEPLOYMENT AND ORCHESTRATION OF ML AND AI WORKFLOWS · TASK STATEMENT 3.3

Implement automated orchestration and continuous integration and continuous delivery (CI/CD) pipelines for MLOps and AI workloads

Test, version, deploy, refresh



1 Gate source changes

Use **AWS CodeConnections** for external repositories; AWS CodeCommit is an alternative source. Source failures: authorization/IAM. Build failures: buildspec, logs, artifact paths. Failed tests block release.

2 Reproduce training and inference

Use **Model Registry** for approved versions; MLflow tracks runs. Pin data, code, parameters and image digest for provenance. Distributed training uses multiple workers; batch jobs differ from persistent real-time endpoints.

3 Release and recover safely

Use **Amazon SageMaker deployment guardrails** for endpoint blue-green/canary releases and configured alarm-driven rollback. AWS CodeDeploy handles supported application targets; it does not deploy managed ML endpoints.

4 Version AI and refresh

Use **incremental RAG refresh**, not fine-tuning, for changed documents. Pin prompt, agent and fine-tuned model IDs in IaC; benchmark prompt revisions before release. Track document revisions for idempotent retries.

Exam traps

- **Canary exposes users** to a small share of candidate responses; shadow testing copies requests but discards candidate responses. Blue-green switches between separate fleets.
- **Retraining is not promotion**: champion/challenger evaluation compares the candidate with the approved baseline. Registering or approving a version does not itself deploy it.
- **LLM-as-a-Judge is not ground truth**: use fixed benchmark datasets, explicit rubrics and human calibration. Unit tests alone cannot establish quality, safety or RAG groundedness.

Get practice exams and lessons for AWS Certified Machine Learning Engineer - Associate

AWS Certified Machine Learning Engineer - Associate Cheat Sheet

MLA-C02 (beta) · Domain 3 of 4

BlowStack

DOMAIN 3 · DEPLOYMENT AND ORCHESTRATION OF ML AND AI WORKFLOWS · WHAT TO REMEMBER

Concepts and services of Deployment and Orchestration of ML and AI Workflows

Concepts to know

Real-Time Inference

Real-time inference returns predictions during an application request. Choose it for **low-latency responses**; asynchronous inference suits queued, longer-running requests, while batch inference suits offline datasets.

CI/CD

CI/CD automates validation and release of code, models, prompts, and infrastructure. Use **evaluation gates before promotion**, version deployed artifacts, and configure monitored rollout and rollback rather than automatically promoting every retrained model.

Infrastructure as code (IaC)

IaC defines infrastructure in versioned templates for repeatable provisioning. Choose **declarative resource deployment** for consistent environments; workflow orchestration coordinates jobs, dependencies, retries, and approvals.

Core services



Amazon SageMaker

Choose **managed custom-model training and hosting**: real-time endpoints for interactive predictions, asynchronous inference for queued requests, and batch transform for offline scoring.

Amazon Bedrock

Choose **managed foundation model access** without operating serving infrastructure; use knowledge bases for RAG, agents for tool-driven tasks, and Prompt Management for versioned prompts.



AWS Step Functions

Choose **stateful workflow orchestration** across AWS services when ML or AI pipelines need sequencing, branching, retries, and error handling; it coordinates work rather than hosting models.

Get practice exams and lessons for AWS Certified Machine Learning Engineer - Associate