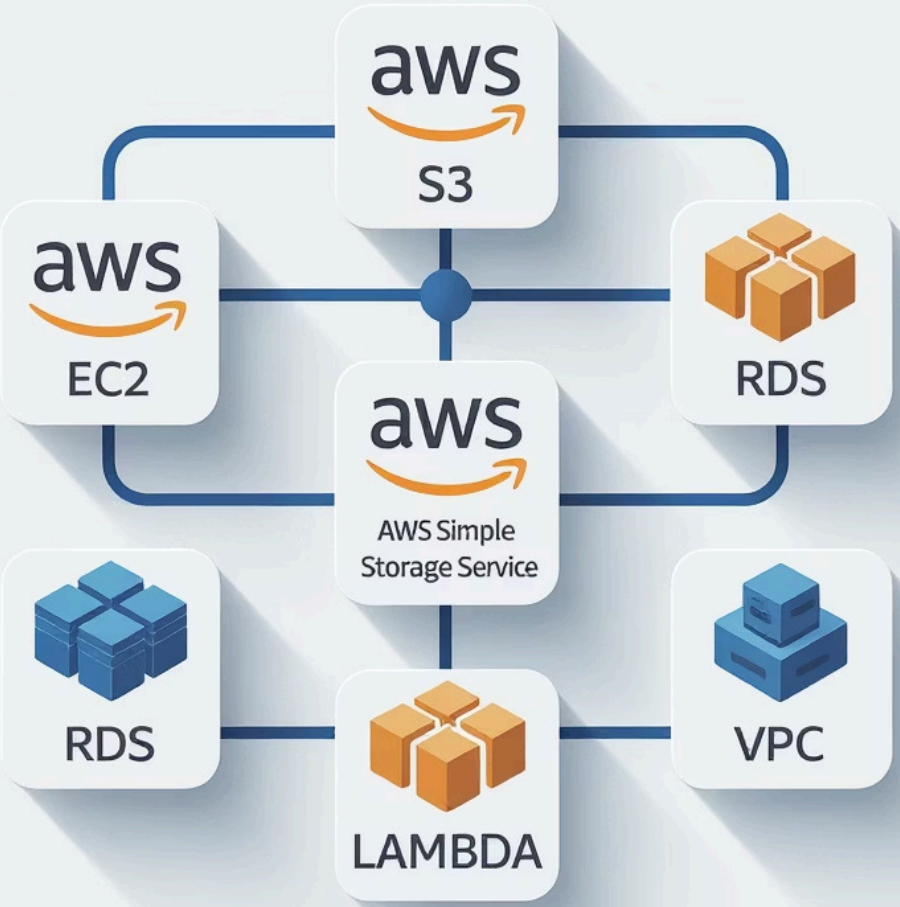
Building on AWS Bedrock

Master the fundamentals of building agentic AI applications using AWS Bedrock's powerful suite of tools. This section covers Foundation Models, intelligent Agents, protective Guardrails, and Action Groups that enable sophisticated AI workflows for business applications.



Setup Essentials

Before diving into building with AWS Bedrock, proper setup is crucial for success. These foundational steps ensure smooth development and deployment of your agentic AI applications.

IAM User & CLI

Create dedicated IAM user with Bedrock permissions. Configure AWS CLI with access keys for seamless command-line operations and programmatic access to services.

Model Access

Enable Model Access in the Bedrock console for your desired foundation models. This step is essential - without it, your applications won't be able to invoke any models.

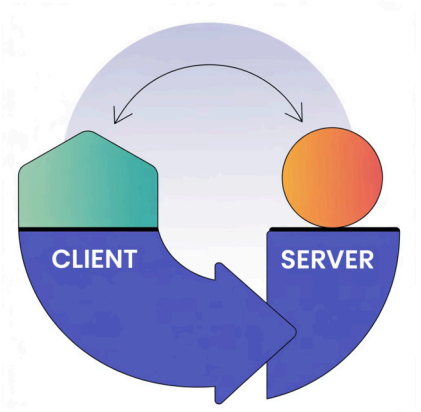
Regions Matter

Choose your AWS region carefully. Not all Bedrock features and models are available in every region. US East and West typically offer the most comprehensive support.

Invoke Models vs Agents

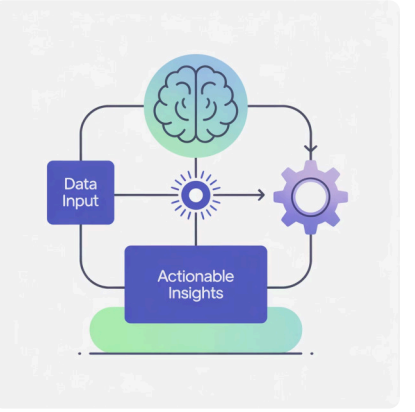
Understanding the distinction between direct model invocation and agent-based interactions is fundamental to architecting effective AI solutions. Each approach serves different use cases and complexity levels.

Direct Model Invocation



Use the **bedrock-runtime** client for straightforward model interactions. Perfect for simple question-answering, text generation, and basic AI tasks where you need direct control over inputs and outputs. This approach provides immediate responses, a simple request-response pattern, and full control over prompts.

Agent-Based Approach



Leverage **bedrock-agent-runtime** for complex workflows requiring planning, tool usage, and multi-step reasoning. Agents can break down complex tasks and use multiple tools autonomously. This includes an Agent **Prepare** phase for setup, the ability to create an **Alias** for version management, and support for multi-turn conversations with memory.

Action Groups & Lambda Integration

Action Groups are the bridge between your AI agents and real-world functionality. They enable agents to perform specific tasks by calling Lambda functions, transforming conversational AI into actionable business tools.

01

Define Parameters

Specify input parameters your Lambda function expects. Clear parameter definitions help the agent understand when and how to use each action group effectively.

02

Function Logic

Build Lambda functions that process agent requests and return structured responses. Keep functions focused on single responsibilities for better reliability and debugging.

03

Recommendation Pattern

Implement the recommendation → explanation pattern. First provide the action or recommendation, then explain the reasoning behind it for better user understanding.

This approach creates transparent, trustworthy AI interactions that users can understand and validate.

Guardrails & User Interface

Protect your AI applications and users with AWS Bedrock's comprehensive safety features, while creating intuitive interfaces that make agentic AI accessible to everyone.

Essential Guardrails

PII Masking

Automatically detect and mask personally identifiable information to ensure privacy compliance and data protection in all AI interactions.

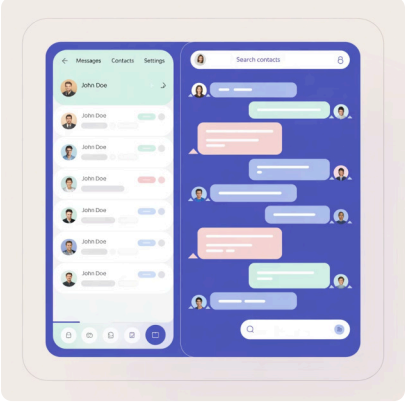
Denied Topics

Configure topic filters to prevent discussions of inappropriate or sensitive subjects, maintaining professional and safe conversations.

Harm Filters

Deploy content filters that detect and block potentially harmful outputs, ensuring your AI applications remain trustworthy and safe.

Streamlit Chat Interface



Build user-friendly chat interfaces using Streamlit's simple framework. Implement session IDs to maintain conversation context and provide seamless multi-turn interactions.

This combination delivers enterprise-grade AI safety with consumer-grade usability.