Date: 9.15.2025



Copyright @ 2024 PibyThree.com All Rights Reserved

AWS Bedrock & Agentic AI

Contents

[1. Introduction 2](#_Toc208842594)

[2. Agentic AI in Bedrock - Amazon Bedrock AgentCore 2](#_Toc208842595)

[3. Bedrock AgentCore Services 2](#_Toc208842596)

[4. Bedrock AgentCore Pricing 3](#_Toc208842597)

[5. Amazon Bedrock Agents 4](#_Toc208842598)

[6. Strands Agent (SDK) 5](#_Toc208842599)

[7. Amazon Bedrock Agents vs Amazon Bedrock AgentCore 6](#_Toc208842600)

[8. Conclusion 6](#_Toc208842601)

# 1. Introduction

**AWS Bedrock:** A secure, flexible service for building and deploying generative AI applications that provides access to top foundation models, tools for fine-tuning, safety, optimization, and integration with knowledge bases, enabling fast transition from testing to real-world use.

**Agentic AI:** Autonomous systems that plan, act, and adapt with minimal input. Using tools and data, they tackle complex tasks and boost efficiency—marking a shift as transformative as the internet or PC.

# 2. Agentic AI in Bedrock - Amazon Bedrock AgentCore

* Enables secure, scalable deployment of advanced AI agents
* Provides infrastructure tailored for dynamic workloads
* Offers tools to enhance agent capabilities
* Includes controls for real-world deployment
* Compatible with any framework (e.g., CrewAI, LangGraph, LlamaIndex, Strands Agents)
* Works with any foundation model, inside or outside Amazon Bedrock
* Removes the need to build specialized agent infrastructure
* Speeds up AI agent production

# 3. Bedrock AgentCore Services

|  |  |
| --- | --- |
| **Service** | **Description** |
| Runtime | Secure, serverless environment for deploying and scaling dynamic AI agents and tools. |
| Browser | Cloud-based, secure browser runtime enabling agents to interact with websites at scale. |
| Code Interpreter | Allows agents to write and run code securely in sandboxed environments for complex tasks. |
| Gateway | Secure tool discovery, transforming APIs, Lambda functions, and services into agent-compatible tools easily. |
| Identity | Simplifies identity and access management, enabling secure access to AWS and third-party services. |
| Memory | Manages context-aware agent memory with full developer control, removing complex infrastructure needs. |
| Observability | Provides visibility for tracing, debugging, and monitoring agent workflows in production environments. |

Here is the [Amazon Bedrock AgentCore (Dev Guide)](https://docs.aws.amazon.com/bedrock-agentcore/latest/devguide/what-is-bedrock-agentcore.html) and [Amazon Bedrock AgentCore (Overview)](https://aws.amazon.com/bedrock/agentcore/)

# 4. Bedrock AgentCore Pricing

All services are priced on a consumption-based pricing scheme.

|  |  |  |
| --- | --- | --- |
| **Service** | **Resource / Metric** | **Price** |
| Runtime | CPU | $0.0895 / ~₹7.90 per vCPU hour |
| Memory | $0.00945 / ~₹0.83 per GB hour |
| Browser Tool | CPU | $0.0895 / ~₹7.90 per vCPU hour |
| Memory | $0.00945 / ~₹0.83 per GB hour |
| Code Interpreter | CPU | $0.0895 / ~₹7.90 per vCPU hour |
| Memory | $0.00945 / ~₹0.83 per GB hour |
| Gateway | API Invocations  (ListTools, InvokeTool) | $0.005 / ~₹0.44 per 1,000 invocations |
| Search API | $0.025 / ~₹2.21 per 1,000 invocations |
| Tool Indexing | $0.02 / ~₹1.76 per 100 tools indexed per month |
| Identity | Token/API key requests (non-AWS resources) | $0.010 / ~₹0.88 per 1,000 requests |
| *Free via AgentCore Runtime or Gateway* |
| Memory | Short-Term Memory | $0.25 / ~₹22.05 per 1,000 new events |
| Long-Term Memory Storage (built-in strategy) | $0.75 / ~₹66.16 per 1,000 memories/month |
| Long-Term Memory Storage (custom strategy) | $0.25 / ~₹22.05 per 1,000 memories/month |
| Long-Term Memory Retrieval | $0.50 / ~₹44.11 per 1,000 retrievals |
| Observability | Logs, metrics, traces | Charged per Amazon CloudWatch pricing |

# 5. Amazon Bedrock Agents

Amazon Bedrock Agents is a fully managed service that builds agents using foundation models, APIs, and company data. It can:

* Break down complex user requests into steps
* Use retrieval‑augmented generation (RAG) to pull in external info
* Retain memory for smoother multi‑step interactions
* Automatically call APIs to take actions
* Support multi‑agent collaboration for tackling complex tasks
* Provide built‑in security and reliability via Guardrails

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Multi-Agent Collaboration | Multiple specialized agents work together under a supervisor to handle complex tasks. |
| Retrieval-Augmented Generation (RAG) | Agents access company data to enrich responses with accurate, real-time info. |
| Orchestrate and Execute | Agents break down tasks, call APIs, and automate workflows using FM reasoning. |
| Memory Retention | Agents retain context across interactions for personalized, seamless experiences. |
| Code Interpretation | Agents generate and execute code securely for analytics, math, and data tasks. |

[AI Agents – Amazon Bedrock Agents – AWS](https://aws.amazon.com/bedrock/agents/)

# 6. Strands Agent (SDK)

**Strands Agents:** A lightweight, model-driven SDK for building AI agents with minimal code that supports everything from simple assistants to complex autonomous workflows and scales from local development to production.

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Model Agnostic | Supports Amazon Bedrock, Anthropic, LiteLLM, Llama, Ollama, OpenAI, Writer, and custom providers |
| Flexible & Lightweight | Simple, customizable agent loop |
| Multi-Agent & Streaming | Build autonomous, collaborative agents with streaming support |
| Built-in MCP Support | Integrates Model Context Protocol servers with thousands of tools |
| Python-Based Tools | Easily create tools using Python decorators |
| Hot Reloading | Auto-reload tools from the ./tools/ directory |
| Multiple Model Providers | Includes Amazon Bedrock, Ollama, LlamaAPI, and more |
| Pre-built Tools | Optional strands-agents-tools package for quick experimentation |

[Strands Agents Docs](https://strandsagents.com/latest/documentation/docs/)

# 7. Amazon Bedrock Agents vs Amazon Bedrock AgentCore

|  |  |  |
| --- | --- | --- |
| **Feature** | **Amazon Bedrock Agents** | **Amazon Bedrock AgentCore** |
| Interface | GUI-based setup via AWS Console, minimal coding required. | Web-based console, extensive coding and infrastructure management required. |
| Agent Creation | Build agents using templates and natural language instructions. | Requires defining agent logic, containerization, and deployment using SDK, CLI, or APIs. |
| Customization | Limited customization, suitable for standard workflows. | Full control over agent behavior and integration with external frameworks. |
| Tool Integration | Limited, primarily through pre-configured connectors. | Extensive via supports custom APIs, Lambda functions, and third-party services. |
| Memory Management | Basic memory retention for context-aware interactions. | Advanced memory management with short-term and long-term storage options. |
| Code Execution | Supports dynamic code generation and execution in a secure environment. | Includes a Code Interpreter tool for executing code in isolated sandbox environments. |
| Browser Interaction | Limited browser interaction capabilities. | Includes a secure, cloud-based browser runtime for web-based workflows. |
| Observability | Basic monitoring through AWS CloudWatch. | Comprehensive observability with detailed dashboards and telemetry support. |
| Deployment | Simplified deployment process via AWS Console. | Requires manual setup and configuration for deployment and scaling. |

# 8. Conclusion

* **Strands Agents SDK**: Ideal for developing and prototyping AI agents locally with minimal code and high flexibility.
* **Amazon Bedrock Agent Builder**: A no-code/low-code visual tool to quickly create and customize AI agents via a GUI, enabling faster development without deep coding.
* **Amazon Bedrock AgentCore**: Built for deploying, scaling, and managing AI agents securely and reliably in production environments.