Building & Scaling Secure AI Agents:

Bedrock, Bedrock Agentcore & AI Agents

aws PartnerEquip

Agenda

01 AI Agent adoption momentum & Challenges

02 Bedrock AgentCore

03 Bedrock Agents

04 Differences

05 When to use which platform

06 Q&A

Implementing an AI Agent is complex

But open source frameworks make it easy to build agents and create proof of concepts (PoCs)

Context

Tools and tools execution

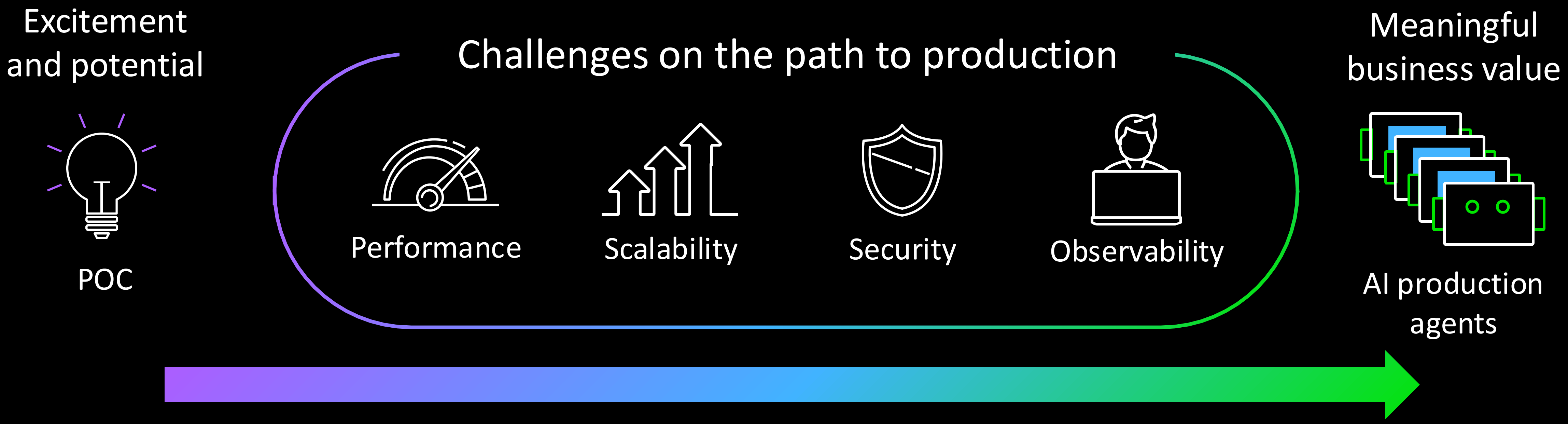
Multi-agent collaboration

Orchestrating between actions

Functionality available

Strands Agents LangChain LangGraph crewai

The prototype to production “chasm”



Excitement and potential

POC

Challenges on the path to production

Performance

Scalability

Security

Observability

Meaningful business value

AI production agents

NEW

Amazon Bedrock
AgentCore

Deploy and operate highly effective agents securely, at scale using any framework and model

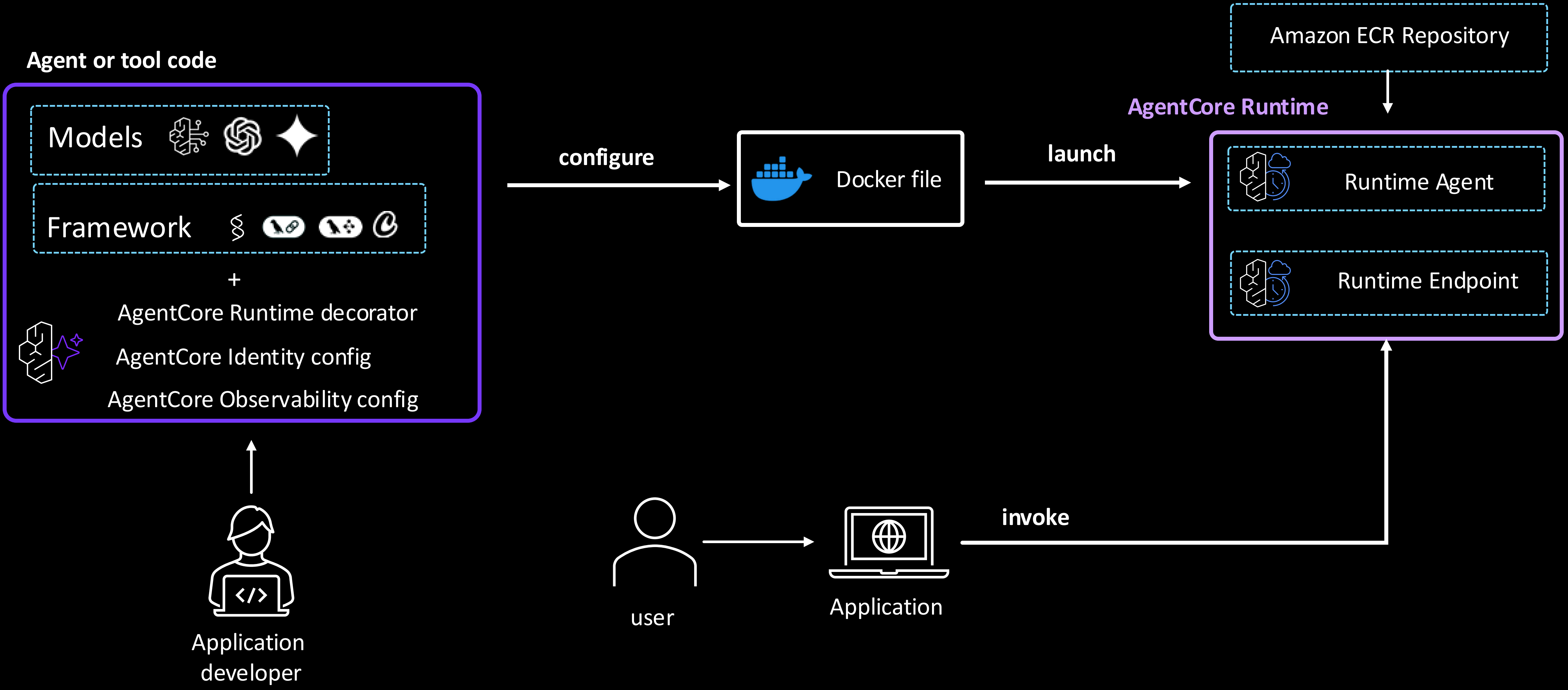
PREVIEW

What is Bedrock AgentCore

In private preview; GA likely to be by re:Invent

- Foundational infrastructure & services designed for deploying, operating, and scaling AI agents securely
- It's not a singular, managed service for building agents from scratch; Instead it provides the underlying building blocks that developers can use with any framework, be it open-source or custom-built
- 7 building blocks
  - Runtime: A serverless environment for executing agents, ensuring session isolation and supporting both low-latency and long-running tasks.
  - Gateway: A tool to transform existing APIs into interfaces that agents can readily consume.
  - Memory: Services for managing both short-term and long-term memory for agents, enabling them to maintain context across interactions.
  - Identity: A secure way to manage agent identity and access to AWS resources and third-party tools.
  - Observability: Dashboards and tools to monitor and debug agent behavior.
  - Code Interpreter: A secure environment for agents to execute code in multiple languages.
  - Browser: A managed web browser for agents to perform web automation tasks.

Secure and scalable runtime for agents and tools



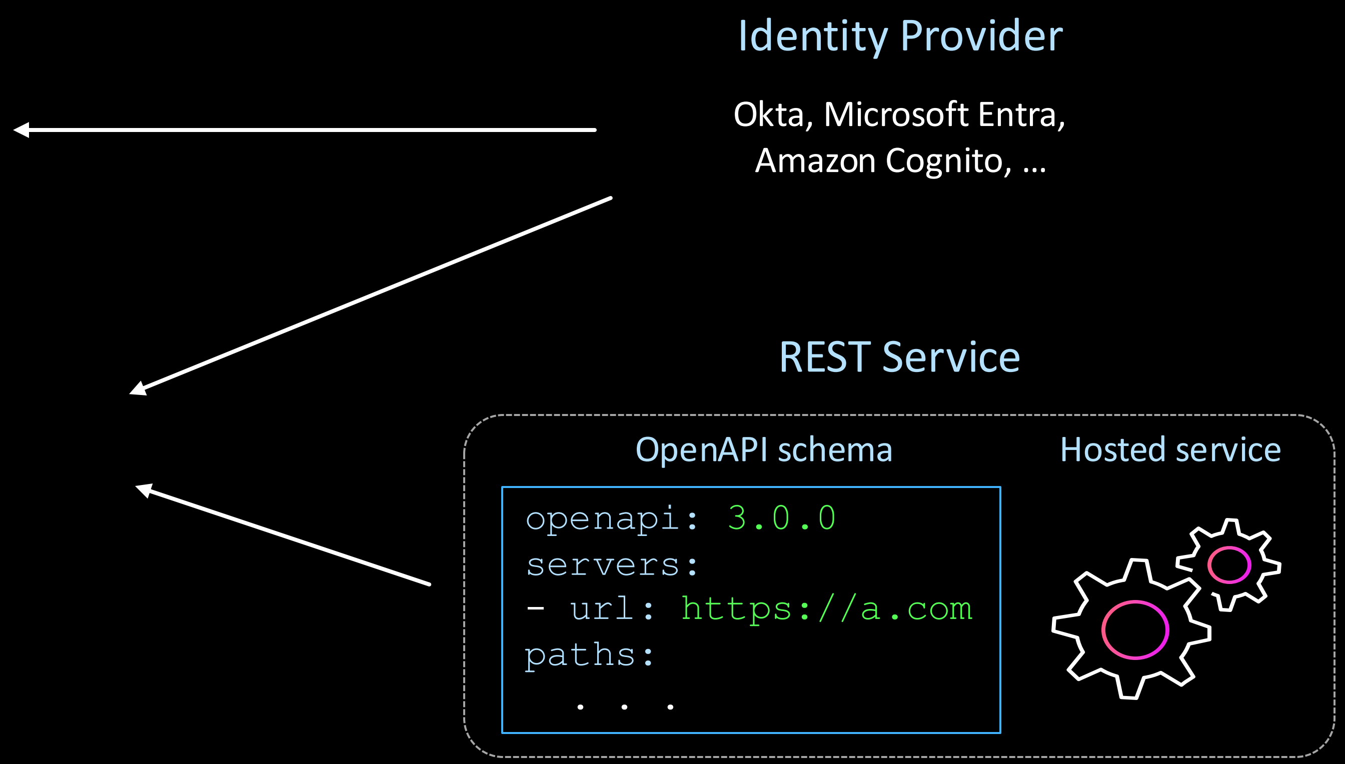
How it works – creating and using a gateway

Exposing MCP tools for an existing REST service

1 **Create Gateway** –
name, description,
inbound auth configuration

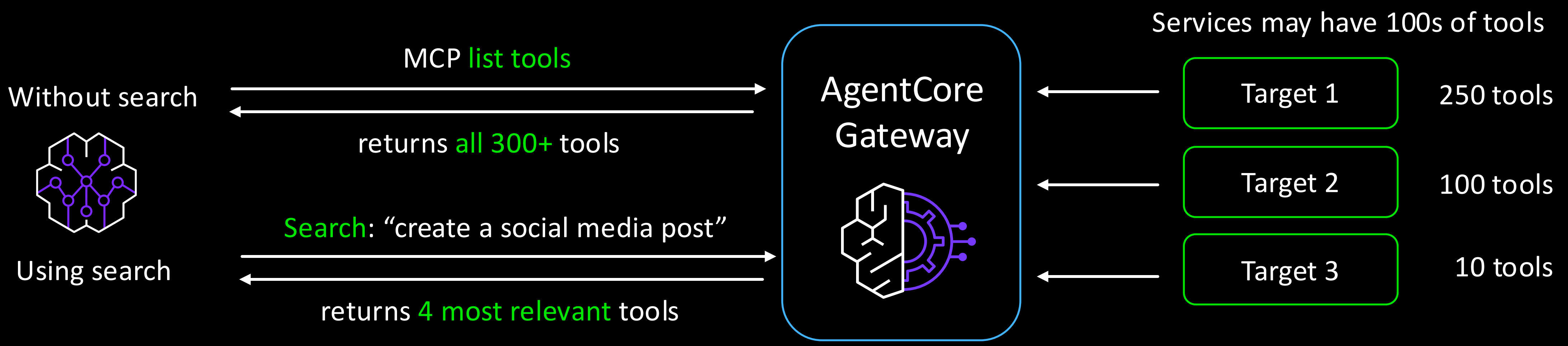
2 **Create Gateway Target** –
name, description, API spec,
credential provider configuration

3 **List, invoke, and search tools** –
using any agent framework,
MCP Inspector, or coding IDE



```
openapi: 3.0.0
servers:
- url: https://a.com
paths:
  . . .
```

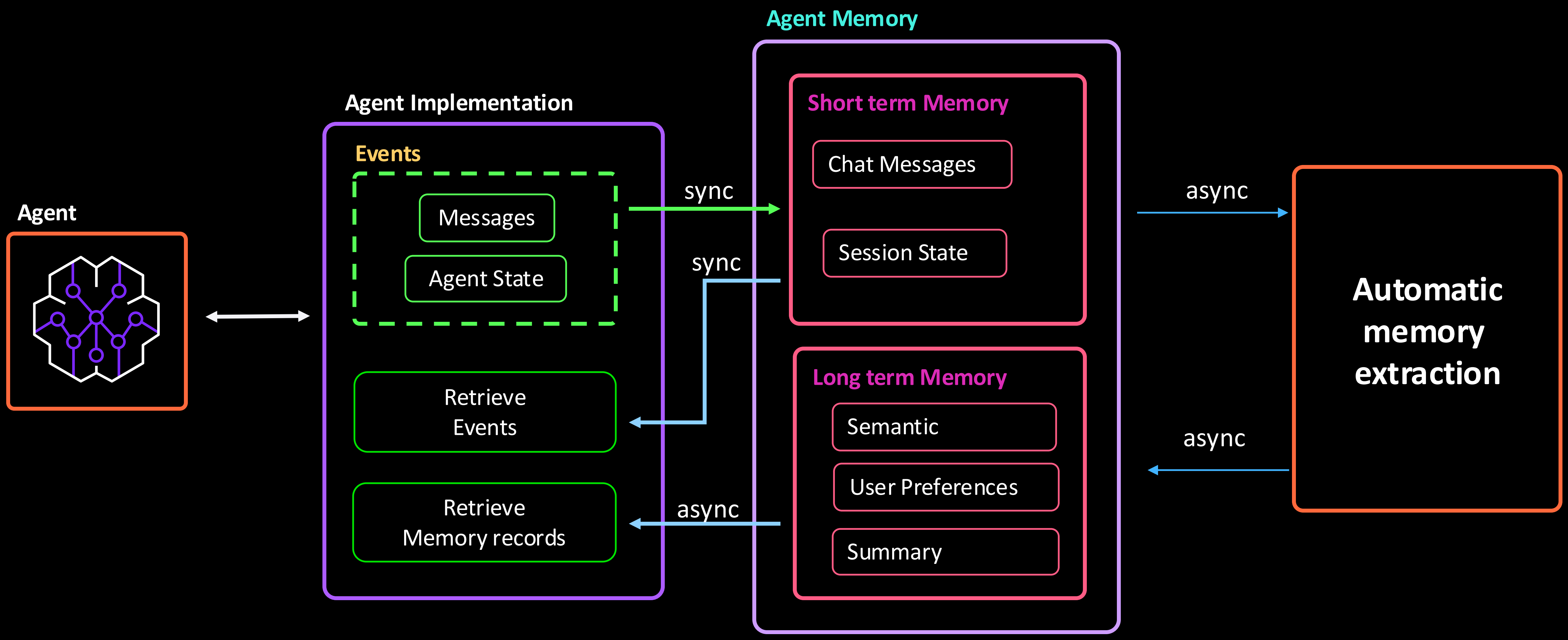
AgentCore Gateway semantic search



Benefits

- AgentCore Gateway automatically indexes tools, and gives serverless semantic search
- Reduces context passed to the agent’s LLM, improving accuracy, speed, and cost
- Lets agent focus on tools relevant for a given task

Short-term and long-term memory capabilities



AgentCore Identity

Secure access to agents and tools

- Distinct identities for secure agent operations at scale
- Authentication with enterprise identity providers
- Secure credential management for external service access and integration

Minimized consent fatigue

- Reduces need for repeated authorization
- Streamlines authentication flows
- Simplifies user experiencer for all agent-powered interactions

Accelerated AI agent development

- Preserves existing identity systems such as Okta, Microsoft Entra ID, or Amazon Cognito
- Inbound and outbound authentication

AgentCore Observability

Complete visibility into agent workflows to trace, debug, and monitor AI agents' performance

Maintain quality and trust

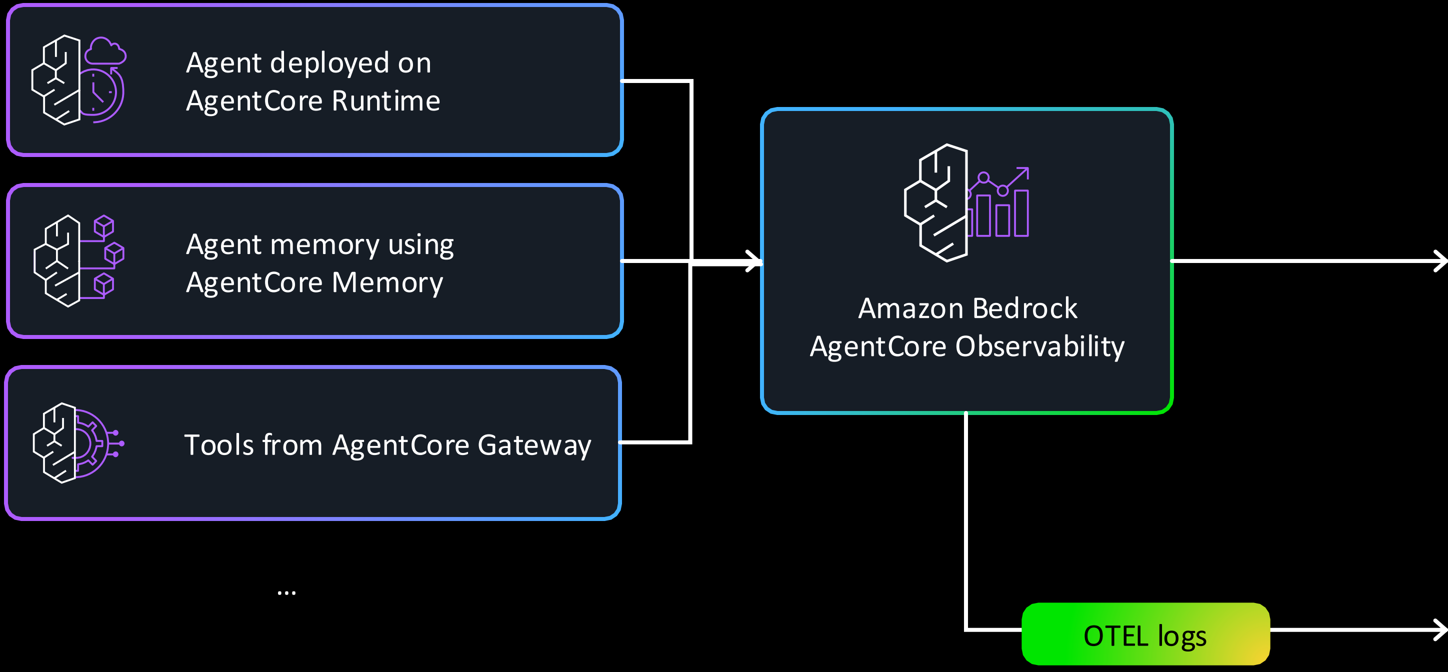
- Comprehensive end-to-end visibility into agent behavior
- Monitoring of traces, cost, latency, tokens, tools, and custom metadata
- Accelerated debugging and quality audits
- Quickly detect issues and assess performance trends

Integrate with 3P observability tools

- Integration with a wide range of monitoring and observability tools, including Cloudwatch
- OpenTelemetry (OTEL) compatible
- Flexibility to leverage your existing observability stack

AgentCore Observability

An example



Agent deployed on AgentCore Runtime

Agent memory using AgentCore Memory

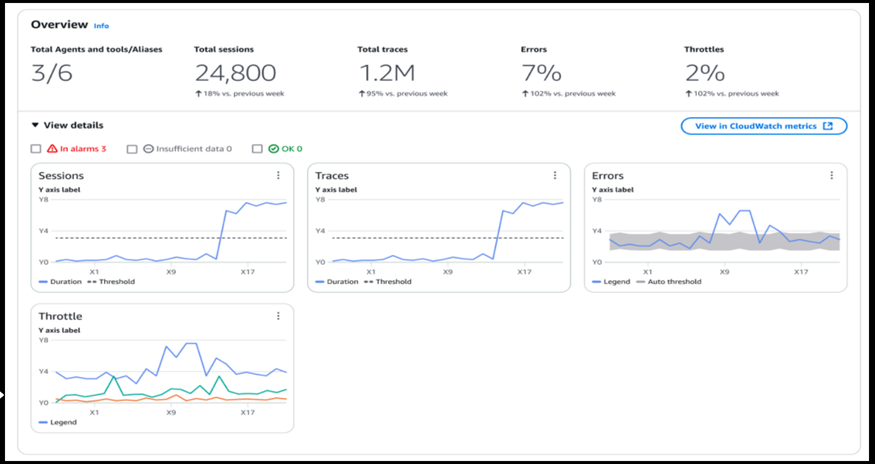
Tools from AgentCore Gateway

...

Amazon Bedrock AgentCore Observability

OTEL logs

AgentCore Observability dashboards



Overview Info

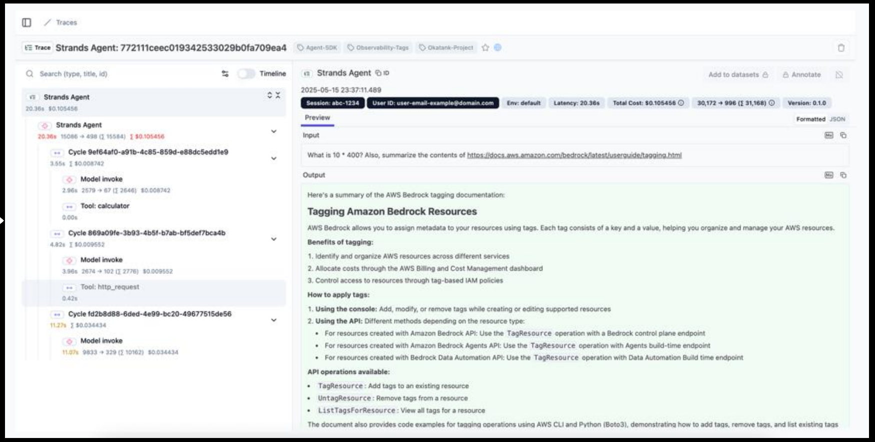
Total Agents and tools/Aliases	Total sessions	Total traces	Errors	Throttles
3/6	24,800	1.2M	7%	2%
	↑ 18% vs. previous week	↑ 95% vs. previous week	↑ 102% vs. previous week	↑ 102% vs. previous week

▼ **View details**

View in CloudWatch metrics

□ In alarms 3 □ Insufficient data 0 □ OK 0

Third-party observability dashboards



Trace Strands Agent: 772111ceec019342533029b0fa709ea4 Agent-SDK Observability-Tags Okatank-Project

Strands Agent
20.36s $0.105456

Strands Agent
20.36s 15086 → 498 (∑ 15584) ∑ $0.105456

Cycle 9ef64af0-a91b-4c85-859d-e88dc5edd1e9
3.55s ∑ $0.008742

Model invoke
2.96s 2579 → 67 (∑ 2646) $0.008742

Tool: calculator
0.00s

Cycle 869a09fe-3b93-4b5f-b7ab-bf5def7bca4b
4.82s ∑ $0.009552

Model invoke
3.96s 2674 → 102 (∑ 2776) $0.009552

Tool: http_request
0.42s

Cycle fd2b8d88-6ded-4e99-bc20-49677515de56
11.27s ∑ $0.034434

Model invoke
11.07s 9833 → 329 (∑ 10162) $0.034434

Strands Agent ID

2025-05-15 23:37:11.489

Session: abc-1234 User ID: user-email-example@domain.com Env: default Latency: 20.36s Total Cost: $0.105456 30,172 → 996 (∑ 31,168) Version: 0.1.0

Input

What is 10 * 400? Also, summarize the contents of https://docs.aws.amazon.com/bedrock/latest/userguide/tagging.html

Output

Here's a summary of the AWS Bedrock tagging documentation:

Tagging Amazon Bedrock Resources

AWS Bedrock allows you to assign metadata to your resources using tags. Each tag consists of a key and a value, helping you organize and manage your AWS resources.

Benefits of tagging:

1. Identify and organize AWS resources across different services
2. Allocate costs through the AWS Billing and Cost Management dashboard
3. Control access to resources through tag-based IAM policies

How to apply tags:

1. **Using the console:** Add, modify, or remove tags while creating or editing supported resources
2. **Using the API:** Different methods depending on the resource type:
   - For resources created with Amazon Bedrock API: Use the `TagResource` operation with a Bedrock control plane endpoint
   - For resources created with Amazon Bedrock Agents API: Use the `TagResource` operation with Agents build-time endpoint
   - For resources created with Bedrock Data Automation API: Use the `TagResource` operation with Data Automation Build time endpoint

API operations available:

- `TagResource`: Add tags to an existing resource
- `UntagResource`: Remove tags from a resource
- `ListTagsForResource`: View all tags for a resource

The document also provides code examples for tagging operations using AWS CLI and Python (Boto3), demonstrating how to add tags, remove tags, and list existing tags

AgentCore Browser

Serverless and fully managed

- Low latency browser sessions
- Auto-scales from 0 to hundreds of concurrent session

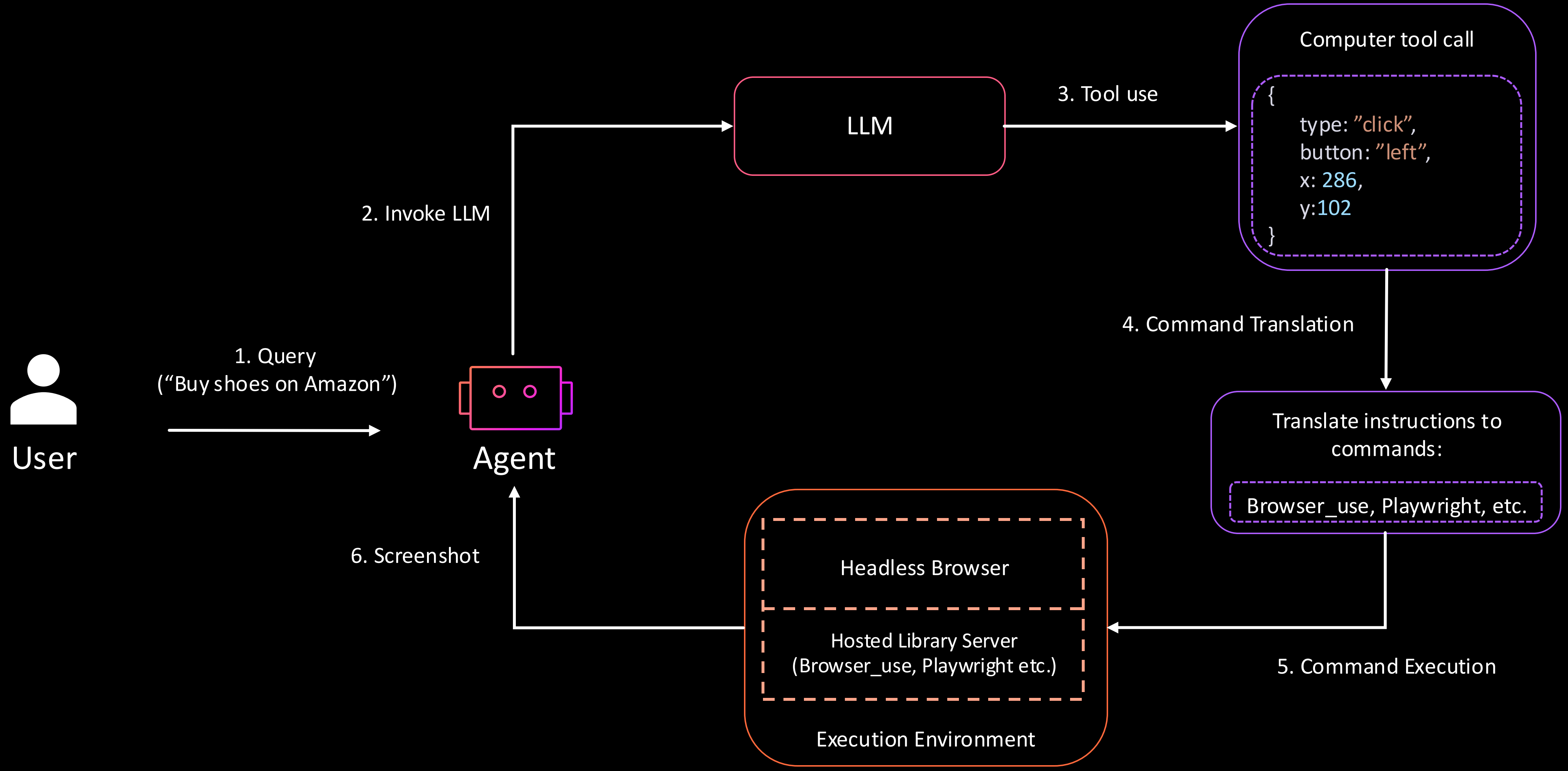
Enterprise-grade security

- Session isolated compute with VM-level isolation per user
- VPC connectivity with configurable network modes
- Secure credential handling

Enterprise Observability

- Live streaming for real-time monitoring
- Session replays for debugging
- Extensive logging of all browser commands to CloudTrail

Web navigation and workflow automation



```
{
    type: ”click”,
    button: ”left”,
    x: 286,
    y:102
}
```

AgentCore Code Interpreter

Execute Code Securely

- Execute complex workflows and data analysis in isolated sandbox environments
- Access internal data sources securely without exposing sensitive data

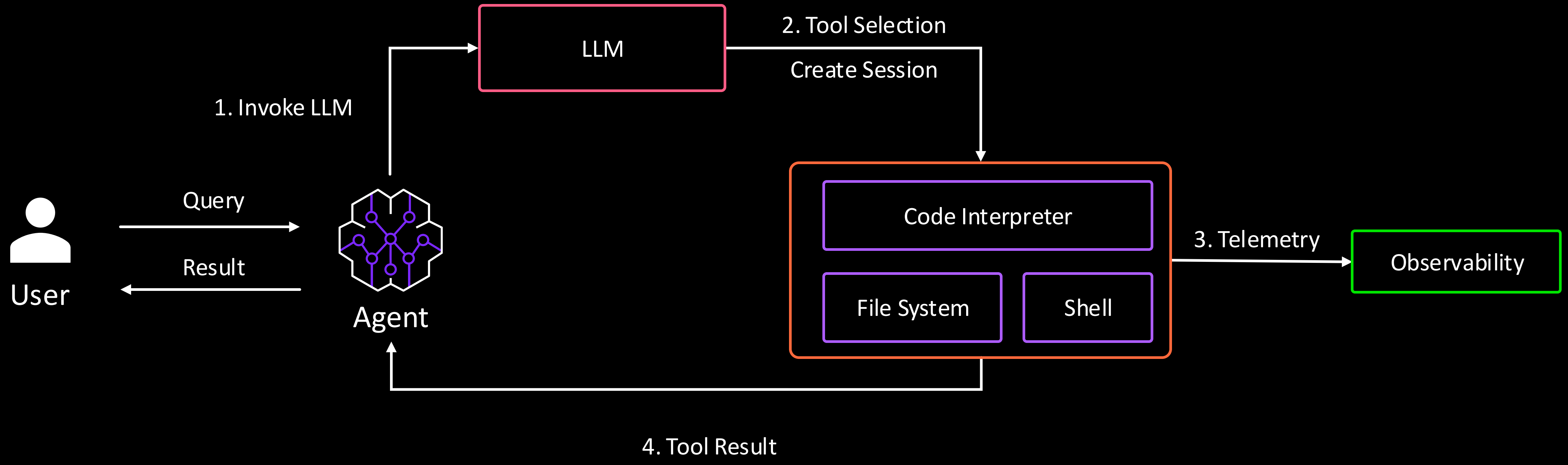
Monitoring and large-scale data processing

- Monitor and troubleshoot code execution with comprehensive observability features
- Process large datasets efficiently using Amazon S3 integration

Ease of use

- Pre-built execution runtimes for JavaScript, TypeScript, and Python with common libraries pre-installed
- Customization to add your own packages

Securely write and execute code in an isolated environment



Bedrock Agents

aws PartnerEquip

What is Bedrock Agents

- Amazon Bedrock Agents is a fully managed service
  - It higher-level approach compared to AgentCore
  - With Bedrock Agents, you define the agent's capabilities through a configuration-based interface, specifying the foundational models to use, the actions the agent can perform (via OpenAPI schemas and AWS Lambda functions), and the knowledge bases it can access for information.
- Bedrock Agents abstracts away much of the underlying complexity of managing infrastructure, memory, and orchestration, allowing developers to focus on the logic and functionality of their agents.

Simplicity vs Scalability

Feature	AgentCore	Bedrock Agents
Function	Infrastructure and services for deploying and operating agents.	A managed service for building agents.
Abstraction	Low-level, providing granular control over the agent's architecture.	High-level, abstracting away much of the underlying infrastructure.
Flexibility	High flexibility to use any framework (e.g., LangChain, CrewAI) and model.	More structured, with a focus on configuration over custom code.
Dev Effort	Requires more development effort to set up and manage.	Faster to get started with a guided, console-based experience.
Target User	Developers who need deep customization and control over their agent's environment.	Developers who want a simplified and managed experience for building agents.

Guided Approach to Choosing the right Service

Attribute	AgentCore	Bedrock Agents
Control	• Highly customized with ability to leverage multiple frameworks	• Guided and configuration-driven approach • Focus on defining the agent's logic without managing the infrastructure
Speed / Complexity	• Dev resources / expertise to manage complex and bespoke agentic system	• Rapid prototyping
Framework and Model Choice	• Use open source frameworks for control over agents memory, RTE, security • Switch between models and framework easily	• Leverage what is supported by the service
Scalability and Operational Mgmt	• Scalability requirements need control over infrastructure • Integrated observability and operational tooling in customized manner	• Fully managed (limited customization) • Built in monitoring and logging

Questions for Bank of New Zealand

1. Who is the target user for the agentic platform i.e. core developer, NC/LC
2. How much flexibility is desired for the reuse of ISV developed agents
3. What is the long-term strategy for platform (agent development, deployment) i.e. self managed or fully managed by Accenture
4. What is the ideal desired platform modularity within infrastructure, models, knowledge layer, agent & observability/governance
5. How often does the Bank intend to consider component choices within the platform and what is the process to update modules/services within the platform
6. Which platform/module/service decisions are you comfortable making now vs later

Amazon Bedrock

BROAD SELECTION OF FULLY MANAGED MODELS FROM LEADING AI COMPANIES

AI21labs

Highly efficient processing & grounded generation for long context lengths

JAMBA

amazon

Frontier intelligence and industry leading price performance

NOVA

ANTHROP\C

Excels at complex reasoning, code generation, and instruction following

CLAUDE

cohere

Powering efficient, multilingual AI agents with advanced search & retrieval

COMMAND

EMBED

RERANK

deepseek

Advanced reasoning models that solve complex problems step-by-step

DEEPSEEK-R1

Luma

High-quality video generation with natural, coherent motion & ultra-realistic details

RAY2

Meta

Advanced image and language reasoning

LLAMA

MISTRAL AI_

Specialized expert models for agentic reasoning and multimodal tasks

MISTRAL

MIXTRAL

PIXTRAL

OpenAI

Automate tasks, enhance creativity, and solve complex problems efficiently

GPT-OSS-120B

GPT-OSS-20B

poolside

Software engineering AI for large enterprises

Coming soon

stability.ai

Professional-grade images with creative control, deployable at scale

STABLE DIFFUSION

STABLE IMAGE

TwelveLabs

CTRL + F for video data: unlock the full potential of enterprise video assets

MARENGO

PEGASUS

WRITER

Purpose-built models for building and scaling AI agents across the enterprise

PALMYRA

Amazon Bedrock is the only service with industry leading Nova models

Amazon Nova models deliver frontier intelligence and industry leading price performance

UNDERSTANDING MODELS

Amazon Nova Micro

Our text only model that delivers the lowest latency responses at very low cost

GENERALLY AVAILABLE

Amazon Nova Lite

Our lowest cost multimodal model that is lightning fast for lightweight tasks

GENERALLY AVAILABLE

Amazon Nova Pro

Our highly capable multimodal model with best combination of accuracy, speed, and cost for a wide range of tasks

GENERALLY AVAILABLE

Amazon Nova Premier

Our most capable multimodal model for complex reasoning tasks and for use as the best teacher for distilling custom models

COMING SOON

CREATIVE CONTENT GENERATION MODELS

Amazon Nova Canvas

State-of-the-art image generation model

GENERALLY AVAILABLE

Amazon Nova Reel

State-of-the-art video generation model

GENERALLY AVAILABLE

Lower Cost & Latency ⟷ Increasing Intelligence

Amazon Bedrock gives you tools to supercharge with data your gen AI applications

Retrieval-augmented Generation (RAG)

Fine-tuning

Continued Pre-training