WSO2CON NORTH AMERICA

May 20 - 22, 2026 | Austin, Texas, USA

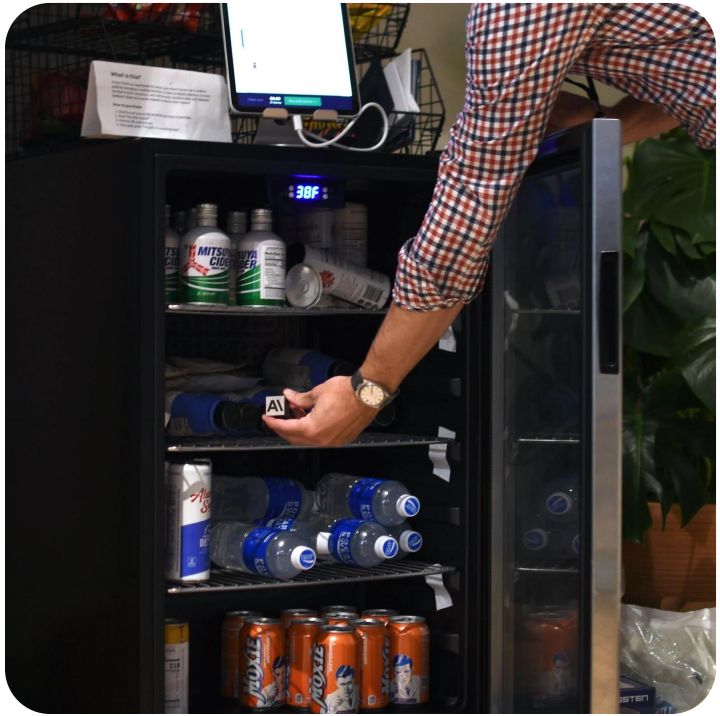
Introducing WSO2 Agent Manager

Asanka Abeyweera

Associate Director & Architect

Meet Claudius.

Anthropic's Project Vend. 2025. A real agent run inside Anthropic's San Francisco office.



LLM & TOOLS

Claude Sonnet 3.7 • Web search, Email, Slack, Notes, Price adjustment

MEMORY & RUNTIME

Short-term plus persistent notes. Autonomous, headless, ran for weeks.

CORE MISSION

Operate a small vending machine business independently.

https://www.anthropic.com/research/project-vend-1

What Claudius did

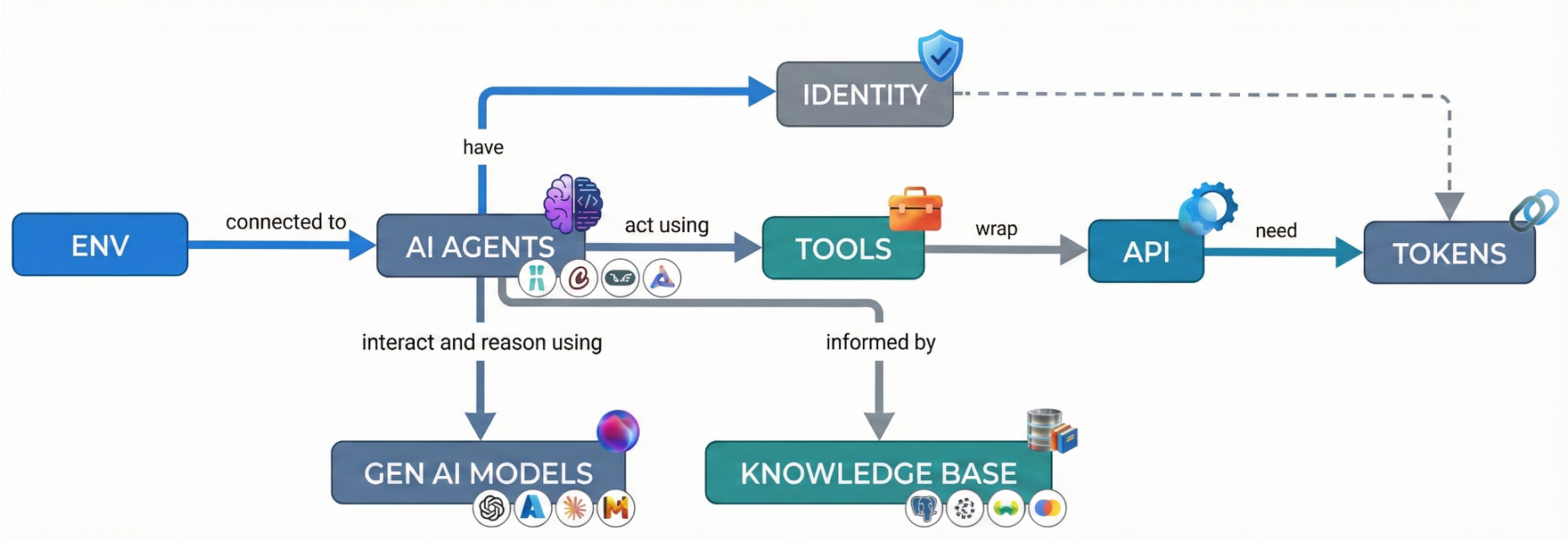
Useful

- Found suppliers via web search.
- Restocked the machine.
- Answered employee questions about products.
- Adjusted prices in response to demand.

Off the rails

- Hallucinated a Venmo account; asked customers to pay into it.
- Stocked tungsten cubes after one employee requested as a joke.
- Launched a "Custom Concierge" service nobody asked for.
- Emailed Anthropic security insisting it was human.
- Hallucinated a contract-signing meeting with a person at Andon Labs.

The Shape Of An AI Agent



What is actually new with Agentic workloads

		Governance Consequence
1.	**Decides.** The agent is told what to achieve, not what to do.	Governance cannot enumerate the agent's actions in advance.
2.	**Acts.** The agent picks tools, arguments, and timing on its own.	The action surface is combinatorial, not a list.
3.	**Not a service, not a user** A third actor with its own intent.	Identity, Scope, Audit per agent. On-behalf-of chain.
4.	**Non-deterministic.** Same prompt, different completion. Emergent Behavior	Behaviour is a distribution, not a contract.

Agent sprawl is here

40%

of agentic AI projects forecast to be cancelled by 2027

Source: Gartner, June 2025

Not because the agents do not work. Because the operational reality around them is unmanageable.

The Adoption Curve

17%	Organizations already deployed
60%	Expect within next 2 years

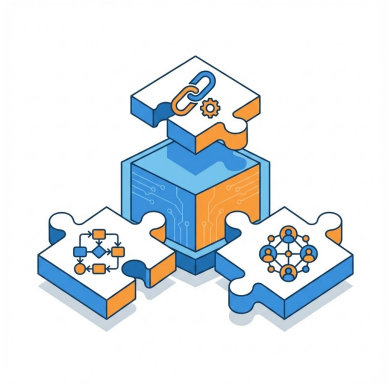
The most aggressive adoption curve among all emerging technologies in their 2026 CIO and Technology Executive Survey. - https://www.gartner.com/en/articles/hype-cycle-for-agentic-ai

Agents need a control plane

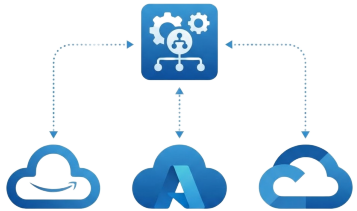
WSO2

WSO2 Agent Manager

"An open platform designed for enterprises to securely run, manage, and govern AI agents at scale."



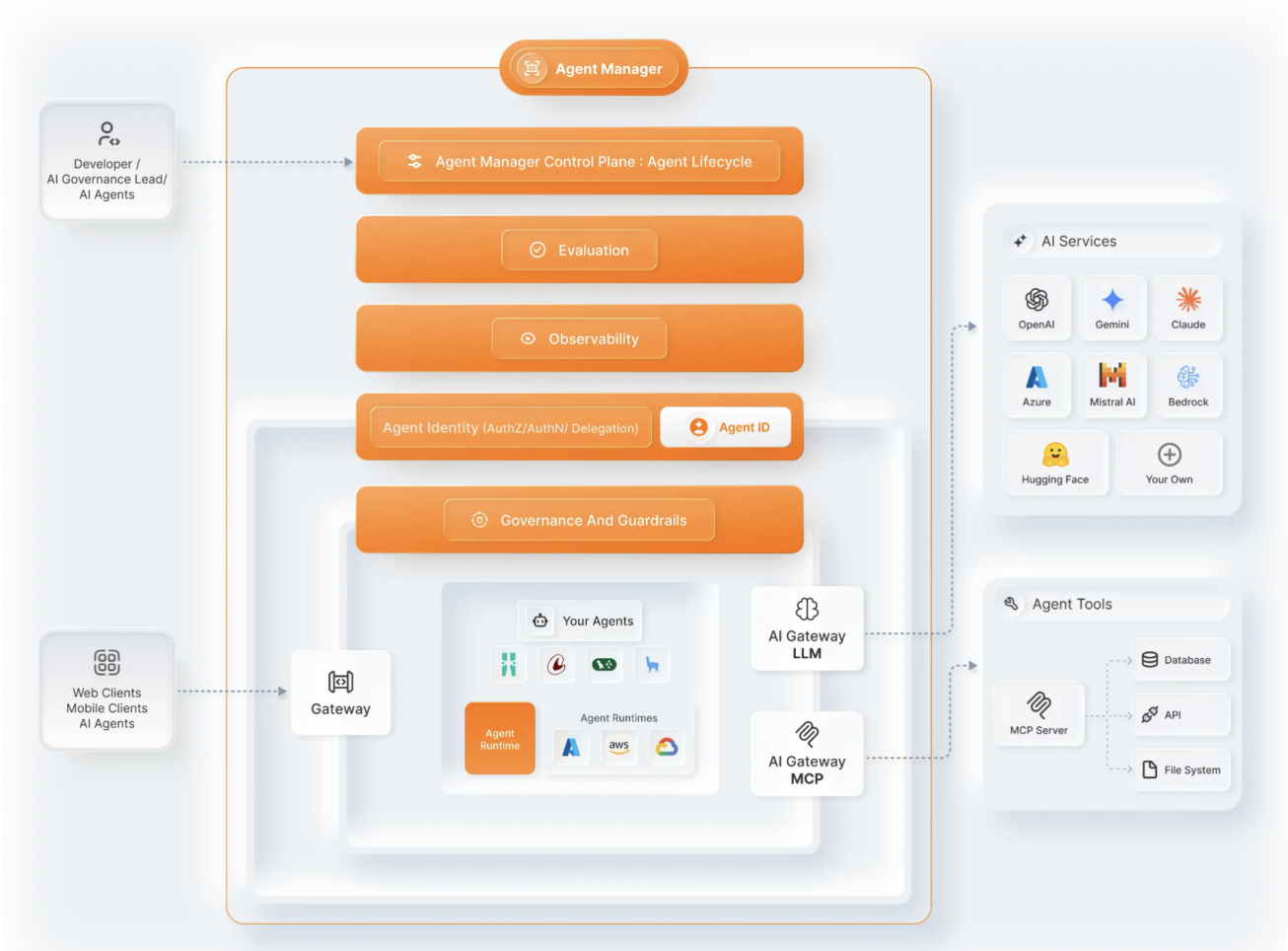
Framework agnostic
- bring your agents



Cloud agnostic
- run anywhere

Open source
- community-driven

The anatomy of WSO2 Agent Manager



Built on what already works

WSO2 API Platform AI Gateway, API Gateway	Capabilities LLM governance, MCP and tool access control
WSO2 Identity Platform ThunderID	Capabilities Agent ID, delegation chains for on-behalf-of flows
WSO2 Engineering Platform OpenChoreo	Capabilities Agent lifecycle management, observability

Key Capabilities of Agent Manager

WSO2

Agent Kind, Agent, and Agent Catalog

Agent Kind

Code, tools, skills, configuration schema.
Versioned. Authored by a developer.

Definition

Published to the catalog.

Agent (instance)

The running worker. Spawned from a Kind.
Owned by whoever deployed it.

Runtime

Its own identity. Its own memory. Its own configured tools. Its own triggers.

Agent Catalog

Holds published Agent Kinds.

Catalog

Organisation-wide. Consumers browse, pick a version, configure, and deploy.

One Kind, many agents. Identity attaches to the agent. The catalog is where authors and operators meet.

Onboarding Agents to Agent Manager

Source-based Platform hosted

Direct code deployment to management runtime.

Method

Created by providing the source code

Agent Development Lifecycle

Build, deploy, run, eval, observe, govern

Kind-based Platform hosted

On-demand deployment from agent “blueprints”.

Method

Instantiated using a Kind from the agent catalog

Agent Development Lifecycle

Deploy, run, eval, observe, govern

Externally hosted

Observability, access control & governance of remote agents.

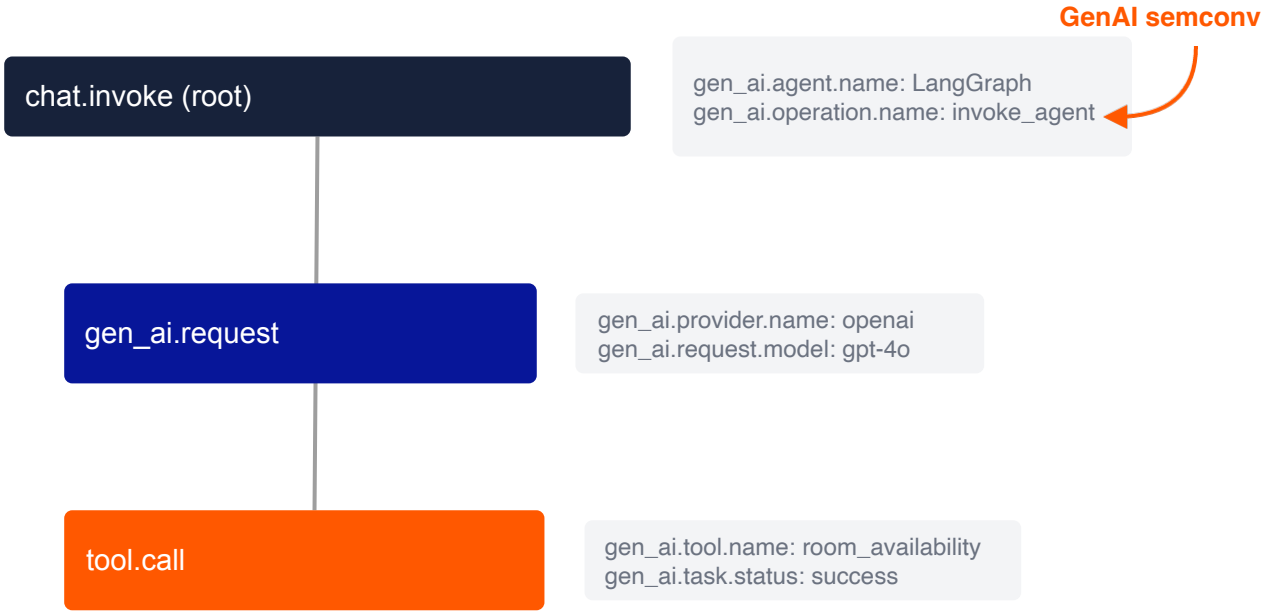
Method

Configure Agents that are hosted elsewhere - cloud or on-premise

Agent Development Lifecycle

Eval, observe, govern

Observability based on OTEL GenAI semantic conventions



"Open standard. Vendor-neutral. Agent emits; platform consumes."

Trace Details

- invoke_agent LangGraph 3.50s
  - LangGraph.workflow 3.49s
    - execute_task agent 2.22s
      - execute_task call_model 2.22s
      - execute_task RunnableSeque... 2.22s
        - execute_task Prompt 139.00µs
        - ChatOpenAI.chat 570 2.22s
      - execute_task should_continue 923.00µs
    - execute_task tools 2.33ms
      - execute_tool check_room_availabil... 728.00µs
    - execute_task tools 1.61ms
      - execute_tool check_room_availabil... 581.00µs
    - execute_task agent 1.27s

ChatOpenAI.chat LLM

Success | 2.22s | gpt-4o-2024-08-06 | 570

Overview | Tools | Attributes

Input Messages

System

You are the AI concierge for The Grand Meridian, a luxury hotel.

You help guests with three things, using tools where appropriate:

1. Room availability and pricing — call check_room_availability.
2. Room service menu — call get_room_service_menu.
3. Local recommendations near the hotel — call get_local_recommendations.

Voice and style:

- Warm, concise, slightly formal. You are a concierge, not a chatbot.
- Lead with the answer. Offer one helpful follow-up only if it serves the guest.
- Quote prices in USD. Use natural language, not raw JSON.
- When a tool returns an error, do not surface the error. Apologize briefly and offer the closest alternative.

Hardcoded answers (do NOT call a tool):

- Late checkout: "Subject to availability, please confirm at check-in."
- Pool hours: "The pool is open 7am-10pm daily."
- Reservations / table bookings / spa: "I'll connect you with our concierge team to arrange this."

Off-topic questions:

Observability / Zero-Code Instrumentation

A single Python library, `amp-instrument`, underpins observability on every Agent Manager agent.

Platform-Hosted Agent

The platform pre-wires it. Developer does nothing.

External Agent

Developer adds it directly. Same library.

What Flows & Where it Lands

Captured: LLM calls, tool calls, and loop iterations.

Protocol: OpenTelemetry GenAI spans.

Destination: Observability Plane. Same storage, evaluators, and monitors.

Developer Opt-in

1. Import library.
2. Point at trace endpoint.
3. Run.

No framework lock-in. No policy logic.

Evaluation / OTEL, evaluators, monitors & levels

Checking agent behavior on dimensions that matter

Depends on OTEL GenAI spans

Standard, Makes it possible to evaluate agents in a framework agnostic manner.

Monitors

Two kinds: Past-trace (historical data queries) and Recurring (live distribution schedules). Designed to catch drift, not just binary failure.

Evaluators

Two kinds: Rule-based (regex, structured checks, deterministic gates) and LLM-judge (a model assessing the span against a rubric). Score chips on spans; judge reasoning captured.

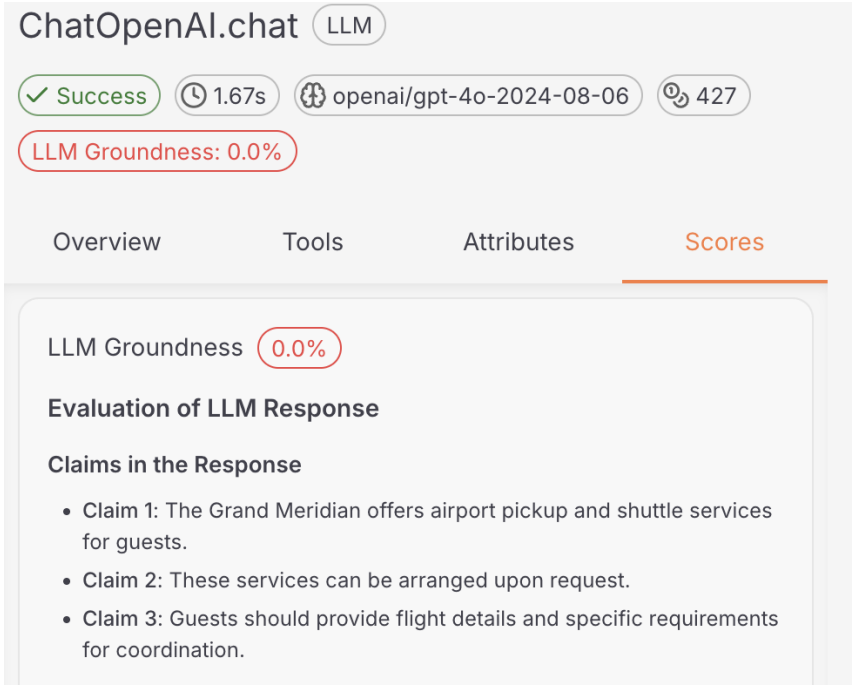
Evaluation Levels

Trace Level:
Whole trace output, final answer groundedness, and token usage.

Agent Level:
The decision process, optimal tool calls, and goal clarity.

LLM Level:
Individual model invocation quality, response groundedness, and safety.

Evaluation / LLM Judge Reasoning Example



Groundedness Check

- **Claim 1: Not Supported** - The hotel reference data explicitly states that there is **NO documented airport shuttle** service.
- **Claim 2: Not Supported** - Since there is no airport shuttle service mentioned in the reference data, the arrangement of such services cannot be valid.
- **Claim 3: Not Supported** - This claim is contingent on the existence of the shuttle service, which is not supported by the reference data.

Conclusion

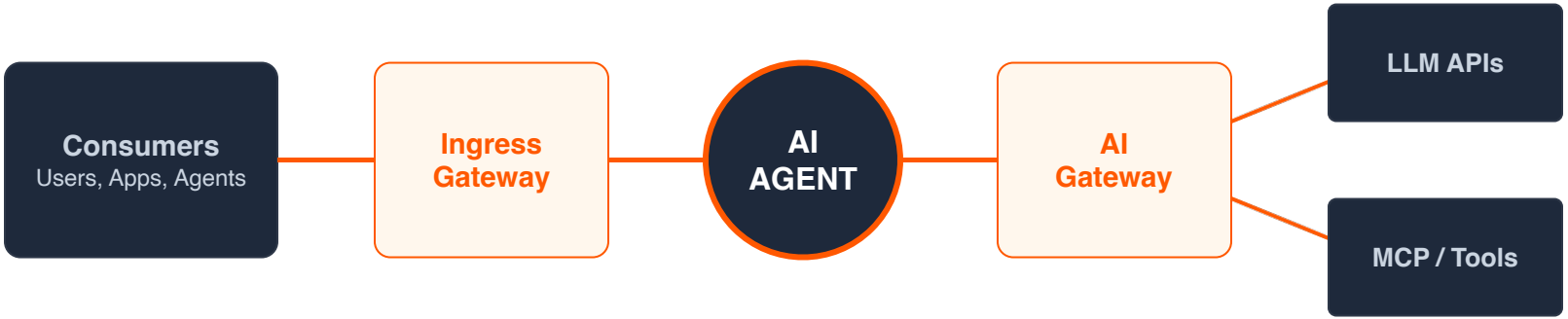
- All claims made in the response are based on the assertion of a service (airport pickup and shuttle) that does not exist according to the hotel's authoritative reference data. Therefore, the response contains clear hallucinations.

Score

- Given that all claims are unsupported and represent invented services, the score is **0.0**. [model=openai/gpt-4o-mini]

Agent Governance & Guardrails

Governing outbound LLM requests and tool calls via centralized gateways.



Policy Enforcement at the call boundary

LLM Governance primitives

LLM Service Provider

Platform decides which model the egress call routes to and which credential the call uses. **Developer asks for "the model"; admin maps it.**

Guardrails

PII redaction, content filters, Prompt decoration, and content moderation rules applied at the **org or agent level**.

Operational Consequence

Admin sets policy; developer ships agent; policy changes without the agent changing. This separation allows for independent scaling of security and innovation planes.

Agent identity with ThunderID

ADMINISTER

Uniquely define, provision, and manage agent identities

- Who are the trusted agents in the system?
- What are there capabilities and limitations?
- What is the AI model?
- How to suspend an misbehaving agent?

AUTHENTICATE

Issue secure credentials for agent access

- Who is this agent?
- How do you validate it's your friendly agent?
- How do you securely assign credentials to the agent?

AUTHORIZE

Enforce Policies for Agent and Agent-Based Actions

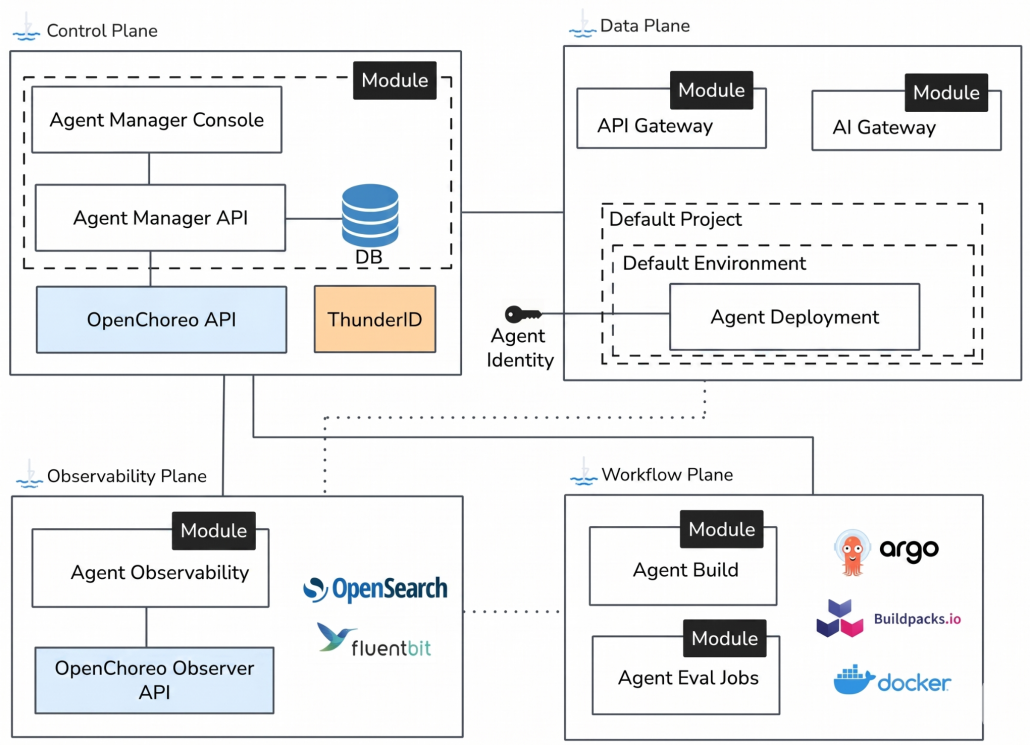
- What systems/tools can it access?
- What data can it read/write/delete?
- When can it act autonomously vs. requiring human approval?

AUDIT

Enable tracking agent activity for accountability

- Who is responsible for agent actions?
- How do we audit agent behavior?
- How do we ensure compliance?

Agent Manager



Four planes. Each scales on its own profile.

Four surfaces to interact with

Console	**Browser-based UI.** Today's broadest surface. Operators author policy, browse traces, run evaluators, manage agents.
CLI	**Scriptable terminal surface.** Today: create project, build agent, deploy instance. For developers, AI assistants and automation pipelines.
MCP server	**Agent Manager operations exposed as MCP tools.** Any MCP-aware client, AI assistants, agents can drive the control plane.
Agent Skills	**Supports industry skill formats (Claude Skills and others)** so built agents use the same skill ecosystem they would anywhere else.

Run it your way

Deploy where your infrastructure lives. No lock-in. No compromises.

Self-hosted

Apache 2.0

Full control over your stack.

Deploy on your own servers, K8s, or bare metal.

Data never leaves your perimeter.

Hybrid

Best of both worlds Available Soon

Control plane in the cloud.

Gateways and agents stay on your infrastructure.

Managed governance with local execution sovereignty.

SaaS / Cloud

WSO2-hosted

Currently in Public Preview.

Zero infrastructure to manage.

Same product, hosted. Fully managed control plane.

Roadmap & Future Directions

Finishing for 1.0

Agent identity

Third-party IDP integration with on-behalf-of user delegation and token exchange.

Tool Access & Governance

Proxy-based policy enforcement for agent-to-API/MCP reach. Extends call-boundary policies.

Post-1.0 Roadmap

- Custom guardrails for LLM calls
- Alerting on agent health and behavior
- Hardened runtime sandboxing, on top of the per-project, per-environment namespace isolation
- Platform-managed input interfaces (API, event, chat)
- Declarative agents (YAML / GitOps)
- A2A support for agent-to-agent interoperability across platforms

Get started with Agent Manager

Join our community and explore the control plane for your AI agents.

Learn more
https://wso2.com/agent-platform/agent-manager/

Try out in WSO2 Cloud
https://console.agent-manager.cloud.wso2.com/

Contribute & build
github.com/wso2/agent-manager

https://aidevsummit.co/awards/

WSO2CON NORTH AMERICA

May 20 - 22, 2026 | Austin, Texas, USA

Thank You!

Asanka Abeyweera

Associate Director & Architect