Google Vertex AI Agents Provide Managed Agent Infrastructure. The Agents Have No Canonical Schema.

by Nick Clark | Published March 28, 2026 | PDF

Google Vertex AI Agents provides managed infrastructure for building and deploying AI agents with grounding, tool use, and conversation management on Google Cloud. The managed service handles hosting, scaling, and integration. But Vertex AI agents have no canonical schema. Each agent is a configuration of model parameters, grounding sources, and tools. The structural definition of what an agent is, what fields it must carry, and how governance, memory, and identity relate is absent. The gap is between managed agent infrastructure and a canonical agent definition.

Google Vertex AI's agent infrastructure and grounding capabilities provide enterprise-grade agent hosting. The gap described here is about agent structural definition, not about infrastructure quality.

Grounding without structural identity

Vertex AI agents can be grounded in enterprise data through Google Search, Vertex AI Search, or custom data stores. Grounding improves response quality. But grounding is a capability, not a structural field. There is no schema defining how grounding relates to the agent's identity, memory, or governance. Grounding is a feature of the configuration. It is not a typed field of a canonical agent.

Conversation state without governed memory

Vertex AI agents maintain conversation state across turns. But conversation state is session context, not governed memory. There is no lineage tracking, no governance on state mutations, and no structural validation of memory integrity. The state is what the conversation produced. It is not a governed field of the agent.

What a canonical agent schema provides

A canonical agent schema would give Vertex AI agents structural definition. Each agent would carry typed fields for identity, memory, governance, capabilities, execution state, and lineage. The managed infrastructure would validate agent structure, enforce governance constraints, and maintain lineage. Agents would be interoperable across platforms because all platforms would agree on what an agent structurally is.

Agent Schema All 21 steps →

Define what an autonomous agent is — structurally.

Patent
US 19/452,651 · filed
Primary Technical Disclosure
○ Cognition-Compatible Semantic Agent Objects and Structural Validation
Secondary Technical
○ Partial Agent Structural Validity: Fewer Fields, Still Deterministic○ Minimum Two-Field Validation Threshold: The Floor of Semantic Structure○ Field Interaction Rules: Deterministic Constraints Between Canonical Fields○ Field-Based Role Typing: Agent Roles Derived From Structural Composition○ Semantic Templates: Predefined Field Arrangements as Agent Class Contracts○ Structural Scaffolding Logic: Resolving Missing Fields Through Inference or Defaulting○ Field-Aware Default Resolution: Deterministic Behavior When Fields Are Absent○ Traceable Semantic Lineage Graph: Mutation History Embedded in Agent Objects○ Serialization With Stateless Compatibility: Reconstruction Without External Session State○ Schema Governance Through Versioned Policies: Cross-Version Structural Interoperability
Applications (General)
○ Enterprise AI Agent Interoperability Through Canonical Schema○ Robotic System Standardization via Structural Field Composition○ Multi-Vendor AI Agent Interoperability○ Digital Twin Standardization Through Canonical Fields○ Healthcare AI Agent Portability○ Defense Coalition Interoperability○ Insurance Claims Processing Through Standard Agents○ Legacy System Integration via Schema Bridging
Applications (Specific)
○ LangChain Built the Agent Framework. It Did Not Define What an Agent Is.○ AutoGen Enabled Multi-Agent Conversations. The Agents Have No Structural Definition.○ CrewAI Organized Agents Into Teams. The Agents Still Have No Schema.○ Semantic Kernel Integrated AI Into Enterprise Code. The Agents It Creates Have No Schema.○ OpenAI Assistants API Provides Agent Tooling. It Does Not Define Agent Structure.● Google Vertex AI Agents Provide Managed Agent Infrastructure. The Agents Have No Canonical Schema.○ Amazon Bedrock Agents Orchestrate Foundation Models. The Agents Have No Structural Definition.○ Haystack Built Composable NLP Pipelines. The Pipeline Components Have No Agent Schema.○ LlamaIndex Built the Data Framework for LLM Applications. The Data Objects Have No Agent Schema.○ Dify Made LLM Application Development Visual. The Applications Have No Agent Schema.
Agent Schema overview →
AQ
deterministic
autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see Patents for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see Licensing. Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See Legal for terms and conditions.

Last updated: 2026-03-03

-
- Inventive Steps
- Licensing
- Patents
- Articles
- Legal
- Opportunities
- Sitemap

-
- nick@qu3ry.net
- 72 28 14 36 01

Invented by Nick Clark | Founding Investors: Devin Wilkie