Agent Metadata Specification

Version 0.2.5

Published by Tavro AI, LLC

January 23, 2026

Table of Contents

License and Usage	3
Acknowledgements	4
Introduction	5
Importance of Agent Metadata Standards	6
Enhancing the Google Agent2Agent(A2A) Protocol	7
Agent Proliferation	7
Agent Data Model	9
Agent Identification	11
Agent Configuration	11
Agent Relations	13
Provider	14
LLM Model	14
Prompt Template	15
Memory	16
Knowledge Source	16
Data Source	17
Tool	18
MCP Server	19
Guardrail	20
AI Use Case	21
Application	22
Business Process	22
Regulation	23
Risk Assessment	24
Agent Metadata for Medical Device Industry	24
Outstanding Items	26

License and Usage

Acknowledgements

Leader Authors:

- Sunil Soares (Tavro AI)
- Sanjeev Varma (Tavro AI)

Key Contributors and Reviewers

The following individuals are acknowledged for their significant contributions to the review and development of this document, **listed alphabetically by last name**:

Key Contributor / Reviewer Name	Affiliation
Prasanna Kumar Akiri	Tavro AI
Venkata Atluri	USAA
Val Calvo	BankUnited
Stan Christiaens	Collibra
Joe DeLuca	Independent Contributor
Tony DiPerna	BankUnited
Mihir Dudhatra	Tavro AI
Pierre Gomes	BankUnited
Shawn Harbaugh	Citizens & Northern Bank
Mike Jennings	Independent Contributor (Formerly Walgreens)
Peter Kapur	CarMax
Manirajkumar Kotha	Tavro AI
Gokula Mishra	OmniProAI (Formerly McDonald's)
Rahul Pandit	Tavro AI
Su Rayburn	Delta Community Credit Union
Khushboo Shah	Tavro AI
Doug Shannon	Independent Contributor
Bryan Swann	Independent Contributor

Introduction

AI agents are multiplying across the enterprise. Similar to the challenges with *shadow IT*, AI agents are creating so called *shadow action*. AI agents are often unmanaged, unmapped and autonomous. The result is a critical visibility vacuum (see Figure 1).

Figure 1: AI agents create a visibility vacuum

Because agents are easy to build and deploy, they create multiple challenges (see Figure 2):

- *Discovery Gap* – Organizations do not know that the agents exist, who owns them, or where they operate.
- *Risk Multiplier* – Agent may operate as unmonitored "digital insiders" with high privileges and broad access.
- *Cost Bloat* – Redundant agents waste compute cycles and inflate cloud bills.

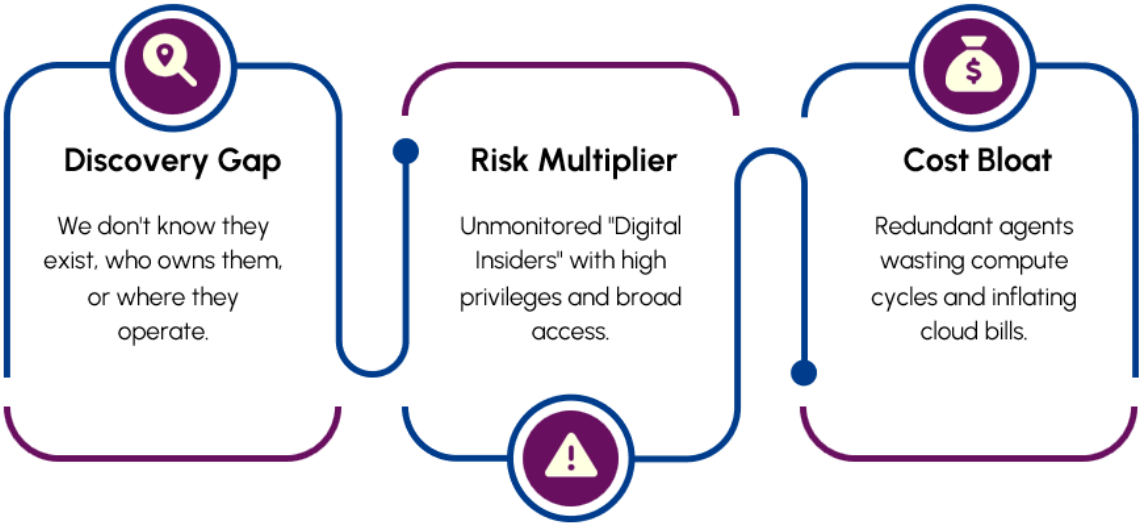

Figure 2: AI agents create a discovery gap, risk multiplier, and cost bloat

This document serves as a comprehensive template for capturing the essential metadata and core configuration parameters of a specific AI agent. The strategic importance of this template lies in its role as a standardized framework for organizations leveraging enterprise-grade agentic solutions.

Importance of Agent Metadata Standards

By implementing this robust metadata specification, enterprises gain several critical, strategic advantages:

- **Unified Enterprise-Wide View and Single Source of Truth of Agents:** The specification provides a mechanism to develop a consolidated, holistic view of all deployed agents across the entire organization. This centralization establishes a single, authoritative source of information for every agent, eliminating data silos and inconsistencies that can plague decentralized management systems.
- **Enhanced Accountability and Transparency of Ownership:** The metadata structure rigorously captures ownership details, ensuring clear accountability for the agent's performance, maintenance, and policy adherence. This transparency is crucial for operational governance and risk mitigation.
- **Automated Risk Management Functions for Agents:** By leveraging the standardized metadata, enterprises can apply systematic risk analysis, monitoring, and control across all business processes, applications, and the underlying agents they consume.
- **Accelerated Audit Readiness for Governance and Compliance:** The standardized and comprehensive nature of the metadata significantly accelerates the process of achieving audit readiness. It provides a structured record necessary for satisfying stringent governance requirements and demonstrating compliance with internal policies and external regulations (e.g., GDPR, CCPA, industry-specific compliance standards).

- **Easier Third-Party Risk Assessments for AI-Enabled Applications:** By standardizing the agent metadata, applications with embedded agents should find it easier to complete third-party risk assessments.

Enhancing the Google Agent2Agent(A2A) Protocol

The Google Agent2Agent (A2A) is an open protocol that provides a standard way for agents to collaborate with each other, regardless of the underlying framework or vendor. Agents can advertise their capabilities using an "Agent Card" in JSON format, allowing the client agent to identify the best agent that can perform a task and leverage A2A to communicate with the remote agent.[1]

The Agent Card lays the operational foundation for agents to find each other, understand basic capabilities (modalities), and handshake for collaboration. However, the Agent Card does not address the business context, risk management, and governance.

The Agent Metadata Specification seeks to enhance the Google Agent Card to address these additional topics.

Agent Proliferation

Traditional metadata platforms capture information primarily from analytical systems. However, the metadata challenges increase exponentially for agents, which also leverage operational systems (see Figure 3).

Figure 3: Agent metadata increases exponentially from analytical to operational systems

[1] Google, "Announcing the Agent2Agent (A2A) Protocol," April 9, 2025, https://developers.googleblog.com/en/a2a-a-new-era-of-agent-interoperability/.

A number of platforms produce or consumer agent metadata (see Figure 4).

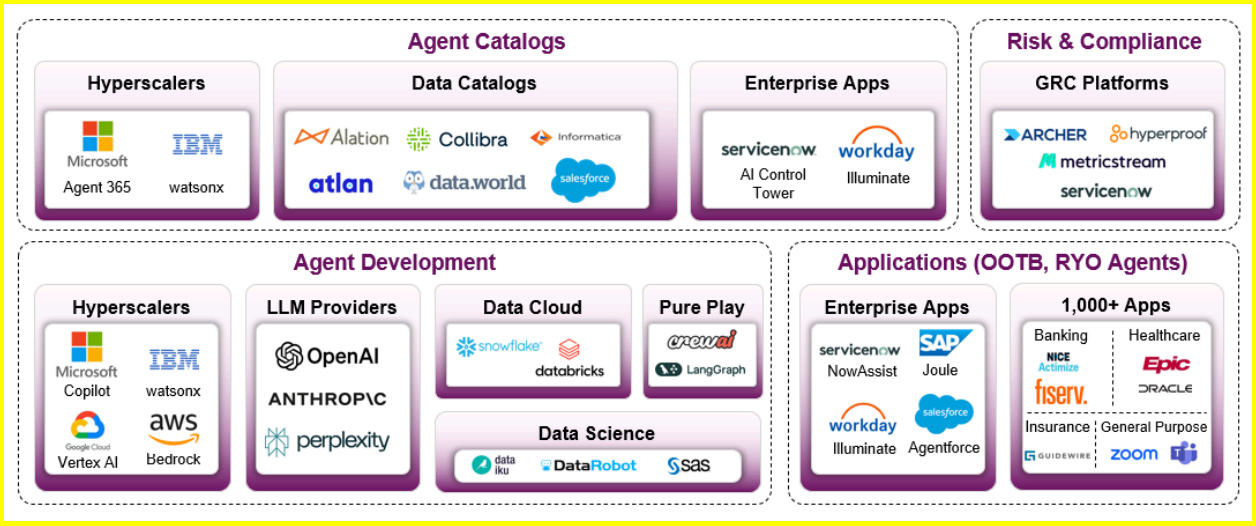

Figure 4: Producers and consumers of agent metadata

Agent Metadata Producers including the following:

- Hyperscalers (e.g., Microsoft Copilot, Google Vertex AI, IBM watsonx, Amazon Bedrock)
- LLM Providers (e.g., OpenAI, Anthropic, Perplexity)
- Data Cloud Providers (e.g., Snowflake, Databricks)
- Pure Play Agent Platforms (e.g., crewAI, LangGraph)
- Data Science Vendors (e.g., Dataiku, DataRobot, SAS)
- Enterprise Applications with out-of-the-box (OOTB) and roll-your-own (RYO) agents (e.g., ServiceNow NowAssist, SAP Joule, Workday Illuminate, Salesforce Agentforce)
- Industry-Specific Applications with OOTB and RYO agents (e.g., Fiserv and NICE Actimize in Banking, Epic and Oracle in healthcare, Guidewire in Insurance)
- The typical enterprise uses more than 1,000 applications[2] and an increasing percentage of these platforms will have OOTB and RYO agents

Agent Metadata Consumers including the following:

- Hyperscalers (e.g., Microsoft Agent 365, IBM watsonx)
- Data Catalogs (e.g., Alation, Atlan, Collibra, data.world/ServiceNow, Informatica/Salesforce)
- Enterprise Applications (e.g., ServiceNow AI Control Tower, Workday Illuminate)
- Governance, Risk, and Compliance (GRC) Platforms (e.g., Archer, Hyperproof, MetricStream, ServiceNow IRM)

[2] Salesforce, February 1, 2023, https://www.salesforce.com/news/stories/connectivity-report-2023/.

Agent Data Model

The conceptual data model for the Agent Metadata Specification is shown in Figure 5.

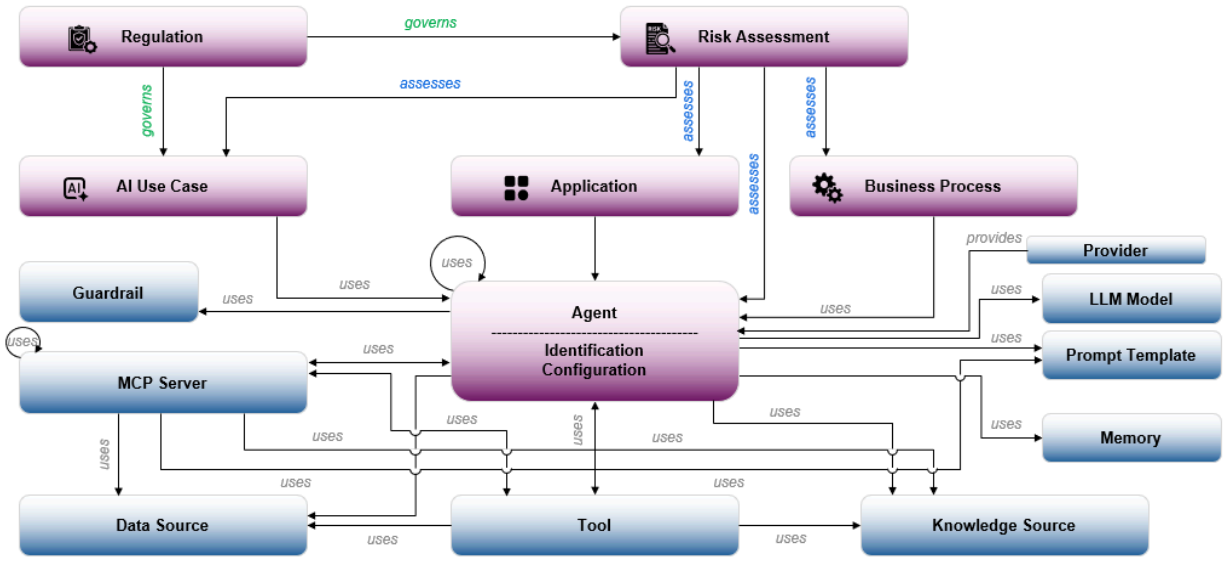

Figure 5: Conceptual data model for agent

The template is logically partitioned into the following key sections, each addressing a distinct facet of the agent's identity, operation, and lifecycle:

- **Agent Identification:** This foundational section is dedicated to capturing the metadata required to uniquely identify the agent. This includes key identifiers, its current deployment status (e.g., development, staging, production, deprecated), versioning information, and the essential details regarding its ownership and organizational context.
- **Agent Configuration:** This section details the technical architecture and underlying components of the Agent. It meticulously categorizes and documents metadata associated with the core technologies, such as the underlying Large Language Model (LLM) being utilized (e.g., model name, version, fine-tuning details), specific Memory models (e.g., type, retention policy, capacity), and other critical computational and operational parameters.
- **Agent Relations:** This section defines the relations between the agent and other agents. It also defines the relations between the agent and LLM models, prompt templates, memory, knowledge sources, data sources, tools, MCP Servers, guardrails, AI use cases, applications, business processes, and risk assessments.
- **Provider:** This section defines the provider of the agent.
- **LLM Model:** This section defines the LLM model used by the agent.
- **Prompt Template:** This section defines the prompt templates used by the agent.

- **Memory:** This section defines the external system used for long-term data/vector storage by the agent.
- **Knowledge Source:** This section provides a deep dive into the information resources the agent relies upon. It specifies the data sources it has been trained on (e.g., dataset identifiers, date of last training), and critically, details the mechanisms and interfaces it uses to access its knowledge base, including Retrieval-Augmented Generation (RAG) system configurations, database connections, and document repositories.
- **Data Source:** This is a vital section for enterprise governance, establishing the end-to-end context for the agent's usage. This mapping provides a comprehensive view of the agent's usage patterns, the data sets it consumes (input lineage), and the resulting data sets it produces (output lineage), which is essential for impact analysis.
- **Tool:** This defines the agent's functional capabilities and its interaction boundary with the external world. It enumerates what the agent is capable of doing (its designated actions and use cases) and precisely how it interacts with external systems, APIs, or business applications, including function call specifications and security protocols.
- **MCP Server:** This defines the agent's interaction with MCP servers.
- **Guardrail**: This defines the safeguards that keep the agent operating safely, responsibly and within defined boundaries.
- **AI Use Case:** This defines the AI uses cases that use the agent.
- **Application:** This defines the applications that use the agent.
- **Business Process:** This defines the business processes that use the agent.
- **Regulation:** This provides the regulatory context for the AI use case and risk assessment.
- **Risk Assessment:** This defines the risk assessment for AI use cases, applications, business processes, and agents.

Agent Identification

Table 1 summarizes the core identifying and descriptive details of the agent.

Attribute	Description
Agent ID	A unique, permanent identifier for the agent (e.g., HR-POL-003)
Agent Version	The current version number of the agent's configuration and logic
Name	The human-readable name of the agent (e.g., Internal HR Policy Assistant)
Description	A concise summary of the agent's purpose, capabilities, and target user
URL	The preferred endpoint URL for interacting with the agent. This URL MUST support the transport specified by Preferred Transport.'

Documentation URL	An optional URL to the agent's documentation
Icon URL	An optional URL to an icon for the agent
Goal Orientation	The specific objective or success metric the agent is designed to achieve
Role	The defined character or communication style that governs its interaction
Owner	The team or department responsible for the agent's maintenance and cost
Environment	The deployment environment (e.g., DEV, UAT, PROD)
Tags	A list of keywords for search and categorization (e.g., HR, policy, internal)
Governance Status	The current governance lifecycle status (e.g., DRAFT, APPROVED, DECOMMISSIONED)
Reviewer	Name of the person who approved the latest governance status

Table 1: Agent identification attributes

Agent Configuration

Table 2 summarizes the configuration details for the agent.

Attribute	Description
Access Scope	The agent's overall data access level (e.g., LOW_PRIVILEGE)
Memory Type	The type of memory storage used (e.g., VECTOR_DB, KEY_VALUE_STORE)
Data Freshness Policy	The maximum acceptable age of the data (caching policy) for the source
Autonomy Level	The degree to which the agent can act independently without human approval (FULL, SEMI-AUTONOMOUS, REACTIVE)
Reasoning Model	The underlying logic or planning paradigm (ReAct, ReWOO, Deductive, Inductive, Goal-based)
Skills	The set of skills, or distinct capabilities, that the agent can perform (e.g., generate recipe, translate text, book a flight, analyze CSV files)
Capabilities	A declaration of optional capabilities supported by the agent (e.g., file uploads, authentication support, human-in-the-loop, tool calling)
Default Input Modes	Default set of supported input MIME types for all skills, which can be overridden on a per-skill basis
Default Output Modes	Default set of supported output MIME types for all skills, which can be overridden on a per-skill basis
Security	A list of security requirement objects that apply to all agent interactions. Each object lists security schemes that can be used. Follows the OpenAPI 3.0 Security

	Requirement Object. This list can be seen as an OR of ANDs. Each object in the list describes one possible set of security requirements that must be present on a request. This allows specifying, for example, "callers must either use OAuth OR an API Key AND mTLS."
Security Schemes	A declaration of the security schemes available to authorize requests. The key is the name. Follows the OpenAPI 3.0 Security Scheme Object.
Signatures	JSON Web Signatures computed for this AgentCard
Supports Authenticated Extended Card	If true, the agent can provide an extended agent card with additional details to authenticated users. Defaults to false.
Additional Interfaces	A list of additional supported interfaces (transport and URL combinations). This allows agents to expose multiple transports, potentially at different URLs. Best practices: - SHOULD include all supported transports for completeness - SHOULD include an entry matching the main 'url' and 'preferredTransport' - MAY reuse URLs if multiple transports are available at the same endpoint - MUST accurately declare the transport available at each URL Clients can select any interface from this list based on their transport capabilities and preferences. This enables transport negotiation and fallback scenarios.
Preferred Transport	The transport protocol for the preferred endpoint (the main 'url' field). If not specified, defaults to 'JSONRPC'. IMPORTANT: The transport specified here MUST be available at the main 'url'. This creates a binding between the main URL and its supported transport protocol. Clients should prefer this transport and URL combination when both are supported.
A2A Protocol Version	The version of the A2A protocol this agent supports

Table 2: Agent configuration attributes

Agent Relations

Table 3 summarizes the relations for the agent with other objects.

Asset	Relation	Asset
Agent	uses	Agent
Agent	is used by	Agent
Agent	Is Provided by	Provider
Agent	uses	LLM Model
Agent	uses	Prompt Template

Agent	uses	Memory
Agent	uses	Knowledge Source
Agent	uses	Data Source
Agent	uses	Tool
Agent	is used by	Tool
Agent	uses	MCP Server
Agent	is used by	MCP Server
Agent	uses	Guardrail
Agent	is used by	AI Use Case
Agent	is used by	Application
Agent	is used by	Business Process
Agent	is assessed by	Risk Assessment

Table 3: Agent relations

Provider

Table 4 summarizes key attributes of the provider of the agent.

Attribute	Description
Name	Name of the agent's service provider

Table 4: Provider details

Table 5 summarizes the relations of the provider.

Asset	Relation	Asset
Provider	provides	Agent

Table 5: Provider relations

LLM Model

Table 6 summarizes the attributes of the LLM model used by the agent.

Attribute	Description
Name	The foundational model used by the agent (e.g., gemini-2.5-flash)
Version Number	The version number of the LLM model

Table 6: LLM model attributes

Table 7 summarizes the relations of the LLM model used by the agent.

Asset	Relation	Asset
LLM Model	is used by	Agent

Table 7: LLM model relations

Prompt Template

Table 8 summarizes the prompt template for the agent.

Attribute	Description
Identifier	A unique identifier for the prompt template used to guide LLM behavior
Name	The name of the prompt template used to guide LLM behavior (e.g., ABC-RAG-Standard-V2)
Description	The actual prompts used to guide LLM behavior

Table 8: Prompt template attributes

Table 9 summarizes the relations of the prompt template used by the agent.

Asset	Relation	Asset
Prompt Template	is used by	Agent

Table 9: Prompt template relations

Memory

Table 10 summarizes the memory for the agent.

Attribute	Description
Identifier	A unique identifier for the external system used for long-term data/vector storage
Name	The name of the external system used for long-term data/vector storage (e.g., Atlas-HR-RAG-VectorDB)
Type	The type of memory storage used (e.g., VECTOR_DB, KEY_VALUE_STORE)

Table 10: Memory attributes

Table 11 summarizes the relations of the memory used by the agent.

Asset	Relation	Asset
Memory	is used by	Agent

Table 11: Memory relations

Knowledge Source

Table 12 summarizes the attributes for the knowledge sources for the agent.

Attribute	Description
Identifier	A Unique ID for the knowledge source
Name	A list of all specific knowledge sources (e.g., databases, documents) the agent can access
Access Mechanism	The protocol or service used to retrieve knowledge (e.g., REST API, SQL connector)

Table 12: Agent knowledge attributes

Table 13 summarizes the relations of the knowledge sources used by the agent.

Asset	Relation	Asset
Knowledge Source	is used by	Agent

Table 13: Knowledge source relations

Data Source

Agentic lineage needs to map the entire decision path from prompt to logic to action to impact (see Figure 6).

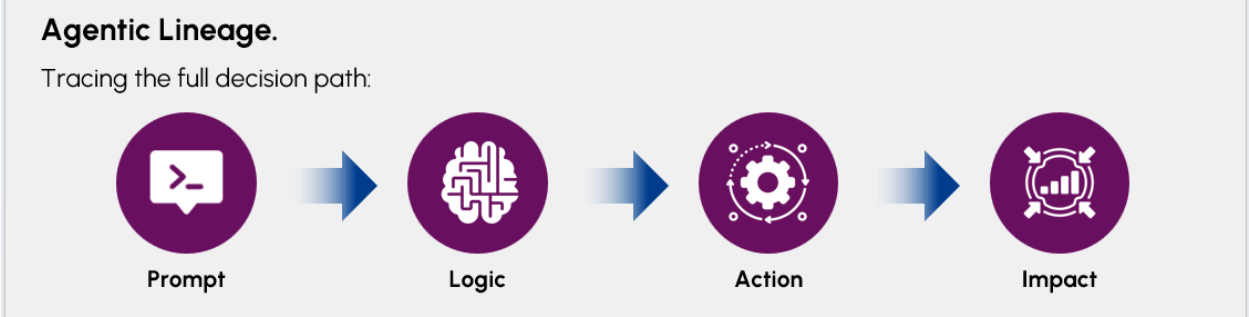

Figure 6: Agentic lineage

Table 14 captures the data sources associated with the agent.

Attribute	Description
Relationship ID	A unique Relationship ID
Parent Relationship ID	Parent ID, if any
Source Object ID	Unique ID as in the Source system
Source Object Domain	Domain
Source Object Name	Object name
Source Object Type	Type of Source Object (e.g., Agent, MCP Server, Table, Column, View, File, Folder)
Target Object ID	Unique ID as in the Target system
Target Object Domain	Domain
Target Object	Object Name

Name	
Target Object Type	Type of Target Object (e.g., Agent, MCP Server, Table, Column, View, File, Folder)
Access Level	READ, WRITE or DELETE

Table 14: Data source attributes

Table 15 summarizes the relations of the data sources used by the agent.

Asset	Relation	Asset
Data Source	is used by	Agent

Table 15: Data source relations

Tool

Table 16 summarizes the tools configured for the agent including its external capabilities and delegation options.

Attribute	Description
Identifier	A unique reference ID for the tool (e.g., PolicySearchTool)
Name	Name of the tool
Description	Detailed explanation of the purpose and functionality
Delegation Possible	Boolean indicating if the agent can pass the request to another agent
Allowed Delegates	A list of Agent IDs, which the agent is allowed to delegate to
Input or Output	Indicates whether it is an input or output parameter
Parameter Name	Name of the parameter
Parameter Type	Required type and format of the parameter
Default Value	Default value, if any

Table 16: Tool attributes

Table 17 summarizes the relations of the tools used by the agent.

Asset	Relation	Asset
Tool	is used by	Agent
Tool	uses	Agent
Tool	uses	Data Source
Tool	uses	Knowledge Source
Tool	uses	MCP Server
Tool	is used by	MCP Server
Tool	uses	Tool
Tool	is used by	Tool

Table 17: Tool relations

MCP Server

Model Context Protocol (MCP) servers are programs that expose specific capabilities to AI applications through standardized protocol interfaces. Common examples include file system servers for document access, database servers for data queries, GitHub servers for code management, Slack servers for team communication, and calendar servers for scheduling.[3] Table 18 summarizes the attributes of an MCP server that is used by or uses an agent

Attribute	Description
Name	The name of the MCP server
URL	The URL of the MCP server
Version Number	The version number of the MCP server

Table 18: MCP server attributes

Table 19 summarizes the relations of the MCP server.

[3] Model Context Protocol, “Understanding MCP Servers,” https://modelcontextprotocol.io/docs/learn/server-concepts.

Asset	Relation	Asset
MCP Server	used by	Agent
MCP Server	uses	Agent
MCP Server	used by	Tool
MCP Server	uses	Tool
MCP Server	uses	Data Source
MCP Server	uses	Knowledge Source
MCP Server	uses	Prompt Template
MCP Server	uses	MCP Server
MCP Server	Is used by	MCP Server

Table 19: MCP server relations

Guardrail

AI guardrails are the safeguards that keep AI systems operating safely, responsibly and within defined boundaries. These safeguards encompass policies, technical controls, and monitoring mechanisms that govern how AI agents generate outputs in real-world use cases.[4]

Table 20 summarizes the attributes of the guardrails used by the agent.

Attribute	Description
Name	The name of the guardrail used by the agent (e.g., Prompt Injection, Toxic Content, Model Denial-of-Service)
Description	Description of the guardrail
Model	The name of the model or application that implements the guardrail

Table 20: Guardrail attributes

Table 21 summarizes the relations of the guardrail used by the agent.

Asset	Relation	Asset
Guardrail	is used by	Agent

Table 21: Guardrail relations

[4] IBM, “What are AI guardrails?” https://www.ibm.com/think/topics/ai-guardrails.

AI Use Case

An AI use case is a specific challenge or opportunity that AI may solve.[5] An enumeration of the attributes of an AI use case is beyond the scope of this document. However, Table 22 provides a brief list of the attributes of an AI use case.

Attribute	Description
Identifier	A unique reference ID for the AI use case
Name	The name of the AI use case
Description	Description of the AI use case
Proposed By	Name of the person who proposed the AI use case
Owner Name	Name of the business owner of the AI use case
Function	Name of the business function that owns the AI use case
Problem Statement	Detailed description of the business problem addressed by the AI use case
Expected Benefits	Description of the quantitative and qualitative benefits of the AI use case
Priority	Priority assigned to the AI use case (e.g., Critical, High, Medium, Low)
Status	Status of the AI use case (e.g., Assess, Authorize, Build, Test, Deploy, Completed, Cancelled)

Table 22: AI use case attributes

Table 23 summarizes the relations of the AI use case.

Asset	Relation	Asset
AI Use Case	uses	Agent
AI Use Case	assessed by	Risk Assessment

Table 23: AI use case relations

Application

Table 24 summarizes the attributes of an application that uses an agent. An enumeration of all the application attributes is beyond the scope of this paper.

[5] GSA Center of Excellence, https://coe.gsa.gov/coe/ai-guide-for-government/identifying-ai-use-cases-in-your-organization/.

Attribute	Description
Identifier	A unique reference ID for the application
Name	The name of the application
Description	Description of the application
Business Criticality	The criticality of the application to the operation of the business (e.g., Low, Medium, High)
Emergency Tier	The impact to the business if the application is inoperable (e.g., Non-Critical, Business Critical, Mission Critical)

Table 24: Application attributes

Table 25 summarizes the relations of the application

Asset	Relation	Asset
Application	uses	Agent
Application	assessed by	Risk Assessment

Table 25: Application relations

Business Process

Table 26 summarizes business processes that consume the agent.

Attribute	Description
Identifier	The ID of the business process that uses the agent
Name	The human-readable name of the business process
Description	A brief description of the business process, its significance and relevance
Business Criticality	The criticality of the process to the operation of the business (e.g., Low, Medium, High)

Table 26: Business process details

Table 27 summarizes the relations of the business processes using the agent.

Asset	Relation	Asset
Business Process	uses	Agent
Business Process	assessed by	Risk Assessment

Table 27: Business process relations

Regulation

Table 28 summarizes the attributes of a regulation that are relevant from an agent perspective. A full enumeration of the regulation attributes is beyond the scope of this paper.

Attribute	Description
Name	The name of the regulation (e.g., EU AI Act)
Regulatory Authority	The name of the regulatory authority (e.g., European Union)
Jurisdiction	The jurisdiction of the regulation (e.g., European Union, California)

Table 28: Regulation attributes

Table 29 summarizes the relations of the regulation. The impact of regulations on applications and business processes is beyond the scope of this paper.

Asset	Relation	Asset
Regulation	governs	AI use case
Regulation	governs	Risk Assessment

Table 29: Regulation relations

Risk Assessment

Table 30 summarizes the attributes of a risk assessment. A full enumeration of the attributes of a risk assessment is beyond the scope of this paper.

Attribute	Description
Identifier	The ID of the risk assessment

Name	The name of the risk assessment (e.g., Agent-02-2Q26)
Assessor	The name of the assessor
Date	The date of the risk assessment
Risk Score	The risk score assigned to the subject (application, business process, AI use case, agent) of the risk assessment
State	Status of the risk assessment (e.g., Ready to Take, In Progress, Completed, Cancelled)

Table 30: Risk assessment attributes

Table 31 summarizes the relations of the risk assessment.

Asset	Relation	Asset
Risk Assessment	is governed by	Regulation
Risk Assessment	assesses	AI Use Case
Risk Assessment	assesses	Application
Risk Assessment	assesses	Business Process
Risk Assessment	assesses	Agent

Table 31: Risk assessment relations

Agent Metadata for Medical Device Industry

AI agents may be integrated with standalone software programs or with physical hardware. The U.S. Food & Drug Administration (FDA) defines Software as a Medical Device broadly to include software that allows a smartphone to view images obtained from a magnetic resonance imaging (MRI) medical device for diagnostic purposes and Computer-Aided Detection (CAD) software that performs image post-processing to help detect breast cancer. Software as a Medical Device may be interfaced with other medical devices, including hardware medical devices, and other software as a medical device software, as well as general purpose software.[6]

Figure 7 shows key areas where the overall agent data model needs to be customized for the medical device industry.

[6] U.S. Food & Drug Administration, "What are examples of Software as a Medical Device?" https://www.fda.gov/medical-devices/software-medical-device-samd/what-are-examples-software-medical-device

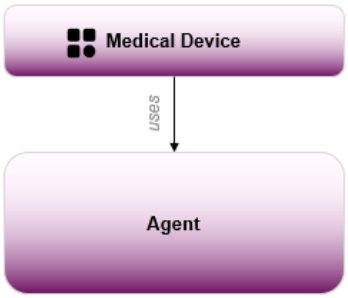

Figure 7: Key aspects of agent metadata for the medical device industry

Table 32 summarizes the attributes of an agent that are specific to the medical device industry.

Attribute / Relation	Description
Agent ID	A unique, permanent identifier for the agent
Name	Name of the agent (e.g., Sepsis ImmunoScore)
Is Medical Device	Boolean indicating if the agent constitutes a medical device
Universal Device Identifier (UDI)	A unique numeric or alphanumeric code that generally consists of the following:[7] ● Device identifier (DI), a mandatory, fixed portion of a UDI that identifies the labeler and the specific version or model of a device ● Production identifier (PI), a conditional, variable portion of a UDI that identifies one or more of the following when included on the label of a device: ○ Lot or batch number within which a device was manufactured ○ Serial number of a specific device ○ Expiration date of a specific device ○ Date a specific device was manufactured ○ Distinct identification code required for a human cell, tissue, or cellular and tissue-based product regulated as a device
CMMS Number	Computerized Maintenance Management System (CMMS) Number ex. Asset Information Management System (AIMS) Number
FDA Device Classification	Software device to aid in the prediction or diagnosis of sepsis
FDA Device Classification Description	Description of the FDA classification. For example: A software device to aid in the prediction or diagnosis of sepsis uses advanced algorithms to analyze patient specific data to aid health care providers in the prediction and/or diagnosis of sepsis. The device is intended for adjunctive use and is not intended to be used as the sole determining factor in assessing a patient's sepsis status. The device may contain alarms that alert the care provider of the patient's status. The device is not intended to monitor response to treatment in patients being treated for

[7] U.S. Food & Drug Administration, "UDI Basics," https://www.fda.gov/medical-devices/unique-device-identification-system-udi-system/udi-basics.

	sepsis.[8]
Uses PII	Boolean indicating if the agent uses Personally Identifiable Information (PII)
Uses PHI	Boolean indicating if the agent uses Protected Health Information (PHI)

Table 32: Agent attributes for the medical device industry

Table 33 summarizes the attributes of a device that are specific to the medical device industry.

Attribute / Relation	Description
Physical State	Physical state of the medical device (e.g., Software Device)

Table 33: Medical device attributes for the medical device industry

Table 34 summarizes the additional relations in the medical device industry

Asset	Relation	Asset
Agent	is used by	Medical Device
Medical Device	uses	Agent

Table 34: Additional relations for the medical device industry

Outstanding Items

Table 35 consolidates outstanding items that will be considered in future releases.

Issue	Description	Date	Raised by
1. OpenAI Spec	Is OpenAI working on an overall Agent Metadata Spec https://platform.openai.com/docs/guides/agents]	Jan 14, 2026	Sunil Soares
2. Business Process	Should benefits be included?	Jan 12, 2026	Tony DiPerna
3. Agent Type	Under Configuration or Application, should the AI Agent type (3rd party or homegrown) be included?	Jan 12, 2026	Tony DiPerna

[8] U.S. Food & Drug Administration, "Product Classification," https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfpcd/classification.cfm?id=2354.

4. Agent Configuration	We don't mention security/vulnerability enterprise impacts or that's considered as part of the Discovery or Risk challenges?	Jan 12, 2026	Tony DiPerna
5. Agent Identification	What's important aside from version is to know when the agent was installed/deployed?	Jan 12, 2026	Tony DiPerna
6. Agent Behavior	Behavior is the missing layer. Once ontology and taxonomy are in place, behavior becomes observable. This is where governance becomes real. The key questions the current document does not address are whether data access aligns with historical role behavior, whether an agent is acting within its normal operational patterns, and whether usage is expected or anomalous. This is where RBAC evolves from static permissions into behavior-aware access control.	Jan 15, 2026	Doug Shannon
7. Payments	Google Agent Payments Protocol 2 (AP2)	Jan 14, 2026	Sanjeev Varma
8. E-Commerce	Google Universal Commerce Protocol (UCP)	Jan 14, 2026	Sanjeev Varma
9. Banking	Agent metadata for model risk management: Federal Reserve Guidance on Model Risk Management SR 11-7 Federal Reserve Guidance on Stress Testing for Banking Organizations with More Than $10 Billion in Total Consolidated Assets (Data Lineage)	Jan 16, 2026	Sunil Soares
10. Healthcare	PHI, PII, agents as medical devices, HITRUST	Jan 16, 2026	Mike Jennings
11. Agent Version	Add agent version as metadata	Jan 22, 2025	Sanjeev Varma
12. Controls	Add controls, inherent and residual risk to agents	Jan 23, 2025	Bryan Swann

Table 35: Outstanding issues