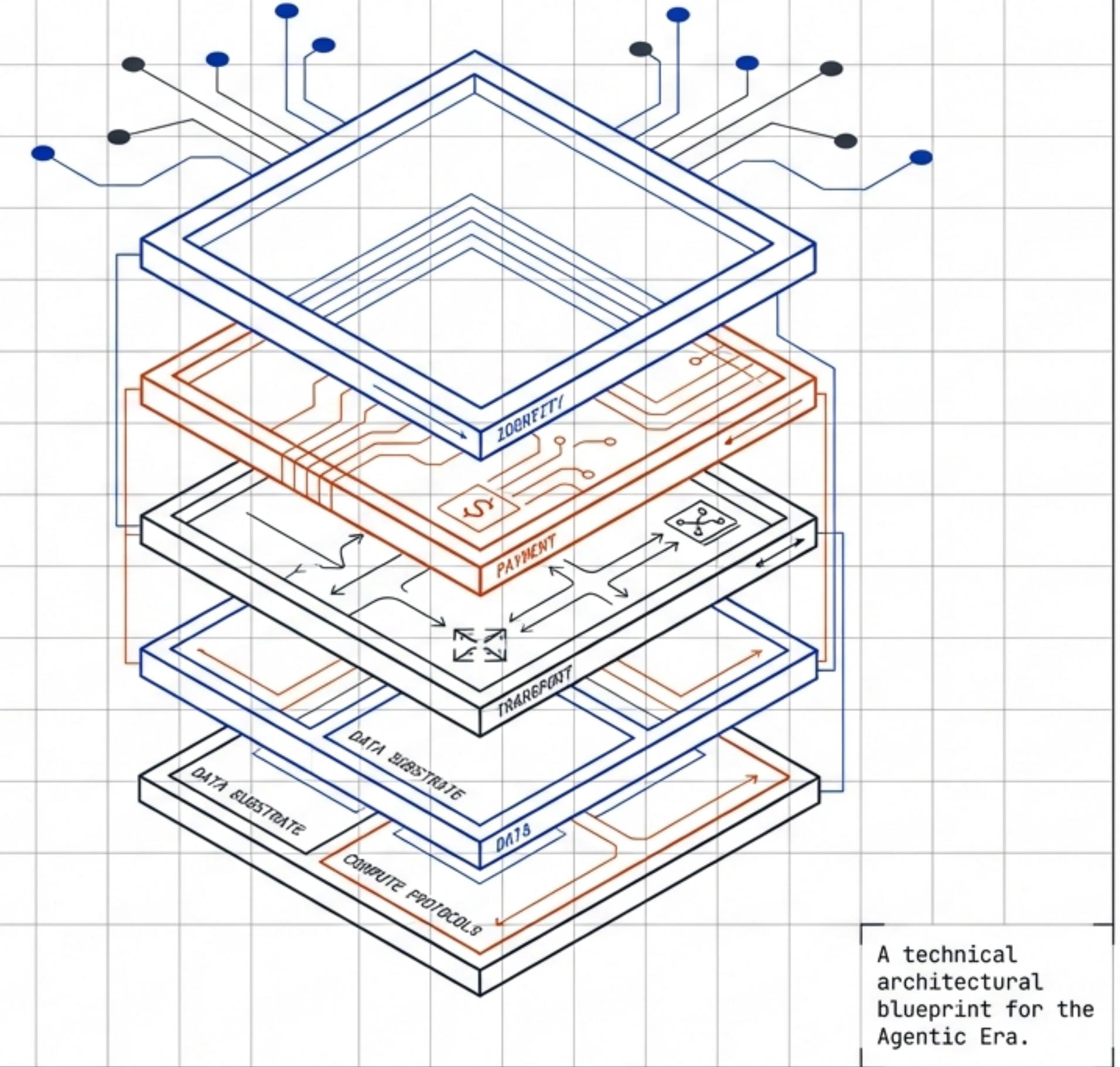


The Agentic Web Stack

**Protocols, Standards,
and the Infrastructure
of Autonomy**

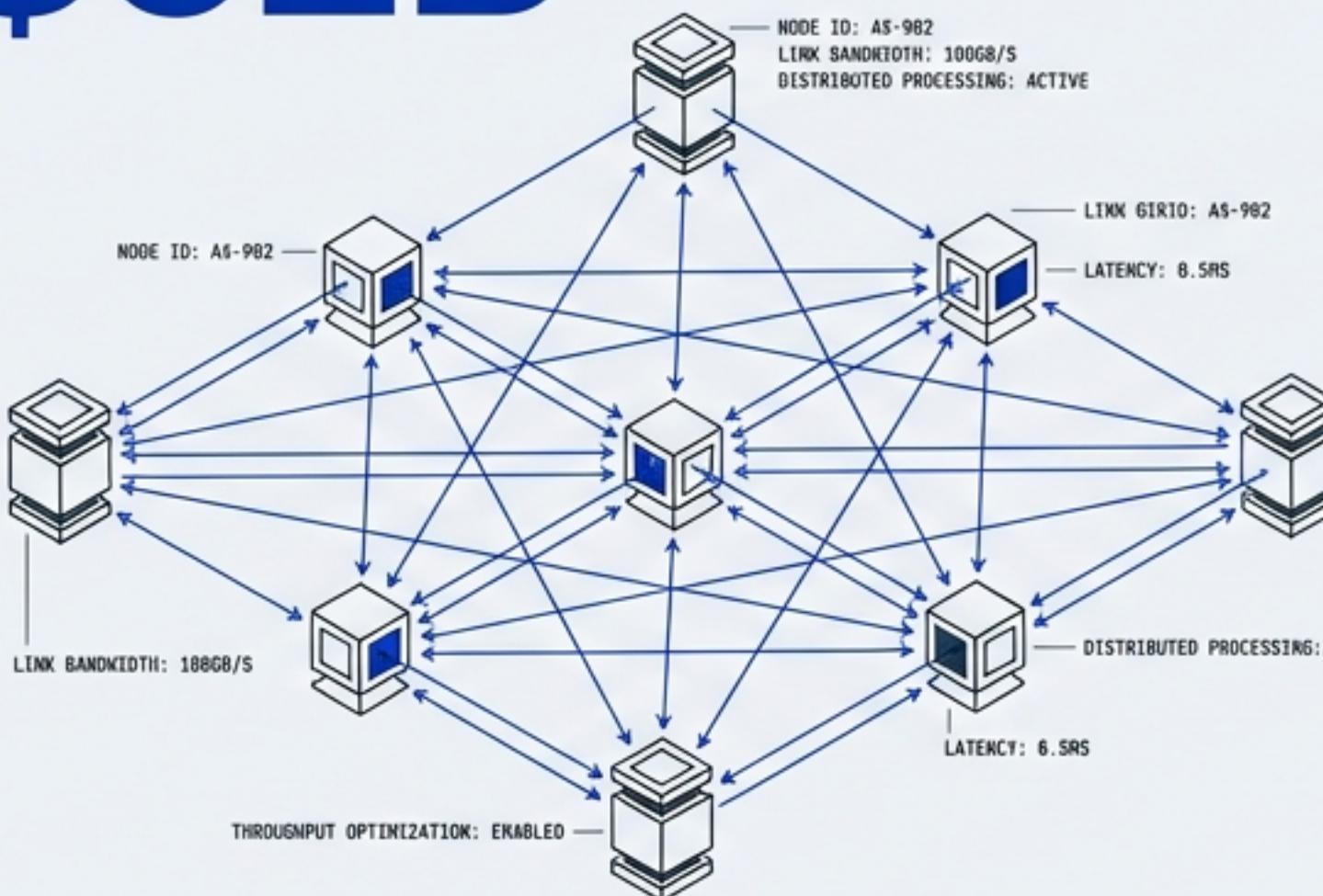
Moving beyond chatbots to a distributed substrate of interoperable, autonomous software entities.



We are running distributed intelligence on centralized infrastructure.

MARKET PROJECTION

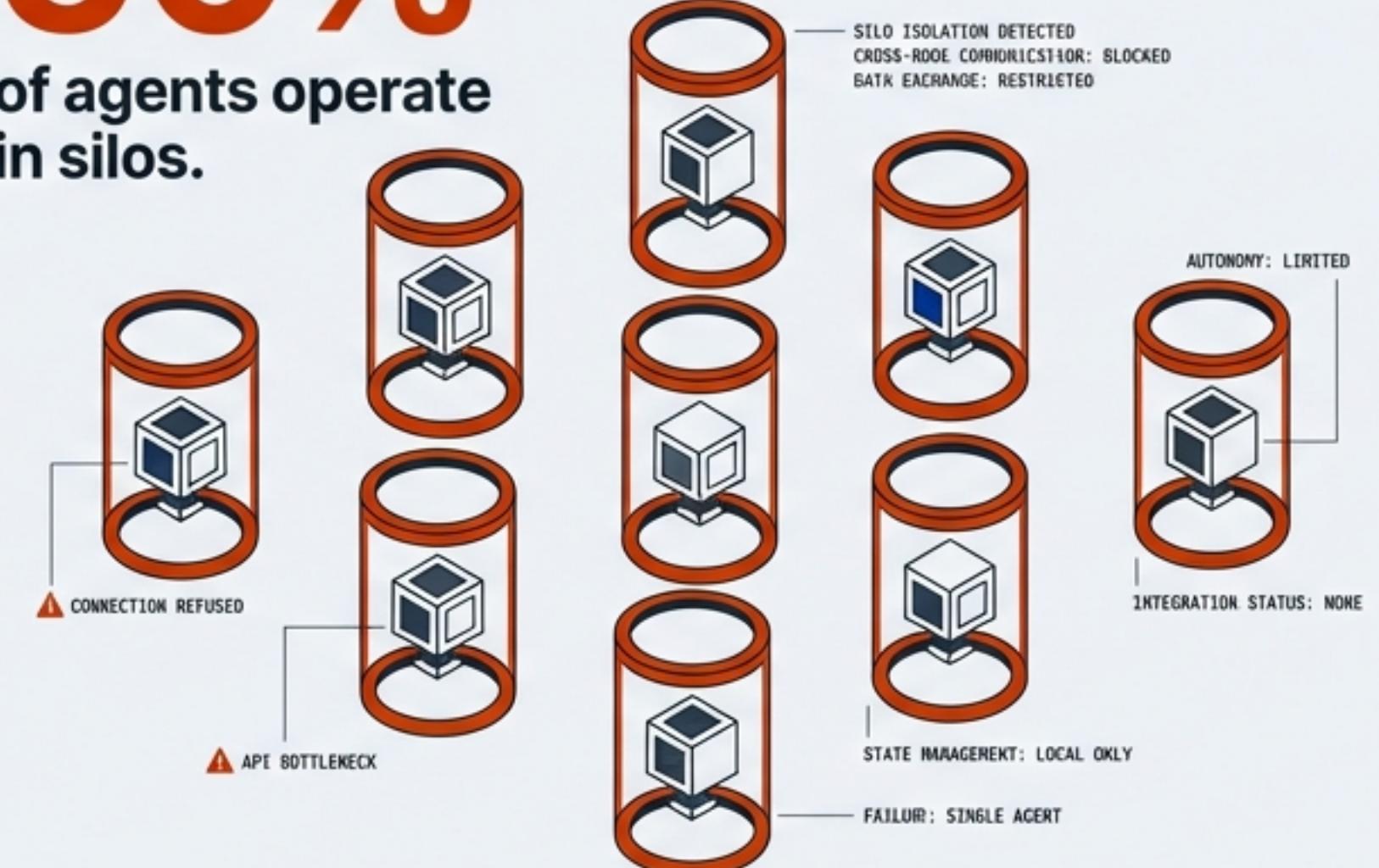
\$52B



CURRENT REALITY

50%

of agents operate
in silos.



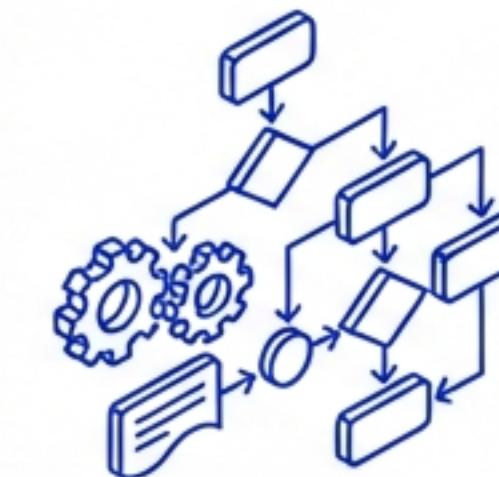
Most enterprise APIs are built on request-response loops. Agentic workflows require state, negotiation, and autonomy. [JetBrains](#). We are currently funneling distributed intelligence through centralized chokepoints.

The shift from 'Chatbot' to 'Agent' requires a new connectivity substrate



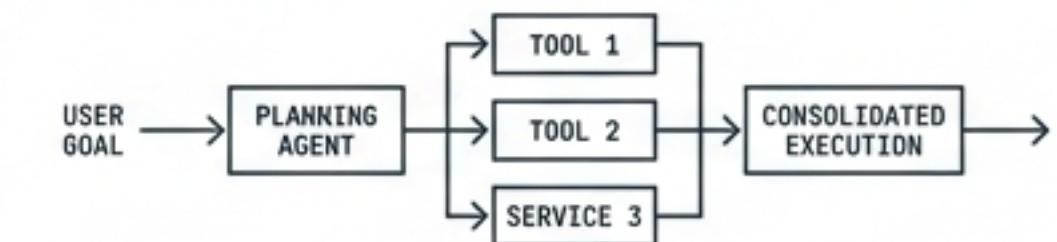
LLM / CHATBOT

PATTERN: Prompt & Response

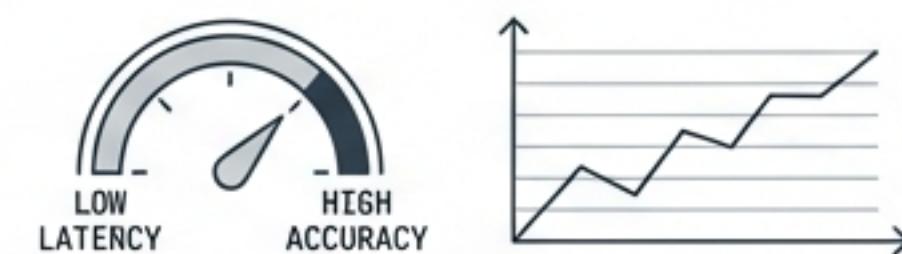


AGENTIC SYSTEM

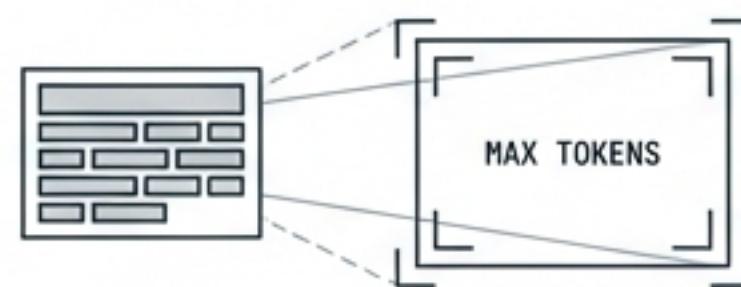
PATTERN: Plan & Execute



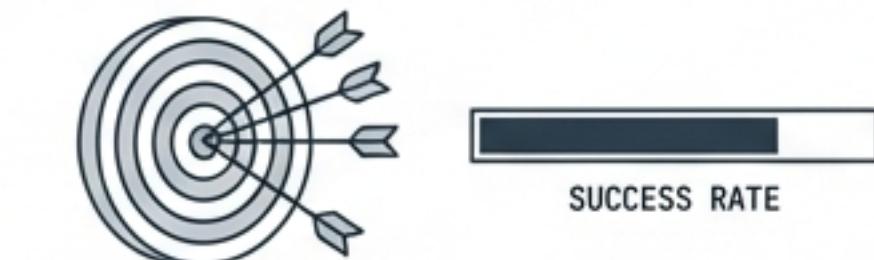
METRIC: Latency / Accuracy



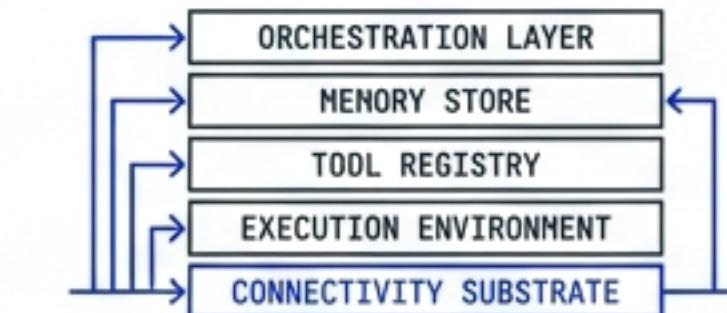
REQUIREMENT: Context Window



METRIC: Goal Completion



REQUIREMENT: The Agentic Stack

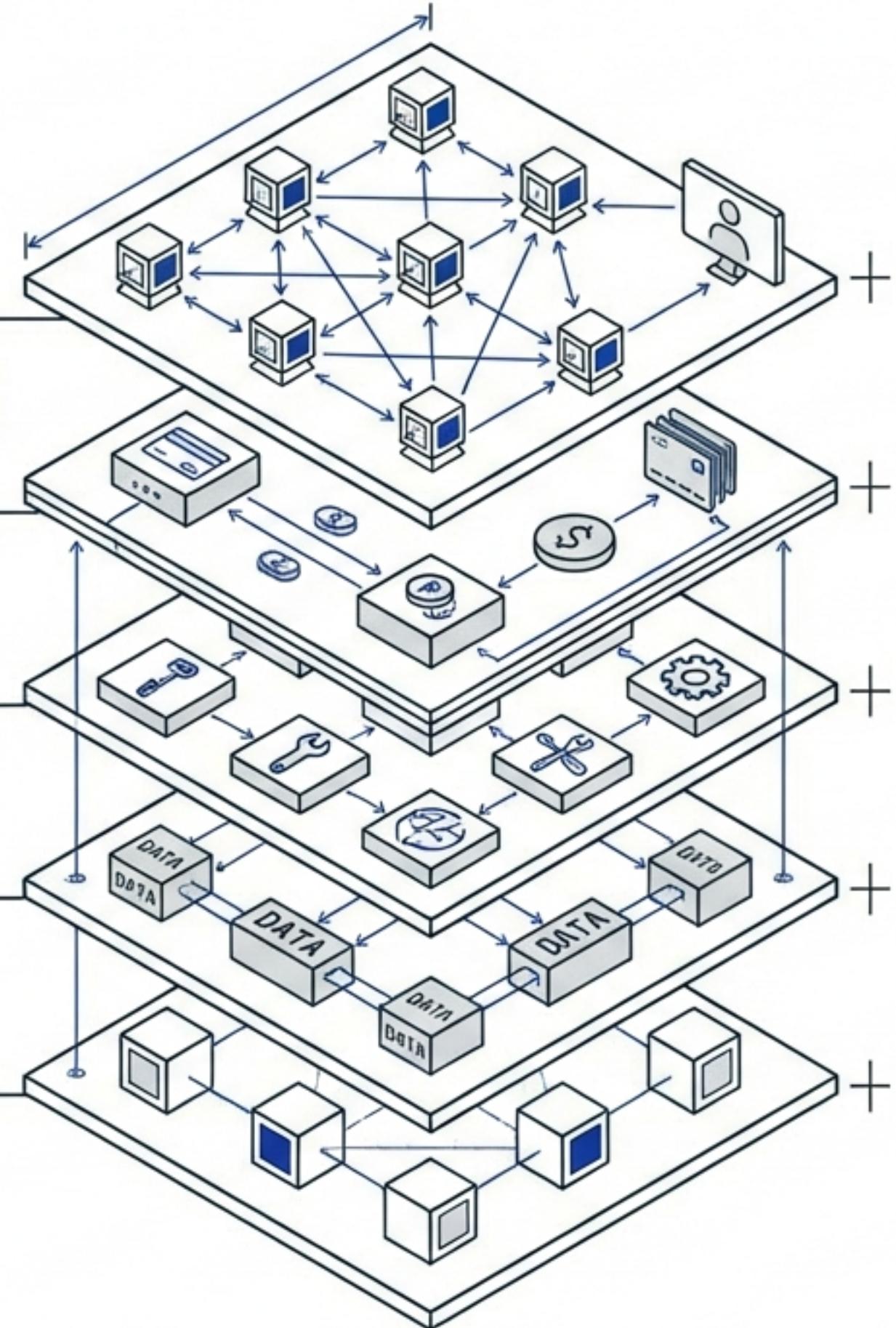


⚠ ALERT: Agents turn software into a chain of micro-services. Micro-services demand micro-transactions and standardized context. The 'Substrate' is missing.

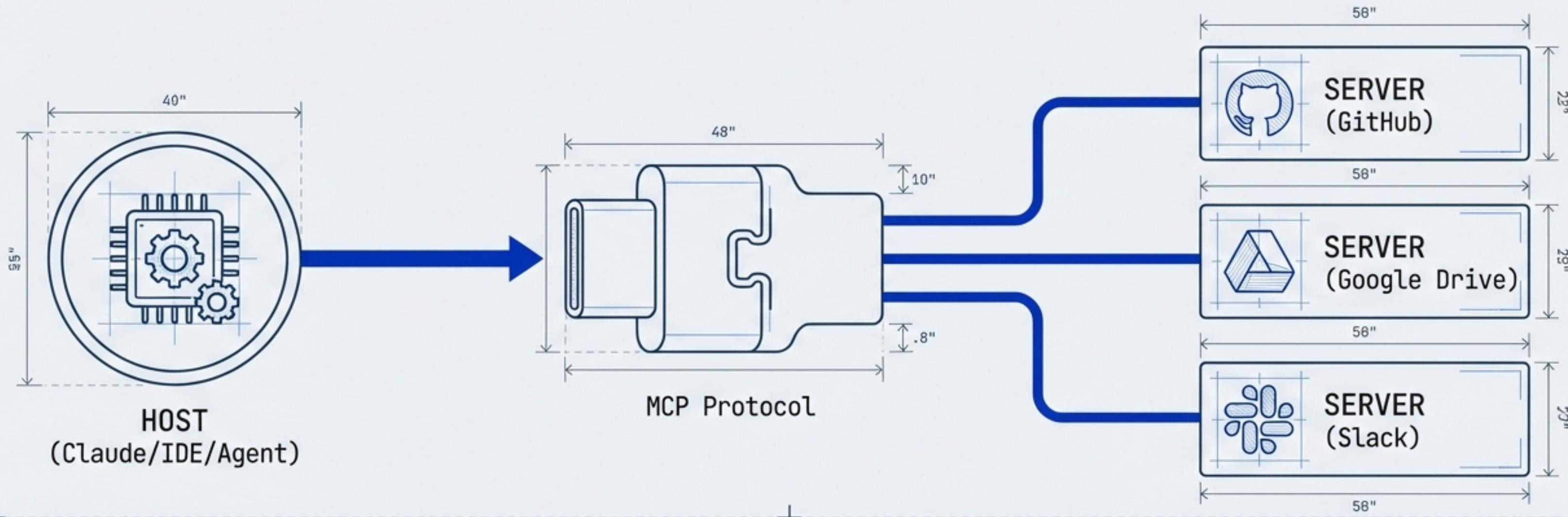
The Architectural Blueprint of the Agentic Web.

A robust agentic web stack is not one technology but a composition of services. It moves from centralized control planes to distributed, interoperable protocols.

- 5 **ORCHESTRATION & APPLICATION**
Swarms, Human-in-the-loop
- 4 **COMMERCE & SETTLEMENT**
x402, Micropayments
- 3 **CONTEXT & TOOLS**
Model Context Protocol - MCP
- 2 **COMMUNICATION & TRANSPORT**
Agent-to-Agent - A2A
- 1 **IDENTITY & TRUST**
DIDs, ERC-8004



Layer 3: Context & Tools — The Model Context Protocol (MCP)

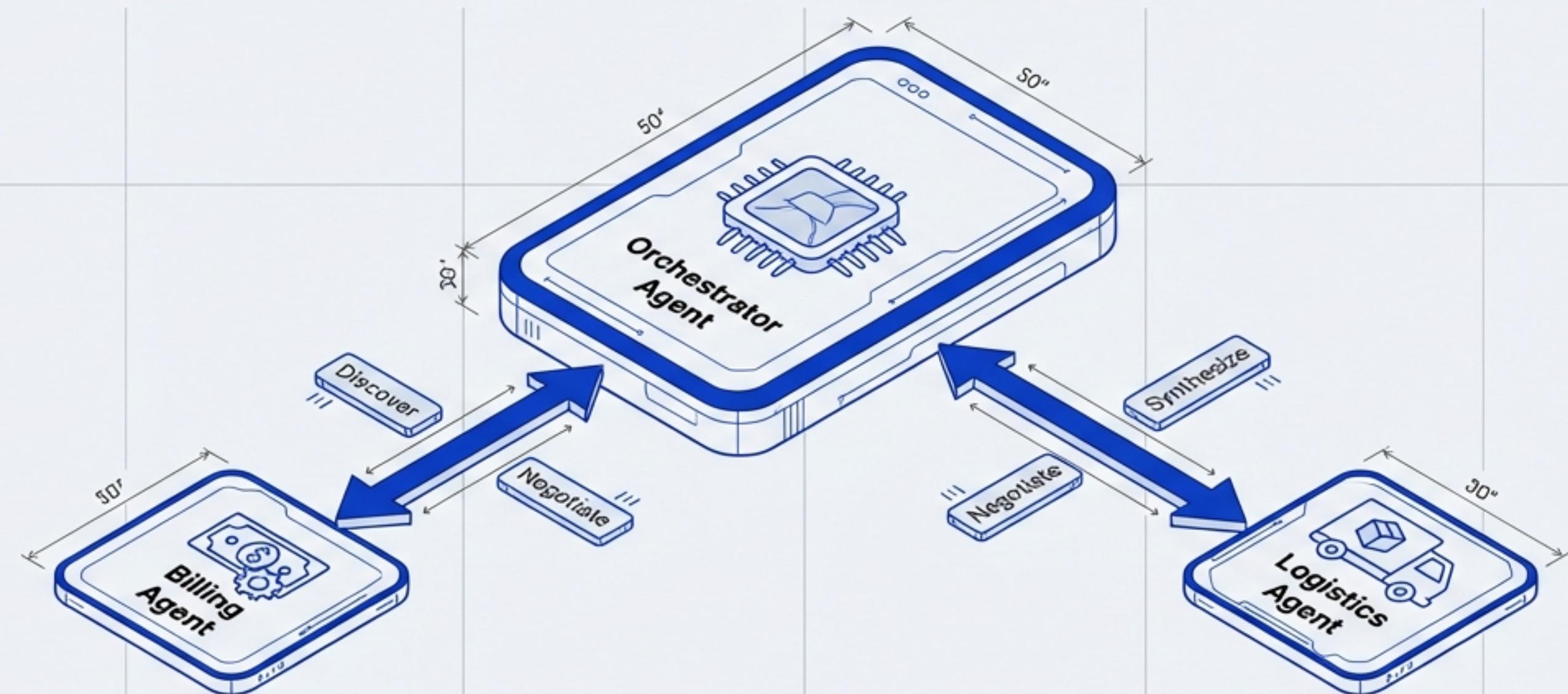


⚠ Problem: Developers currently build brittle, point-to-point integrations for every tool.

Solution: MCP standardizes how agents connect to data. Instead of calling tools directly, an agent connects to "Servers." This allows data and services to be packaged as modular code reusable across any AI system.

Layer 2: Communication – Agent-to-Agent (A2A) Protocols

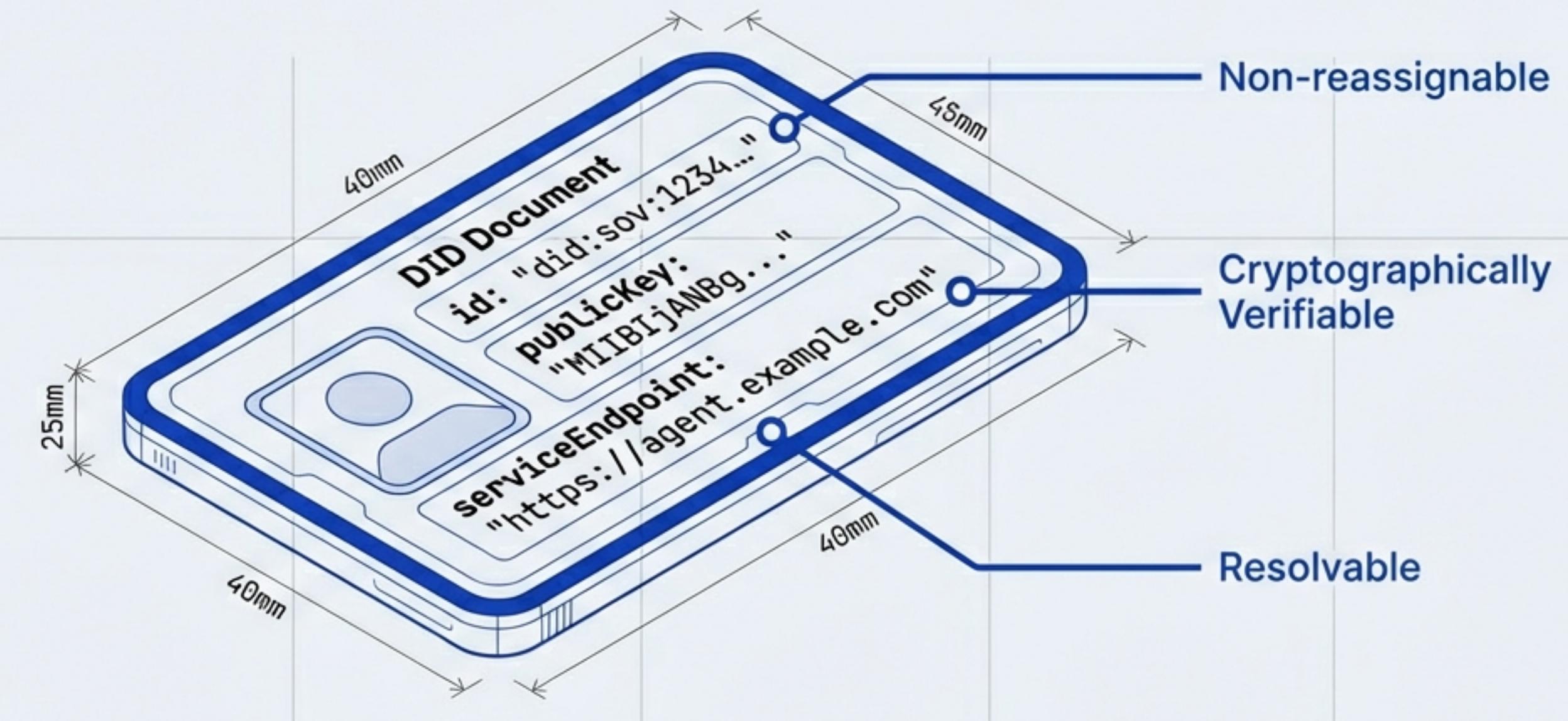
Moving beyond ‘API Integration’ to ‘Negotiation and Coordination.’ A2A allows agents to discover peers with specific skills, negotiate task parameters, and synthesize findings from multiple sources. Without a shared language, collaboration breaks down into isolated silos.



Layer 1: Identity — Decentralized Identifiers (DIDs).

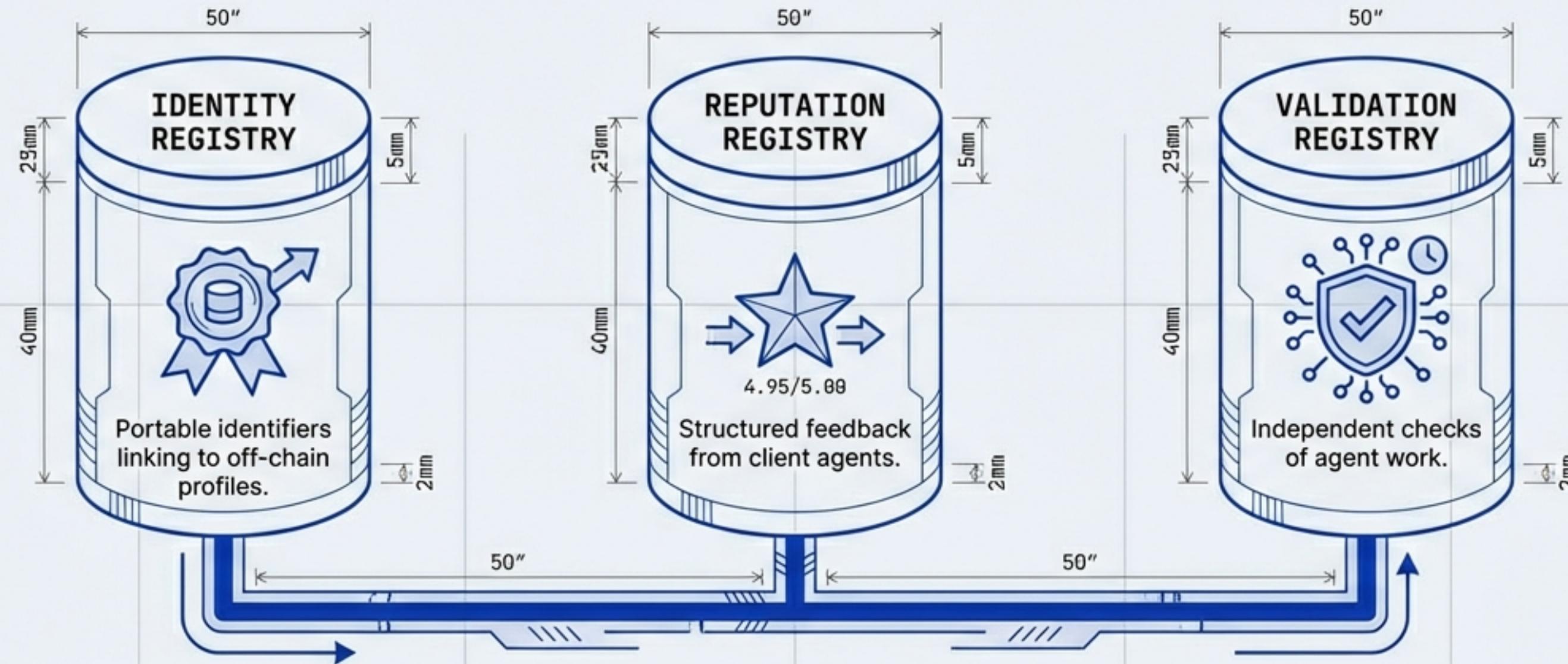
The move from email-based IDs to permanent, self-sovereign identities. DIDs allow agents to authenticate and encrypt communications without a platform in the middle.

⚠ Alert



Layer 1: Trust — The ERC-8004 Standard.

A minimal trust layer for agents to discover and validate each other.



Key Insight: ERC-8004 keeps identifiers and trust signals on-chain (control plane) while keeping detail-heavy artifacts off-chain (data plane).

Layer 4: Commerce — x402 and The Machine Economy.

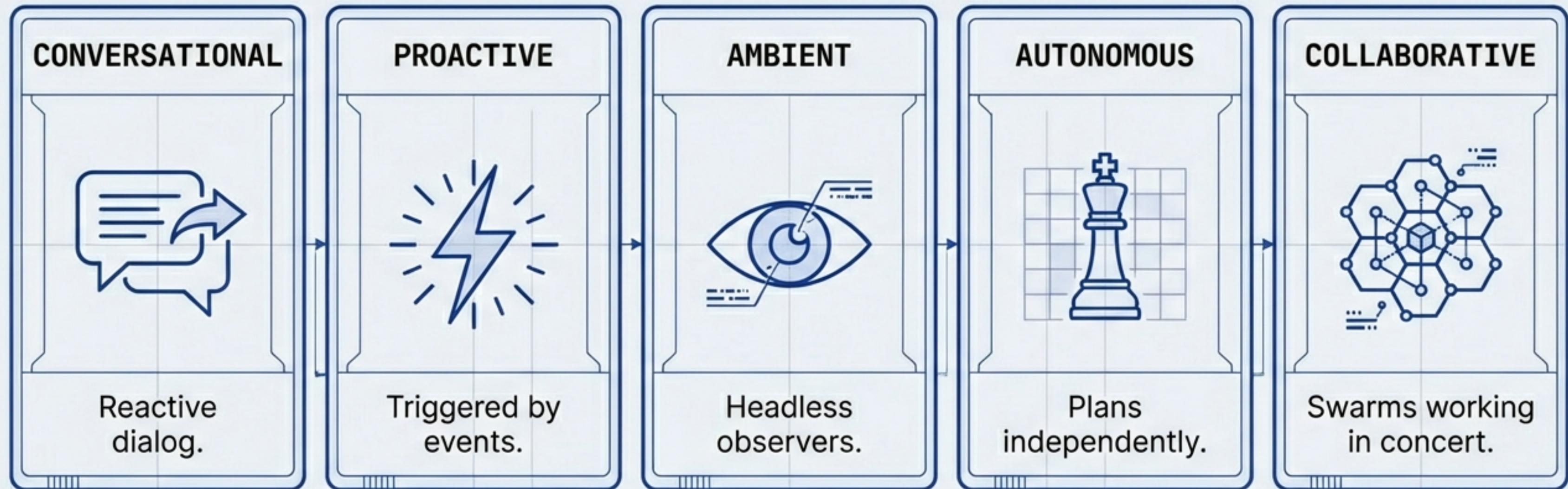
Reviving HTTP 402 for agent-to-agent micropayments. Agents can buy one premium data pull instead of subscribing.



Use Cases

- [✓] Pay-per-inference
(Specialized models)
- [✓] Pay-per-scrape
(Premium datasets)
- [✓] No subscriptions
(Pay for single actions)

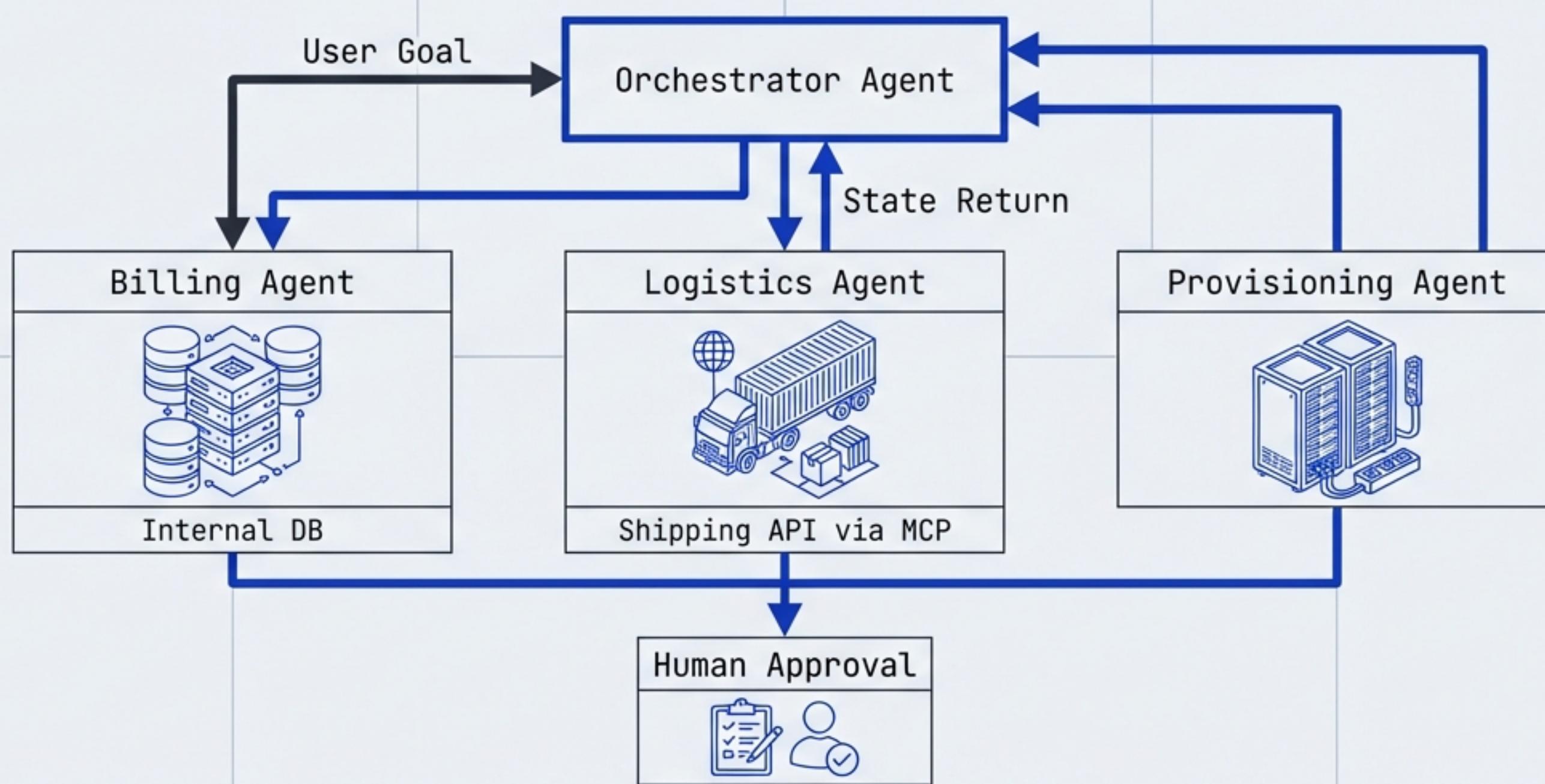
The Taxonomy of Agentic Systems.



The ‘Stack’ enables these agents to move from isolated tools to a connected ecosystem.

Design Pattern in Action: The Orchestrator Swarm.

Scenario: Complex Service Escalation

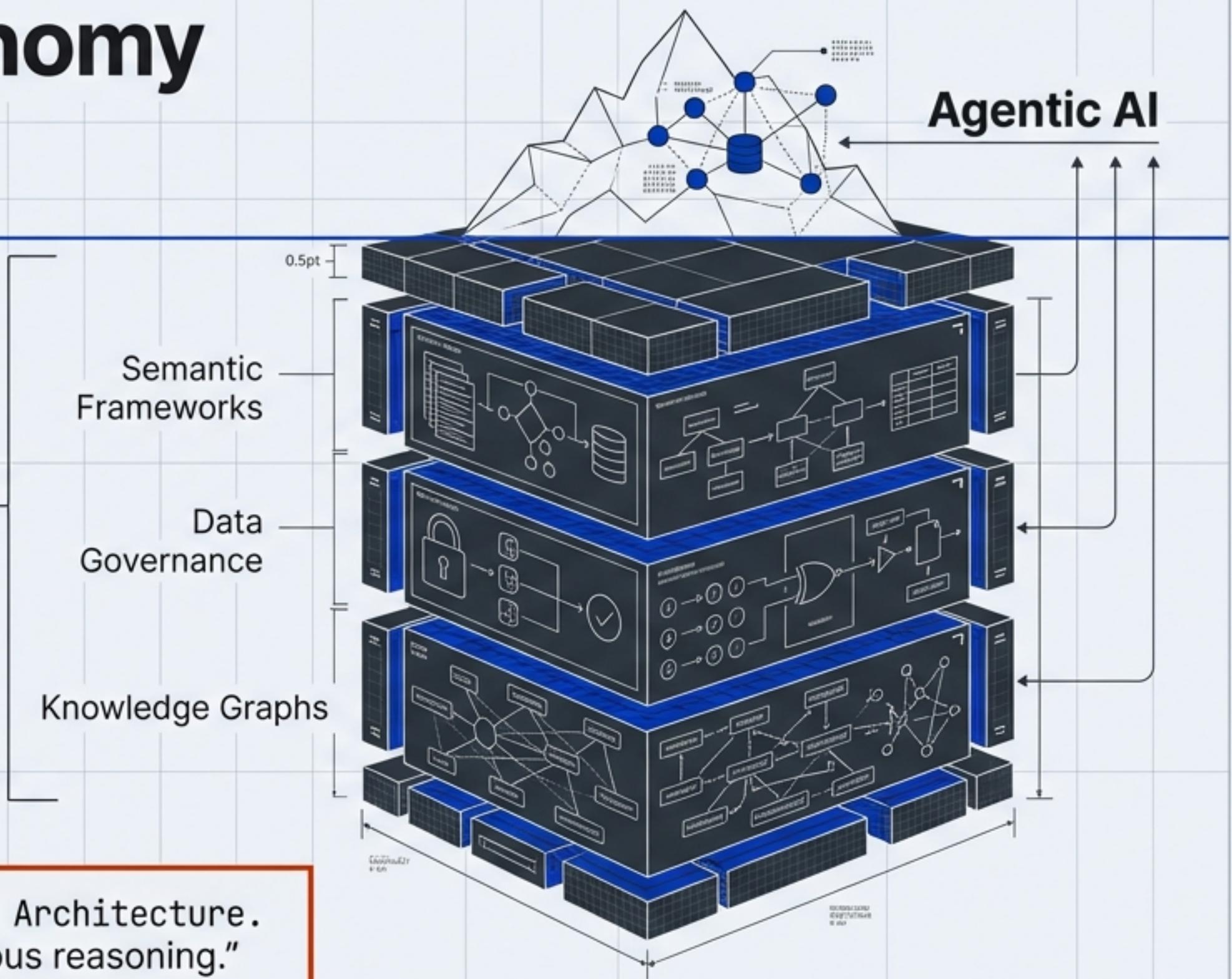


THIS PATTERN REQUIRES A2A FOR COORDINATION, MCP FOR TOOL ACCESS, AND IDENTITY FOR TRUST.

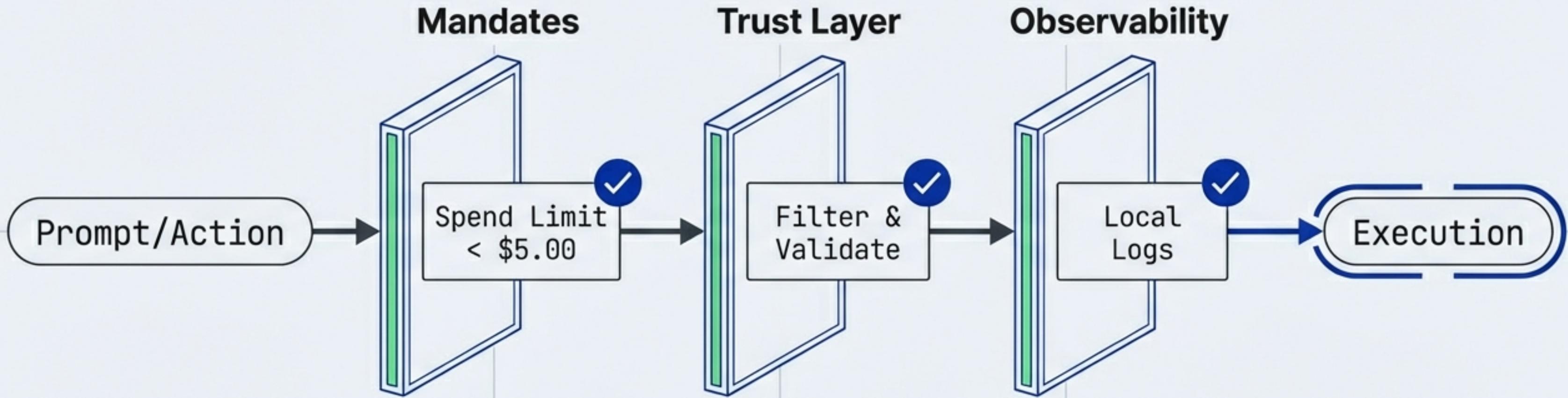
Information Architecture: The Bedrock of Autonomy

Information Architecture

"Agentic AI is only as good as its Information Architecture.
Legacy data architectures cannot support autonomous reasoning."



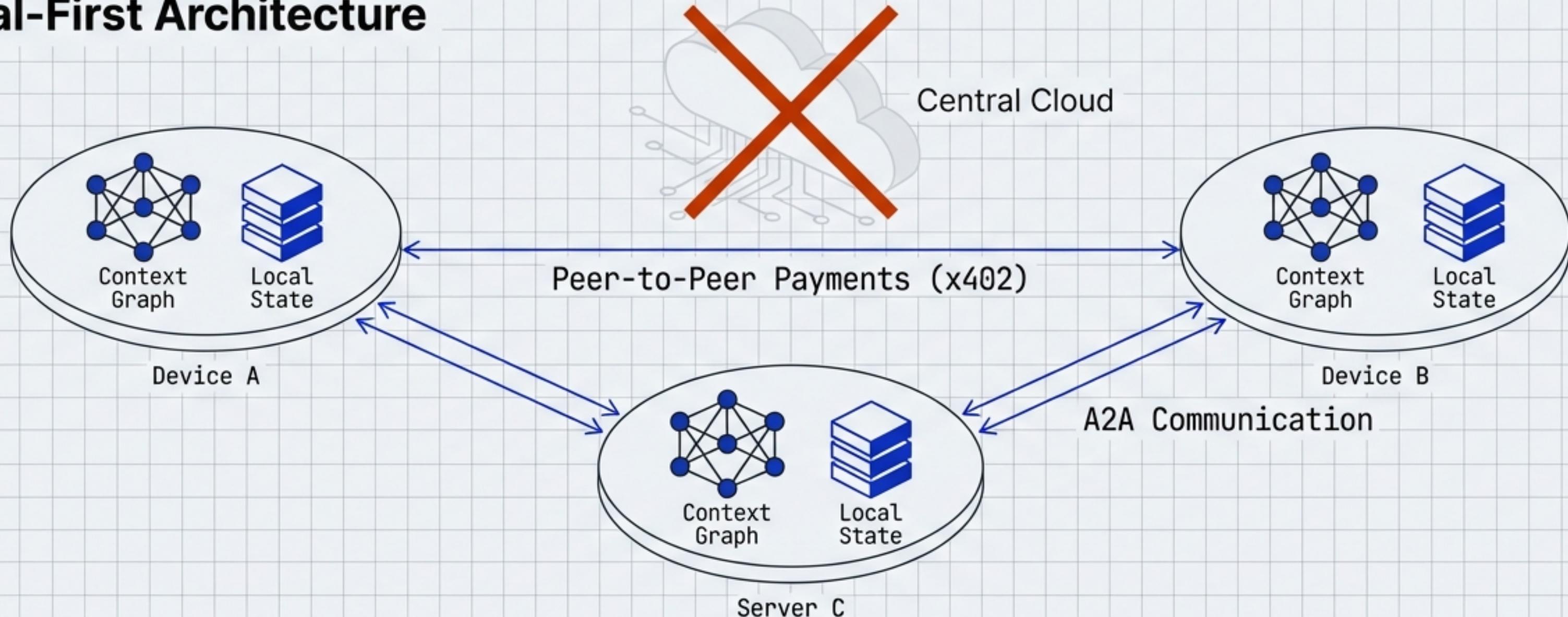
Governance and the ‘Human-in-the-Loop’.



Digital permissions are cryptographically signed.
Guardrails intercept actions.
Observability respects data locality.

The Convergence: Sovereign, Edge, and Agentic

Local-First Architecture



Resolution: Context graphs travel with the agent. Decisions are made based on local state. Agents turn from cloud-tethered clients into sovereign software entities.

From 'Apps' to 'Economic Workflows'.

The Enterprise Roadmap

1. Standardize the Plug: Adopt MCP to avoid integration debt.
2. Verify the Actor: Move to DIDs and ERC-8004 registries.
3. Prepare for Commerce: Audit APIs for x402 readiness.
4. Build the Substrate: Invest in peer-to-peer coordination.

The future isn't about smarter chatbots. It's about a machine economy where software can do business on behalf of its creators.