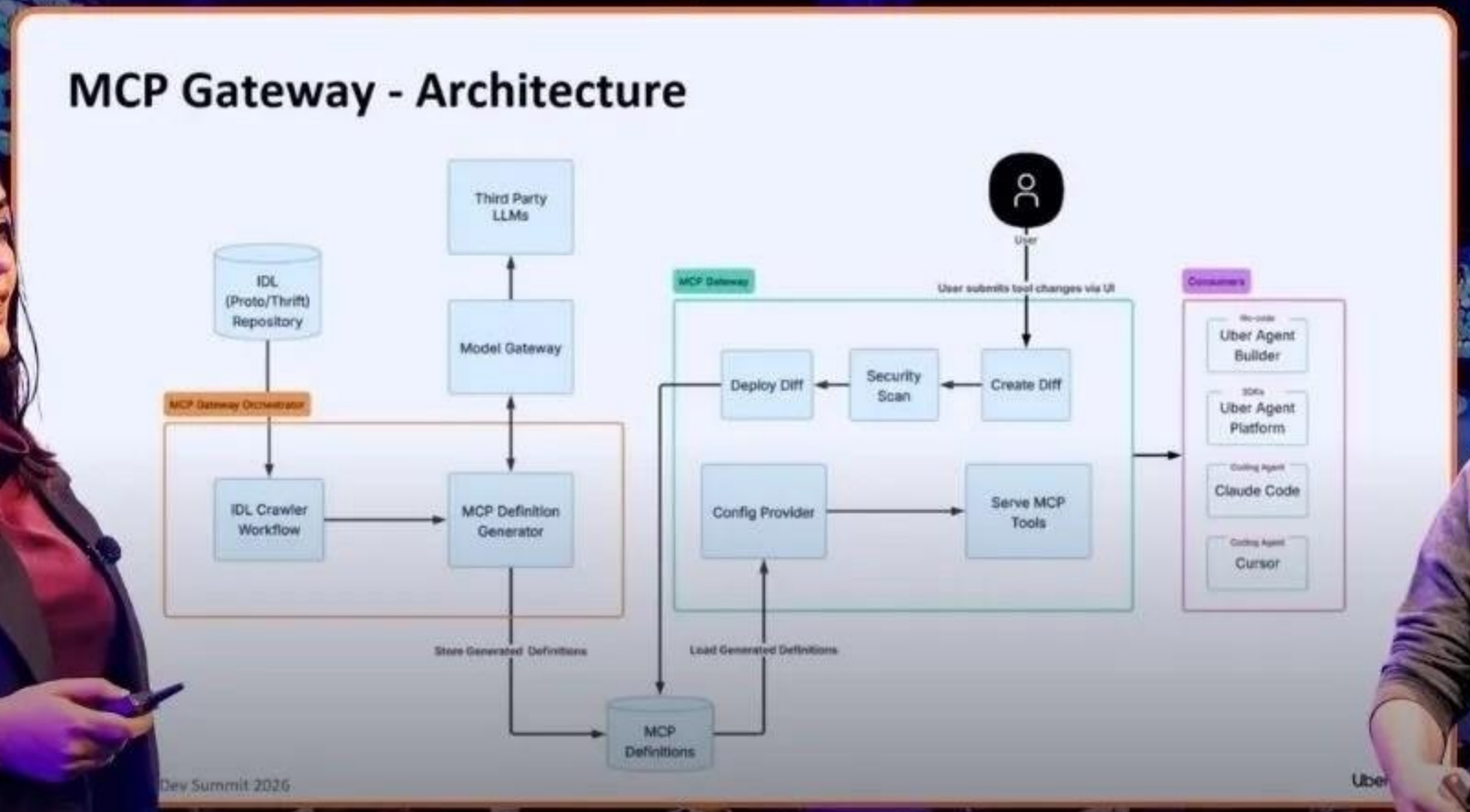
Agentic AI Foundation

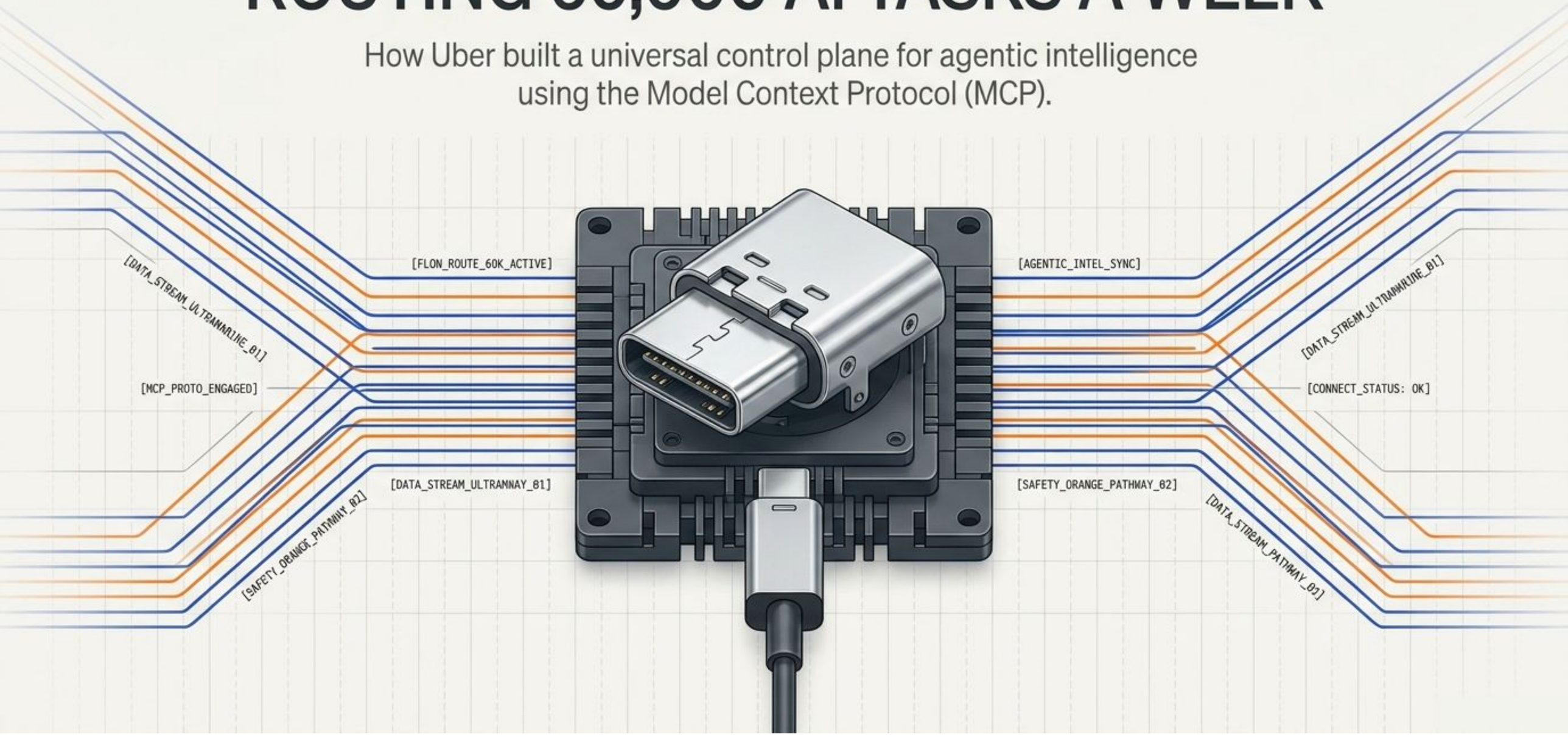
Uber



UBER'S AI SECRETS

ROUTING 60,000 AI TASKS A WEEK

How Uber built a universal control plane for agentic intelligence using the Model Context Protocol (MCP).



Scaling AI Agents: From Protocol to Production with MCP

Universal Agent Connector Hub

AI Models
(Claude, ChatGPT, Cursor, etc.)

The MCP Standard:

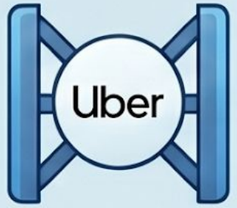
IDEs

Local Files

Broad Ecosystem Support: Build once, integrate across Claude, ChatGPT, Cursor, and IDEs.

Reduced Development Complexity:
Eliminates need for custom integrations for every new agent-service pair.

Enterprise Scaling: The Uber Blueprint



Uber's Centralized Gateway & Registry

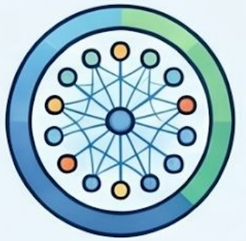
Production-Grade Security:
Centralized governance includes PII redaction, automated code scanning, and strict authorization layers.

60,000+
Weekly Agent Executions

Automates Tool Generation:
Translates 10,000+ service endpoints | tool descriptions automatically.

Uber Scale Metrics

1,500+
Monthly Active Agents
Used by 95% of engineering staff

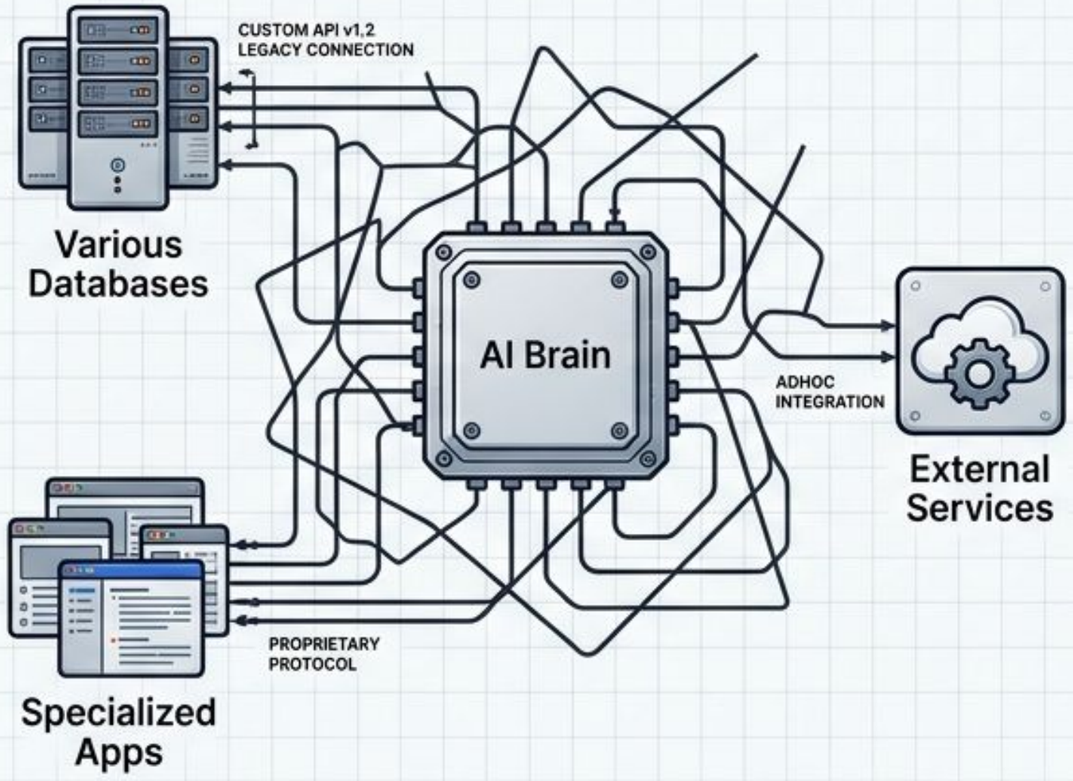


10,000+
Internal Services Connected

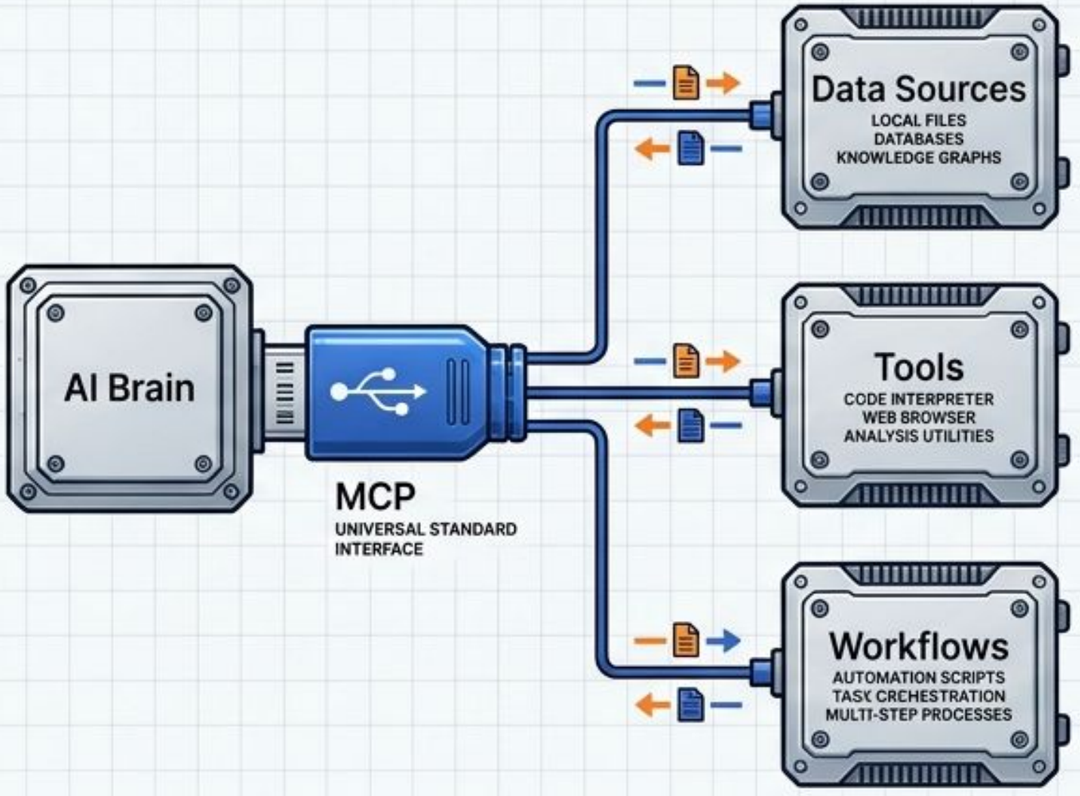
The Universal Port for AI Applications

The Model Context Protocol (MCP) is an open-source standard connecting AI models to external systems. Instead of building bespoke integrations for every service, MCP allows AI applications to connect to local files, databases, and specialized tools through a single, standardized architecture.

Before: Custom API Integrations



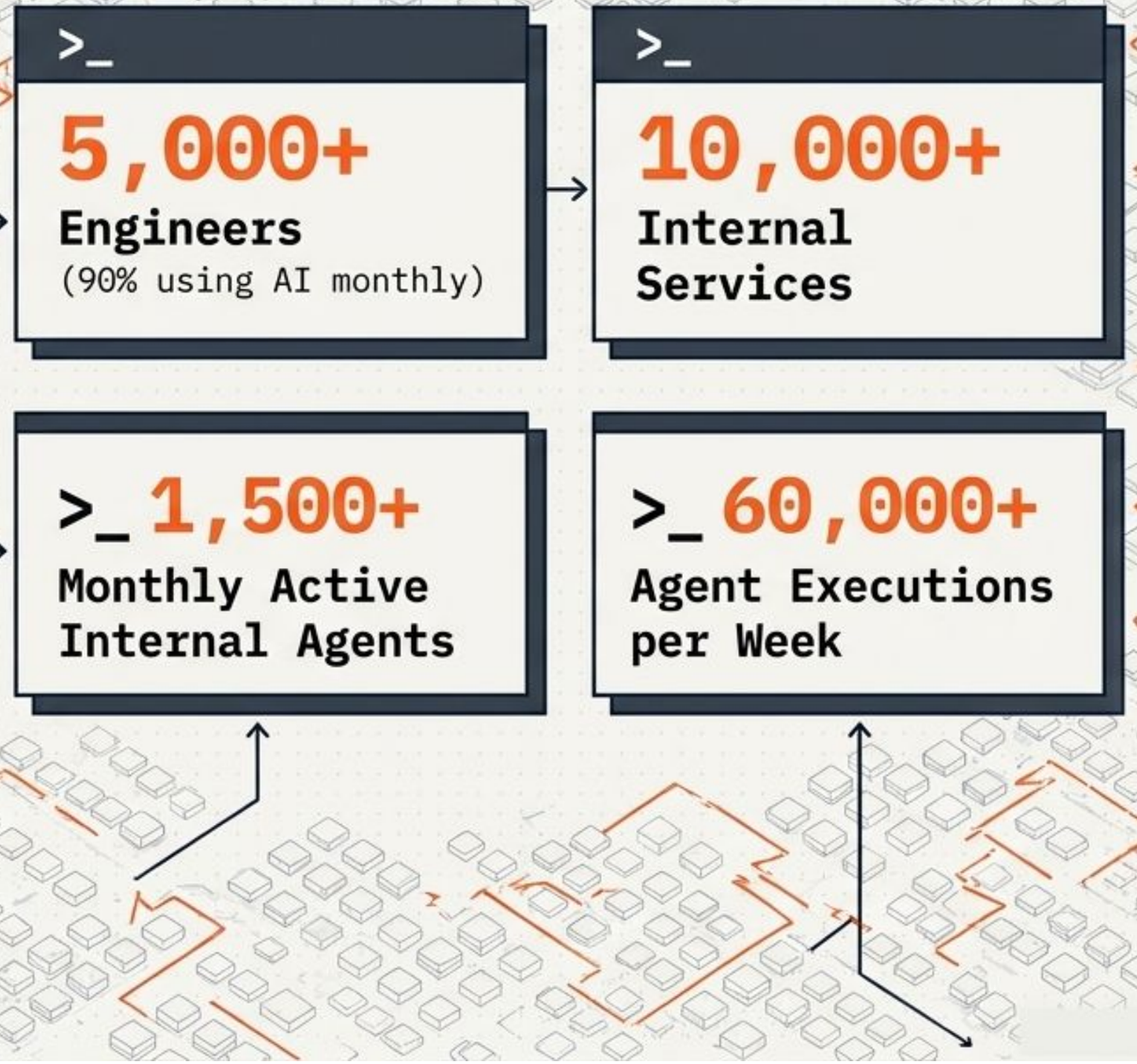
After: The MCP Standard



Hyper-Growth Breaks Standard Workflows

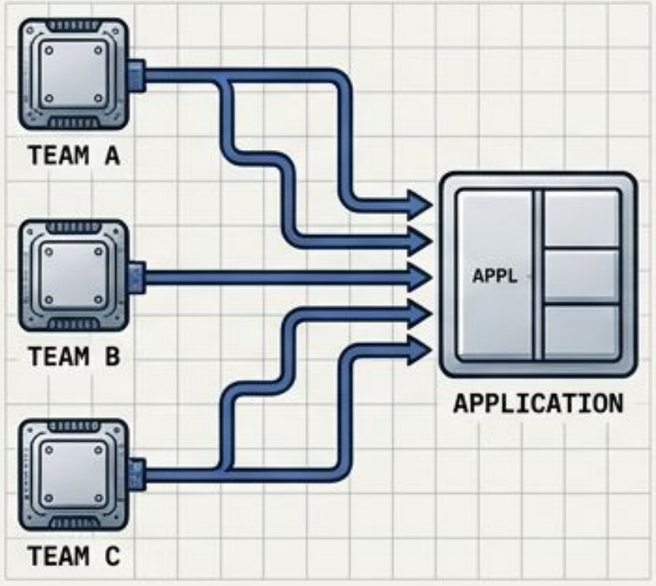
At this scale, this is no longer a pilot program. It is the new standard for work.

But without standardized routing, every agent must rediscover how to interact with thousands of complex, scattered services.



The Friction of Bespoke Integration

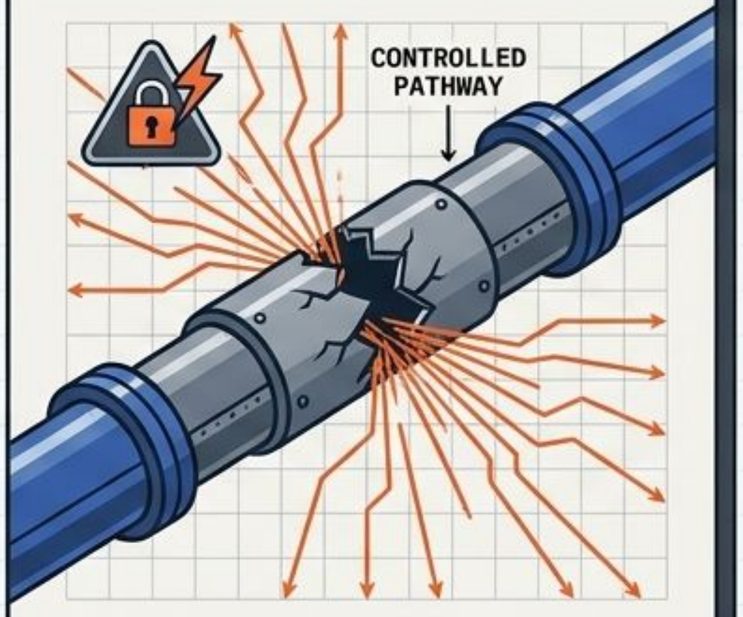
Development Lifecycle



Risk: `Siloed Problem-Solving`

Symptom: Teams building custom, non-reusable integrations independently without central guidance.

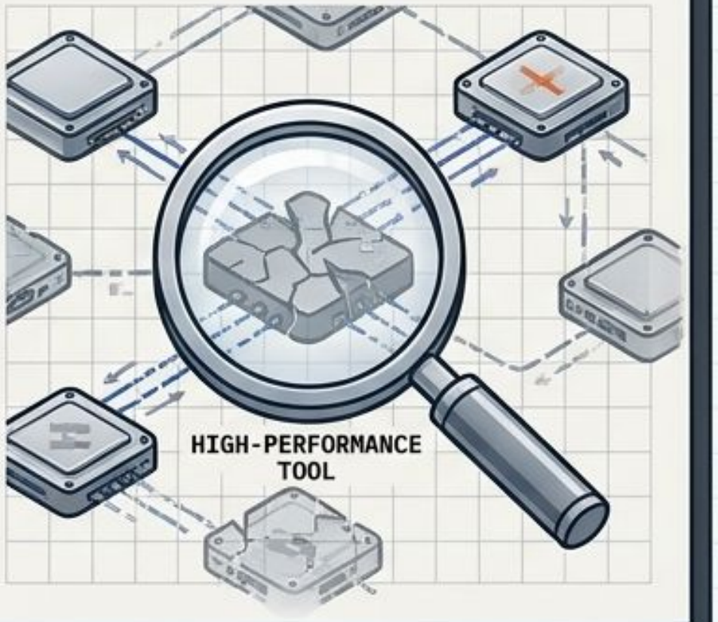
Security & Governance



Risk: `High Blast Radius`

Symptom: Agents execute tasks faster than humans. Bespoke tools create immediate risks of unauthorized data access and unmonitored call patterns.

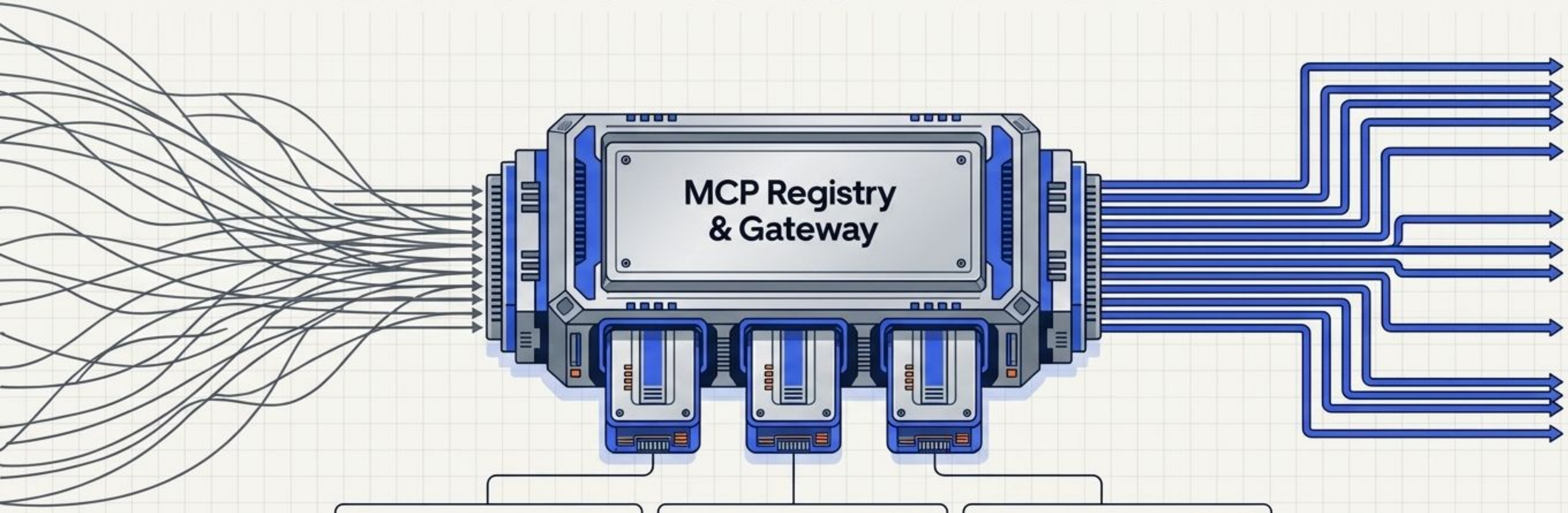
Discovery & Quality



Risk: `Agent Degradation`

Symptom: Engineers cannot reliably find high-performance, safe tools. Failing tools actively degrade overall agent performance.

The Central Control Plane



Config-Driven Standardization

All Uber service endpoints are automatically translated into MCP tools. No more one-off playground environments.

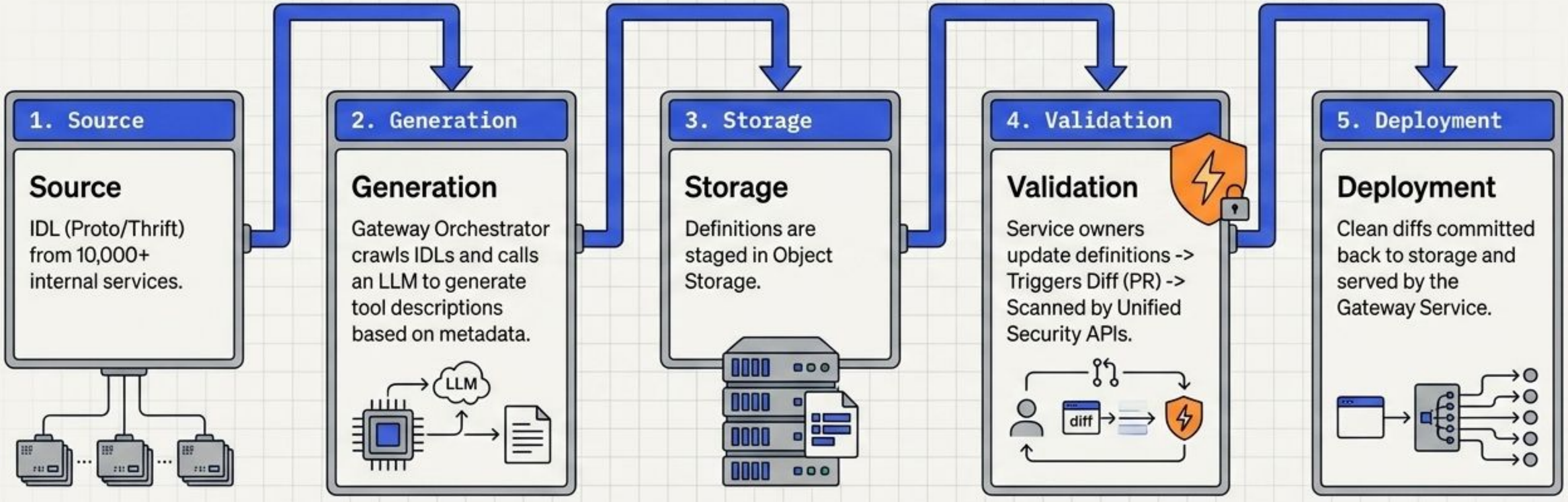
Single Source of Truth

A central registry manages all MCP discovery, versions, and trusted metadata.

Expert Control

Service owners retain absolute control over which tools get exposed and fine-tune tool descriptions specifically for LLMs.

The IDL-to-MCP Automation Pipeline



Zero-Trust Agent Governance

Central Authorization Layer

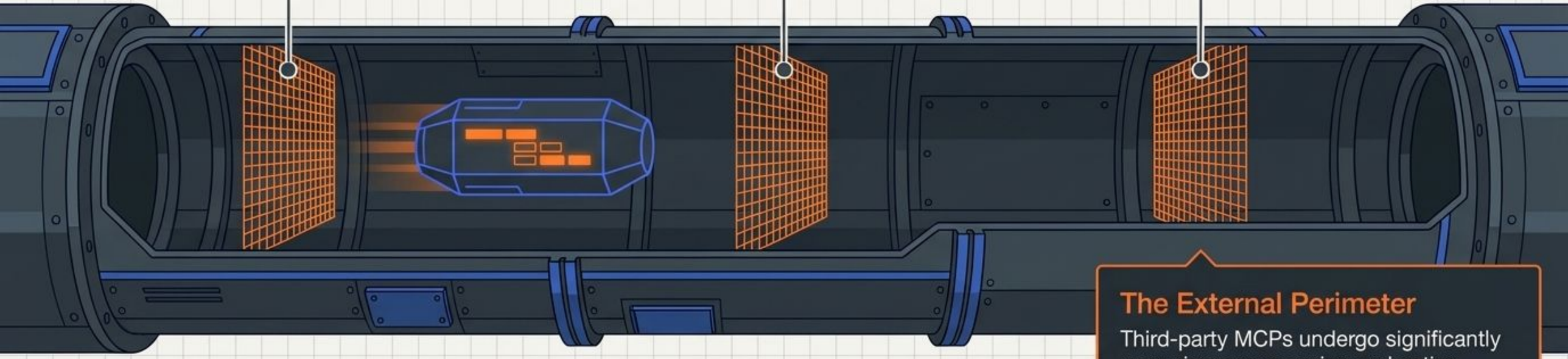
Integrated directly into Uber's auth service. Blockades any mutable endpoints that could bring down critical services.

Continuous Code Scanning

Automated detection of bad patterns, risky tool metadata, or unknown endpoint exposures at both diff commit time and via periodic checks.

PII Redactor Service

Automatic, inline redaction of sensitive data before the agent processes the context.



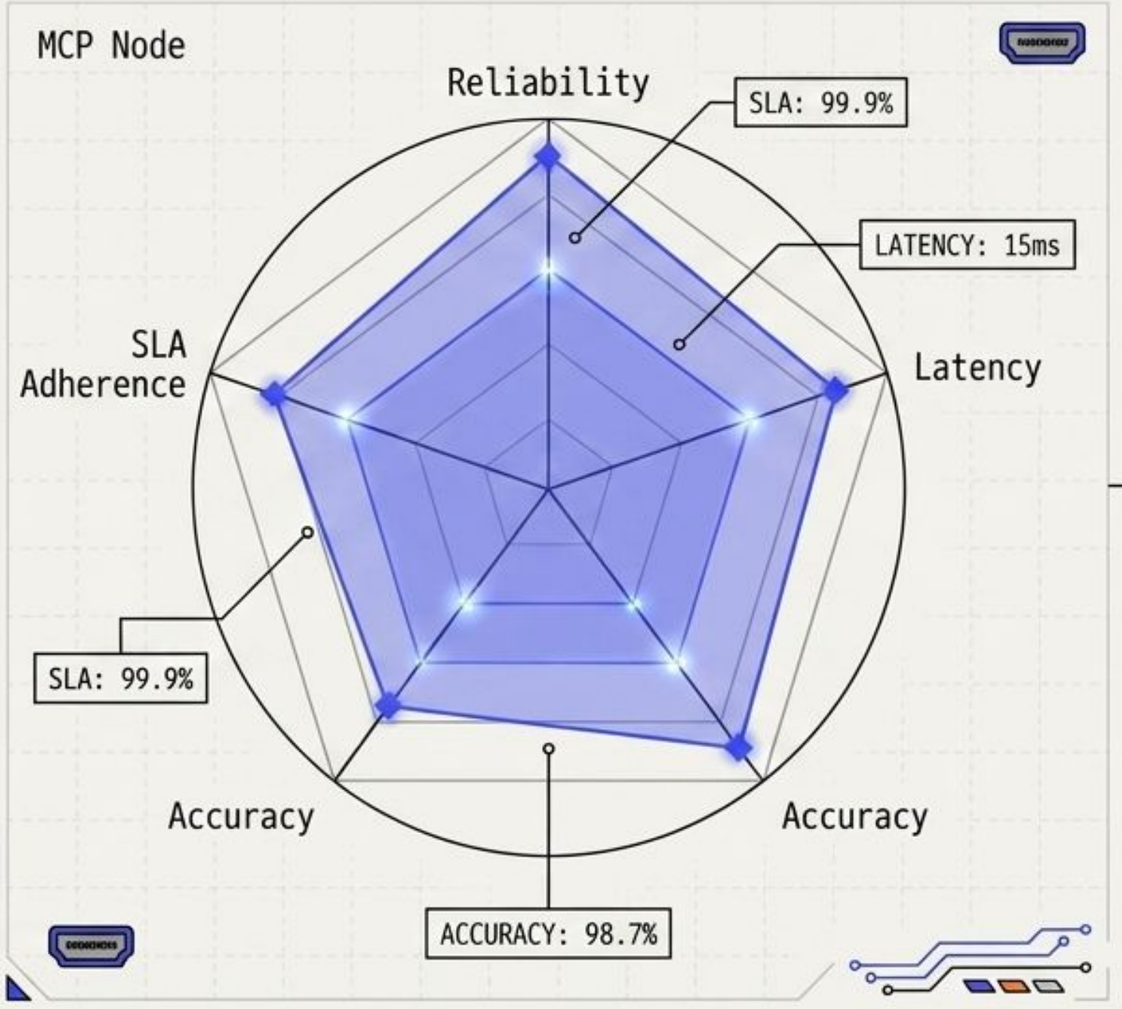
The External Perimeter

Third-party MCPs undergo significantly more rigorous scanning and gating compared to trusted in-house systems.

The Agent Surface Matrix

	Uber Agent Builder (No-Code)	**Uber Agent SDK** (Code-First)	**Coding Agents & Minions** (IDE/Background)
Target Persona	Productivity & Team Workflows (1,000s active/month)	Custom App Engineers	All Developers (95% usage)
Configuration Mechanism	`System Instructions (@MCP_Name)`	`yaml Configuration File`	`AIFX CLI Tool (mcp add)`
Tool Selection Strategy	UI-based explicit tool selection & static parameter overrides to prevent LLM hallucination	SDK automatic load with forced parameter overrides	Remote or local availability across Cloud Code and Cursor
Hero Use Cases	Internal workflow automation	Grocery Assistant, Auto Care Coordination, Customer Support	Background agents generating 1,800 code changes per week

Registry 2.0: Optimizing for Discovery



Surfacing Quality via SLAs

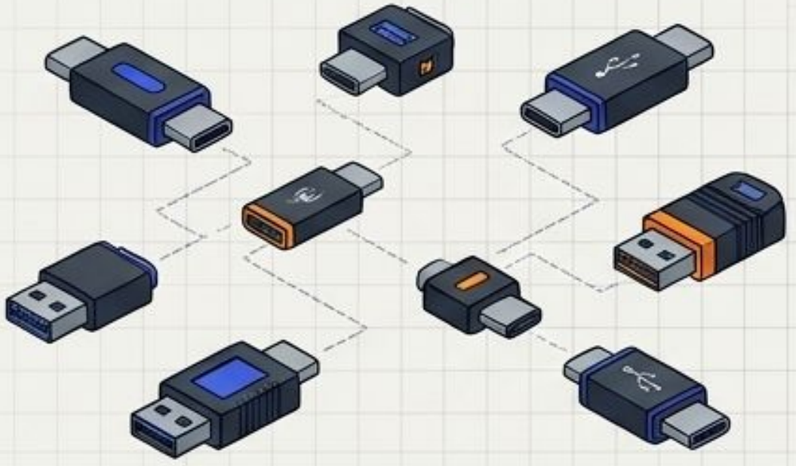
The registry is expanding to include strict evaluation metrics. MCP servers will be tiered based on historical service SLAs, reliability, and availability, ensuring agents only connect to high-performing tools.

On-Demand Tool Search

Implementing dynamic tool discovery to drastically reduce context bloat. Agents will automatically search for and load only the precise tools needed for a specific prompt, rather than carrying the weight of the entire registry.

The MCP Omniverse: From Tools to Sharable Skills

The ultimate goal of the MCP infrastructure isn't just basic connectivity—it is composable intelligence.



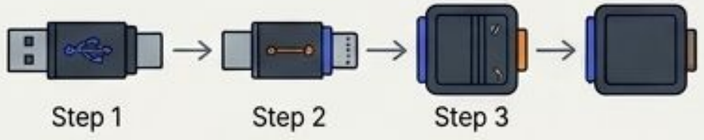
Isolated Tools

The Evolution of Skills

Composable Skills Cluster

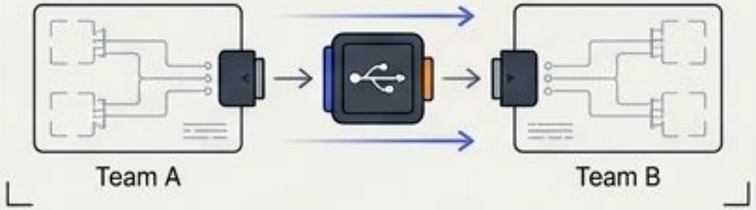
Recipes of Execution

Moving beyond single endpoints to complex "skills" (recipes for utiilzing multiple MCPs in sequence).



Cross-Boundary Sharing

Standardizing conventions so processes can be shared seamlessly across entirely different engineering teams.



Rigorous A/B Testing

Measuring the correctness of skill invocations and A/B testing different versions of identical skills to find the optimal AI workflow.