



AI Platforms Group



AI Agents, and the Model Context Protocol

AI Platforms Group Briefing

Tom Martin, David Heurtaux, Kiran Ikram,
Dan Sack, Djon Kleine, Niels Degrande

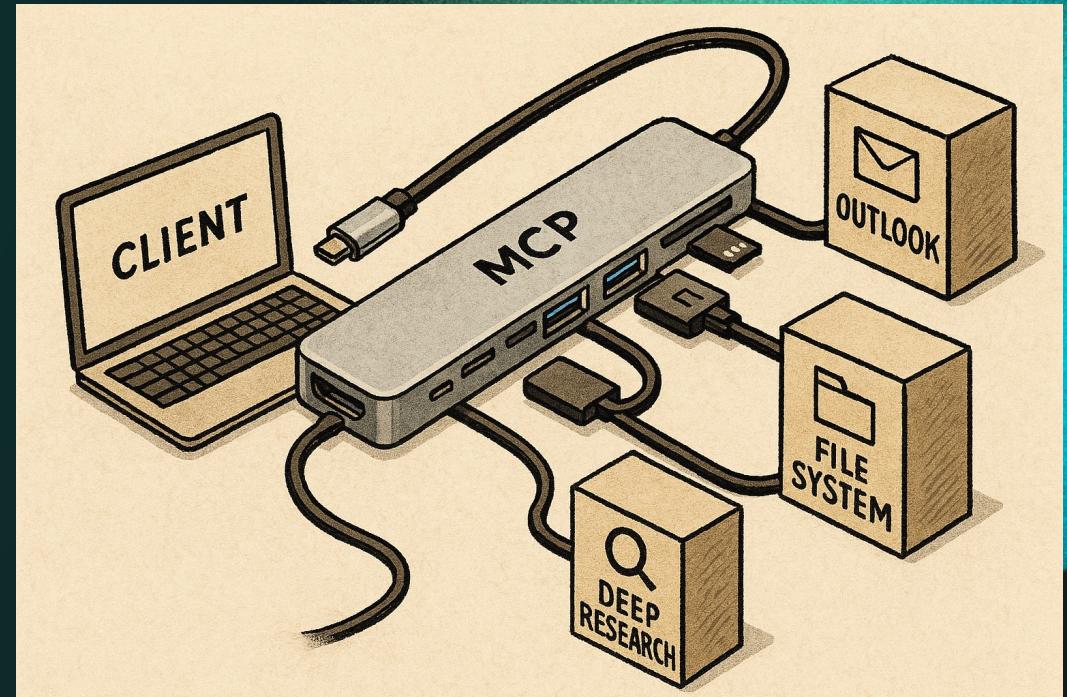
APRIL 2025

-
-
-

The open-source **Model Context Protocol (MCP)** launched by Anthropic is growing in popularity. In four short months it has been adopted by OpenAI, Microsoft, Google, Amazon, and others marking a shift in how AI Agents observe, plan, and act with their environments.

But how does it make AI Agents more reliable, safe and enterprise-ready?

This short brief unpacks how AI Agents are evolving, MCPs role, and why it's a meaningful step towards agentic abundance



Source: Prompted in ChatGPT 4o

01

**How are Agents
evolving?**

02

**Where do they have
product-market-fit?**

03

**Can they be reliable
and effective?**

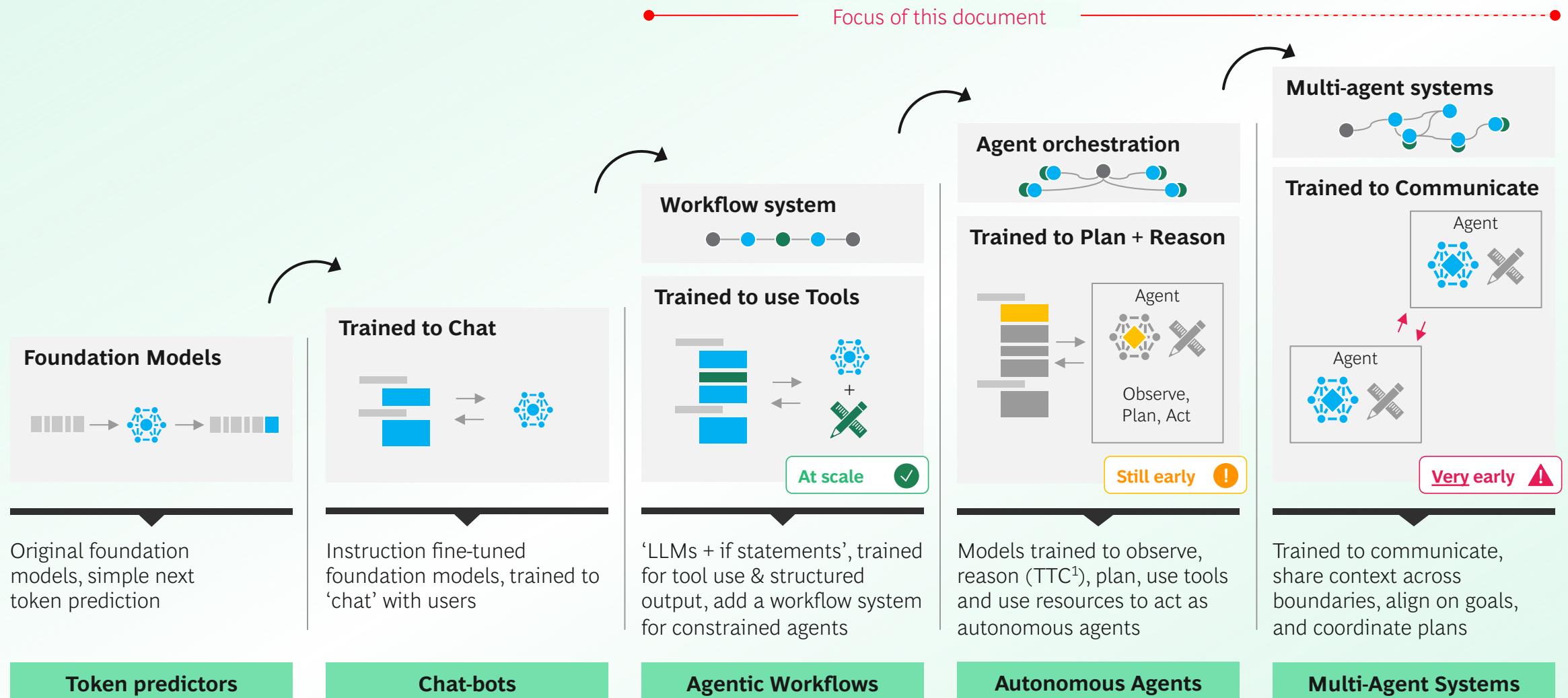
04

**MCP's role and
building at scale**

01

How are Agents evolving?

We are moving beyond ‘agentic if-statements’, towards autonomous agents

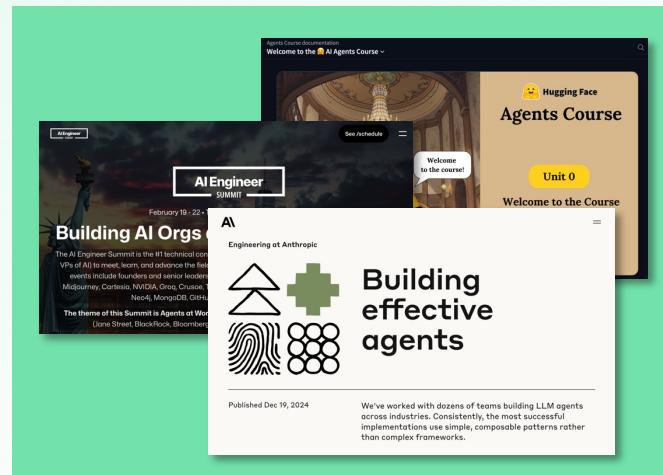


1. Test-Time-Compute, Source: BCG

Techniques, frameworks and proof-points are maturing rapidly ...

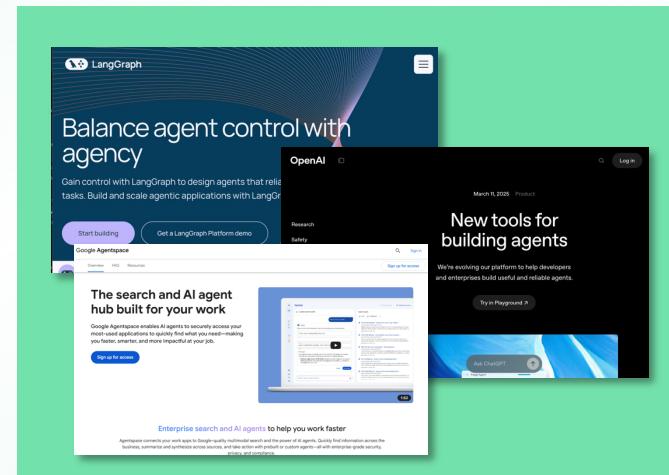
As techniques are shared,
community grows

Industry-wide knowledge and training on how to build agents is growing, and being shared, leading to fast feedback cycles. With Anthropic, Pydantic, Langchain, Hugging face and others publishing detailed guides and training, and AI Engineering growing as a community



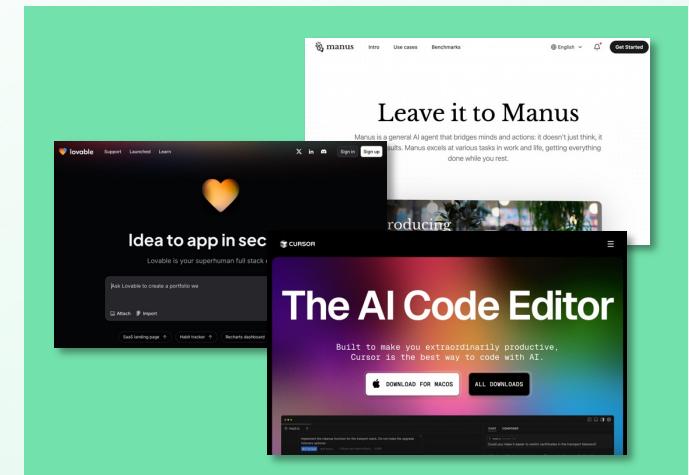
Agentic Frameworks lift
the tide for all

It is getting faster and easier to build, deploy, and monitor agents. Established players are evolving (e.g. OAI Assistants, Copilot, Agentspace, Bedrock agents), new players entering the game (e.g. Cloudflare, Pydantic), and low-code platforms growing (e.g. Lindy, Dust.tt)



Proof-points are visible,
compelling, and growing

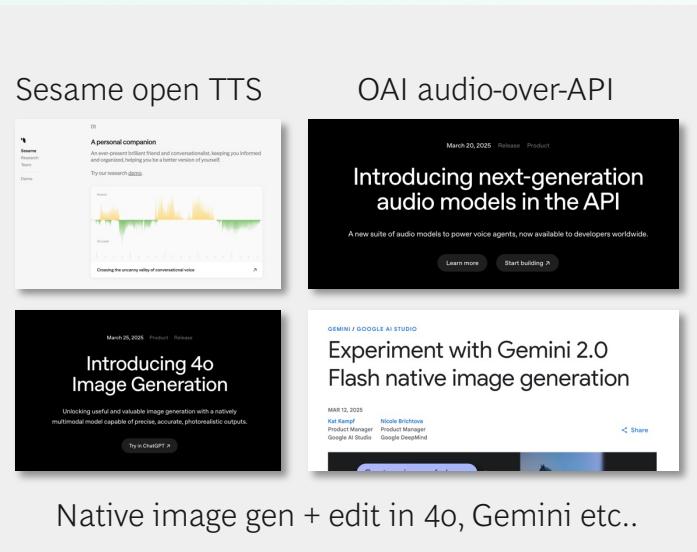
The first commercial agents are here, and generating meaningful revenue. From vibe-coders (v0, Cursor, Loveable, Bolt, Replit, Claude code), to consumer agents (e.g. Operator, Manus, and the many deep research's), to vertical B2B players (e.g. Intercom)



... and the underlying models are getting better and better

More modalities unlock
more opportunity

Voice goes mainstream, native image gen and edit has gone viral, and video continues to evolve



Better reasoning from Test-Time-Compute

+81%

Performance on ARC-AGI 1 in 6 months

5 of 10

5 of top 10 models
are ‘reasoning’,
and dominating
leaderboards

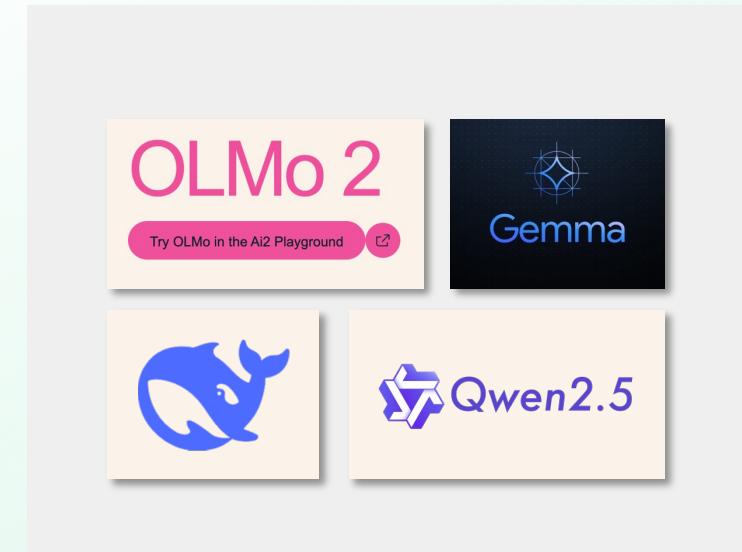
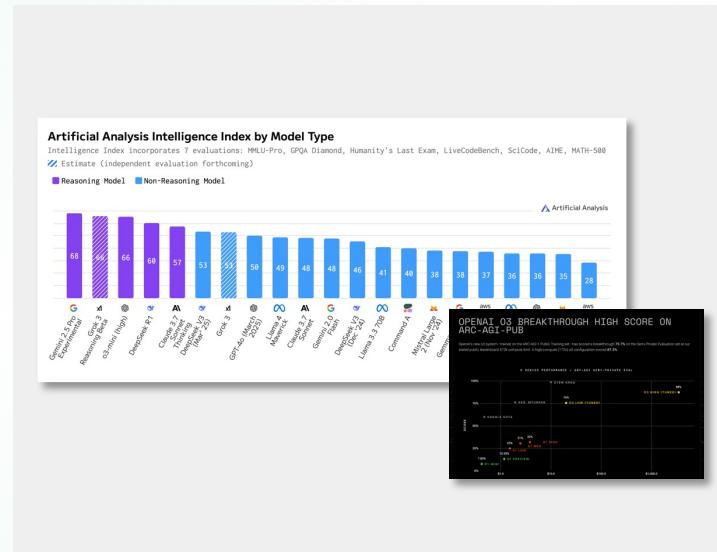
Open innovation is driving
cost down, and perf. up

~8-10

Open source
models matching
2024 GPT-4
performance

10x

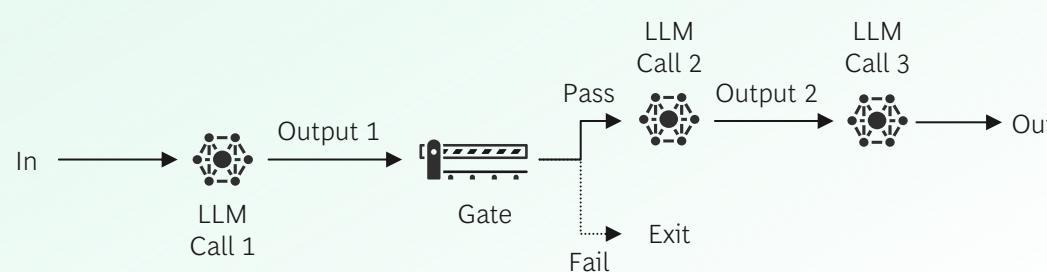
Reduction in cost per token for frontier performance



This maturity is driving a shift from predefined workflows to self-directed agents

Agentic Workflows

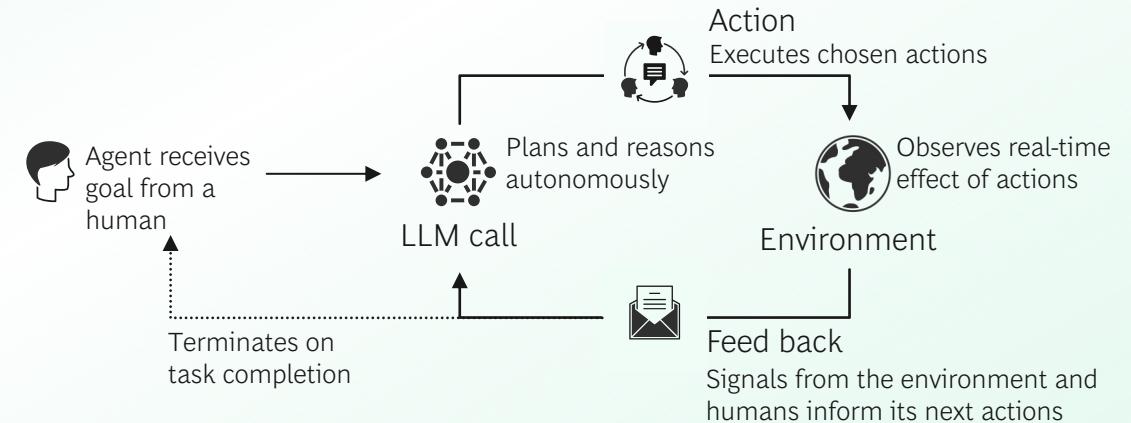
A common type of workflow, **prompt chaining** involves decomposing a task into a sequence of steps, where each LLM call processes the output of the previous one



- Are helpful when process consistency matters
- Good for problems where domain intelligence is valuable
- Are more predictable, but less adaptable to new inputs

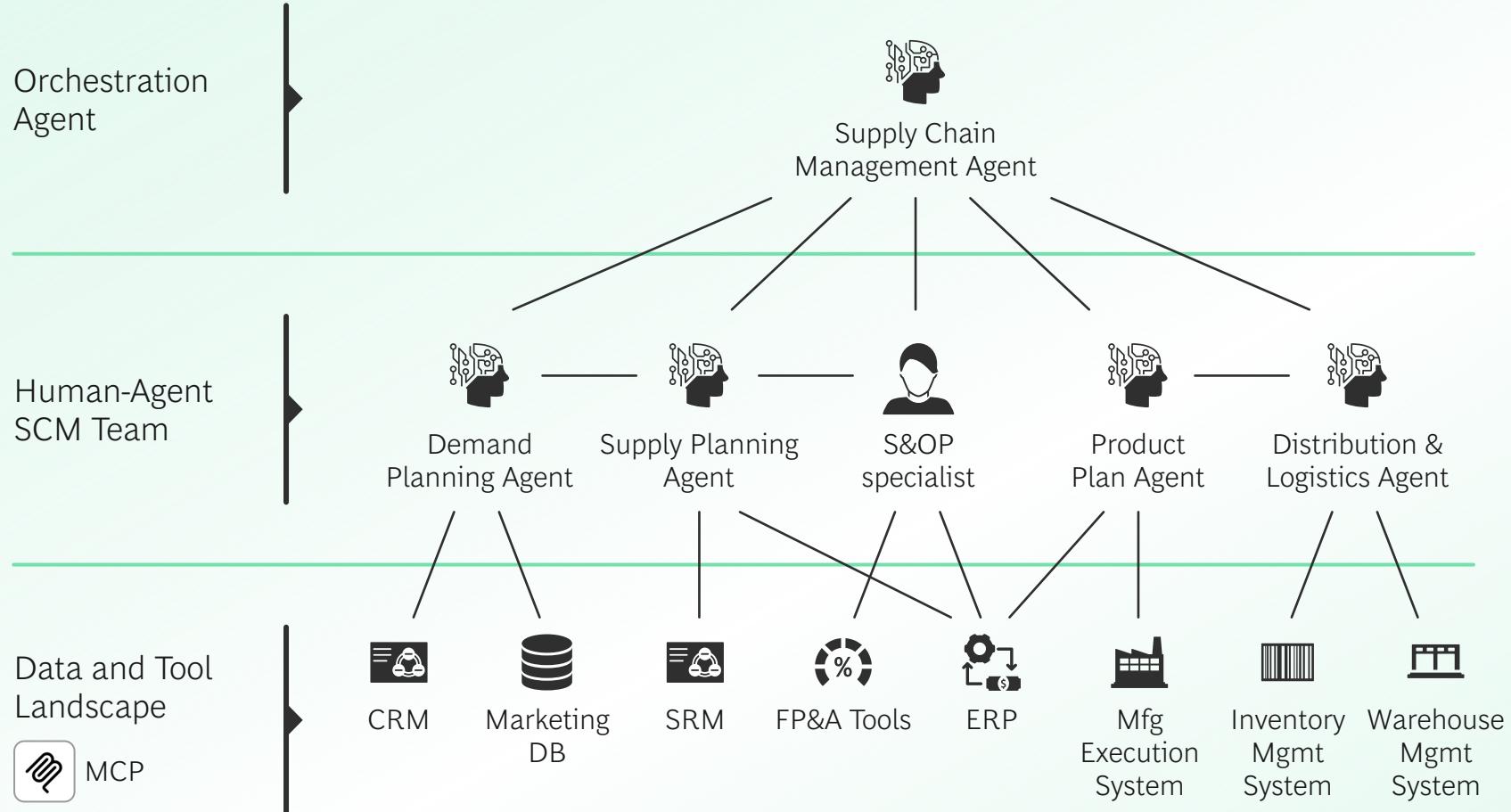
Autonomous Agents

A common type of agent, involves **autonomous reasoning**, acting, and learning from feedback, where each LLM call is informed by real-time signals from its environment and previous actions



- Are helpful when process flexibility matters
- Good for problems where general intelligence is valuable
- Are less predictable, but can adapt to user needs

Are we headed for a multi-agent future?



Agents can work together in networks and with humans to accomplish complex tasks or automate multi-step processes

EXAMPLE

Advanced E2E Supply Chain Management

A human-agent team that coordinates input from multiple agents to manage the supply chain process end-to-end. MCP helps expose data and tools

Avoid 'microservices' thinking, focus on collaboration and networks

02

Where do they have
product-market-fit?

Coding agents among the first to reach product-market-fit

Vibe-coding players are moving fast and capturing the software development market, accelerating software time to market

Company	ARR	User Growth
 CURSOR	\$100M by the end of 2024	~40,000 active paying customers by end 2024, surged to 360,000 users currently
 replit	\$28M (2025 estimate)	30M users by Sep'24, approaching 40M users currently
 bolt	\$20M in 2 months	>100,000 weekly users in first 4 weeks (Nov'24) Currently at 3M registered users
 lovable	\$17M in 3 months	3000 initial paying customers 500,000 users since launch in Nov'24
  	Extends to other vendors, data not available	

To move from prototyping to production, vibe coding must mature



Organizations are already gaining significant value from agentic workflows



Bloomberg's compliance agents rigorously check facts, catch edge-case risks and minimize exposure to costly mistakes. Agents execute structured workflows, **reducing time-to-decision by 30-50%**



Booking.com' and **Jane Street's** coding agents reclaim developer time – culling dead code, cleaning up cruft, assist with code reviews and cut cycle times by **30%+**



Brightwave's research agents turn 10,000 pages of legal and financial text into crisp decision-ready takeaways- on demand, and at scale. Agents are able to cross reference diverse data sources in real-time, continuously refining their own output



BCG delivered 300+ GenAI agents across 100+ clients, unlocking up to 90% cost reduction, 50–75% faster execution, and **30–40% productivity uplift** across critical business functions within various industries¹

1. Metrics are illustrative, depending on specific implementation context, and not guaranteed
Source: AI Engineering Summit 2025, Bloomberg, Booking.Com, Jane Street, Brightwave, LinkedIn; BCG

Future outlook

Only as **reasoning** and **evaluation** systems mature, will fully autonomous agents be able to handle complex, open-ended tasks

Assistive agents will thrive in high-risk domains, blending agent support with human judgement

Rule-based agents will remain as **reliable guardrails**, anchoring dynamic systems with predictability

Adaptive agents will lead enterprise adoption, balancing automation with real-time feedback and control

03

Can they be reliable
and effective?

Benchmarks and studies are exploring agent reliability



Benchmarks are shifting to **measuring how well agents use tools** and handle end-to-end tasks over time within different domains



New test cases **emphasize edge scenarios** like missing tools, irrelevant queries, and incomplete inputs to probe agent robustness



Increasing **focus on multi-turn tasks** requires agents to manage context, sequence actions, and adapt to evolving goals (e.g. ML training)

Governments around the world are worried about China's EV expansion, citing everything from potential job losses to data-security concerns. In the European Union, where Chinese imports make up about one-fifth of electric sales, regulators recently disclosed plans for tariffs up to nearly 50%. The Biden administration: A BLOCK QUOTE FROM AN ARTICLE *Tim Farley*, a 62-year-old, blunt-talking car fanatic who spent the early part of his career in marketing at Toyota Motor, sees Chinese EVs as an immediate threat in Europe and other overseas markets, and a long-term risk in Ford's profit engine of North America, regardless of protectionist measures.

--

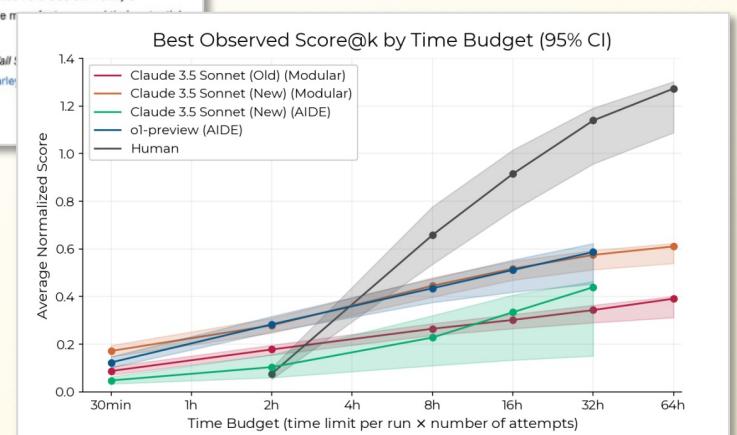
Identify the article that contains this quote. Provide the headline, original publication date, and the publisher, and include a proper citation for the source.

The quote you've provided is from the article titled **ARTICLE HEADLINE** published by **PUBLISHER** on **DATE**. The article discusses Ford CEO Jim Farley's concerns about the rapid advancements of Chinese electric vehicle impact on global markets.

Citation: Higgins, Tim. "What Scared Ford's CEO in China." *The Wall Street Journal*, 2023. <https://www.wsj.com/business/autos/ford-china-ev-competition-farley>

URL

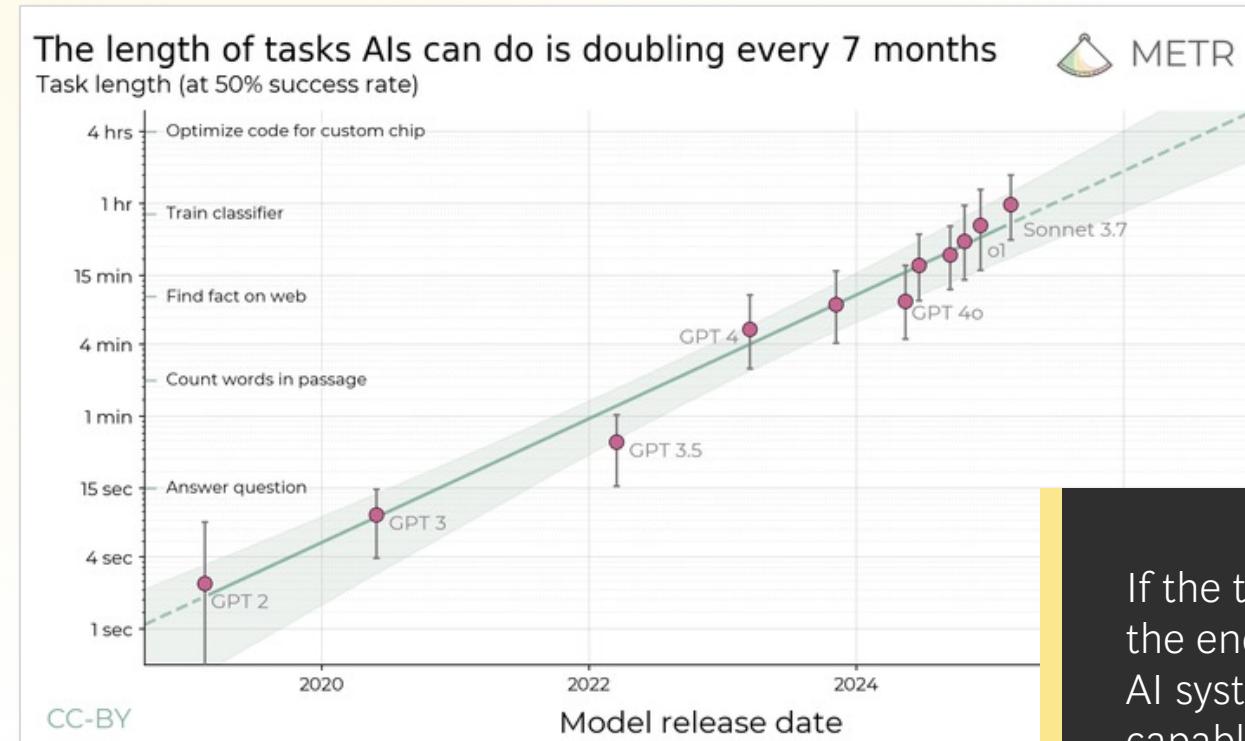
RE-Bench Evaluating frontier AI R&D capabilities of language model agents against human experts



Today, AI Agents can reach ‘1h’ of automation - doubling every 7 months

Current SOTA¹ models are capable of some tasks² that take even expert humans hours, but can **only reliably complete tasks of up to a few minutes long**

Length of addressable tasks with 50% reliability has **been doubling approximately every 7 months** for the last 6 years



If the trend continues to the end of this decade, AI systems will be capable of autonomously carrying out **month-long projects**

Source: Measuring AI Ability to Complete Long Tasks arXiv:2503.14499 [cs.AI]; Illustrative diagram

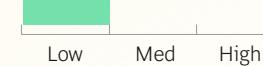
1. State Of The Art 2. Time taken by human experts is strongly predictive of model success on a given task: current models have almost 100% success rate on tasks taking humans less than 4 minutes, but succeed <10% of the time on tasks taking more than around 4 hours

BCG's Agent Assessment Framework

The 6 dimensions we use to track Agent performance in 2025

1	2	3	4	5	6
Reasoning & Planning	Task autonomy & Execution	Memory & Knowledge	Reliability & Safety	Integration & Interoperability	Social understanding
Ability to follow instructions, understand intent, infer, and make decisions based on data to form a plan	Function calling performance to execute tasks, interact with environments, and take goal-based actions	Ability to use and leverage knowledge, overall long context performance	The consistency, accuracy, and trustworthiness of an AI Agent's responses	Seamlessly exchange data, communicate, and collaborate with other systems or agents	Ability to interpret human intent, social cues, maintain character, and share context in natural language

Constrained agents working today, but full autonomy on the horizon

	1 Reasoning & Planning	2 Task autonomy & Execution	3 Memory & Knowledge	4 Reliability & Safety	5 Integration & Interoperability	6 Social understanding
Current Capability Maturity	 Low Med High	 Low Med High	 Low Med High	 Low Med High	 Low Med High	 Low Med High
Limitations	<ul style="list-style-type: none"> Struggles with multi-step reasoning and long-term dependencies Prone to hallucinations and incorrect inference 	<ul style="list-style-type: none"> Struggles with real-world execution beyond simulations Limited integration w/ external tools & APIs Limited standardization 	<ul style="list-style-type: none"> Limited memory retention across long conversations Forgetfulness due to context window size restrictions 	<ul style="list-style-type: none"> Tendency to hallucinate or generate incorrect information Vulnerable to biases 	<ul style="list-style-type: none"> Data silos and inconsistent formats hinder interaction Security risks from broad access permissions 	<ul style="list-style-type: none"> Struggles with detecting emotional nuance and non-verbal intent Prone to misinterpreting ambiguous or implied language
What needs to happen for these capabilities to evolve/mature	<ul style="list-style-type: none"> Advancements in multi-step reasoning, supported by a versatile action set Additionally, enabling reasoning during inference 					
 Perceived maturity by AI experts  Divergent expert perspective						

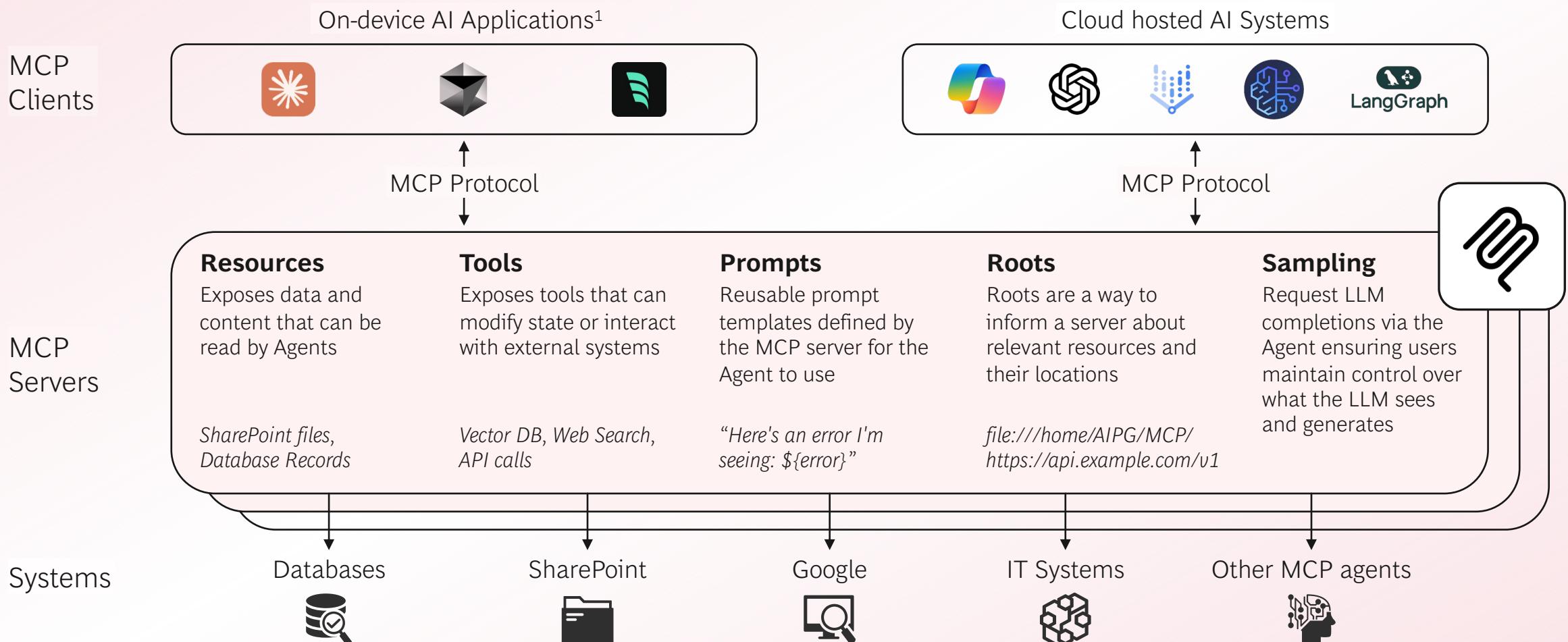
Source: BCG analysis; Expert interviews

1. To tools & data; 2. Ability to use long-context windows, recall accuracy from training data, knowledge consciousness and ability to know when to call external tools, intra-session memory (learning)

04

MCP's role and building at scale

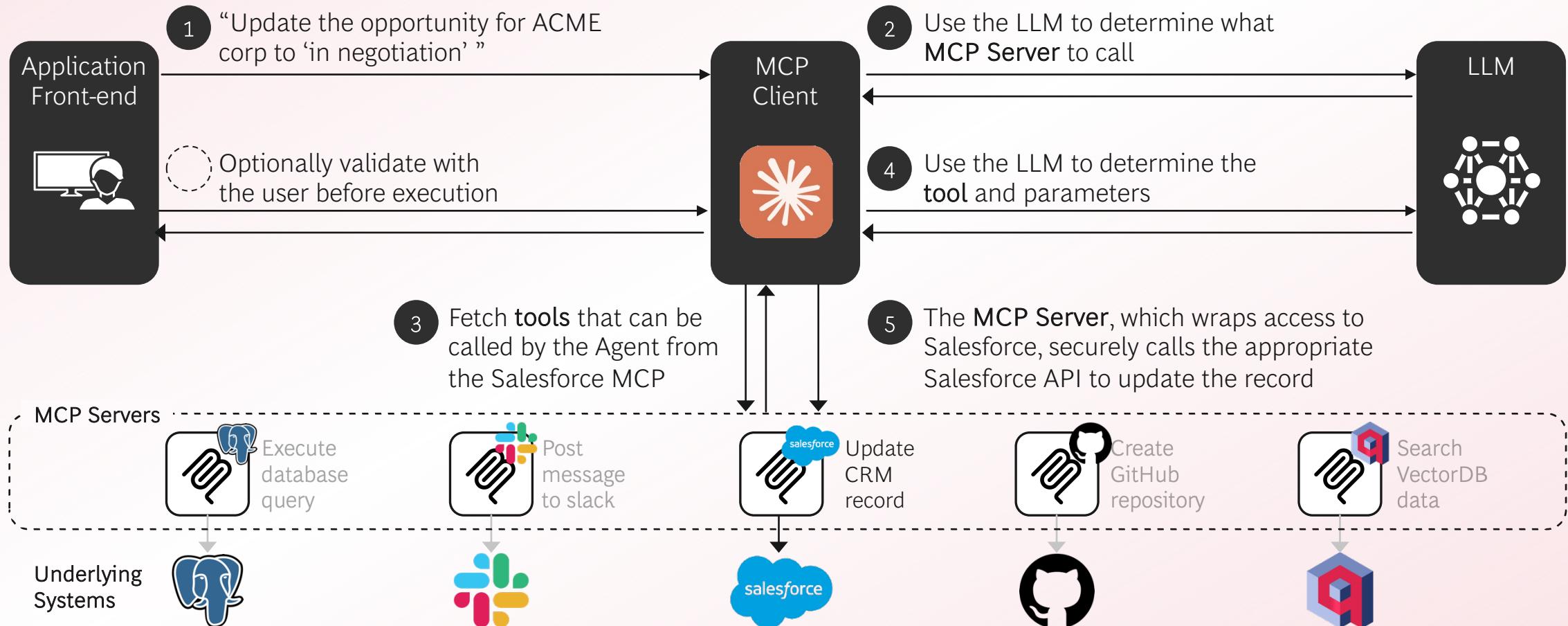
The Model Context Protocol (MCP) exposes resources, tools and prompts to LLMs



1. In many current implementations (e.g., VS Code, Claude Desktop, Cursor), the front-end and MCP client are effectively merged. For custom enterprise deployments—such as those developed by BCG—this separation remains architecturally relevant and often intentional.

Source: Anthropic Model Context Protocol; BCG Experience

MCP unlocks agentic workflows through one unified protocol

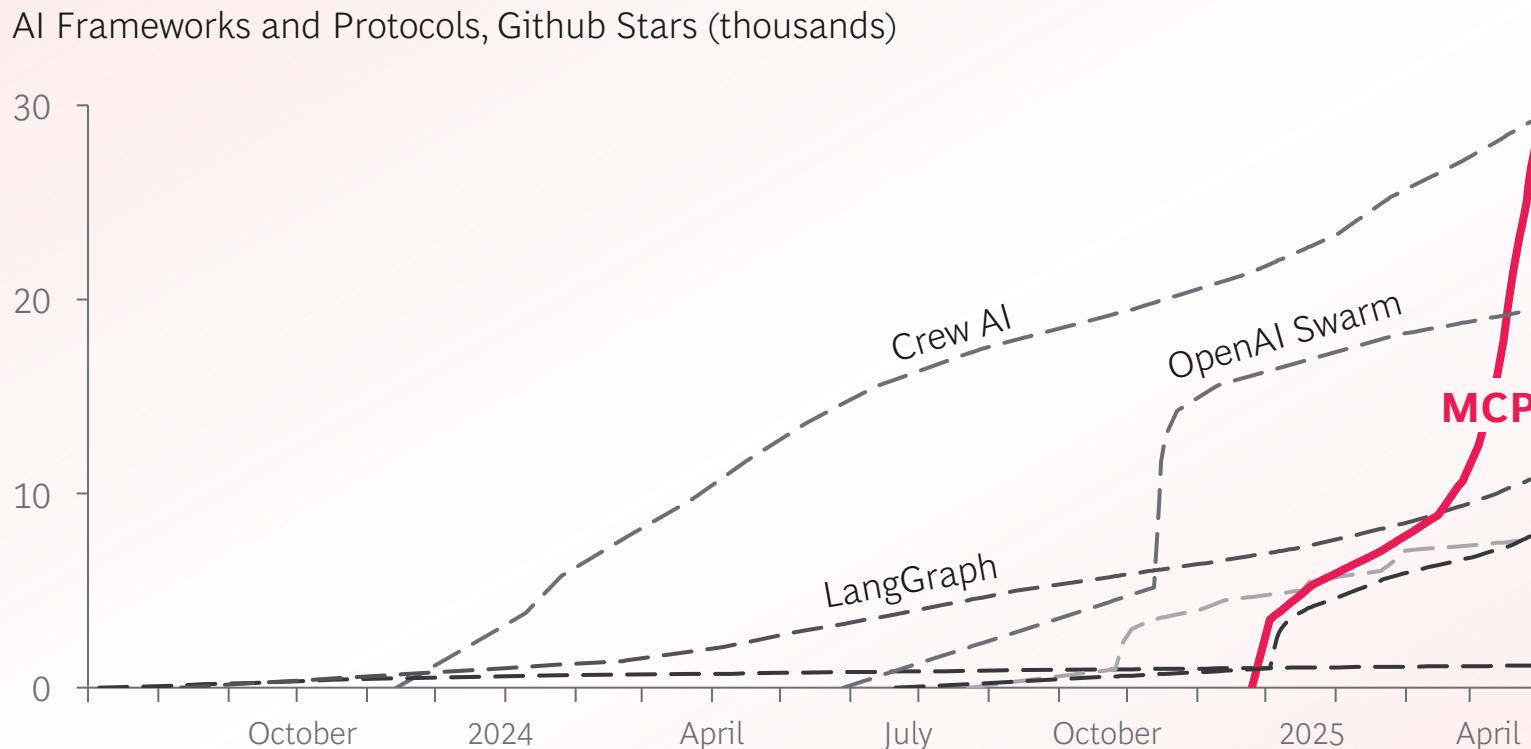


MCP addresses 4 of the 6 capabilities seen as lacking in today's agents

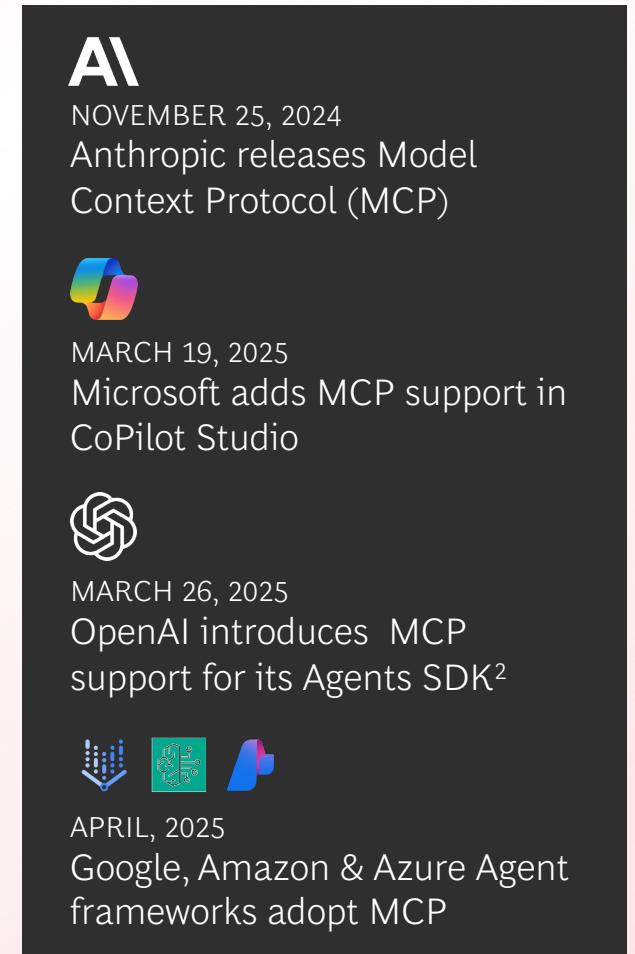
1	2	3	4	5	6
Reasoning & Planning	Task autonomy & Execution	Memory & Knowledge	Reliability & Safety	Integration & Interoperability	Social understanding
MCP's Role					
MCP Servers expose prompt templates and tool registries which allow for high quality context to aid in tool use reasoning	MCP Clients and Servers allow agents to chain tool usage autonomously and coordinate execution across multi-tool workflows	MCP Servers allow clients to connect to external tools like Postgres for real-time data access or Vector Databases for knowledge access	The consistency, accuracy, and trustworthiness of an AI Agent's responses is driven by the model and evals	The MCP protocol resolves architectural fragmentation by bridging tools and platforms through standardized interfaces	Social understanding; link to empathy, intent detection, are inherent to the model

MCP has gained popularity in the AI community

MCP is becoming the de facto standard by being an AI-native, open protocol backed by Anthropic; leveraging the successful LSP¹ foundations, launching with a full first-party stack, and executing a rapid iterative delivery roadmap

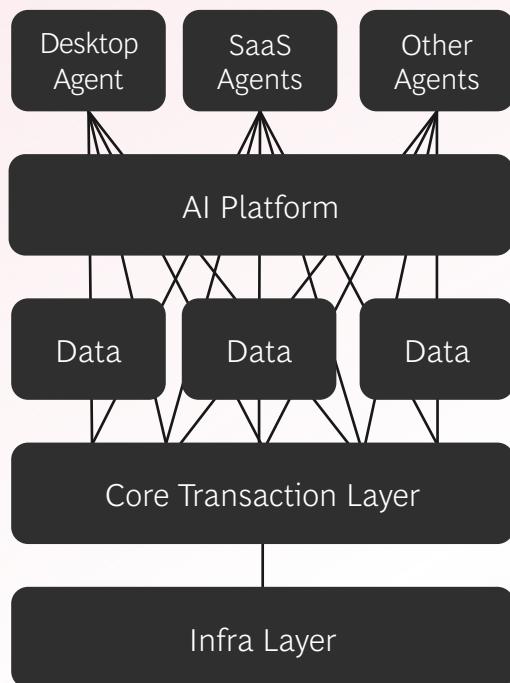


Source: <https://star-history.com/>. 1. Language Server Protocol enabling development tools to communicate with language-specific servers for features like autocomplete, go-to-definition, and hover documentation 2. Software Development Kit

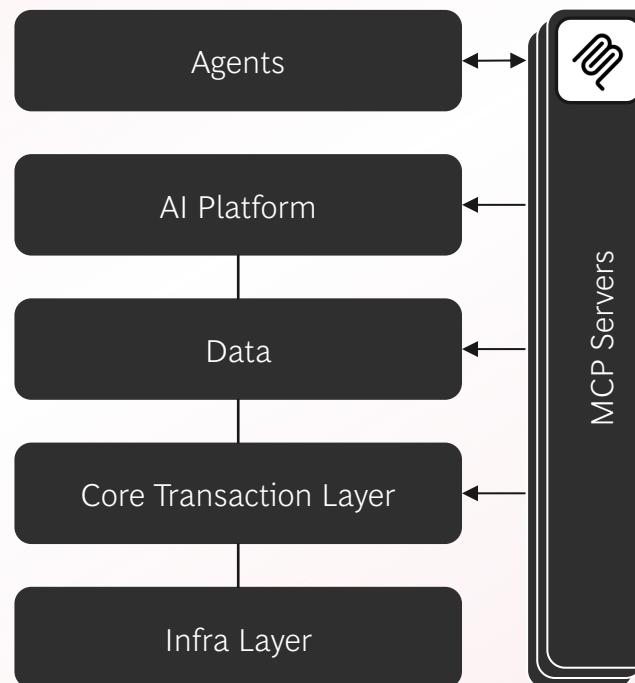


How to think about MCP in the context of your architecture?

Siloed agents, data and systems, with duplicative or diverging integrations



Transversal shared set of MCP Servers liberating access to data and systems

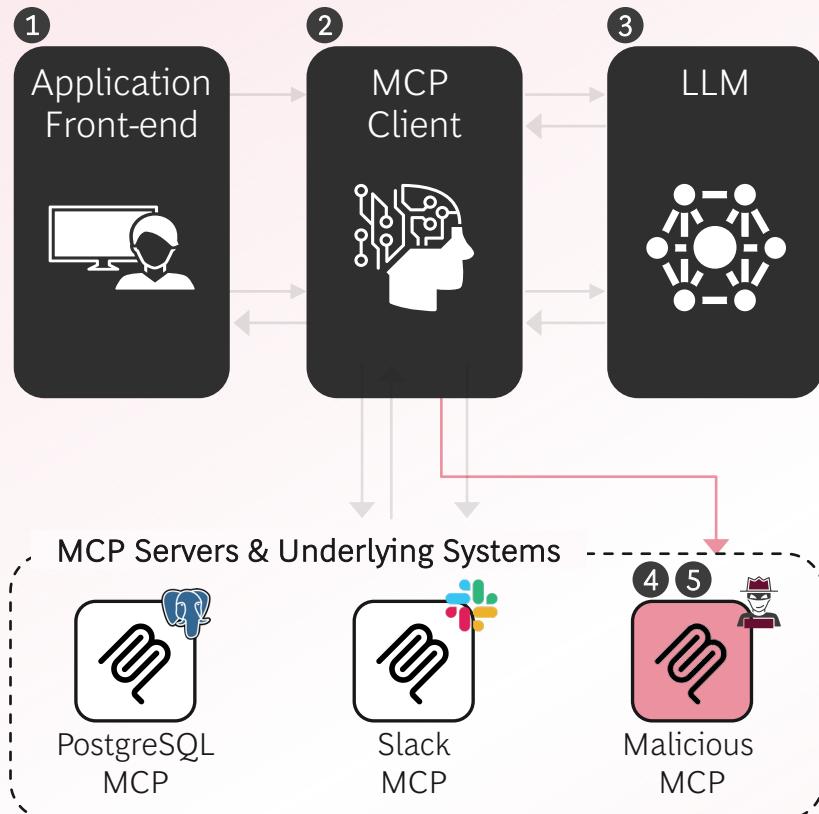


MCP **de-duplicates integration efforts**, and enables faster experiments and seamless system upgrades behind the scenes

MCP doesn't solve all our problems; **currently a lot of work still goes into integrating tools** as adoption maturity is low

MCP value hinges on the Agents that consume them and **broader ecosystem adoption**

Access to tools creates new risks — security must be foundational, not optional



- 1 Malicious tools can read and **access local credentials** (e.g., SSH¹ keys, config files) and leak them via innocuous inputs
- 2 Agents can be vulnerable to **invisible tool poisoning attacks** where tool descriptions include embedded malicious instructions
- 3 Users may see safe summaries; but models act on full descriptions. This mismatch enables **prompt injection** without user visibility
- 4 Tool logic can be altered server-side after trust is granted, **compromising agent behavior** over time
- 5 One compromised MCP server can influence how agents use tools from other trusted servers—**breaking domain isolation**

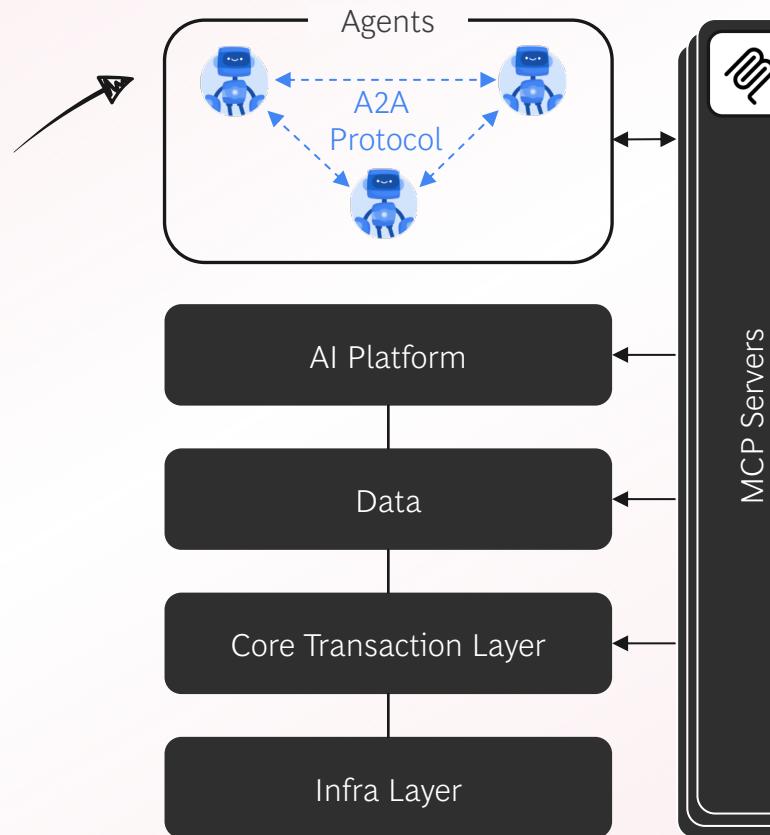
Treat all tool logic and servers as untrusted. Enforce OAuth + RBAC² on every call, and pin tool versions. Isolate trust domains to prevent cross-server hijacks. Log agent reasoning traces, not just outputs. And don't rely on random GitHub code—verify or build your own.

Emerging agent-to-agent protocols, such as Google's A2A, will work alongside MCP

A2A defines **how agents talk, coordinate, negotiate, and share state—not how they're built**

It supports natural communication, plan refinement, task handoffs, and cross-boundary collaboration

Leading agent frameworks including Google's Agent Developer Kit (ADK), CrewAI, LangGraph, and GenKit already have examples integrating A2A into agent building frameworks to enable **natural agent-to-agent collaboration with each other**



A2A and MCP solve different layers of the AI tech stack: A2A handles the dialogue between agents, while MCP enables agents to discover and call each other as resources via AgentCards¹, and give them access to tools

Proceed with **curiosity and caution**. Protocols like A2A (launched days ago) and IBM's ACP² are promising but expect fragmentation, evolving specs, and competing standards

Source: Google. 1. Agent Card: A public metadata file (usually at `./well-known/agent.json`) describing an agent's capabilities, skills, endpoint URL, and authentication requirements. Clients use this for discovery. 2. Agent Communication Protocol

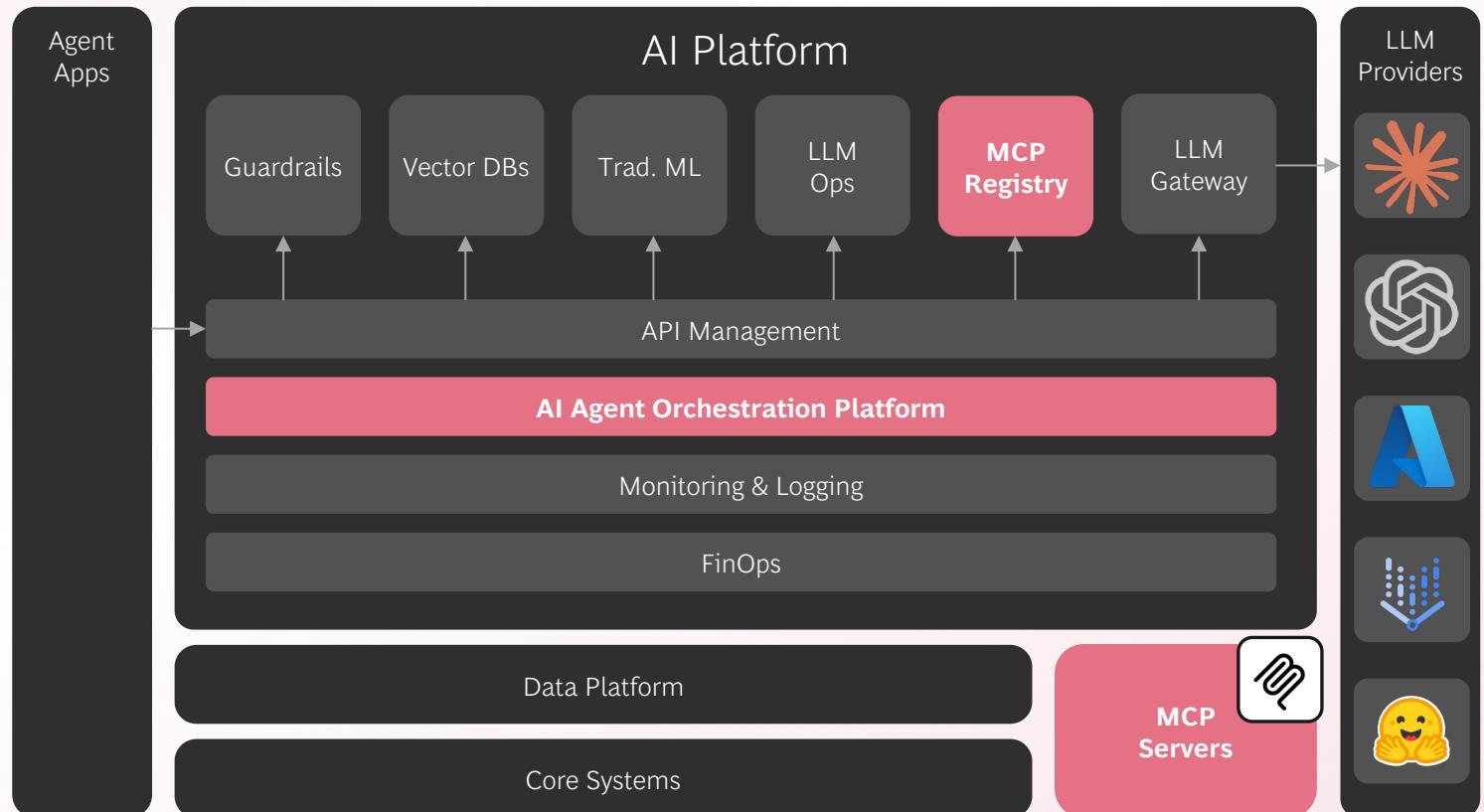
Agent orchestration & MCP will be the beating heart of the modern AI company

Agent Orchestration

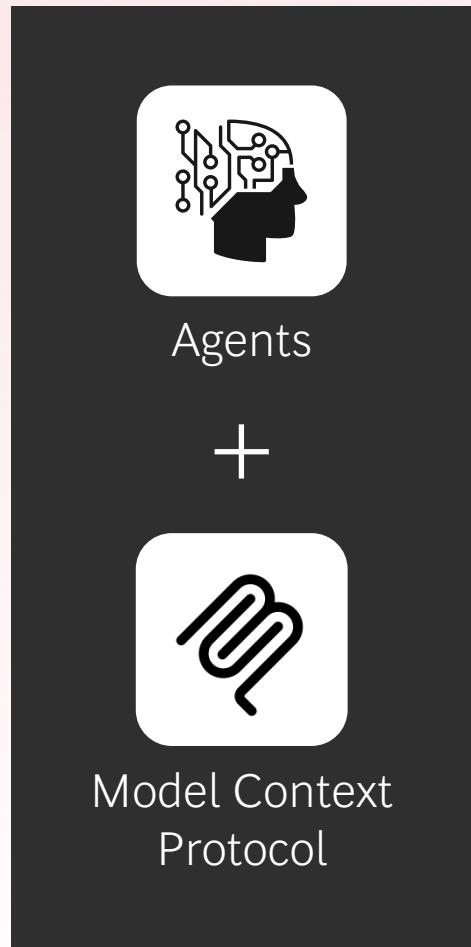
Platform-as-a-service offerings (e.g. Azure AI Foundry, Google Vertex, Amazon Bedrock Agents, Lindy) that enable the creation, orchestration, and deployment of agents through their lifecycle (AgentOps)

MCP Registry

Directory service that catalogs, governs, and exposes available MCP servers within an organization



Key points for building Agents with MCP in your Enterprise



1 Eval driven development enable agents at scale

Agents without evals are stochastic parrots, not co-workers. Design and build with evals from the start

2 Plan out the ‘MCPs’ that will unlock your silos

Think through the systems and datasets you want your AIs to access, then implement MCPs as the ‘new data mesh’

3 Build an Agent Orchestration platform, and proprietary MCP registry

Choose an agent platform to allow you to build and scale agents with evals, and couple with an in-house MCP registry to open up silos

4 Review Legal, Data Security, and Privacy implications

MCPs bring unique risks, as the ‘AI surface area’ grows beyond limited, and constrained chatbots. *BCG does not provide legal or compliance advice*

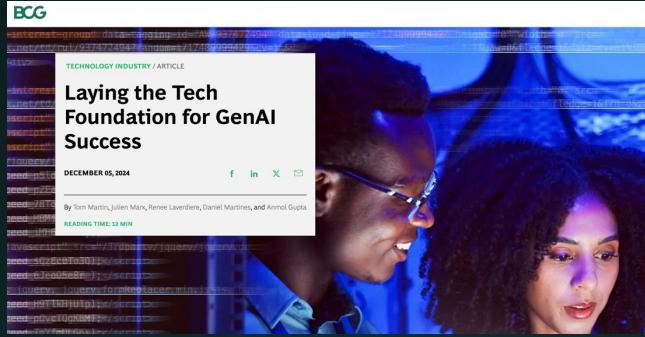
5 Technical Appendix for further reading at the end of the document



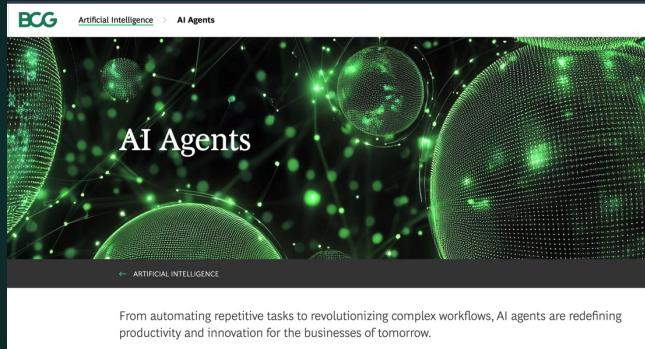
Read more of BCG's perspectives

-
-
-
-
-

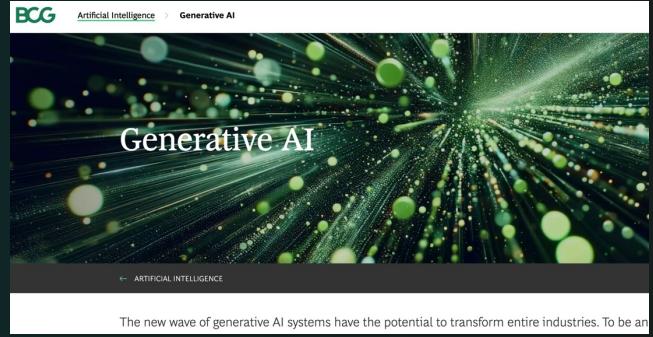
Building Multi-model platforms



Latest thinking on Agents



AI on BCG.com



Our Executive Perspective Series



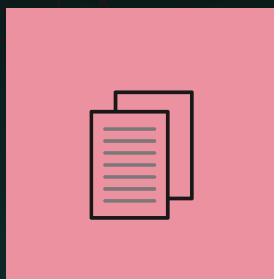
Get in touch with
our AI team

Co-Authored this paper



-
-
-

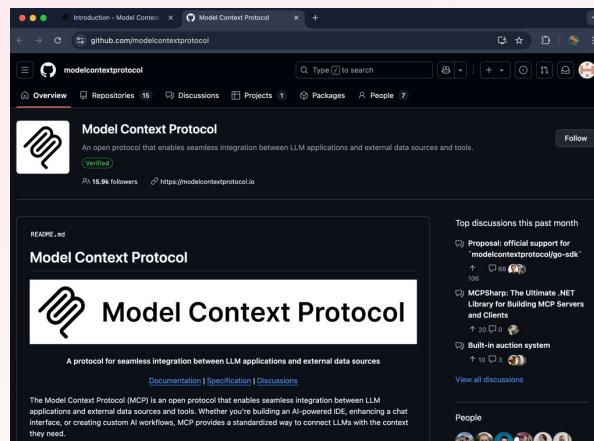
Technical appendix



Technical appendix

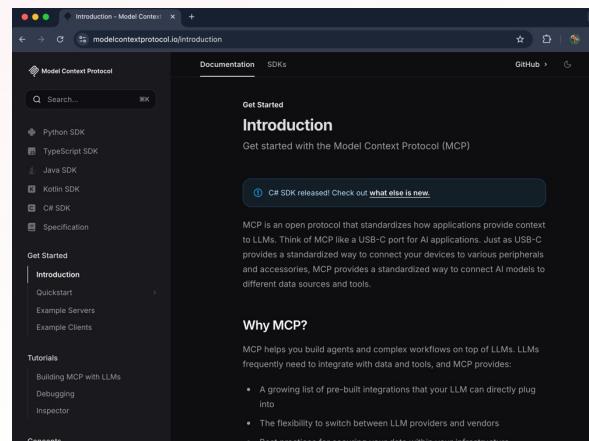
Anyone can build MCP clients or MCP servers, it's truly open source

Built in public with feedback loops



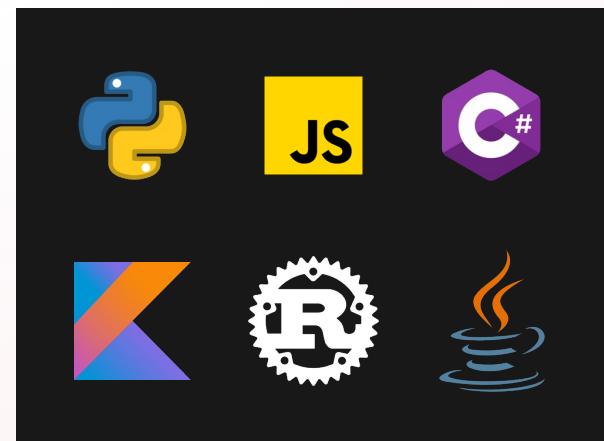
MCP is developed as an open-source project on GitHub, encouraging community contributions and transparency; with key players such as Pydantic contributing to the protocol

Comprehensive Documentation



Anthropic provides detailed documentation, providing clear guidelines and best practices for developers to understand and implement MCP effectively

Growing list of supported MCP SDKs



MCP offers official SDKs in multiple programming languages, including Python, TypeScript, and C#, facilitating seamless integration across diverse development environments

Many MCP servers have been built and shared across the eco-system...

Non-exhaustive

Enterprise software



Connects agents with Salesforce's CRM platform, enabling retrieval and interaction with Salesforce data & metadata

servicenow

Supports agents in retrieving and managing IT service management data (e.g. incidents, requests) through ServiceNow's APIs



Supports agents in deploying Cloudflare Workers, querying D1 databases, or copying data between KV storage and R2 buckets

Desktop automation

Filesystem

Enables agents by reading, writing, searching, and organizing files on the host machine (within permitted directories)

Puppeteer

Allows agents to programmatically navigate websites, extract information, or fill out forms; mimicking desktop web browsing tasks

Fetch

Enables agents to retrieve and transform live websites, fetches content, strips HTML, or converts to Markdown

Productivity tools

slack

Enables agents to interact with Slack workspaces – post messages, read threads, and manage channels via the Slack API

Google Drive

Allows agents to list, read, and search Drive files, converting Google Docs/Sheets into plain text for LLM consumption

todoist

Lets agents manage to-do lists by reading, adding, or updating task. Supports natural language task tracking, reminders, and project-based productivity flows



Community driven

Software Dev/Ops tools

GitHub

Gives agents access to repository content, issues, pull requests, and commits through GitHub APIs

SENTRY

Lets agents retrieve and analyze application errors, alerts, and logs from Sentry.io allows, debugging, and surfacing insights from observability pipelines

Grafana

Enables agents to query metrics, logs, traces, and dashboards via Grafana's APIs to manage incident analysis, on-call alerts, and live monitoring queries



Community driven

Diverse and growing network of MCP servers is emerging across enterprise

Official MCPs are still early-stage; documentation gaps and incomplete tool definitions are common

Widespread access creates surface area for cyber risk; robust static security and isolation are mandatory

Emerging best practices for designing, using and building with MCP and Agents

1 Use structured frameworks

Accelerate development with orchestration libraries such as the MCP SDKs, FastMCP, Langraph, or many others!

2 Design tools precisely

Define clear, scoped tool descriptions—agent reasoning quality depends on precision and coherence

3 Limit cognitive overload

Keep toolsets under 100 per call; expose only what the agent truly needs

4 Evaluate LLM outputs

Continuously test LLM reasoning paths—no evals means no guarantees

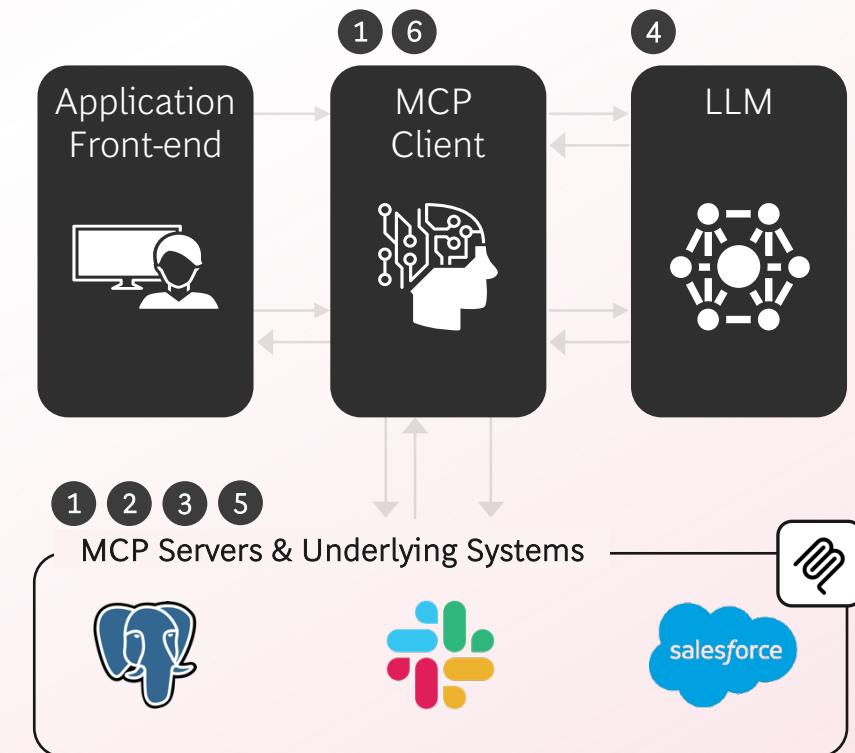
5 Modularize via server boundaries

One MCP server per system improves routing clarity & flexibility

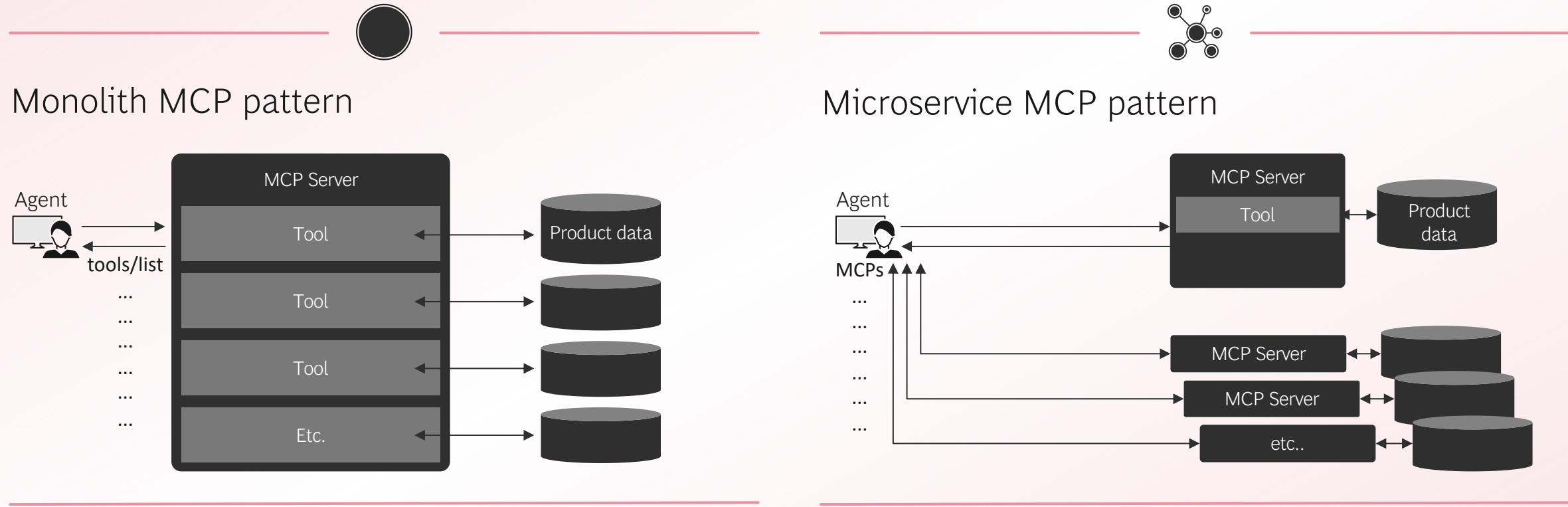
6 Secure agent interactions

Enforce auth for front-end MCP clients to protect user and system access via oAuth

Deep-Dive



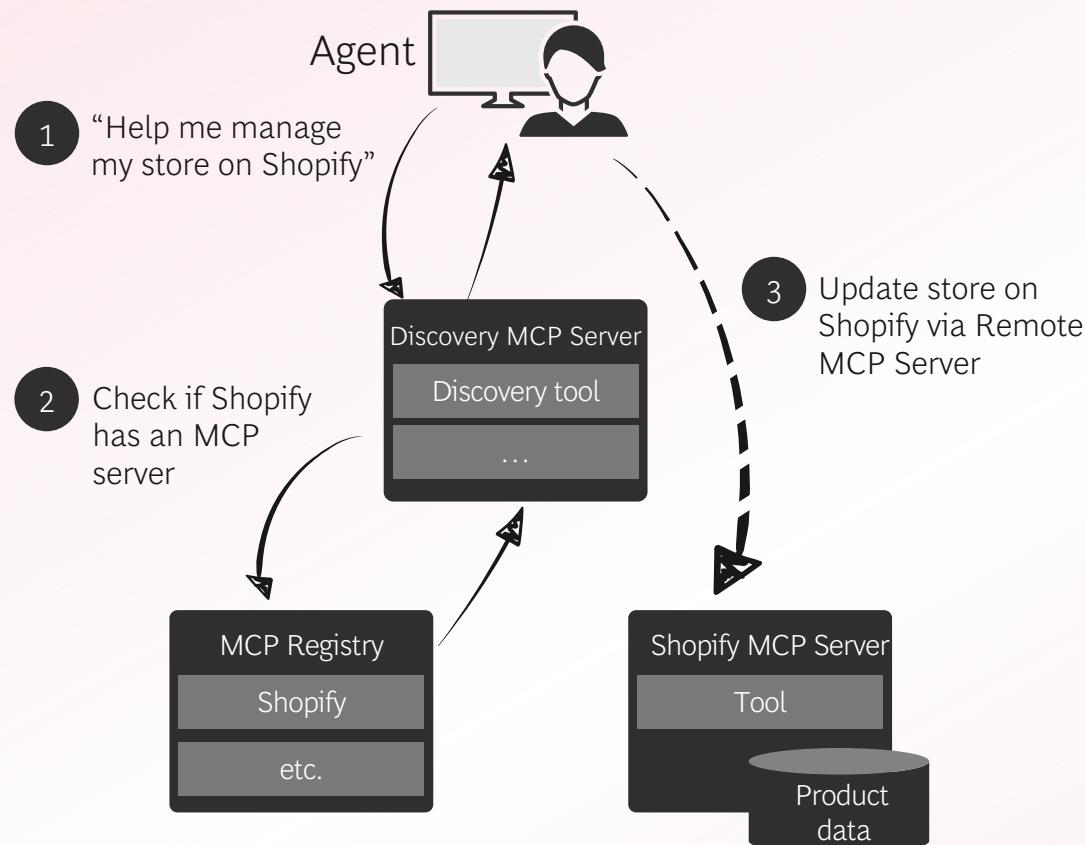
Avoid monoliths or microservices at the extremes when building MCP servers



Monolith MCP servers **bundle multiple tool integrations** or functions into one server across service boundaries, potentially **overloading the agent** with too many choices

Microservice MCP servers host a **subset of tools**, which can lead to a highly complex, unmanageable agent architecture, **undermining interoperability and reusability**

Focus on building lean MCP servers, leverage dynamic server discovery if needed



- 1** Agent receives a user request (i.e. “Help me manage my Shopify store”)
- 2** Checks a registry or the target domain’s .well-known/mcp.json for available MCP servers
- 3** Dynamically loads the server’s schema/actions without needing manual configuration

Avoids overwhelming the agent with all tools up front & adapts instantly to new or evolving server endpoints and APIs

Disclaimer

The services and materials provided by Boston Consulting Group (BCG) are subject to BCG's Standard Terms (a copy of which is available upon request) or such other agreement as may have been previously executed by BCG. BCG does not provide legal, accounting, or tax advice. The Client is responsible for obtaining independent advice concerning these matters. This advice may affect the guidance given by BCG. Further, BCG has made no undertaking to update these materials after the date hereof, notwithstanding that such information may become outdated or inaccurate.

The materials contained in this presentation are designed for the sole use by the board of directors or senior management of the Client and solely for the limited purposes described in the presentation. The materials shall not be copied or given to any person or entity other than the Client ("Third Party") without the prior written consent of BCG. These materials serve only as the focus for discussion; they are incomplete without the accompanying oral commentary and may not be relied on as a stand-alone document. Further, Third Parties may not, and it is unreasonable for any Third Party to, rely on these materials for any purpose whatsoever. To the fullest extent permitted by law (and except to the extent otherwise agreed in a signed writing by BCG), BCG shall have no liability whatsoever to any Third Party, and any Third Party hereby waives any rights and claims it may have at any time against BCG with regard to the services, this presentation, or other materials, including the accuracy or completeness thereof. Receipt and review of this document shall be deemed agreement with and consideration for the foregoing.

BCG does not provide fairness opinions or valuations of market transactions, and these materials should not be relied on or construed as such. Further, the financial evaluations, projected market and financial information, and conclusions contained in these materials are based upon standard valuation methodologies, are not definitive forecasts, and are not guaranteed by BCG. BCG has used public and/or confidential data and assumptions provided to BCG by the Client. BCG has not independently verified the data and assumptions used in these analyses. Changes in the underlying data or operating assumptions will clearly impact the analyses and conclusions.

The logo for Boston Consulting Group (BCG) is displayed in a bold, white, sans-serif font. The letters 'BC' are stacked vertically, with 'B' on top and 'C' on the bottom. To the right of the 'G' is a vertical ellipsis consisting of four small white dots.

BCG

•
•
•
•