AI DISCLOSURES PROJECT

AUGUST 2025

POLICY BRIEF

Protocols and Power

This AI Disclosures policy brief argues that as AI models become commodities, tech giants are racing to lock in users by hoarding their data. If “context” is king, we need to ensure it remains fluid and accessible by third-party developers. Open protocols, backed by open APIs, can facilitate broad data sharing and healthier AI markets.

Isobel Moure
Program Associate
Social Science Research Council

Tim O’Reilly
Program Director
Social Science Research Council

Ilan Strauss
Program Director
Social Science Research Council

100 YEARS | SOCIAL SCIENCE RESEARCH COUNCIL

About The AI Disclosures Project

Led by technologist Tim O'Reilly and economist Ilan Strauss, the AI Disclosures Project addresses the potentially harmful societal impacts of AI's unrestrained commercialization. By improving corporate and technological transparency and disclosure mechanisms, it aims to ensure that economic incentives don't compromise safety or equity, and avoid fostering excessive risks. Disclosures are vital for well-functioning markets yet remain lacking in AI. Just as financial disclosure standards fostered robust securities markets, standardized AI disclosures can build trust, expedite adoption, and spur innovation. Through research, collaboration, and policy engagement, the AI Disclosures Project aims to develop a systematic framework for meaningful "Generally Accepted AI Management Principles." The project is generously funded by the Omidyar Network, Alfred P. Sloan Foundation, and Patrick J. McGovern Foundation.

DOI: 10.35650/AIDP.4118.d.2025

This policy brief can be referenced as follows:

Moure, Isobel, O'Reilly, Tim, and Ilan Strauss. "Protocols and Power." Social Science Research Council, August 2025. https://www.ssrc.org/publications/protocols-and-power/ [1]

[1] Alphabetical author order. The AI Disclosures Project (SSRC), https://www.ssrc.org/programs/ai-disclosures-project/. A shorter version of this was first published by AI Frontiers. We thank them for allowing us to reproduce the piece.

100 YEARS | SOCIAL SCIENCE RESEARCH COUNCIL

Website: AI Disclosures Project
Substack: Asimov's Addendum
X: @AIDisclosures

1. Summary Of Recommendations

Can we head off AI monopolies before they harden? As AI models commoditize, incumbent Big Tech platforms are racing to rebuild their moats at the application layer, around context – the sticky user- and project-level data that makes AI applications genuinely useful. With the right context-aware AI applications, each additional user-chatbot conversation, file upload, or coding interaction improves results; better results attract more users; and more users mean more data. This 'context flywheel' can drive up switching costs, creating lock-in effects when that accumulated data remains trapped within the platform.

We argue that open protocols – exemplified by Anthropic's Model Context Protocol (MCP) – are a powerful rulebook to keep AI markets 'healthy' – meaning interoperable, decentralized, and transparent. MCP *keeps context moving* between application programming interfaces (APIs) – the data arteries of internet applications – rather than trapped within the platform where it was created. This helps limit the advantages of an incumbent Big Tech platform to uniquely create new AI applications off their enormous pre-existing data reserves. And if done right, MCP can also help AI-specific context, such as conversation history and memory, move between AI applications, instead of being captured by a first mover, such as OpenAI.

MCP's impact, its reach – as an API wrapper or universal adapter – is limited by two things:

- **What data a service, such as GitHub or Slack, chooses to expose through their API.** This is the data context an AI application is limited to and reliant on.
- **What context AI applications, such as ChatGPT or Cursor, themselves are prepared to expose**, such as the user's conversation history and agentic memory.

We argue that open protocols are a powerful rulebook to keep AI markets "healthy" – meaning interoperable, decentralized, and transparent.

To enable MCP to access the full spectrum of user context, at least two things are needed:

1. Open APIs at existing platforms. Keeping APIs open has at least two layers:

    a. **Guaranteed read access** permission for authorized AI applications to user owned & permissioned *data* at major platform services, such as Slack, Gmail, Facebook. This provides essential data context to AI agents. Unbundling context helps unbundle highly integrated tech companies, allowing competing apps the opportunity to build products with the same rich user data.

    b. API write access permission for authorized AI applications that allows them to go from an assistant to an *actor*, who can send emails, make calendar arrangements, update CRM, and more.

Recommendation: Major MCP servers already leverage "*read*" (data) and "*write*" (action / service) permissions; but important platforms, such as Slack, Reddit, and Facebook, restrict outside access. Targeted regulations may be needed to keep the data pipelines of platforms open, as in open banking and the UK's new Smart Data scheme. Open banking regulations show that secure open APIs are possible, either through a single bottom-up market-driven API data sharing ("read") standard (as in the U.S. with the FDX API), or through a top-down regulatory push that made banks share their ("write") services infrastructure and core consumer data (as in the EU's PSD2 / UK Open Banking).

2. **Make agentic AI memory portable.** Strengthen competition within the AI ecosystem through compelling AI applications to make their memory context – the user inferred profile – available via MCP to other third-party AI applications. AI context – conversation history and memory – should be reusable (portable) across the AI ecosystem. The beauty of the MCP's client-server architecture is that any service can be made into a 'server' subject to external calls by a 'client'. In this case, the AI applications would be a server as well as a client.

   MCP memory-specific servers already exist but no standard has emerged for what a memory server might expose (tools, fields, scopes, consent semantics), or for the required security profile. Memory could live in a user-chosen dedicated memory server "hub", per-app servers, or a hybrid (apps expose memory via MCP and can sync to a user's preferred memory hub). A hub would act as a place to hold a user's cross-app context (conversations, summaries, embeddings, tool logs). Apps connect to it to read/write/import/export, as the source of truth across apps.

Recommendation: Structuring market incentives such that AI applications make their memory context portable, is the key challenge. (i) Use MCP to standardize how apps expose and consume memory (a standard "memory server profile"). (ii) Given the sensitive nature of memory and conversations, build security into the architectural standards, as with open banking's FDX API in the U.S. (iii) Existing user data rights laws need to be pushed to their limits to help foster portability in memory. (iv) If third-party memory services don't take off, consider penalties for vendors that lock in user context, including treating the forced use of their own memory backend as illegal tying.

3. **We recommend data usage guardrails to limit how AI services can store and monetize user data.** Open APIs require users to trust developers with shared context. Industry data-usage standards would both level the playing field against incumbents and improve safety.

Recommendation: Data firewalls that separate and protect intimate conversations from commercial targeting, as already appears to be occurring; user erasure rights over their data (already offered by OpenAI); and a default to privacy mode for sensitive queries detected by the AI router or model system.

Architecting an open, interoperable AI stack through the protocol layer is about supporting broad value creation in commercial AI markets, rather than value capture by a few gatekeeping firms. Policy efforts such as the EU's General-Purpose AI Code of Practice do matter, but ultimately it is software architecture that most immediately and decisively shapes market outcomes. ***Protocols*** – the shared standards that let different systems communicate with one another – **function as the deeper law** enabling independent, decentralized, and secure action in digital markets.

2. Protocols As Rules Of The Road

Before getting to the core of our argument, we first motivate why protocols matter for constructing healthy AI markets from the bottom up. Many regulatory interventions in digital markets are imposed from the top down, once market dynamics are already baked into the technology (i.e., the code) that governs the architecture and nature of the service. This strongly shapes whether it is an open, interoperable, value-creating service, or more closed, proprietary, and value-capturing.

The language of protocols should not be the preserve of engineers. Any policymaker or regulator thinking about how to architect durable, dynamic guardrails for digital markets needs to be familiar with it. Regulations layered on top of existing markets can quickly become outdated, easily bypassed, or difficult to enforce. Unlike laws, protocols – as the underlying digital architecture – can embed governance directly into a system's operation, making compliance through disclosure automatic, enforcement continuous, and adaptation to new circumstances far faster than legislative change.

The internet works because of **open protocols**: shared technical rules that let different systems "speak the same language." Some protocols connect within the same layer (like SMTP, which lets email servers exchange messages), others connect across layers (like HTTP running over TCP/IP, or DNS translating web addresses into the numerical IP addresses computers use).

The internet works because of **open protocols:** shared technical rules that let different systems "speak the same language."

By keeping these communication rules open and consistent, the internet allows devices, networks, and applications around the world to interoperate without needing to be built by the same company. This reduces any one firm's ability to act as a chokepoint – gatekeeping access and control – to decide "who gets what and why" from the system.

As Tim O'Reilly noted previously for the AI Disclosures Project:

> We are increasingly coming to see disclosures through the lens of networking protocols and standards. Every networking protocol can also be thought of as a system of disclosures. But these disclosures are far more than just a warning label, or a mandated set of reports. They are a form of structured communication that enables independent, decentralized action.

Tim then notes why this matters for AI's "market structure":

> The race for first mover advantage by the large centralized AI providers like OpenAI and their business model of providing AI access through metered API subscriptions suggests a hub and spoke railroad design, while a world of open weight AI models connected by new modes of standardized communication could look more like a road system, or today's World Wide Web....If we want a world where everyone, not just AI model developers and those building on top of their centralized networks, is able to innovate and to offer their work to others without paying a tax to access centralized networks, we need a system of disclosures that enables interoperability and discovery.

In this approach, protocols, as a type of disclosure, can architect healthier AI markets, not after things are already too far gone, but through operating as foundational "rules of the road ... that enable interoperability":

> In short, we need to stop thinking of disclosures as some kind of mandated transparency that acts as an inhibition to innovation. Instead, we should understand them as an enabler. The more control rests with systems whose ownership is limited, and whose behavior is self interested and opaque, the more permission is required to innovate. The more we have built "the rule of law" (i.e. standards) into our systems, the more distributed innovation can flourish.

3. How We Got Here: From Commoditized Models To Context-Rich Applications

In a fevered race to blitzscale its way to platform dominance, OpenAI took an early lead. ChatGPT became the fastest-growing application in history, and it was easy to assume that the next step was to turn it into a platform. OpenAI attempted to become a developer platform first with plugins, and then with its GPT Store.

But it hasn't entirely gone to plan. OpenAI's models don't seem so special anymore. Open-source models like Kimi K2 (by Moonshot AI) have competitive capabilities and are free to use. Sensing the turning tide, application-specific companies like Perplexity and Cursor struck gold by taking off-the-shelf models from multiple providers, scaffolding them for specific uses, and charging for premium access while avoiding vendor lock-in. Developers can easily choose their preferred model within these applications. And using platforms like OpenRouter, developers can even switch models dynamically based on pricing or features.

Cursor, an AI-first code editor, went from $0 to over $100 million annual recurring revenue (ARR) in 18 months, proof that context-driven retrieval-augmented generation (RAG), with a native AI design, can beat incumbents sitting on more user data.

As foundation models commoditize, competition is shifting up the stack to the application layer, where proprietary user and project data – **the "context"** – is the secret sauce. Tech giants are racing to enclose and own this context exclusively: conversation histories, memory stores, workspaces, codebases, documents – anything that helps their agents predict and assist better. OpenAI, Google, and other model vendors lean on chatbot interaction logs as persistent memory, while application specialists like Anysphere's Cursor and Perplexity similarly harness project and user data to boost their models' usefulness.

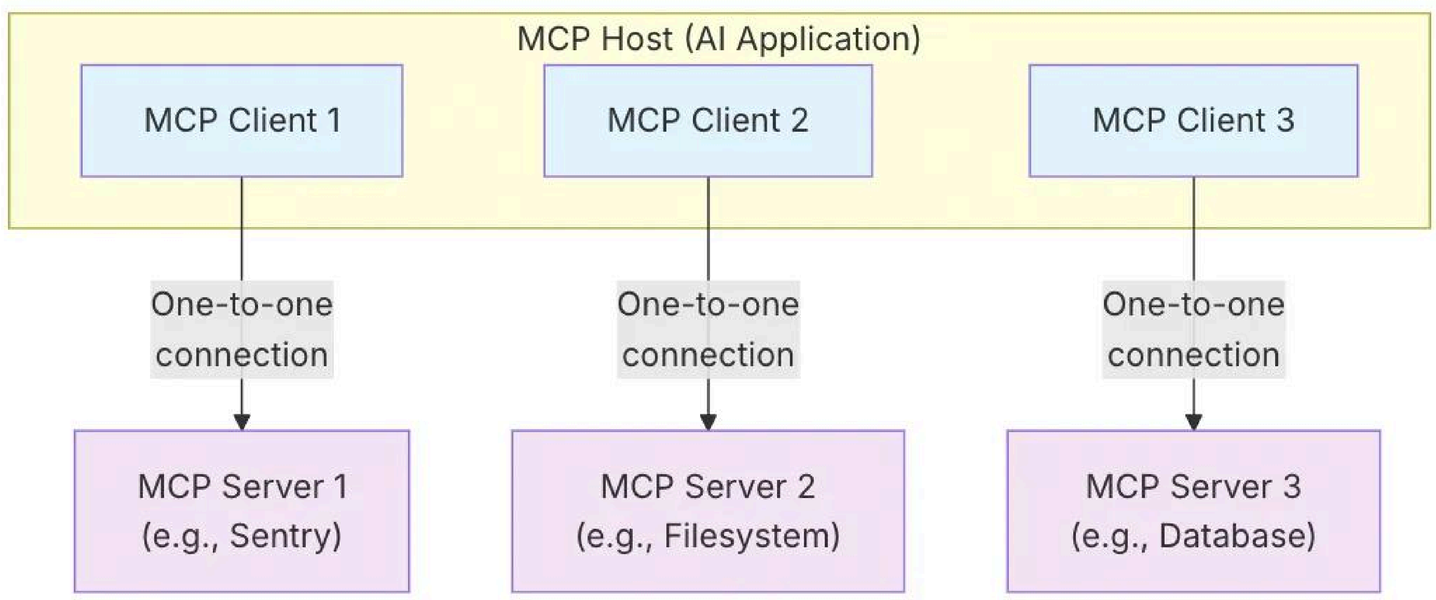
This forces a crucial decision for the market: *will AI applications grow on closed standards that let a few gatekeepers dictate terms and extract outsized rents, or on open standards that keep context portable and the architecture permissionless?*

The stakes are high. Born on open protocols, the web evolved into ecosystem applications dominated by Amazon, Google, and Meta, which first beat rivals by simply working better. Google was the best at matching searchers with information and ads; Amazon surfaced the best products at low prices; and Facebook matched its users with a unique feed crafted only from their friends and people they chose to follow.

But success conferred durable power that could be abused. As growth slowed, the winning companies shifted from creating value to extracting it. In our past work, we described this process using the language of economic rents: winners first gain "Schumpeterian rents" for innovation, but once markets mature, these turn into extractive rents aimed at preserving dominance and squeezing users and developers. Cory Doctorow frames this process vividly as "enshittification". AI's enshittification could involve weaker safety guardrails, higher prices, less user privacy, and lower-quality information or agentic assistance. In short, when commercial incentives go unchecked, models get tuned to serve the provider's interests over those of users.

Attempts by OpenAI to build a platform by locking in developers and users resemble Facebook's failed attempt to build a platform. But as Bill Gates is said to have commented: "This isn't a platform. A platform is when the economic value of everybody that uses it, exceeds the value of the company that creates it. Then it's a platform." That kind of platform is almost always enabled by open standards. By contrast, enclosure stifles complements, spurs multi-homing, and draws regulatory fire.

Anthropic has taken a different route, developing the Model Context Protocol (MCP) as an open protocol – a shared set of rules that anyone can use for free – that standardizes how AI applications request information and actions from external services, thereby facilitating equitable developer access to external tools and data (context). This is how networked markets grow: by enabling an architecture of participation through which every new entrant makes the market more valuable for everyone else.



The basic client-server architecture of MCP.
Source: https://modelcontextprotocol.io/docs/learn/architecture

MCP's take-up has been explosive. Today there are well over 5,000 MCP servers that can connect to the hundreds of AI apps that have integrated MCP. Faced with rapid adoption by third-party developers, AI model developers like OpenAI and Google have announced that they too will support MCP. But these same incumbents are already pushing back.

4. Context Is Now King

Context underpins the new gold rush in AI markets – as applications look to design more efficient workflows. AI systems thrive on context – the user data that lets an AI system tailor its behavior to users, their requests, and the tasks at hand. When properly mined, this user data allows for personalized and efficient predictions. Think of a stateless, factory-setting AI model as a borrowed phone: the hardware is powerful, but without your contacts, messages, location, and log-ins it can't really help you.

Context has many layers: *across time* as a living "state", so that each user prompt builds on what came before; and *across people*, as a multi-user setting (say a Slack thread or collaborative document). We emphasize two layers: *micro-context* captures whom the system is helping right now (associated with their preferences, language, and current query). While *macro-context* covers the task environment, as the external interface frame that shapes what a sensible answer looks like. This includes project files and live data feeds.

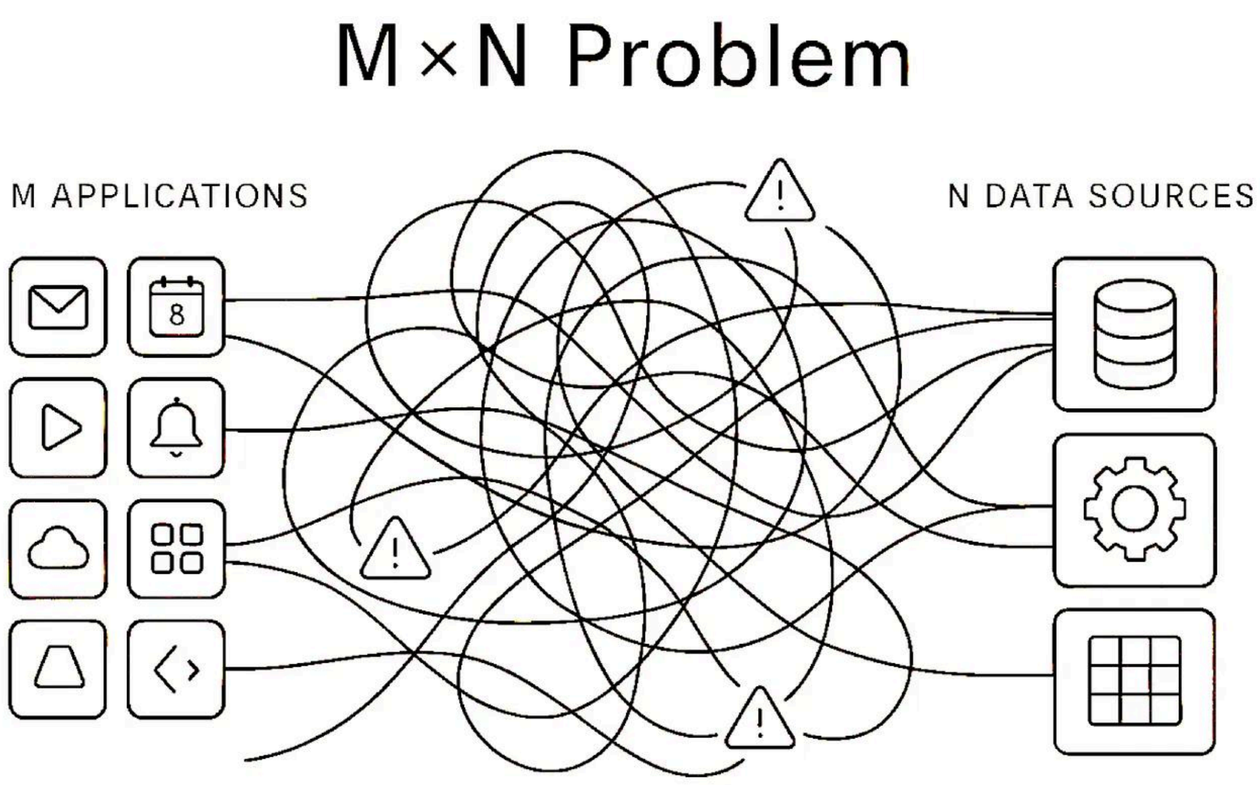
Big AI companies are using context to grow their moats and lock in users in at least two ways. *The first is through product bundling.* OpenAI pushing into search, research, and coding (including through acquisitions); Google threading Gemini into Workspace; Microsoft embedding Copilot across its 365 productivity apps. Bundling aggregates the data surface and raises switching costs.

The second is through building context as a central product feature. OpenAI now offers persistent memory that stores personal details – like "has a child" or "diagnosed with ADHD" – to shape future replies; Meta announced it will collect cross-site user data to personalize its AI assistants; and Google will remember your writing style to tune its AI-generated Gmail replies. Binding the app and its context to the model locks users in and starves rivals. Such bundling is fertile ground for enshittification.

Importantly, this process relies on Big AI gathering ***explicit*** *user signals* – their prompts, docs, API calls – and distilling them into an ***inferred*** (i.e. *implicit*) *user preferences* profile that lets their model deliver more relevant, efficient predictions inside each user's unique workspace. How to enable this flywheel for new entrants is now the challenge. Enter Anthropic's Model Context Protocol (MCP).

5. Can Protocols Create A Level Playing Field?

MCP standardizes how models get context. Through MCP, AI applications request tools, data, and actions from external services through a universal adapter. Instead of custom integrations for each AI application → service pairing (Cursor → GitHub, Claude → Google Drive), any AI app (**the MCP client**) can use any MCP-compatible service (**the MCP server**), making models more interchangeable. MCP also creates an agentic interface that allows an AI agent to decide what to do, based on the language of tasks, not endpoints. This reduces the N×M integration tax, allows small firms to rent rather than build tooling, and weakens vertical exclusives, but services must still opt in through APIs.



Why connectivity can become a hard problem.
Source: https://www.shakudo.io/blog/mcp-model-context-protocol

Because MCP is client-agnostic, any AI app can use any external service, which in turn makes switching between models far easier – either by switching between model service providers that support MCP, or by building your own MCP client and using any model service. When an AI app's context is portable, models become more interchangeable.

MCP is the ultimate ***un-bundler*** *of context*: any compatible AI app can reach any service that exposes an MCP server, allowing an enriched prompt to then be sent to the model. But services still have to opt in by making their content available through APIs.

This shifts the competitive gravity "up the stack," away from the model developers and to the application that develops the winning "context flywheel" – the rich, structured user and project data layer. App-level data portability and governance – including pricing, permissioning, and any preferential access into Big Tech controlled data sources – then become the new battleground.

Although MCP reduces integration friction, interoperability alone doesn't guarantee competitive markets. We've seen this before: open protocols like HTTP for web browsing and SMTP for email enabled permissionless entry of new applications, yet markets still tipped. Google now dominates both email and the browser because of a superior product and cross-app integrations.

5.1 MCP's Impact on the AI Market So Far

Incumbents have rushed to generically insert AI into every legacy product – the quickest go-to-market strategy but the shallowest integration. Meta surfaces an assistant in nearly every app. This has only made building cleaner MCP-enabled alternative applications far more attractive. AI-native tools like Perplexity offer further encouragement to developers, showing that users will pick a specialized experience over a retrofitted one, like the AI-layered Google Search.

Incumbents have rushed to generically insert AI into every legacy product – the quickest go-to-market strategy but the shallowest integration.

Unsurprisingly, new MCP servers have surged, as noted above. MCP is now widely integrated into cloud computing offerings, exploding among developer communities, and has a dedicated steering committee, along with a governance and stewardship framework. However, MCP-enabled integrations may also be boosting usage of incumbent model developers' chatbots as they gain access to more tools and third-party context. And for repeated production workflows, code-based frameworks might be more efficient than an inference-driven workflow.

Security is another limitation. In agentic AI workflows, models can autonomously query databases, call APIs, and edit documents on a user's behalf. That creates new security needs: binding actions to the human behind the agent, preventing overreach, and scoping permissions to specific tasks and time windows. Two controls matter: authentication (Who is the user/client? Is the server genuine?) and authorization (What can this client do – and for how long and on whose behalf?). MCP standardizes authorization (for HTTP transports) via OAuth 2.1.[2] This has a number of limitations, however.

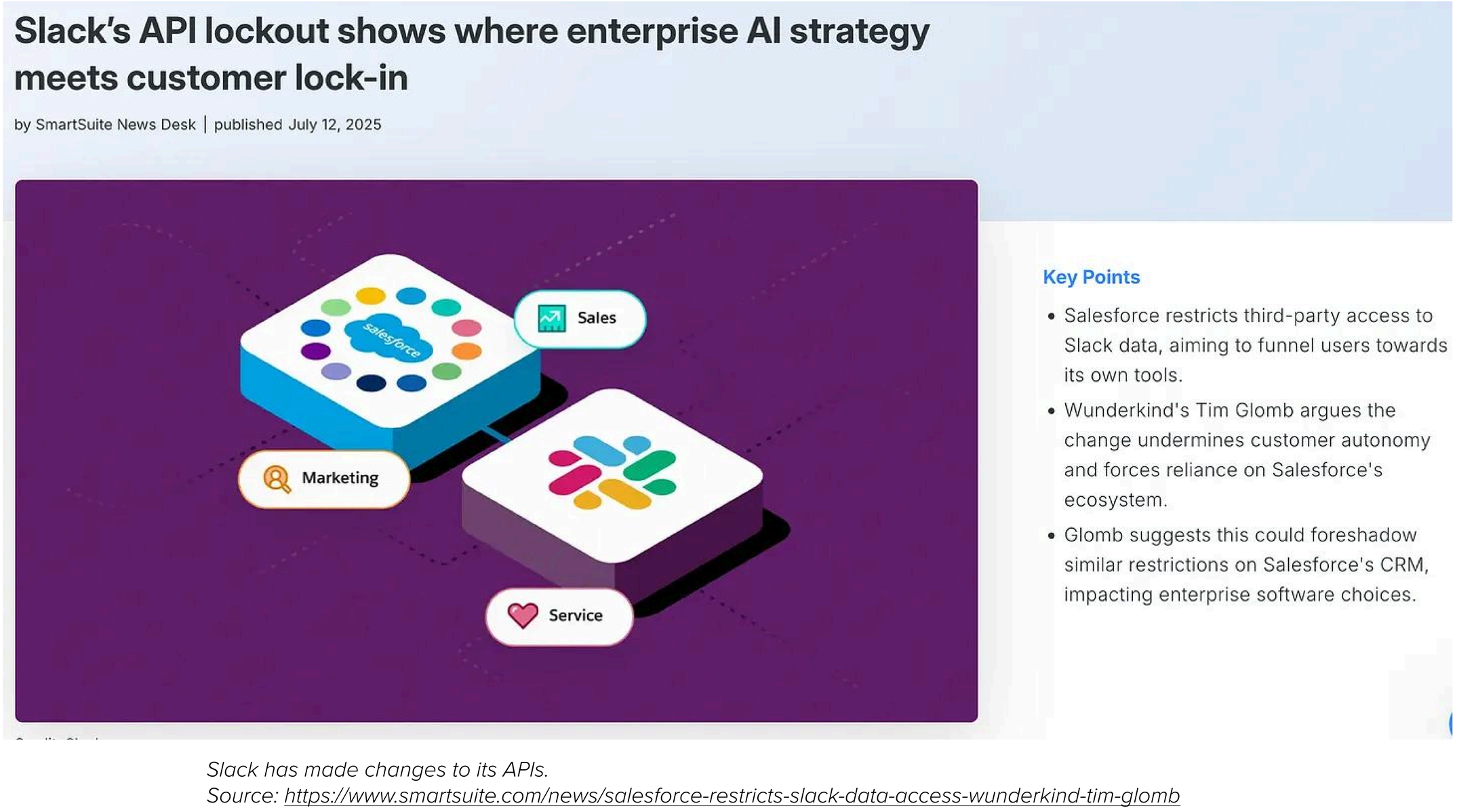
Lastly, there are early signs that AI model developers may resist interoperability more broadly. Anthropic temporarily cut off the coding application Windsurf's direct (first-party) access to its high-performing Claude models. Windsurf was growing too popular and was set to be acquired by OpenAI, a direct competitor to Anthropic.

[2] Authorization is optional in MCP, but when used, implementations must follow the OAuth 2.1 profile, PKCE, and related requirements over HTTPS. Server identity is provided by HTTPS/TLS (optionally mTLS). Scopes and access tokens then constrain what the client can do.

6. The API Gatekeeping Problem: MCP Vs. Walled Gardens

APIs are the gateway through which an MCP client — the AI application — can access third-party data and tools, thereby breaking down a platform's "walled garden" of proprietary services and datasets. **But MCP can liberate context only when a third-party service offers a sufficiently rich API** (and keeps it open). And because platform owners control those APIs, they have an incentive to constrain what MCP can touch, to protect their competitive edge. This manifests in two ways:

1. **Access Risk.** Services can simply shut off API access entirely or greatly degrade access. Recent API paywalls and shutdowns at Reddit, Twitter, and Meta show how access can vanish overnight. Enterprise services like Salesforce (owner of Slack), Atlassian, and Notion are now limiting Glean's API access – a context platform – as they launch competing products, while Slack's new API limits may harm developers in general.



Slack has made changes to its APIs.
Source: https://www.smartsuite.com/news/salesforce-restricts-slack-data-access-wunderkind-tim-glomb

2. **Context-depth risk ("personalization gap")**. Platform APIs expose posts and files but rarely the behavioral profiles that power their own personalization, leaving newcomers with a cold-start handicap. Meta, for example, personalizes its own chatbot with Facebook and Instagram history, but offers no Graph API for third parties to fetch that full profile, or even just detailed aspects of their explicit data and implicit (inferred) profile. Similarly, OpenAI's "memory" feature is confined to ChatGPT. OpenAI does not allow developers to access a user's 'memories' via an API, even with user consent.

7. To Save AI From Enshittification, Support Protocol-Level Interventions

The promise of open AI ecosystems hinges on a critical bottleneck: access to user context. While protocols like MCP offer the technical infrastructure to connect AI applications with diverse data sources, their potential to create new markets remains constrained by two fundamental barriers.

First, the data – and tools – that platforms choose to expose through their APIs determine the contextual richness and utility of AI applications. When GitHub, Slack, or Gmail restrict API access, they effectively control what AI agents can know and do. **Second, AI applications themselves risk turning into gatekeepers** when they refuse to share the user context they have accumulated – conversation histories, learned preferences, and the evolving memory that makes AI interactions increasingly personal and powerful. The technical means for interoperability exist. But a lack of interconnecting infrastructure, especially effective protocols, limits the formation of these markets.

This double-sided gatekeeping threatens to recreate the same platform monopolies that have dominated the internet era, only now with context as the new moat. The solution requires coordinated intervention at both ends of the data pipeline – compelling platforms to open their APIs while ensuring AI applications make their accumulated context portable. As Robin Bjorn aptly puts it, the challenge is "to make markets, not run them" – building the structures that enable competition without micromanaging outcomes.

7.1 Breaking Open the Platform Gates: Mandatory API access

The way forward begins with dismantling API gatekeeping at major platforms. This is why we recommend open API access to user-related data held by major platforms and digital services for developers. Real-time API access would be needed for active context, while batch exports would work for historical data. This isn't about granting blanket access to all of the data on platforms, but rather ensuring that user-owned information – the posts they write, the messages they send, the behaviors they generate – can flow to authorized AI applications with user consent. The principle is straightforward: if data originates from the user, the user – and any developer the user authorizes – should be able to say where it goes ("portability").

The UK's new Smart Data scheme advances something similar to an "open API" mandate, while EU data portability mandates are advancing through Article 6(10) of the Digital Markets Act (DMA) for gatekeepers, GDPR Article 20 for "data subjects", and the new EU Data Act (starting September 12 2025) for businesses and consumers.

Open APIs should ideally have two layers:

First, guaranteed core "read" access. Authorized AI applications should have real-time, privacy-respecting permission to read user-owned and user-permissioned data at major platforms via their APIs. This is the raw material of useful agents. Unbundling context helps unbundle highly

integrated firms: if competing apps can lawfully and securely reach the same consented data, they can build substitutable services without striking bespoke deals.

Under GDPR, CCPA/CPRA, and related data regimes, users gain control rights – access, portability, deletion – over their raw personal (such as posts, conversations, contact information) and basic behavioral data (such as clicks, views, time spent). Companies own the algorithms and models that process that data. California goes further, though. The CPRA classifies even inferred attributes (e.g., "likely to buy a car") as personal information that must be disclosed on request. But turning those individual rights into a market-wide lever requires an open API layer.

Second, guaranteed core "write" access. Authorized AI applications should be able to act on the user's behalf – send email, schedule meetings, update a CRM, post to a workspace – under tight scopes and time-boxed consent. Without the application getting "write" permission from a service's API, agents remain spectators; with it, they become genuinely useful.

For AI and digital platforms, we recommend starting with a fairly targeted approach, data ("read") rights for APIs and core actors covering: designated "gatekeepers" under EU Digital Markets Act criteria, plus all major AI model providers who control user memory. They should be required to expose user-owned contextual data through APIs to accredited developers at zero cost.

This would include three categories of data: (1) provided data that users knowingly supply (profiles, settings, uploaded content), (2) observed data generated through user activity (clickstreams, usage logs, purchase history), and (3) functionally necessary inferences that materially shape the user experience – critically including AI memory. This combines elements from GDPR with California privacy laws. [3]

Consider what comprehensive open APIs would enable. An AI assistant could seamlessly access a user's Gmail to understand communication patterns (read access), parse GitHub repositories to assist with coding (write access), review Slack conversations (read access) to prepare meeting summaries (write access), and analyze Facebook interactions to understand social context (deeper read access). Currently, these capabilities exist in fragments – some platforms offer limited access, others wall off their data – and tools – entirely. This patchwork approach fragments the AI experience and entrenches incumbent advantages.

A successful precedent already exists in banking. Under the EU's PSD2 directive,[4] banks must provide licensed fintech companies with free, real-time access to core account data and

[3] Where portability – available via access – covers "provided" and "observed" data but typically excludes inferred / derived data. And California law that treats inferences as personal information with a right to know.

[4] PSD2 (EU/UK) is a regulatory law that mandates banks provide licensed third parties with both account information services (reading transaction data) and payment initiation services (pushing payments from user accounts), while also requiring strong customer authentication. In contrast, FDX (U.S.) is a voluntary industry standard that only harmonizes how financial data is shared between institutions through standardized APIs and data models, but creates no legal obligations for access and excludes payment initiation entirely. While PSD2 establishes legally enforceable rights for third parties to both access data and move money, FDX simply provides a technical framework for consumer-authorized data sharing, with U.S. payments continuing to rely on separate traditional rails like ACH, RTP, and FedNow through independent arrangements.

payment functions. The U.S.[5] has taken a more market-driven approach with the FDX API standard, which enables secure, standardized sharing of financial data ("read" access) between banks and third-party developers. Both models show that sensitive tooling access and data can be shared securely at scale when proper frameworks and incentives exist.

Of course, not everyone embraces this open approach. JPMorgan's CEO Jamie Dimon has publicly criticized open banking, arguing that fintech intermediaries create security risks and strain bank systems. Similar resistance from tech platforms is inevitable, who might argue that opening APIs compromises security, enables bad actors, or undermines their business models. But the banking experience shows these concerns can be addressed through proper accreditation, rate limiting for abuse prevention, and clear liability frameworks.

Not everyone is happy with open banking in the U.S., including JPMorgan's CEO, Jamie Dimon. Source: https://www.cnbc.com/2025/07/28/jpmorgan-fintech-middlemen-plaid-data-requests-taxing-systems.html

Unlike banking's standardized records, AI context and digital services span many more data types and abstractions – and are rapidly evolving. Moreover, core user context categories differ by platform type and platform-specific data elements exist. Despite the complexity, developer needs can quickly help guide changing access requirements over time. In the U.S., the market-driven open-banking API remarkably provides access to 660 unique financial data elements.

The EU's Digital Markets Act provides an ideal testing ground for this approach. It already designates gatekeepers, mandates free API access under Article 6(9) for individuals and authorized third parties, includes provisions for inferred data, and carries substantial fines for non-compliance. Small-scale experiments through the DMA could demonstrate feasibility before broader implementation.

---

[5] The U.S. banking sector has been moving in the direction of open banking too. Though, it may be moving backwards now. In the U.S., open banking grew from the bottom up through markets. Screen scraping gave way to bilateral API deals, to avoid unauthorized access and upstarts potentially stealing market share. To avoid hundreds of one-off contracts, banks and fintechs formed the Financial Data Exchange (FDX). FDX now maintains the de-facto U.S. open-banking spec (FDX API v6.4) and runs a membership programme that is voluntary, not a statutory accreditation. 114 million customer connections now flow through FDX-aligned APIs. The CFPB created a recognition program for "standard-setting bodies" and in January 2025 formally recognized FDX – cementing it as the de-facto U.S. open-banking standard even though participation remains voluntary. Even before the 1033 rule, CFPB principles (2017) and Treasury guidance (2018) pointed toward consumer-authorized sharing – nudging the market to self-organize around a standard. So government pushes helped.

7.2 Making AI Memory Truly Portable

Opening platform APIs addresses one side of the context bottleneck. The other side requires making AI memory itself portable between AI applications. As AI assistants accumulate rich contextual understanding of users – their preferences, communication styles, goals, and histories – this memory becomes a powerful competitive moat. Users risk becoming locked into specific AI providers not because of superior models, but because switching means losing months or years of accumulated context.

The technical architecture for memory portability already exists within MCP's client-server model. Any service can function as a server exposing data to external clients. AI applications could simultaneously act as both clients (consuming context from other sources) and servers (exposing their memory to other applications). This bidirectional capability transforms memory from a proprietary asset into a portable resource that follows the user.

Memory portability could take several forms. A user might choose a dedicated memory server hub that acts as the central repository for cross-application context – storing conversations, summaries, embeddings, and tool logs. *Applications would connect to this hub to "read" and "write" memory, treating it as the authoritative source across the AI ecosystem.* Alternatively, each AI application could expose its memory via MCP while maintaining synchronization with the user's preferred memory backup. Some users might prefer a hybrid approach, with certain sensitive memories kept local while others sync across applications.

The challenge lies not in technical feasibility but in creating market incentives for adoption. MCP memory-specific servers already exist. But AI companies currently have every reason to hoard user context and no compelling reason to share it. Several interventions could shift these incentives.

First, standardizing how applications expose and consume memory through MCP would reduce implementation costs and complexity. A common 'memory server profile' would define standard tools, fields, scopes, and consent semantics that all applications could adopt. However, the market is still emerging here, with Anthropic only very recently including a competing memory function of sorts with its AI product, Claude.

Second, security must be built into these standards from the ground up. The sensitive nature of AI conversations – which might include therapy-like discussions, business strategies, or personal reflections – requires strong protections. Following open banking's example, the memory portability standard should prescribe specific security profiles, consent flows, and audit requirements rather than leaving these as implementation details.

Third, existing data rights laws need to include AI memory. As noted above, under GDPR, users already have portability rights over provided and observed data. California's CPRA goes further, classifying inferred attributes as personal information subject to disclosure requirements. AI memory clearly falls within these frameworks – it's both observed (based on

and inferred (patterns learned from those interactions). Regulators should explicitly confirm that AI memory qualifies for portability rights and enforce these provisions.

Lastly, if market-driven adoption fails, stronger antitrust measures may be necessary. Forcing users to use a platform's own memory backend, in conjunction with its app, could be explored as 'illegal tying' under competition law. Platforms that lock in user context without providing memory and context export capabilities would then face penalties. The goal isn't to punish innovation but to ensure that user context remains under user control.

7.3 Establishing Data Usage Guardrails

Open APIs and portable memory create new risks alongside their benefits. When users grant AI applications access to their email, calendar, and conversation history, they are trusting developers to handle this intimate data responsibly. Without clear boundaries, the same context that enables helpful personalization could fuel manipulative advertising or privacy violations. Industry-specific data usage rules would both level the playing field against incumbents and create safer AI ecosystems. **These guardrails should start with three core principles:**

First, data firewalls must separate and protect intimate conversations from commercial exploitation. An AI learning that a user is vegetarian to provide restaurant recommendations serves a legitimate purpose. But mining therapy-like conversations to target vulnerable users with manipulative advertising crosses an ethical line that regulation must enforce.

Second, users need comprehensive erasure rights over their preference profiles and memories. They should be able to review what an AI has learned about them, edit incorrect inferences, and delete entire memory segments at will. OpenAI's ChatGPT already offers a basic version of this capability, proving it's technically feasible. Making erasure rights universal would prevent lock-in through accumulated context while giving users meaningful control over their digital selves.

Third, privacy defaults should protect users during sensitive interactions. When an AI router or model detects queries about health conditions, financial problems, or personal crises, it should default to a privacy mode without long-term memory or behavioral tracking unless users explicitly opt in. This protects vulnerable users while preserving the benefits of personalization for routine interactions.

These guardrails might seem to constrain AI development, but they actually are far more likely to enable sustainable growth by building user trust. When users know their conversations won't be exploited, they engage more openly. When they can take their context with them, they're more willing to try new services. When privacy is protected by default, the entire ecosystem becomes safer for experimentation and innovation.

When privacy is protected by default, the entire ecosystem becomes safer for experimentation and innovation.

Ultimately, control over user context – not raw model power – will determine the winners in the AI race. The most sophisticated language model becomes useless without understanding user needs, preferences, and goals. The simplest model becomes powerful when it can access rich, relevant context. By keeping context fluid between competitors through open protocols and portable memory, we can ensure that AI markets develop around user value rather than platform lock-in. The choice is ours: design competitive AI markets around open principles, or accept another generation of platform monopolies.

The choice is ours: design competitive AI markets around open principles, or accept another generation of platform monopolies.

Thanks to Alex Komoroske, Chris Riley, David Soria Parra, Guangya Liu, Benjamin Mathes, and Andrew Trask for reading and/or commenting on this article. All errors are ours.