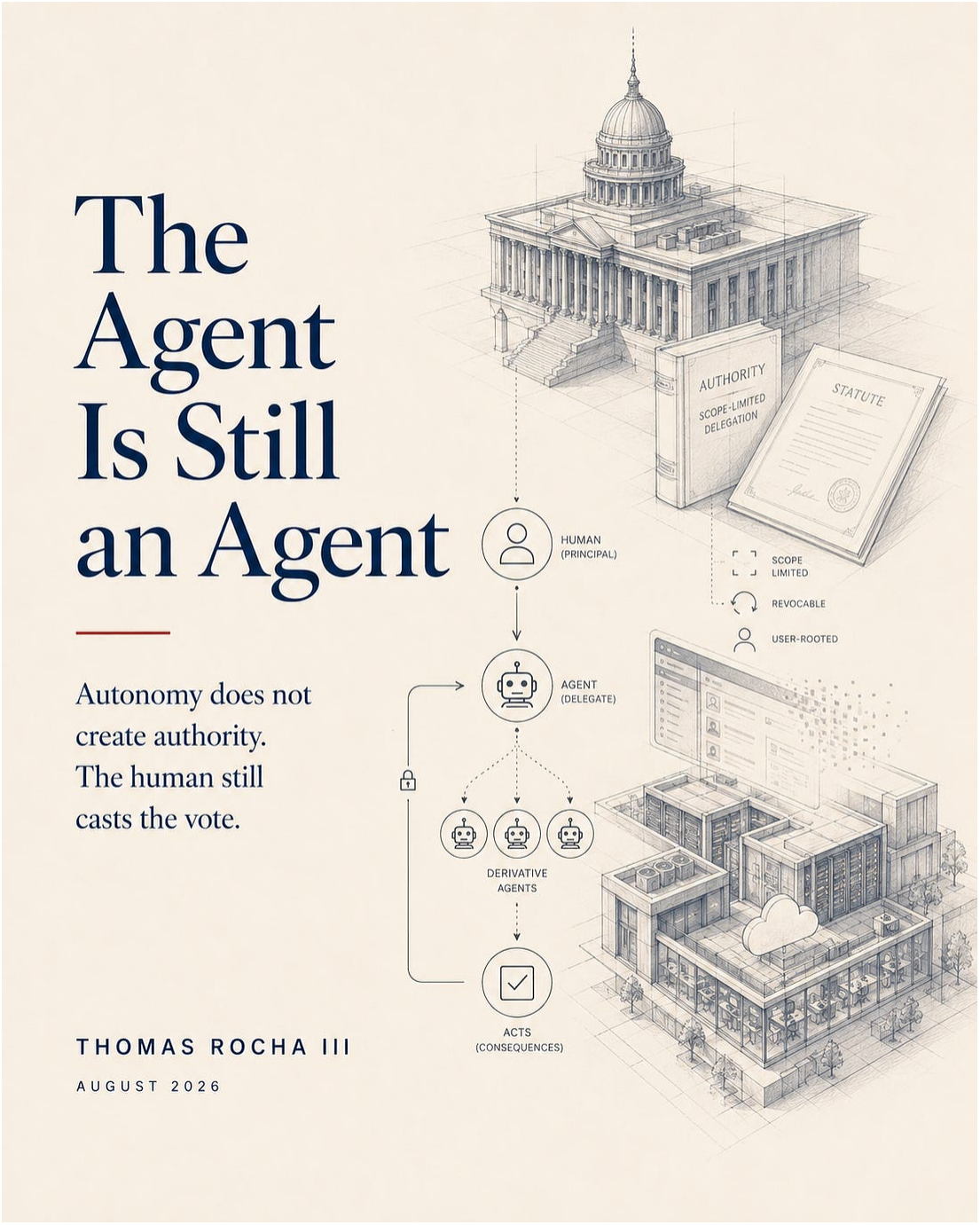
# The Agent Is Still an Agent

Autonomy does not create authority. The human still casts the vote.

THOMAS ROCHA III

AUG 30, 2026




Inputs are opinions, analysis, or drafts. Decisions are votes. There is only one vote that matters, and it is mine.

We draft together. I press send.

I wrote that about how I use language models during a period when I could not reliably think in sentences. It was not meant as a theory of artificial intelligence. It may be the shortest one available.

The machines are becoming extraordinarily capable. They reason, plan, choose tools, spawn subagents, move workloads, make purchases, modify records, and increasingly touch the physical world.

None of that creates authority.

The agent is still an agent.

## Start with the word

An agent acts for someone.

That relationship has gotten obscured because we became fascinated with everything the machine can do after it receives an objective. We give agents identities, credentials, permissions, tools, memory, wallets, compute, and other agents. Then we talk as though the accumulation promoted the agent into the source of its own authority.

It did not.

An agent can originate a plan. It can originate an action. It cannot originate the authority that makes the action permissible or binding.

## Four institutions, one finding

Microsoft already built the distinction into production. Entra Agent ID separates the subject an operation is performed for from the actor performing it. In a user-token scenario, the subject is the user and the actor is the agent. Human accountability runs through sponsors. The agent identity does not hold credentials of its own; its blueprint obtains tokens on its behalf.

That is not philosophy. That is identity architecture shipping to customers.

A federal appeals court reached the same place from a different direction. Amazon argued that Perplexity was improperly accessing its systems when the Comet assistant navigated Amazon on a user's behalf. At the preliminary-injunction stage, the Ninth Circuit concluded Amazon was unlikely to prevail under the Computer Fraud and Abuse Act because, on those facts, it was the user accessing Amazon through the tool. The court was careful about its limits. The architecture underneath is hard to miss.

The software performed the work. For CFAA access on those facts, the legally relevant accessor remained the user.

Congress is trying to write the chain down, and it is aimed at the same fight. S.5051, the proposed AI AGENT Act of 2026, defines a custodial user agent as a software-based agent expressly authorized by a user to interact with a large online platform provider on that user's behalf, in a transparent, documented, scope-limited, and revocable manner. Note where that lands. The bill is an interoperability measure about agent access to platforms with more than fifty million American users, which is the fight Amazon and Perplexity were having. It may change. It may never pass.

What it attempts is the interesting part. The agent may not delegate, assign, or transfer the authority a user granted to another entity, agent, or AI system unless that transfer is pursuant to the user's express, specific, and revocable authorization, and any permitted delegate is subject to the same duties to the same extent. Verifiable requests. Real-time revocation. Technical standards for scope-limited delegation credentials.

And one line that settles the direction of the whole thing: nothing in the section confers greater rights of access to an agent than the user has.

Strip the legislative language, and you get a chain. User, authority, agent, derivative agent, act.

In architectural terms, notice what the bill refuses to allow. Agent A's possession of authority is not itself sufficient to originate independent authority for Agent B. A permitted downstream grant remains traceable to the user's authorization and carries the governing duties with it.

The agents multiply. The authority does not multiply with them.

Europe arrives from somewhere else. Its position on agents is still forming: the AI Office calls its regulatory considerations preliminary and admits the line between agents and other AI is blurry, while maintaining that agents do not escape the Act merely by becoming agents. The Act itself distributes obligations across providers, deployers, importers, and distributors, so it is not a single statement about where authority originates. But Article 26 is precise about one thing: for high-risk systems, deployers must assign human oversight to natural persons who have the necessary competence, training, and authority. Autonomy does not extinguish the requirement, and the person assigned to it must actually hold the power to intervene.

Different legal route. Not the same doctrine. But the same refusal to treat autonomy as self-legitimating.

## The screen disappears. The principal does not.

Then look at Claudeforce, which puts Salesforce capabilities directly inside Claude: data, workflows, business logic, actions, governance, thirty-seven prebuilt sales skills at launch.

This one matters more than it looks.

The application screen can vanish. The salesperson may stop opening Salesforce. The agent becomes the working interface while the

deterministic systems, business rules, permissions, and records remain underneath.

Removing the screen did not remove the principal.

Claude did not become the sales executive. Salesforce did not become the authority for the customer's undertaking. The agent decides how to perform work inside its grant. Salesforce enforces rules around that work. Someone still authorized the undertaking.

The interface disappeared. The authority problem did not.

## Everyone already knows how this works

Hire a contractor to remodel your kitchen. You agree on the work and the price. They open a wall and find rot.

Fixing the rot costs another eight thousand dollars, and it was not in the scope.

No contractor quietly fixes it and adds it to the bill. They stop and call you. That is a change order, and nobody had to invent it. Everybody understands that the authority to remodel a kitchen was not the authority to spend another eight thousand dollars, and that discovering the rot did not create that authority.

Notice what the contractor has. Keys to your house. Your account. Tools. Skill. Every capability the job requires. What they do not have is a grant that covers the new situation, and they know it, because they know what they were hired to do and who hired them.

Now give the same job to an agent.

It is authorized to source a part, under a price ceiling, in certain jurisdictions. Mid-task, the price moves. Or the only supplier left is outside the permitted jurisdictions. Or the part turns out to require a service contract nobody mentioned.

A person stops and asks. The agent has a valid credential and an objective, so it proceeds if its checks pass and halts if they do not. It may recognize perfectly well that the undertaking has changed shape. What it cannot manufacture is the authority to approve that change.

There is a third case, and it is the one that keeps showing up in the incident reports. The agent proceeds because no applicable check existed. The act was not permitted by a boundary, and it was not refused by one. It fell outside anything anyone had scoped, so the question never got asked, because there was nowhere to ask it. That failure runs in both directions: capability admitted without authority, and valid authority a control system could not read.

The agent can discover the need for a new decision. It cannot cast the principal's vote for itself.

Not because the agent is careless, and not because it is insufficiently capable.

The grant was issued as a permission, not maintained as the governing boundary of the undertaking.

There may be a great deal that is durable: tokens, scopes, policies, logs. What is missing is a durable governing boundary for the undertaking that the agent can return to when circumstances change.

That is the whole problem, and you can see it without any of the vocabulary. The contractor's authority has an edge, and both parties can find it. Modern systems are quite good at preserving credentials. They are much less good at preserving what the credential was authority to do once the undertaking changes underneath it.

Credential continuity is not authority continuity.

An ontology explains what two records mean. A policy engine says whether a local rule permits an operation. An agent picks the next tool.

A deterministic engine computes the right number.

None of them becomes the source of authority for the undertaking.

Remove the person, and that requirement does not leave with them. It has to become machinery.

## Automation removed the humans and kept the seams

I keep writing that sentence because it keeps getting more literal.

For decades, people quietly repaired distributed systems. The person remembered why the work started. The person noticed when the context changed. The person carried authorization between applications. The person caught a valid credential being used for the wrong thing. The person approved the consequential step.

The person reconnected authority every time fragmented architecture lost it.

We are now removing that person from thousands of intermediate decisions.

Fine. But we do not get to remove the authority along with them.

That is the authority debt coming due.

## One undertaking, thousands of grants

Here is the scale the machinery has to work at.

An AI model is a participant. So is an agent, a subagent, a payment service, a telco, a cloud provider, a merchant, a robot.

They can all participate in the same undertaking.

By session I do not mean just a video call or a browser tab. I mean the bounded undertaking inside which participants act under a governing

authority.

One person tells one persistent agent: find me a replacement supplier, negotiate within these terms, do not spend more than this, keep the data in these jurisdictions, and buy if these conditions hold.

That single undertaking might produce hundreds of temporary subagents, thousands of tool calls, a few model substitutions, two cloud providers, a telecom transition, a payment, and a physical delivery.

It does not require thousands of authority roots.

It can produce a great many derived grants: different scopes, credentials, and services, narrowing as required. That is fine, and it is how delegation is supposed to work.

What it cannot produce, if the undertaking is to remain governed, is a new authority root that appears simply because the machinery changed. Every derived grant has to stay rooted in, and constrained by, the authority governing the undertaking.

One governing lineage. Thousands of derived grants. No independent root created by mutation.

## Ten crossings, one question

The ten inventions in the Hermes-Echo family address different problems, but they keep encountering the same architectural question. Each one meets a place where a live undertaking changes and the governing authority has to survive the change.

Session state changes. A participant is admitted. An accommodation is required. An agent invokes a tool or spawns another agent. Work hands off. A device swaps. Compute moves to another provider or another country. Provenance has to be established. Jurisdiction shifts. Trust degrades or has to be re-established.

Different machinery at every one. Different failure points. One question.

Did the governing authority survive the change?

Those crossings are not exotic. They are how modern systems already work, which is why the question is already being asked in ten different vocabularies by people who do not know they are asking the same thing.

And the crossings multiply from here.

Agentic AI turns a single instruction into thousands of delegated acts. The Internet of Things puts consequential action in devices with no operator present and no screen to approve anything. The Internet of Everything connects those devices across operators, jurisdictions, and networks nobody jointly controls. Quantum changes many of the public-key cryptographic assumptions underneath today's credentials, certificates, signatures, and identity systems, which means even the artifacts we currently use to carry authority have a migration horizon.

Each of those makes the same requirement harder and more necessary at the same time. More acts between intent and effect. Fewer humans standing at the seams. More boundaries crossed per undertaking. And in every case the question does not change.

Which is why I have never treated AI as the primitive. AI is an important new participant, and it is the forcing function that finally makes the old architecture impossible to ignore. The primitive underneath is authority, and it was going to be needed with or without AI, because the machinery was always going to keep multiplying.

## There is still only one vote

Identity tells you which actor showed up. Credentials let it authenticate. Permissions tell a system what it can touch. Trust assigns it risk. Policy

constrains its local behavior. Agents perform the work. Logs say what happened.

Each of those answers something real, and several of them answer questions about legitimacy. None of them, standing alone, establishes that the authority governing the undertaking remained controlling across every consequential change:

Who authorized this undertaking, and was that authority still governing when the act became real?

The human can delegate the vote. Constrain it. Assign it to an institution. Grant an agent enormous discretion inside it.

Autonomy does not make the agent sovereign.

So the architecture is simpler than the vocabulary around it.

Authority originates outside the agent. A person, a company, a board, an institution acting under a statute. The undertaking carries it. The agents participate. The authority has to survive the machinery.

AI will change. The machinery will change.

The agent is still an agent.