Tokens and Authority

When inference moves onto the phone bill, the model becomes a session participant.

THOMAS ROCHA III
MAY 22, 2026

Chinese carriers are starting to sell AI tokens the way they once sold voice minutes, text messages, and gigabytes. China Mobile, China Telecom, and China Unicom are rolling out tiered subscriber plans that meter not bandwidth but inference. Ten million tokens a month at the consumer tier. Two hundred and fifty million at the enterprise tier. Bundled connectivity, security, API access, cloud PC, multi-agent routing, and model ecosystem entitlements packaged with the plan. The 1990s were voice minutes. The 2000s were SMS. The 2010s were

megabytes and gigabytes. The 2026 plan is intelligence, billed by the token.

Most analysts will read this as a pricing story. A new metering unit. A way for carriers to climb the value stack now that data is commoditized. A reasonable response to compute scarcity. All of those readings are true. None of them is the story.

The story is that inference has crossed from application usage into subscriber entitlement. Once that happens, the model is no longer merely called. It is admitted.

The unit that changed

Bandwidth was transport. A gigabyte is a quantity of data that moved through a pipe. It has no opinion about who sent it, who received it, what it contained, or what was done with it. The carrier's job, historically, was to deliver the data and bill for delivery. The metering unit and the governance unit were the same: the pipe.

Tokens are not transport. A token is a unit of inference. It represents a participant act inside the interaction the subscriber is having. It carries a who, a what, a where, a why, and a downstream consequence. It is not the result of moving bits across a wire. It is the result of a model entering the interaction, reading something, producing something, and leaving a state change behind.

The metering unit is the same word as before, but the governance unit is not the pipe anymore. The governance unit is the act of participation.

That distinction is the entire essay. The carrier is no longer selling transport. The carrier is selling access to model participation inside the subscriber relationship, with everything that follows when that participation enters a live interaction.

Token is not one thing

The vocabulary of the field has collapsed several different functions under one word, the way it collapsed storage, retention, memory, and focus under the word *memory*. The collapse hides the architecture. Pulling the word apart is the first move.

Token as model unit. The actual computational quantity the LLM consumes or produces. This is what the GPU accounts for. It is a property of the inference operation.

Token as billing unit. The thing the subscriber pays for. This is what the carrier counts. It is a property of the subscriber relationship.

Token as entitlement unit. The thing the plan allows or denies. A family plan with a shared cap. An enterprise plan with role-specific limits. A school plan with content restrictions. A vehicle plan with model-specific permissions. This is a property of the policy layer.

Token as authority event. The moment a non-human participant is admitted into a live interaction. This is a property of the session, and it is the function nobody is currently building.

Four different things. One word. The carrier marketing material calls all four *tokens* and lets the reader assume they are interchangeable. They are not. The first is engineering. The second is accounting. The third is policy. The fourth is governance, and the fourth is what every other layer assumes is already handled when it is not.

The pre-inference layer

Bandwidth billing was simple because the network only had to know that data moved and how much. The metering question was a quantity question. The governance question was downstream.

Tokenized inference is different because the network has to know several things before the model can answer. Identity: which subscriber invoked this. Entitlement: which plan authorizes this invocation.

Authority: who has the right to authorize this model for this purpose at this moment. Data permission: which data may be used as context. Compute placement: which jurisdiction may the inference run in. Model admission: is this specific model allowed inside this specific interaction. Billing assignment: which account is responsible for the cost. Policy reconciliation: which of the overlapping rules (carrier, plan, enterprise, family, jurisdiction, model provider, regulator) governs the conflict if one occurs. Audit binding: which evidentiary thread captures the decision so it can be reviewed later.

Every one of those is a pre-inference decision. All of them must resolve correctly before the model produces a single output token. If any of them resolves under the wrong authority, the resulting output is contaminated by the time it returns.

The visible product is the output token. The invisible cost is the coordination that had to occur before any output existed. The carrier sells the visible unit. The system consumes the invisible coordination.

The first cost of tokenized AI is not inference. It is proving that inference is allowed.

The participant problem

Most current architectures treat the LLM as infrastructure. The model is like the database, or the cache, or the API gateway. Something the application calls. Something invisible to the user. Something the system orchestrates.

That framing was always wrong. It becomes operationally wrong the moment the model is metered as a participant in a subscriber relationship.

A participant in an interaction can observe. It can transform. It can summarize. It can remember. It can recommend. It can act. It can trigger downstream effects. It can persist state. It can return later. It can be

invoked again with the history of what it did the first time. None of that is what infrastructure does. All of it is what participants do.

The LLM is not the session. It is a participant admitted into the session. Admission is an authority act. Treating admission as if it were infrastructure provisioning is what produces the failures the field has been documenting all year. The Cursor agent that deleted a production database was admitted to a session that gave it production-grade authority because the system had no place to evaluate the admission as an authority act. It was provisioned. It was not admitted.

Tokenized inference forces the question into the open. When the carrier bills the token, it acknowledges that something was participating. The act of billing is the public form of the admission. The architectural question is whether the admission was governed.

Not every inference event carries the same governance burden. A consumer asking for a dinner recipe does not create the same authority problem as an agent using private context, invoking tools, processing regulated data, routing through partner compute, or acting inside an enterprise account. The architectural pressure increases as inference becomes contextual, privileged, persistent, action-capable, cross-border, or multi-party. The token plan matters because it creates the commercial substrate on which all of those higher-risk uses will ride.

Agents make the token problem recursive

A model produces tokens. An agent consumes tokens in pursuit of an objective.

That difference matters, and the carrier plans are not yet drawn for it.

A model invocation is usually bounded by a prompt and a response. An agent invocation is bounded by a task, and a task may require planning, tool calls, retries, model selection, memory reads, memory writes, API calls, delegation to other agents, and downstream state changes. The

token meter may count the model's output. The system now has to govern the agent's entire path through the interaction.

That makes tokenized inference recursive. The first authorization is not enough. Each agent step may create another pre-inference decision: which model may be used for this sub-task, which tool may be called, which data may be read, which account pays, which jurisdiction applies, which log receives the event, which output may re-enter the session, and whether the agent is still operating inside the authority originally granted at the start of the chain.

An agent is therefore not just a participant. It is a participant that can generate additional participation.

That is where the billing model starts to strain. A subscriber may authorize an AI assistant to *handle this*. The assistant may then call a translation model, a summarization model, a calendar API, a payment tool, a customer record, a routing engine, and a second agent. Each step consumes tokens or triggers token-consuming work. Each step appears locally valid. The question is whether the session still has authority over the chain. The token meter cannot answer that. It can only show the meter advanced.

A multi-agent system makes the problem worse by an order of magnitude. Token cascades cross participants, tools, and authority domains in patterns no upstream actor planned. Agent A delegates to Agent B, which calls a third-party model, which routes to a tool, which writes to a memory shared with Agent C, which then acts on what was written. The original subscriber granted authority at the start. The fifth or sixth step is operating somewhere downstream of that grant, on data that did not exist when the grant was made, against a policy stack that has not been reconciled.

A model spends tokens to answer. An agent spends authority to act. The carrier meter counts the first. Nothing yet counts the second.

The model creates tokens. The agent creates token liability.

In a model world, the question is: was this inference allowed? In an agent world, the question becomes: was this chain of inferences, tool calls, state changes, and delegated actions still operating under the authority originally granted? That is not a metering problem. It is a session-governance problem, and the carrier plans currently being marketed do not have a place to evaluate it.

The failure domains in play

Once inference becomes a billable participant event inside a carrier subscriber relationship, the architecture is asked to coordinate several failure domains simultaneously. Each of these is a well-known site of industry failure on its own. The carrier-as-AI-distributor pattern activates them at once.

Authority ambiguity. Who authorized the model? The end user? The device? The carrier? The application? The enterprise IT policy? The family plan owner? The model provider's terms of service? When inference happens inside an enterprise account on a personal device on a corporate plan calling a third-party model under a regional regulation, the authority is fragmented across at least six actors, none of whom is currently structurally responsible for resolving the fragmentation. The default behavior is reconciliation after the fact, which is reconciliation against an interaction that has already changed state.

Billing-authority mismatch. The model may be invoked by one actor, billed to another, routed by a third, governed by a fourth, and held accountable by a fifth. A toll road cannot charge a vehicle coherently unless the system knows who entered, where they entered, which account applies, which rules govern the trip, and which jurisdiction is responsible if the rules conflict. Tokenized inference has the same structure. It cannot be billed coherently unless the system knows who invoked the model, under what authority, against what entitlement, in what jurisdiction, with what data, for what purpose.

Model admission failure. Most current systems do not distinguish between *the model is available* and *the model is admitted into this specific interaction for this specific participant with this specific authority*. The first is a capability statement. The second is an authority statement. The carrier billing for tokens conflates the two because, once the meter starts, the model is, by definition, participating. Whether the model should be participating in this session, at this moment, under this policy, is the question the meter does not ask.

Policy fragmentation. Consumer plan, family plan, enterprise plan, jurisdictional rule, model provider rule, carrier rule, device rule, and application rule can all disagree with one another in any given session. None of them is currently authoritative over the others. The interaction proceeds under whichever rule was checked last, or whichever rule the model decides to weight, which means the rule that governs is not the rule that should govern.

Compute placement and residency. Once a token can route through carrier edge compute, regional cloud, private cloud, model partner infrastructure, or third-party GPU capacity, the placement decision is not just an optimization. It is a policy act. *Where can this compute run* is a sovereignty question, a privacy question, a contract question, an audit question, and a liability question. The placement decision arrives before the inference does, and it must be resolved against the session's policy stack, not the network's load balancer.

Audit discontinuity. Token consumption, model selection, prompt context, output return, billing event, and policy decision are typically logged to different systems run by different parties. There is no single evidentiary thread. The carrier can prove that tokens were used. The model provider can prove the model was invoked. The enterprise can prove the user was authenticated. None of them can prove, jointly, why this specific inference was admitted under what authority. The forensic gap that is created is not a software bug. It is a structural property of how the participants in this market are configured to log.

Concurrency. When a household's family-plan subscribers invoke models simultaneously across phones, tablets, vehicles, and home devices, the policy reconciliation between subscribers, devices, and the family-plan owner has to converge faster than the rate of new invocations. At low traffic, the reconciliation is invisible. At the carrier scale, it stops converging, and the policy that governs the next invocation is whichever rule the system last had time to evaluate, not whichever rule should have applied.

The efficiency paradox. This is the brutal one. AI was introduced to reduce friction. Tokenized AI creates a pre-inference coordination layer that consumes infrastructure before any output exists. Authority resolution, entitlement check, model admission, compute placement, policy reconciliation, billing assignment, and audit binding: all of these run before the model answers. The token meter starts after this work has already happened. The system spends on coordination, then sells inference. The marketing claims efficiency. The infrastructure absorbs the cost of producing the efficiency. Tokenized inference does not reduce coordination pressure. It monetizes the event that creates it.

Each of these is a known failure site. The carrier-as-AI-distributor pattern is one of the first mass-market deployments likely to activate all seven at once.

What the meter reveals

The carriers are building pricing and metering structures on top of existing model APIs.

That is the visible layer.

The architectural observation is different. Once intelligence is sold as a subscriber-session product, the architecture required to deliver it cleanly is not the architecture currently deployed anywhere. The carriers are not implementing the missing layer. They are creating the commercial pressure that makes the missing layer's absence visible.

The carrier is not necessarily the right owner of this layer. That is not the point. The point is that once inference spans the carrier account, device context, application state, model provider, compute location, billing ledger, and policy obligations, no single participant can govern the entire event from within its own stack. The required layer is not carrier governance. It is session governance across participants.

The convergence is interesting because the absence is universal. Every party in the value chain is treating model admission as if it were already governed. The carrier assumes the model provider handles it. The model provider assumes the application handles it. The application assumes the carrier handles it. The user assumes the network handles it.

The system proceeds on the assumption that someone, somewhere, evaluated whether this specific inference was admissible inside this specific live interaction at this specific moment. Nobody did. The meter ran anyway.

That is what failure mode looks like when it has been monetized.

Tokenized inference puts carriers on a path where account metering eventually must become authority governance. They are not there yet. The meter points there.

The boundary

The visible event is that Chinese telcos are selling AI tokens. The deeper event is that inference has become a billable participant event without a session-native authority layer to evaluate the participation.

The boundary that has been crossed is not technical. It is architectural. As long as inference occurred within applications, the application could pretend to own the governance question. The model was a feature. The vendor handled it. The user accepted the application’s framing as authoritative.

Once inference is metered inside the subscriber relationship, that framing no longer holds. The carrier is not inside the application. The model is not inside the carrier. The session is happening across all of them, with policy obligations attaching to each at different points and to none of them continuously. The interaction is the place where authority would have to live to govern any of this coherently. The interaction is the layer nobody is building.

A toll authority cannot run a toll road by assuming each driver self-reports. A bank cannot clear settlements by assuming each counterparty self-attests. A power grid cannot remain synchronized if each generator decides locally what frequency to produce. In every case where capability has consequences at scale, the system requires an authority structure that exists outside the participants and refuses operations that would let the system slip into incoherence, regardless of what any individual participant decides.

Tokenized inference is now in that category. The carrier acknowledged it the moment the meter started counting participant acts instead of bytes. The architecture for governing what the meter is counting is not yet built. The market will discover the absence the way it always discovers architectural absences: through the failures that accumulate while the substrate is still convinced it is selling a feature.

The first token is not generated by the model. It is spent proving the model is allowed to participate. Until something exists that can prove it at the session boundary, each token is, at best, an account-authorized participant act. The system may know who paid. It may know which model answered. It may know how many tokens were consumed. What it cannot prove is whether that inference was admissible inside that live interaction, under that authority, at that moment.

The model does not govern the session. The session governs the model.

That is what the carriers have not yet figured out they are selling.