Nick Clark · Published July 29, 2026 · PDF

WorkOS: Agent Identity Infrastructure and Source Metering

An agent that meters refusals per assertion can be pushed into withholding by anyone willing to present more identities. Enterprise identity infrastructure, WorkOS among the platforms publicly serving that category, addresses the issuance and authentication side of this problem: who an actor is, which directory they belong to, and what an application should trust about them. U.S. Provisional Application No. 64/117,812 discloses a mechanism sitting downstream of that question. In Chapter 2, a semantic agent derives origin-equivalence classes from its own append-only lineage field and its own counterparty identity records, then charges its refusal counter per class rather than per artifact. The class is strictly local, is not a third-party-attested identity, and confers no portable standing.

One Origin, Many Faces

An autonomous agent in an open network receives claims about its own conduct from parties it has never transacted with. Some are false. Some cannot be resolved against the record the agent holds. Both outcomes are refusals, and if the agent counts refusals one per claim, the cost of pushing it into a withholding posture is the cost of presenting more identities.

The filed specification names this condition directly. Merit-independent metering accumulates increments upon both refusal paths: artifacts that are false produce the rejected determination (124), and artifacts unresolvable against the append-only lineage field (104) produce the not-determinable determination (126). Volume alone therefore drives the authorization gate (300) of the semantic agent (100) to the withheld state (310), and the volume available to an adverse party is limited only by the cost of presenting further identities.

Note what the agent gets right and what it gets wrong under that arrangement. It is not wrong about any individual artifact; each determination stands on its own path. It is wrong about how many independent sources it is hearing from, because the count it keeps is a count of artifacts. Two layers can respond to that gap: one constrains which identities can be presented credibly at all, the other changes what a count of refusals is a count of. Identity infrastructure and the disclosed mechanism sit on opposite sides of that line.

Where Identity Platforms Sit in the Stack

WorkOS is, as publicly described, a developer platform for the identity and authentication features that enterprise buyers expect from the software they adopt. The public description of its purpose is to let an application team add enterprise identity capability without building each protocol integration from scratch, and the capabilities named in those materials include single sign-on against a customer's own identity provider, directory synchronization so that user and group records stay current between a customer's system of record and the application, audit logging, and administrative surfaces through which a customer's own staff configure their tenant.

That work is substantial, and it is aimed at a different question than the one the filing addresses. An identity platform of this kind answers questions of the form: is this actor who it claims to be, does it belong to a tenant the application recognizes, and was the

directory's assertion about it recorded where an auditor can inspect it. Those answers are portable by design. The point of an attested identity is that a second relying party can accept it without re-deriving it.

Extending that model to non-human actors raises questions of the same shape: how an agent obtains credentials, how those credentials are scoped, how an agent acts on behalf of a principal without becoming indistinguishable from it, and how the activity is attributable after the fact. For what any given product does today, the vendor's own documentation governs. Everything below describes what the disclosed architecture requires, leaving the comparison to the reader.

Charging the Counter Per Source

Chapter 2 of U.S. Provisional Application No. 64/117,812 discloses origin-equivalence normalization. The refusal counter (304) of the semantic agent (100) is incremented per origin-equivalence class (200) rather than per conduct evaluation artifact (116). The filing describes this as charging the counter per source of assertion rather than per assertion, a source being a set of asserting parties (118) between which the agent demonstrates a relation from its own records. No registry, directory, or shared scoring service is consulted.

The derivation is an ordered procedure. On receipt of an artifact, the agent retrieves the counterparty identity record (114) of the asserting party (118), or instantiates one where none exists. It then retrieves from the signed policy object (112) in force an enumeration of declared relation types, each specifying a class of lineage field (104) entry and a matching condition over such entries. For each asserting party already assigned to a class, in the current window or a preceding one, the agent evaluates each declared relation type against the present party.

The filing names three such relation types. A **shared dispatch lineage (202)** is evidenced where the lineage field contains an entry recording a dispatch to the present party and an entry recording a dispatch to the compared party, and both record a common parent dispatch entry as their immediate antecedent. A **co-signature (204)** is evidenced where a single lineage entry bears a signature verifiable against an identity primitive of the present party and a signature verifiable against an identity primitive of the compared party. A **common introduction path (206)** is evidenced where the counterparty identity records of both parties record an introducing party and those introducing parties are identical. Each declared relation type is independently sufficient.

Where a relation is evidenced, the present party joins the class of the compared party; where more than one class is identified, the classes merge; where nothing is evidenced, a new class is formed. The assignment, the relation types evaluated, the entries relied upon, and the class identifier are appended to the lineage field, and the identifier is written into the counterparty identity record of each party in the class. The filing is explicit that the identifier records class membership and is not a determination concerning conduct.

The class is **local**. The evaluation runs on the agent's own lineage entries and its own counterparty identity records, without reference to a centralized registry, without query to a directory, without participation in a consensus procedure, and without coordination with a further execution node. A second semantic agent holding different records derives, from the same population, a partition that need not agree. No procedure reconciles the two, no class identifier is transmitted between them, and neither agent admits a class identifier derived by the other. The identifier is not an identity attested by a third party and confers no portable standing.

The class is **persistent while the meter is not**. Assignment survives elapse of the window, and a party assigned in one window is a compared party in each succeeding window without re-derivation. Only the per-class increment register is window-scoped.

Normalization **governs metering alone**. An artifact from a party in a class already recorded in the register still produces a determination and is still appended to the lineage field; it applies no further increment. The filing states that normalization suppresses no determination and withholds no adjudication.

Chapter 2 also discloses a cost multiplier for classes the agent has not yet tested. Under the severance-survival test, each constituent edge resolves to a reason-type against recorded severance events of a payment class and an obligation class, survived or not within a declared continuation interval. Where every constituent edge of a class resolves not-typeable, the class is designated an untested class. The reduced contribution is conditioned: the multiplier declared in the signed policy object (112) applies where the class is designated untested **and** its recorded introduction paths converge upon a common ancestor entry within a depth declared in that policy object. Otherwise the contribution is applied as a full increment. The multiplier is greater than zero and less than unity, and its value, like the rate threshold, the continuation interval, and the depth, is left to policy; the filing fixes no operative value, the figures in its worked trace being illustrative.

Two Different Questions, One Pipeline

Identity infrastructure produces assertions intended to travel. Its value comes from portability: an attestation issued once is accepted by many relying parties, which is why the category invests in protocols, directories, and audit trails. The origin-equivalence class travels nowhere. It is derived by one agent from that agent's records, and no other agent admits it. The filing states that affirmatively, and draws the same line against a Sybil-resistance scheme that resolves identity against a shared registry or a coordinating authority.

Convergence sits at the level of the concern. Both layers respond to the fact that an actor in an open network can present more than one face. Credential binding governs which faces can be presented credibly at all. Origin-equivalence normalization takes

whatever faces arrive and bounds the contribution of a single origin to withholding, irrespective of how many identities that origin presents.

Read structurally, the positioning is complementary rather than competitive. Credential binding narrows the population of unrelated-looking parties that can reach an application; locally derived source metering bounds the contribution of whatever population does reach the agent, including parties whose credentials are entirely legitimate.

Running Both Layers, and Where This One Stops

Consider a deployment that runs an identity layer ahead of a semantic agent. Whatever that layer resolves about a caller, the disclosed derivation does not consume it: class derivation is specified over the agent's own dispatch lineage entries, over signatures verifiable against identity primitives, and over introducing parties recorded in counterparty identity records the agent itself holds. An external attestation is not among its inputs, so the two layers run in series without either depending on the other's output.

The boundaries of the disclosed architecture are worth stating plainly. Authentication is outside it: the procedure specifies no credential check, and a party the agent treats as an asserting party is simply one from which an artifact arrived. Reputation is outside it by construction, the class identifier being unattested and conferring no portable standing. Adjudication is untouched; an artifact from a class already in the register still produces a determination and is still appended to the lineage field. Genuinely independent origins remain separately costly, because where no declared relation type is evidenced between parties, each is assigned to a distinct class. And policy still matters: the filing notes that an empty enumeration of relation types assigns every asserting party to a distinct class, restoring the very condition the chapter addresses.

Teams evaluating agent trust architecture do well to keep the two questions apart. Who is this actor, and how many independent sources am I hearing from, are not the same question, and no single layer answers both.

Disclosure Scope

This article describes subject matter disclosed in U.S. Provisional Application No. 64/117,812, Chapter 2, origin-equivalence normalization. The application is pending. Reference numerals correspond to elements as bound in that filing. Quantities described as policy-declared, including the rate threshold, the cost multiplier, the continuation interval, and the introduction-path depth, are declared in the signed policy object (112) and are given no fixed operative value by the filing, values appearing in its worked trace being illustrative.

References to WorkOS are to public materials and are used for comparison only; no relationship, endorsement, or infringement is asserted. Nothing here characterizes any third-party product as practicing, or as failing to practice, any claim, and nothing here asserts that any license is required. Readers should consult vendor documentation for the current capabilities of any product named.

Origin-Equivalence Normalization (/origin-equivalence)

All 49 steps → (/inventive-steps)

Metering that can't be gamed by splitting one accuser into many.

Provisional application

PRIMARY TECHNICAL DISCLOSURE

- Origin-Equivalence Normalization: Metering Refusal Per Source, Not Per Assertion (/articles/origin-equivalence/origin-equivalence-normalization)

SECONDARY TECHNICAL

- Content-Blind Encounter Commitment and the Paired Encounter Receipt (/articles/origin-equivalence/content-blind-encounter-commitment)
- Cross-Receipt Equivocation Detection: Catching an Agent That Tells Two Stories (/articles/origin-equivalence/cross-receipt-equivocation-detection)
- The Untested-Origin Saturation Bound: Pricing a Flood of Freshly Minted Identities (/articles/origin-equivalence/untested-origin-weighted-saturation-bound)
- Counterparty-Side Fork Detection: Recognizing a Split Identity From Outside (/articles/origin-equivalence/counterparty-side-fork-detection)
- The Attestation Origin-Equivalence Gate: An Anti-Collusion Bound (/articles/origin-equivalence/attestation-anti-collusion-bound)
- The Bridging Party and the Deferred Merge: Charging One Windowed Increment Against Every Origin-Equivalence Class a Single Asserting Party Bridges (/articles/origin-equivalence/bridging-party-deferred-merge)
- The Common-Execution-Node Relation Type: Charging Co-Hosted Asserting Parties as One Origin (/articles/origin-equivalence/common-execution-node-relation-type)
- Receiver-Side First-Encounter Budget Decrement: Pricing a Stranger's Arrival Per Presented-Material Origin Class (/articles/origin-equivalence/receiver-side-first-encounter-budget-decrement)
- Mutual First Encounter Without Cross-Budget Coordination: Two Strangers, Two Matched Pairs, Two Separate Budget Decrements (/articles/origin-equivalence/mutual-first-encounter-no-cross-budget)

APPLICATIONS · GENERAL

- Sybil-Resistant Agent Networks: When One Accuser Becomes Many (/articles/origin-equivalence/sybil-resistant-agent-networks)
- Fraud Rings in Autonomous Commerce: Metering Per Source, Not Per Claim (/articles/origin-equivalence/fraud-rings-in-autonomous-commerce)

APPLICATIONS · SPECIFIC

- W3C Decentralized Identifiers and Verifiable Credentials: Verifying an Identifier Versus Metering Its Origin (/articles/origin-equivalence/w3c-decentralized-identifiers)
- Okta Auth for GenAI: Agent Identity and Origin Equivalence (/articles/origin-equivalence/okta-auth-for-genai)
- **WorkOS: Agent Identity Infrastructure and Source Metering (/articles/origin-equivalence/workos-agent-identity)**
- Entra Agent ID: Directory Identity and Derived Origin Classes (/articles/origin-equivalence/microsoft-entra-agent-id)

- SPIFFE and SPIRE: Workload Identity and Origin Equivalence (/articles/origin-equivalence/spiffe-spire)
- Cloudflare Bot Management: Blocking Agents and Metering Sources (/articles/origin-equivalence/cloudflare-bot-management)

TERMINOLOGY

- asserting party (/glossary#asserting-party)
- origin-equivalence class (/glossary#origin-equivalence-class)

Origin-Equivalence Normalization overview → (/origin-equivalence)