Nick Clark · Published April 25, 2026 · PDF

On-Device Agent Identity for Robots and Autonomous Vehicles: An Auditable Substrate for Embodied Physical-World Agents

Robots and autonomous vehicles need a perception-to-actuation loop that stays local, keeps working when the network drops, and produces an auditable record of who the machine met and what it did. This application is built on the Agent-Resident Execution Substrate, disclosed in U.S. Provisional Application No. 64/070,239, which hosts a persistent, hardware-anchored agent identity on the device itself, dispatches sensor inputs to governed on-device inference endpoints, and records every perception and actuation event in an append-only lineage.

What This Application Specifies

An embodied machine is not a chatbot with wheels. A delivery robot, a warehouse mover, or a passenger vehicle runs a continuous loop: sensors observe the world, models interpret what they see, and effectors act on that interpretation with physical consequences. That loop must run whether or not a cell tower is in range, and every decision it makes may later need to be reconstructed by an operator, an insurer, or a safety regulator.

The Agent-Resident Execution Substrate, disclosed in U.S. Provisional Application No. 64/070,239, applies directly to this setting. The specification names robotic and vehicular computing as first-class deployment configurations and describes a substrate that is operable in configurations involving physical-world perception and actuation, including robotic, vehicular, and embodied computing contexts.

In these deployments the pieces map cleanly onto the machine. A persistent semantic agent lives on the vehicle or robot as its execution substrate, holding a persistent identity field, a cognitive state field, an append-only lineage field, and a governance policy field. Perception primitives feed the substrate: the specification describes sensor inputs from perception primitives being received by the substrate as inference requests dispatched through the agent-to-tool dispatcher. Inference runs on managed inference endpoints held in a local tool registry. The results drive motion: inference outputs are supplied to actuation primitives as governed side effects under the substrate's governance policy field, with perception events and actuation events recorded in the lineage field. Effector output adapters are described as converting inference outputs into governed side effects, expressly including physical actuation in robotic and vehicular deployments.

Why It Matters

Three problems recur in embodied autonomy, and the substrate addresses each.

First, connectivity is not guaranteed. A robot in a basement or a vehicle in a tunnel cannot wait on a round trip to a datacenter to decide whether to brake. The substrate is built for a device with bounded local memory, storage, and compute, running one or more model artifacts of bounded size on local processors, so the perception-to-actuation loop stays resident on the machine.

Second, embodied decisions carry liability. When a machine acts in the physical world, someone eventually asks why. The substrate's lineage field is an append-only sequence of records structured so that the complete operational history of the agent is deterministically reconstructible, with each record verifiable against its predecessor under a continuity proof. Perception events, dispatch events, and actuation events all land in that record, giving an operator a tamper-evident account of what the machine sensed and what it did.

Third, a fleet vehicle is expected to last through many hardware and software revisions without becoming a different machine each time. The substrate's continuity guarantee keeps the agent's identity, cognitive state, and lineage intact across replacement, retraining, or removal of any subordinate model, and across governed updates to the substrate runtime itself. A perception model can be swapped for a better one, and the vehicle's accumulated operational identity carries forward unbroken.

How It Composes With the Domain

Start with perception routing. Sensor inputs arrive through input adapters, which the specification lists as covering image and video, motion and orientation, location, and ambient sensor input, among others. Each adapter declares a governance scope and records each input as a deterministic event in the lineage field. The agent-to-tool dispatcher then routes each request to a managed inference endpoint whose per-endpoint capability declaration matches the input modality and task category. Because multiple specialized endpoints can be co-resident, a machine can carry a compact obstacle classifier, a separate pedestrian-detection endpoint, and a natural-language endpoint for operator interaction, each independently managed and each with its own retraining schedule and governance scope.

Context follows the environment. The substrate lets the active operating context be selected by sensor input: the specification describes evaluating sensor inputs received through perception adapters against a scope-selection rule to update the active-scope

indicator, so the agent's operational context adapts to the device's physical environment. It gives concrete triggers, including a location-based scope selecting a home, work, or roaming context and a motion-based scope selecting a stationary or in-motion context. A vehicle can carry a stricter policy scope while in motion and a more permissive one while parked and charging, with the switch driven by the machine's own sensors rather than a manual toggle.

Encounters become records. An autonomous machine meets other actors constantly, and the substrate maintains counterparty identity records for them. The specification anticipates an autonomous vehicle that encounters other vehicles and pedestrians during operation, and a robotic device that encounters human users in the environments where it operates. Most such encounters are fleeting, so a counterparty record can be ephemeral, retained only for the duration of an active encounter, then archived under retention policy. A record earns persistence only through a promotion policy, triggered by explicit authorization or by accumulated interaction frequency or duration above a policy-declared threshold. A one-time pedestrian crossing does not become a permanent profile; a recurring loading-dock worker can, under policy.

Actuation is gated. Effector output adapters carry per-adapter safety envelopes restricting the magnitude or category of side effects permitted in a given scope, and every output is recorded as a deterministic output event. Resource governance is tuned for the physical machine: budgets are configured to respect thermal, power, real-time-response, and safety-critical constraints, including foreground prioritization of perception-driven inference over background retraining operations. Learning happens in the background and yields to the sensing loop.

When local capability is not enough, the machine can reach out under strict control. The cloud-burst forwarding subsystem selectively forwards inference requests to a remote endpoint when local endpoints lack capability, capacity, or both, but only after an admissibility test covering capability, capacity within a latency budget, disclosure admissibility, and cost. Critically for embodied use, a deferred forwarding mode queues

requests for a later connectivity event while the agent operates in a degraded mode and returns partial or surrogate responses, so a disconnected machine keeps functioning rather than stalling.

What This Enables

A concrete picture: a fleet of autonomous yard trucks at a logistics site. Each truck runs the substrate locally. Perception adapters feed camera and motion input to on-device endpoints that detect obstacles and workers; the dispatcher routes by modality; effector adapters drive steering and braking under safety envelopes. Because the identity is hardware-anchored to a secure element on each truck, a truck's agent identity is bound to that physical machine and cannot be silently transplanted. When the site upgrades its obstacle-detection model, the substitution runs as a governed lifecycle operation and each truck's accumulated lineage survives the swap.

Fleet coordination without a central brain follows from the federation layer, which coordinates devices through lineage exchange and does not require exchange of model artifacts. Outcome signals observed on one truck can inform another's routing and training-corpus assembly under federation policy, and a federated agent identity record can treat a single operator's devices as one identity across hardware refreshes. Because model weights need not leave each machine, a truck that handled a difficult loading-dock geometry well can share the outcome signal, not its raw sensor data.

Auditability becomes a product feature rather than an afterthought. The privacy invariant holds that lineage records, model artifacts, and counterparty records are not transmitted off the device except under an explicit disclosure policy object, and each off-device disclosure is itself a recorded, verifiable lineage event that a user or a regulatory authority can audit. For an operator answering an incident review, the machine can produce a continuity-proofed account of what it perceived, which endpoint it dispatched to, what it actuated, and every time any of that left the device.

Boundary Conditions

This substrate is an execution and governance architecture, not a perception stack or a control system. It does not supply the sensor-fusion algorithms, the object detectors, or the motion planners; it hosts, dispatches to, and records around them. The specification points to separate perception and actuation primitives for the sensing and effector layer and integrates with them through adapters.

Nothing here relaxes physical safety engineering. The substrate meters thermal, power, and real-time constraints and enforces safety envelopes on effector outputs, but functional-safety validation, redundancy, and certification of an actual robot or vehicle remain the responsibility of the system builder and are outside what the application specifies. On-device inference is bounded by the device's real compute envelope, which is why cloud-burst forwarding and quantized endpoints exist; the largest models may still need to burst off-device under policy, and the specification makes no representation about latency or accuracy figures for any particular machine. Federation and cloud-burst both cross the device boundary and are therefore constrained by the disclosure policy and, where connectivity is intermittent, by the degraded-mode behavior described above.

Disclosure Scope

The technical architecture described here, including the persistent agent identity, the managed inference tool registry, the agent-to-tool dispatcher, the append-only lineage field, the governance policy field, counterparty identity records, the privacy invariant, cloud-burst forwarding, and federation, is disclosed in U.S. Provisional Application No. 64/070,239. All statements about what the invention does trace to that disclosure, which expressly names robotic and vehicular computing as deployment configurations and describes physical-world perception and actuation through input and effector adapters. The robotics, autonomous-vehicle, and fleet-logistics framing in this article, along with any references to operational, insurance, or regulatory review, is external

domain context provided to illustrate an enabling implementation; it is not part of the disclosed invention and does not represent any specific regulatory determination, safety certification, or performance benchmark. Functional-safety standards, transportation regulations, and the sensing and control technologies of any particular embodied system are independent of this application and are cited, where mentioned, only as external context.

Agent-Resident Execution Substrate (/agent-resident-execution-substrate)

All 40 steps → (/inventive-steps)

Persistent execution environment carried by the agent, not the host — identity, state, and lineage across power cycles, devices, and upgrades.

Provisional application

PRIMARY TECHNICAL DISCLOSURE

- Agent-Resident Execution Substrate, Articles (/articles/agent-resident-execution-substrate)

SECONDARY TECHNICAL

- Persistent Semantic Agent (/articles/agent-resident-execution-substrate/persistent-semantic-agent)
- Managed Inference Tool Registry (/articles/agent-resident-execution-substrate/managed-inference-tool-registry)
- Agent-to-Tool Dispatcher (/articles/agent-resident-execution-substrate/agent-to-tool-dispatcher)
- Lineage-Derived Training Signal (/articles/agent-resident-execution-substrate/lineage-derived-training-signal)
- Identity Preservation Across Upgrades (/articles/agent-resident-execution-substrate/identity-preservation-across-upgrades)
- Cognitive State-Conditioned Dispatch (/articles/agent-resident-execution-substrate/cognitive-state-conditioned-dispatch)
- Governed Tool Lifecycle (/articles/agent-resident-execution-substrate/governed-tool-lifecycle)
- Continuity-Proof Lineage (/articles/agent-resident-execution-substrate/continuity-proof-lineage)

- Substrate Runtime Continuity (/articles/agent-resident-execution-substrate/substrate-runtime-continuity)
- Personal Corpus Model Training (/articles/agent-resident-execution-substrate/personal-corpus-model-training)
- Heterogeneous Inference Endpoints (/articles/agent-resident-execution-substrate/heterogeneous-inference-endpoints)
- Atomic Lifecycle Substitution (/articles/agent-resident-execution-substrate/atomic-lifecycle-substitution)
- Integrity Signal Feedback (/articles/agent-resident-execution-substrate/integrity-signal-feedback)
- Hardware-Bound Identity (/articles/agent-resident-execution-substrate/hardware-bound-identity)
- Cognitive State Append-Only Invariant (/articles/agent-resident-execution-substrate/cognitive-state-append-only-invariant)
- Counterparty Identity Records (/articles/agent-resident-execution-substrate/counterparty-identity-records)
- Privacy Egress-Controlled Disclosure (/articles/agent-resident-execution-substrate/privacy-egress-controlled-disclosure)
- Federated Cross-Device Agent Identity (/articles/agent-resident-execution-substrate/federated-cross-device-agent-identity)

APPLICATIONS · GENERAL

- Personal AI Agents That Survive Device Loss: One Continuous Identity and a Private Corpus Across Every Device (/articles/agent-resident-execution-substrate/personal-cross-device-agents)
- Enterprise Agent Fleets: Stable Agent Identity and Governed Tool Access Across Model Upgrades and Infrastructure Migration (/articles/agent-resident-execution-substrate/enterprise-agent-fleets)
- Audit-Grade Agent Identity for Regulated Finance and Healthcare: Continuity-Proof Lineage Across the Agent Lifecycle (/articles/agent-resident-execution-substrate/regulated-industry-agents)
- Edge and On-Device Agents: Hardware-Bound Identity Across Heterogeneous Inference Endpoints (/articles/agent-resident-execution-substrate/edge-and-on-device-agents)
- Agent-to-Agent Commerce With Counterparty Identity Records and Egress-Controlled Disclosure (/articles/agent-resident-execution-substrate/agent-to-agent-commerce)
- Governed Tool Lifecycles for Managed Inference-Provider Ecosystems: A Substrate Approach to Owning, Routing, and Retiring AI Tools (/articles/agent-resident-execution-substrate/managed-tool-ecosystems)
- Proving Unbroken Continuity in Long-Lived Autonomous Systems Across Substrate Migration and Atomic Model Substitution (/articles/agent-resident-execution-substrate/long-lived-autonomous-systems)

- Personal-Model Personalization: A User's Own Corpus-Internalized Model on the Agent-Resident Execution Substrate (/articles/agent-resident-execution-substrate/personal-model-personalization)
- **On-Device Agent Identity for Robots and Autonomous Vehicles: An Auditable Substrate for Embodied Physical-World Agents (/articles/agent-resident-execution-substrate/embodied-physical-world-agents)**

APPLICATIONS · SPECIFIC

- LangGraph Platform (LangChain) vs an agent-resident execution substrate: orchestration-graph state versus a portable, hardware-anchored agent runtime (/articles/agent-resident-execution-substrate/langgraph-platform)
- OpenAI AgentKit and the Assistants/Responses API vs agent-carried, hardware-anchored identity with governed tool lifecycle (/articles/agent-resident-execution-substrate/openai-agentkit)
- Microsoft Copilot Studio vs an agent-resident execution substrate: platform-hosted agent authoring versus portable, device-resident agent identity and continuity (/articles/agent-resident-execution-substrate/microsoft-copilot-studio)
- Google Vertex AI Agent Engine (managed runtime for deploying and scaling agents, with sessions/memory) vs an agent-carried, continuity-proofed identity substrate (/articles/agent-resident-execution-substrate/google-vertex-agent-engine)
- AWS Bedrock AgentCore (runtime, memory, identity, and gateway services for deploying agents at scale) vs an agent-resident execution substrate: where does the agent identity actually live? (/articles/agent-resident-execution-substrate/aws-bedrock-agentcore)
- Letta (formerly MemGPT) vs an append-only cognitive-state substrate: what a memory-management framework does not provide (/articles/agent-resident-execution-substrate/letta-memgpt)
- Cognition's Devin, an autonomous AI software-engineering agent vs a portable, continuity-proofed agent-resident runtime (/articles/agent-resident-execution-substrate/cognition-devin)
- Cloudflare Agents (Durable Objects) vs an agent-resident execution substrate: portable hardware-bound identity and continuity-proof lineage (/articles/agent-resident-execution-substrate/cloudflare-agents)
- Ollama alternative: from local model runner to a governed agent-resident substrate (/articles/agent-resident-execution-substrate/ollama)
- Apple Intelligence (on-device foundation models, Private Cloud Compute) vs a persistent agent-resident substrate: who owns identity, lineage, and the model? (/articles/agent-resident-execution-substrate/apple-intelligence)

Agent-Resident Execution Substrate overview → (/agent-resident-execution-substrate)