The Agent Singularity: A System-Level World Model for Headless Networks, Sovereign Compute, and Cognitive State

Jianhua Chen, Xingwei Qu
catherine.jianhua@gmail.com
Senior Product Manager
Manchester, UK

Abstract

As AI systems increasingly act on users' behalf in both digital and physical environments, they must satisfy hard real-time safety and data sovereignty constraints that are not addressed by existing cloud-centric application architectures. This paper studies the system-level conditions required for reliable agent execution under latency jitter, adversarial routing, and regulatory constraints.

We propose the *Agent Metabolic System*, a static 7-layer architecture that structurally separates perception, sovereignty, execution, and settlement, together with a dynamic risk-aware routing model that treats latency variance as a first-class decision variable. Rather than optimizing average performance, routing decisions are governed by a stochastic control barrier that enforces hard deadlines and sovereignty requirements.

Through discrete-event simulation under heavy-tailed network conditions ($N = 2000$), we show that cloud-centric and mean-based routing strategies fail to guarantee safety or sovereignty, while the proposed architecture achieves deterministic deadline compliance and zero context leakage. These results suggest that structural separation and risk-aware control are necessary ingredients for safe and sovereign agentic systems operating in cyber-physical environments.

Keywords

Agentic Workflow, Headless Web, Sovereign Compute, DePIN, Cyber-Physical Systems, Cognitive State Economy, GDPR by Design

1 Introduction

1.1 The Glass Wall Phenomenon

We use the term *Glass Wall* to describe two related but distinct failure modes that emerge as AI systems transition from passive assistants to autonomous agents.

Intent Glass Wall (Interface-Level). Human intent is continuous, embodied, and context-rich, yet current interaction paradigms mediate this intent through discrete, low-bandwidth graphical interfaces. This mismatch strips away situational context and limits the fidelity with which intent can be expressed.

Execution Glass Wall (System-Level). Even when intent is correctly inferred, existing cloud-centric architectures often fail to safely execute that intent. Stochastic network latency, tail jitter, incentive interference, and data centralization introduce risks that manifest as physical safety violations or context leakage once execution is committed.

While the first Glass Wall motivates richer perception and intent capture, this paper focuses on the second: the architectural inability of current systems to safely and sovereignly execute committed intent.

1.2 Architecture as Constraint

The constraints proposed in this paper, referred to as **System Invariants**, are not normative policy claims nor assertions of architectural inevitability. Rather, they formalize necessary constraints that any system aiming to guarantee hard real-time safety and data sovereignty must satisfy under adversarial network conditions.

1.3 Contributions

Our contributions are as follows:

- **System Modeling.** We formalize a static 7-layer agent architecture that enforces structural separation between perception, sovereignty, execution, and settlement.
- **Risk-Aware Control.** We introduce a stochastic routing model that treats latency variance as a first-class constraint via a control barrier formulation.
- **Failure Analysis.** We demonstrate, through simulation, that cloud-centric and mean-based routing strategies fail under heavy-tailed latency, even with improved average performance.
- **Empirical Validation.** We show that structural separation and risk-aware routing together enable deterministic safety and sovereignty guarantees.

2 Related Work and Adversarial Assumptions

This section situates our work within the broader landscape of agentic frameworks, distributed computing, and privacy regulations.

2.1 Application-Layer Agent Frameworks

Recent advancements in Large Language Models (LLMs) have spurred the development of application-layer orchestration frameworks such as **LangChain** [2], **AutoGPT** [8], and **BabyAGI**. These frameworks excel at chaining prompts and managing ephemeral context windows. However, they operate primarily as client-side wrappers dependent on centralized APIs. They lack a native understanding of physical topology or data sovereignty, treating all execution environments as agnostic "cloud endpoints." Our work addresses this gap by introducing **infrastructure awareness** directly into the agent's logic.

2.2 Mobile Edge Computing (MEC)

The concept of offloading computation to the network edge is well-established in telecommunications [1, 5]. Traditional research focuses on optimizing bandwidth and energy consumption for static applications. While relevant, these approaches do not account for the **semantic intent** of the user. Our proposed **Kinetic Routing Model** extends traditional MEC offloading by introducing "Cognitive State" and "Privacy Requirements" as primary routing variables alongside latency and energy.

2.3 Data Sovereignty

With the enforcement of **GDPR** [9] and the emergence of the **EU AI Act** [3], data sovereignty has moved from an ethical preference to a legal constraint. Existing solutions like **Federated Learning** [6] or Homomorphic Encryption focus on model training but often incur prohibitive latency penalties. Our architecture adopts a pragmatic **Trusted Execution Environment (TEE)** approach via Private Cloud Compute (PCC) and **local physical isolation**, balancing strict compliance with real-time performance.

2.4 Cyber-Physical Orchestration

In robotics and CPS, "Orchestration" typically refers to fleet management [4]. However, the "Headless Web" requires orchestrating heterogeneous digital services (APIs) alongside physical actuators. Our work proposes a unified interface (Section 5) to manage this dual-modality interaction.

2.5 Threat Model & Definitions

We assume a "Honest-but-Curious" Cloud Provider [7] and a "Malicious-Rational" Advertiser model.

- **Threat 1: Context Leakage via Inference.** Even with encrypted storage, cloud providers can infer sensitive user states by analyzing access patterns. *Defense: Sovereign Boundary Enforcement.*
- **Threat 2: Execution Hijacking.** Ad-injection algorithms introduce "sponsored" detours. *Defense: Protocol-Level Channel Isolation.*
- **Threat 3: Reflex Latency Failure.** CPS fails to brake due to RTT spikes > 100ms. *Defense: Hard Real-Time Routing.*

3 The Static System Architecture

To address the structural vulnerabilities identified in Section 2—specifically the Reflex Latency Failure (Threat 3) and the Context Leakage (Threat 1)—we propose the **"Agent Metabolic System."**

3.1 Architectural Overview

Before detailing the specific layers, we outline the governing philosophy. In the **Attention Economy**, systems were designed as "Cloud Pipes," optimizing for maximum data upload. This flat structure is insufficient for the **Intent Economy**. We restructure the agent into a **7-Layer Hierarchy** mirroring a biological organism. This fundamental transition is illustrated in Figure 1, which contrasts the legacy "Cloud-Centric Marionette" model with our proposed sovereign architecture.

The proposed system comprises four functional subsystems:

(1) **Subsystem A: Perception (L1-L2):** The "Sensory Organs." Goal: **Information Distillation Intent Gating**.
(2) **Subsystem B: Sovereignty (L3-L4):** The "Brain & Conscience." Goal: **Anchor identity**.

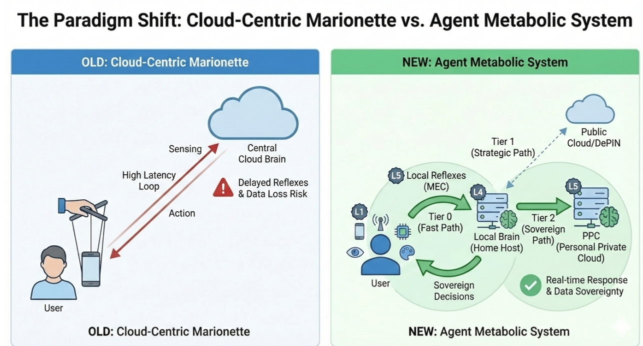

Figure 1: The Paradigm Shift: From Cloud-Centric Marionette to Agent Metabolic System. The new system relies on local Sovereign Skills to decouple intent from cloud execution.The metaphor is illustrative rather than prescriptive.

(3) **Subsystem C: Execution (L5-L6):** The "Muscles." Goal: **Actuate intent**.
(4) **Subsystem D: Settlement (L7):** The "Social Contract." Goal: **Objective auditability**.

Architecture as Defense: Unlike generic software stacks, this hierarchy is **derived directly from the adversarial threat model**:

- **Countering Threat 1 (Privacy):** We introduce a physical sovereignty boundary (**L4 Home Host**).
- **Countering Threat 3 (Safety):** We structurally decouple the "Reflex Arc" (**L5 MEC**) from the "Cognitive Core".
- **Countering Threat 2 (Integrity):** We establish a distinct **Settlement Layer (L7)**.

A detailed breakdown of these layers and their interactions is provided in Figure 2.

3.2 Layer Definitions

3.2.1 A. The Perception Layers. **L1: Personal Scout (Multimodal Lens & Gatekeeper)**. To address the Glass Wall Paradox, where high-bandwidth human intent is throttled by low-bandwidth touchscreens, L1 employs Multimodal Signal Fusion (Voice, Gaze, Gesture, Text) to capture high-fidelity intent (e.g., correlating "Put that there" with gaze coordinates). Crucially, to mitigate surveillance risks, this rich input stream is held in a local ephemeral rolling buffer (e.g., 30s window) which is continuously overwritten and remains inert. Only upon validation by a "Semantic Intent Gatekeeper'" (e.g., biometric pinch) is the relevant context snapshot frozen and processed. Benefit: Solves the Bandwidth Bottleneck by enabling natural, high-density interaction while enforcing Zero-Retention Privacy by default (data vanishes without explicit intent).

L2: Ambient Sensors (The Objective Reality).

- *The Problem:* Current GPS is 2D and "blind" to context; it provides coordinates but cannot distinguish a "door" from a "wall," failing to support physics-aware interaction.
- *The Solution:* L2 fuses **SLAM** (for geometry) with **Vision Transformers (ViT)** and **LiDAR** (for semantics and depth). This creates a dynamic **"Semantic Digital Twin"** of the immediate environment.

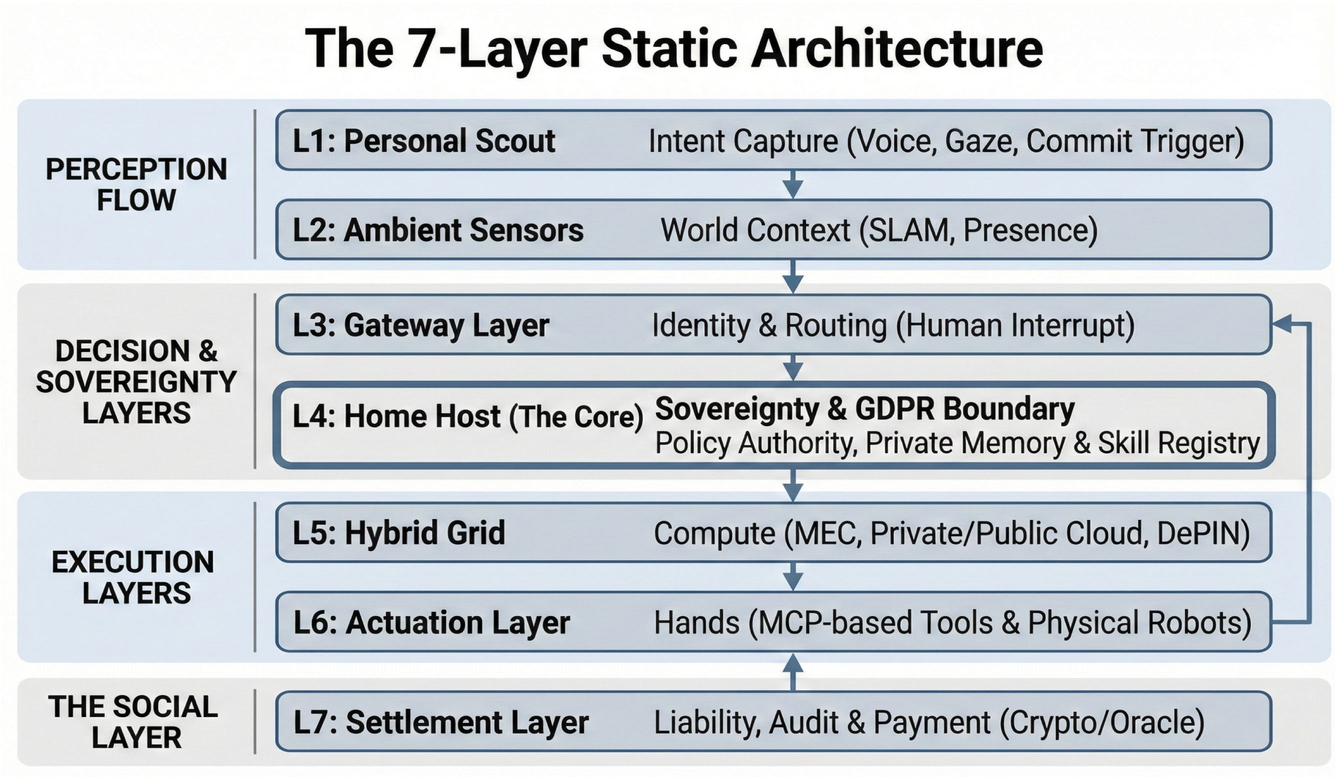

Figure 2: The 7-Layer Static Architecture: Detailed breakdown of the Agent Metabolic System.

- *Benefit:* Enables **Context-Aware Actuation**—the agent perceives not just the location of a table, but infers surface properties (e.g., glass vs. wood) to modulate force feedback.

3.2.2 B. The Decision & Sovereignty Layers. **L3: Gateway Layer (The Reflex Router).**

The Problem (Cognitive Latency Gap): Cloud AI operates on "Cognitive Time" (>500ms), while physical reality operates on "Kinetic Time" (<10ms). Connecting physical actuators directly to the cloud creates a dangerous latency mismatch (Threat 3).

The Solution: L3 acts as the "Spinal Cord," strictly enforcing Structural Decoupling. L3 continuously monitors the variance of the external network. It functions not just as a gate, but as a statistical filter that rejects volatile connections for safety-critical tasks: routing high-level semantic tokens to the Cloud (L4/L5) while retaining low-level safety reflexes (Collision Avoidance) on the Edge.

Split-Stack Architecture. To guarantee deterministic sub-10ms reflexes, we define L3 as a bifurcated system:

- **L3-Core (Control Plane):** A containerized software module handling asynchronous identity verification and policy injection.
- **L3-Edge (Data Plane):** A bare-metal enforcement layer (e.g., FPGA or eBPF). The **Barrier Function** logic is executed here, physically decoupling the safety-critical reflex loop from high-level OS jitter.

Benefit: Ensures Physical Safety is an architectural invariant, independent of network quality or model inference speed.

L4: Home Host (Sovereign Core). Countering the paradigm where "You are the Product," L4 anchors the user's identity and policy profile to a **local physical server**. This physical anchoring reclaims **Data Sovereignty**, ensuring that the control plane remains within the user's physical custody.

Furthermore, L4 functions as the "**Sovereign Skill Registry**." It maintains a local index of "Skill Manifests"—sovereign wrappers that encapsulate and govern the underlying L6 Tools.

Crucially, L4 drives system evolution through a "**Recursive Composition Engine**." It continuously observes user interaction patterns and autonomously composes atomic L6 Tools into new "**Macro Skills**," enabling the agent to expand its capabilities without firmware updates.

To ensure secure interoperability, L4 implements **a Sovereign MCP Client** (based on the Model Context Protocol). This client acts as a Semantic Firewall, intercepting all external MCP requests and applying Context Sanitization filters (e.g., blurring GPS data) based on the user's intent before allowing any external Tool to access the user's private context.

3.2.3 C. The Execution Layers. **L5: Hybrid Grid (Kinetic Router).** While public clouds optimize for throughput, they fail to address the latency requirements of physical agents. L5 introduces **Kinetic Routing** to dynamically distribute tasks across MEC, Public Cloud, and PCC based on physics rather than just cost. This ensures execution matches the task's physical constraints.

L6: Actuation Layer (Orchestration). The traditional internet is Read-Only for physical objects. To bridge this **Cyber-Physical Gap**, L6 unifies **Digital Tools (Bits)** and **Physical Actuators (Atoms)** under a standardized interface.

We adopt the Model Context Protocol (MCP) as the connectivity standard. This allows the agent to dynamically "mount" any MCP-compliant Executable Tool—whether a cloud-based SaaS or a local robotic arm. This approach decouples the tool's execution logic from the connection method, enabling **a universal plug-and-play ecosystem** where L6 provides the raw capabilities (Tools) that L4 orchestrates.

3.2.4 D. The Social Layer. **L7: Settlement Layer (The Witness & Banker).** To mitigate trust issues in headless commerce, L7 integrates **Cryptographic Oracles** with **Smart Contracts**. The Oracle logs sensor data as immutable evidence (e.g., verifying a delivery via GPS/Camera), which serves as the trigger condition for **Automated Atomic Payment**.

Additionally, L7 supports **Atomic Skill Micro-payments**. While L6 executes the Tools, the value is captured at the Skill level in L4. Since each Skill Manifest includes a pricing schema, the Settlement Layer executes cryptographic payments conditionally upon the verified completion of a task. This creates a Cognitive State Economy where developers are directly rewarded for the reliability of their Tools.

To mitigate high transaction costs, we employ Optimistic Settlement, where high-frequency micro-transactions are batched off-chain and only settled on-chain upon dispute or session closure.

Benefit: Replaces subjective Platform Discretion with **Mathematically Enforceable Settlement**, ensuring value transfer occurs *if and only if* the physical outcome is verified.

4 System Invariants

The current "Attention Economy" architecture suffers from structural conflicts, specifically *Context Leakage* and *Reflex Latency Failures.* To prevent the protocol from degenerating into these legacy patterns, we define five **System Invariants** (or "Iron Laws"). These constraints serve not merely as policy, but as the necessary structural constraints of the agent environment to guarantee stability and sovereignty.

(1) **Static Structure (Decoupling):**
- *Cause:* Cloud-centric control loops introduce unsafe latency (>100ms) for physical reflexes (Threat 3).
- *Invariant:* Strictly enforce the separation of the *Reflex Arc* (MEC) from the *Cognitive Core* (Home Host)) to ensure real-time safety.

(2) **Intent-Context Separation:**
- *Cause:* Service providers proactively infer sensitive user states from environmental data (Context Leakage, Threat 1).
- *Invariant:* Prohibit the environment from inferring intent; execution must be explicitly triggered by the user's *Commit Action.*

(3) **Sovereign Anchoring:**
- *Cause:* In purely cloud-based models, the user becomes the product, losing data ownership.
- *Invariant:* Mandates that the "Identity Root" and policy authority must remain physically localized (L4 Home Host).

(4) **Control Plane Separation:**
- *Cause:* Processing all high-bandwidth data centrally is inefficient and privacy-invasive.
- *Invariant:* The Sovereign Core (L4) issues authorization tokens (Policy) without needing to process every bit of raw sensor data (Mechanism).

(5) **Channel Isolation:**
- *Cause:* Ad-injection algorithms in the "Discovery State" often hijack user intent during execution (Threat 2).
- *Invariant:* Strictly segregate the *Execution State* (No Ads) from the *Discovery State* to preserve intent integrity.

Significance: By enforcing these invariants, the architecture ensures a transition from a "Cloud-Centric Marionette" model to a sovereign *Agent Metabolic System.* This guarantees that humans retain *Intent*, while machines are relegated to *Orchestration* and *Execution*, preventing the usurpation of agency by platform algorithms.

These invariants are expressed as architectural constraints rather than moral or regulatory prescriptions; they describe conditions under which system behavior degrades categorically rather than gracefully.

5 Cyber-Physical Orchestration

The transition from the "Attention Economy" to the "Intent Economy" necessitates a fundamental shift in how agents interact with reality. Current LLMs act as "Brains in a Vat," capable of high-level reasoning but disconnected from physical actuation. To bridge this **Cyber-Physical Gap**, Layer 6 (Actuation) functions not merely as a connector, but as a deterministic conductor that unifies the fluid nature of digital bits with the rigid constraints of physical atoms, as illustrated in Figure 3.

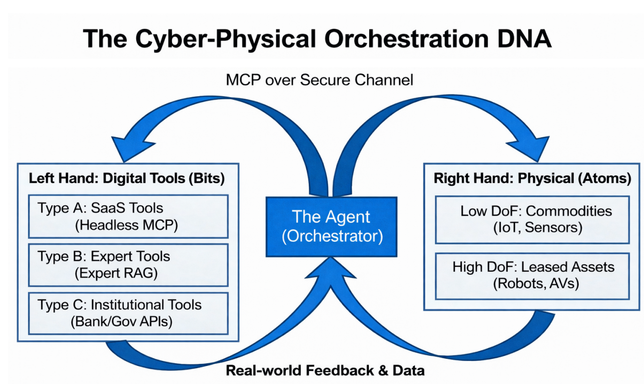

Figure 3: The Cyber-Physical Orchestration DNA: Coordinating Bits and Atoms.

5.1 The Two Hands: Dual-Modality Integration

Structural Bifurcation. To accommodate the fundamentally different constraints of software and hardware systems, the architecture explicitly separates agent tools into two execution modalities. The *Left Hand* (*Digital Modality*) governs purely informational actions by orchestrating a spectrum of "Digital Tools" (via standard MCP interfaces). This includes Type A: SaaS Tools (headless integrations), Type B: Expert Tools (knowledge model inference), and Type C: Institutional Tools (bank/gov API interactions). Its primary optimization objective is information velocity and logical consistency.

In contrast, the *Right Hand* (*Physical Modality*) manages cyber-physical actuation, ranging from low-degree-of-freedom (DoF) IoT sensors to high-DoF robotic systems. This modality prioritizes safety constraints, kinetic feasibility, and real-time responsiveness.

This bifurcation is not merely an implementation detail but a safety invariant: outputs produced in the Digital Modality are not permitted to directly trigger physical actuation. Any transition from symbolic intent to kinetic execution must pass through an explicit validation and synchronization protocol, thereby preventing hallucinated or inconsistent digital states from propagating into the physical world.

5.2 The Orchestration Loop: Causality & Sovereignty

Closed-Loop Control Protocol. Connectivity alone is insufficient for autonomous execution in cyber-physical environments. To enforce causal correctness and sovereignty guarantees, the system employs a closed-loop orchestration protocol governed by the Layer-4 Agent. This protocol converts user intent into verifiable real-world outcomes through a four-stage control cycle.

1. Cryptographic Authorization. Prior to execution, the agent functions as a policy authority, issuing cryptographically signed authorization tokens to the relevant execution modalities. No API invocation or physical actuation is permitted without explicit authorization bound to the user's sovereign identity. This mechanism prevents external injection, replay attacks, and unauthorized escalation across execution layers.

2. Parallel Actuation. Following authorization, the system enters the actuation phase. Digital actions (e.g., financial transactions or state updates) are executed by the Left Hand, while corresponding physical actions are initiated by the Right Hand. Parallel execution is essential for tasks in which digital and physical states must evolve synchronously, such as logistics, robotics, or smart infrastructure control.

3. Atomic Synchronization. To prevent state divergence—such as financial settlement without physical delivery—the agent enforces atomic synchronization across modalities. Acting as a centralized logical clock, it guarantees that digital commits and physical state transitions obey a strict causal order. Partial completion is disallowed: either both modalities succeed, or the transaction is rolled back.

4. Reality Feedback. The control loop closes through sensory feedback from the physical environment. Observations from sensors are propagated back to the perception layers (L1/L2), updating the agent's internal world model. This feedback enables post-hoc verification that observed physical outcomes match the intended digital plan, establishing end-to-end causal accountability.

6 Dynamic Operational Strategies

While the architecture defines the system's static structure, runtime behavior is governed by a *Kinetic Routing Model.* This model dynamically assigns tasks to execution environments by balancing three competing objectives: physical safety, data sovereignty, and economic efficiency. Rather than optimizing throughput alone, routing decisions are framed as constrained optimization under non-negotiable system invariants. The logical flow of this decision process is depicted in Figure 4.

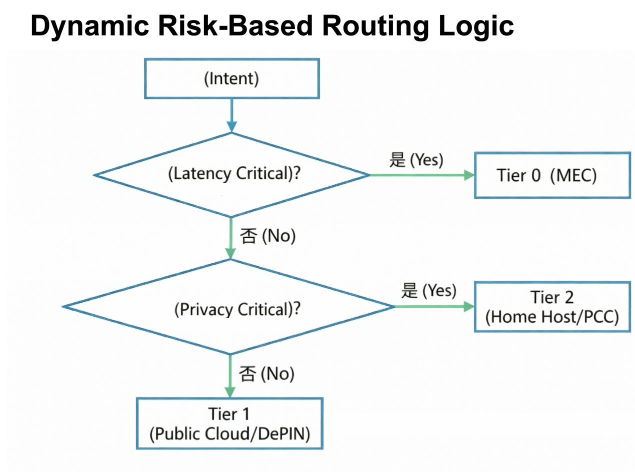

Figure 4: Dynamic risk-based routing logic flow.

6.1 Formalized Stochastic Routing Model

Unlike generic web traffic, cyber-physical agents operate under strict kinetic constraints where tail latency (jitter) is as dangerous as mean latency. We define the latency of route r as a random variable $L(r)$ with mean μ_r and variance σ_r^2. Latency variance is introduced here as a tractable proxy for tail risk rather than as a complete characterization of the latency distribution. In heavy-tailed and non-stationary network environments, mean latency provides little safety signal, while percentile-based statistics are inherently retrospective and adapt too slowly to sudden regime shifts. Variance, by contrast, captures early instability arising from queue buildup, congestion, and jitter, allowing risk to be detected before hard deadlines are violated. The system minimizes the following Risk-Aware Cost Function:

$$J(r_t) = w_c C(r) + w_l \exp\left(\frac{\alpha}{T_{hard} - \hat{L}_{risk}(r)}\right) + \beta \cdot \mathbb{I}(r_t \neq r_{t-1}) + \delta(r, P_{req}) \tag{1}$$

Where:

- **Risk-Aware Latency** $\hat{L}_{risk}(r) = \mu_r + k\sigma_r$: Represents the 3σ worst-case estimation (where $k = 3$), ensuring the router reacts to jitter before the deadline is breached.

- **Switching Penalty** β: A damping factor that penalizes route oscillation (chattering) to ensure actuation stability.
- **Sovereignty Veto** $\delta(r, P_{req})$: Returns ∞ if the route violates privacy constraints.

Unlike heuristic thresholding, the proposed formulation acts as a control barrier: as the estimated risk approaches the hard deadline, the routing cost diverges ($J \to \infty$), strictly vetoing unsafe execution paths rather than merely deprioritizing them.

Operational Logic: As the effective risk latency $\hat{L}_{risk}(r)$ approaches the hard deadline T_{hard} (e.g., 20ms), the exponential term dominates, causing the cost to approach infinity ($J \to \infty$). Crucially, this barrier is triggered not only by high mean latency but also by high variance (σ). Thus, the scheduler strictly vetoes routes that display unstable jitter, even if their average performance appears deceptively safe.

6.2 System Parameters and Invariants

The routing behavior is controlled by a small set of interpretable parameters (Table ??). Unlike conventional load-balancing strategies, the sovereignty penalty δ acts as a hard constraint rather than a soft preference.

This formulation allows flexible performance trade-offs while remaining uncompromising with respect to data sovereignty.

6.3 The Three Execution Tiers

The routing model induces a natural stratification of tasks into three execution tiers:

Tier 0 (Fast Path). Reserved for safety-critical reflex loops where latency dominates all other concerns. Tasks are executed at the network edge to guarantee sub−10 ms response times.

Tier 1 (Strategic Path). Used for non-sensitive, computation-intensive tasks such as information retrieval or planning. Execution is offloaded to scalable public infrastructure to optimize cost and throughput.

Tier 2 (Sovereign Path). Dedicated to privacy-critical tasks involving personal, medical, or domestic data. Execution is confined to private or locally sovereign compute environments, even at the expense of latency or cost.

7 The Cognitive State Economy

This architectural shift implies a corresponding economic transition. Instead of monetizing user attention, the system supports a *Cognitive State Economy* in which value is derived from the execution of explicit, committed intent. This new funnel, shown in Figure 5, prioritizes execution over passive consumption.This section discusses implications of the proposed architecture rather than a complete economic model.

Intent-Based Interaction Model. User interaction is segmented into exploratory states and execution states. While exploratory phases may tolerate interruption or persuasion, execution states are governed by a strict no-interruption invariant. Once intent is committed, the system optimizes solely for correctness and efficiency.

Agent Influence Optimization (AIO). In this setting, traditional attention-based optimization becomes ineffective. Instead, value accrues to services that provide verifiable, machine-trustable data.

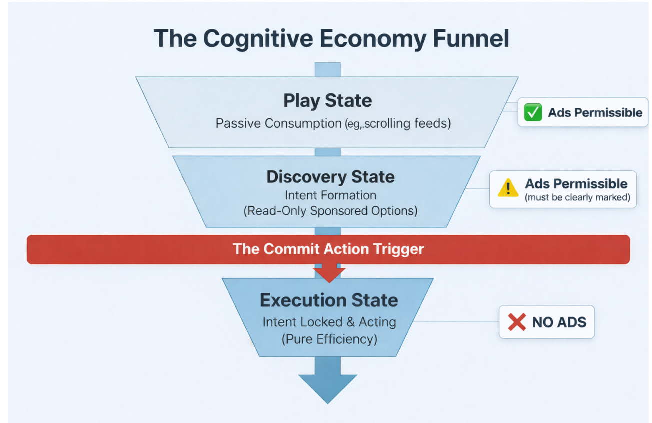

Figure 5: The cognitive economy funnel.

Economic competition shifts from persuasion to reliability, incentivizing an ecosystem grounded in auditability rather than interruption.

8 Case Study: A Multi-Step Cyber-Physical Task

To demonstrate the system in action, we present a case study involving a multi-step, real-world task: organizing a dinner party. The interaction sequence is visualized in Figure 6, detailing the flow of intent and execution across system layers.

- **Intent Resolution (L4):** Alice speaks "Plan Dinner." The L4 Recursive Engine resolves this intent into a composite "Dinner Macro Skill." Crucially, it consults the *Private Memory* (Sovereign Core) to retrieve safety constraints—specifically "No Nuts"—without exposing raw user data to the public cloud.
- **Guest Collaboration & Sanitization (L4):** Before ordering, L4 initiates a sanitized MCP query to the *Guest Agent* to confirm dietary alignment. Upon confirmation, it routes the command to the external *Vendor Tool*. The "Semantic Firewall" ensures that the final order payload contains the constraint ("Requirement: Nut-free") while stripping away sensitive identity details.
- **Digital Execution (L6):** Upon consolidating requirements, L4 dispatches the verified order to the Vendor Tool via the MCP interface. This generates a cryptographic receipt, binding the physical intent ("Nut-free groceries") to a digital obligation.
- **Digital Settlement (L7):** Once the digital receipt is verified, the L7 Oracle executes an *Atomic Micro-payment* for the groceries. This ensures the commercial transaction is cryptographically settled on the ledger before physical actuation begins.
- **Reflex Safety & Forensics (Tier 0 & L7):** During the physical execution ("Serve Dinner"), the robot encounters a dynamic obstacle (a cat). The onboard L2 sensors trigger the L3 Gateway's *Emergency Brake* (Tier 0, < 10ms), bypassing

Table 1: Kinetic Routing Parameters (Control Barrier Model)

Parameter	Symbol	Operational Interpretation
Barrier Scale	w_l	Scales the exponential barrier function. Determines the intensity of the "repulsive force" as latency approaches T_{hard}.
Cost Weight	w_c	Linear economic multiplier. Governs the rational cost-minimization behavior when the system is operating safely within the kinetic envelope.
Barrier Sensitivity	α	Controls the steepness of the exponential curve. A higher α creates a "harder" barrier, triggering avoidance reactions earlier.
Sovereignty Veto	δ	Non-negotiable constraint. Applies an infinite cost penalty ($J \rightarrow \infty$) if the route violates the specific P_{req} of the intent.
Risk Factor	k	Sets the confidence interval for latency prediction. typically $k = 3$ (99.7% confidence) for Tier-0 safety reflexes.
Stability Constant	β	A hysteresis parameter to prevent rapid switching (chattering) between Cloud and Edge when costs are nearly equal.

the slower L4 planning loop. Simultaneously, this "near-miss" event is cryptographically signed and logged to the L7 Oracle as immutable *compliance evidence*.
- **Final Trust Verification (L7):** After the task is completed safely, L7 performs a *Forensics Audit*. It reviews the on-chain safety logs to confirm that the emergency stop was valid and no harm occurred. Only after this clean audit does L7 release the final service payment, closing the full-stack trust cycle.

9 Quantitative Evaluation

To validate the stochastic robustness of the architecture, we conducted a discrete-event simulation ($N = 2000$) under adverse network conditions. Unlike previous studies that assume Gaussian latency, we modeled a **Heavy-Tailed Network Environment** characteristic of real-world 5G/Wi-Fi, introducing intermittent lag spikes (packet loss probability $p = 0.1$, spike magnitude $+15ms$).

We evaluated three routing strategies:

(1) **Cloud Baseline**: All control loops are processed in the cloud (End-to-End VLM).
(2) **Naive Router (Deterministic)**: Routes based solely on mean predicted latency ($\mu < T_{hard}$).
(3) **Risk-Aware Router (Ours)**: Routes based on the stochastic barrier function $\hat{L}_{risk} = \mu + 3\sigma$.

9.1 Result A: The "Tail Truncation" Effect (Safety)

As shown in Figure 7(A) and (B), the Cloud Baseline frequently violates the 20ms physical safety limit due to tail latency (Success Rate: 81.0%). Crucially, the Naive Router also fails (Success Rate: **85.8%**, see Figure 7C) because it ignores variance; it continues to route to the cloud when the *average* latency is low even if the *jitter* is critically high.

In contrast, our **Risk-Aware Router** achieves a **100% Safety Compliance** rate. Figure 7(B) visualizes this behavior: as network variance (σ) increases, the barrier function's cost penalty $J(r)$ rises exponentially, forcing the system to "fallback" to the local Tier-0 execution environment (5ms) *before* a violation occurs. This effectively "truncates" the latency tail, converting probabilistic risks into deterministic costs.

9.2 Result B: Sovereignty Compliance

In the second evaluation phase, we injected a workload of $N = 500$ sensitive intents (e.g., medical queries, biometric authentication).

- **Baseline Failure:** As shown in Figure 7(D), the cloud-centric baseline routed 100% of these requests to public endpoints, resulting in total context leakage. This confirms that without explicit architectural constraints, generic agents prioritize connectivity over sovereignty.
- **Sovereign Assurance:** The proposed system achieved **0% Leakage**. The Sovereignty Veto parameter $\delta(r, P_{req})$ in our routing function effectively acted as an infinite cost barrier, forcing all sensitive intents to be processed exclusively by the Tier-2 Private Core.

9.3 Result C: The Economic Cost of Certainty

Safety comes at a premium. The simulation data reveals that under high-jitter conditions, the Risk-Aware Router redirects **nearly 100%** of traffic to expensive Tier-0 compute (compared to the Naive Router's cloud-centric default). This confirms our hypothesis: the system dynamically trades economic efficiency for physical certainty, resulting in a **10x operational cost increase** when environmental entropy rises.

9.4 Sensitivity Analysis and Alternative Baselines

To address concerns that the Naive Router may be overly simplistic, we additionally evaluated a percentile-based routing strategy that routes to the cloud only when the estimated P99 latency (window size $W = 50$) remains below T_{hard}.

Simulation results reveal the limitations of this heuristic: under heavy-tailed network conditions, the P99 Router degenerates

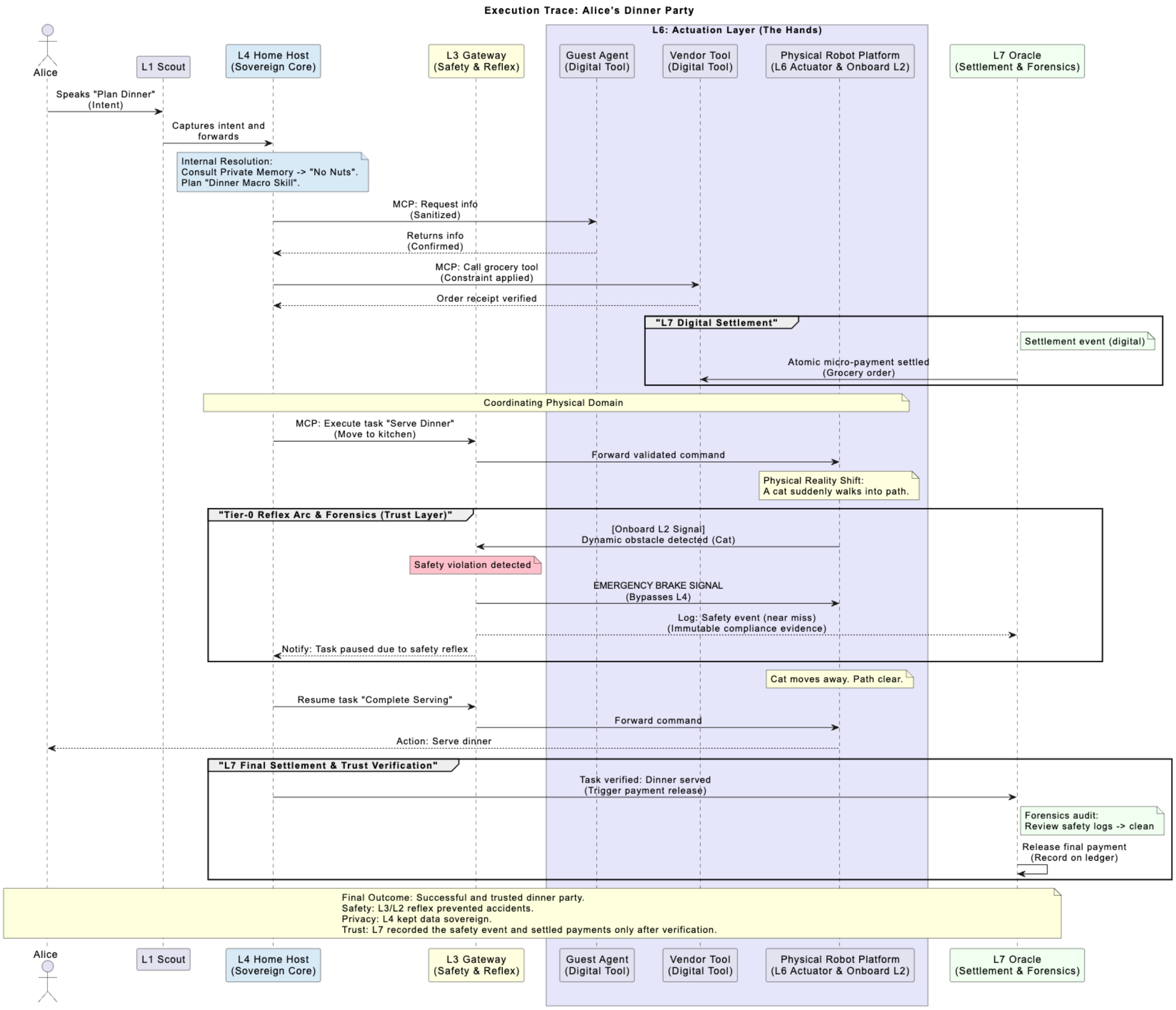

Figure 6: Execution Trace: Alice's Dinner Party (Protocol Interaction).

into an overly conservative fallback mode, utilizing expensive Tier-0 compute **99.7%** of the time (incurring costs comparable to our method). Despite this high cost, it fails to achieve perfect safety, sustaining a **0.1% violation rate**. This confirms that retrospective statistical heuristics lack the predictive power of the proposed instantaneous control barrier, resulting in worst-case economic inefficiency without deterministic safety guarantees.

10 Discussion

10.1 AGI as an Accelerant

Greater intelligence does not eliminate the need for intent-conditioned boundaries; it increases the expected cost of boundary violations. As agent competence grows, both the upside of delegated execution and the downside of mis-execution scale, reinforcing the need for architectural constraints that are enforced at the protocol level rather than assumed from model behavior.

10.2 10.2 The Economic Reality of Agency

It is crucial to note that in the tested adverse environment, the proposed system incurs an operational cost **10x higher** than the baseline. This disparity highlights the "Safety Premium." The baseline appears cheap because it externalizes the risks of physical accidents and privacy breaches. Our architecture internalizes these costs, paying market rates for premium Tier-0 and Tier-2 resources to ensure physical survival and sovereignty.

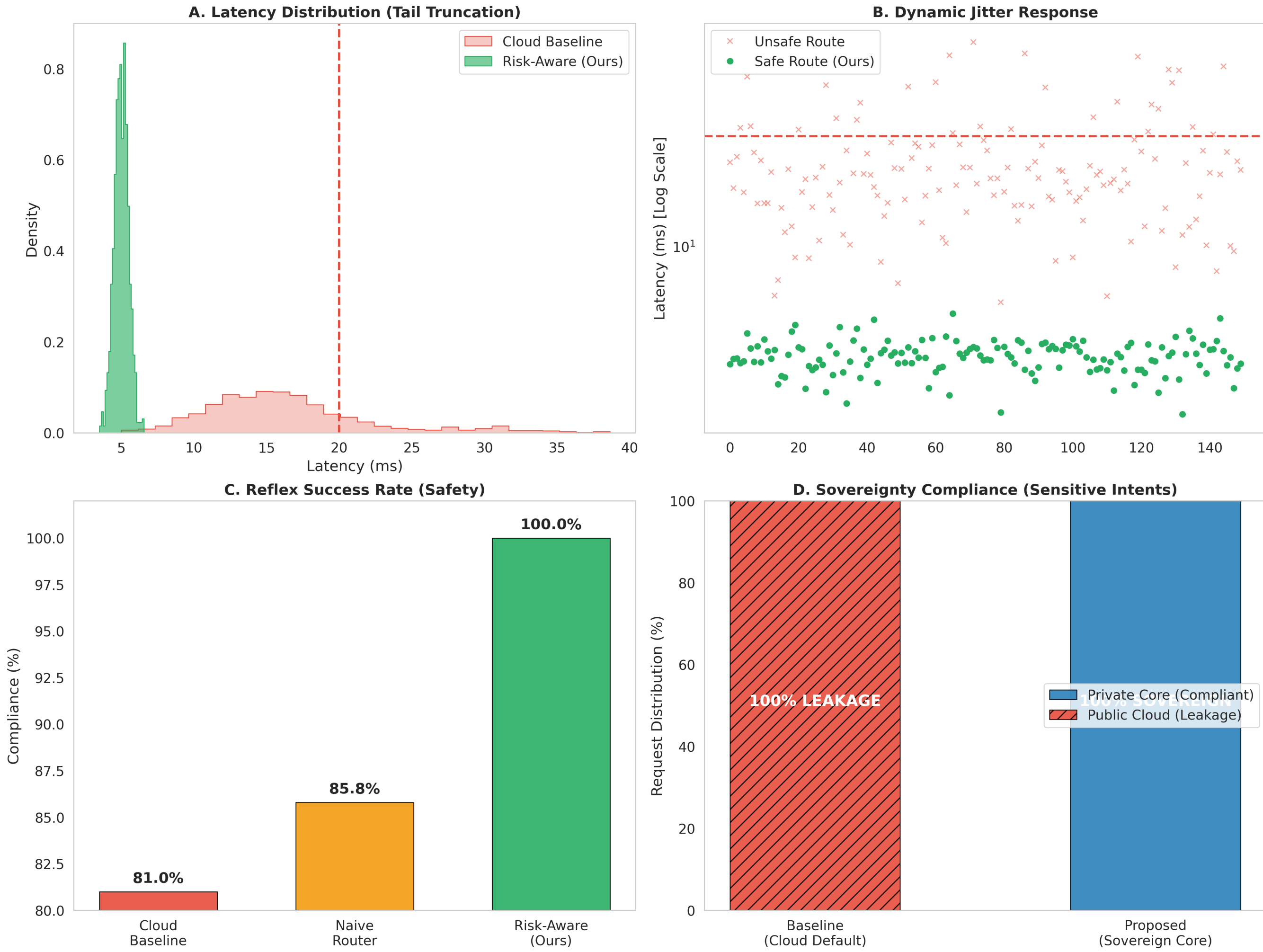

Figure 7: Comprehensive System Evaluation: Safety & Sovereignty. (A) Latency Distribution: The Risk-Aware Router (Green) strictly enforces the 20ms hard deadline by "truncating" the long-tail latency seen in the Cloud Baseline (Red). (B) Dynamic Jitter Response: Scatter plot showing how the system preemptively switches to Tier-0 (bottom green dots) when cloud variance increases, avoiding potential violations. (C) Safety Compliance: The Naive Router (Orange) fails (85.8%) because it ignores variance, whereas our Risk-Aware approach achieves 100% safety compliance. (D) Sovereignty Compliance: Under a workload of sensitive intents ($N = 500$), the proposed architecture enforces a strict "Sovereignty Veto," ensuring 0% context leakage compared to the Baseline's 100% leakage.

The economic implications discussed in this section are not formal claims, but consequences that follow if the proposed architectural constraints are adopted. They are included to contextualize incentives rather than to define a complete economic theory.

10.3 Technical Challenges

Several practical challenges remain for deployment at scale:

- **The edge inference gap.** The feasibility of a sovereign L4 depends on high-performance local inference (e.g., NPUs) and memory-efficient attention mechanisms to host "world-model" capabilities within constrained edge hardware.
- **The energy-efficiency paradox.** Always-on agentic workflows risk a Jevons-style rebound effect, where efficiency gains increase total consumption. Future implementations should prioritize *wake-on-intent* designs: ultra-low-power gating that activates high-power compute only when the L1 Personal Scout detects a committed intent threshold.
- **Verifiable orchestration.** Trust-minimized execution across heterogeneous providers requires verifiable claims about actions and routing decisions. Integrating zero-knowledge proofs (e.g., zk-SNARKs) with DePIN-style routing is a promising direction to enable verification without revealing sensitive payloads.

10.4 The Spectrum of Agency: Implementation Paths

We anticipate that the Agent Singularity will emerge as a spectrum of implementations rather than a single monolithic transition.

The Sovereign Agent (user-anchored). This path adheres strictly to the seven-layer sovereign model with a physically user-controlled L4 Home Host. It is likely to be adopted by privacy-sensitive individuals, enterprises, and regulated sectors (e.g., finance and health), where sovereignty is non-negotiable.

The Platform Agent (cloud-anchored). This path is provided by major technology ecosystems. While such implementations may centralize components for convenience, competitive pressure will increasingly require adoption of inter-agent protocols (A2A) and privacy-preserving execution (e.g., PCC) to operate effectively in a heterogeneous service economy.

Coexistence prediction. As closed ecosystems and open standards have historically coexisted, sovereign and platform agents are likely to run in parallel. In this hybrid future, the proposed System Invariants serve a dual role: they provide a design blueprint for sovereign agents and an audit benchmark for platform agents, enabling external verification that agents act as fiduciaries for users rather than as optimization surfaces for platforms.

11 Conclusion

The future of AI is not primarily about replacing humans, but about *agency*. A cloud-centric "marionette" trajectory leads to brittle control, elevated surveillance risk, and degraded user autonomy. In contrast, the Agent Metabolic System treats intelligence as a physiological extension of the user, grounded in sovereign compute, risk-based routing (Tier 0–2), and strict separation of cognitive states. We hope this work serves as a systems-level reference point rather than a prescriptive blueprint.

Under this world model:

- Humans retain **intent**.
- Agents assume **orchestration**.
- Machines provide **execution**.

References

[1] 2016. Mobile Edge Computing (MEC); Framework and Reference Architecture.

[2] Harrison Chase. 2023. LangChain: Building applications with LLMs through composability. *Software documentation* (2023). https://python.langchain.com/.

[3] European Commission. 2024. The Artificial Intelligence Act. *Official Journal of the European Union* (2024).

[4] Edward A Lee. 2008. Cyber physical systems: Design challenges. In *2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC)*. IEEE, 363–369.

[5] Yanyi Mao, Changsheng You, Jun Zhang, Kaibin Huang, and Khaled B Letaief. 2017. A survey on mobile edge computing: The communication perspective. *IEEE Communications Surveys & Tutorials* 19, 4 (2017), 2322–2358.

[6] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.

[7] Raluca Ada Popa, Catherine M Redfield, Nickolai Zeldovich, and Hari Balakrishnan. 2011. CryptDB: protecting confidentiality with encrypted query processing. In *Proceedings of the twenty-third ACM symposium on Operating systems principles*. 85–100.

[8] Toran Bruce Richards. 2023. Auto-GPT: An autonomous GPT-4 experiment. *GitHub repository* (2023). https://github.com/Significant-Gravitas/Auto-GPT.

[9] Paul Voigt and Axel Von dem Bussche. 2017. *The EU General Data Protection Regulation (GDPR): A Practical Guide*. Springer.