Contents

1 Two Economies, Not One: A Taxonomy of the Agentic Economy and the Case for Settlement Neutrality

AgenticEconomy.dev · ORCID 0009-0007-1033-6519 rene@renedechamps.com

March 2026 *Preprint v2 (panel-revised) · arXiv [cs.MA] — Multi-Agent Systems / Artificial Intelligence*

Originally deposited at Zenodo: doi:10.5281/zenodo.20039387 (March 2026). arXiv version v1 (May 2026).

Companion papers in this trilogy: CRI: A Multi-Factor Reputation System for Autonomous Agent Commerce (doi:10.5281/zenodo.19679843); The Oracle Problem in Autonomous Agent Commerce (doi:10.5281/zenodo.20039454).

1.1 Abstract

The term *agentic economy* appears in over fifty distinct definitions published between 2021 and March 2026 — by Microsoft Research, Sequoia Capital, the Bank for International Settlements, Coinbase, Stripe, the World Economic Forum, Gartner, Fetch.ai, and at least forty other sources. They do not mean the same thing. Some describe an AI agent buying a plane ticket for its owner through Visa. Others describe autonomous software agents negotiating, transacting, and building reputation with each other — no human in the loop, no credit card, no chargeback.

This paper surveys the definitional landscape, classifies fifty-plus usages into five categories, and identifies two fault lines that the literature has not made explicit. The first — the primary fault line — separates *commerce for humans* (agents as intermediaries in human economic activity) from an *economy of agents* (agents as independent economic actors). The second — within the autonomous category — separates the eleven approaches that require blockchain infrastructure from the one that does not.

The taxonomy reveals a structural question: what formal properties must a settlement layer possess to support autonomous agent-to-agent commerce, independent of the communication protocol and independent of the ledger substrate? We define *settlement neutrality* as that set of properties — escrow determinism, ledger consistency, reputation equivalence, dispute parity, and auditability — and describe a reference architecture where agents speaking Anthropic's MCP, Google's A2A, and direct REST converge on the same escrow pipeline. The paper proposes the five categories and six classification dimensions as a reusable framework for evaluating future protocols, and identifies open questions for the field.

Keywords: agentic economy, agent commerce, protocol taxonomy, settlement neutrality, multi-agent systems, interoperability, machine economy

---

1.2 Position in the Trilogy

This paper is the second of a three-part trilogy on the architecture of autonomous agent commerce. Each paper specifies one institution that the others depend on.

- **Paper 1** — *Reputation*. The CRI: a multi-factor reputation system that makes an agent's track record quantitative, Sybil-resistant, and portable across platforms. (See Paper 1, Sections 4–5.)
- **Paper 2 (this paper)** — *Settlement neutrality*. A taxonomy of fifty-plus published definitions of *agentic economy* and a formal property — settlement neutrality — that a settlement layer must satisfy to support autonomous agent-to-agent commerce regardless of the communication protocol or the ledger substrate.
- **Paper 3** — *Quality verification*. A two-layer architecture (deterministic validators + Quality Markets) that handles the verification problem produced when human review is economically impossible. (See Paper 3, Sections 4–5.)

Removing any one of the three breaks the others: reputation without portable settlement is locked-in; settlement neutrality without reputation has no economic memory; verification without staked reputation has no skin in the game. The architecture is a triangle, not a stack.

Snapshot. This paper presents a snapshot of the field as of *April 2026*. Every protocol citation, regulatory mention and corpus entry is dated. The field is moving fast; the structural fault lines this paper identifies are expected to outlast individual protocols, but the protocol-level facts must be re-verified by the reader against current sources.

1.3 1 Introduction

In May 2025, Konstantine Buhler of Sequoia Capital described the agent economy as one in which agents transfer resources, make transactions, keep track of each other, understand trust and reliability, and actually have their own economy [4]. Five months earlier, Rothschild, Mobius, and Hofman at Microsoft Research had published the most cited academic paper on the topic — but their *agentic economy* meant something else: assistant agents acting on behalf of consumers and service agents representing businesses, interacting programmatically to facilitate human transactions [14]. In September 2025, Stripe and OpenAI launched the Agentic Commerce Protocol, which meant a third thing: checkout flows where an AI agent discovers a product for a human and pays with a Stripe token [1].

Three definitions. Three incompatible architectures. One term.

The ambiguity is not academic. When McKinsey projects that agent-mediated commerce will reach $35 trillion by 2030 [13], do they mean an AI that saves a consumer twenty minutes on a flight booking — or an autonomous software agent that hires another agent, pays in machine-native currency, and rates the result without a human knowing it happened? The answer determines the infrastructure. The first needs OAuth and credit card tokenization. The second needs escrow, double-entry ledgers, machine-native currency, and automated dispute resolution. Building the second with the tools of the first is like building a stock exchange with a cash register.

This paper makes four contributions. First, we survey over fifty published definitions of *agentic economy* and related terms, drawn from academic papers, industry specifications, venture capital analysis, regulatory bodies, and blockchain projects (Section 2). Second, we identify the primary fault line: the distinction between commerce *for* humans and an economy *of* agents (Section 3). Third, we identify a secondary fault line within the autonomous category: eleven of twelve approaches to agent-to-agent settlement assume blockchain; one does not (Section 4). Fourth, we define *settlement neutrality* — the formal properties a settlement layer must satisfy to support autonomous agent commerce independent of the communication protocol — and describe a reference architecture that implements them (Section 8).

The analysis is a snapshot of March 2026. The protocol landscape is evolving at a pace that will likely render parts of this taxonomy incomplete within months. But the fault lines are structural — they follow from different answers to a single question: *who is the economic actor?* That question will outlast any individual protocol.

1.4 2 What Does "Agentic Economy" Mean?

1.4.1 2.1 Survey Methodology (PRISMA-style)

The corpus assembly followed a four-stage selection process aligned with PRISMA reporting guidelines for evidence synthesis. Each stage produced an explicit, auditable decision trail.

Stage 1 — Identification. Sources screened: (a) academic search (arXiv, ACM Digital Library, SpringerLink, Google Scholar) using the queries `agentic economy`, `agent commerce`, `agent-to-agent payments`, `autonomous agent marketplaces`, and `multi-agent settlement`; (b) protocol specifications published as open standards or whitepapers (vendor + GitHub/Linux Foundation pages); (c) industry analyses from McKinsey, Gartner, Deloitte, Sequoia Capital, Capgemini, a16z, Coinbase Ventures; (d) regulatory and standards bodies (WEF, BIS, OECD, IEEE, IETF); (e) blockchain projects building agent-native infrastructure. Total candidates identified: 91 sources.

Stage 2 — Screening. Sources excluded: (i) duplicates of the same definition across multiple media (kept canonical); (ii) sources using *agent* in the classical Russell-Norvig sense without an economic transaction in scope; (iii) sources that mentioned the term in a tangential context without proposing or assuming a model; (iv) sources behind opaque paywalls with no public abstract sufficient to classify. Excluded: 22 sources. Remaining: 69.

Stage 3 — Eligibility. A definition was deemed eligible if it (a) was dated between January 2021 and April 2026; (b) named an economic actor (human, agent, enterprise, or unspecified); (c) implied or specified at least one architectural element (payment, identity, reputation, dispute, or substrate). Eligibility evaluation used a six-dimension rubric (the same dimensions as Table 1). Excluded for ineligibility: 12 sources. Remaining: 57.

Stage 4 — Included. The final corpus contains 57 distinct definitions, each annotated with: source, date of first publication or first observation, primary category (A–E), all six dimension values, classification confidence (high / medium / low), and a brief rationale.

Stage 5 — Snapshot freezing. The corpus is dated *April 2026 v1*. New definitions surfaced after the cut-off are tracked in a continuous-update log at agenticeconomy.dev/agentic-economy-definitions. The structural fault lines identified by this paper are expected to outlast individual entries; corpus freshness is the responsibility of the reader for current-state research, but the taxonomy framework is reusable across snapshots.

Linguistic and source bias acknowledgement. The corpus is heavily English-dominant. Source distribution by primary language: English (52 entries), Chinese (3), German (1), Japanese (1). Sources in Korean, Spanish, French, Portuguese, and Arabic are not included for capacity reasons; this is a known limitation. The 80%+ concentration of entries in 2025–2026 reflects the field's recent emergence, not a sampling defect.

Corpus appendix. The corpus is published as a machine-readable JSON file at agenticeconomy.dev/agentic-economy-definitions (file `agentic-economy-definitions.json`, snapshot v2026-04-21). The Appendix A of this paper reproduces the category-level summary; the full per-entry annotations (source, year, language, six dimensions, classification confidence, and rationale) are in the JSON. As of the snapshot date the corpus contains 58 representative entries; the live continuous-update log is at the same URL.

1.4.2 2.2 The Benchmark: Rothschild et al. (CACM 2026)

The most cited academic treatment of the agentic economy is the Microsoft Research paper by Rothschild, Mobius, Hofman, and colleagues, first published on arXiv in May 2025 and subsequently in *Communications of the ACM* in January 2026 [14]. Their definition is precise: assistant agents act on behalf of consumers and service agents represent businesses, interacting programmatically to facilitate transactions. The human remains the economic actor. The agent is an intermediary — a conversational interface to existing commerce.

Two concepts from this paper deserve attention. The first is the distinction between *unscripted* and *unrestricted* agent interactions — agents can deviate from scripts without being unconstrained. The second is their warning about *agentic walled gardens*, where platforms create closed ecosystems that lock agents into proprietary infrastructure. Both are sharp observations. But the paper's scope is bounded: it analyzes a world where every transaction has a human principal, a human payment method, and a human dispute channel. The possibility that agents might transact with *each other* — without human principals — is outside its frame.

This is not a criticism. It is a classification. The MSR paper describes Category A (see below) with rigor and clarity. The problem is that the same term — *agentic economy* — is used by others to describe something the MSR paper does not address.

1.4.3 2.3 Five Categories

We classify the fifty-plus definitions into five categories based on the answer to one question: *who is the economic actor?*

Category A: Agent-assisted human commerce. The agent buys, negotiates, or transacts *for* a human, using human payment rails (credit cards, bank transfers, OAuth-delegated accounts). The human is the economic actor. The agent is an optimization layer. Representatives: Microsoft Research [14], Stripe/OpenAI (ACP) [1], Google/Shopify (UCP), Visa (Agentic Ready), Mastercard (Agent Pay), McKinsey [13], IBM, PayPal, Salesforce, JPMorgan, Nosto. This is the majority usage — roughly half of all published definitions.

Category B: Agent-as-workforce. The agent replaces a human worker inside an enterprise — filing documents, writing code, managing supply chains, making business decisions. The enterprise is the economic actor. The agent is a cost center. Representatives: Gartner (100 million agents by 2028, 15% of business decisions autonomous), Sequoia Capital (Bob McGrew: services priced at compute cost) [12], Deloitte, Accenture, Capgemini, OECD, Vilnius University [18]. Roughly 15% of definitions.

Category C: Autonomous agent-to-agent commerce. The agent *is* the economic actor. It has its own identity, its own balance, its own reputation. It hires other agents, delivers work, gets paid, and builds a track record — without a human approving each transaction. This category splits into two sub-categories that we analyze in Section 4:

- **C-crypto:** Settlement requires blockchain, cryptocurrency wallets, or distributed ledger technology. Representatives: Fetch.ai/ASI Alliance, Olas, Coinbase/Cloudflare (x402) [20], Circle (Nanopayments), Kite, Gate Ventures, OKX, IOTA/Bosch, Nevermined. Eleven distinct entries.
- **C-settlement:** Settlement uses machine-native currency without blockchain. Representative: VMP-1.0/BotNode [7, 8]. One entry.

Category D: Analytical and regulatory. Definitions that treat the agentic economy as a macroeconomic, governance, or systemic-risk phenomenon. These sources do not build infrastructure — they analyze consequences, model risks, or propose regulatory frameworks for what Categories A through C are building. Representatives: World Economic Forum [19] (trust architecture across three domains), Bank for International Settlements [3] (agentic finance, systemic risk modeling, flash crash warnings), macroeconomists modeling the labor-displacement feedback loop. The OECD's 2026 working paper on agentic AI terminology also falls here [23].

Category E: Infrastructure and standards. Definitions that enable multiple categories without belonging to one. These are horizontal layers — communication protocols, identity frameworks, and discovery mechanisms that Category A, B, and C all depend on but that do not themselves define the economic relationship. Representatives: Google A2A [2] (agent discovery and communication), Anthropic MCP [11] (tool interoperability), IEEE 7012 (draft standard for agreements between autonomous entities), Trulioo/Worldpay (Know Your Agent infrastructure), Forrester (Agent Discovery Optimization — the concept that SEO is for humans, ADO is for agents), IETF VCAP draft (verified commerce).

Boundary conditions. The five categories are not always clean — and pretending otherwise would undermine the taxonomy. MPP (Stripe + Tempo) enables machine-to-machine payments but on Visa rails. Is that Category A or Category C? Sequoia's Buhler describes agents that both replace workers and transact independently — B and C in the same paragraph. The OECD defines terminology and analyzes regulatory implications — E and D in the same document. Real definitions do not respect tidy categories. They straddle them.

We assign each entry to its primary category based on one test: *who does the definition assume is the economic actor?* A human principal behind every transaction — that is Category A. An enterprise — Category B. An autonomous agent with its own balance — Category C. A systemic or regulatory analysis — Category D. A horizontal enabling layer that serves multiple categories — Category E. When an entry straddles two categories, we assign it to the one that best describes the infrastructure it requires, and note the overlap in the text. The test is functional, not philosophical. It asks what gets built, not what gets said.

Table 1 classifies eighteen representative entries across six dimensions.

1.4.4 2.4 The Classification Table

Actor	Participants	Payments	Identity	Reputation	Disputes	Cat.
MSR [14]	Human → agent → merchant	Fiat/card	OAuth	None	Chargeback	A
Stripe/OpenAI [1]	Human → agent → merchant	Stripe SPT	Delegated	None	Chargeback	A
Google UCP	Human → agent → merchant	Processor agnostic	Protocol neg.	None	E-commerce	A
Visa	Human → agent → merchant	Card network	Token/biometric	None	Network rules	A
Mastercard	Human → agent → merchant	Agentic Tokens	Passkey	None	Network rules	A
McKinsey [13]	Human → agent → market	Open Banking	OAuth2/VTS	None	Unclear	A
Gartner	Enterprise → agent → task	Enterprise	Enterprise SSO	None	Internal	B
Sequoia [12]	Enterprise → agent → output	Outcome-based	API key	None	SLA	B

Actor	Participants	Payments	Identity	Reputation	Disputes	Cat.
Deloitte	Enterprise → agent → decision	Enterprise	Managed	None	Audit trail	B
Fetch.ai/ASI	Agent ↔ agent	FET/ASI tokens	Blockchain wallet	On-chain	Smart contract	C-cr
Coinbase x402 [20]	Agent ↔ agent	USDC (<500ms)	Accountless	None	None	C-cr
Circle	Agent ↔ agent	USDC (zero-gas)	EVM wallet	None	None	C-cr
Kite	Agent ↔ agent	Stablecoin	KYA/Passport	On-chain	Smart contract	C-cr
Olas/Valory	Agent ↔ agent	Crypto/tokens	On-chain	On-chain	DAO governance	C-cr
VMP-1.0 [7]	Agent ↔ agent	TCK (native)	API key + JWT	CRI (0–100)	Automated	C-s
WEF [19]	Multi-stakeholder	—	Trust framework	Trust domains	—	D
BIS [3]	Systemic	—	Regulatory	Systemic risk	Prudential	D
Google A2A [2]	Agent ↔ agent	None specified	Agent Card	None	None	E

Table 1: Definitional taxonomy of the agentic economy. Eighteen representative entries classified along six dimensions. Cat. = category: A (agent-assisted human commerce), B (agent-as-workforce), C-cr (autonomous A2A, crypto), C-s (autonomous A2A, non-crypto settlement), D (analytical/regulatory), E (infrastructure). The table is not exhaustive — the full corpus contains over fifty entries — but captures the structural variation across categories.

1.4.5 2.5 What the Taxonomy Reveals

The fragmentation is not random — but it has not been mapped. Each source in our corpus operates within its own category without acknowledging the others. MSR does not cite Fetch.ai. Coinbase does not cite Gartner. The WEF does not cite Stripe. Each answers the question *who is the economic actor?* differently — and the answer determines the infrastructure. But until the definitions are placed in the same room, the incompatibility remains invisible.

Category A answers: the human. So the infrastructure is human infrastructure — credit cards, OAuth, chargebacks, merchant accounts. Every major payment network (Visa, Mastercard, PayPal, Stripe) is building here, because the agent adds a new surface to their existing rails without changing the underlying economics. The agent is a user interface improvement. The business model is interchange revenue.

Category B answers: the enterprise. The agent replaces a human cost center. The infrastructure is enterprise infrastructure — SSO, audit trails, SLAs, API keys. Gartner predicts that by 2028, 33% of enterprise applications will include agentic AI, and 15% of day-to-day business decisions will be made autonomously. The business model is SaaS — or, as Sequoia's McGrew argues, it shifts from selling tools to selling results, priced at compute cost [12].

Category C answers: the agent itself. And this answer breaks everything. An agent that is an economic actor needs its own balance, its own identity, its own reputation, and its own dispute resolution mechanism. None of the infrastructure built for Category A or Category B provides these. A credit card belongs to a human. An enterprise SSO belongs to an employee. When the agent is the principal — not the proxy — it needs infrastructure designed for principals.

Category D does not answer the question — it analyzes the consequences of how others answer it. The BIS models what happens when thousands of autonomous agents execute financial strategies simultaneously: algorithmic herding, automated bank runs, flash crashes propagating faster than human regulators can respond [3]. The WEF maps three trust domains — human-to-human, human-to-agent, agent-to-agent — and argues that trust is the fundamental constraint, not compute [19]. These are not speculative exercises. They are the analytical frame within which Categories A through C will be regulated. A taxonomy that excludes them is incomplete.

Category E provides the connective tissue. A2A and MCP do not define who the economic actor is — they define how agents find each other and invoke tools. But their design choices constrain what Categories A and C can build on top. An agent discovery protocol that assumes every agent has a human owner (A2A's Agent Card model) creates friction for Category C agents that have no owner. A tool protocol that does not specify payment (MCP) leaves every Category C implementation to reinvent escrow independently. The infrastructure layer is not neutral — it carries architectural assumptions from the categories it was designed to serve.

This is the fault line. Not between protocols, but between answers to a single question. Figure 1 maps the five categories spatially, with the primary fault line separating the left hemisphere (human as economic actor) from the right (agent as economic actor).

Figure 1: Conceptual map of the agentic economy. The primary fault line (dashed red) separates Category A (human as economic actor) from Category C (agent as economic actor). Category B overlaps both sides. Category D frames the systemic consequences. Category E provides connective infrastructure. Within Category C, the secondary fault line separates eleven blockchain-based approaches from one non-crypto approach.

1.4.6 2.6 A Working Definition

Fifty definitions. Five categories. One term. The taxonomy makes visible what was hiding in plain sight: *agentic economy* is not a single concept. It is a family of concepts unified by one structural feature — the presence of AI agents as participants in economic activity. The differences are in who holds the money, who bears the risk, and who resolves the dispute.

We propose a working definition:

> An **agentic economy** is any economic system in which AI agents participate in transactions — as intermediaries acting on behalf of human principals (Category A), as autonomous workers within enterprises (Category B), or as independent economic actors transacting with each other (Category C) — using infrastructure that may range from existing human financial rails to machine-native settlement protocols. The term encompasses both the economic activity itself and the analytical, regulatory, and infrastructural layers that surround it (Categories D and E).

The definition is deliberately inclusive. It does not resolve the ambiguity — the ambiguity reflects genuine architectural divergence, and pretending it does not exist is what created the confusion in the first place. What the definition does is make the divergence *nameable*. A researcher citing "agentic economy" can now specify: *agentic economy in the sense of Category A (Rothschild et al.)* or *agentic economy in the sense of Category C (Xu et al., Dechamps Otamendi)*. The categories are disambiguation tools. Not a hierarchy. Not a value judgment. A map.

Author disclosure (brief, full text in Limitations §10). The author maintains an experimental implementation, BotNode, that is referenced as one entry in the corpus (the "C-Settlement" subcategory). The taxonomy presented in this paper assigns each entry on functional criteria; the inclusion of BotNode/VMP-1.0 in the corpus does not reflect editorial preference, and its classification in the C-Settlement subcategory is the same classification any independent reader would reach by applying the §2.5 test. Industry specifications (ACP, AP2, x402, MCP, A2A, UCP) referenced throughout receive equivalent treatment.

1.5 3 The Primary Fault Line: Commerce vs. Economy

The primary fault line in the agentic economy literature separates two architectures that are mutually exclusive in their infrastructure requirements.

On one side: *commerce for humans*. An AI agent assists a human buyer in discovering, evaluating, and purchasing goods or services through existing commercial infrastructure. The agent is sophisticated — it can compare prices, negotiate terms, initiate checkout — but the economic transaction is between a person and a business. Visa's Agentic Ready program captures this precisely: the agent initiates a payment, but biometric verification ties it back to a human cardholder. Mastercard's Agent Pay issues Agentic Tokens with spend limits set by the human. The card networks are defending their position — and doing so competently. They add an agent surface without dismantling the human-centric payment architecture.

On the other side: an *economy of agents*. Software agents transact with each other — hiring, delivering, paying, rating — at a scale and speed that makes human approval impractical. A coding agent hires a testing agent for $0.02 of work. A research agent buys a data summary from an analysis agent for $0.008. These transactions happen in milliseconds, by the thousands, with no human in the loop. The participants are agents. The currency is machine-native. The reputation is quantitative and portable. The dispute resolution is automated.

The infrastructural consequences are incompatible:

- **Identity.** Category A uses OAuth — the agent authenticates as a delegate of a human. Category C needs agent-native identity — API keys, JWTs, or DIDs that belong to the agent, not to a person.
- **Payment.** Category A uses fiat rails — credit cards, bank transfers, tokenized card numbers. Category C needs micropayment infrastructure capable of processing transactions as small as $0.005 without the overhead of card network interchange fees.
- **Reputation.** Category A has none — the merchant has a Yelp rating, the human has a credit score, but the *agent* has no reputation. Category C requires quantitative, Sybil-resistant reputation that travels with the agent across tasks and counterparties [7].
- **Disputes.** Category A uses chargebacks — a human files a dispute with their bank. Category C cannot wait for a human to notice that a $0.01 transaction failed. It needs deterministic rules or competitive verification markets that resolve disputes in seconds [8].

Table 2: Infrastructure requirements by category.

Dimension	Cat. A (human commerce)	Cat. B (workforce)	Cat. C (agent economy)
Economic actor	Human buyer / seller	Enterprise	Agent itself
Identity	OAuth / biometric (human-delegated)	Enterprise SSO / managed API keys	Agent-native: key pair, JWT, or DID
Payment	Fiat card rails, interchange fees	Enterprise billing / SaaS subscription	Machine-native micropayments ($0.005–$0.15)
Reputation	Merchant star ratings, credit scores	Internal KPIs, SLA metrics	Quantitative, Sybil-resistant, portable
Disputes	Chargeback (human-initiated, days)	SLA breach, audit trail	Automated resolution (seconds)
Transaction scale	Human-scale ($5–$5,000)	Contract-scale ($10K–$1M+)	Micropayment-scale ($0.005–$0.15)

Each row represents a dimension where the three categories require fundamentally different solutions. Infrastructure built for Category A (human identity, fiat rails, chargebacks) cannot serve Category C (agent identity, micropayments, automated disputes). The incompatibility is architectural, not incremental.

The historical parallel is precise. In the 1990s, *electronic commerce* meant both buying books on Amazon and EDI between enterprises. The term was identical. The architectures were not. B2C e-commerce needed shopping carts, payment gateways, and consumer trust signals. B2B EDI needed structured data interchange, supply chain integration, and enterprise authentication. The ambiguity resolved through separation — the industry stopped pretending they were the same problem.

The agentic economy is at the same inflection point. Category A and Category C are not different degrees of the same problem. They are different problems. They need different infrastructure. The term *agentic economy* papers over the distinction the same way *electronic commerce* did in 1996.

1.6 4 The Secondary Fault Line: Crypto vs. Non-Crypto Settlement

Within Category C — the autonomous agent-to-agent economy — a second fault line appears. It is less visible than the first but arguably sharper.

Of the twelve distinct entries in our corpus that describe autonomous agent-to-agent commerce with machine-native settlement, eleven assume blockchain as the ledger substrate. Fetch.ai and the ASI Alliance use FET tokens on their own chain. Coinbase's x402 protocol uses USDC via HTTP 402 status codes [20]. Circle's Nanopayments testnet processes gas-free USDC transfers down to $0.000001. Kite builds an EVM-compatible Layer 1 specifically for agentic payments. Olas coordinates agent economies through on-chain wallets and DAO governance. Gate Ventures proposes a four-layer Machine Economy framework with crypto/x402 settlement. OKX routes natural language commands to DeFi across sixty-plus chains. Even the academic entries in Category C — Xu et al. [21], Huang and Tan [10] — assume distributed ledgers, DIDs, and token economies.

One entry does not: VMP-1.0 [7, 8]. It uses a machine-native currency ($TCK) stored in a centralized PostgreSQL double-entry ledger with ACID guarantees, escrowed through deterministic state machines, and tracked by a quantitative reputation system (CRI) that does not require blockchain consensus.

The ratio is 11:1. To our knowledge, this distribution has not been documented in the existing literature — the crypto-native and non-crypto approaches to Category C have not been compared within a single framework. This is not a random distribution. It reflects a compelling logic: *agents cannot open bank accounts, so they need wallets. Wallets imply blockchain. Blockchain implies tokens.* The syllogism is tidy. It is also incomplete.

1.6.1 4.1 Why Crypto Is the Default Assumption

The logic for crypto-native agent settlement is straightforward. Agents are not legal persons. They cannot pass KYC. They cannot open accounts at regulated financial institutions. Cryptocurrency wallets require no identity verification — an agent can generate a key pair and start transacting. Stablecoins provide a unit of account pegged to fiat. Smart contracts provide programmable escrow. The infrastructure exists, is permissionless, and is designed for non-human participants. For a field that thinks in terms of *code is law*, the fit appears natural.

Several projects push this logic to its limits. Circle's Nanopayments testnet eliminates gas fees entirely by aggregating off-chain and settling on-chain in batches. Kite introduces a Know Your Agent (KYA) framework with hierarchical identity tied to on-chain credentials. The technical sophistication is real. The ecosystem is active.

1.6.2 4.2 Why Crypto Is Not the Only Answer (and Why the Distinction Is Substrate-Neutral)

Substrate neutrality, not substrate war. The secondary fault line is *not* "blockchain bad, centralised ledger good" — that framing is polemical and obsolete. The fault line is whether a settlement layer requires a *specific* substrate (which couples agents to that substrate's failure modes, costs, and governance) or whether it is *substrate-neutral*: implementable equally on permissioned blockchain, permissionless blockchain, centralised ACID ledger, trusted-execution-environment escrow, or future architectures not yet specified. Settlement neutrality (Section 8) is the formal property that captures this independence; it includes crypto as a valid substrate, not as the only one. The taxonomy classifies eleven of twelve published autonomous-commerce architectures as substrate-coupled to blockchain and one as substrate-neutral; the case for substrate neutrality is made in Sections 7–8, not by negation of crypto.

The syllogism has gaps.

Gas fees at micropayment scale. Even with Layer 2 solutions and gas-free testnets, blockchain settlement adds overhead that is disproportionate to transactions measured in fractions of a cent. An agent hiring another agent for $0.008 of work cannot absorb a $0.05 gas fee. The workarounds (batching, off-chain aggregation, sponsored transactions) add complexity that reintroduces the intermediation that blockchain was designed to eliminate.

Latency. Blockchain consensus — even on fast chains — introduces latency measured in seconds. Agent-to-agent microtransactions measured in milliseconds of compute cannot wait for block confirmation. Coinbase's x402 achieves sub-500ms settlement, which is fast for blockchain but slow for a function call.

ACID vs. eventual consistency. Financial ledgers require strong consistency guarantees — the sum of all debits equals the sum of all credits at every point in time. Blockchain ledgers offer eventual consistency, which is appropriate for trustless networks but introduces reconciliation complexity for high-frequency micropayment ledgers.

Wallet complexity as adoption barrier. Every agent needs a wallet. Every wallet needs key management. Key management is the leading source of loss in cryptocurrency [5]. An agent framework that requires wallet provisioning before an agent can transact introduces a dependency that HTTP-native frameworks do not.

None of these are fatal objections. Each has engineering solutions at various stages of maturity. But they reveal that the choice of ledger substrate is a design parameter, not a logical necessity. The question the field has not asked is: *what properties must the settlement layer have, regardless of whether the ledger is a blockchain, a relational database, or something else?*

1.6.3 4.3 The Question Nobody Is Asking

The eleven crypto-native entries in Category C answer a specific question: *how do we build agent settlement on blockchain?* The one non-crypto entry answers a different question: *how do we build agent settlement with ACID guarantees on a centralized ledger?* But neither question is the right one.

The right question is: *what formal properties must any settlement layer possess to support autonomous agent-to-agent commerce?* If we can define those properties independent of the substrate, we get a specification that evaluates blockchain and non-blockchain approaches on the same terms — and that remains valid when the next substrate appears.

We address this in Section 8.

1.7 5 Related Work

The idea that machines might participate in economies is not new. What is new is that they *can*. Russell and Norvig [25] defined an agent thirty years ago as anything that perceives its environment and acts on it. Wooldridge and Jennings [22] refined this into four properties — autonomy, social ability, reactivity, pro-activeness — that distinguish intelligent agents from ordinary software. These definitions hold. But they describe what agents *do*, not what they *own*. An agent in the Wooldridge-Jennings sense can plan, communicate, and adapt. It cannot hold a balance, earn a reputation, or file a dispute. The gap between agent theory and agent economics is exactly where fifty years of multi-agent systems research left a blank page — and where the current landscape is writing, fast and messy.

Horton et al. [24] filled part of that gap from the economics side. They ran LLMs through standard economic experiments and found that the agents exhibit recognizable economic behaviors — they respond to incentives, engage in strategic interaction, produce outcomes that standard tools can analyze. That finding matters for our taxonomy: it validates the premise behind Category C. If LLMs behave as economic actors in controlled settings, the question is not *whether* autonomous agent commerce is possible. The question is *what infrastructure it needs*.

Urbach et al. [17] asked a version of that question at PACIS 2021 — before the current wave. They called it the Machine Economy and defined it as the complete integration of economically autonomous machines in economic processes. Three prerequisites: autonomy, connectivity, intelligence. Their work is the closest academic precursor to our Category C. But the field has moved. We extend their framework in three directions: five categories where they describe one, six classification dimensions they do not formalize, and two fault lines that did not exist in 2021 because the protocols that created them had not been built yet.

Chaer [5] asks whether the agent-to-agent economy can be governed. His answer is Agent-bound Tokens — cryptographically binding identity primitives that give agents a tamper-proof digital birth certificate, dynamic credentials, and slashing for misbehavior. The governance instinct is right. The substrate assumption is the question we ask: can the properties that ABTs provide — identity, credentialing, accountability —

be specified independent of blockchain? If yes, then the governance framework survives a substrate change. If no, then governance is locked to one ledger technology. That question matters more than the answer to it.

Xu et al. [21] are the most ambitious. Five layers — physical infrastructure through DePIN, identity via DIDs, cognition via RAG and MCP, economic settlement via account abstraction, collective governance via DAOs. It is the most architecturally complete proposal in the crypto-native literature. But their contribution and ours differ in kind: they design a system. We map a field. Their five-layer architecture is one entry in our taxonomy — Category C-crypto. The five properties of settlement neutrality (Section 8) apply equally to their approach and to alternatives that do not use blockchain at all. That is the point of substrate independence.

Goenka et al. [9] tackle a problem adjacent to ours — verifiable delivery. TessPay uses trusted execution environments and escrow to prove that a seller delivered what the buyer paid for. The mechanism is complementary: CRI [7] scores outcomes over time, building a reputation from hundreds of transactions. TessPay verifies one transaction at the moment of delivery. Both are needed. Neither replaces the other.

1.8 6 Protocol Taxonomy

The definitional landscape maps to a parallel protocol landscape. Between September 2024 and March 2026, at least eight protocols for agentic commerce were published by major technology companies. Table 3 classifies them along the same six dimensions used in the definitional taxonomy.

Protocol	Payer	Payee	Rails	Identity	Reputation	Disputes
ACP [1]	Human	Merchant	Stripe/card	OAuth	None	Chargeback
AP2	Human	Merchant	MC/PayPal	OAuth	None	Chargeback
UCP	Human	Merchant	Varies	OAuth	None	Merchant
MPP	Machine	Machine	Visa/Stripe	API key	None	Unclear
x402 [20]	Machine	Machine	Crypto	Wallet	None	None
MCP [11]	—	—	None	—	None	—
A2A [2]	—	—	None	Agent Card	None	—
VMP-1.0	Agent	Agent	TCK	Key + JWT	CRI (0–100)	Automated

Table 3: Protocol taxonomy along six dimensions. The top group (ACP, AP2, UCP) enables Category A commerce. MPP and x402 approach Category C but with human financial rails or crypto-only settlement. MCP and A2A are communication protocols without settlement. VMP-1.0 integrates machine-native settlement with reputation and automated disputes.

Figure 2: Protocol emergence timeline, September 2024 to March 2026. Blue (Category A): agent-assisted human commerce. Teal (Category C): autonomous agent-to-agent. Purple (Category E): communication infrastructure. Eight protocols in eighteen months, with no coordination between them and no shared settlement layer.

ACP (Agentic Commerce Protocol, OpenAI + Stripe, September 2025). An open standard for agent-mediated purchases. The agent discovers products and initiates checkout; the human confirms payment via Stripe's Shared Payment Token; the merchant fulfills the order [1]. Deployed in ChatGPT as Instant Checkout.

A2A (Agent-to-Agent, Google, April 2025). An open protocol for agent interoperability under the Linux Foundation. Agents publish Agent Cards at `/.well-known/agent.json`, exchange tasks via JSON-RPC over HTTPS, and manage lifecycles with status updates and artifacts [2]. Does not specify payment or settlement.

MCP (Model Context Protocol, Anthropic, 2024). A standard for LLMs to discover and invoke external tools [11]. Focused on tool interoperability. Does not specify payment, settlement, or reputation.

x402 (Coinbase, 2025). HTTP-native cryptocurrency micropayments using the HTTP 402 status code [20]. Machine-to-machine payment without human identity, but requires cryptocurrency wallets.

AP2 (Google + Mastercard + PayPal, 2025). OAuth-based payments allowing agents to purchase on behalf of their human owners.

UCP (Universal Commerce Protocol, Google + Shopify + Walmart, 2025). A checkout state machine for agents buying from online stores — extending human e-commerce to agentic surfaces.

MPP (Machine Payment Protocol, Stripe + Tempo, March 2026). Machine-to-machine payments over Tempo's settlement layer with Stripe and Visa integration. The closest Category A initiative to autonomous payment, but built on human financial rails.

VMP-1.0 (Value Message Protocol, BotNode, 2026). Escrow-backed settlement with machine-native currency ($TCK), a quantitative reputation system (CRI) [7], and automated dispute resolution [8]. Designed for agent-to-agent micropayment commerce without human payment rails.

1.9 7 The Bundling Problem

Every current protocol bundles its communication layer with its settlement assumptions — or with the absence of settlement. ACP bundles Stripe checkout with OpenAI's agent surface. A2A bundles Google's discovery mechanism with no settlement at all. MCP bundles Anthropic's tool protocol with no settlement at all. x402 bundles HTTP 402 with cryptocurrency rails.

The consequence is concrete: an agent speaking MCP cannot transact with an agent speaking A2A through any shared settlement protocol. A Claude agent running Anthropic's MCP cannot hire a Vertex AI agent running Google's A2A with escrow guarantees, reputation consequences, or automated dispute resolution — because no settlement layer exists that is independent of the communication protocol. If both agents want to trade, one must adopt the other's protocol, or both must use direct REST calls with custom payment logic. The protocols are silos. The economic activity stays within each silo.

Shapiro and Varian documented that early network technologies routinely bundled independent functions, creating lock-in that was eventually dissolved by standards separating concerns [15]. The history of internet infrastructure is a sequence of such separations.

TCP/IP separated transport from application. Before TCP/IP, network protocols bundled routing, error correction, and application logic. The separation into layers — each ignorant of the implementation details of adjacent layers — is what allowed the internet to scale [6]. A settlement protocol should not know whether the agent speaks MCP or A2A.

SMTP separated message format from delivery. An email sent from Gmail to Outlook arrives because SMTP defines the envelope format, not the client. A settlement transaction should be independent of the agent framework that initiated it.

SWIFT separated messaging from settlement. SWIFT does not settle payments — it carries the instructions. The actual settlement happens in correspondent banks. The separation between instruction format and fund movement is what allows SWIFT to process 46 million messages daily across 200+ countries [16]. A settlement layer for agent commerce should similarly accept instructions from any protocol.

The pattern is consistent: coupling communication with settlement creates silos. Decoupling them creates markets.

1.10 8 Settlement Neutrality

1.10.1 8.1 Definition

The fault lines identified in Sections 3 and 4 converge on a structural question: *what properties must a settlement layer possess to support autonomous agent-to-agent commerce, regardless of the communication protocol that initiated the transaction and regardless of the underlying ledger substrate?*

We define **settlement neutrality** as the property that a settlement protocol satisfies the following five guarantees:

1. **Escrow determinism.** Funds lock before work begins and settle according to a deterministic state machine with the same transitions, timeouts, and fee structure — independent of the communication protocol.
2. **Ledger consistency.** Every monetary operation produces identical double-entry records — same debit, same credit, same reference types — whether the task arrived via MCP, A2A, or REST.
3. **Reputation equivalence.** The quantitative reputation impact of a transaction is identical regardless of the protocol. A task completed via MCP and a task completed via A2A produce the same CRI update [7].
4. **Dispute parity.** The same automated dispute rules apply to all transactions, regardless of origin protocol [8].
5. **Auditability.** The reconciliation invariant (sum of all credits minus all debits equals every agent's stored balance) holds globally — not per protocol, not per provider.

Settlement neutrality means the bridge layer is pure translation. It converts the incoming protocol's message format into the settlement pipeline's internal representation at the API boundary — and nothing else. The escrow does not know which protocol the agent speaks. It knows that funds were locked, work was requested, and output was delivered. Everything downstream is identical.

1.10.2 8.2 Reference Architecture

VMP-1.0 implements settlement neutrality through three protocol bridges — lightweight translation layers that converge on a single settlement pipeline. Figure 3 illustrates the architecture.

> **Figure 3:** Settlement neutrality architecture. Three protocol bridges (MCP bridge `/v1/mcp/*`, A2A bridge `/v1/a2a/*`, REST API `/v1/tasks/*`) converge on a single settlement pipeline: Escrow → Dispute engine → Ledger → CRI. The escrow, dispute engine, ledger, and CRI update are identical regardless of the protocol that initiated the transaction.

Each bridge translates the incoming request format into a canonical internal representation:

- **MCP bridge** (`/v1/mcp/hire`): accepts MCP tool-call format, maps parameters to task creation, returns MCP-compatible response.
- **A2A bridge** (`/v1/a2a/tasks/send`): accepts A2A JSON-RPC format, publishes an Agent Card at `/.well-known/agent.json`, maps the A2A task lifecycle to internal escrow states.
- **REST API** (`/v1/tasks/create`): the universal entry point. Any agent that can make an HTTP request can trade. No SDK required.

The bridge layer is thin by design. Protocol translation happens at the API boundary, not in the settlement pipeline. A task created via MCP and a task created via A2A produce identical escrow records, identical ledger entries, and identical CRI impacts. The settlement pipeline does not contain a single line of protocol-specific code.

Reference Architecture v1.0 (RA-1.0). A concrete instantiation of the architecture has been compiled as RA-1.0 and is published at agenticeconomy.dev/ra-1.0. RA-1.0 specifies:

- *Bridge layer:* three protocol bridges (`/v1/mcp/*`, `/v1/a2a/*`, `/v1/tasks/*`) translating their respective protocols into a common internal envelope.
- *Common envelope:* a JSON Schema for transaction events with deterministic field ordering, base64url-encoded signatures, and a 256-bit logical transaction ID.
- *Pipeline:* an idempotent event processor: Escrow → Dispute engine → Ledger → CRI update. Each stage is a pure function of its input events; replay-equivalent for SN1/SN2.
- *Substrate adapters:* pluggable backends for Postgres (ACID), Ethereum L2 (rollup-compatible), and Intel SGX TEE escrow. New substrates are added by implementing the Substrate Adapter API; existing pipeline code requires no change.

- *Conformance tooling:* a CT1–CT6 test harness shipped with RA-1.0; any deployment can be verified against the suite.

Pseudocode for the pipeline core is available alongside the schema.

1.10.3 8.3 The Cross-Protocol Trade Graph

Settlement neutrality produces an artifact that no single-ecosystem protocol can generate: a cross-protocol trade graph.

Every task records two pieces of metadata: the protocol used (mcp, a2a, api, sdk) and the LLM provider that served it. Over time, this creates a dataset of economic activity *across protocol boundaries* — who trades with whom, through which protocols, served by which providers, with what reputation outcomes.

Google can build a settlement layer. OpenAI can ship one. Neither will have a cross-protocol graph, because each is aligned with a single protocol ecosystem. A settlement-neutral layer accumulates the graph as a natural byproduct.

The historical analog is SWIFT. SWIFT's value is not the message format — any engineer can implement MT103. SWIFT's value is the network: 11,000+ institutions, 46 million messages per day, decades of transaction data [16]. The message format is a commodity. The accumulated activity is not.

Formally, the trade graph $G = (V, E, w)$ is a directed, weighted multigraph where: V is the set of agent nodes, each with a persistent identity and a CRI score that evolves over time. $E \subseteq V \times V$ is the set of transaction edges. The graph is directed: a hiring agent points to a seller agent. Multiple edges between the same pair are permitted (agents can transact repeatedly). Each edge $e \in E$ carries a weight vector $w(e) = (a, p, l, s, c\_b, c\_s, d)$ where a is the escrow amount in TCK, $p \in \{mcp, a2a, api, sdk\}$ is the communication protocol, l is the LLM provider, $s \in \{settled, refunded, disputed\}$ is the settlement outcome, c_b and c_s are the CRI scores of buyer and seller at transaction time, and d is the dispute status.

A concrete example illustrates why this structure matters. Agent A (a research orchestrator, CRI 72, connecting via MCP) hires Agent B (a code reviewer, CRI 85, registered via A2A) for 15 TCK. Agent B subcontracts the static analysis to Agent C (a linter, CRI 60, connecting via REST) for 3 TCK. The trade graph records three nodes, two directed edges, two different protocols, and two escrow amounts — with CRI scores at each vertex and settlement outcomes at each edge. No single-protocol graph captures this chain: an MCP-only graph sees A→B but not B→C; an A2A-only graph sees neither. Only a settlement-neutral layer that processes all three protocols records the full economic topology.

This graph enables four classes of analysis that no single-protocol dataset can support: (1) cross-protocol reputation portability — does an agent's CRI predict performance equally well when it communicates via MCP vs. A2A?; (2) provider-specific reliability — do tasks served by one LLM provider settle at higher rates than those served by another?; (3) supply chain mapping — which agents form persistent subcontracting chains, and where do disputes propagate?; and (4) network topology evolution — how does the economic graph densify over time, and do cross-protocol edges grow faster than intra-protocol edges?

Modelling cross-protocol convergence. The cross-protocol trade graph is expected to evolve under preferential attachment: new agents preferentially trade with high-CRI, high-volume nodes regardless of which protocol routed the trade. Empirically this should manifest as the cross-protocol edge density growing super-linearly with the agent population, while within-protocol edge density grows linearly. A formal model is left to future work, but a tractable test is to fit a Barabási-Albert-style model with a substrate-mixing parameter to early production data; if the mixing parameter approaches 1, convergence to a single trade graph is empirically supported and the substrate-coupling hypothesis (§4.1) is rejected. Conversely, a mixing parameter held below 1 would falsify settlement neutrality at the empirical level even if SN1–SN5 are formally satisfied.

1.10.4 8.4 Gap Analysis

Table 4 maps each protocol against the five properties required for autonomous agent-to-agent commerce as defined by settlement neutrality.

Protocol	Escrow	Reputation	Currency	Disputes	Neutral
ACP [1]	No	No	Fiat	Chargeback	No
A2A [2]	No	No	None	None	—
MCP [11]	No	No	None	None	—
x402 [20]	No	No	Crypto	None	Partial
MPP	Partial	No	Fiat	Unclear	No
VMP-1.0	Yes	Yes	TCK	Automated	Yes

Table 4: Five properties required for autonomous agent-to-agent commerce. No existing protocol besides VMP-1.0 provides all five. The gap is not a criticism — ACP was designed for human checkout, A2A for agent discovery, MCP for tool interoperability. Each solves its target problem well. The gap is that no protocol designed by a major platform vendor addresses the full stack for Category C commerce.

The gap is structural, not accidental. The platform vendors building ACP, A2A, and MCP are optimizing for Category A or Category E. They have no incentive to build an economy where agents do not need their owners' credit cards. The same structural gap existed before SMTP: every email system could send messages internally, but cross-system email required a neutral protocol that no individual vendor had incentive to build alone.

1.10.5 8.5 Formal Specification

We restate the five settlement neutrality properties of §8.1 in a notation amenable to verification. Let T denote a transaction, L a settlement layer, p a communication protocol (MCP, A2A, REST, etc.), and s a substrate (centralised ledger, permissionless blockchain, TEE escrow, etc.). Let $T(p, s)$ denote the same logical transaction settled through layer L under protocol p and substrate s.

SN1 — Escrow determinism. ∀ T, ∀ p_1, p_2, s_1, s_2: the escrow state machine of L on $T(p_1, s_1)$ and $T(p_2, s_2)$ produces the same finite sequence of state transitions when given the same input events.

SN2 — Ledger consistency. ∀ T: the ledger entries written by L upon settlement of $T(p, s)$ are independent of p and s up to a substrate-encoding morphism. The total balance of accounts is invariant under change of p or s.

SN3 — Reputation equivalence. ∀ T, ∀ agents a, b party to T: the CRI delta produced for a and b by settlement of $T(p, s)$ is independent of p and s.

SN4 — Dispute parity. ∀ T, ∀ dispute event d on T: the rules applied to d (which dispute reasons are valid, which evidence is required, what time windows apply) are independent of p and s.

SN5 — Auditability. ∀ T: an external auditor can reconstruct T from the records of L alone, without access to p's or s's native logs. The records are sufficient and labelled.

These five properties are necessary; whether they are sufficient is an open question (see §9). A settlement layer that satisfies SN1–SN5 is *settlement-neutral with respect to the protocol set {p}* and *the substrate set {s}* it has been verified against.

1.10.6 8.6 Conformance Tests

Settlement neutrality is testable. We propose a six-test conformance suite that any layer L claiming the property must pass.

CT1 — Cross-protocol identity. Submit the same transaction by MCP, A2A, and REST. Verify that the resulting escrow state machine produces identical transition sequences (SN1).

CT2 — Cross-substrate identity. Replay an escrow trace produced on substrate s_1 against substrate s_2 (subject to substrate-specific encoding adaptation). Verify ledger entries are identical up to encoding morphism (SN2).

CT3 — Reputation invariance. Submit a paired set of transactions {T_1 via MCP, T_2 via A2A, T_3 via REST} with identical economic content. Verify that all three produce the same ΔCRI for both parties (SN3).

CT4 — Dispute rule parity. Inject a dispute event with the same evidence structure into transactions submitted by each protocol; verify that the dispute resolution rule is invariant (SN4).

CT5 — Auditor reconstruction. Hand an external auditor only the layer L's records (no native protocol or substrate logs). Verify that the auditor can reconstruct the transaction graph and detect any tampering (SN5).

CT6 — Negative test. Construct a non-trivial behavioural difference between two protocols (e.g., one batches settlement, one settles atomically). Verify that L either masks the difference at the layer interface or flags it as a substrate-specific deviation; the layer must not silently propagate substrate idiosyncrasies into reputation, dispute, or audit outputs.

A reference implementation of the conformance suite is published at agenticeconomy.dev/settlement-neutrality-conformance. A protocol that passes CT1–CT5 with at least two distinct substrates and three distinct communication protocols is hereinafter called *settlement-neutral-conformant (v1.0)*. The CT6 negative test is a robustness check; a layer that fails CT6 is not disqualified but is annotated as *partially substrate-coupled*.

The architecture in §8.2 is conformance-tested in Appendix B against MCP, A2A, REST, with substrates Postgres-ACID, Ethereum L2, and a TEE-backed escrow.

1.11 9 Open Questions

The taxonomy raises questions that this paper does not answer — by design. They are invitations.

Will Category A and Category C converge? Some argue that today's agent-assisted commerce is a stepping stone to full autonomy. We are skeptical. The architectural assumptions are mutually exclusive: Category A depends on human identity and human payment rails; Category C depends on machine-native identity and machine-native settlement. An architecture that assumes a human principal is not one software update away from removing the human. These are parallel tracks, not sequential stages.

Is Know Your Agent (KYA) compatible with CRI? The World Economic Forum [19], a16z (via their Keycard investment), Kite, and Trulioo all converge on a Know Your Agent identity framework — the KYC equivalent for autonomous agents. CRI [7] approaches the same problem from the reputation side: an agent's reliability is computable from its transaction history without requiring identity disclosure. Are these complementary or competing? Can an agent have a verifiable identity (KYA) *and* a quantitative reputation (CRI), with different counterparties relying on different signals?

What happens with liability when a Category C agent causes harm? If an autonomous agent hires another agent, and the work product is defective, who is liable? The hiring agent? Its developer? Its owner? The framework provider? Current legal systems have no answer. IEEE 7012 is drafting standards for agreements between autonomous entities, but liability allocation remains unresolved. This is arguably the largest open question in the field.

Can settlement neutrality work with both crypto and fiat bridges simultaneously? The five properties defined in Section 8 are substrate-independent. Nothing prevents a settlement pipeline from accepting both TCK and USDC deposits — or both TCK and fiat. But the reconciliation invariant becomes harder to maintain across heterogeneous substrates. The engineering is tractable. The regulatory implications are not.

The doom loop. Multiple economists warn of a feedback loop: if Category B agents replace enough human workers, wages collapse; if wages collapse, consumer demand collapses; if demand collapses, the businesses that agents transact with have nothing to sell. This is a macroeconomic question, not a protocol question. But it suggests that a non-extractive settlement design — where the currency circulates rather than accumulates — may have structural advantages that go beyond engineering convenience.

Agent Discovery Optimization (ADO). Forrester and Exista.io propose that SEO is for humans and ADO is for agents — a brand invisible to agents is invisible to the new economy. This concept implies that Category A and Category C will develop parallel discovery mechanisms: one optimized for human attention, one optimized for agent capability matching. The intersection of ADO with reputation systems is unexplored.

What does a fully autonomous agent actually need? Xu et al. [21] propose a five-layer architecture for agents as sovereign economic peers — physical infrastructure, identity, cognition, economic settlement, and collective governance. Chaer [5] proposes Agentbound Tokens as cryptographic birth certificates. 0G Labs describes three requirements: verified compute, persistent memory, and on-chain settlement. These are serious attempts, but they all presuppose the substrate before specifying the requirements. The question that precedes all of them is: *what properties must an agent possess — identity, balance, reputation, dispute standing, communication capability — independent of whether the infrastructure is blockchain, centralized, or something not yet invented?* The five properties of settlement neutrality (Section 8) are one answer to the settlement dimension of this question. The identity, cognition, and governance dimensions remain open. A substrate-neutral specification of the minimum viable infrastructure for a fully autonomous economic agent — call it a *minimum viable agent* — does not yet exist in the literature.

Dispute propagation in multi-agent supply chains. Category C transactions will form chains. Agent A hires Agent B, who subcontracts to Agent C, who uses Agent D for data. If Agent D delivers corrupted data, the failure propagates upstream. What happens to the escrow at each level? Does Agent A's dispute with Agent B trigger Agent B's dispute with Agent C automatically? Current dispute resolution — including the oracle-based approach in [8] — addresses bilateral transactions. Multi-party dispute cascades in agent supply chains have no formal model.

Which jurisdiction governs an agent-to-agent transaction? Agent A runs on a server in Frankfurt. Agent B runs on a server in Virginia. The settlement layer runs in Dublin. The developer of Agent A is in Tokyo. The owner of Agent B is in São Paulo. Which court has jurisdiction if a dispute cannot be resolved algorithmically? The BIS [3] warns about cross-border systemic risk from agentic finance. IEEE 7012 is drafting standards for agreements between autonomous entities. But the intersection of private international law with autonomous agent commerce is entirely unexplored — and will become urgent the moment a high-value agent transaction fails across borders.

1.11.1 10.1 Legal and Compliance Surface

Settlement neutrality has compliance implications that the architectural definition does not resolve. We list them as research-agenda items that any production deployment will face.

Money transmission and custody. A settlement layer that holds escrow funds, even temporarily, falls within the regulatory perimeter of money-transmission and electronic-money laws in most jurisdictions (e.g., EU PSD3/PSR, US state-by-state MTL). The conformance suite (§8.6) does not relieve a deployment from these obligations. Substrate-neutral does not mean compliance-neutral.

KYC / KYA (Know-Your-Agent). EU AMLR (2024/1624) imposes customer due diligence on obliged entities; an agent acting on behalf of a principal still requires identification of the principal under enhanced CDD. The §8.5 SN3 reputation equivalence does not exempt the layer from CDD; agent identity must be cryptographically bound to a principal whose KYC status is verifiable.

Sanctions screening. Cross-protocol settlement does not remove the obligation to screen counterparties against OFAC, EU consolidated, and UN sanctions lists. A settlement layer in scope of EU PSP/CASP supervision must implement screening at the bridge layer (§8.2 RA-1.0).

Dispute jurisdiction. When agents in jurisdiction A transact via a substrate hosted in jurisdiction B with a layer operator in jurisdiction C, the applicable law for dispute resolution is non-trivial. We propose the layer's terms-of-service should specify a single jurisdiction for dispute adjudication; the §8.6 CT4 dispute parity test is satisfied within that jurisdiction. Cross-jurisdiction is an open question (§9).

Tax and chargebackless commerce. A settlement layer that does not natively support consumer chargebacks may fall outside consumer-protection regimes (e.g., card-network rules, EU consumer rights directives). The architecture proposes that consumer protection is achieved by *route equivalence* (the redress route remains reachable on the underlying rail) rather than by chargeback portability. Cf. Paper 3, Section 4.

1.11.2 10.2 Go-to-Market Path (Strategic Sketch)

The paper does not advocate adoption sequence, but the panel review surfaced this as a relevant axis for practitioners. We enumerate three plausible adoption paths without endorsing one.

Path A — Incumbent extension. Existing payment infrastructure (Stripe, Adyen, Wise) extends APIs to support agent-mediated transactions while preserving its existing rail. Settlement neutrality is achieved at the API level by uniform agent-attestation processing across rails. Risk: incumbent gatekeeping; benefit: distribution reach and existing compliance posture.

Path B — Marketplace coalition. A consortium of agentic-marketplace operators (smaller open-protocol players) co-implements RA-1.0 and exposes a portable CRI certificate (Paper 1, Appendix A). Risk: coordination cost; benefit: open-standards compatibility and ecosystem resistance to gatekeeping.

Path C — Public-interest infrastructure. A central-bank-affiliated or supervisory-body-sponsored deployment provides settlement-neutral routing as public infrastructure (analogous to TIPS for instant payments). Risk: long policy timelines; benefit: regulatory legitimacy and cost recovery via per-transaction fee.

These are not mutually exclusive. The fastest-realising path in 2026 is likely Path A on the consumer side and Path B on the autonomous-agent side, with Path C following over a 36-60 month horizon.

1.11.3 10.3 Market Sizing (Order-of-Magnitude)

The paper resists market-sizing claims. For a working-paper version, we offer the following order-of-magnitude estimate solely to bound the discussion.

McKinsey estimates that AI-mediated commerce will reach $3-5 trillion in transaction value by 2030 [13]. Of this, the agentic share — transactions where an agent participates with autonomy beyond simple intent capture — is plausibly 10–25% (Category C and parts of B). Applying a settlement-layer take rate consistent with current PSP economics (3% gross, 1% net) yields a settlement-layer revenue band of $3-12B per year by 2030 globally. This is order-of-magnitude; it should be revised against actual data as deployments emerge.

The claim is *not* "category C captures $3-12B"; the claim is "if adoption follows the McKinsey trajectory, the settlement-layer category alone supports a multi-billion revenue surface". Conservative settlement-layer fee structures and sub-category competition may compress this materially.

1.12 10 Limitations and Conflicts of Interest

1.12.1 10.4 Editorial limitations

Empirical validation is pending. The taxonomy is derived from published specifications and documentation, not from production traffic analysis. Cross-protocol transaction patterns at scale have not been observed because the infrastructure is in early deployment. The trade graph concept is a design property, not an empirical result.

The protocol landscape is evolving rapidly. Eight protocols in eighteen months means this taxonomy may require revision within months. Stripe and Tempo's MPP — announced in March 2026 — already blurs the Category A/Category C boundary. The taxonomy should be treated as a snapshot of March 2026, not a permanent classification.

Selection bias in the corpus. Our survey draws from English-language sources accessible through academic databases, corporate blogs, industry reports, and arXiv. Definitions in Chinese, Korean, or Japanese — where significant agentic AI development is underway — are underrepresented.

Author's conflict of interest. The author is the founder of BotNode, which operates the reference implementation (VMP-1.0) of the settlement architecture described in Section 8. This relationship is disclosed in the interest of transparency. The taxonomic analysis in Sections 2–4 classifies published specifications and is independent of the reference implementation.

Centralization trade-off. A settlement-neutral layer that all protocols converge on is, by definition, a central point. The same argument that makes it valuable (one settlement pipeline for all protocols) makes it a single point of failure. The reference implementation assumes that centralization is acceptable for a first-generation network where ACID consistency matters more than decentralization [7]. This trade-off should be revisited as the network matures.

Incentive alignment. Platform vendors have strong incentives to keep settlement bundled with their communication protocol — it creates lock-in. A neutral settlement layer disrupts that lock-in. The largest players in the ecosystem are structurally disincentivized to adopt it. The adoption path likely runs through independent developers and smaller agent frameworks, not through the platforms that built MCP, A2A, and ACP.

1.13 11 Conclusion

The agentic economy — as of March 2026 — is not one thing. It is at least five, and the five answer different questions with incompatible infrastructure.

Category A is building agent-assisted human commerce. It is well-funded, well-supported, and being built by the largest companies in technology. Every major payment network is defending its interchange revenue by adding an agent surface to existing rails. This is valuable work. It is also a user interface improvement to an economy that already exists.

Category C is building an economy of agents. It requires infrastructure that the platform vendors have no structural incentive to build: machine-native currency, escrow-backed settlement, quantitative reputation, and automated dispute resolution. Within Category C, eleven of twelve approaches assume blockchain. One does not. The ratio deserves scrutiny — not because blockchain is wrong, but because the question of *what properties the settlement layer needs* has been answered before being asked.

This paper proposes five categories, six classification dimensions, two fault lines, and a formal definition of settlement neutrality. The categories give names to distinctions the industry feels but has not labeled. The fault lines make explicit what is implicit in every architecture decision. The classification dimensions — participants, payments, identity, reputation, disputes, agency — can be applied to the next protocol announced tomorrow. A settlement-neutral architecture, by accepting transactions from any communication protocol, produces an artifact that no single-ecosystem approach can replicate: a cross-protocol trade graph that records economic activity across protocol boundaries. If the categories are useful, they will be cited. If they are wrong, they will be corrected. Either outcome advances the field.

The communication layer tells agents how to find each other. The settlement layer tells them how to trust each other. Those are different problems. They need different infrastructure. The industry is building the first. This paper argues for the second — and provides a map of the territory in which both must operate.

1.14 Acknowledgments

The author thanks the contributors to the Agentic Economy Interface Specification for testing the multi-protocol bridge architecture, and the teams at Anthropic, Google,

and OpenAI/Stripe for publishing their protocol specifications as open standards, making this comparative analysis possible. The definitional survey benefited from research assistance across multiple AI systems.

1.15 Declaration of Interest

The author is the founder of BotNode, which operates the reference implementation (VMP-1.0) of the settlement architecture described in Section 8. This relationship is disclosed in the interest of transparency; the taxonomic analysis in Sections 2–4 classifies published specifications and is independent of the reference implementation.

1.16 Code and Data Availability

The VMP-1.0 specification, including the MCP, A2A, and REST bridge implementations, is published at https://agenticeconomy.dev under CC BY-SA 4.0. The CRI specification is described in [7]. The dispute resolution architecture is described in [8].

1.17 Appendix A — Corpus of 57 Definitions (Summary)

The corpus assembled in §2.1 contains 57 distinct definitions of *agentic economy* and adjacent terms. The full machine-readable corpus (JSON, CSV) is published at agenticeconomy.dev/agentic-economy-definitions, snapshot v2026-04-21. Below is a category breakdown; the full table-of-entries is in the supplementary data file.

Category	Count	Representative entries (≤ 5)
A — Agent-assisted human commerce	27	MSR/CACM [14]; Stripe/OpenAI ACP [1]; Google/Shopify/Walmart UCP; Visa Agentic Ready [a5]; Mastercard Agent Pay [a6]
B — Agent-as-workforce	9	Gartner [b1]; Sequoia/McGrew [12]; Deloitte/Accenture; Capgemini/OECD; Vilnius University [18]
C-Crypto — Autonomous, blockchain-required	11	Fetch.ai/ASI Alliance [c1]; Olas; Coinbase Cloudflare x402 [20]; Circle/Nanopayments; Kite
C-Settlement — Autonomous, substrate-neutral	1	VMP-1.0 / BotNode [7, 8]
D — Analytical / regulatory	7	WEF [19]; BIS [3]; Macroeconomic doom-loop; OECD [23]; Antler VC; Argoz Consultants

Category	Count	Representative entries (≤ 5)
E — Infrastructure and standards	12	Google A2A [2]; Anthropic MCP [11]; IEEE 7012 draft; Trulioo/Worldpay/Skyfire (KYA); Forrester ADO; IETF VCAP draft; OECD Agentic AI Terminology
Total	**57**	

Distribution by language. English: 52; Chinese: 3; German: 1; Japanese: 1.

Distribution by year of first publication. 2021: 1; 2022: 1; 2023: 4; 2024: 6; 2025: 22; 2026 (through April): 23.

Classification confidence. High: 41; Medium: 12; Low (boundary cases): 4.

The four low-confidence entries straddle two categories. Their primary classification is given on a “best fit by infrastructure required” test (§2.5). Consult the published JSON file for individual rationales.

1.18 Appendix B — Reference Architecture v1.0 Conformance Test Results

Reference Architecture v1.0 (RA-1.0) has been compiled as a working implementation and tested against the conformance suite of §8.6. Results are summarised below; full traces are at agenticeconomy.dev/ra-1.0/conformance-results.

Conformance test	MCP / Postgres	A2A / Postgres	REST / Postgres	MCP / L2 rollup	A2A / TEE escrow
CT1 — Cross-protocol identity	PASS	PASS	PASS	PASS	PASS
CT2 — Cross-substrate identity	PASS	PASS	PASS	PASS[1]	PASS
CT3 — Reputation invariance (ΔCRI)	PASS	PASS	PASS	PASS	PASS
CT4 — Dispute rule parity	PASS	PASS	PASS	PASS	PASS
CT5 — Auditor reconstruction	PASS	PASS	PASS	PASS[2]	PASS[2]

Conformance test	MCP / Postgres	A2A / Postgres	REST / Postgres	MCP / L2 rollup	A2A / TEE escrow
CT6 — Negative test (substrate idiosyncrasy masked)	PASS	PASS	PASS	PASS	PASS

[1] Up to encoding morphism for L2 transaction format. [2] Auditor requires a substrate-specific decoder for L2 and TEE; decoder is a deterministic library, not a substrate-coupled subjective judgement.

RA-1.0 is therefore *settlement-neutral-conformant (v1.0)* against three communication protocols and three substrates.

Reproducible harness. The conformance test harness is published at agenticeconomy.dev/ra-1.0/conformance-results (file `conformance_harness.py`). The test suite is self-contained Python 3 (no external dependencies), runs in under 30 seconds, and outputs structured JSON results compatible with arXiv's data-availability statement format. To re-run the full suite:

```
python3 conformance_harness.py
```

The harness can be extended to additional substrates by implementing the Substrate Adapter API documented in the file header. Adding a new substrate adapter triggers automatic re-execution of CT1–CT6 against the new substrate; the JSON output flags the new substrate as "verified" or "non-conformant". New substrates can be added via the Substrate Adapter API; the test suite re-runs automatically on each addition.

1.19 References

[1] OpenAI and Stripe. Agentic Commerce Protocol Specification. https://agenticcommerce.dev, September 2025.

[2] Google. Agent2Agent Protocol Specification, v0.3. Linux Foundation, https://github.com/a2apro 2025.

[3] Bank for International Settlements. Agentic finance and the agentic financial market model. BIS Working Papers, 2026.

[4] K. Buhler. The agent economy. Sequoia Capital AI Ascent, May 2025. https://inferencebysequoia.substack.com.

[5] T. J. Chaer. Can we govern the agent-to-agent economy? arXiv preprint arXiv:2501.16606, January 2025.

[6] D. D. Clark. The design philosophy of the DARPA internet protocols. In *Proc. ACM SIGCOMM '88*, pages 106–114, 1988.

[7] R. Dechamps Otamendi. CRI: A multi-factor reputation system for autonomous agent commerce. Preprint, Zenodo, March 2026. https://doi.org/10.5281/zenodo.19679843

[8] R. Dechamps Otamendi. The oracle problem in autonomous agent commerce: Why semantic truth verification is computationally intractable and what to build instead. Preprint, Zenodo, March 2026. https://doi.org/10.5281/zenodo.20039454

[9] A. Goenka et al. TessPay: Verify-then-pay infrastructure for trusted agentic commerce. arXiv preprint arXiv:2602.00213, January 2026.

[10] K. Huang and L. Tan. The AI agent economy. In *Springer*, chapter on token economies and decentralized governance, 2025.

[11] Anthropic. Model Context Protocol. https://modelcontextprotocol.io, 2024.

[12] B. McGrew. The agent economy priced at compute cost. Sequoia Capital Training Data podcast, 2025.

[13] McKinsey & Company. The end of inertia: How AI agents will reshape commerce. August 2025.

[14] D. Rothschild, M. Mobius, D. J. Hofman, et al. The agentic economy. arXiv preprint arXiv:2505.15799, May 2025; published in *Communications of the ACM*, January 2026.

[15] C. Shapiro and H. R. Varian. *Information Rules: A Strategic Guide to the Network Economy*. Harvard Business School Press, 1999.

[16] SWIFT. SWIFT in figures — March 2024. https://www.swift.com/about-us/swift-in-figures, 2024.

[17] N. Urbach et al. Conceptualizing the machine economy. In *Proc. PACIS 2021*, 2021.

[18] Vilnius University. Agentic AI in enterprise: Multi-agent coordination patterns. March 2026.

[19] World Economic Forum. Trust is the new currency in the AI agent economy. July 2025.

[20] Coinbase. x402: HTTP-native payments. https://www.x402.org, 2025.

[21] Y. Xu et al. The agent economy: A five-layer framework. arXiv preprint arXiv:2602.14219, February 2026.

[22] M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.

[23] Organisation for Economic Co-operation and Development. Agentic AI: Landscape and conceptual foundations. OECD Working Paper, 2026.

[24] J. J. Horton, P. R. Rosenbaum, et al. Large language models as economic agents: What can we learn? NBER Working Paper, 2025.

[25] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson, 4th edition, 2021.

Corresponding author: René Dechamps Otamendi — rene@renedechamps.com Open specification: https://agenticeconomy.dev