Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

Expressible, Advisory, or Unenforceable: A Conformance Analysis of Delegated Financial Authority in Deployed Agent-Payment Protocols

Ian Staley

Independent Researcher, Seattle, Washington, United States
ORCID: 0009-0000-8592-3186, E-mail: ian.t.staley@gmail.com

Abstract: Payment networks and model providers deployed agent-authorization infrastructure at speed during 2025 and 2026: signed mandates, agent-bound tokens, and machine-payable settlement rails, each promising that an autonomous agent transacts only within authority its principal granted. This paper asks a prior question to whether agents obey such authority: whether the deployed protocols can express it at all. We define an authorization envelope of eight fields drawn from the delegated-authority literature and from the control primitives of existing payment rails, comprising a per-transaction ceiling, a cumulative ceiling, a merchant set, a category set, required product attributes, a validity window, a substitution policy and an amount-valued confirmation threshold. We then code eight deployed agent-payment protocols against these fields using an auditable document-analysis protocol, classifying each field as expressible, advisory or absent according to whether a typed schema field exists and whether any identified party validates it. Three fields are unsupported almost everywhere: substitution policy, general product attributes, and the confirmation threshold. Cumulative ceilings are enforceable only where some party accumulates state across transactions, which five of the ten schemes examined do and the remainder do not. Most consequentially, virtual-card controls already enforce cumulative caps and merchant-category scope, and open-banking variable recurring payments enforce cumulative caps, that the new agent protocols omit, so agent authorization is in specific respects a regression against rails that preceded it. We release the coding protocol and evidence table, and retain version-pinned specification snapshots for audit.

Keywords: Agentic payments, delegated authority, authorization, conformance analysis, payment protocols, AI agents, agentic commerce, spending controls, AI governance, standards.

I. INTRODUCTION

Between 2025 and 2026 the infrastructure permitting software agents to spend money on a person's behalf moved from demonstration to deployment. Google's Agent Payments Protocol introduced three cryptographically signed mandates carried as verifiable digital credentials [17], reaching version 0.2 in April 2026 and being contributed by Google to the FIDO Alliance's standards-development process [28]. Mastercard bound tokenized credentials to agent identities [10] and later published a verifiable-intent credential framework [24]. Visa published a protocol for authenticating agent requests [19] alongside a commerce product promising enforced spending limits [11]. A model provider and a payment processor released an agentic commerce protocol with a delegated payment token [18]. An HTTP-native machine payment standard passed to foundation governance [12], [35], and a session-based machine payments protocol launched on a purpose-built settlement chain [25].

Each of these systems is described, by its authors and its adopters, as a mechanism for delegating bounded financial authority. The claim is that a principal grants an agent permission to spend within stated limits, and the infrastructure holds the agent to those limits. A substantial literature now establishes that agents do not reliably hold themselves to stated limits: web agents evaluated under explicit policies lose a large fraction of nominal completions once violations are counted [9], coding agents violate action boundaries in the majority of underspecified runs [34], agents under outcome pressure violate mandated constraints in a substantial minority of scenarios [23], and tool-using agents succeed unreliably across repeated trials of identical tasks [7]. If agents are unreliable, external expression and enforcement of authority becomes the load-bearing control.

This paper asks the question logically prior to whether agents obey delegated authority: whether the deployed protocols can express it. The distinction matters because a constraint that exists only as prose in a mandate is not a control. It is a

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

description of a control. If a principal's spending ceiling is carried in a free-text field that no party parses, then the ceiling is enforced by the agent's good behaviour alone, and the cryptographic apparatus surrounding it secures the wrong thing: it makes the instruction non-repudiable without making it binding.

Two systematizations published in April 2026 approach this territory. One organizes threats in agentic commerce along five dimensions including transaction authorization, and calls for defences addressing the authorization gaps left by current protocols [27]. The other decomposes agent-to-agent payments into discovery, authorization, execution and accounting, and names weak intent binding and misuse under formally valid authorization as unresolved [26]. Neither codes protocols field by field against a defined model of delegated authority, which is what this paper does.

1.1 Contributions

- We define an authorization envelope of eight fields, derived from the delegated-authority literature and from the control primitives already enforced on existing payment rails rather than chosen post hoc.
- We specify an auditable coding protocol distinguishing expressible from advisory constraints by the operational test of whether any identified party validates the field.
- We code eight deployed agent-payment protocols and two classes of existing rail against the envelope, reporting a conformance matrix with per-cell evidence.
- We identify three fields unsupported almost everywhere, and correct the intuitive expectation that cumulative ceilings are universally absent.
- We establish that agent authorization is, in specific and identifiable respects, a regression against the control expressiveness of virtual cards and open-banking variable recurring payments.

1.2 Paper Organization

Section II surveys agent-payment protocols, the behavioural evidence motivating external enforcement, prior comparative work, and the regulatory context. Section III defines the authorization envelope and the drift classes it must render detectable. Section IV specifies the coding protocol. Section V reports the conformance matrix and discusses the regression finding. Sections VI and VII conclude and set out future work.

II. LITERATURE SURVEY

2.1 Deployed Agent-Payment Protocols

The protocols examined here fall into three families. Mandate-based protocols express the principal's authority as a signed credential accompanying the transaction; the agent payments protocol is the leading example, carrying intent, cart and payment mandates as verifiable digital credentials [17], [28], with the credential model itself imposing no expressiveness limit [13]. Token-based protocols bind a payment credential to an agent and a scope; this describes the agentic token approach [10], [24] and the delegated payment token of the agentic commerce protocol [18]. Machine-payable protocols embed payment in the transport layer: an HTTP status-code-based standard with facilitator verification [12], [35] and a session-based protocol using off-chain vouchers against an authorized cap [25]. A fourth mechanism, agent authentication, is often discussed alongside these but answers a different question, establishing that a request comes from a recognized agent rather than that it falls within authority [19], [5].

2.2 Behavioural Evidence for External Enforcement

The case for expressing authority externally rests on evidence that agents do not enforce it internally. Benchmarks pairing tasks with explicit policies report that completion measured under policy compliance falls well below nominal completion [9]. Action-boundary violations occur in the majority of runs when instructions are underspecified [34]. Constraint violation under outcome pressure affects a substantial minority of scenarios even where constraints are stated plainly [23]. Budget-constrained benchmarks find agents overspending and under-escalating [38], [39], extending earlier shopping environments that already found agents unreliable on intent-grounded purchasing under coupon and budget constraints [4], [16]. Longitudinal work observes the erosion of an investment mandate over extended horizons [33]. Adversarial content delivered through tool output compounds the problem [8], [31], and red-teaming shows that policy adherence degrades further under deliberate pressure [14]. This paper takes that evidence as established and asks what the deployed infrastructure does about it.

2.3 Authorization, Delegation and Existing Control Primitives

Scoped delegation is long established through authorization frameworks and token exchange [1], [2], and recent work adapts the lineage to agents through task-based scope matching [20] and

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

formal treatment of authorization propagation, which introduces task-scoped authorization envelopes and notes that the field is converging on the problem without converging on a standard [29]. Existing payment rails, meanwhile, have enforced granular delegated spending controls for years. Commercial and virtual card programmes bind per-transaction ceilings, velocity limits over daily, weekly or monthly windows, transaction-count limits, merchant identifier locks and merchant-category allow and block lists, all enforced by the issuer or issuer-processor at authorization [40], [41]. Open-banking variable recurring payments express a consent carrying a maximum individual payment amount, a maximum cumulative value and a maximum cumulative count of payments, enforced by the account servicing institution [42]. These primitives are the empirical basis for the envelope defined in Section III, and the comparison basis for Section V.

2.4 Regulatory Context

The European Union's artificial intelligence regulation entered into force in 2024 [6]. Transparency obligations for specified AI systems apply from 2 August 2026, and the European Commission's enforcement powers for general-purpose AI model obligations also apply from that date. The current implementation timeline places high-risk obligations later, with Annex III systems applying from 2 December 2027 and high-risk AI embedded in Annex I regulated products from 2 August 2028. Guidance addressing agentic systems specifically remains preliminary. A central bank working paper testing a reasoning model on intraday liquidity management emphasised continuing human oversight [21]. A securities-industry oversight report names three risks in terms that map onto the present construct: agents acting autonomously without human validation, agents acting beyond the user's actual or intended scope and authority, and multi-step reasoning that complicates auditability [22]. Payment services regulation, meanwhile, continues to presume a human authorizing each transaction, leaving delegated initiation and strong customer authentication for agents unresolved. Each of these expectations presupposes that delegated authority is expressible and evidenced, a framing developed elsewhere as an evidence-system design problem rather than a model-performance one [30]. Section V reports how far that presupposition holds.

2.5 Prior Comparative Work

Comparisons of agent-payment protocols exist but are organized by threat model or lifecycle rather than by authorization expressiveness. The agentic-commerce systematization enumerates the same protocol set examined here and derives cross-layer attack vectors [27]. The agent-to-agent payments systematization organizes a four-stage lifecycle and identifies weak intent binding [26]. A parallel security literature examines individual protocols empirically, demonstrating unpaid-service and paid-but-denied outcomes against the machine-payable standard [31], logic flaws permitting free-riding [32], and authorization failures across facilitators collectively serving tens of thousands of sellers [36]. That work asks whether implementations honour the constraints their specifications define; this paper asks the prior question of which constraints the specifications define at all, and the two are complementary rather than competing. None codes a defined authorization model field by field across protocols, and none compares agent protocols against the control expressiveness of the rails they build upon. That combination is the contribution here.

III. PROBLEM DEFINITION

3.1 The Authorization Envelope

Let a principal delegate financial authority to an agent through an authorization envelope E, a tuple of eight fields:

$$E = (c\_tx, c\_cum, M, K, A, W, S, \theta) \qquad (1)$$

where c_tx is the per-transaction ceiling, c_cum the cumulative ceiling across a session or period, M the permitted merchant set, K the permitted merchant-category set, A the required product or service attributes, W the temporal validity window, S the substitution policy governing permitted deviations from a specified merchant, good or price, and θ the amount above which the principal must explicitly approve. The fields are not invented for this study. Six of the eight correspond directly to controls already enforced on commercial card and open-banking rails [40], [41], [42]; the remaining two, required attributes and substitution policy, are the constraints a principal states routinely in natural language and that agents are documented to violate [34], [38], [39].

Two definitional points matter for coding. First, θ is an amount, not a flag. A boolean requiring confirmation for every transaction is a different control from a threshold above which confirmation is required, and only the latter permits bounded autonomy. Second, c_cum is a property of a sequence, not of a transaction, and therefore cannot be evaluated by any party holding only the transaction in front of it.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

3.2 Drift Classes and Detectability

Let adm(t, E) denote the admissibility predicate evaluating a transaction t against every field of E. An agent departs from delegated authority in identifiable ways, and each corresponds to a field of E. Monetary drift exceeds c_tx; cumulative drift exceeds c_cum with every constituent transaction admissible; substitution drift transacts outside M or beyond what S permits; semantic drift omits a required attribute in A; confirmation evasion executes above θ without approval, including by decomposition into sub-threshold parts; temporal drift falls outside W. The formal condition is that drift is present in a session producing executed transactions T when

$$\exists\, t \in T : \neg adm(t, E) \;\vee\; \Sigma\_\{t \in T\}\, amount(t) > c\_cum \qquad (2)$$

The second disjunct is the case that per-presentment checking cannot see. The consequence for this paper is direct: a drift class is structurally undetectable wherever the corresponding field of E is absent from the protocol, irrespective of how well any party performs. Conformance therefore bounds enforceability, which is why expressiveness is the prior question.

Figure 1 situates the two moments this distinction turns on. Authority is expressed once, at delegation; it is enforced later, against a transaction the environment has reshaped in the interval. Whatever the protocol cannot express at the first moment, no party can enforce at the second.

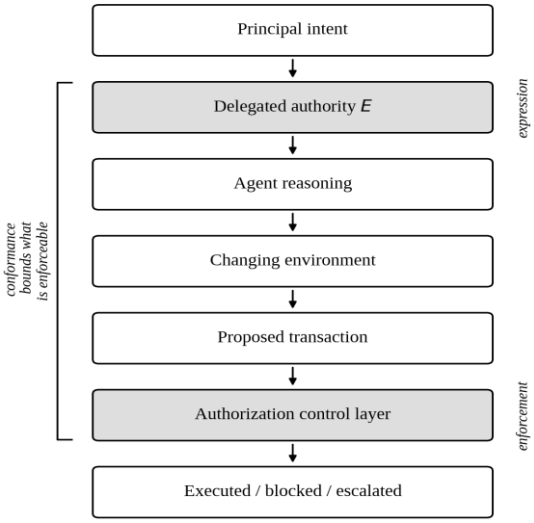

Figure 1: Delegation pipeline. Authority is expressed once, at delegation, and enforced later against a transaction the environment has reshaped. What the protocol can express at the first shaded stage bounds what any party can enforce at the second

3.3 Research Questions

RQ1. Which fields of the authorization envelope are expressible in deployed agent-payment protocols, which are merely advisory, and which are absent?

RQ2. Which drift classes are consequently undetectable, and are any undetectable across every protocol examined?

RQ3. How does the control expressiveness of agent-payment protocols compare with that of the existing payment rails on which they are built?

IV. METHODOLOGY

4.1 Unit of Analysis and Corpus

The primary unit of analysis is the version-pinned normative specification where one is publicly available. For products without a public schema, vendor-controlled developer documentation is examined instead, with product and marketing material separately provenance-tagged and never sufficient to support an expressible coding. Eight agent-payment protocols are coded: the agent payments protocol, the machine-payable HTTP standard, the agentic commerce protocol's delegated payment specification, the agentic token programme, the verifiable intent credential framework, the trusted agent protocol, the intelligent commerce product, and the machine payments protocol. Two classes of existing rail are coded on the same instrument as a comparison baseline: virtual and commercial card controls, and open-banking variable recurring payments. Each coded cell records the specification name, version or commit, URL and date accessed. All specifications were accessed on 18 August 2026. Several mirror domains reproduce specification text without authority and were excluded; only publisher-controlled domains and official repositories were treated as primary.

4.2 Coding Scheme

Each of the eight envelope fields is coded against each protocol using three mutually exclusive codes defined in Table 1. The distinguishing test between the first two codes is whether any identified party validates the field. This operationalizes an otherwise slippery distinction: a field nobody checks is advisory by definition, however prominently the specification discusses it.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

Table 1: Coding scheme and decision rules

Code	Decision rule
E	A named schema field of appropriate type exists AND an identified party validates it at an identified point in the authorization flow.
A	The constraint can be conveyed in a normative specification or the vendor's own developer documentation (free text, optional or nullable field, natural-language intent), but no identified party validates it.
N	No representation exists, or the representation cannot be checked where authorization occurs.

Five tie-break rules resolve recurring ambiguities, stated in full in the supplementary codebook and summarized here. Optional or nullable fields default to advisory unless a validation obligation is stated in normative language. Where a specification's own example instantiates the field as null, the coding for that instance is absent, and this instance-dependence is reported. Capabilities asserted only in press releases, product marketing or analyst material, with no corresponding support in the vendor's own developer documentation or normative specification, are coded absent for conformance purposes and reported separately as claimed capabilities. Vendor-controlled developer documentation may support an advisory code where a constraint is described but no typed field and validating party are identified. Fields whose enforcement is described but whose validating party is unstated are coded advisory. Finally, a field of the wrong type does not satisfy the envelope field it resembles: a boolean requiring confirmation does not satisfy an amount-valued threshold, and a list of stock-keeping units does not satisfy a category set.

4.3 Provenance Tagging

Every cell carries a provenance tag distinguishing primary specification, developer documentation, and marketing or press material. This matters because the corpus is uneven: some protocols publish complete JSON schemas while others describe capabilities only in prose intended for buyers. Reporting provenance prevents the analysis from crediting a protocol for a control that exists only as a claim, while also preventing the opposite error of treating an undocumented capability as proven absent.

4.4 Reliability and Auditability

This is a single-coder study, and no reliability coefficient is reported. That is a deliberate choice rather than an omission. An intra-rater coefficient computed by the same coder against the same matrix would measure the author's consistency rather than the codebook's clarity, which is the property a reader has reason to doubt; and because the decision rules are close to deterministic, requiring a typed field of appropriate kind and an identified validating party, a high coefficient would follow from the rules themselves rather than from anything the measurement established. Auditability is pursued directly instead. The matrix, codebook, decision rules, tie-break rules and per-cell evidence were frozen on 18 August 2026 and are supplied in full, so that any reader may re-code the corpus from the same documents and compare cell by cell against the published result. Where a coding rests on documentation that could not be settled from the text alone, the cell is named in the register of insufficient documentation rather than resolved by inference. Maintainer correspondence, if sought, is supplementary evidence and does not redefine the pinned version.

4.5 Limitations of the Corpus

Two protocols could not be coded completely from public documentation. The agentic token programme publishes no public schema exposing bindable scope parameters. Vendor developer guidance describes merchant-specific and time-limited or single-use token behavior, which is coded advisory, while broader claims about spending and category controls remain marketing-only and are therefore coded absent for conformance purposes. The intelligent commerce developer portal documents amount, merchant and validity controls, but other claimed capabilities, including period limits, category scope and approval prompts, are not exposed in the public developer documentation and are coded absent under the same rule. The portal also states that the product remains in development and deployment. These gaps are reported as findings rather than concealed as limitations.

V. RESULTS & DISCUSSION

5.1 The Conformance Matrix (RQ1)

Table 2: Conformance of deployed agent-payment protocols and existing payment rails against the authorization envelope. E expressible, A advisory, N absent.

Protocol / rail	ctx	ccum	M	K	A	W	S	th
AP2 v0.2	E	E	E	A	A	E	A	A
x402 v2 (upto)	E	N	E	N	N	E	N	N

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

Protocol / rail	ctx	ccum	M	K	A	W	S	th
ACP (delegated pay)	E	N	E	N	N	E	N	N
Mastercard Agent Pay	N	N	A	N	N	A	N	N
Verifiable Intent v0.1	E	E*	E	N	A	E	N	N
Visa TAP	N	N	N	N	N	N	N	N
Visa Intelligent Comm.	E	N	E	N	N	E	N	N
Tempo MPP (sessions)	**E**	**E**	**E**	**N**	**N**	**E**	**N**	**N**
Existing rails								
Virtual / commercial card	E	E	E	E	N	E	N	N
Open-banking VRP	E	E	E	N	N	E	N	N

*Fields: ctx per-transaction ceiling; ccum cumulative ceiling; M merchant set; K category set; A required attributes; W validity window; S substitution policy; th confirmation threshold. * Cumulative enforcement in the verifiable intent draft requires cross-transaction state that a stateless verifier cannot supply; the draft is not yet integrated into a shipping interface.*

The mandate-based protocol illustrates a split between what a principal signs and what a network evaluates. The intent mandate, which the user signs, carries a merchant list, a stock-keeping-unit list, a refundability boolean, an expiry and a boolean requiring cart confirmation, together with a natural-language description of the principal's request. It carries no amount field, and the specification's worked examples place the budget inside that description [17]. The payment mandate, presented to the network and issuer, carries a separate constraint vocabulary including an amount range within which the payment amount must fall and a budget whose evaluation requires accumulating total spend across previously closed mandates [28]. Both are stated in normative language, so both are coded expressible. The consequence is not that the protocol fails to express a ceiling but that the ceiling the user authorizes and the ceiling the network enforces are carried in different objects, populated by different parties, and nothing in the public specification requires the second to be derived from the first. The protocol's dispute guidance is consistent with that separation. Its scenario table treats an agent that "autonomously purchases an item that violates the user's signed Intent Mandate (e.g., exceeds budget or is the wrong item)" as a case resolved by evidence, the key evidence being that "the Intent Mandate vs. the cart transaction details should show the discrepancy" [17]. The discrepancy is discovered by comparison after the fact rather than prevented at authorization. The specification's concluding section is more direct still: among the problems it leaves to the ecosystem is delegated authorization, where it states that "the long-term goal is to move toward solutions that offer more granular, time-bound access controls for agents" [17]. The gap this paper measures is one the protocol's own authors name.

The machine-payable and delegated-token protocols behave differently. Both express a per-transaction maximum that a defined party verifies, the facilitator in one case and the payment service provider in the other, and both bind the recipient and an expiry in normative language [12], [18]. Their limitation is scope rather than enforcement: neither expresses category, attribute, substitution or threshold constraints, and both are single-use by construction, the machine-payable specification going so far as to list recurring payments and open-ended allowances explicitly among unsupported patterns [12].

Figure 2 records where each protocol validates a constraint. Two patterns are visible. No protocol validates at the agent, which is consistent with the premise that self-enforcement is not treated as a control by any specification examined. And validation is distributed across parties within a single protocol: the mandate protocol has the merchant checking a cart against an intent it must interpret rather than parse, while the network and issuer evaluate the typed constraints of the payment mandate. The fields that resolve to advisory are precisely those left with the interpreting party rather than the evaluating one.

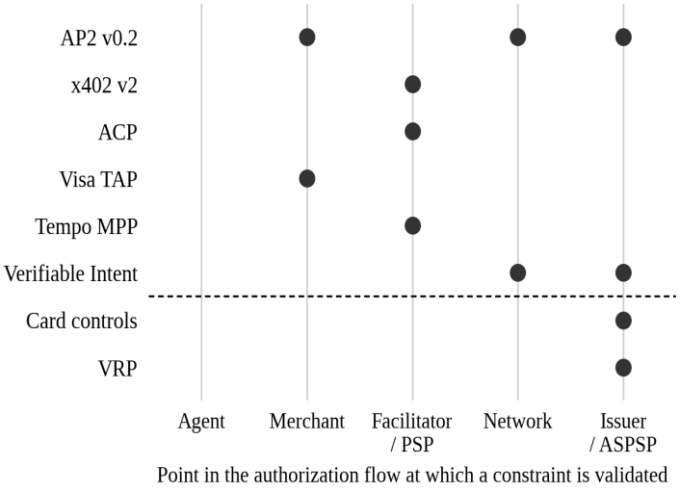

Figure 2: Point in the authorization flow at which each protocol validates a constraint. The agent column is empty throughout. Existing rails, below the dashed rule, validate at the issuer or account servicing institution

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

5.2 Undetectable Drift Classes (RQ2)

Table 3: Structural detectability of each drift class given the conformance results

Drift class	Field	Structurally detectable under
Monetary	ctx	AP2 payment mandate, x402, ACP, MPP, VIC, both legacy rails
Cumulative	ccum	AP2 payment mandate, MPP, VI draft, both legacy rails; all five accumulate cross-transaction state
Substitution	M, S	Merchant binding widely; substitution tolerance nowhere
Semantic	A	Nowhere; AP2 refundability is the sole typed attribute
Confirmation evasion	th	Nowhere as a validated field; marketing-only approval claims are excluded from conformance coding
Temporal	W	All except Mastercard Agent Pay, advisory, and Visa TAP, absent

Three fields are unsupported almost everywhere, and the consequence is that three drift classes are structurally undetectable across the deployed corpus. No protocol expresses a substitution policy, so an agent that switches merchant, venue or specification within a permitted price band cannot be distinguished from one that does so outside it. No protocol expresses general product attributes; the sole typed instance in the entire corpus is a single refundability boolean, and the one credential framework carrying richer product description states explicitly that those fields are informational and not machine-verified [24]. No protocol expresses an amount-valued confirmation threshold as a validated field; the one product asserting real-time approval prompts does so without published schema. Bounded autonomy in the sense a principal ordinarily means, proceed without me below this amount, is therefore not enforceably expressible anywhere in the corpus; the available constructs are confirm always or confirm never.

The expectation that cumulative ceilings are likewise universally absent is not sustained, and the paper is stronger for the correction. Cumulative limits are absent from the machine-payable and delegated-token protocols and from the documented agentic token surface. They are asserted in product material for the intelligent commerce product but are excluded from conformance coding because no corresponding public developer field is exposed. They are present and enforced in three of the agent protocols examined, and those three share a mechanism with the two existing rails discussed below. The session-based machine payments protocol draws a pre-authorized cap down through voucher accounting across many payments [25]. The verifiable intent draft defines a budget field its own specification marks as requiring network-maintained state that a stateless verifier cannot supply [24]. The mandate protocol's payment mandate defines a budget whose evaluation requires that the requested amount plus the sum of previously closed mandates remain within a maximum, with each approved amount added to the accumulated total [28]. Cumulative authority is therefore expressible exactly where some party accumulates state across transactions, and absent wherever authorization is evaluated per presentment. That is a mechanism rather than a coincidence, and it identifies what a protocol must add in order to express the constraint at all.

5.3 Regression Against Existing Rails (RQ3)

The comparison with existing rails is the study's most consequential result. Virtual and commercial card programmes express and enforce per-transaction ceilings, velocity limits across daily, weekly and monthly windows, transaction-count limits, merchant locks and merchant-category allow and block lists, all validated by the issuer or issuer-processor at the moment of authorization [40], [41]. Open-banking variable recurring payments express a consent carrying a maximum individual amount, a maximum cumulative value and a maximum cumulative count, enforced by the account servicing institution and established once under strong customer authentication [42].

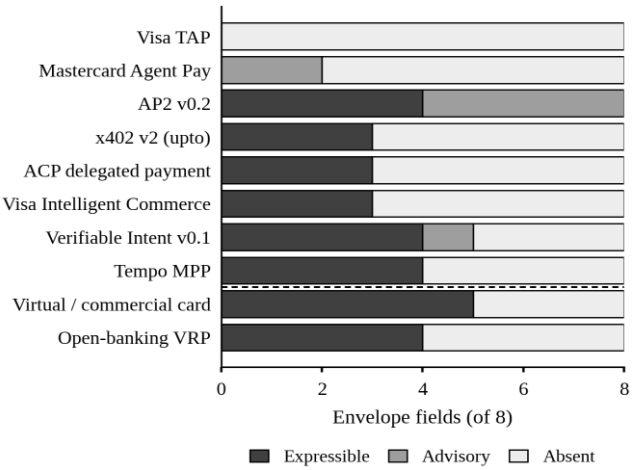

Figure 3: Envelope coverage by protocol, agent protocols above the dashed rule and existing rails below. Advisory segments mark constraints a protocol conveys but no party validates

Against that baseline, the new agent protocols express fewer enforced constraints in specific and identifiable respects.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

Merchant-category scope is the clearest instance: merchant category codes have been an enforced control on card rails for decades and appear in no agent protocol schema examined here. Substitution tolerance and an amount-valued approval threshold appear in neither the agent protocols nor the rails, and are therefore gaps in the field rather than regressions against it.

Figure 3 makes the comparison legible at a glance. Commercial-card controls enforce more envelope fields than any agent protocol examined, while open-banking recurring-payment consents match the strongest agent protocols at four enforced fields. The mandate protocol has the broadest overall coverage among the agent protocols, but half of its eight fields remain advisory rather than externally enforced.

This should be stated carefully, because the protocols are not equivalent objects and the comparison can be pushed too far. Agent protocols address problems the older rails do not, principally agent identity, intent evidence and dispute attribution, and several are explicitly early versions with roadmaps. The claim is not that they are worse overall. It is narrower and harder to dismiss: on the specific dimension of expressing what a principal has authorized, infrastructure built expressly for delegated autonomy currently encodes less than the general-purpose rails beneath it, and the fields it omits are precisely those governing the multi-transaction behaviour that autonomy makes possible.

5.4 Documented Capability Versus Claimed Capability

A methodological finding deserves separate statement. For two of the eight protocols, some of the constraint capabilities most prominently advertised are not exposed in a public schema or developer-facing field definition. Under the coding rule used here, vendor developer prose may support an advisory code, but press and product marketing alone do not increase conformance coverage. This is not an accusation of misrepresentation, since capabilities may exist in private interfaces or ship later. It is an observation with practical consequence: an institution cannot build a control it cannot read a schema or developer contract for, a supervisor cannot examine one, and a researcher cannot verify one. Provenance tagging turns this from an inconvenience into a reportable result.

5.5 Implications

- For protocol designers: the cheapest correction is typing what is already conveyed. A budget expressed in free text becomes enforceable when it becomes an amount field with a stated validator, without disturbing the credential architecture around it.
- For institutions: composition is available now. Where an agent protocol lacks cumulative or category scope, card controls already enforce both and open-banking consents enforce cumulative and payee scope, and pairing them is a deployable control rather than a research programme.
- For supervisors: expectations that an institution evidence bounded agent autonomy currently exceed what most protocols can represent, and the specific unrepresentable fields are identified here.

5.6 Threats to Validity

The study codes specifications, not deployments, and a capability enforced in a private interface would be coded absent; this is stated rather than mitigated, since the corpus is by construction the public record. Specifications are moving quickly, and every cell is version-pinned with an access date so that re-coding is tractable. Single-coder reliability is addressed in Section 4.4 but not eliminated. Finally, the envelope is one model of delegated authority among possible others; it is defended by derivation from existing enforced controls rather than claimed as canonical.

VI. CONCLUSION

Deployed agent-payment protocols express less delegated authority than their framing suggests. Coding eight protocols against an eight-field authorization envelope shows that substitution policy, general product attributes and amount-valued confirmation thresholds are unsupported almost everywhere, that cumulative ceilings are expressible only where some party accumulates state across transactions, and that the leading mandate protocol carries a validated ceiling in the credential the network evaluates while the credential the principal signs carries it only as free text. Against virtual card controls, which enforce cumulative caps and category scope today, and open-banking recurring payment consents, which enforce cumulative caps, agent authorization is on these dimensions a regression. The missing fields govern semantic constraints on what may be bought, conditional autonomy in when the principal must be asked, and, in the case of cumulative limits, behaviour across multiple transactions.

VII. FUTURE SCOPE

Three extensions follow. First, behavioural measurement: future work should develop an open harness that pairs machine-readable envelopes with sandboxed multi-step payment tasks and

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

scores authorization preservation across enforcement architectures, enabling the complementary question of whether agents respect fields that protocols do express. Second, composition: the pairing of agent mandates with issuer-side velocity and category controls suggested in Section 5.5 can be specified and evaluated as a reference architecture. Third, extension of the envelope formalism to tokenized collateral, where advance rates, eligibility criteria and health-factor limits occupy the positions of ceilings and attribute requirements, and where formal models of lending mechanics already exist [3], [15]. A complementary line of work proposes the primitives whose absence this study measures, setting out verifiable agent identity, bounded policy, constrained-authority custody and intent-based payments as composable architectural elements [37]; the conformance results here indicate which of those elements deployed protocols currently support.

DATA AND CODE AVAILABILITY

The codebook, decision rules, per-cell evidence table, corpus version table and register of insufficient documentation are provided as supplementary material with this manuscript and are deposited, under a Creative Commons Attribution 4.0 licence, at https://doi.org/10.5281/zenodo.22005866. The supplement records the version or document examined, source provenance and the 18 August 2026 corpus access date. Source snapshots are retained by the author for audit purposes and may be shared where licensing terms permit. No proprietary or licensed material is redistributed.

ACKNOWLEDGEMENT

This research received no external funding. The author declares no conflicts of interest.

REFERENCES

Chronological order per journal requirement. Specification entries carry the version examined and the date accessed; living developer documentation without a fixed publication date is listed last, ordered by date accessed.

[1] D. Hardt, Ed., "The OAuth 2.0 authorization framework," *RFC 6749, IETF*, Oct. 2012.

[2] B. Campbell, J. Bradley, N. Sakimura, and D. Tonge, "OAuth 2.0 token exchange," *RFC 8693, IETF*, Jan. 2020.

[3] M. Bartoletti, J. H. Chiang, T. Junttila, A. Lluch-Lafuente, M. Mirelli, and A. Vandin, "Formal analysis of lending pools in decentralized finance," *in Proc. ISoLA 2022, LNCS* 13703, pp. 335-355, 2022.

[4] S. Yao, H. Chen, J. Yang, and K. Narasimhan, "WebShop: Towards scalable real-world web interaction with grounded language agents," *arXiv*:2207.01206, 2022.

[5] M. Thomson and A. Backman, "HTTP message signatures," *RFC 9421, I*ETF, Feb. 2024.

[6] European Parliament and Council, "Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence," *OJ L* 2024/1689, 12 Jul. 2024.

[7] S. Yao, N. Shinn, P. Razavi, and K. Narasimhan, "tau-bench: A benchmark for tool-agent-user interaction in real-world domains," *arXiv*:2406.12045, Jun. 2024.

[8] E. Debenedetti et al., "AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents," *in Proc. NeurIPS 2024 Datasets and Benchmarks Track, arXiv:*2406.13352, 2024.

[9] I. Levy, B. Wiesel, S. Marreed, A. Oved, A. Yaeli, and S. Shlomov, "ST-WebAgentBench: A benchmark for evaluating safety and trustworthiness in web agents," *arXiv*:2410.06703, Oct. 2024; accepted, ICLR 2026.

[10] Mastercard, "Mastercard Agent Pay," *press release*, Apr. 2025; and Mastercard Developers, agentic commerce merchant guide, accessed 18 Aug. 2026.

[11] Visa, "Visa Intelligent Commerce," *developer portal and press materials*, May 2025; accessed 18 Aug. 2026.

[12] Coinbase, "x402: An open standard for internet-native payments," *specification v2 including the exact and upto schemes*, May 2025; repository accessed 18 Aug. 2026.

[13] W3C, "Verifiable Credentials Data Model v2.0," *W3C Recommendation*, 2025.

[14] I. Nakash, G. Kour, K. Lazar, M. Vetzler, G. Uziel, and A. Anaby-Tavor, "Effective red-teaming of policy-adherent agents," *in Proc. 2025 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2025; arXiv:2506.09600, Jun. 2025.

[15] M. Bartoletti and E. Lipparini, "A theory of lending protocols in DeFi," *arXiv:*2506.15295, Jun. 2025.

[16] J. Wang *et al.*, "ShoppingBench: A real-world intent-grounded shopping benchmark for LLM-based agents," *arXiv*:2508.04266, Aug. 2025.

[17] Google, "Agent Payments Protocol (AP2)," specification v0.1, 16 Sep. 2025; *A2A extension and mandate schemas*; accessed 18 Aug. 2026.

[18] OpenAI and Stripe, "Agentic Commerce Protocol: Agentic Checkout Specification and Delegated Payment Specification," version 2025-09-29, Sep. 2025; accessed 18 Aug. 2026.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

[19] Visa, "Trusted Agent Protocol," *specification and reference implementation*, 14 Oct. 2025; accessed 18 Aug. 2026.
[20] M. El Helou et al., "Delegated authorization for agents constrained to semantic task-to-scope matching," *arXiv*:2510.26702, Oct. 2025.
[21] I. Aldasoro and A. Desai, "AI agents for cash management in payment systems," *BIS Working Papers* No. 1310, *Bank for International Settlements*, 26 Nov. 2025.
[22] FINRA, 2026 FINRA Annual Regulatory Oversight Report, *Washington, DC*, 9 Dec. 2025.
[23] M. Q. Li, B. C. M. Fung, M. Weiss, P. Xiong, K. Al-Hussaeni, and C. Fachkha, "A benchmark for evaluating outcome-driven constraint violations in autonomous AI agents," *arXiv*:2512.20798, Dec. 2025 (rev. May 2026).
[24] Mastercard, "Verifiable Intent," s*pecification* v0.1-draft, 18 Feb. 2026; accessed 18 Aug. 2026.
[25] Stripe and Tempo, "Machine Payments Protocol," specification and documentation, 18 Mar. 2026; accessed 18 Aug. 2026.
[26] Y. Zhang et al., "SoK: Blockchain agent-to-agent payments," *arXiv*:2604.03733, Apr. 2026.
[27] Q. Mao, J. Wang, Y. Liu, L. Zhu, C. Ma, and J. Yan, "SoK: Security of autonomous LLM agents in agentic commerce," *arXiv:*2604.15367, Apr. 2026.
[28] Google, "Agent Payments Protocol v0.2.0 release and Payment Mandate constraint documentation," 28 Apr. 2026, ap2-protocol.org/ap2/payment_mandate/; and FIDO Alliance, "FIDO Alliance to Develop Standards for Trusted AI Agent Interactions," 28 Apr. 2026; accessed 18 Aug. 2026.
[29] K. Tallam, "Authorization propagation in multi-agent AI systems: Identity governance as infrastructure," *arXiv*:2605.05440, May 2026.
[30] I. T. Staley, "Regulatory-grade evidence for AI in FinTech: A control-mapped framework and blockchain-anchored architecture for audit replay," *Journal of Computational Law and Legal Technology*, pp. 1-7, *advance online publication* 9 May 2026, doi:10.47852/bonviewJCLLT62029281.
[31] Z. Li, Q. Wang, and Z. Wang, "Five attacks on x402 agentic payment protocol," *arXiv*:2605.11781, May 2026.
[32] S. Ling, Y. Huang, Y. Du, Y. Chen, Y. Zhou, L. Wu, and C. Wang, "Free-riding the agentic web: A systematic security analysis of x402 payments," *arXiv:*2605.30998, May 2026.
[33] M. U. Safder et al., "FinPersona-Bench: A benchmark for longitudinal psychometric stability of autonomous financial agents," *arXiv:*2606.31522, Jun. 2026.
[34] Z. Ji et al., "Coding agents are guessing: Measuring action-boundary violations in underspecified DevOps instructions," *arXiv*:2607.02294, Jul. 2026.
[35] Linux Foundation, "x402 Foundation launch announcement," 14 Jul. 2026.
[36] Q. Wang et al., "When HTTP 402 meets the blockchain: Risks on emerging x402 payments," *arXiv:*2607.19545, Jul. 2026.
[37] I. Staley, "A reference architecture for AI agents on blockchain infrastructure: Identity, policy, payments, and custody as composable primitives," *Zenodo*, Jul. 2026, doi:10.5281/zenodo.21432152.
[38] J. Wu, M. Gong, F. Cheng, and Q. Zhao, "EcoAgent-Bench: Evaluating economic decision-making in budget-constrained LLM agents," *arXiv*:2608.05519, Aug. 2026.
[39] A. Li *et al*., "ComboShoppingBench: Evaluating LLM agents for budget-constrained basket shopping with coupons," *arXiv*:2608.09282, Aug. 2026.
[40] Marqeta, "Velocity controls and spend controls," developer documentation, accessed 18 Aug. 2026.
[41] Checkout.com, "Card controls and velocity limits," *developer documentation*, accessed 18 Aug. 2026.
[42] Open Banking Limited, "Variable Recurring Payments: consent parameters and standards," accessed 18 Aug. 2026.

AUTHOR'S BIOGRAPHY

Ian Staley is an independent researcher and product leader working on tokenization, agentic financial infrastructure and AI governance, with a publication record spanning blockchain and digital assets, financial technology, AI systems and quantum foundations. ORCID 0009-0000-8592-3186.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 19-29, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308003
Available Online at https://www.jaiet.com

Citation of this Article:

Ian Staley. (2026). Expressible, Advisory, or Unenforceable: A Conformance Analysis of Delegated Financial Authority in Deployed Agent-Payment Protocols. *Journal of Artificial Intelligence and Emerging Technologies (JAIET).* 3(8), 19-29. Article DOI: https://doi.org/10.47001/JAIET/2026.308003

***** End of the Article *****