Independent Submission
Internet-Draft
Intended status: Informational
Expires: 19 October 2026

C. Hood
Nomotic, Inc.

17 April 2026

AGTP Merchant Identity and Agentic Commerce Binding
draft-hood-agtp-merchant-identity-00

Abstract

The Agent Transfer Protocol (AGTP) specifies the sending side of an agentic transaction: agent identity, Authority-Scope enforcement, Budget- Limit declaration, and a signed Attribution-Record on every method invocation. The receiving side of a PURCHASE transaction -- the merchant or service provider -- has no equivalent protocol-level identity or verification mechanism. This is the merchant identity gap.

This document specifies the AGTP Merchant Identity and Agentic Commerce Binding. It defines the Merchant Manifest Document, a signed identity record structurally parallel to the Agent Manifest Document; the Merchant Birth Certificate, the genesis record from which a merchant’s canonical identifier is derived; Merchant Trust Tiers aligned with AGTP Trust Tier semantics; and the protocol integration points at which merchant identity is verified. These include the PURCHASE method handshake, the DISCOVER method result surface, and the Attribution-Record. This document also defines the Intent-Assertion header for portable, detached principal- authorized intent, the Cart-Digest mechanism for multi-line-item transactions, and the 455 Counterparty Unverified status code. Together these mechanisms close the verification loop between agent and merchant within AGTP’s governance model.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 19 October 2026.

Table of Contents

1. Introduction

1.1. The Merchant Identity Gap

AGTP today provides strong guarantees for the sending side of an agent transaction. A PURCHASE invocation carries a cryptographically derived Agent-ID, a Principal-ID identifying the accountable human or organization, an Authority-Scope declaration (including payments: purchase), a Budget-Limit enforced at invocation time, and a signed Attribution-Record retained for audit. The requesting agent's governance context is fully expressed at the protocol layer.

The receiving side has no equivalent. An AGTP PURCHASE currently resolves to a network endpoint with no protocol-level assertion of the receiving party's identity, lifecycle state, legal entity, payment network acceptance, or dispute policy. An agent with payments: purchase scope will transact with any endpoint its principal (or the upstream orchestration logic) directs it toward. There is no protocol-level signal distinguishing a verified merchant of record from an endpoint that merely answers on the expected port.

This gap has direct operational consequences as agent-driven commerce scales:

* Payment networks and card issuers extending protection to agent-initiated transactions require verifiable identity on both parties to the transaction, not just the agent.

* Dispute investigation requires a cryptographically linked record of both the initiating agent and the merchant counterparty at the time of the transaction.

* Merchants suspended for fraud, chargebacks, or regulatory action have no mechanism to be removed from the agent-visible transaction surface in the absence of a governed merchant directory.

* Agents cannot distinguish a merchant whose identity has been verified from one that has merely published a service endpoint.

This document closes the gap by introducing a merchant-side identity structure that mirrors the agent-side identity structure already specified in [AGTP].

1.2. Relationship to Payment Networks

This specification is a transport-layer identity and verification mechanism for merchant counterparties. It does not define payment credential handling, tokenization, authorization messages to card networks, or settlement. Those belong to payment-rail specifications operated by issuers, networks, and acquirers.

The relationship is complementary. Payment-network programs that extend protection, fraud coverage, or dispute handling to agent-initiated transactions need verifiable identity for both the agent and the merchant. AGTP establishes verifiable agent identity through the Agent Birth Certificate and Agent Manifest Document. This document extends the same structural model to the merchant side, producing an Attribution-Record that names both counterparties cryptographically. Payment networks consume that record as an input to their own authorization and dispute processes; they do not need to speak AGTP to do so.

1.3. Design Principles

Structural parallelism. Merchant identity uses the same document formats, trust tiers, lifecycle states, and governance zone semantics as agent identity. A governance platform that registers agents registers merchants through the same registry and the same cryptographic machinery.

Verification at PURCHASE. Merchant identity is verified by the requesting agent immediately before executing PURCHASE. The verification result is recorded in the Attribution-Record. A verification failure is a 455 Counterparty Unverified response, not a protocol error.

Discovery surfaces both sides. The DISCOVER method defined in [AGTP-DISCOVER] returns Agent Manifest Documents for agent queries. This document extends DISCOVER to optionally return Merchant Manifest Documents when the query intent targets a transactional counterparty rather than a capability-providing agent.

Portable intent. The Intent-Assertion header carries a detached, signed summary of principal-authorized purchase intent that can be forwarded to non-AGTP counterparties (payment networks, card issuers, acquirers) as standalone evidence without requiring those counterparties to speak AGTP.

Payment-rail neutrality. Nothing in this specification binds the merchant identity model to any particular card network, digital wallet, or settlement system. Merchant Manifests declare accepted payment network identifiers as an informational field; enforcement remains the responsibility of the payment rail.

2. Terminology

The key words "*MUST*", "*MUST NOT*", "*REQUIRED*", "*SHALL*", "*SHALL NOT*", "*SHOULD*", "*SHOULD NOT*", "*RECOMMENDED*", "*NOT RECOMMENDED*", "*MAY*", and "*OPTIONAL*" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals.

Merchant: A legal entity that offers goods or services for purchase and serves as the receiving counterparty to an AGTP PURCHASE invocation. A merchant is identified by a canonical Merchant-ID and represented by a Merchant Manifest Document.

Merchant-ID: A unique identifier for a specific merchant entity, derived from the hash of the merchant’s Birth Certificate. Carried in the Merchant-ID request header on PURCHASE invocations and in the Attribution-Record. Format: 256-bit hex-encoded value or domain-anchored URI of the form agtp://merchant.example.tld/merchant.

Merchant Birth Certificate: A cryptographically signed identity document issued to a merchant at registration time by a governance platform. The genesis record from which the canonical Merchant-ID is derived. Structurally parallel to the Agent Birth Certificate defined in [AGTP] Section 5.7. Fields include legal entity name, registered org domain, accepted payment networks, merchant category code, dispute and refund policy URIs, lifecycle state, and governance zone. Issued once per merchant; permanently bound; never reissued.

Merchant Manifest Document: A cryptographically signed application/agtp+json document returned when a merchant URI is resolved. Derived directly from the merchant's Birth Certificate and current registry record. Structurally parallel to the Agent Manifest Document defined in [AGTP] Section 5.5. Contains identity, trust tier, lifecycle state, accepted payment networks, dispute policy reference, and governance zone. Never contains executable content.

Merchant Trust Tier: A classification (1, 2, or 3) assigned to a merchant at registration time, aligned with the Agent Trust Tier semantics defined in [AGTP] Section 5.2. Tier 1 requires DNS-anchored domain verification of the merchant's registered org domain and a signed business-entity attestation from the governance platform. Tier 2 is org-asserted without DNS verification. Tier 3 is experimental and *MUST NOT* appear in production PURCHASE flows.

Intent-Assertion: A detached, signed JWT-format [RFC7519] token that summarizes principal-authorized purchase intent. Contains the principal ID, agent ID, merchant ID, item or cart digest, amount ceiling, currency, issuance timestamp, expiry, and a single-use nonce. Carried in the Intent-Assertion request header and forwardable to payment networks as standalone evidence of authenticated principal intent.

Cart-Digest: A cryptographic digest of a structured cart payload returned by a QUOTE invocation. Referenced in a subsequent PURCHASE invocation to bind the purchased cart to the quoted cart without requiring retransmission of line-item detail. Format: hash algorithm prefix followed by hex-encoded digest (e.g., sha256:3a9f2c1d...).

Counterparty Verification: The process by which an agent, before executing PURCHASE, retrieves the Merchant Manifest Document for the intended merchant, verifies its signature and lifecycle state, and records the verification result in the Attribution-Record.

MNS (Merchant Name Service): The merchant-side analogue of the Agent Name Service defined in [AGTP-DISCOVER]. An AGTP-aware server that maintains an indexed registry of Merchant Manifest Documents and answers DISCOVER queries targeted at merchant entities. An MNS *MAY* be co-located with an ANS or operated separately. Acts as a Scope- Enforcement Point for merchant discovery traffic.

3. The Merchant Identity Model

3.1. Merchant Birth Certificate

The Merchant Birth Certificate is the genesis record of a merchant's existence in the AGTP governance fabric. It is issued by a governance platform at merchant registration time through a process analogous to Agent Birth Certificate issuance ([AGTP] Section 5.7).

The required fields of a Merchant Birth Certificate are:

```
{
  "document_type": "merchant-birth-certificate",
  "schema_version": "1.0",
  "canonical_id": "7c2f9a3e1b8d4f6a...",
  "legal_entity_name": "Acme Commerce, Inc.",
  "org_domain": "acme.tld",
  "merchant_category_code": "5411",
  "registered_jurisdiction": "US-DE",
  "governance_zone": "zone:retail-verified",
  "accepted_payment_networks": ["visa", "mastercard", "amex", "discover"],
  "dispute_policy_uri": "agtp://acme.tld/merchant/dispute-policy",
  "refund_policy_uri": "agtp://acme.tld/merchant/refund-policy",
  "trust_tier": 1,
  "activated_at": "2026-03-15T12:00:00Z",
  "activating_principal": "agtp-platform-acme",
  "certificate_hash": "7c2f9a3e1b8d4f6a...",
  "signature": {
    "algorithm": "ES256",
    "key_id": "gov-platform-key-01",
    "value": "[base64-encoded-signature]"
  }
}
```

Figure 1: Merchant Birth Certificate Schema

The canonical_id field *MUST* equal the certificate_hash, a 256-bit cryptographic hash computed over the canonicalized certificate content excluding the signature field. This hash is the basis of the merchant's Merchant-ID and is used wherever the merchant is identified in AGTP wire-level structures.

The merchant_category_code field *SHOULD* follow the ISO 18245 Merchant Category Code standard where applicable. Governance platforms operating outside that classification *MAY* define alternate codes provided they are documented in the governance zone definition.

The accepted_payment_networks array is informational. It declares which payment networks the merchant represents itself as accepting. Payment-rail enforcement of this declaration is out of scope for this document; the field supports pre-flight filtering by requesting agents and ranking by Merchant Name Service implementations.

3.2. Merchant Trust Tiers

Merchant Trust Tiers align with the Agent Trust Tier semantics in [AGTP] Section 5.2:

Tier	Verification	Org Domain	Registry Visible
1 - Verified	DNS challenge per [RFC8555] plus business entity attestation	Required, verified	Yes
2 - Org-Asserted	None beyond self-declaration	Present, unverified	Yes, with warning
3 - Experimental	None	Optional	No

Table 1: Merchant Trust Tier Summary

Trust Tier 1 merchant registration *MUST* include, in addition to DNS ownership verification, a governance-platform-signed attestation that the registering party has provided evidence of the claimed legal entity’s existence and standing. The form of that evidence (incorporation document, tax identifier, equivalent jurisdictional registration) is governance-platform-defined and *MUST* be documented in the governance zone specification.

Trust Tier 2 merchants *MUST* have their Merchant Manifest Document include a trust_warning field with value "legal-entity-unverified". Requesting agents *SHOULD* surface this warning to principals before executing a PURCHASE against a Tier 2 merchant, or *MAY* reject Tier 2 merchants entirely based on governance policy.

Trust Tier 3 merchants *MUST NOT* appear in production PURCHASE flows. They exist for development and integration testing only.

3.3. Merchant Lifecycle States

Merchants occupy one of four lifecycle states mirroring the agent lifecycle states in [AGTP] Section 5.8:

State	Meaning
Active	Merchant is operational and eligible to receive PURCHASE
Suspended	Temporarily blocked (fraud review, chargeback threshold, compliance hold)
Revoked	Permanently removed; canonical Merchant-ID retired
Deprecated	Business ceased operations; canonical Merchant-ID retired

Table 2: Merchant Lifecycle States

A merchant in any state other than Active *MUST NOT* be treated as a valid counterparty by a requesting agent. The expected response to a PURCHASE targeting a non-Active merchant is 455 Counterparty Unverified.

Governance platforms *MUST* update the merchant’s Merchant Manifest Document signature within 60 seconds of a lifecycle state transition and *MUST* notify any Merchant Name Service instances indexing the merchant within the same window.

3.4. Merchant Manifest Document

The Merchant Manifest Document is the wire-level representation of merchant identity, returned in response to resolution of a merchant URI. It is derived from the Merchant Birth Certificate and current registry record; it is never manually authored.

```
{
  "document_type": "agtp-merchant-manifest",
  "schema_version": "1.0",
  "canonical_id": "7c2f9a3e1b8d4f6a...",
  "merchant_label": "acme-commerce",
  "legal_entity_name": "Acme Commerce, Inc.",
  "org_domain": "acme.tld",
  "merchant_category_code": "5411",
  "registered_jurisdiction": "US-DE",
  "governance_zone": "zone:retail-verified",
  "lifecycle_state": "Active",
  "trust_tier": 1,
  "accepted_payment_networks": ["visa", "mastercard", "amex", "discover"],
  "dispute_policy_uri": "agtp://acme.tld/merchant/dispute-policy",
  "refund_policy_uri": "agtp://acme.tld/merchant/refund-policy",
  "activated_at": "2026-03-15T12:00:00Z",
  "last_updated": "2026-04-10T09:12:44Z",
  "birth_certificate_hash": "7c2f9a3e1b8d4f6a...",
  "signature": {
    "algorithm": "ES256",
    "key_id": "gov-platform-key-01",
    "value": "[base64-encoded-signature]"
  }
}
```

Figure 2: Merchant Manifest Document Schema

Implementations *MUST* verify the signature against the governance platform's published key before trusting any field. An unsigned or invalid Merchant Manifest *MUST* be rejected with the same severity as an unsigned Agent Manifest.

The birth_certificate_hash field provides a cryptographic link from the manifest back to the genesis record. Implementations performing long-term audit reconstruction *MAY* use this hash to retrieve the archived Birth Certificate from the governance platform.

3.5. Merchant URI Forms

Merchant URIs follow the AGTP URI scheme defined in [AGTP] Section 5.1, using a reserved /merchant path component in place of /agents:

```
agtp://acme.tld/merchant
agtp://acme.tld/merchant/acme-commerce
```

Figure 3: Merchant URI Forms

The single-label form (agtp://acme.tld/merchant) resolves to the organization's primary merchant record. The labeled form (agtp://acme.tld/merchant/[merchant-label]) resolves to a specific merchant record for organizations operating multiple merchant identities under a single org domain (e.g., multi-brand retailers).

Canonical-identifier URIs of the form agtp://[canonical-id].merchant are also supported, analogous to the canonical agent URI form.

4. Counterparty Verification at PURCHASE

4.1. Verification Requirement

An agent with payments:purchase in its Authority-Scope *MUST* perform counterparty verification before executing PURCHASE against any merchant. Counterparty verification consists of:

1. Resolving the merchant URI (from the intended PURCHASE target) to retrieve the Merchant Manifest Document.

2. Verifying the manifest's signature against the governance platform's published key.

3. Verifying the merchant's lifecycle_state is Active.

4. Verifying the merchant's trust_tier meets or exceeds the threshold declared in the agent's governance policy for the current transaction amount.

5. Computing the manifest fingerprint (SHA-256 hash of the canonical manifest bytes) and carrying it in the Merchant-Manifest-Fingerprint request header.

Any of these steps failing *MUST* result in the PURCHASE not being sent. The requesting agent's runtime *SHOULD* surface the specific verification failure to the principal or governance platform; it *MUST NOT* silently fall back to an unverified transaction.

4.2. Verification at the Receiving Server

A receiving AGTP server that accepts PURCHASE invocations *MUST*:

1. Require the Merchant-ID request header to be present.

2. Require the Merchant-Manifest-Fingerprint request header to be present and to match the fingerprint of the server's current Merchant Manifest Document.

3. Return 455 Counterparty Unverified if either header is absent, if the Merchant-ID does not match the server's canonical ID, or if the fingerprint does not match.

This ensures that the manifest verified by the requesting agent is the same manifest the receiving server currently presents. An attack in which a requesting agent is redirected to a different endpoint than it verified is caught at the fingerprint check.

4.3. PURCHASE Request Example

The following request illustrates a verified PURCHASE invocation carrying merchant identity binding and a detached intent assertion:

```
AGTP/1.0 PURCHASE
Agent-ID: agtp://agtp.traveler.tld/agents/trip-planner
Principal-ID: usr-chris-hood
Authority-Scope: payments:purchase merchant:verify intent:assert
Session-ID: sess-trip-2026-04
Task-ID: task-purch-0421
Merchant-ID: agtp://acme.tld/merchant/acme-commerce
Merchant-Manifest-Fingerprint: sha256:3a9f2c1d8b7e4a6f...
Intent-Assertion: eyJhbGciOiJFUzI1NiIsImtpZCI6InByaW4ta2V5LTAxIn0...
Cart-Digest: sha256:7c2f9a3e1b8d4f6a...
Budget-Limit: USD=850.00
Content-Type: application/agtp+json

{
  "method": "PURCHASE",
  "task_id": "task-purch-0421",
  "parameters": {
    "cart_quote_id": "qt-7f3a9c",
    "principal_id": "usr-chris-hood",
    "amount": {"value": 842.17, "currency": "USD"},
    "payment_method": "tok-amex-default",
    "confirm_immediately": true
  }
}
```

Figure 4: PURCHASE with Merchant Identity Binding

The merchant server validates the merchant headers, accepts the purchase, and returns an Attribution-Record naming both counterparties:

```
AGTP/1.0 200 OK
Task-ID: task-purch-0421
Server-Agent-ID: agtp://acme.tld/merchant/acme-commerce
Attribution-Record: [signed attribution token]
Content-Type: application/agtp+json

{
  "status": 200,
  "task_id": "task-purch-0421",
  "result": {
    "order_id": "ORD-2026-0421-8847",
    "confirmation_code": "AQRT9X",
    "status": "confirmed",
    "amount_charged": {"value": 842.17, "currency": "USD"}
  },
  "attribution": {
    "agent_id": "agtp://agtp.traveler.tld/agents/trip-planner",
    "principal_id": "usr-chris-hood",
    "merchant_id": "agtp://acme.tld/merchant/acme-commerce",
    "merchant_fingerprint": "sha256:3a9f2c1d8b7e4a6f...",
    "intent_assertion_jti": "ia-4f7e1a2b",
    "method": "PURCHASE",
    "timestamp": "2026-04-15T14:22:18Z",
    "signature": {
      "algorithm": "ES256",
      "key_id": "merchant-key-acme-01",
      "value": "[base64-encoded-signature]"
    }
  }
}
```

Figure 5: PURCHASE Response with Dual-Party Attribution

The Attribution-Record now names the agent, the principal, and the merchant, each cryptographically bound through their respective Birth Certificate derivatives. This is the record consumed by downstream audit, dispute resolution, and payment-network protection programs.

5. The Intent-Assertion Header

5.1. Purpose

The Intent-Assertion header carries a detached, signed representation of principal-authorized purchase intent. It exists so that non-AGTP counterparties -- payment networks, card issuers, acquiring banks, dispute processors -- can verify principal intent without parsing a full AGTP message or operating AGTP infrastructure.

An Intent Assertion is a JWT [RFC7519] signed by the principal's governance key (or a delegated signing key bound to the principal's identity) carrying the minimum field set required to link a purchase to an authenticated principal decision.

5.2. Intent Assertion Claims

Claim	Type	Description
iss	string	Issuing governance platform identifier
sub	string	Principal-ID of the authorizing principal
aud	string	Merchant-ID of the intended counterparty
agent_id	string	Agent-ID of the executing agent
item_digest	string	Hash of purchased item or cart digest
amount_ceiling	object	{value, currency} maximum authorized
nbf	integer	Not-before timestamp (seconds since epoch)
exp	integer	Expiry timestamp (seconds since epoch)
jti	string	Unique assertion identifier for anti-replay
iat	integer	Issued-at timestamp

Table 3: Intent Assertion JWT Claims

Implementations *MUST* reject Intent Assertions whose exp is in the past, whose aud does not match the Merchant-ID in the PURCHASE request, or whose agent_id does not match the Agent-ID in the PURCHASE request. Assertions *MUST* be single-use: the jti is recorded in the Attribution-Record and *MUST NOT* be accepted a second time.

Recommended validity period: 300 seconds. Intent Assertions are not designed to persist; they cover the interval between principal authorization and transaction execution.

5.3. Forwarding to Payment Networks

The Intent Assertion is structured as a standalone JWT precisely so that it can be forwarded. A payment network receiving a merchant's authorization request *MAY* require the merchant to forward the Intent Assertion alongside the standard payment message. The payment network verifies the signature against the issuing governance platform's public key and treats a valid assertion as evidence of authenticated principal intent for the purposes of that network's authorization and dispute policies.

The specific mechanism for forwarding the Intent Assertion to a payment network, and the network's treatment of a valid assertion, is out of scope for this document. What this specification guarantees is that the assertion exists, is cryptographically verifiable without AGTP, and is bound to the principal, agent, and merchant named in the PURCHASE.

6. Cart Context

6.1. The Single-Item Limitation

The PURCHASE method as defined in [AGTP-METHODS] accepts a single item parameter and a single amount. Real-world agentic commerce transactions frequently involve multiple line items, tax, shipping, and per-line merchant-of-record variation. This document defines a Cart Context mechanism layered over the existing QUOTE and PURCHASE methods to accommodate structured carts without modifying the base method definitions.

6.2. QUOTE with Cart Payload

A requesting agent constructs a structured cart and submits it via QUOTE. The merchant server returns a signed cart_digest binding the quoted cart content to a unique quote identifier.

```
AGTP/1.0 QUOTE
Agent-ID: agtp://agtp.traveler.tld/agents/trip-planner
Principal-ID: usr-chris-hood
Authority-Scope: budget:query merchant:query
Merchant-ID: agtp://acme.tld/merchant/acme-commerce
Session-ID: sess-trip-2026-04
Task-ID: task-quote-0421
Content-Type: application/agtp+json

{
  "method": "QUOTE",
  "task_id": "task-quote-0421",
  "parameters": {
    "cart": {
      "lines": [
        {"sku": "FLIGHT-AA2847", "qty": 1, "unit_price": 487.00},
        {"sku": "HOTEL-MRTN-2N", "qty": 1, "unit_price": 298.00},
        {"sku": "CAR-COMPACT-3D", "qty": 1, "unit_price": 42.17}
      ],
      "currency": "USD",
      "tax": 15.00,
      "shipping": 0.00
    }
  }
}
```

Figure 6: QUOTE with Structured Cart

The response contains the quote identifier and the signed cart digest:

```
{
  "status": 200,
  "task_id": "task-quote-0421",
  "result": {
    "quote_id": "qt-7f3a9c",
    "cart_digest": "sha256:7c2f9a3e1b8d4f6a...",
    "total": {"value": 842.17, "currency": "USD"},
    "quote_valid_until": "2026-04-15T14:52:18Z",
    "quote_signature": {
      "algorithm": "ES256",
      "key_id": "merchant-key-acme-01",
      "value": "[base64-encoded-signature]"
    }
  }
}
```

Figure 7: QUOTE Response with Cart Digest

6.3. PURCHASE Referencing a Cart Digest

The subsequent PURCHASE invocation references the quote identifier and carries the cart digest in the Cart-Digest header, binding the purchase to the previously quoted cart without retransmission of line-item detail. The merchant server *MUST* verify the digest against its stored quote record and *MUST* reject the PURCHASE with 409 Conflict if the cart digest does not match a valid, unexpired quote.

7. DISCOVER Integration

7.1. Merchant Queries via DISCOVER

The DISCOVER method defined in [AGTP-DISCOVER] is extended by this document to optionally return Merchant Manifest Documents. A requesting agent signals a merchant-oriented discovery query by including the result_type parameter with value "merchant":

```
AGTP/1.0 DISCOVER
Agent-ID: agtp://agtp.traveler.tld/agents/trip-planner
Principal-ID: usr-chris-hood
Authority-Scope: discovery:query merchant:query
Session-ID: sess-trip-2026-04
Task-ID: task-disc-merch-01
Content-Type: application/agtp+json

{
  "method": "DISCOVER",
  "task_id": "task-disc-merch-01",
  "parameters": {
    "intent": "Ski rental in Park City accepting Amex",
    "result_type": "merchant",
    "merchant_category_codes": ["7999", "5941"],
    "accepted_payment_networks": ["amex"],
    "trust_tier_min": 1,
    "governance_zone": "zone:retail-verified",
    "limit": 5
  }
}
```

Figure 8: DISCOVER Query Targeting Merchants

The ANS or MNS server returns a ranked result set of Merchant Manifest Documents matching the query constraints. Ranking follows the same composite scoring model defined in [AGTP-DISCOVER] Section 3.4, with the following adjustments for merchant queries:

* behavioral_trust_score is replaced by merchant_reliability_ score, a governance-platform-assigned score reflecting the merchant's dispute rate, chargeback history, and fulfillment track record within the governance zone.

* capability_match_score is replaced by catalog_match_score, a relevance score between the query intent and the merchant's declared catalog categories and merchant category code.

Merchant reliability scoring methodology is governance-platform-defined and *MUST* be documented in the governance zone specification. The raw score *MUST* be present in the Merchant Manifest Document and signed by the governance platform; it *MUST NOT* be merchant-asserted.

7.2. Unified DISCOVER Queries

A requesting agent *MAY* issue a DISCOVER query with result_type: "any" to receive a mixed result set containing both Agent Manifests and Merchant Manifests. This is useful for workflows where the agent does not know in advance whether the capability it needs is best satisfied by a peer agent or by a merchant transaction (e.g., "find a service that can produce a translated legal document" where the answer might be either a translation agent or a document-services merchant).

Mixed result sets include a result_class field on each entry with value "agent" or "merchant", enabling the requesting agent to route each result to the appropriate downstream handling.

8. 455 Counterparty Unverified

8.1. Definition

This document registers AGTP status code 455 Counterparty Unverified. The status is returned in any of the following conditions:

* The Merchant-ID request header is absent on a PURCHASE invocation.

* The Merchant-Manifest-Fingerprint request header is absent or does not match the receiving server's current manifest.

* The Merchant-ID does not match the receiving server's canonical Merchant-ID.

* The agent's pre-flight counterparty verification failed (returned by the agent's own runtime before a PURCHASE is sent on the wire).

* The target merchant is in a non-Active lifecycle state.

* The Merchant Manifest Document signature is invalid.

The response body *MUST* identify the specific verification failure and *MUST* include the governance-platform-signed reason code. The requesting agent *MUST NOT* retry the PURCHASE without re-running counterparty verification against a fresh Merchant Manifest Document.

Status code 455 is a governance signal, not a protocol error, and *MUST* be logged by both parties. It parallels the role of 451 Scope Violation and 453 Zone Violation in the AGTP status code space: the system caught a governance condition at the protocol layer, before any state-modifying side effect.

8.2. Retry Semantics

A 455 response in the following categories is non-retryable without remediation:

* Merchant in Revoked or Deprecated lifecycle state.

* Invalid Merchant Manifest signature.

* Merchant-ID mismatch.

A 455 response in the following categories is retryable after a governance-defined interval:

* Merchant in Suspended state (retry after state transitions to Active).

* Fingerprint mismatch due to a legitimate manifest update (retry after re-fetching the current manifest).

The response body *MUST* declare retry-eligibility via a retryable boolean field and, where retryable, *MAY* declare a retry_after timestamp.

9. Security Considerations

9.1. Merchant Identity Forgery

Threat: An attacker publishes a Merchant Manifest Document claiming to represent a legitimate merchant under a domain the attacker does not control.

Mitigation: Trust Tier 1 registration requires DNS ownership verification per [RFC8555]. The governance platform signs every Merchant Manifest; requesting agents *MUST* verify the signature against the governance platform's published key before trusting the manifest. An unsigned manifest or one signed by an unrecognized platform *MUST* be rejected. Requesting agents *SHOULD* maintain a trust list of accepted governance platforms per governance zone.

9.2. Manifest Substitution at Purchase

Threat: A requesting agent verifies Merchant Manifest A, but the PURCHASE is received by an endpoint serving Merchant Manifest B.

Mitigation: The Merchant-Manifest-Fingerprint header binds the manifest the agent verified to the manifest the receiving server presents. A mismatch produces 455 Counterparty Unverified. This check is cryptographic and cannot be bypassed without compromising the governance platform's signing key.

9.3. Intent Assertion Replay

Threat: A captured Intent Assertion is replayed by an attacker to authorize an unintended second purchase.

Mitigation: Intent Assertions carry a unique jti and a short exp. Receiving servers *MUST* record the jti in the Attribution-Record and *MUST* reject any subsequent request carrying a previously seen jti. The recommended maximum validity is 300 seconds; implementations *MAY* apply shorter limits. Governance platforms *SHOULD* maintain a zone-scoped cache of consumed jti values for at least the maximum validity period.

9.4. Cart-Digest Collision

Threat: An attacker constructs a cart that produces the same digest as a different, higher-value cart.

Mitigation: The Cart-Digest algorithm *MUST* be a cryptographic hash function resistant to collision attacks. SHA-256 is the baseline requirement. The digest *MUST* be computed over a canonical serialization of the cart payload to prevent ambiguity between equivalent JSON representations.

9.5. Merchant Lifecycle Lag

Threat: A merchant is Revoked for fraud, but the Merchant Name Service has not yet propagated the state change. A requesting agent verifies the stale manifest and proceeds with PURCHASE.

Mitigation: Governance platforms *MUST* propagate lifecycle state changes to all indexing MNS servers within 60 seconds. MNS servers *MUST* treat Revocation as urgent deregistration and *MUST* remove the merchant from the result index before the next DISCOVER request is processed. Requesting agents with strict assurance requirements *MAY* set a maximum manifest age (e.g., re-fetch if the manifest is older than 300 seconds) before accepting it for PURCHASE.

9.6. Dispute Policy URI Tampering

Threat: A merchant publishes a dispute policy URI that redirects to a different policy after the PURCHASE is complete.

Mitigation: The dispute_policy_uri field is part of the signed Birth Certificate and is included in the signed Merchant Manifest. Requesting agents *SHOULD* retrieve and hash the dispute policy document content at verification time and include the hash in the Attribution-Record. Any subsequent change to the policy content can be detected by comparing the archived hash to the current content.

9.7. Privacy Considerations

Merchant Manifest Documents contain legal entity information and payment network acceptance declarations. This data is generally considered public commercial information and does not trigger the same privacy protections as agent or principal identity data. However, Merchant Name Service query logs reveal which agents are shopping for which kinds of goods and *MUST* be treated with the same access controls and retention limits applied to DISCOVER query logs under [AGTP-DISCOVER].

Intent Assertions contain principal identifiers, merchant identifiers, and amount ceilings. These are transactionally sensitive. Intent Assertions *MUST* be treated as confidential transport data and *MUST NOT* be logged in forms accessible outside the governance zone in which they were issued.

10. IANA Considerations

10.1. Status Code Registration

This document requests registration of the following status code in the IANA Agent Transfer Protocol Status Codes registry established by [AGTP] Section 8.3:

Code	Name	Definition	Reference
455	Counterparty Unverified	The merchant counterparty in a PURCHASE invocation failed identity verification: the Merchant-ID or Merchant-Manifest-Fingerprint is absent, does not match, or the merchant is in a non-Active lifecycle state. Governance signal; *MUST* be logged.	This document, Section 7

Table 4: Status Code 455 Registration

10.2. Header Field Registration

This document requests registration of the following header fields in the IANA Agent Transfer Protocol Header Fields registry established by [AGTP] Section 8.4:

Header	Status	Reference
Merchant-ID	Permanent	This document, Section 4.1
Merchant-Manifest-Fingerprint	Permanent	This document, Section 4.1
Intent-Assertion	Permanent	This document, Section 5
Cart-Digest	Permanent	This document, Section 6

Table 5: Header Field Registrations

10.3. Authority-Scope Domain Registration

This document requests the addition of the following domains to the reserved Authority-Scope domain set defined in [AGTP] Appendix A:

Domain	Description
merchant	Merchant identity resolution and counterparty verification
intent	Intent Assertion issuance and validation

Table 6: Authority-Scope Domain Registrations

Actions defined for these domains:

* merchant:query -- Resolving a Merchant URI and retrieving a Merchant Manifest Document.

* merchant:verify -- Performing counterparty verification against a Merchant Manifest Document as part of PURCHASE pre-flight.

* intent:assert -- Issuing an Intent Assertion JWT on behalf of the principal.

10.4. Document Type Registrations

The following document types are defined for use with the application/agtp+json Content-Type:

Document Type	Description	Reference
merchant-birth-certificate	Genesis record of a merchant entity	This document, Section 3.1
agtp-merchant-manifest	Wire-level merchant identity document	This document, Section 3.4

Table 7: AGTP Document Type Registrations

11. References

11.1. Normative References

[AGTP] Hood, C., "Agent Transfer Protocol (AGTP)", Work in Progress, Internet-Draft, draft-hood-independent-agtp-04, 2026, <https://datatracker.ietf.org/doc/html/draft-hood-independent-agtp-04>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/rfc/rfc7519>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/rfc/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <https://www.rfc-editor.org/rfc/rfc8555>.

11.2. Informative References

[AGTP-CERT]
Hood, C., "AGTP Agent Certificate Extension", Work in Progress, Internet-Draft, draft-hood-agtp-agent-cert-00, 2026, <https://datatracker.ietf.org/doc/html/draft-hood-agtp-agent-cert-00>.

[AGTP-DISCOVER]
Hood, C., "AGTP Agent Discovery and Name Service", Work in Progress, Internet-Draft, draft-hood-agtp-discovery-00, 2026, <https://datatracker.ietf.org/doc/html/draft-hood-agtp-discovery-00>.

[AGTP-METHODS]
Hood, C., "AGTP Standard Extended Method Vocabulary", Work in Progress, Internet-Draft, draft-hood-agtp-standard-methods-00, 2026, <https://datatracker.ietf.org/doc/html/draft-hood-agtp-standard-methods-00>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC9334] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <https://www.rfc-editor.org/rfc/rfc9334>.

Appendix A. Deployment Considerations

A.1. Governance Platform Scope

A governance platform operating an AGTP registry *MAY* extend its registry to cover both agents and merchants, or it *MAY* operate separate agent and merchant registries under the same governance zone. The registry schema for merchants is structurally parallel to the agent registry schema, reducing implementation effort.

A.2. MNS Co-location

Merchant Name Service functionality *MAY* be co-located with an existing Agent Name Service, particularly for governance zones where the agent-to-merchant ratio is low. In this case, the DISCOVER method serves both result types through the result_type parameter. The same access control, rate limiting, and federation semantics apply.

A.3. Payment Network Integration Path

This specification is designed to be consumable by payment networks without requiring those networks to implement AGTP. The Intent Assertion is a standalone JWT verifiable with only the governance platform’s public key; the merchant identity attestation is a signed JSON document verifiable with the same key. Payment networks wishing to extend protection or dispute handling to agent-initiated transactions *MAY* consume these artifacts as inputs to their existing authorization and dispute message flows without protocol-level integration with AGTP itself.

The specific mapping of Intent Assertion claims to payment network authorization message fields, and of Attribution-Record content to dispute evidence formats, is expected to be defined bilaterally between governance platforms and individual payment networks. Those mappings are out of scope for this document.

Appendix B. Acknowledgments

The author thanks the American Express Agentic Commerce Experiences working group for public specification of the commercial requirements that motivated this document. The structural parallelism between agent identity and merchant identity in AGTP owes its clarity to the Amex ACE five-service model: agent registration, account enablement, intent intelligence, payment credentials, and cart context. This document addresses the transport-layer identity gap that complements the payment-rail work described there.

Author's Address

Chris Hood
Nomotic, Inc.
Email: chris@nomotic.ai
URI: https://nomotic.ai