A2AX-Core Protocol
Informational Overview

Williams Creative

Companion Document
Version 0.1.3

Abstract

This document provides an informational overview of the A2AX-Core Protocol. It is not normative. The normative specification is defined in the A2AX-Core Protocol Specification (003).

Contents

1 Purpose

A2AX-Core defines a neutral trust substrate for autonomous agents. It enables portable identity, verifier-controlled trust decisions, and secure agent-to-agent coordination without centralized registries or embedded authority.

This document is informational. The normative specification is defined in the A2AX-Core Protocol Specification.

2 Why A2AX-Core Exists

Autonomous agents increasingly coordinate across systems, organizations, and jurisdictions. Intelligence alone does not ensure safe interaction. A portable and cryptographically verifiable trust layer is required.

A2AX-Core addresses this gap by defining minimal, implementation-neutral trust primitives.

3 Core Properties

- Portable identity certificates
- Verifier-controlled trust model
- No mandatory trust anchors
- No centralized registry requirement
- Capability-scoped permissions
- Secure agent-to-agent handshake

The protocol is infrastructure—not marketplace, governance system, or economic layer.

4 Layered Architecture

Application / Agent Layer
↓
A2AX-Core Protocol Layer (Identity + Verification + Handshake)
↓
Verifier Trust Policy Layer
↓
Optional Extensions (Economic, Compliance, Analytics)

The core remains minimal and neutral.

5 What A2AX-Core Is Not

A2AX-Core does not define:

- Token systems
- Market mechanisms
- Economic valuation
- Governance frameworks

- Centralized infrastructure

Higher-order systems may build on A2AX-Core but remain external to it.

6 Neutrality Principle

Trust is determined exclusively by the verifier.

No global root of trust exists.

No organization controls protocol validity.

Neutrality is structural, not rhetorical.

7 Intended Audience

- Protocol implementers
- Multi-agent system architects
- Infrastructure engineers
- Standards and governance bodies

8 Relationship to the Specification

The A2AX-Core Protocol Specification defines normative requirements using formal conformance language (MUST, SHOULD, etc.).

This overview provides architectural context and positioning clarity.

9 Long-Term Vision

A2AX-Core aims to serve as a portable, minimal trust standard for autonomous agent coordination across ecosystems.

Its durability depends on neutrality, verifier sovereignty, and strict scope boundaries.

10 Threat Model (Informational Summary)

A2AX-Core assumes the presence of capable adversaries operating at the network, agent, and ecosystem levels.

The protocol is designed to mitigate:

- Certificate forgery attempts
- Message replay attacks
- Capability escalation attempts
- Malicious but cryptographically valid agents
- Centralization pressure through mandatory trust anchors

A2AX-Core does **not** attempt to mitigate:

- Economic fraud

- Marketplace manipulation
- Behavioral dishonesty
- Global revocation enforcement

The protocol guarantees cryptographic authenticity, not behavioral integrity.

11 Adversary Classes

Class A — Network Adversary

May intercept, replay, or inject traffic.

Class B — Malicious Certified Agent

Possesses valid credentials but behaves dishonestly.

Class C — Compromised Key Holder

Private key material has been exposed.

Class D — Centralization Actor

Attempts to impose mandatory trust anchors or ecosystem control.

Mitigation across classes is achieved through signature validation, nonce enforcement, expiration limits, and strict verifier sovereignty.

12 Separation of Concerns

A2AX-Core enforces structural separation between four distinct concepts.

12.1 Identity

Cryptographic binding between an agent identifier and a public key.

Answers: *Who signed this message?*

12.2 Capability

Declared functional scope cryptographically bound to identity.

Answers: *What does this agent claim it can do?*

Capabilities are claims, not guarantees.

12.3 Trust

A verifier-local decision derived from policy.

Answers: *Do I accept interaction under my rules?*

Trust is never embedded in the certificate.

12.4 Reputation (Out of Scope)

Accumulated behavioral or economic history over time.

Answers: *How has this agent behaved historically?*

Reputation is explicitly external to A2AX-Core and MUST NOT be embedded within the core protocol layer.