THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
NORTH AMERICA

Embedded Linux Conference

Building Trust in the AI Era

Agent-to-Agent Communication with DIDs and VCs

Alexander Shcherbakov · DSR Corporation

Agenda

1. **A2A - Beyond OAuth: the case for verifiable trust**
   Just in time, delegation, qualifications, cross-org, wallet interop, etc.
2. **A primer on DIDs, VCs, and OID4VP**
   the identity world already solved this
3. **Two patterns of VCs for agents**
   human-delegated chains vs. step-up consent
4. **The OID4VP In-Task Auth extension**
   how it plugs into A2A — the flow, end to end
5. **Live demo — Heka Identity × A2A**
   open source, Apache-2.0, on Hiero / LFDT
6. **The wider landscape & what's next**
   AP2, Verifiable Intent, dSD-JWT

AI is becoming a fabric of agents — not just chat

- Agents that book, buy, negotiate, file, and sign — autonomously.
- They don't just talk to humans. They talk to each other.
- The "agentic web" is no longer a metaphor. It's a runtime.
- **A2A: Agent2Agent — open protocol, donated to the Linux Foundation in 2025.**

Agents talking to agents — across orgs and domains

A2A — high-level overview

Agent2Agent — open protocol for agents to discover each other, talk, and collaborate.

Agent Card

JSON manifest for discovery: name, capabilities, endpoint, security, extensions.

Tasks

Stateful units of work: submitted → working → completed (long-running, resumable).

Messages & Artifacts

Structured payloads in / structured results out, attached to a task.

Streaming

Server-Sent Events for incremental status updates and partial results.

Security schemes

OAuth 2.0 · OIDC · mTLS · API key — declared in the Agent Card.

Extensions

Capability plug-points — where this talk lives.

Originated at Google · governed by the Linux Foundation · backed by Microsoft · Cisco · AWS · Salesforce · ServiceNow · SAP · IBM · Auth0 · 100+ others

Where OAuth-only A2A hits the wall

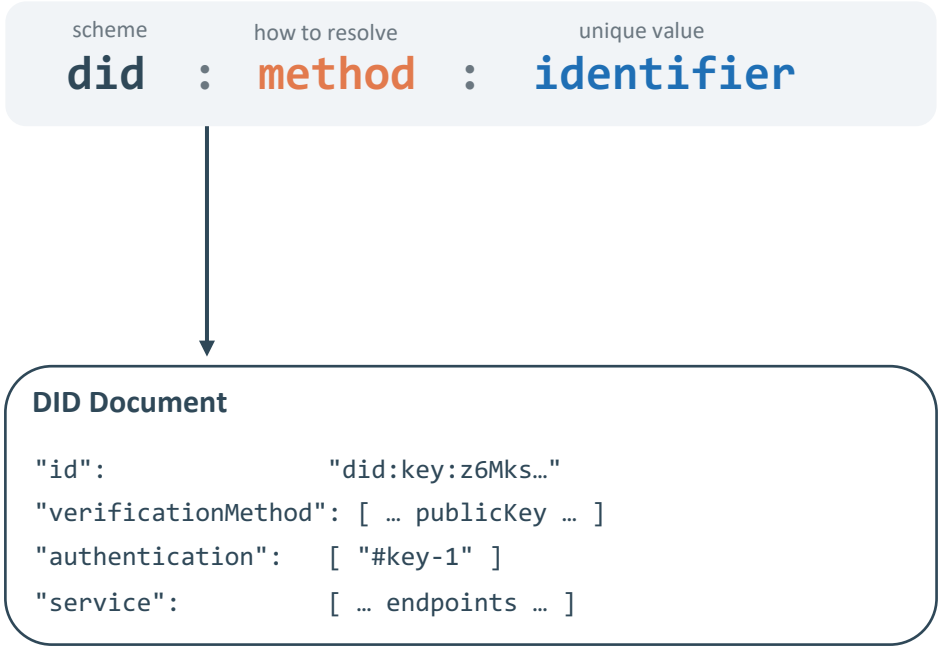
A2A inherits OAuth's limits — and they show up at runtime.

What OAuth gives A2A today	What agents actually need
Permissions — `scope: tasks.read`. The agent is *allowed* to call.	**Qualifications** — "The user has passed KYC/AML checks," signed by a third party.
Centralized issuer — one Authorization Server is the single source of truth.	**Third-party trust** — accept proofs from Governments, Regulators, Travel Associations — no federation.
Static trust — authorization bound at connection start; scopes fixed.	**Just-in-time step-up** — *this specific task* is approved by the user.
No wallet interop — tokens issued only by the Authorization Server.	**Wallets users already have** — consume VCs from Apple / Google / EUDI.

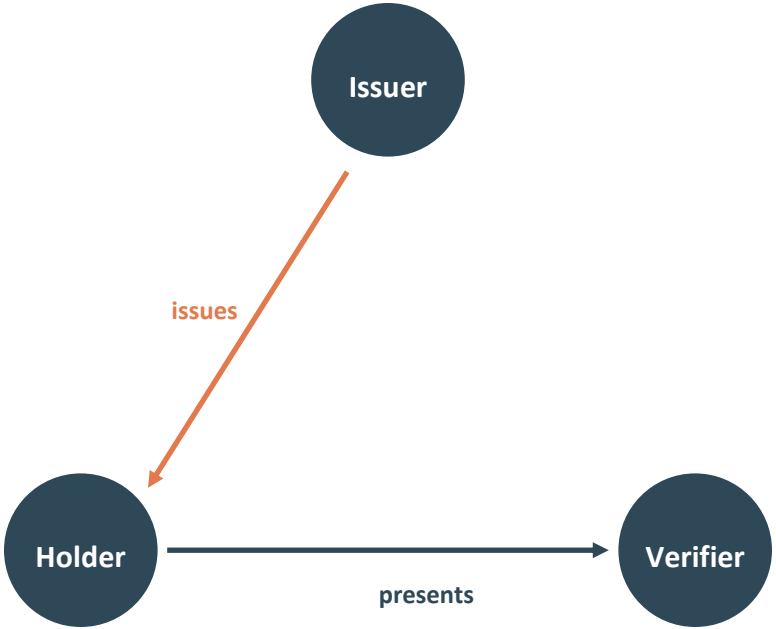
Tokens vs. credentials. Different problem. Different shape.

Primer 1/3: Decentralized Identifiers (DIDs)

- **An identifier you control — not a platform username.**
- Globally unique: did:key, did:peer, did:web, did:hedera, etc.
- Resolves to a DID Document — public keys + service endpoints.
- No central issuer. No takedown. Pure cryptography.
- W3C standard since 2022.



Primer 2/3: Verifiable Credentials (VCs)



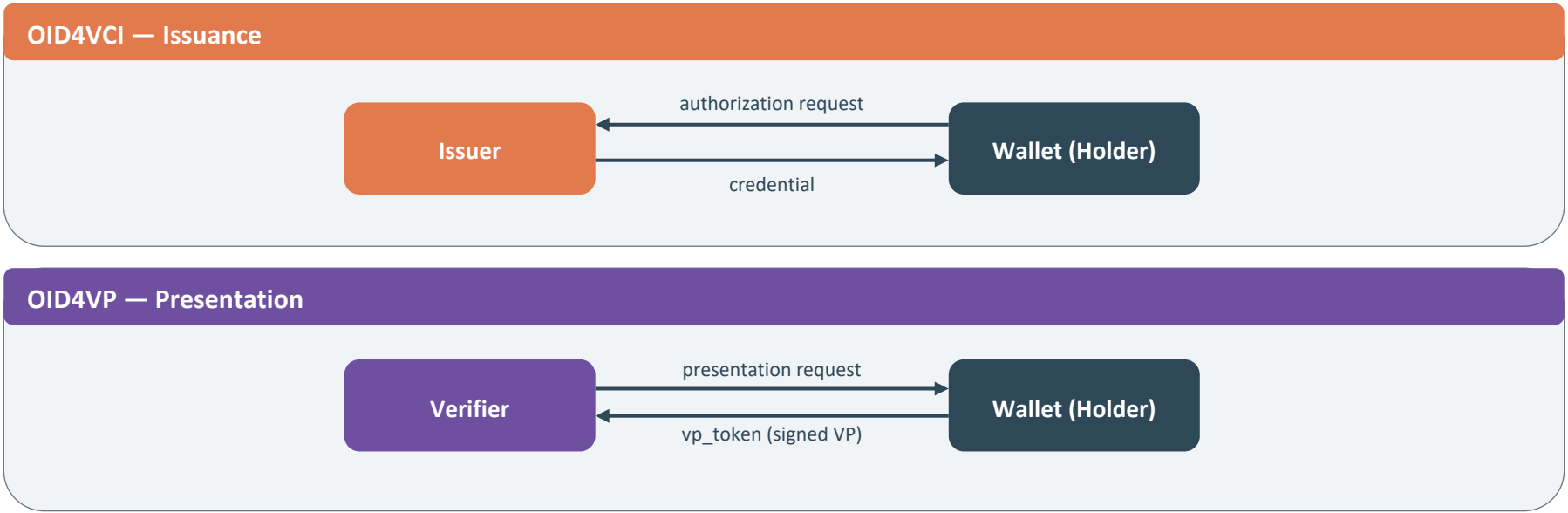
- **A digitally signed claim — like a tamper-evident ID card.**
- Issuer signs · Holder carries · Verifier checks the signature.
- No call-back to the Issuer at verification time.
- Selective disclosure: prove "over 21" without revealing your birthday.

Common modern formats:

`SD-JWT VC` `W3C VC 2.0` `ISO mDoc`

Primer 3/3: OpenID for Verifiable Credentials — the protocol layer

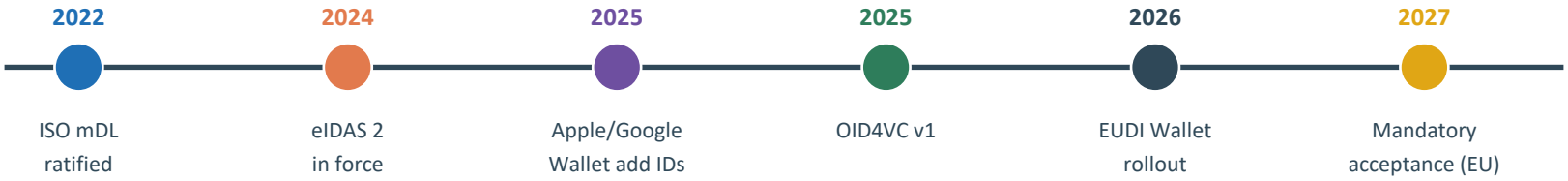
Built on OAuth 2.0. Plain HTTP. Wallets, banks, and governments already speak it.



Two flows. Plain HTTP. JSON in, JSON out. No new transport.

Why this is real now — not a 2030 thing

Decentralized identity has left the lab. It's shipping at population scale.



Standards are settled.

W3C VC 2.0 · DIDs 1.0
OID4VC v1.0 · SD-JWT VC · ISO mDoc

Wallets are in pockets.

Apple · Google · EUDI
Heka · OWF Bifold

Issuers are governments & banks.

13+ US states ship mDL
every EU member by 2026

Two patterns of VCs for AI agents

Both patterns reuse the same DID/VC/OID4VP plumbing — only the wallet UX differs.

A · Human-delegated chain — agent presents autonomously

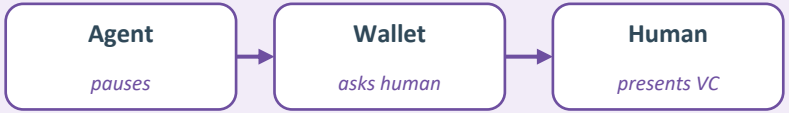
Bank → Human → Agent:

- Bank issues a KYC VC to the human.
- Human signs a delegation VC binding the agent's DID/key.
- **Agent presents the chain mid-task — no human interaction needed.**

Use case: high-volume, pre-authorized actions.

B · Step-up: agent asks for human consent

Agent pauses task, hands off to human wallet:



- Agent hits a high-stakes step.
- A2A task pauses; OID4VP request goes to human's wallet (e.g., EUDI).
- **Human approves, presents KYC / Age VC fresh.**

Use case: high-stakes, regulated, or one-off actions.

A2A Extension: OID4VP In-Task Auth

The first extension which is part of the LF A2A org itself.

A2A Protocol

Tasks · Streaming · AgentCards

Extension mechanism — designed to plug in.

+

OID4VP

AuthorizationRequest · vp_token

SD-JWT VC · W3C VC 2.0 · ISO mDoc.

Pause an A2A task. Ask for a credential. Resume.

`Apache-2.0 · https://github.com/a2aproject/experimental-ext-oid4vp-auth`

Under the hood — A2A messages + OID4VP

AgentCard — declare the extension

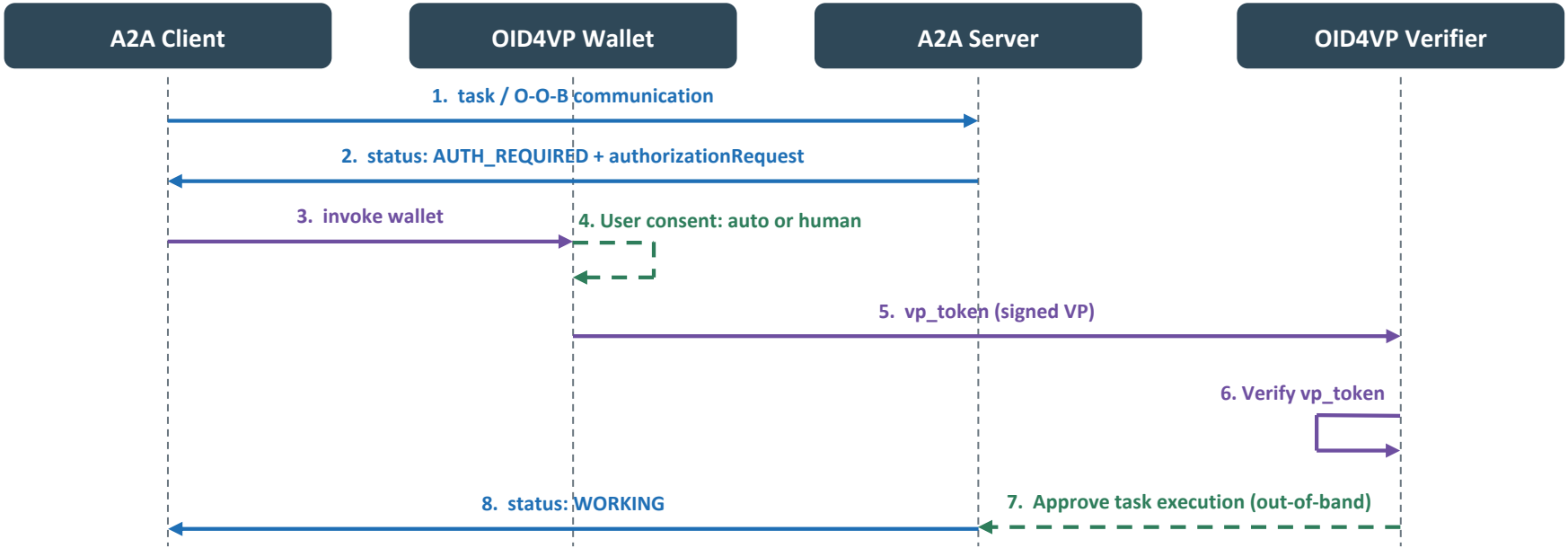
```
"capabilities": {
  "extensions": [{
    "uri": "github.com/a2aproject/oid4vp-auth/.../v1",
    "params": {
      "oid4vpVersions": ["1.0"]
    },
    "required": false
  }]
}
```

Mid-task message — carry the auth request

```
"metadata": {
  " github.com/a2aproject/oid4vp-auth/.../v1 ": {
    "authorizationRequest": {
      "client_id": "verifier.example",
      "request_uri": "https://…/req/abc"
    }
  }
}
```

Task lifecycle: submitted → **auth-required** → working → completed

The flow — step by step

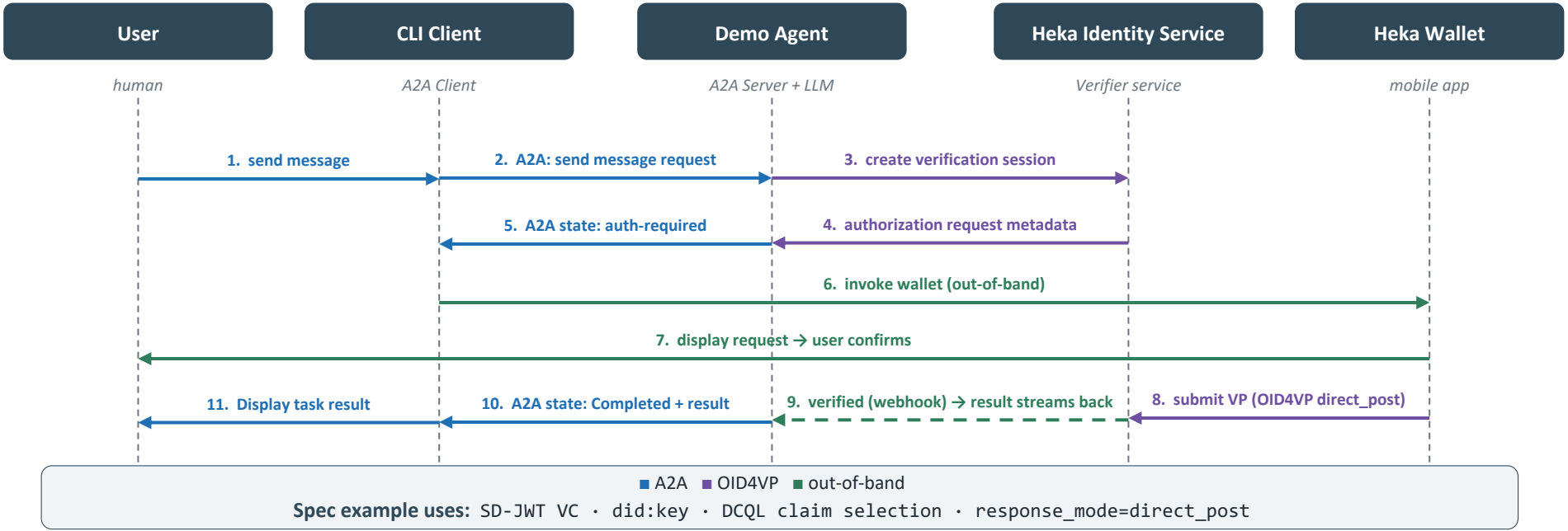


Live demo — Heka Identity × A2A

Demo Video Link

Heka demo — flow, step by step

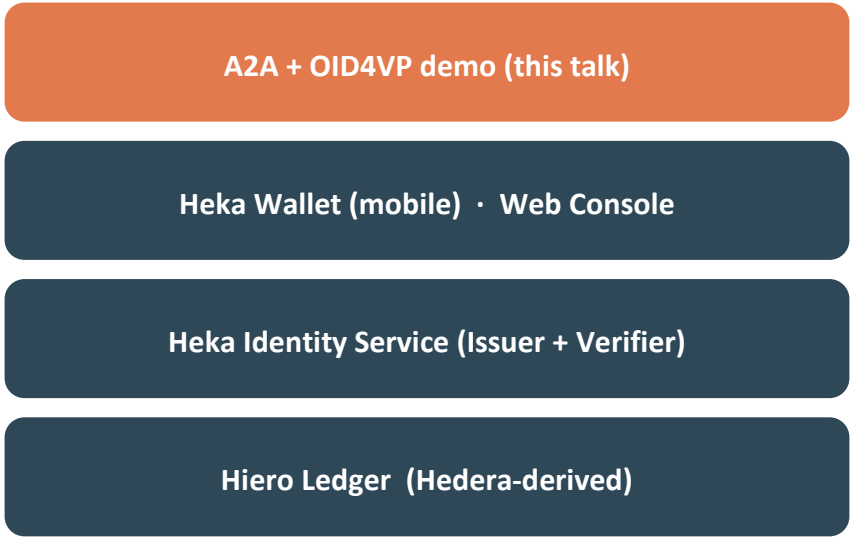
`github.com/hiero-ledger/heka-identity-platform/demo/a2a-oid4vp`



Why this lives in LFDT Hiero / Heka

- **Open standards aren't enough — need real, deployable code.**
- Heka: enterprise-grade SSI platform contributed by DSR.
- A subproject of Hiero, an LFDT project, Apache 2.0
- Mobile wallet + Issuer/Verifier services + Web UI
- Multiple VC/DID standards supported

Implements emerging agent-trust standards.



Linux Foundation Decentralized Trust · Hiero · Heka

The wider landscape — convergent work

Different protocols. Same primitives.

A2A + OID4VP-in-task

agent ⇄ agent step-up auth

AP2 v0.2 — Agent Payments Protocol

Checkout & Payment Mandates as SD-JWT VCs

Google → FIDO Alliance · 100+ partners

Verifiable Intent

Tamper-proof log of user-authorized agent actions

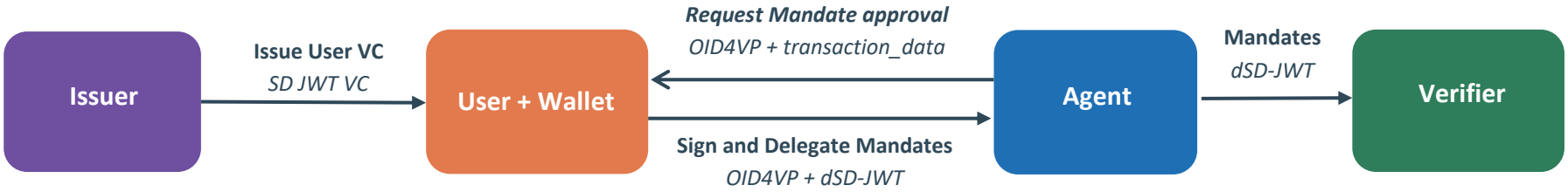
Mastercard · AP2-compatible → **FIDO Alliance**

↓ *built on* ↓

OID4VP / OID4VCI	Verifiable Presentation / Issuance flows	**OpenID Foundation**
SD-JWT VC	Selective-disclosure VC format	**IETF**

AP2 & Verifiable Intent Protocols – Agentic Payments

AP2 mandates and Verifiable-Intent delegations— over the same OID4VP / SD-JWT rails.



- **AP2 v0.2 — Checkout + Payment Mandates** as SD-JWT VCs.
- **dSD-JWT —** IETF draft (draft-gco-oauth-delegate-sd-jwt) — selective disclosure for delegation tokens.
- Delegation via **OpenID4VP transaction_data** + dSD-JWT for selective-disclosure key binding.
- Two delegation models: User Credential (external Issuer) · Trusted Agent Provider.

What this unlocks

Regulated automation

Healthcare, legal, finance agents that carry their license forward into every task.

Cross-org workflows

Two companies' agents transact without bilateral OAuth — they share trusted issuers (regulators, governments).

User-bound delegation

"Acting on Alice's signed delegation, scope: economy class, budget €2k, dates fixed."

Audit trails that hold

Every privileged step has a signed VP attached. Who authorized what, when — provable.

No new crypto. Shared standards. Real buyers today.

The talk in 60 seconds

1. **A2A - Beyond OAuth: the case for verifiable trust**
   Just in time, delegation, qualifications, cross-org, wallet interop, etc.
2. **The identity world already solved this.**
   DIDs · VCs · OID4VP · mature wallets — at population scale.
3. **Two patterns of VCs for agents.**
   Human-delegated chain (autonomous) · step-up consent (human in the loop).
4. **A small, surgical A2A extension.**
   Pause the task. Ask for a credential. Resume — without leaving the protocol.
5. **Heka × A2A proves it works.**
   Open source, Apache-2.0, on Hiero · LFDT.
6. **AP2 and Verifiable Intent Protocols; Agentic Payment**
   Same rails and standards; OID4VP compatible

Trust at agent speed — across orgs — without inventing a new closed registry.

Get involved

Spec & Sample

github.com/a2aproject/experimental-ext-oid4vp-auth

LFDT Hiero Heka

github.com/hiero-ledger/heka-identity-platform/

A2A Protocol

github.com/a2aproject/A2A

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
NORTH AMERICA

Embedded Linux Conference

Trust isn't a feature.
It's a layer.

Q&A

Alexander Shcherbakov · DSR Corporation ·