SICPA

SICPA

Confidentiality note

All information and material contained in these pages, including text, layout, presentation, logos, icons, photos, processes, data and all other artwork including – but not limited to – any derivative works are business sensitive and confidential information and/or information and material protected by patents, designs, trademarks or copyrights in the name of SICPA or any of its affiliates and shall be kept strictly confidential.

The material and information contained in – or derived from – these pages may therefore not be copied, exploited, disclosed or otherwise disseminated, in whole or in part, without SICPA's prior written approval.

SICPA Digital Identity

Overview

Presentation to:
HL Identity Implementors WG

Presented by:
Xavier Vila & Victor Martinez

Company: SICPA

September 23rd, 2021

PUBLIC

SICPA

Enabling trust

SICPA

Agenda

1

About SICPA

2

SICPA’s contribution

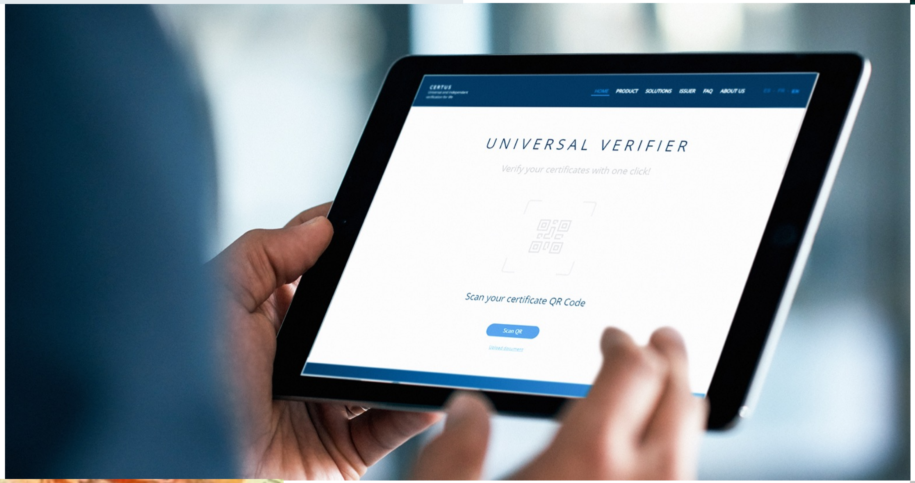
3

SICPA’s Edison project

We **protect** and **trace** valuable goods and data

140 Billion Banknotes secured annually in over **160 countries.**

Leading provider of **Proofs of Provenance, Integrity, Authentication and Presence**


UNIVERSAL VERIFIER
Verify your certificates with one click!
Scan your certificate QR Code
Scan QR

80+ Billion

Products marked and traced annually

A long-trusted advisor to governments, central banks, high security printers and industry

SICPA's SSI initiative so far

Digital credential platform (Edison)

- A standards-based, interoperable building block for verifiable data to increase assurance and trust of information exchanged between parties in a peer-to-peer and privacy-preserving way.

Comprising :

1. A system to issue, manage and verify decentralized identifiers and portable verifiable data
2. A secure communications protocol to exchange information (DIDcomm)
3. An extensible verification toolkit for online and offline use

Powered by ACA-py

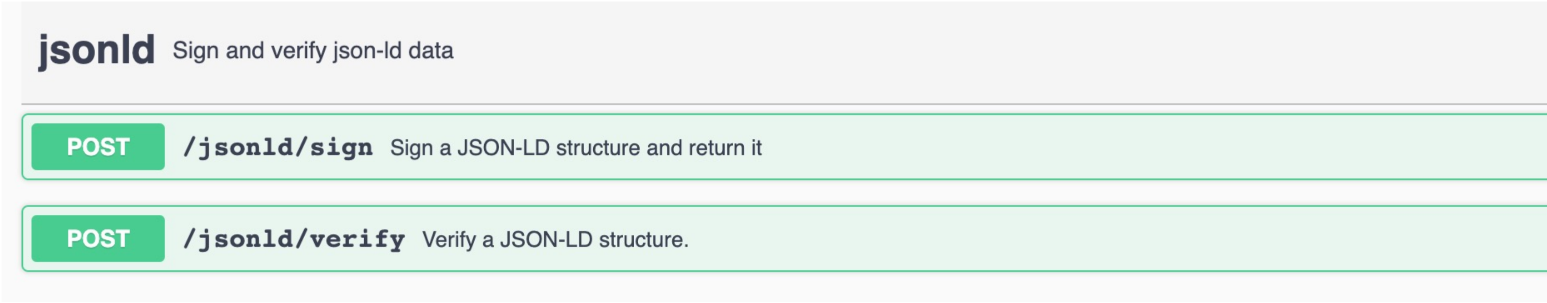
SICPA contributions to ACA-Py

- JSON-LD verifiable credentials
- DID resolver interface and plugins
- Contribution to Mediator
- Multi-tenant agency
- Kafka for async messaging

JSON-LD Verifiable Credentials

- Implemented as part of the DHS SVIP program by SICPA

- Created an initial component (PoC) that was further evolved by BCGov / Animo and included in ACA-Py



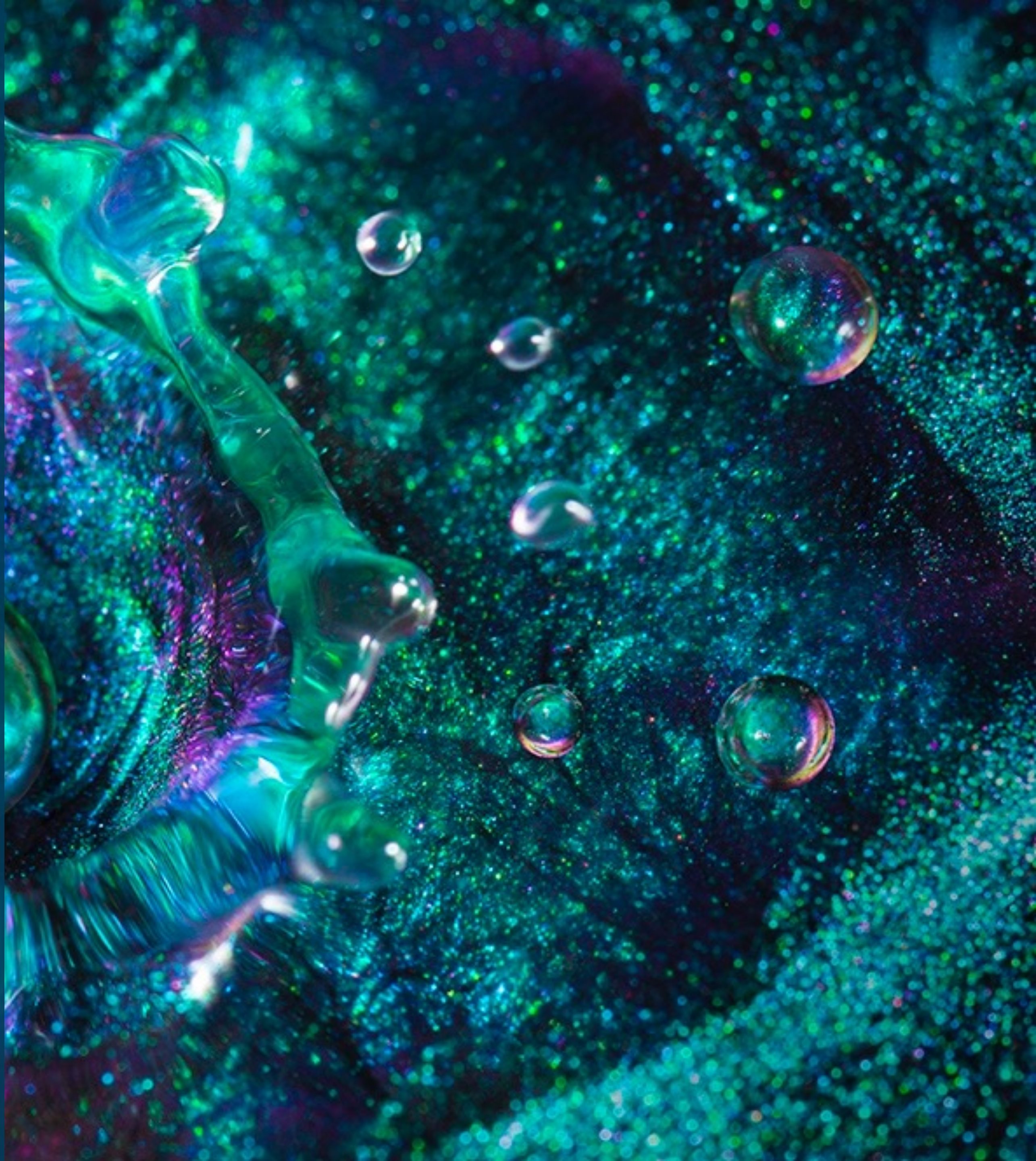
jsonld Sign and verify json-ld data

POST `/jsonld/sign` Sign a JSON-LD structure and return it

POST `/jsonld/verify` Verify a JSON-LD structure.

SICPA

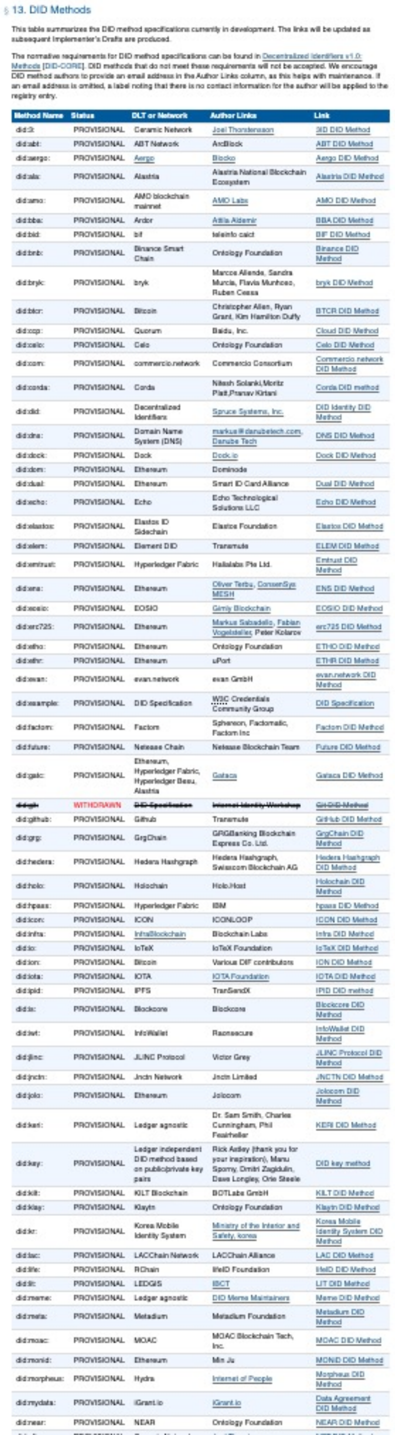
DID resolver



Motivation

–

- DIDs are everywhere: the number of DID methods is constantly growing, also we don’t want be locked-in to any single DID method.
- We want to leverage in ACA-Py the addition of JSON-LD credentials (plain and BBS+)
- https://github.com/hyperledger/aries-rfcs/tree/main/features/0124-did-resolution-protocol



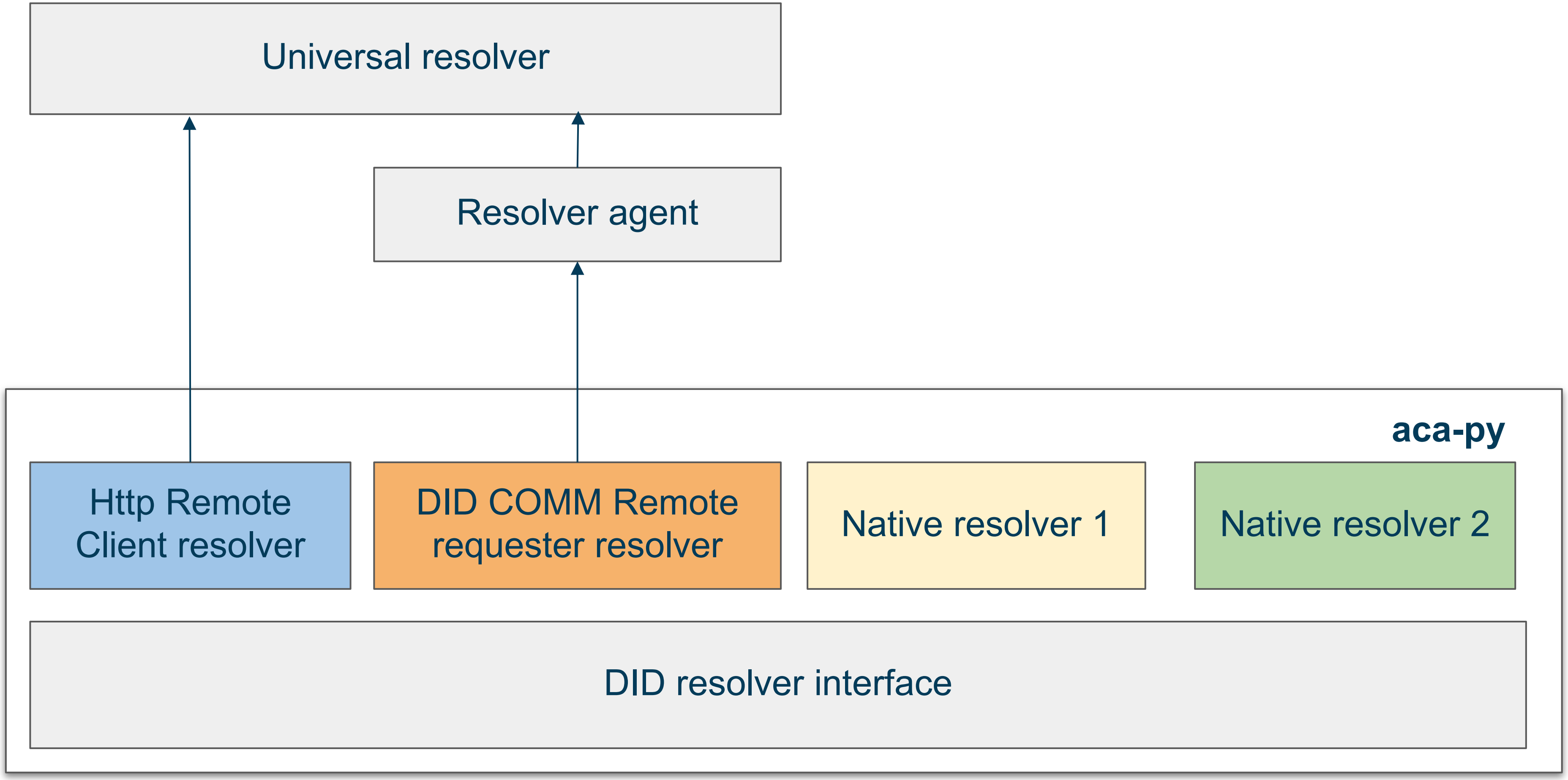
§ 13. DID Methods

This table summarizes the DID method specifications currently in development. The links will be updated as subsequent Implementer's Drafts are produced.

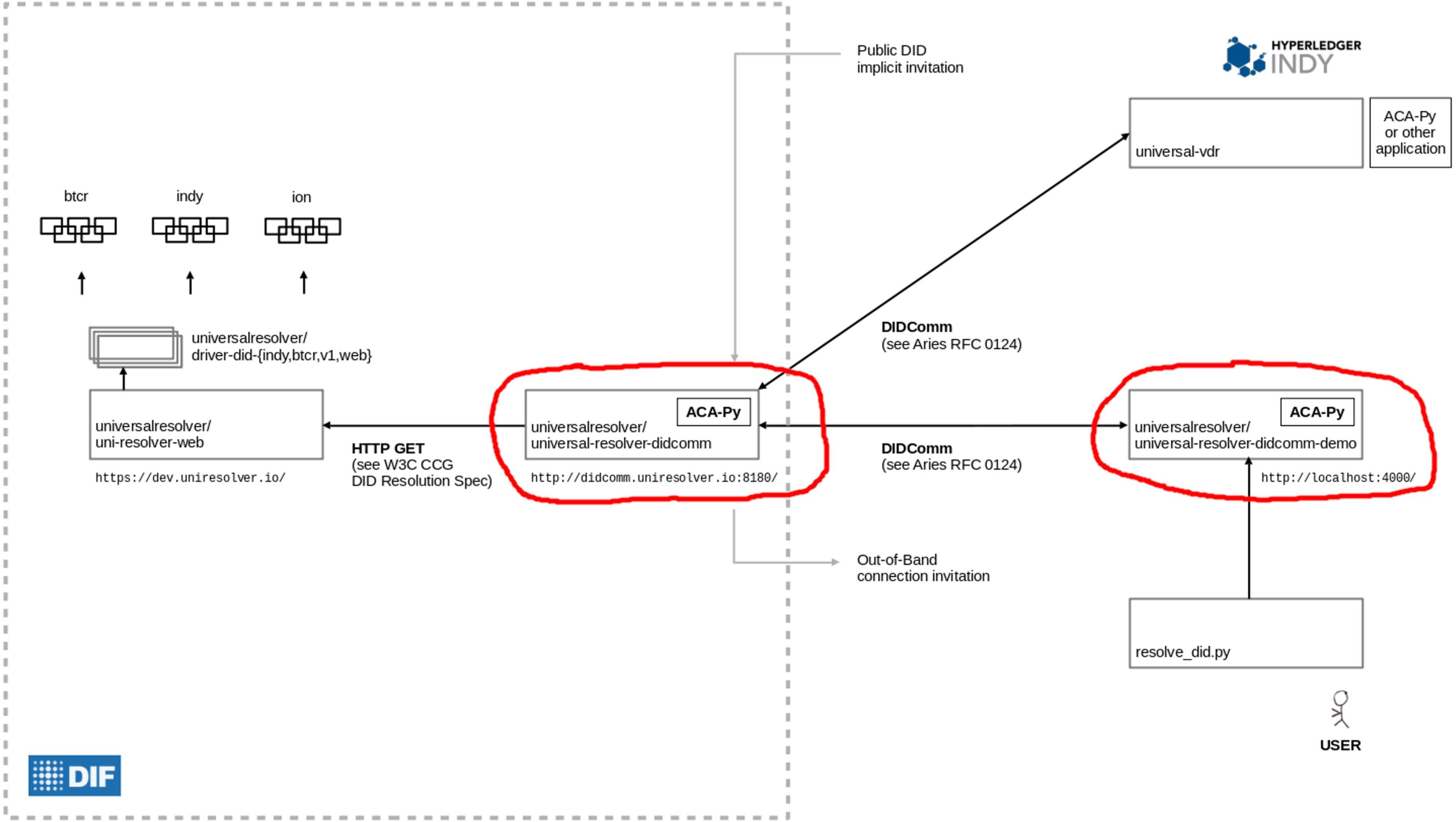
The normative requirements for DID method specifications can be found in Decentralized Identifiers v1.0: Methods [DID-CORE]. DID methods that do not meet these requirements will not be accepted. We encourage DID method authors to provide an email address in the Author Links column, as this helps with maintenance. If an email address is omitted, a label noting that there is no contact information for the author will be applied to the registry entry.

Method Name	Status	DLT or Network	Author Links	Link
did:3:	PROVISIONAL	Ceramic Network	Joel Thorstensson	3ID DID Method
did:abt:	PROVISIONAL	ABT Network	ArcBlock	ABT DID Method
did:aergo:	PROVISIONAL	Aergo	Blocko	Aergo DID Method
did:ala:	PROVISIONAL	Alastria	Alastria National Blockchain Ecosystem	Alastria DID Method
did:amo:	PROVISIONAL	AMO blockchain mainnet	AMO Labs	AMO DID Method
did:bba:	PROVISIONAL	Ardor	Attila Aldemir	BBA DID Method
did:bid:	PROVISIONAL	bif	teleinfo caict	BIF DID Method
did:bnb:	PROVISIONAL	Binance Smart Chain	Ontology Foundation	Binance DID Method
did:bryk:	PROVISIONAL	bryk	Marcos Allende, Sandra Murcia, Flavia Munhoso, Ruben Cessa	bryk DID Method
did:btcr:	PROVISIONAL	Bitcoin	Christopher Allen, Ryan Grant, Kim Hamilton Duffy	BTCR DID Method
did:ccp:	PROVISIONAL	Quorum	Baidu, Inc.	Cloud DID Method
did:celo:	PROVISIONAL	Celo	Ontology Foundation	Celo DID Method
did:com:	PROVISIONAL	commercio.network	Commercio Consortium	Commercio.network DID Method
did:corda:	PROVISIONAL	Corda	Nitesh Solanki,Moritz Platt,Pranav Kirtani	Corda DID method
did:did:	PROVISIONAL	Decentralized Identifiers	Spruce Systems, Inc.	DID Identity DID Method
did:dns:	PROVISIONAL	Domain Name System (DNS)	markus@danubetech.com, Danube Tech	DNS DID Method
did:dock:	PROVISIONAL	Dock	Dock.io	Dock DID Method
did:dom:	PROVISIONAL	Ethereum	Dominode	
did:dual:	PROVISIONAL	Ethereum	Smart ID Card Alliance	Dual DID Method
did:echo:	PROVISIONAL	Echo	Echo Technological Solutions LLC	Echo DID Method
did:elastos:	PROVISIONAL	Elastos ID Sidechain	Elastos Foundation	Elastos DID Method
did:elem:	PROVISIONAL	Element DID	Transmute	ELEM DID Method
did:emtrust:	PROVISIONAL	Hyperledger Fabric	Halialabs Pte Ltd.	Emtrust DID Method
did:ens:	PROVISIONAL	Ethereum	Oliver Terbu, ConsenSys MESH	ENS DID Method
did:eosio:	PROVISIONAL	EOSIO	Gimly Blockchain	EOSIO DID Method
did:erc725:	PROVISIONAL	Ethereum	Markus Sabadello, Fabian Vogelsteller, Peter Kolarov	erc725 DID Method
did:etho:	PROVISIONAL	Ethereum	Ontology Foundation	ETHO DID Method
did:ethr:	PROVISIONAL	Ethereum	uPort	ETHR DID Method
did:evan:	PROVISIONAL	evan.network	evan GmbH	evan.network DID Method
did:example:	PROVISIONAL	DID Specification	W3C Credentials Community Group	DID Specification
did:factom:	PROVISIONAL	Factom	Sphereon, Factomatic, Factom Inc	Factom DID Method
did:future:	PROVISIONAL	Netease Chain	Netease Blockchain Team	Future DID Method
did:gatc:	PROVISIONAL	Ethereum, Hyperledger Fabric, Hyperledger Besu, Alastria	Gataca	Gataca DID Method
~~did:git:~~	WITHDRAWN	~~DID Specification~~	~~Internet Identity Workshop~~	~~Git DID Method~~
did:github:	PROVISIONAL	Github	Transmute	GitHub DID Method
did:grg:	PROVISIONAL	GrgChain	GRGBanking Blockchain Express Co. Ltd.	GrgChain DID Method
did:hedera:	PROVISIONAL	Hedera Hashgraph	Hedera Hashgraph, Swisscom Blockchain AG	Hedera Hashgraph DID Method
did:holo:	PROVISIONAL	Holochain	Holo.Host	Holochain DID Method
did:hpass:	PROVISIONAL	Hyperledger Fabric	IBM	hpass DID Method
did:icon:	PROVISIONAL	ICON	ICONLOOP	ICON DID Method
did:infra:	PROVISIONAL	InfraBlockchain	Blockchain Labs	Infra DID Method
did:io:	PROVISIONAL	IoTeX	IoTeX Foundation	IoTeX DID Method
did:ion:	PROVISIONAL	Bitcoin	Various DIF contributors	ION DID Method
did:iota:	PROVISIONAL	IOTA	IOTA Foundation	IOTA DID Method
did:ipid:	PROVISIONAL	IPFS	TranSendX	IPID DID method
did:is:	PROVISIONAL	Blockcore	Blockcore	Blockcore DID Method
did:iwt:	PROVISIONAL	InfoWallet	Raonsecure	InfoWallet DID Method
did:jlinc:	PROVISIONAL	JLINC Protocol	Victor Grey	JLINC Protocol DID Method
did:jnctn:	PROVISIONAL	Jnctn Network	Jnctn Limited	JNCTN DID Method
did:jolo:	PROVISIONAL	Ethereum	Jolocom	Jolocom DID Method
did:keri:	PROVISIONAL	Ledger agnostic	Dr. Sam Smith, Charles Cunningham, Phil Feairheller	KERI DID Method
did:key:	PROVISIONAL	Ledger independent DID method based on public/private key pairs	Rick Astley (thank you for your inspiration), Manu Sporny, Dmitri Zagidulin, Dave Longley, Orie Steele	DID key method
did:kilt:	PROVISIONAL	KILT Blockchain	BOTLabs GmbH	KILT DID Method
did:klay:	PROVISIONAL	Klaytn	Ontology Foundation	Klaytn DID Method
did:kr:	PROVISIONAL	Korea Mobile Identity System	Ministry of the Interior and Safety, korea	Korea Mobile Identity System DID Method
did:lac:	PROVISIONAL	LACChain Network	LACChain Alliance	LAC DID Method
did:life:	PROVISIONAL	RChain	lifeID Foundation	lifeID DID Method
did:lit:	PROVISIONAL	LEDGIS	IBCT	LIT DID Method
did:meme:	PROVISIONAL	Ledger agnostic	DID Meme Maintainers	Meme DID Method
did:meta:	PROVISIONAL	Metadium	Metadium Foundation	Metadium DID Method
did:moac:	PROVISIONAL	MOAC	MOAC Blockchain Tech, Inc.	MOAC DID Method
did:monid:	PROVISIONAL	Ethereum	Min Ju	MONID DID Method
did:morpheus:	PROVISIONAL	Hydra	Internet of People	Morpheus DID Method
did:mydata:	PROVISIONAL	iGrant.io	iGrant.io	Data Agreement DID Method
did:near:	PROVISIONAL	NEAR	Ontology Foundation	NEAR DID Method

High level architecture



Universal Resolver DIDComm Agent



Method Resolvers

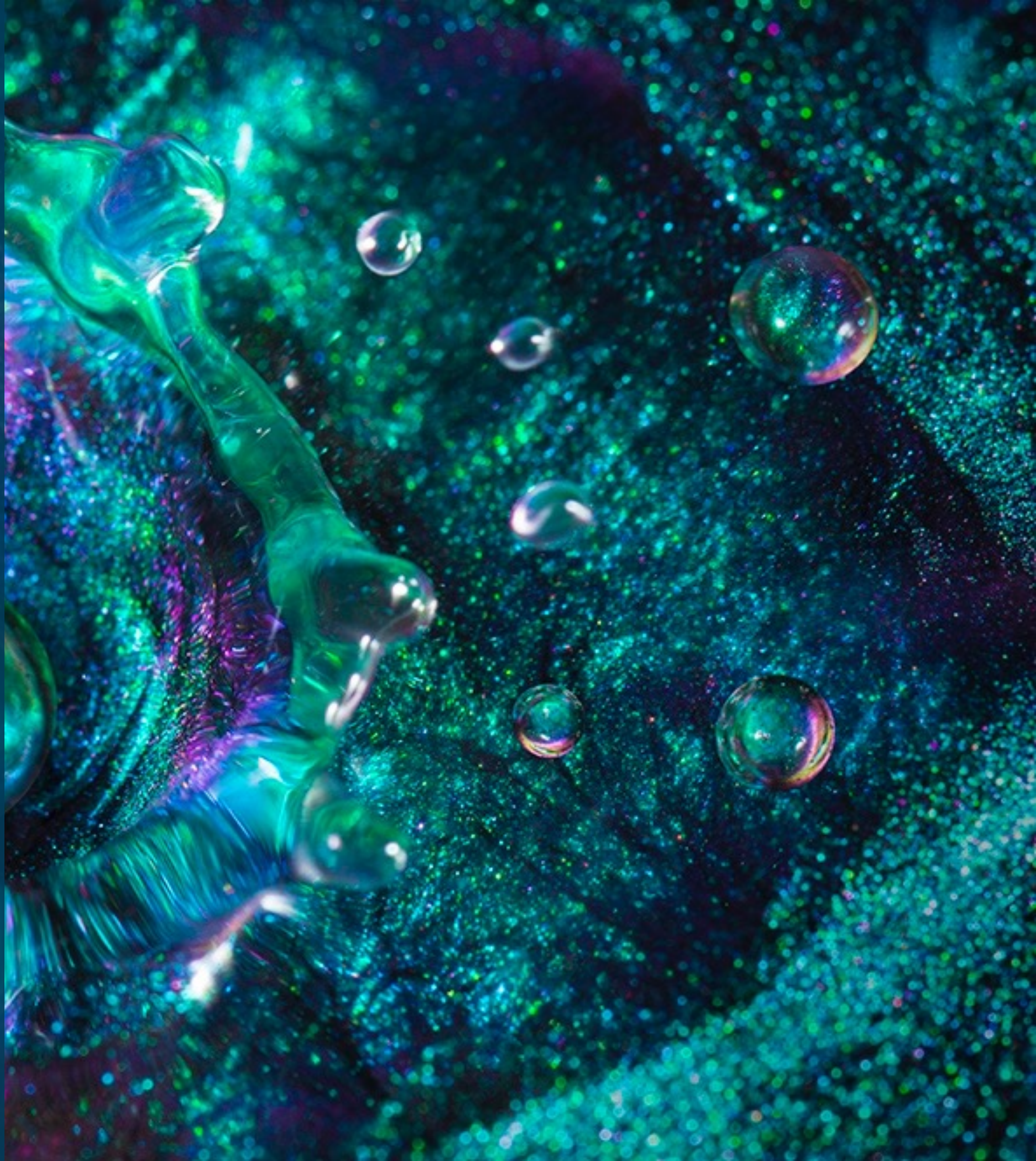
- (Built-in) did:sov - For backwards compatibility.
- (Soon to be built-in) did:key - Introduced with BBS+ work by Animo.
- did:github - Fully functional example resolver plugin.
- did:web - Recent did:web resolver implementation from Bosch Research.
- Universal Resolver - Resolve through Universal Resolver over HTTP.
- DIDComm Resolver - DID Resolution via remote resolver over DIDComm.

Resources

- https://hackmd.io/@dbluhm/uniresolver-acapy
- https://github.com/hyperledger/aries-rfcs/blob/master/features/0124-did-resolution-protocol/README.md
- https://github.com/sicpa-dlab/aries-acapy-plugin-didcomm-resolver
- https://github.com/sicpa-dlab/aries-acapy-plugin-http-uniresolver

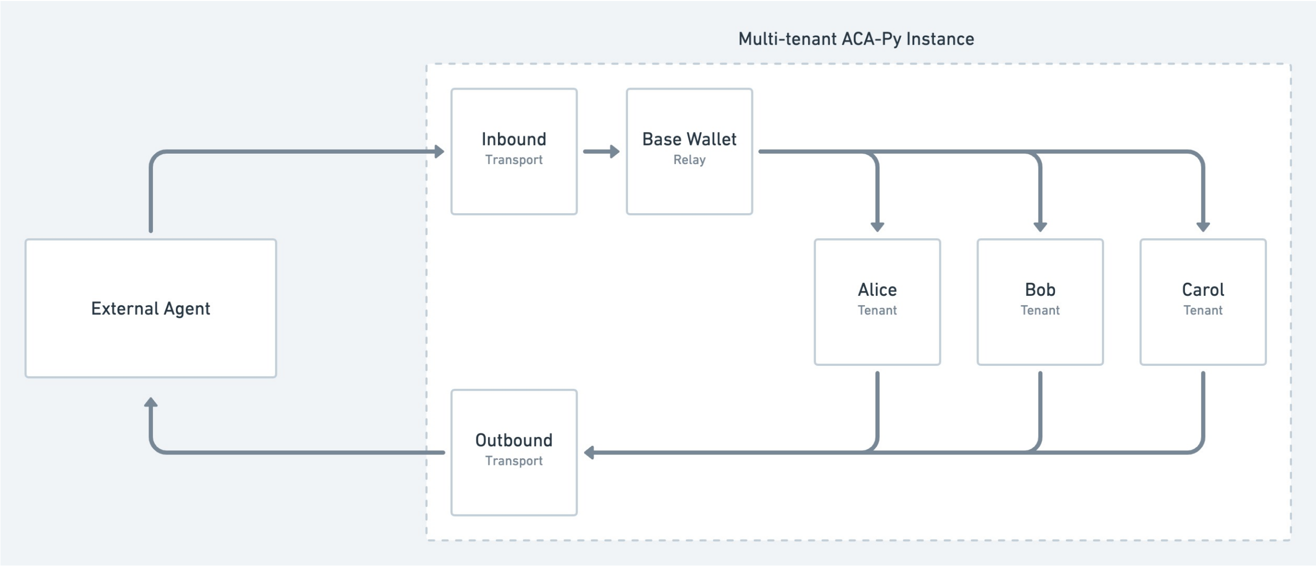
SICPA

Multi-tenant agency



ACA-py Multitenancy

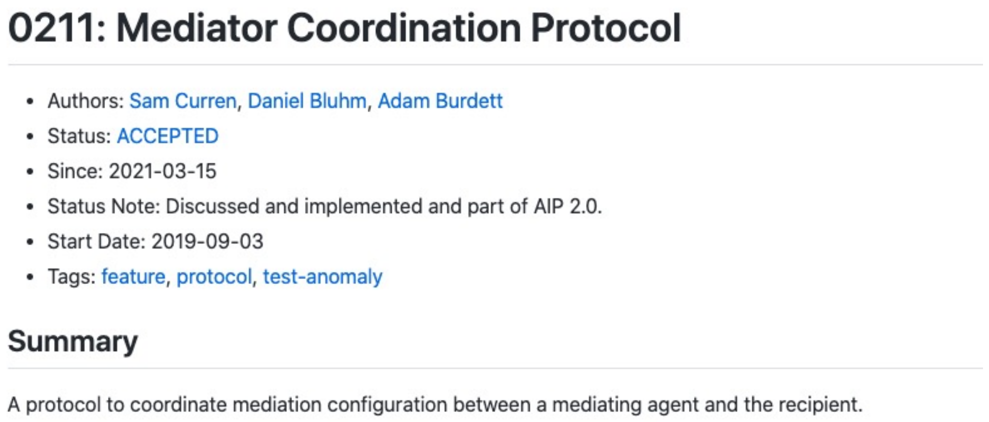
- Multi-tenancy in ACA-Py allows multiple tenants to use the same ACA-Py instance with a different context. All tenants get their own encrypted wallet that only holds their own data.



Implementation of Mediator

- A service that hosts many cloud agents at a single endpoint to provide herd privacy (an "agency") is a *mediator*.

- Aries RFC 0211 - https://github.com/hyperledger/aries-rfcs/tree/main/features/0211-route-coordination
  - A protocol to coordinate mediation configuration between a mediating agent (base wallet) and the recipient.



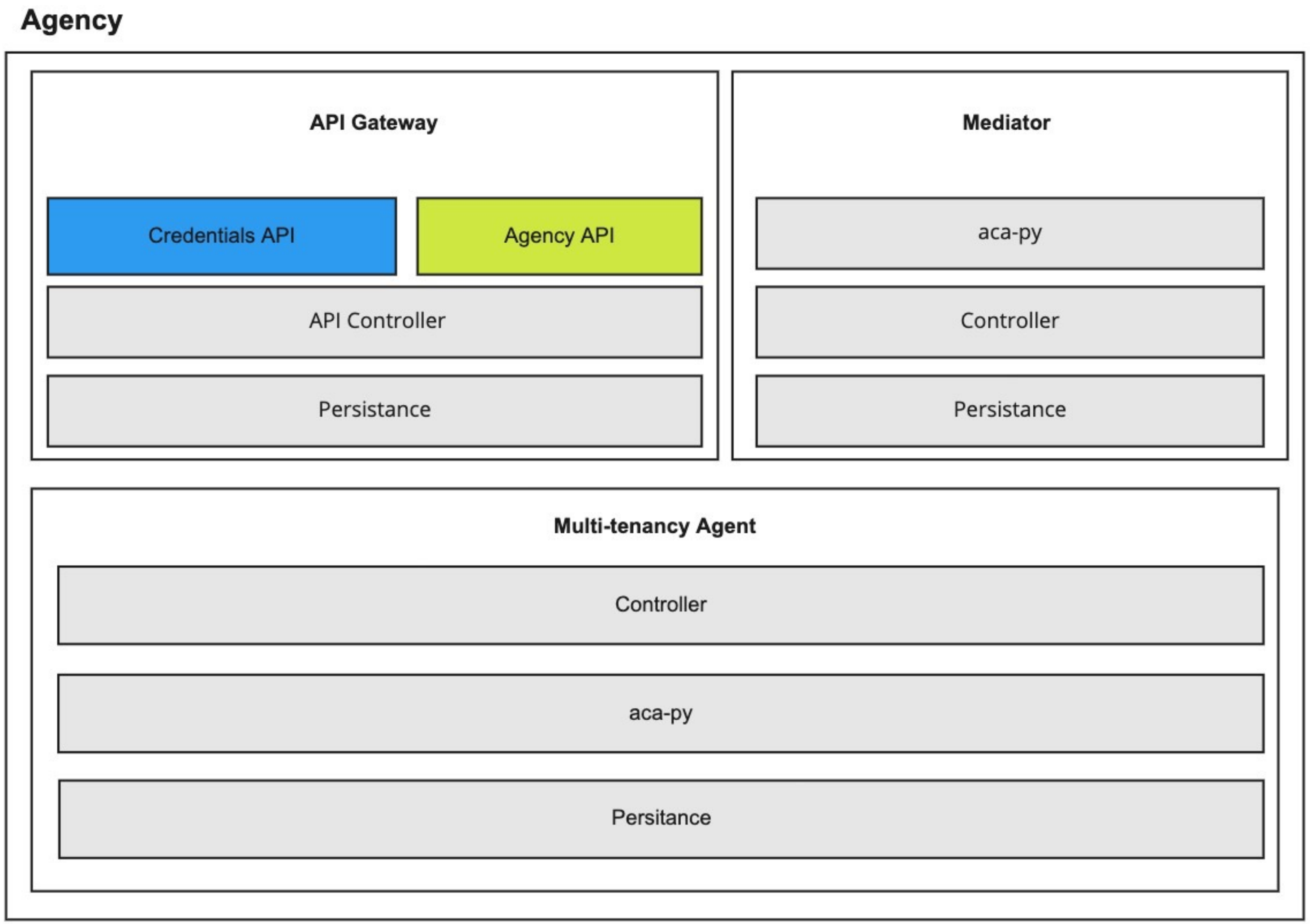
0211: Mediator Coordination Protocol

- Authors: Sam Curren, Daniel Bluhm, Adam Burdett
- Status: ACCEPTED
- Since: 2021-03-15
- Status Note: Discussed and implemented and part of AIP 2.0.
- Start Date: 2019-09-03
- Tags: feature, protocol, test-anomaly

Summary

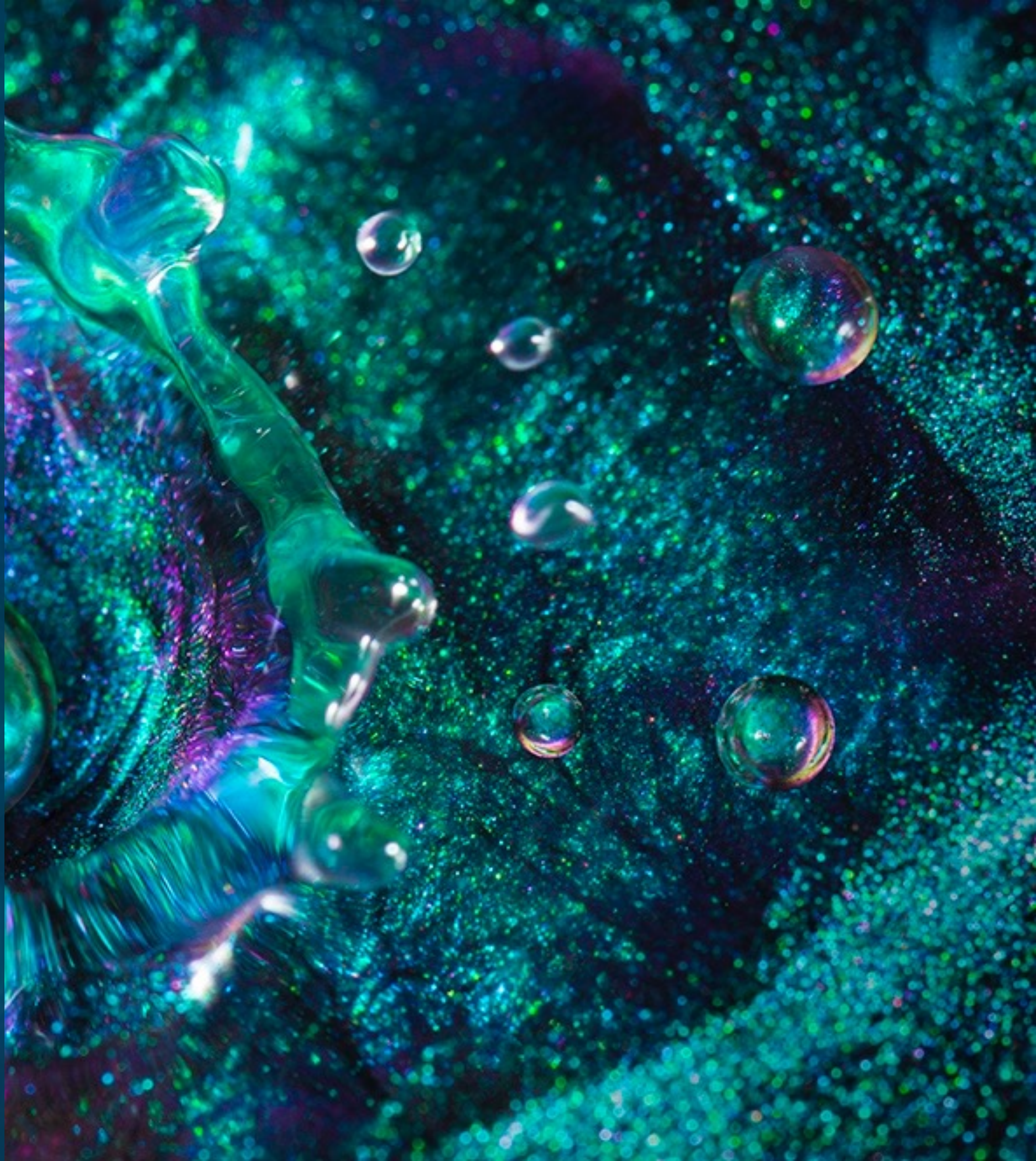
A protocol to coordinate mediation configuration between a mediating agent and the recipient.

Multi-tenant Agency



SICPA

Managing events with Kafka

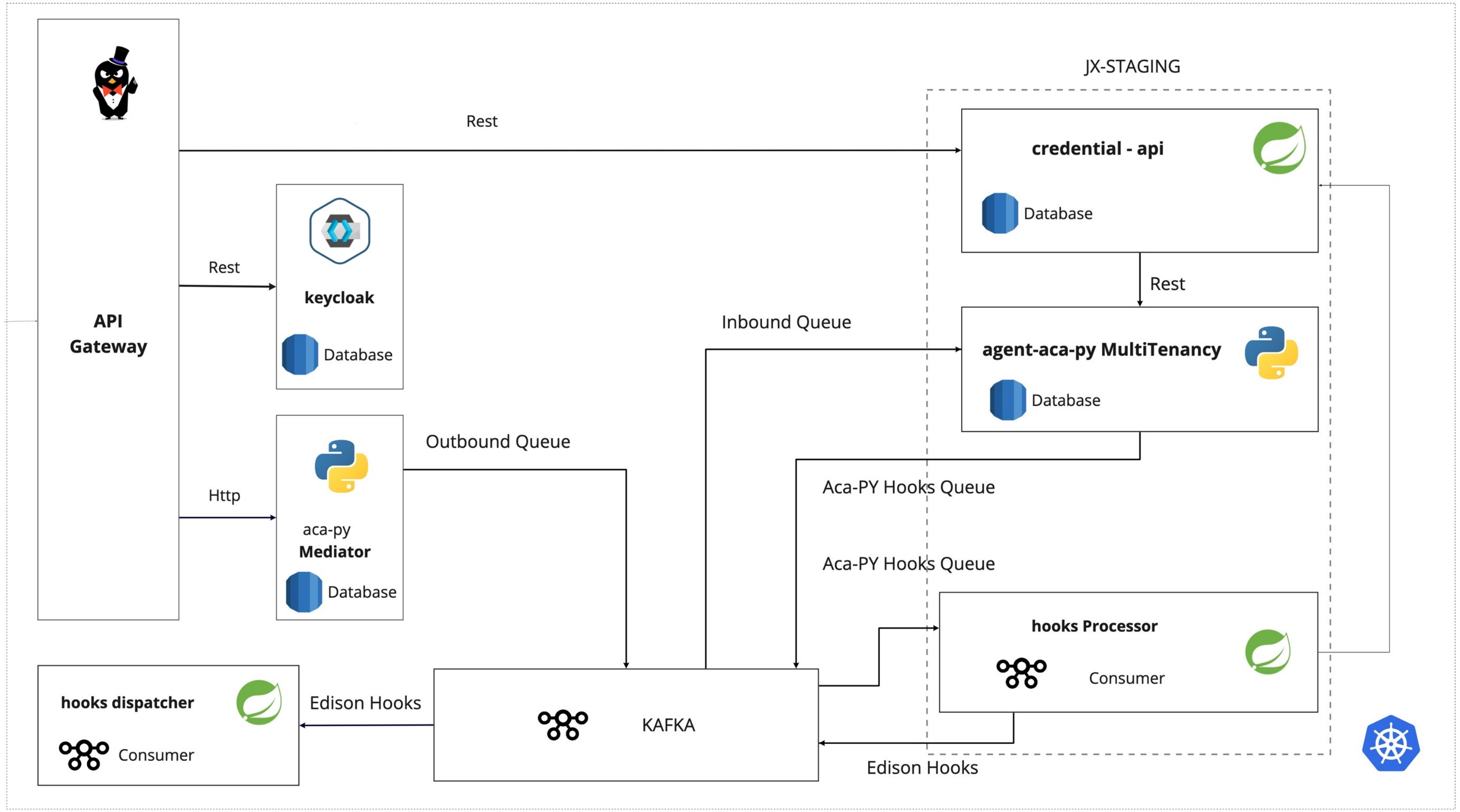


ACA-Py Kafka Events

- In order to scale processing of ACA-Py events without the use of a “middleman” webhook listener, we want to push ACA-Py events directly to a Kafka Queue.

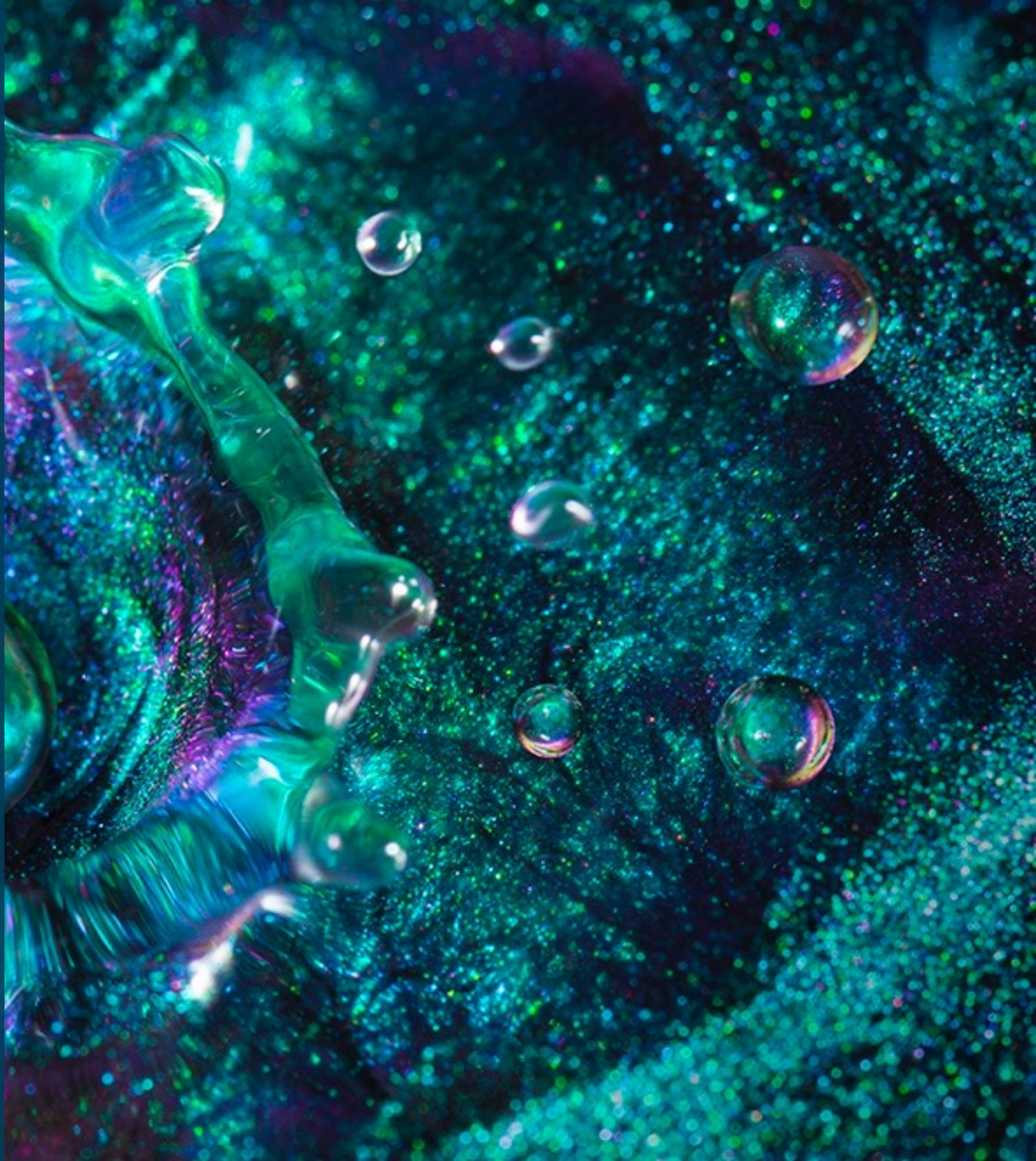
- **Why Kafka?**
  - Message system (**Transport**):
    - High performance
    - Native data partition
    - Replication
    - Fault tolerant
  - Activity tracer (**Analytics, Monitoring & Security**)
    - Rebuild an activity tracking pipeline
    - Operational surveillance

ACA-Py with Kafka

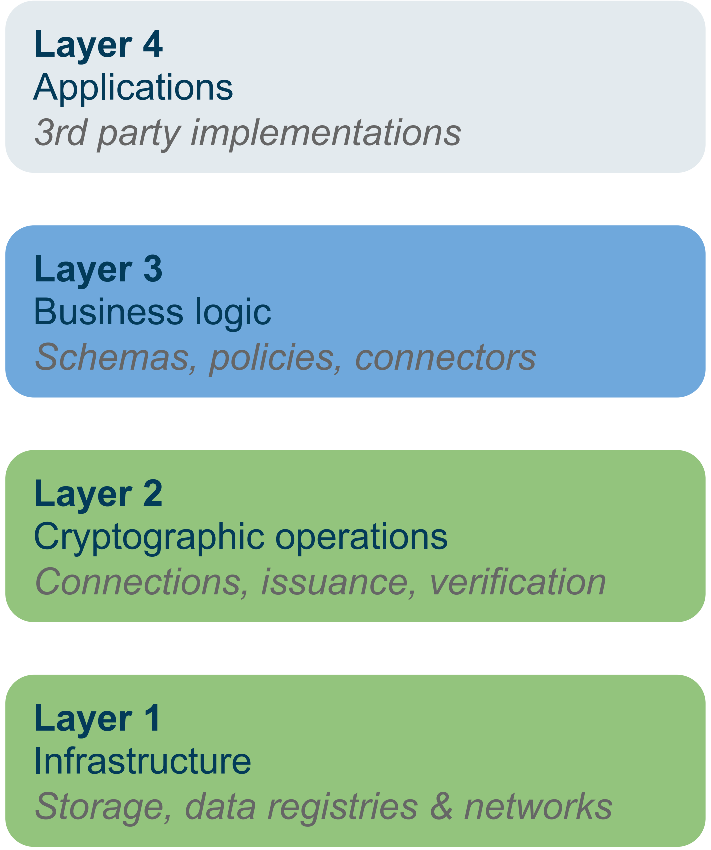


SICPA

Edison



Edison – A key building block to digitally *enable trust*



Layer 4
Applications
3rd party implementations

Layer 3
Business logic
Schemas, policies, connectors

Layer 2
Cryptographic operations
Connections, issuance, verification

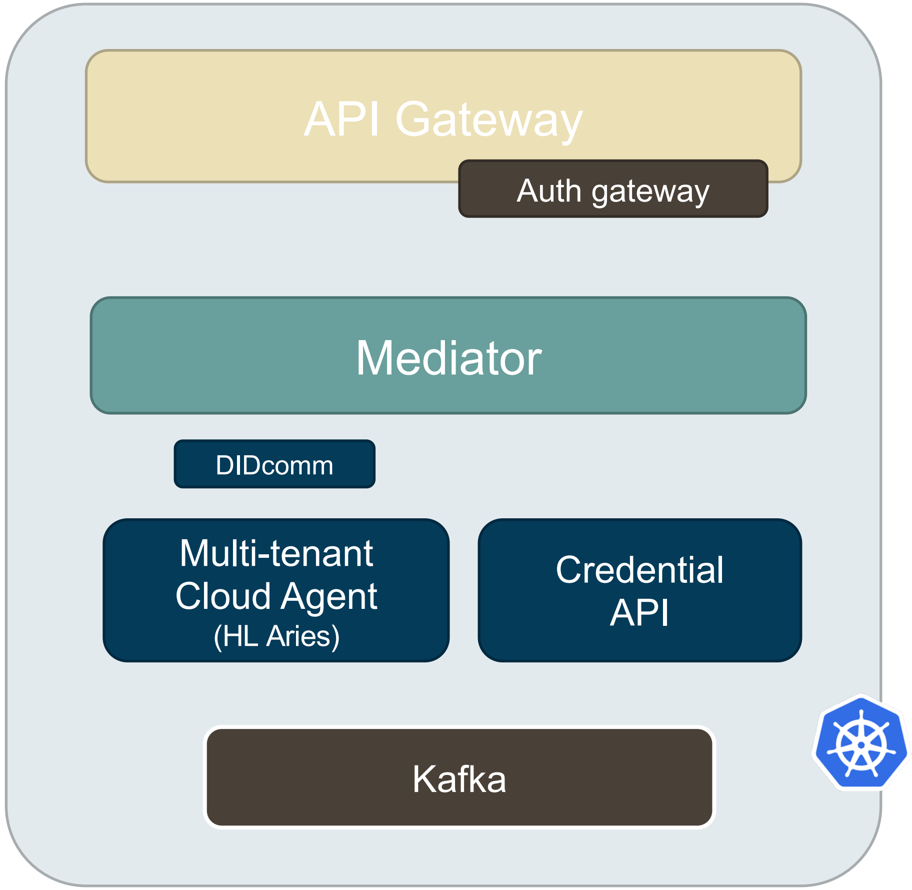
Layer 1
Infrastructure
Storage, data registries & networks

Edison is an engine for the digital issuance, management and verification of verifiable credentials.

It is built on global open standards that ensure interoperability, combined with specific business logic based on SICPA’s experience in the field of authentication.

Edison features

- Issue, verify, revoke, and manage verifiable credentials
- Multi-tenant agency (ISP for identity)
- Provide features via APIs
- Build it for scale, with enterprise grade architecture, to be deployed on-premise
- Built it on ledger-agnostic, open standards and opensource technology (Aries)
- Aries Interop Profile (formal tests passed by several vendors)
- DIDcomm can enable building bridges with other verticals



SICPA

Thank you
for your attention