Decentralized Identifier WG TPAC Sessions

Day 1: September 16, 2019
Chairs: Dan Burnett, Brent Zundel
Location: Hilton Fukuoka Sea Hawk, 1st Floor, Navis B

Welcome!

- Logistics
- IRC and Scribes
- W3C WG IPR Policy
- Introductions & Dinner
- Agenda

Logistics

- Location: Hilton Fukuoka Sea Hawk
- WiFi: SSID is “W3C-TPAC-2019”, password is “fukuoka2019”
- Dial-in information: +1-617-324-0000, Meeting ID 640 843 420
- Restrooms: Out the room doors, head left
- Meeting time: 8:30 am - 6 pm, Sep. 16-17
- Breaks: 10:30-11 am, 1-2 pm, 3:30-4 pm
- TPAC Agenda: https://www.w3.org/2019/09/TPAC/schedule.html
- DID WG Agenda: https://tinyurl.com/didwg-tpac2019-agenda (HTML)
- Live slides: https://tinyurl.com/didwg-tpac2019-slides (Google Slides)
- Dinner Details: See the “Dinner Tonight” slide at the end of the deck

W3C WG IPR Policy

- This group abides by the W3C patent policy https://www.w3.org/Consortium/Patent-Policy-20040205
- Only people and companies listed at https://www.w3.org/2004/01/pp-impl/117488/status are allowed to make substantive contributions to the specs
- Code of Conduct https://www.w3.org/Consortium/cepc/

Today’s agenda

Time	Topic	Discussion Leader
8:30	Welcome, Introductions, and Logistics	Chairs
9:00	Getting to Candidate Recommendation	Chairs
10:00	A short history of DIDs	Drummond Reed
10:30	Break	
11:00	Perspectives on DIDs ("DLT-based DIDs" sov/v1/btcr/eth/uport?)	Markus Sabadello
11:30	Perspectives on DIDs ("Unanchored DIDs" peer/key?)	Ken Ebert
12:00	Perspectives on DIDs ("Layer 2 DIDs" ion/stack?)	Daniel Buchner
12:30	Perspectives on DIDs ("Alternative DIDs" git/web/ipfs/private network)	Manu Sporny
13:00	Lunch	
14:00	Documenting Use Cases	Joe Andrieu
15:00		
15:30	Break	
16:00	Technical direction	Chairs
17:00	CCG GitHub Issues	Chairs

IRC and Scribes

- Meeting discussions will be documented
    - Text Chat: http://irc.w3.org/?channels=did
    - IRC://irc.w3.org:6665/#did
- Telecon info
    - https://mit.webex.com/mit/j.php?MTID=m74edf1b6bcf79ff59facbc72e9f0c49d
    - meeting 640 843 420
    - Phone number tel:%2B1-617-324-0000,,*01*640843420%23%23*01*

	Monday	**Tuesday**
AM1	Charles	Ken Ebert
AM2	Joe Andrieu	Markus Sabadello
PM1	David Ezell	Gregg Kellogg
PM2	Manu Sporny	Mike Jones

Introductions & Dinner

- Introductions

- Expected count for dinner:
- Dinner proposals:
  - Suggestions here
  -

Potential topics for the “Open Topics” sessions

- Volunteers for making our home page visitor-friendly
- Controller?
- David Huseby’s DID:GIT comments
- Editors -
  - who are they?
  - work mode for editors (may they do a pass and clean up the spec w/o needing PRs?)
    - what can we rip out before adding stuff
- external communication
  - some outreach to the outside world should be made regularly, e.g., twitter, blogs, etc.
  - monthly blog would be very good

Getting to Candidate Recommendation

Getting to Candidate Recommendation (60 min)

- Charter Summary
  - DID Mission and Goals (Dan 5 min)
  - DID WG Scope (Brent 10 min)
  - DID WG Out of Scope (Brent 5 min)
- Process Review (Dan 15 min)
- Deliverable Review and Status (Brent 10 min)
- Timing (Dan 15 min)

DID WG Mission and Goals (Dan)

- “… standardize the DID URI scheme, the data model and syntax of DID Documents, which contain information related to DIDs that enable the aforementioned initial use cases, and the requirements for DID Method specifications.”

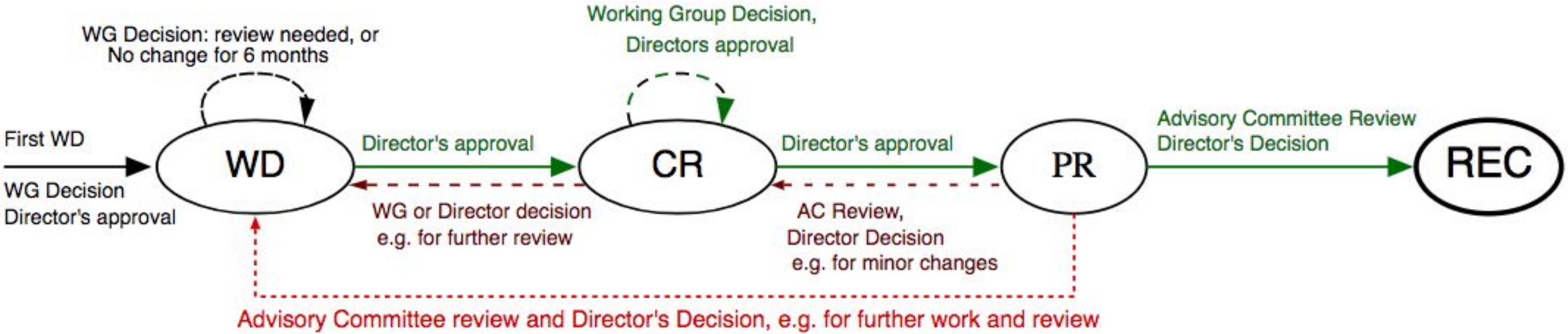
DID WG Scope (Brent)

- Define the DID URI scheme.
- Recommend a data model and syntax(es) for the expression of Decentralized Identifier Documents, including one or more core vocabularies.
- Recommend a set of requirements for DID Method specifications that are conformant with the data model and syntax(es).
- Provide a rubric of decentralized characteristics for DID Method specifications.
- Concentrate their efforts on the initial use cases with a particular focus on enabling future specification and implementation of Identity and Access Management.
- Define extension points enabling authentication, signing and cryptography mechanisms.
- With the initial use cases document as input, the WG will produce a NOTE at the end of the process that is a refined Use Cases document.
- Establish a deterministic mapping between DID method identifiers and the resolution process used to resolve that DID method.

DID WG Out of Scope (Brent)

- Authentication or Authorization Protocols
- Browser APIs
- Specific DID Method specifications or Protocol specifications
- "Solving Identity" on the Web
- Defining specific authentication, signing, or cryptography mechanisms. Scope is limited to defining extension points for these mechanisms.

W3C Technical Report Progression Process (Dan)



https://www.w3.org/2019/Process-20190301/

W3C Technical Report Process (Dan)

- WD - does not imply consensus
- CR
  - Entry - to publish as CR, the document is expected to be feature complete, have had wide review, and must specify the implementation requirements needed to exit
  - Exit - to exit CR (and move to PR), the document must satisfy the stated implementation requirements; it must also not have made any substantive change not warned about upon entry
- PR
  - Basically a one-month sanity check during which the AC is encouraged to have any final review and discussion, but if anything major happens it's a fail (requiring a move back to CR or earlier)
- Recommendation - Done
  - But errata are possible

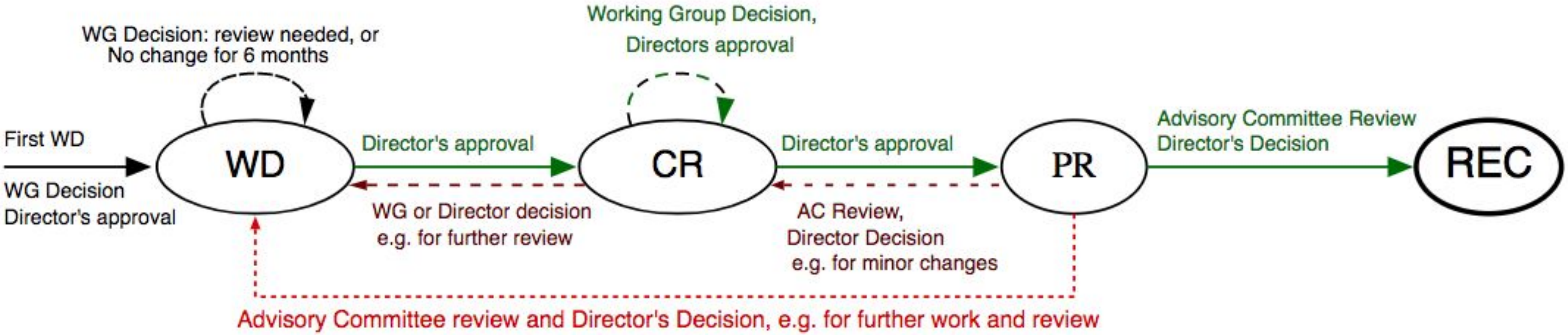
Documents and Background (Brent)

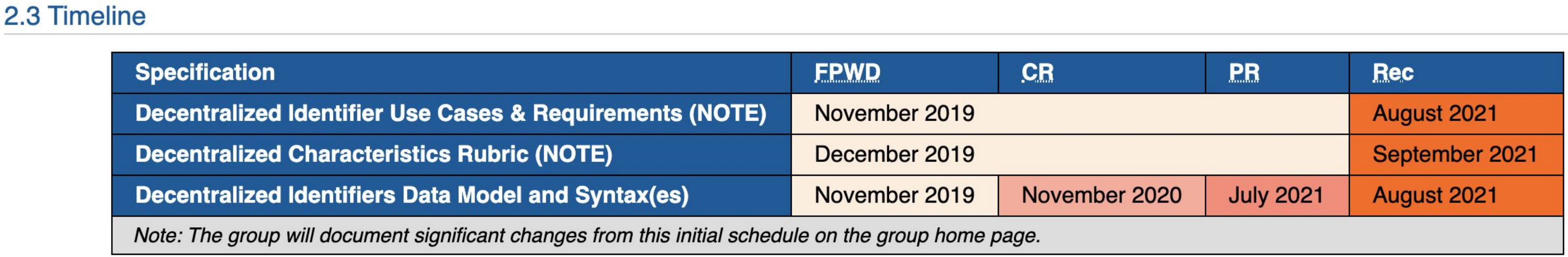
- Home: https://www.w3.org/2019/did-wg/
- Charter: https://www.w3.org/2019/09/did-wg-charter.html
- Primer: https://w3c-ccg.github.io/did-primer/

Charter Deliverables (Brent)

- Recommendation-track Specification
  - Decentralized Identifiers v1.0
- W3C Notes
  - Decentralized Identifier Use Cases v1.0
  - Decentralized Characteristics Rubric v1.0
- Other Deliverables
  - Test Suite and Implementation Report

Timing (Dan)

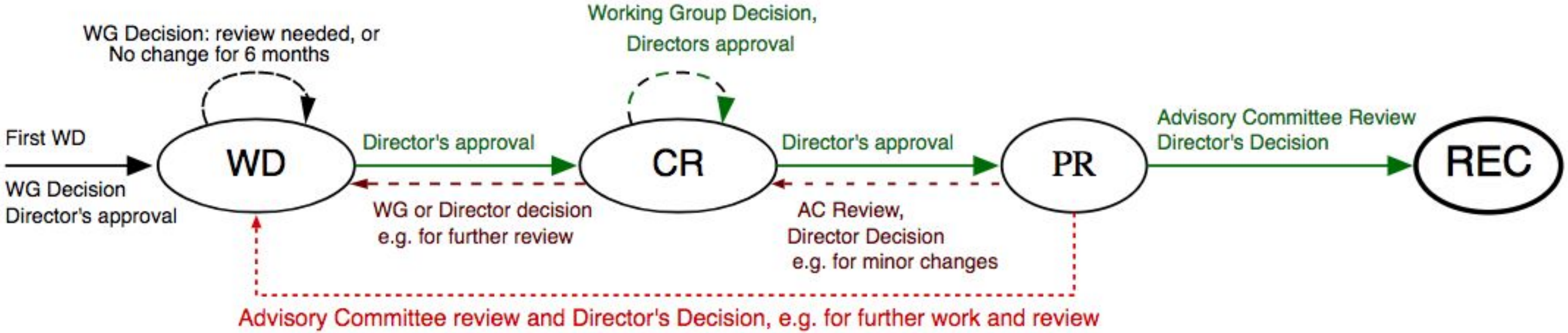




2.3 Timeline

Specification	**FPWD**	**CR**	**PR**	**Rec**
Decentralized Identifier Use Cases & Requirements (NOTE)	November 2019			August 2021
Decentralized Characteristics Rubric (NOTE)	December 2019			September 2021
Decentralized Identifiers Data Model and Syntax(es)	November 2019	November 2020	July 2021	August 2021
Note: The group will document significant changes from this initial schedule on the group home page.				

Timing of our primary spec (Dan)



Nov 2019
(FPWD)

May 2020
(Feature freeze)

Nov 2020
(CR1)

March 2021
(CR2)

Jul 2021

Aug 2021

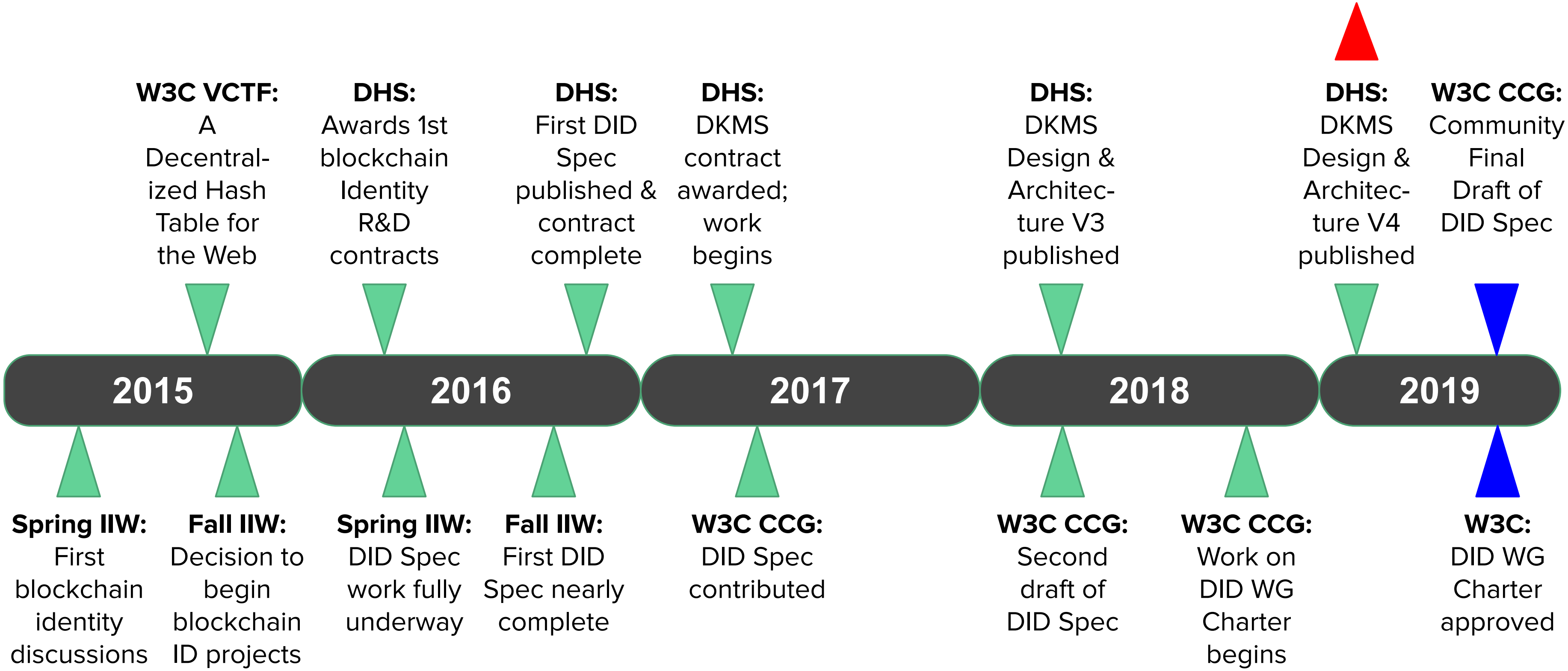
Decentralized Identifiers Data Model and Syntax(es)	November 2019	November 2020	July 2021	August 2021

https://www.w3.org/2019/Process-20190301/

A brief history of DIDs (Drummond, 30 min)

Timeline

http://bit.ly/dkms-v4



W3C VCTF: A Decentralized Hash Table for the Web

DHS: Awards 1st blockchain Identity R&D contracts

DHS: First DID Spec published & contract complete

DHS: DKMS contract awarded; work begins

DHS: DKMS Design & Architecture V3 published

DHS: DKMS Design & Architecture V4 published

W3C CCG: Community Final Draft of DID Spec

Spring IIW: First blockchain identity discussions

Fall IIW: Decision to begin blockchain ID projects

Spring IIW: DID Spec work fully underway

Fall IIW: First DID Spec nearly complete

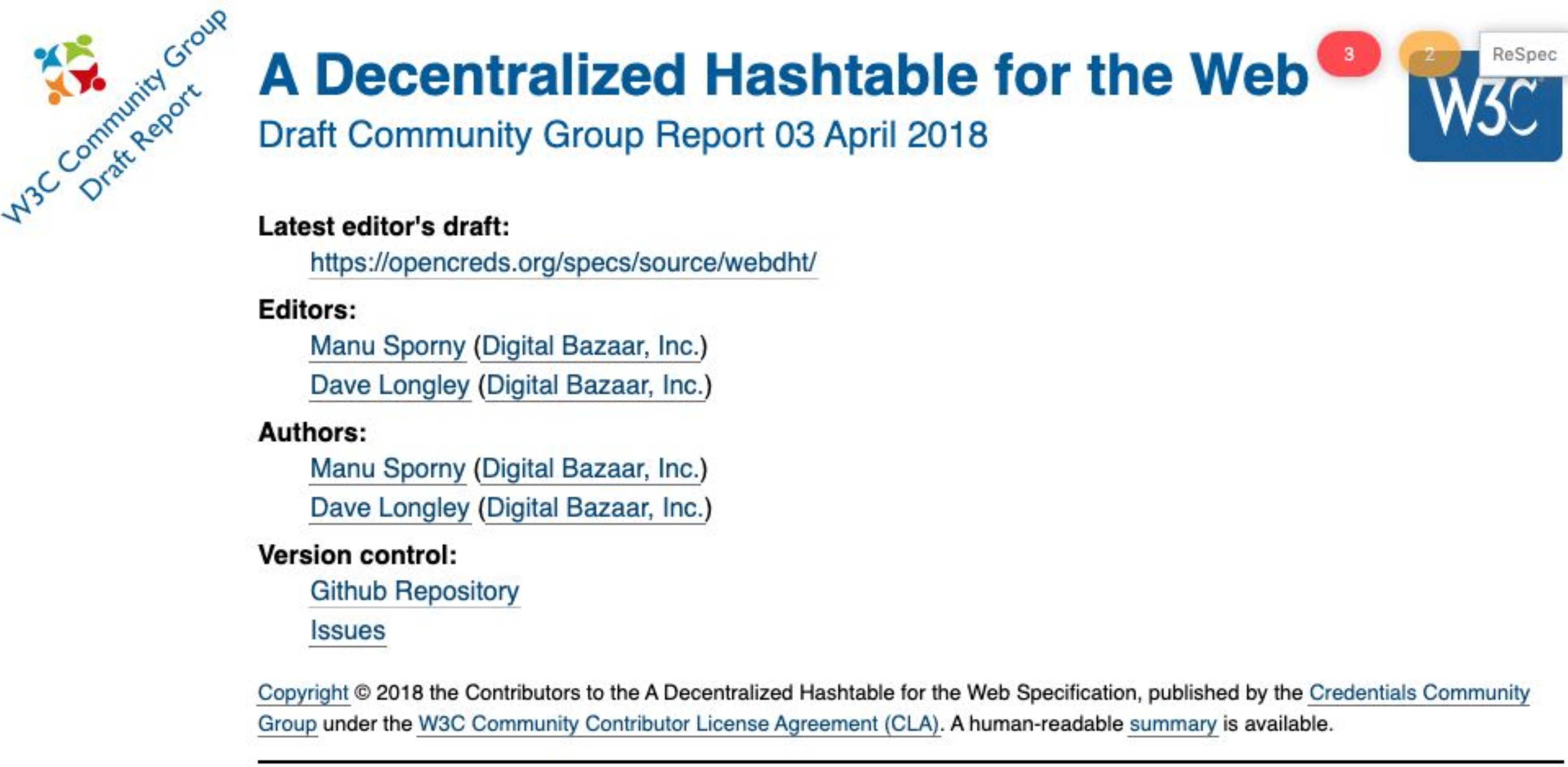
W3C CCG: DID Spec contributed

W3C CCG: Second draft of DID Spec

W3C CCG: Work on DID WG Charter begins

W3C: DID WG Charter approved

Where did the term “DID” come from?



A Decentralized Hashtable for the Web

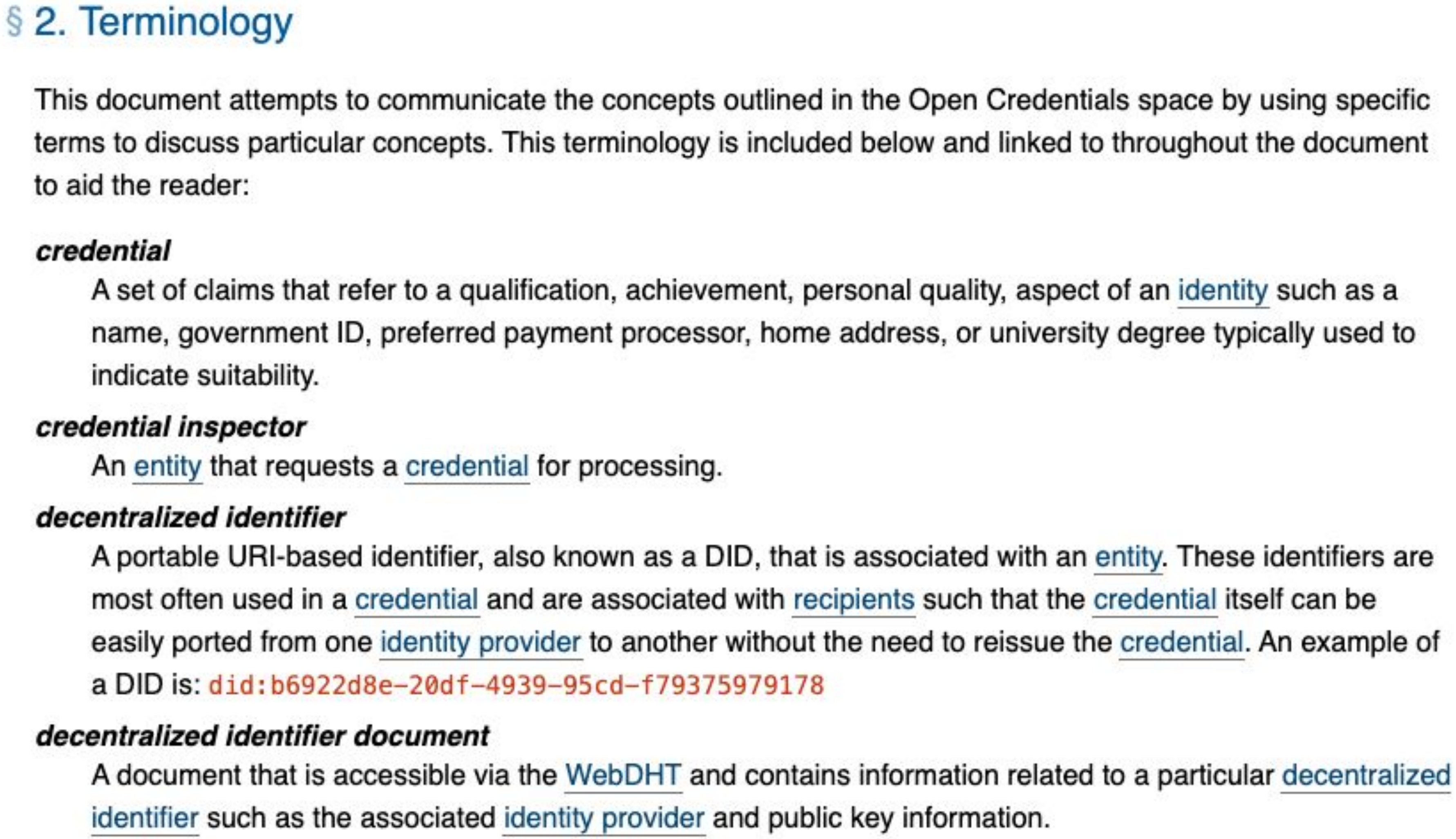
Draft Community Group Report 03 April 2018

Latest editor's draft:
https://opencreds.org/specs/source/webdht/

Editors:
Manu Sporny (Digital Bazaar, Inc.)
Dave Longley (Digital Bazaar, Inc.)

Authors:
Manu Sporny (Digital Bazaar, Inc.)
Dave Longley (Digital Bazaar, Inc.)

Version control:
Github Repository
Issues



§ 2. Terminology

This document attempts to communicate the concepts outlined in the Open Credentials space by using specific terms to discuss particular concepts. This terminology is included below and linked to throughout the document to aid the reader:

credential
: A set of claims that refer to a qualification, achievement, personal quality, aspect of an identity such as a name, government ID, preferred payment processor, home address, or university degree typically used to indicate suitability.

credential inspector
: An entity that requests a credential for processing.

decentralized identifier
: A portable URI-based identifier, also known as a DID, that is associated with an entity. These identifiers are most often used in a credential and are associated with recipients such that the credential itself can be easily ported from one identity provider to another without the need to reissue the credential. An example of a DID is: `did:b6922d8e-20df-4939-95cd-f79375979178`

decentralized identifier document
: A document that is accessible via the WebDHT and contains information related to a particular decentralized identifier such as the associated identity provider and public key information.

Why did the U.S. Department of Homeland Security fund development of the DID spec?

Four reasons:

1. **A permanent (persistent) identifier**

   It never needs to change

2. **A resolvable identifier**

   You can look it up to discover metadata

3. **A cryptographically-verifiable identifier**

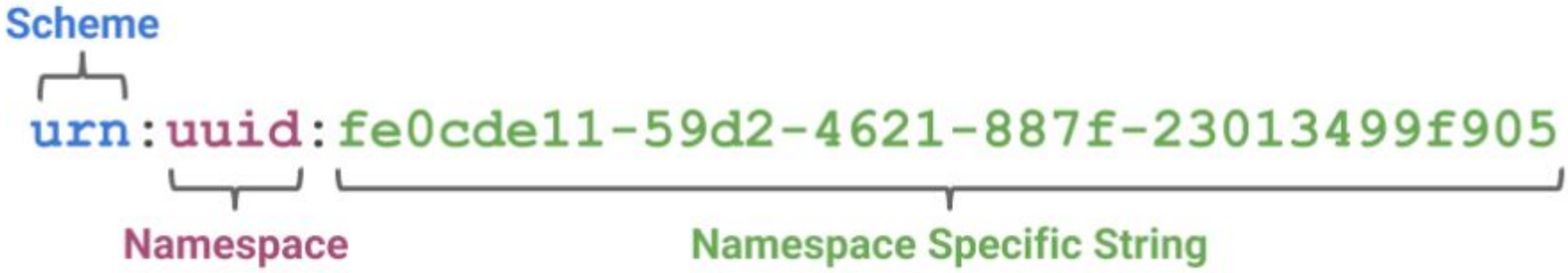
   You can prove control using cryptography

4. **A decentralized identifier**

   No centralized registration authority is required

What does a DID look like?

URNs (Uniform Resource Names, RFC 8141)



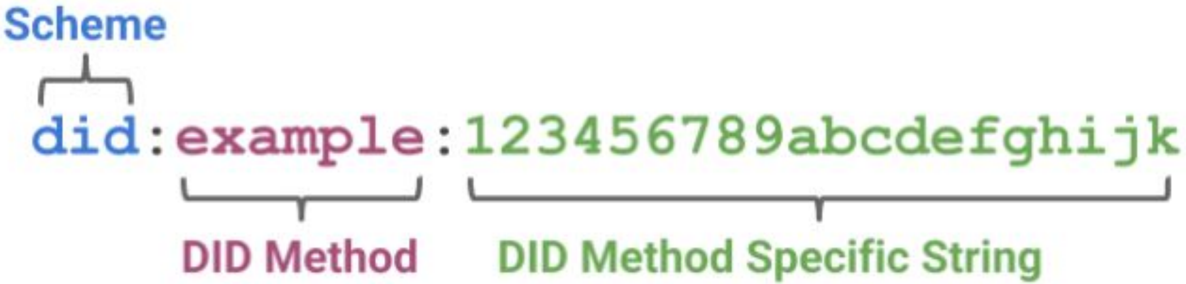
Scheme

urn:uuid:fe0cde11-59d2-4621-887f-23013499f905

Namespace

Namespace Specific String

DIDs



Scheme

did:example:123456789abcdefghijk

DID Method

DID Method Specific String

How widely are DIDs in use today?

Some statistics

- There are currently **32 DID methods** registered in the informal W3C Credentials Community Group DID Method Registry
  - https://w3c-ccg.github.io/did-method-registry/
  - Three for Bitcoin
  - Six for Ethereum
- The Sovrin Foundation currently has **71 stewards** around the world hosting a public permissioned distributed ledger for DIDs
- The Canadian provinces of British Columbia and Ontario have issued **over 1.4 million verifiable business license credentials** based on DIDs

For a full history, see:

https://github.com/WebOfTrustInfo/rwot9-prague/blob/master/topics-and-advance-readings/did-primer-extended.md

Morning break

Perspectives on DIDs - Ledger-Based (Markus, 30 min)

sov btrc eth v1

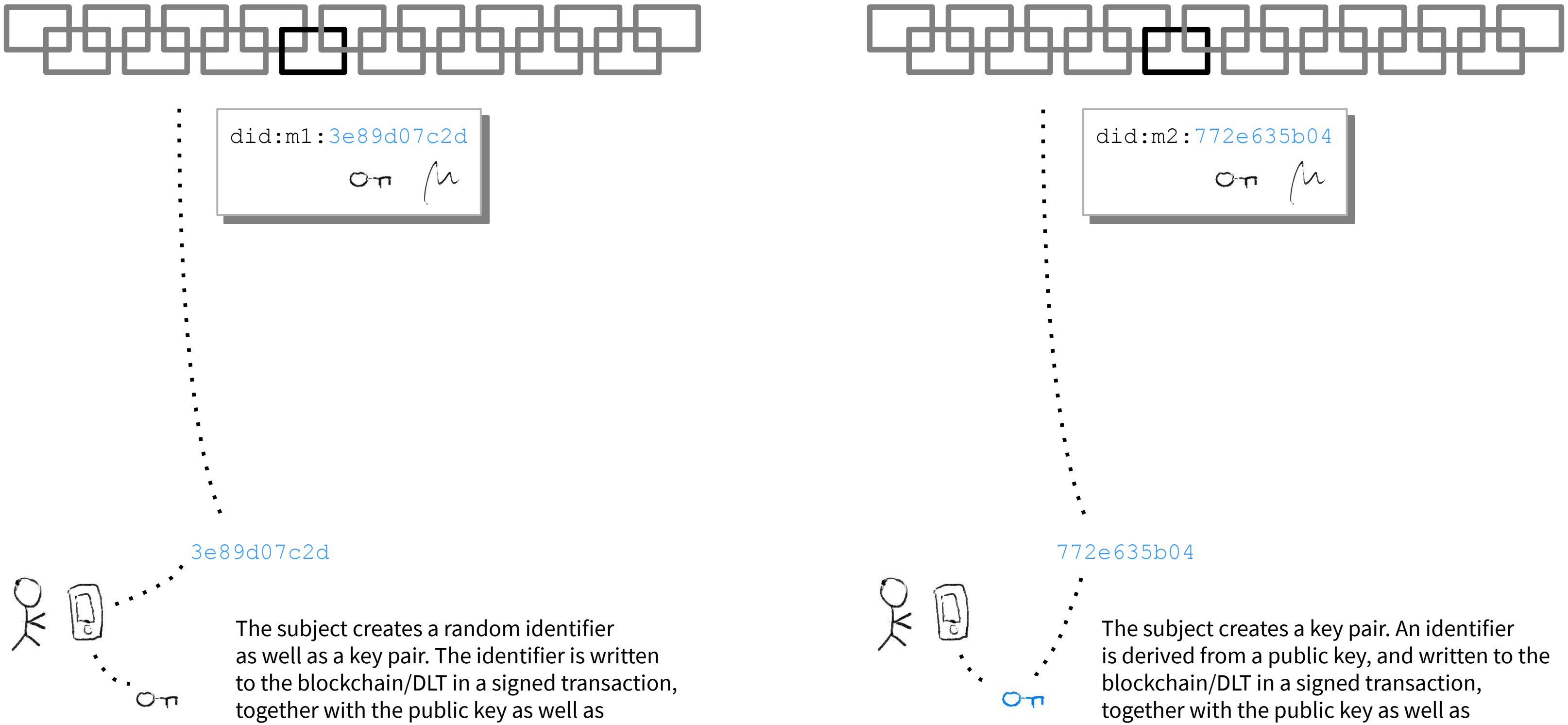
Ledger-Based DIDs

Markus Sabadello
Danube Tech, Decentralized Identity Foundation,
Sovrin Foundation, W3C DID WG + CCG, OASIS XDI TC

https://danubetech.com/
TPAC, Fukuoka, 16th September 2019

Introduction

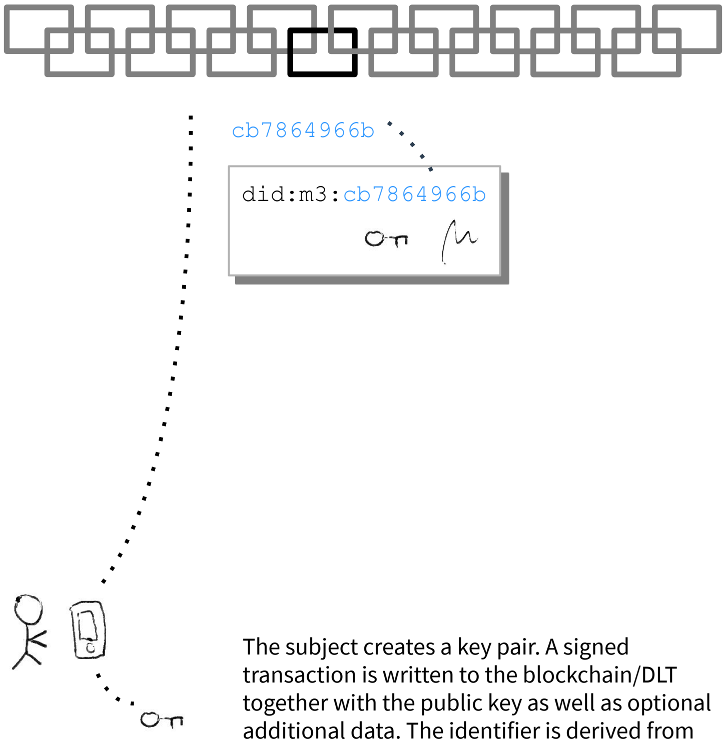
The DID is “created” by writing data to the blockchain/DLT that allows the subject to prove control of the DID. This data is secured by the blockchain/DLT.



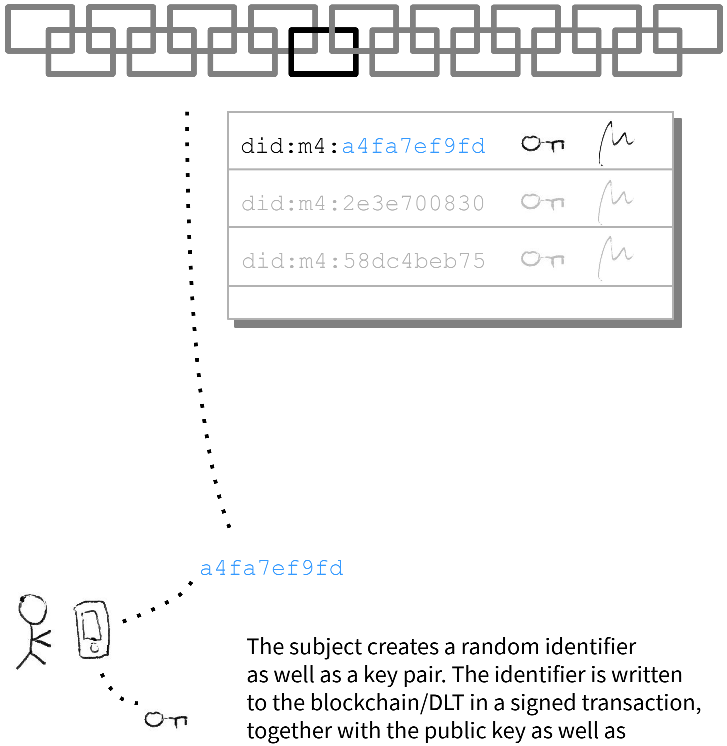
The subject creates a random identifier as well as a key pair. The identifier is written to the blockchain/DLT in a signed transaction, together with the public key as well as optional additional data.

The subject creates a key pair. An identifier is derived from a public key, and written to the blockchain/DLT in a signed transaction, together with the public key as well as optional additional data.

The DID is “registered” by writing data to the blockchain/DLT that allows the subject to prove control of the DID. This data is secured by the blockchain/DLT.



The subject creates a key pair. A signed transaction is written to the blockchain/DLT together with the public key as well as optional additional data. The identifier is derived from the transaction on the blockchain/DLT.



The subject creates a random identifier as well as a key pair. The identifier is written to the blockchain/DLT in a signed transaction, together with the public key as well as optional additional data.

The DID can be “updated” or “deactivated” by writing additional transactions to the blockchain/DLT.

All cumulative transactions pertaining to a DID constitute its latest state.

did:sov:WRfXPg8dantKVubE3HX8pw

DID Registry

Sovrin Ledger

```
NYM: [18,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"BrYDA5NubejDVHkCYBbpY5","reqId":1501522732982387,"signature":"5HGRA...",
"verkey":"~P7F3BNs5VmQ6eVpwkNKJ5D"}]
```

```
ATTRIB: [19,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"WRfXPg8dantKVubE3HX8pw","raw":"0249fedf5246b...","reqId":1504718156368788,
"signature":"3jL1ZNjLAzyAm5"}]
```

…

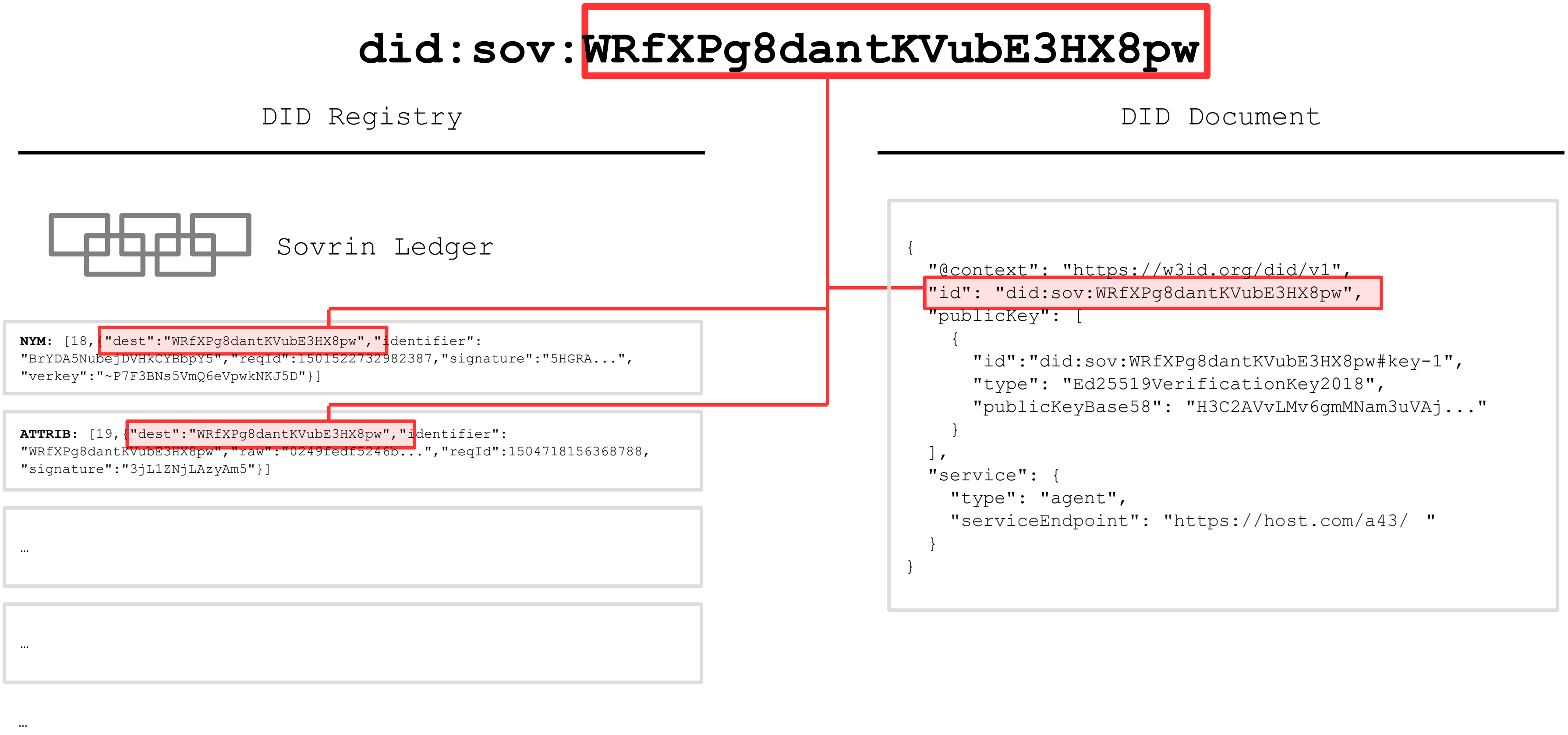
…

…

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "publicKey": [
    {
      "id":"did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

did:sov:WRfXPg8dantKVubE3HX8pw



DID Registry

Sovrin Ledger

```
NYM: [18,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"BrYDA5NubejDVHkCYBbpY5","reqId":1501522732982387,"signature":"5HGRA...",
"verkey":"~P7F3BNs5VmQ6eVpwkNKJ5D"}]
```

```
ATTRIB: [19,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"WRfXPg8dantKVubE3HX8pw","raw":"0249fedf5246b...","reqId":1504718156368788,
"signature":"3jL1ZNjLAzyAm5"}]
```

…

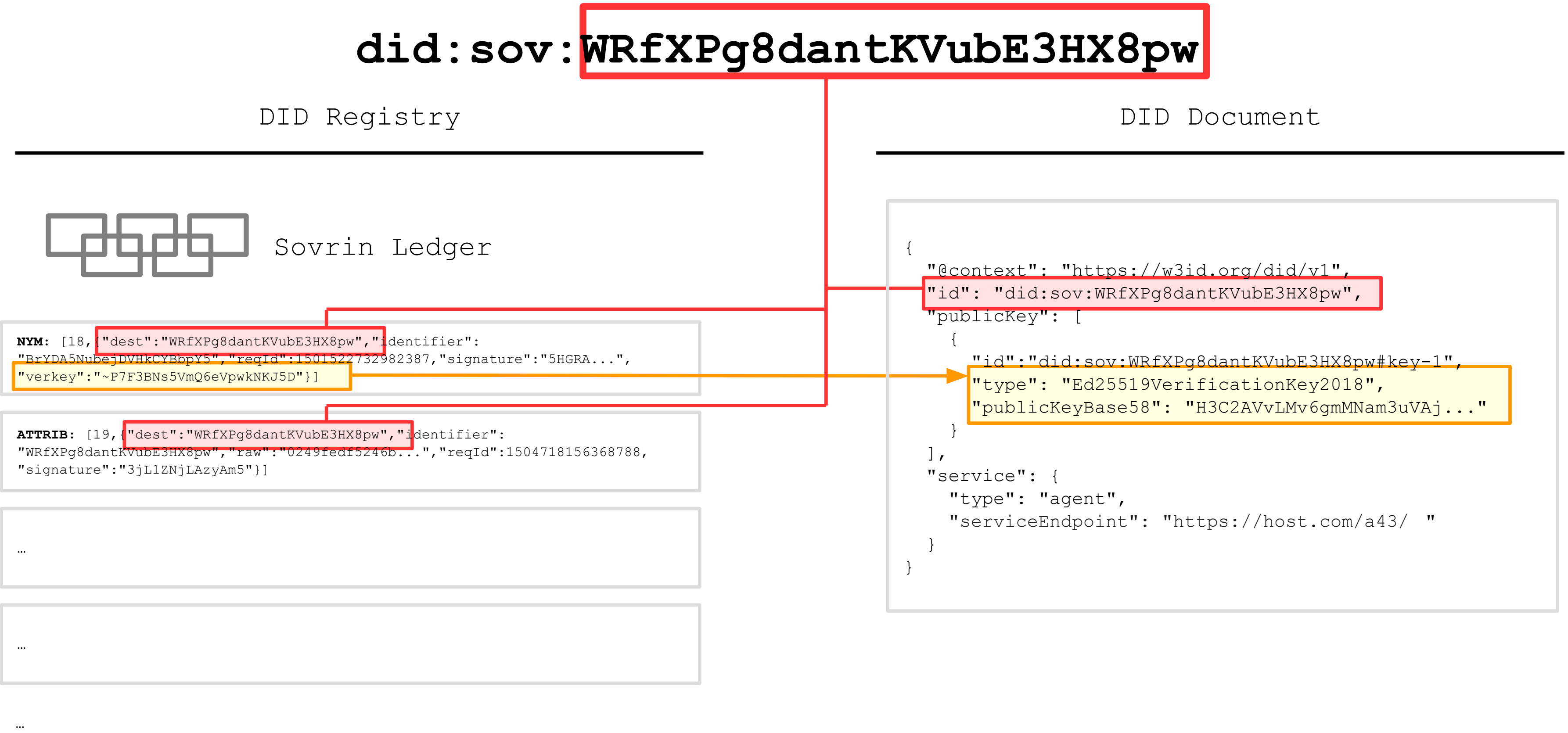
…

…

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "publicKey": [
    {
      "id":"did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE
TECH GMBH



did:sov:WRfXPg8dantKVubE3HX8pw

DID Registry

Sovrin Ledger

```
NYM: [18,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"BrYDA5NubejDVHkCYBbpY5","reqId":1501522732982387,"signature":"5HGRA...",
"verkey":"~P7F3BNs5VmQ6eVpwkNKJ5D"}]
```

```
ATTRIB: [19,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"WRfXPg8dantKVubE3HX8pw","raw":"0249fedf5246b...","reqId":1504718156368788,
"signature":"3jL1ZNjLAzyAm5"}]
```

…

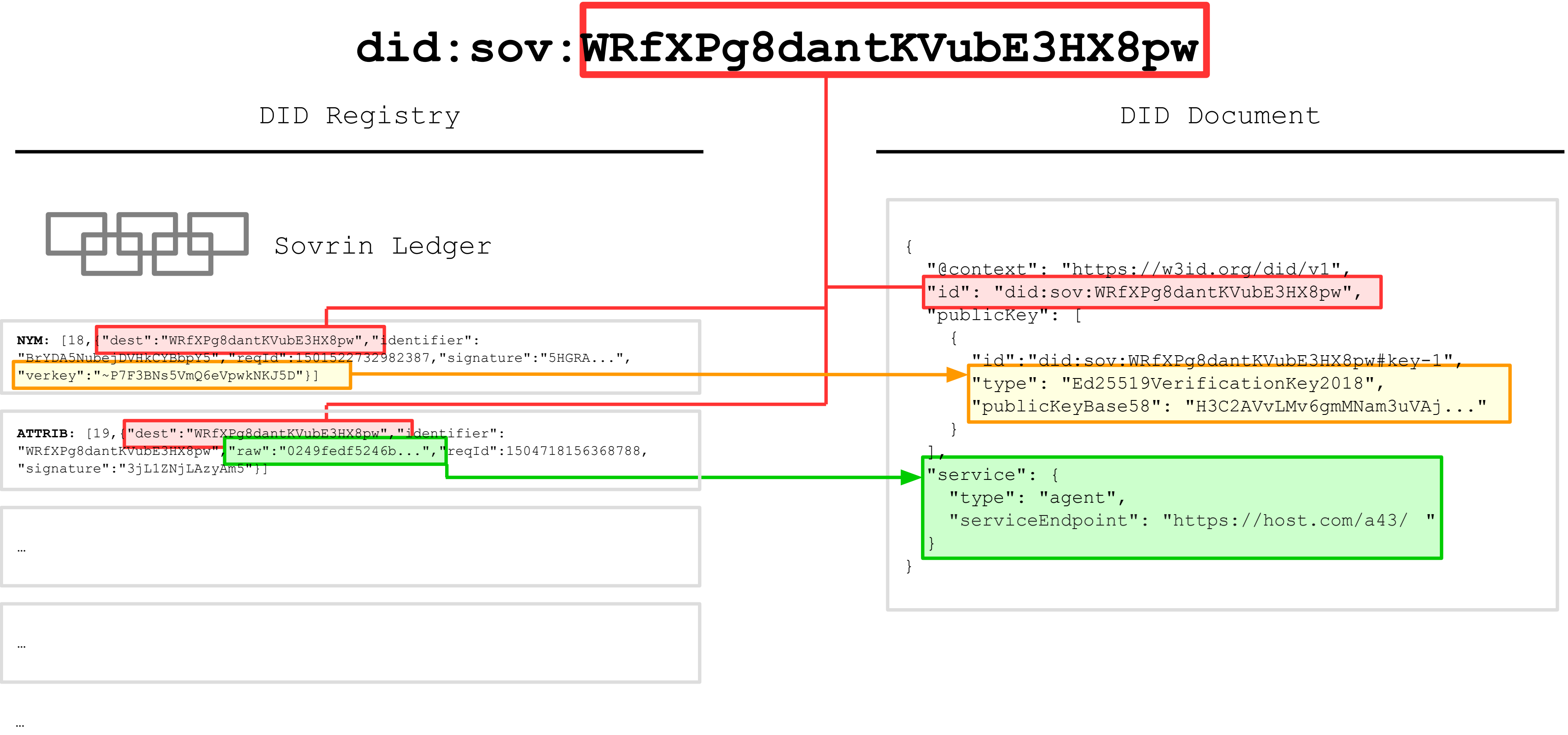
…

…

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "publicKey": [
    {
      "id":"did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

did:sov:WRfXPg8dantKVubE3HX8pw



DID Registry

Sovrin Ledger

```
NYM: [18,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"BrYDA5NubejDVHkCYBbpY5","reqId":1501522732982387,"signature":"5HGRA...",
"verkey":"~P7F3BNs5VmQ6eVpwkNKJ5D"}]
```

```
ATTRIB: [19,{"dest":"WRfXPg8dantKVubE3HX8pw","identifier":
"WRfXPg8dantKVubE3HX8pw","raw":"0249fedf5246b...","reqId":1504718156368788,
"signature":"3jL1ZNjLAzyAm5"}]
```

…

…

…

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "publicKey": [
    {
      "id":"did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH

did:btcr:xz35-jzv2-qqs2-9wjt

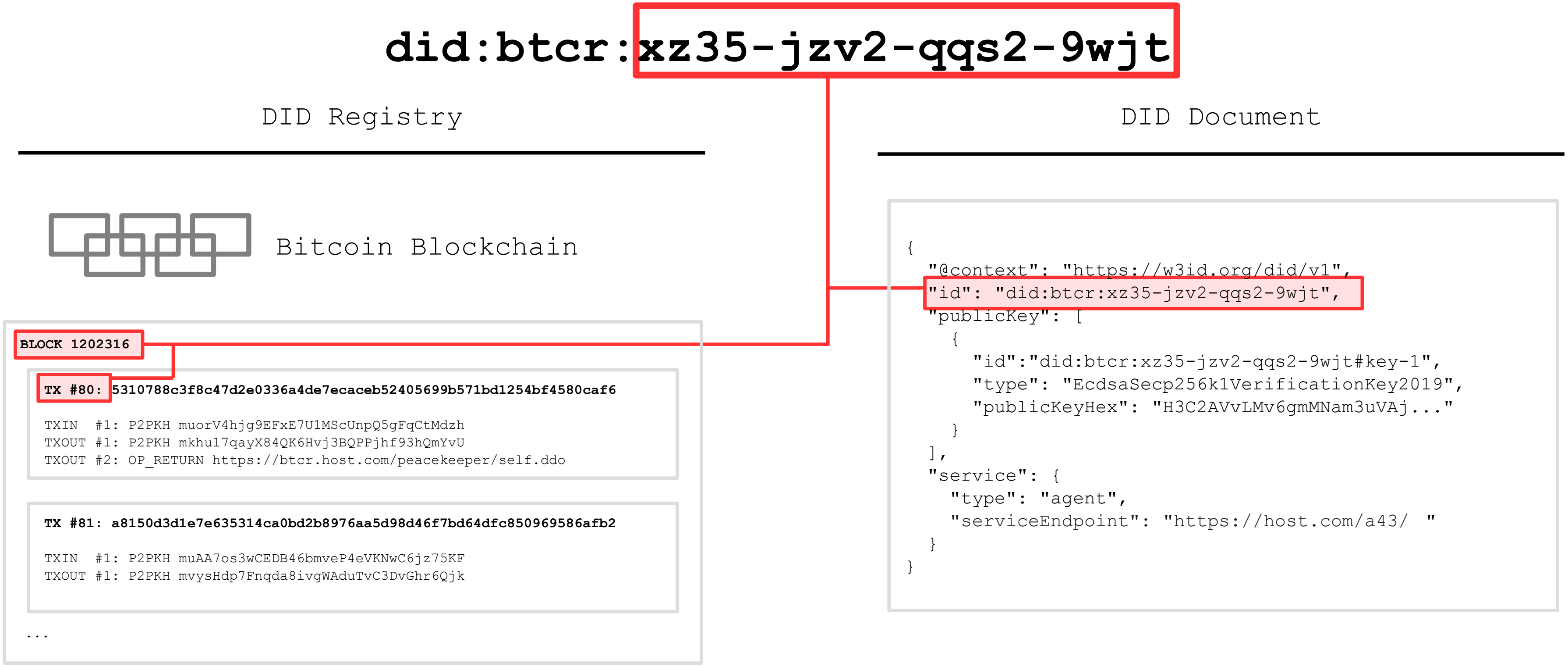
DID Registry

Bitcoin Blockchain

```
BLOCK 1202316

TX #80: 5310788c3f8c47d2e0336a4de7ecaceb52405699b571bd1254bf4580caf6

TXIN  #1: P2PKH muorV4hjg9EFxE7U1MScUnpQ5gFqCtMdzh
TXOUT #1: P2PKH mkhu17qayX84QK6Hvj3BQPPjhf93hQmYvU
TXOUT #2: OP_RETURN https://btcr.host.com/peacekeeper/self.ddo

TX #81: a8150d3d1e7e635314ca0bd2b8976aa5d98d46f7bd64dfc850969586afb2

TXIN  #1: P2PKH muAA7os3wCEDB46bmveP4eVKNwC6jz75KF
TXOUT #1: P2PKH mvysHdp7Fnqda8ivgWAduTvC3DvGhr6Qjk

...
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
  "publicKey": [
    {
      "id":"did:btcr:xz35-jzv2-qqs2-9wjt#key-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "publicKeyHex": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

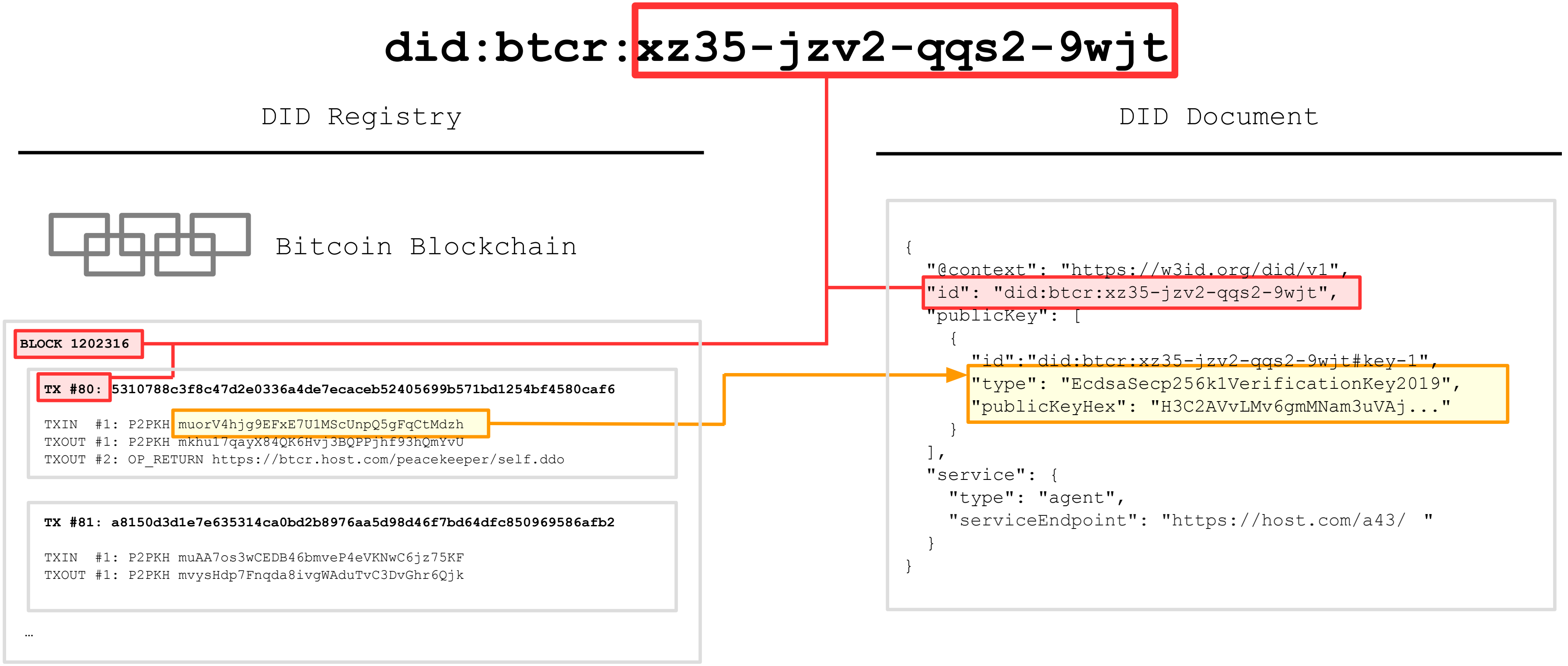


did:btcr:xz35-jzv2-qqs2-9wjt

```
BLOCK 1202316

TX #80: 5310788c3f8c47d2e0336a4de7ecaceb52405699b571bd1254bf4580caf6

TXIN  #1: P2PKH muorV4hjg9EFxE7U1MScUnpQ5gFqCtMdzh
TXOUT #1: P2PKH mkhu17qayX84QK6Hvj3BQPPjhf93hQmYvU
TXOUT #2: OP_RETURN https://btcr.host.com/peacekeeper/self.ddo

TX #81: a8150d3d1e7e635314ca0bd2b8976aa5d98d46f7bd64dfc850969586afb2

TXIN  #1: P2PKH muAA7os3wCEDB46bmveP4eVKNwC6jz75KF
TXOUT #1: P2PKH mvysHdp7Fnqda8ivgWAduTvC3DvGhr6Qjk

...
```

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
  "publicKey": [
    {
      "id":"did:btcr:xz35-jzv2-qqs2-9wjt#key-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "publicKeyHex": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH



did:btcr:xz35-jzv2-qqs2-9wjt

DID Registry

Bitcoin Blockchain

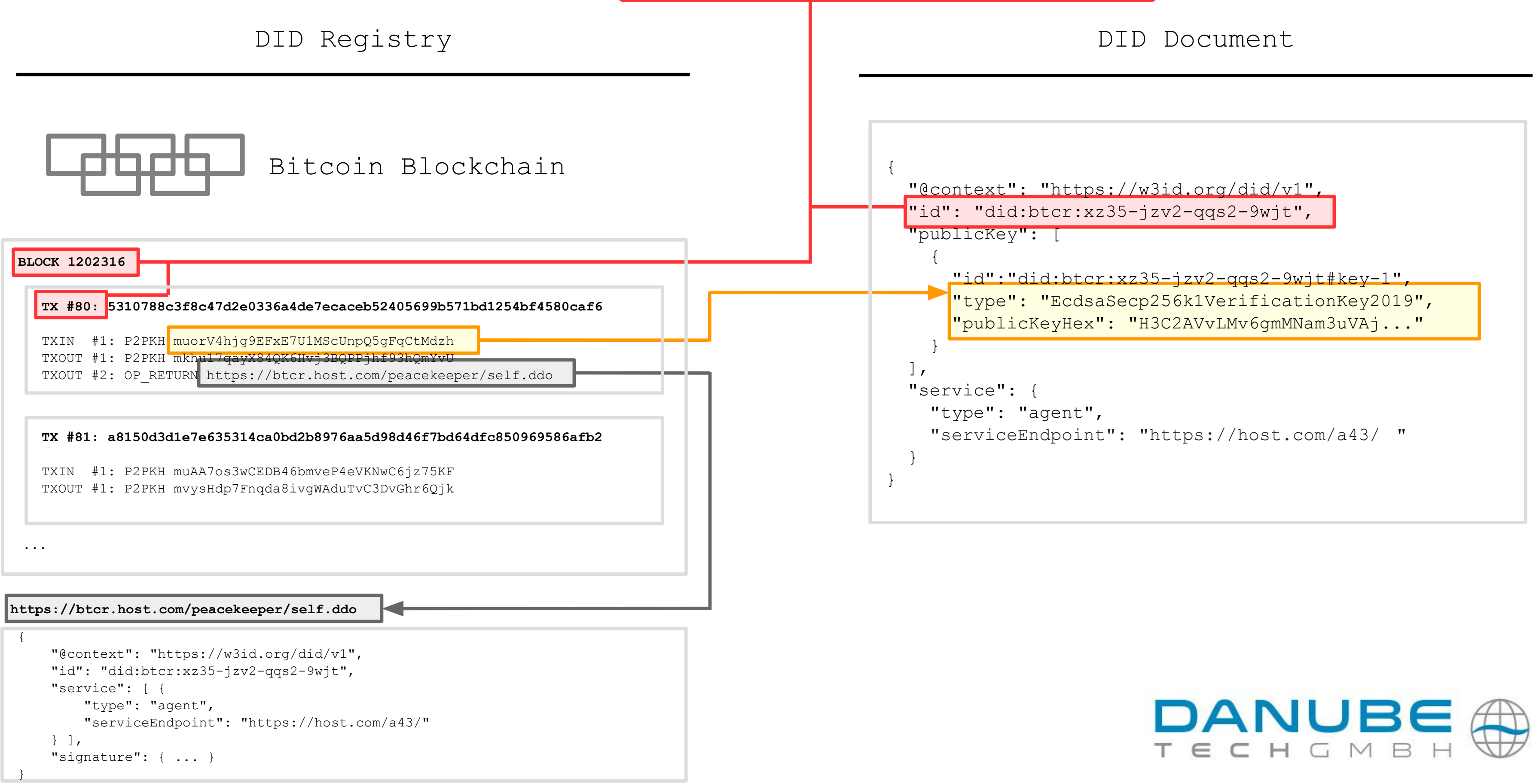
```
BLOCK 1202316

  TX #80: 5310788c3f8c47d2e0336a4de7ecaceb52405699b571bd1254bf4580caf6

  TXIN  #1: P2PKH muorV4hjg9EFxE7U1MScUnpQ5gFqCtMdzh
  TXOUT #1: P2PKH mkhu17qayX84QK6Hvj3BQPPjhf93hQmYvU
  TXOUT #2: OP_RETURN https://btcr.host.com/peacekeeper/self.ddo

  TX #81: a8150d3d1e7e635314ca0bd2b8976aa5d98d46f7bd64dfc850969586afb2

  TXIN  #1: P2PKH muAA7os3wCEDB46bmveP4eVKNwC6jz75KF
  TXOUT #1: P2PKH mvysHdp7Fnqda8ivgWAduTvC3DvGhr6Qjk

…
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
  "publicKey": [
    {
      "id":"did:btcr:xz35-jzv2-qqs2-9wjt#key-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "publicKeyHex": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH

did:btcr:xz35-jzv2-qqs2-9wjt



DID Registry

Bitcoin Blockchain

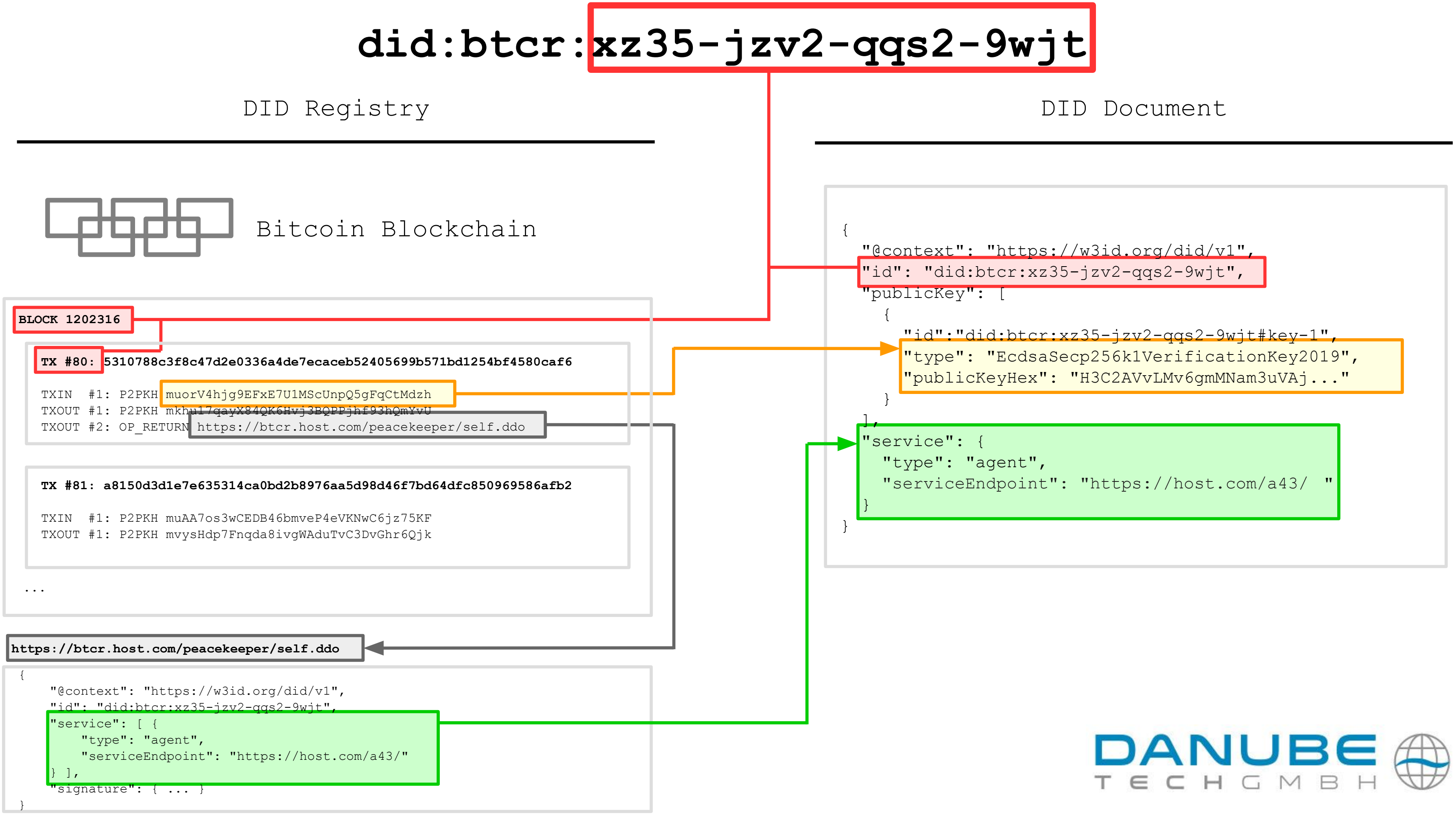
```
BLOCK 1202316

  TX #80: 5310788c3f8c47d2e0336a4de7ecaceb52405699b571bd1254bf4580caf6

  TXIN  #1: P2PKH muorV4hjg9EFxE7U1MScUnpQ5gFqCtMdzh
  TXOUT #1: P2PKH mkhu17qayX84QK6Hvj3BQPPjhf93hQmYvU
  TXOUT #2: OP_RETURN https://btcr.host.com/peacekeeper/self.ddo

  TX #81: a8150d3d1e7e635314ca0bd2b8976aa5d98d46f7bd64dfc850969586afb2

  TXIN  #1: P2PKH muAA7os3wCEDB46bmveP4eVKNwC6jz75KF
  TXOUT #1: P2PKH mvysHdp7Fnqda8ivgWAduTvC3DvGhr6Qjk

...
```

```
https://btcr.host.com/peacekeeper/self.ddo
{
    "@context": "https://w3id.org/did/v1",
    "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
    "service": [ {
        "type": "agent",
        "serviceEndpoint": "https://host.com/a43/"
    } ],
    "signature": { ... }
}
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
  "publicKey": [
    {
      "id":"did:btcr:xz35-jzv2-qqs2-9wjt#key-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "publicKeyHex": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH



did:btcr:xz35-jzv2-qqs2-9wjt

DID Registry

Bitcoin Blockchain

```
BLOCK 1202316

TX #80: 5310788c3f8c47d2e0336a4de7ecaceb52405699b571bd1254bf4580caf6

TXIN  #1: P2PKH muorV4hjg9EFxE7U1MScUnpQ5gFqCtMdzh
TXOUT #1: P2PKH mkhu17qayX84QK6Hvj3BQPPjhf93hQmYvU
TXOUT #2: OP_RETURN https://btcr.host.com/peacekeeper/self.ddo

TX #81: a8150d3d1e7e635314ca0bd2b8976aa5d98d46f7bd64dfc850969586afb2

TXIN  #1: P2PKH muAA7os3wCEDB46bmveP4eVKNwC6jz75KF
TXOUT #1: P2PKH mvysHdp7Fnqda8ivgWAduTvC3DvGhr6Qjk

...
```

https://btcr.host.com/peacekeeper/self.ddo

```
{
    "@context": "https://w3id.org/did/v1",
    "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
    "service": [ {
        "type": "agent",
        "serviceEndpoint": "https://host.com/a43/"
    } ],
    "signature": { ... }
}
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xz35-jzv2-qqs2-9wjt",
  "publicKey": [
    {
      "id":"did:btcr:xz35-jzv2-qqs2-9wjt#key-1",
      "type": "EcdsaSecp256k1VerificationKey2019",
      "publicKeyHex": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

did:v1:test:nym:3AEJTDMSxDDQpyUftju

DID Registry

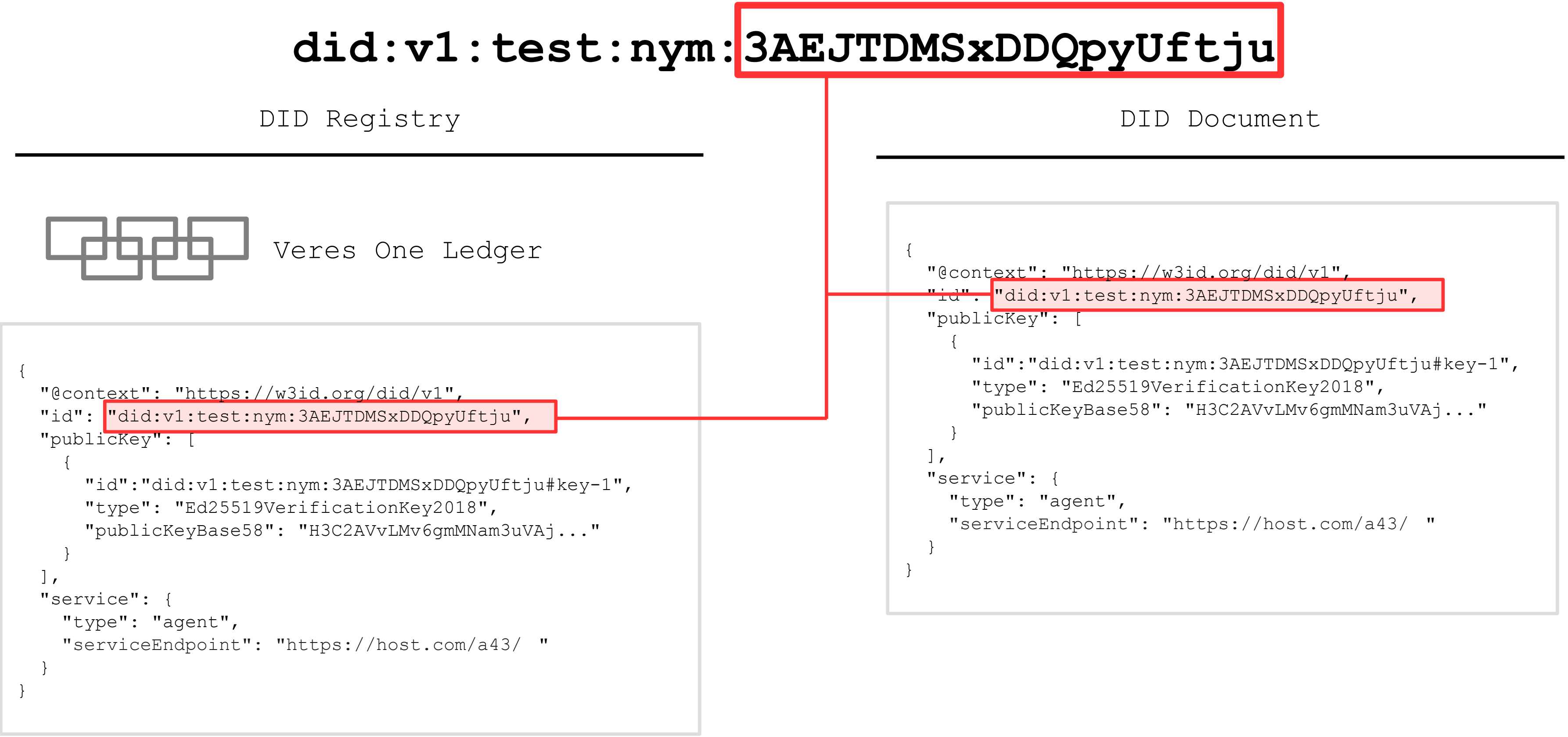
Veres One Ledger

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH



did:v1:test:nym:3AEJTDMSxDDQpyUftju

DID Registry

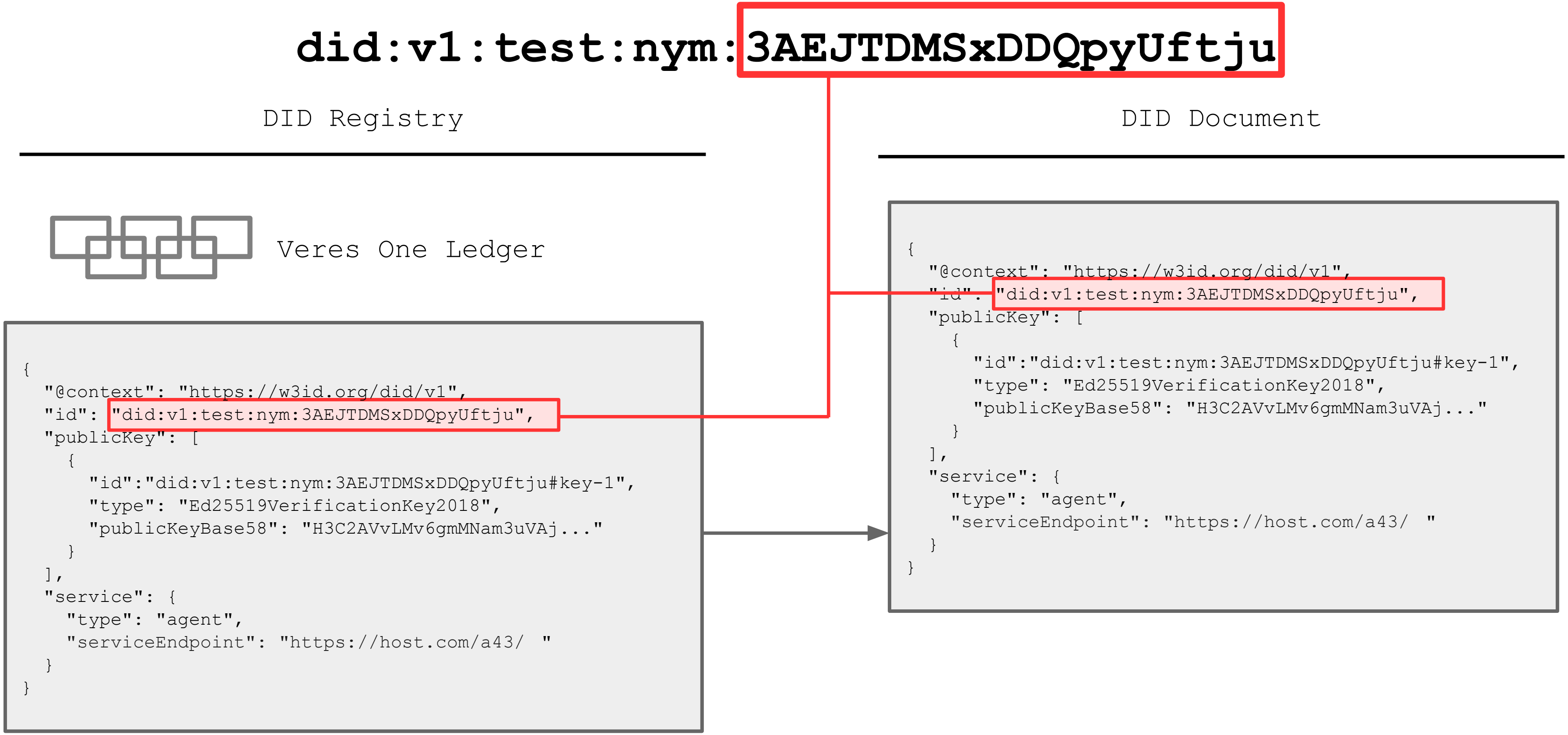
Veres One Ledger

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH



did:v1:test:nym:3AEJTDMSxDDQpyUftju

DID Registry

Veres One Ledger

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:v1:test:nym:3AEJTDMSxDDQpyUftju",
  "publicKey": [
    {
      "id":"did:v1:test:nym:3AEJTDMSxDDQpyUftju#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAj..."
    }
  ],
  "service": {
    "type": "agent",
    "serviceEndpoint": "https://host.com/a43/  "
  }
}
```

DANUBE TECH GMBH

Summary

Perspectives on DIDs - Unanchored (30 min) - Ken Ebert

peer key

sovrin

Peer DIDs and Public Key-based DIDs

PRESENTED BY

Ken Ebert

Software Architect

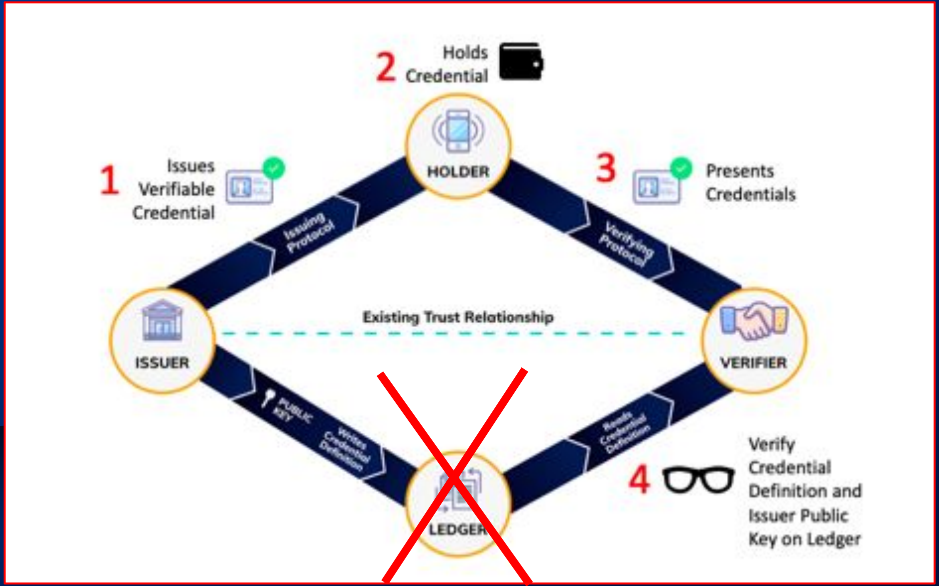
Peer DIDs

https://openssi.github.io/peer-did-method-spec/

What is Peer DID?

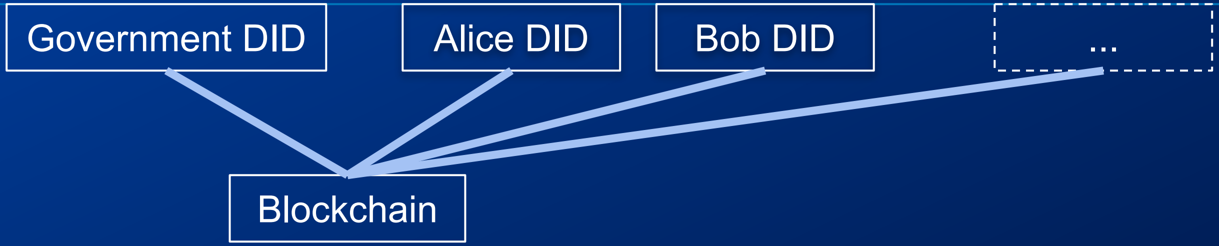
sovrin

- It is a DID!
- A Peer DID is NOT anchored to a public source of truth.

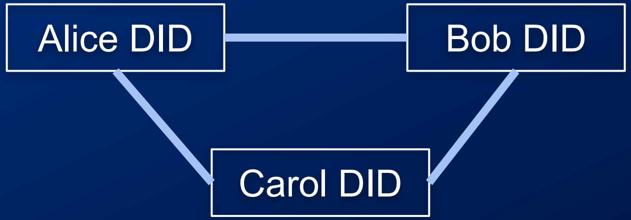

2 Holds Credential
HOLDER
1 Issues Verifiable Credential
Issuing Protocol
3 Presents Credentials
Verifying Protocol
Existing Trust Relationship
ISSUER
VERIFIER
PUBLIC KEY
Writes Credential Definition
Reads Credential Definition
LEDGER
4 Verify Credential Definition and Issuer Public Key on Ledger

DIDs Are About Relationships

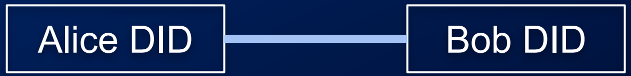
- Anywise DID
  - Unknowable parties
  - Publicly resolveable



- N-wise DID
  - N enumerated parties
  - Privately resolveable



- Pairwise DID
  - 2 parties
  - Privately resolveable



What's a Peer DID Look Like?

Sample

did:peer:11-479cbc07c3f991725836a3aa2a581ca2029198aa420b9d99bc0e131d9f3e2cbe

ABNF

peer-did = "did:peer:" numalgo encalgo "-" numbasis

numalgo = "1"

encalgo = "1"

numbasis = 64*HEXDIGCI

HEXDIGCI = HEXDIG / "a" / "b" / "c" / "d" / "e" / "f”

Benefits of Peer DIDs

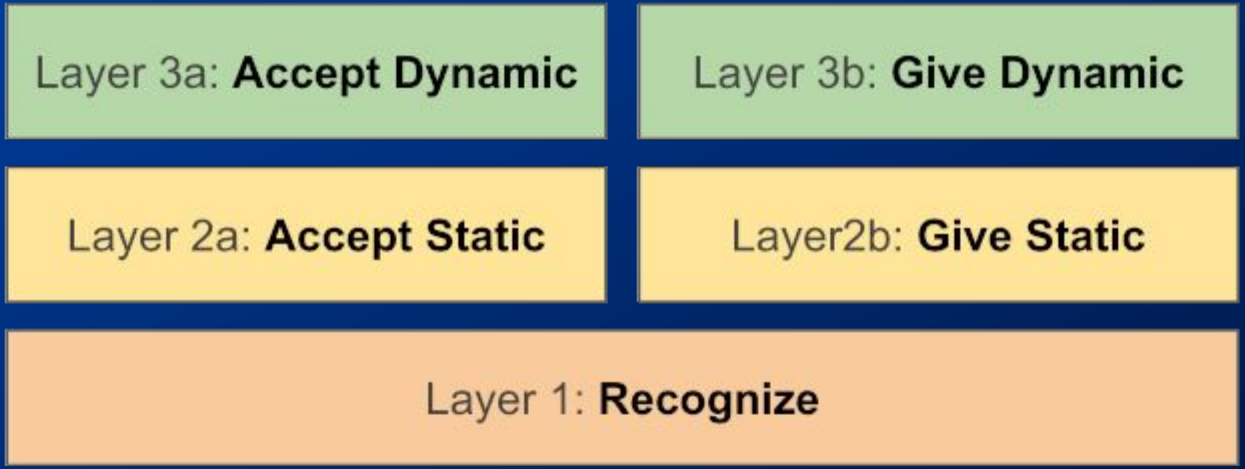
sovrin

- Cheap: no transaction costs
- Fast
- Scalable: as a function of the participants
- Secure
- Reduced PI and privacy concerns
- Independent of any ledger: minimal political or technical baggage
- Graftable into other DID ecosystems

Challenges of Peer DIDs

- Backing storage
  - DID doc + metadata + history
- Synchronization
- Multiple agents!
- Conflict-free replicated data type (CRDT)
- Protocol (CRUD)

Layers of Support



Layer 3a: **Accept Dynamic**

Layer 3b: **Give Dynamic**

Layer 2a: **Accept Static**

Layer2b: **Give Static**

Layer 1: **Recognize**

Public Key-based DIDs

https://digitalbazaar.github.io/did-method-key/

What's a Public Key-based DID Look Like?

Sample (ed25519 public key)

did:key:z6MkpTHR8VNsBxYAAWHut2Geadd9jSwuBV8xRoAnwWsdvktH

ABNF

```
did-key = "did:key:" multibase( multicodec( public-key ) )

multibase = function(bytes) => [1-9A-Za-z]

multicodec = function(codec, bytes) => codec[ed25519publickey -> 0xed, ...] bytes

public-key = [0x00-0xff]
```

Benefits of Public Key DIDs

- Self-describing
- Cheap: no transaction costs
- Fast
- Scalable: as a function of the participants
- Secure
- No PI and privacy concerns
- Independent of any ledger: minimal political or technical baggage
- Graftable into other DID ecosystems

Challenges of Public Key DIDs

- No key rotation
- Unclear if they should be different from did:peer
  - Digital Bazaar and Daniel Hardman are puzzling it out at present

sovrin

identity for all

Questions?

✉ ken@sovrin.org

🌐 sovrin.org

Ken Ebert
Software Architect

Thank you

Perspectives on DIDs - Layer 2 (30 min)

ION Layer 2 DID Network

ION

Layer 2 DID Network

1.

The Scaling Trilema

Creating secure, decentralized systems that run at world-scale

Three critical components:

Decentralization

Without this property, many proposed solutions do not deliver sufficiently differentiated benefit over those built using traditional systems.

Scalability

If decentralized systems (i.e. blockchains, DLTs) are to deliver on the benefits they promise, they must support billions of participating entities.

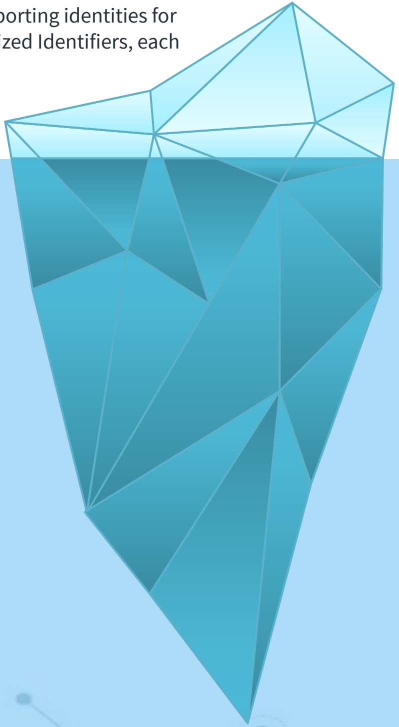
Security

These systems must achieve decentralization at global scale, while maintaining a high level of security.

The Scale of Decentralized Identity:

Human Identity

There are 7.5 billion humans on Earth currently. At bare minimum, a decentralized identity system must be capable of supporting identities for all of them. Each person may have multiple Decentralized Identifiers, each requiring their own PKI lineage.



Identity of All Things.

Human identity is just the tip of the iceberg – there is an entire world containing hundreds of billions of devices, machines, apps, and other entities, both tangible and virtual.

Requirements for DPKI:

- Global, immutable, append-only log
- No central providers or authorities
- Censorship and tamper resistant

Key Realization

Identifiers and PKI do not suffer from the same double spend problem money does, because DIDs do not need to be transferred between parties like assets. However, you must still prevent double issuance and ensure all parties on the DID network can derive a single deterministic PKI state for an identifier.

How might these differences in requirements affect how we approach the architecture of a DID network?

2.

Technical Overview

Architecture and Protocol Details

What is ION?

ION is a public, permissionless, decentralized DID overlay network that runs on Bitcoin, and leverages a deterministic DPKI protocol, called Sidetree.

Technical Assumptions:

No secondary consensus required

ION nodes do not require a secondary consensus system to derive the correct PKI state of IDs.

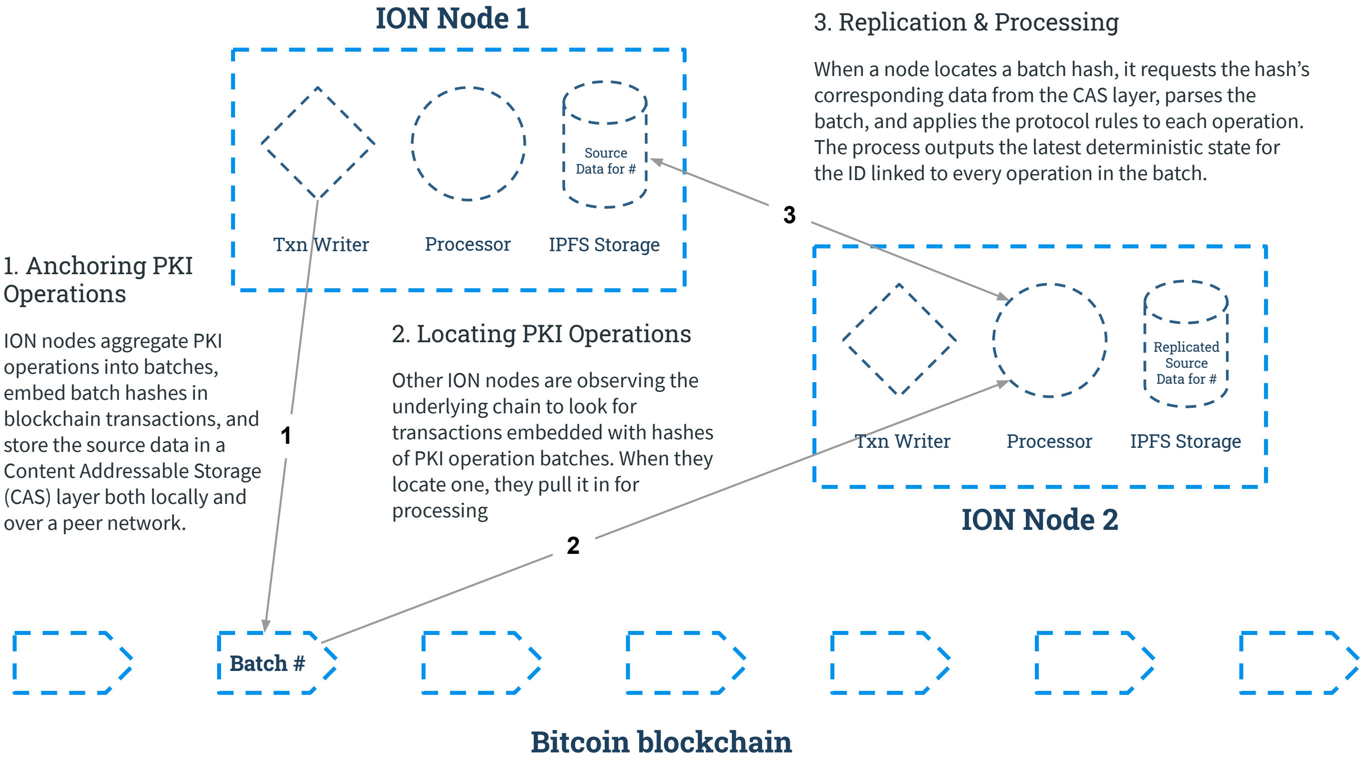
No conflicting states are allowed

The protocol eliminates conflicting PKI states via a strict, deterministic rule set that each node applies individually.

IDs are not transferable between entities

Transferring ownership of IDs between untrusting parties, as you would crypto-assets like Bitcoin, is not a supported function.

System Overview



1. Anchoring PKI Operations

ION nodes aggregate PKI operations into batches, embed batch hashes in blockchain transactions, and store the source data in a Content Addressable Storage (CAS) layer both locally and over a peer network.

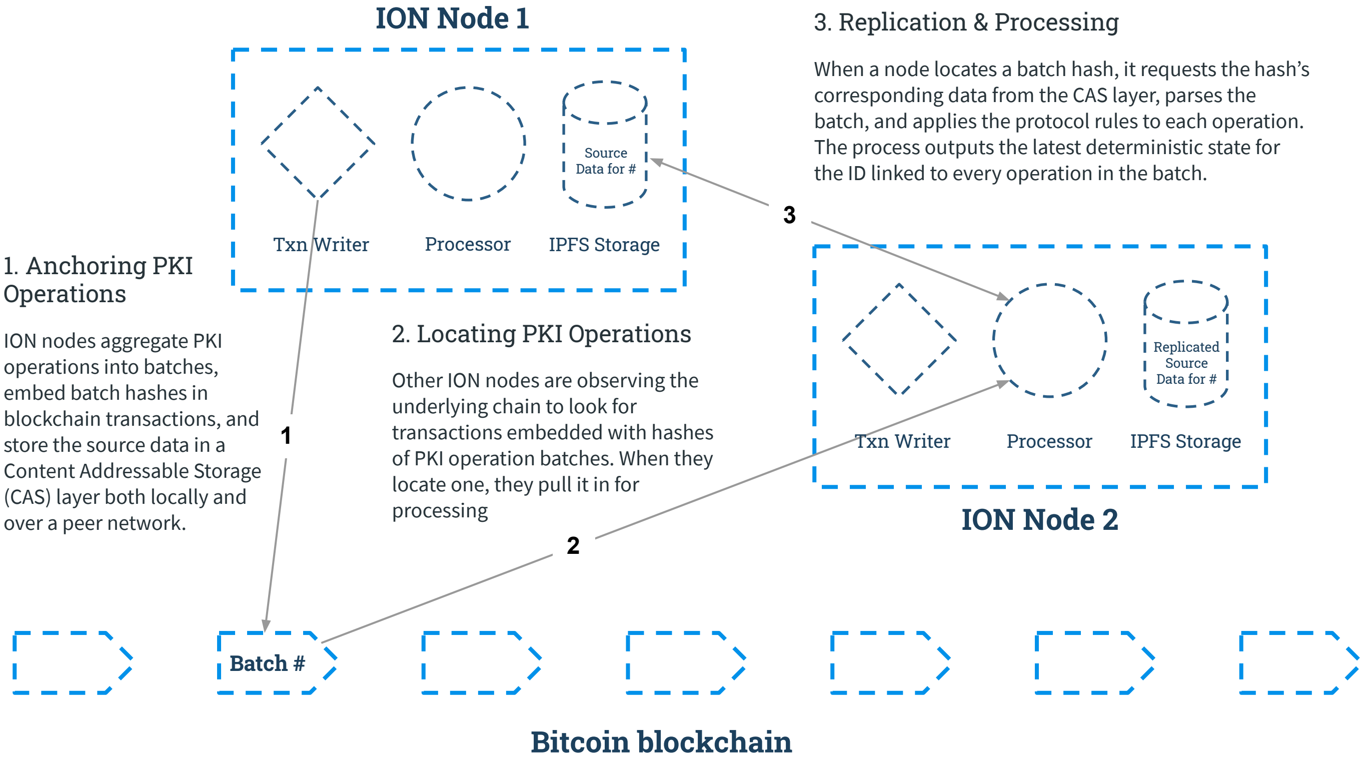
2. Locating PKI Operations

Other ION nodes are observing the underlying chain to look for transactions embedded with hashes of PKI operation batches. When they locate one, they pull it in for processing

3. Replication & Processing

When a node locates a batch hash, it requests the hash's corresponding data from the CAS layer, parses the batch, and applies the protocol rules to each operation. The process outputs the latest deterministic state for the ID linked to every operation in the batch.

System Overview



1. Anchoring PKI Operations

ION nodes aggregate PKI operations into batches, embed batch hashes in blockchain transactions, and store the source data in a Content Addressable Storage (CAS) layer both locally and over a peer network.

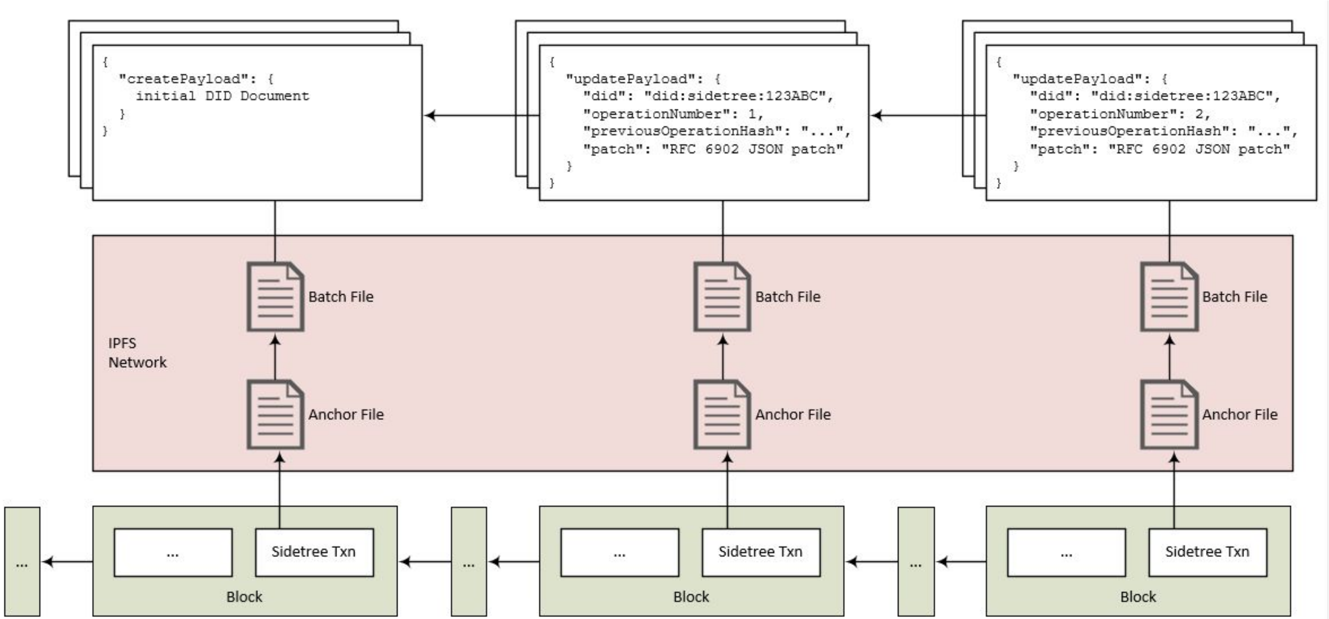
2. Locating PKI Operations

Other ION nodes are observing the underlying chain to look for transactions embedded with hashes of PKI operation batches. When they locate one, they pull it in for processing

3. Replication & Processing

When a node locates a batch hash, it requests the hash's corresponding data from the CAS layer, parses the batch, and applies the protocol rules to each operation. The process outputs the latest deterministic state for the ID linked to every operation in the batch.

Anatomy of an Operation



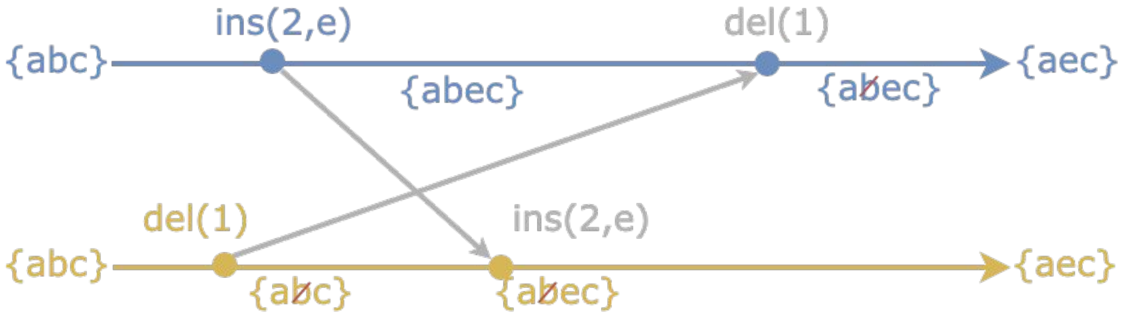
```
{
  "createPayload": {
    initial DID Document
  }
}
```

```
{
  "updatePayload": {
    "did": "did:sidetree:123ABC",
    "operationNumber": 1,
    "previousOperationHash": "...",
    "patch": "RFC 6902 JSON patch"
  }
}
```

```
{
  "updatePayload": {
    "did": "did:sidetree:123ABC",
    "operationNumber": 2,
    "previousOperationHash": "...",
    "patch": "RFC 6902 JSON patch"
  }
}
```

DID PKI State Convergence

- The Sidetree protocol that underpins ION uses a form of Conflict-Free Resolution Datatype to converge the PKI state of DIDs.
- CRDTs deterministically merge changes to objects without a centralized database, trusted coordinator, etc. Typically, ordering of operations in a CRDT is based on vector clocks (Lamport timestamps).
- Sidetree uses a Delta-based CRDT, but instead of writers subjectively incremented vector clocks, operations are anchored in batches to the blockchain, which acts as a decentralized sequencing oracle that orders operations in a single, deterministic, linear history.



Traditional Delta-based CRDT converging changes using vector clocks

ION enables key features to enhance our offerings:

Massive Scale

The network can collectively process tens of thousands of operations per second, even on consumer-grade machines.

Cost Efficient

Decentralized blockchains provide unique features, but the come at a high monetary/energy cost. ION's batching mechanism reduces per-unit op costs by several orders of magnitude.

Permissionless

Many other blockchain-based systems used for identity purposes rely on central authority schemes to scale their networks. ION is able to meet and exceed scale requirements while remaining decentralized.

Flexible Nodes

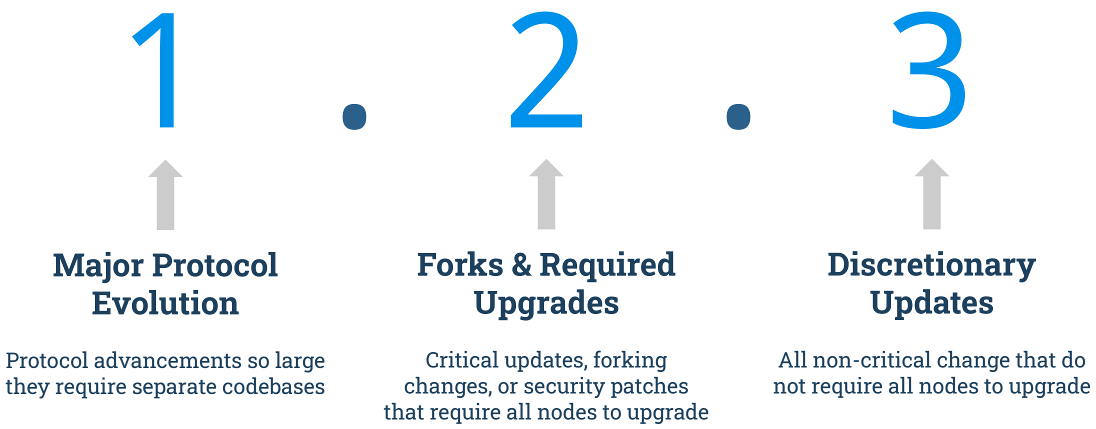
Unlike a blockchain, nodes of the ION network that runs atop the underlying decentralized system do not need to maintain the full history of transactions.

3.

Building the Network

ION is an organic system that requires care to develop, grow, and flourish.

Protocol Development and Network Upgrades



1 . 2 . 3

Major Protocol Evolution

Protocol advancements so large they require separate codebases

Forks & Required Upgrades

Critical updates, forking changes, or security patches that require all nodes to upgrade

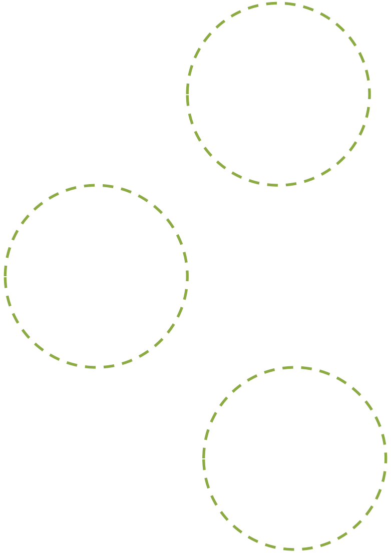
Discretionary Updates

All non-critical change that do not require all nodes to upgrade

Upgrade Process

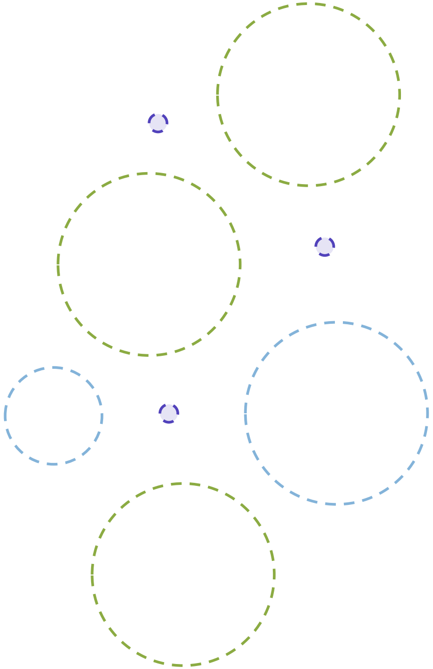
1. Tag release
2. Update install guides
3. Add an entry to the change log
4. Broadcast upgrade to node operators

The path to a robust network - a three stage journey:



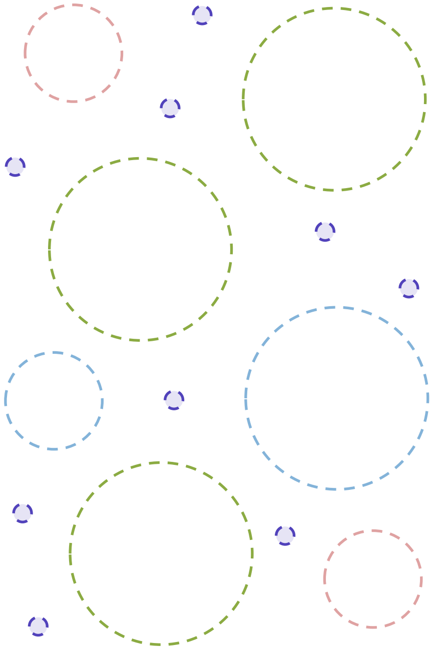
Stage 1

Larger entities run full nodes to jumpstart the network



Stage 2

Entities with product needs and early adopter hobbyists start running full nodes ad hoc



Stage 3

The long tail of developers, users, and organizations run a mix of light and full nodes to suit their needs

How to get involved:

Help shape specifications

To ensure these systems meet the needs of all the individuals, organizations, and use cases that will rely on them, help shape the Sidetree protocol spec and technical decisions in ION.

Contribute to open source development

Contribute open source code to the DIF Sidetree protocol and ION node code in the DIF repositories on GitHub.

Run a node, participate in the ecosystem

In order to realize the value decentralized identity can deliver, participate in running the foundational components it relies on.

Perspectives on DIDs - Alternatives (30 min)

git web ipld etc

Alternative DID Methods

W3C TPAC 2019 - Fukuoka

Manu Sporny - CEO - Digital Bazaar

Manu Sporny | CEO | Digital Bazaar

- Co-Inventor of Verifiable Credentials & Decentralized Identifiers
- Co-Inventor of JSON-LD

- Co-Founder of Digital Bazaar & Veres One
- 10+ Years in Web Standards
- Customers in Government, Supply Chain, Finance, Education, and Healthcare

Email: msporny@digitalbazaar.com

https://www.linkedin.com/in/manusporny/

What is an Alternative DID Method?

Alternative DID Methods...

Typically fall into at least one of these categories.

- Based on deployed tech
- Utilize existing large networks
- May not be truly "decentralized"
- Doesn't use a cryptocurrency
- Bridge the old world to the new, making the adjacent possible... possible.

did:web

did:web

A DID Method for the Web

- did:web:example.com/jdoe
- Pros
  - It's a resource on the Web
  - Works today, zero changes to Web
  - Uses existing CA system
- Cons
  - No revision control
  - No audit trail
  - Uses existing CA system

did:git

did:git

A DID Method for developers

- did:git:a7c...38a/b2f...9d1
- Pros
  - Blockchain-like version control
  - Digitally signed transaction history
  - Highly decentralized
- Cons
  - Undetectable "forking" possible
  - No single point of truth
  - High potential for DoS

did:ipid

did:ipid

A DID Method layered on top of a DHT-based clustered file system

- did:ipid:12D...y5w
- Pros
  - Cheap to create (self-hosted)
  - Possible to replicate
  - Network is fault-tolerant
- Cons
  - DIDs can disappear
  - Possibly expensive to maintain

did:PROPRIETARY

did:PROPRIETARY

DID Methods where the namespace is owned by an organization.

- did:facebook:jdoe, did:gmail:jdoe, did:linkedin:jdoe
- Pros
  - Cheap to create and maintain
  - Clear responsibilities
  - Extremely reliable network
- Cons
  - Centralized network
  - Centralized governance
  - Not portable

Questions?

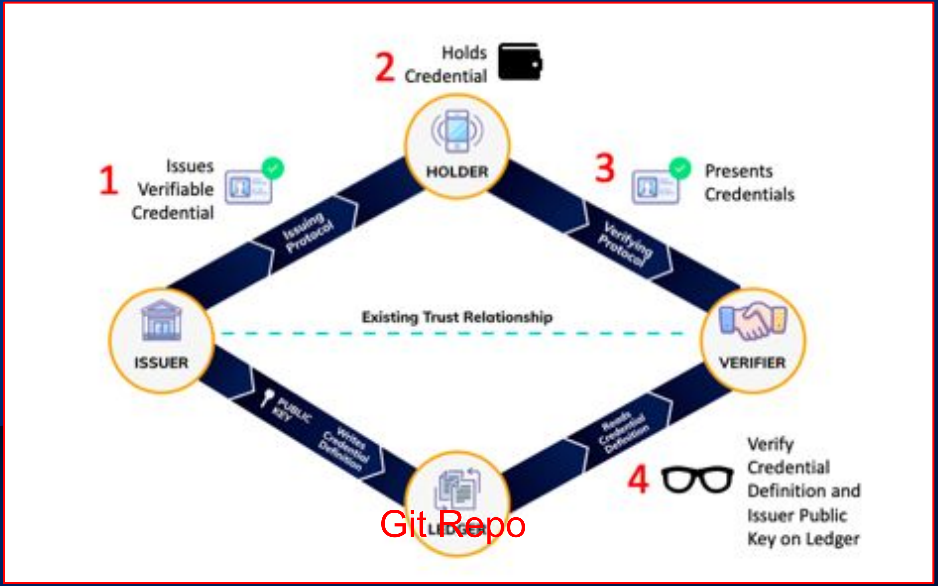
Appendix: Git DIDs

https://github.com/dhuseby/did-git-spec/blob/master/did-git-spec.md

What is Git DID?

sovrin

- It is a DID!
- A Git DID is anchored to a git repository as its source of truth.



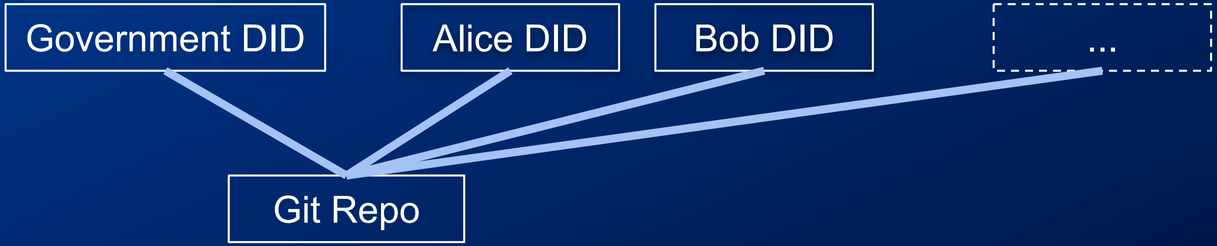
2 Holds Credential
HOLDER
1 Issues Verifiable Credential
Issuing Protocol
3 Presents Credentials
Verifying Protocol
Existing Trust Relationship
ISSUER
VERIFIER
PUBLIC KEY
Writes Credential Definition
Reads Credential Definition
LEDGER
Git Repo
4 Verify Credential Definition and Issuer Public Key on Ledger

DIDs Are About Relationships

sovrin

Relationships are with the git repo as an entity and store

- Anywise DID
- N-wise DID
- Pairwise DID



What's Git DID Look Like?

sovrin

Sample

did:git:625557b5a9cdf399205820a2a716da897e2f9657

ABNF

```
git-did = "did:git:repo-id" 1*(":" contributor-id) 1*(";" did-service)
        1*("/" did-path) 1*("?" did-query) 1*("#" did-fragment)
repo-id = commit-id
contributor-id = commit-id
commit-id = 40*(lowerhex)
lowerhex = "0" / "1" / "2" / "3" / "4" / "5" / "6" / "7" / "8" / "9" / "a" /"b" / "c" / "d" / "e" / "f"
```

Benefits of Git DIDs

sovrin

- Better solution for digitally signed commits
- In-band identity management means Git repos become self-verifiable.
- Git hook enforcement means project governance can be automatic.
- Fully anonymous open-source projects are possible.
- Independent of any ledger
- Contributions to open-source projects can form proof-of-work trust in anonymous identities.

Challenges of Git DIDs

sovrin

- Currently modifying Git to support signing tools other than GPG
- Currently building a new DID-powered signing tool
- Selling the Git community on the value of SSI
- Currently tied to git & SHA-1
- Merge conflict resolution performed by humans
- Git repos are not CRDTs
- PII Risks similar to a blockchain
- No global source of truth to establish which is the canonical repo

Lunch

Documenting Use Cases (Joe, 60 min)

Use Cases for Decentralized Identifiers

W3C TPAC 2019
DID WORKING GROUP

JOE ANDRIEU JOE@LEGREQ.COM

DID WG Charter

"Other non-normative documents may be created such as:

Decentralized Identifier Use Cases v1.0

...

The Credentials Community Group has developed a set of initial use cases and requirements that will serve as input for this document."

Agenda

- Why Use Cases
- Examples
- Good Use Cases
- Lessons Learned from Verifiable Credentials
- The CCG’s DID Use Case Document
- Moving forward

Why Use Cases

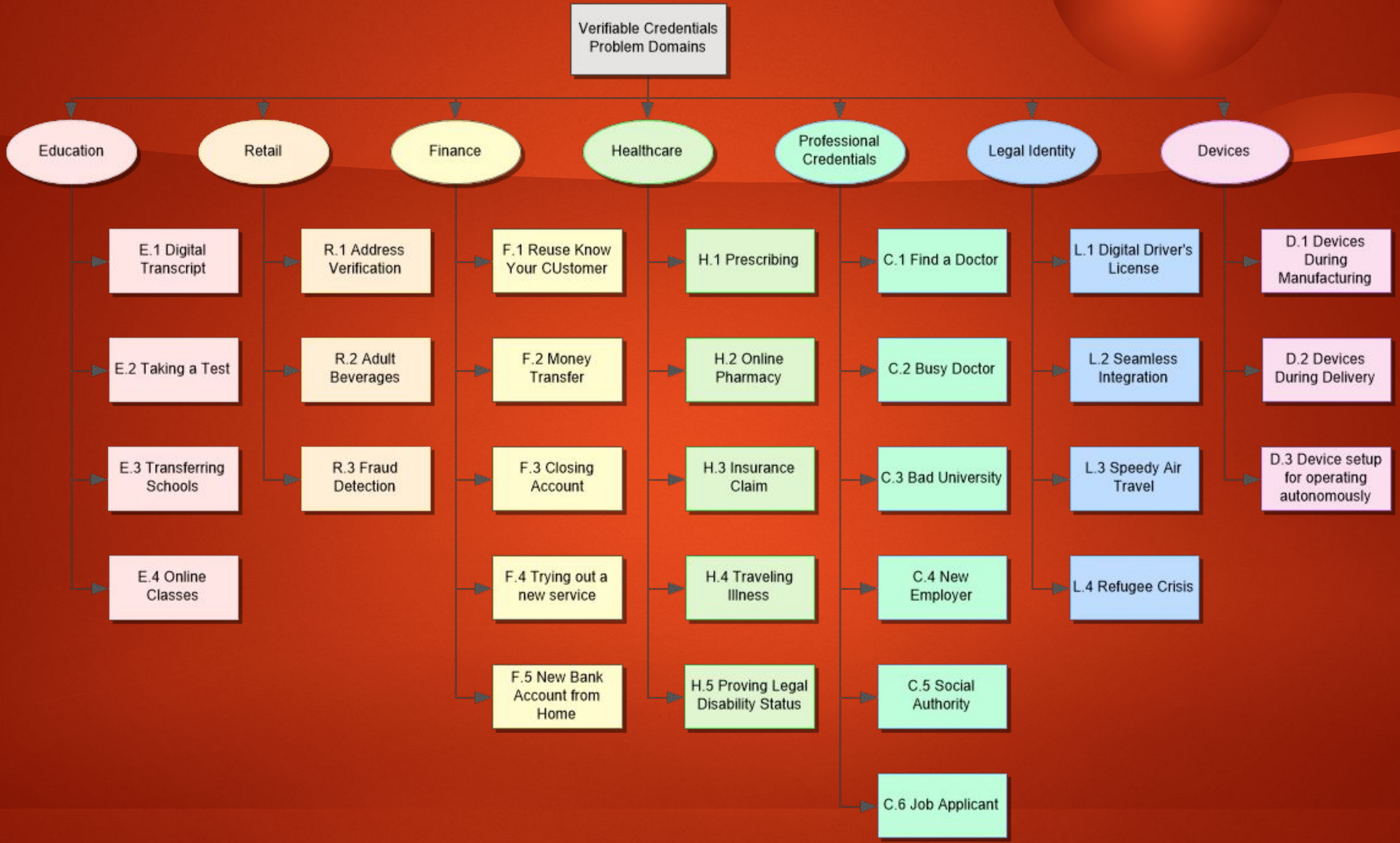
- Illustrate how technology can be used.
- Provide guidance for technical decisions.
- Separate discussions of what is possible from the solution.
- Focus attention on the human requirements driving technical choices.

Example – Title & Scenario

E.2 Taking a test

Eunice is about to take her ACT (a test used to evaluate her readiness for college). When she arrives at the testing center, she is required to present identification. Her government-issued identity certificate is acceptable, as the verifiable credentials contained in it reflect all of the required attributes and it is difficult to counterfeit.

Example – Domain Map



Verifiable Credentials Problem Domains

Education

E.1 Digital Transcript

E.2 Taking a Test

E.3 Transferring Schools

E.4 Online Classes

Retail

R.1 Address Verification

R.2 Adult Beverages

R.3 Fraud Detection

Finance

F.1 Reuse Know Your CUstomer

F.2 Money Transfer

F.3 Closing Account

F.4 Trying out a new service

F.5 New Bank Account from Home

Healthcare

H.1 Prescribing

H.2 Online Pharmacy

H.3 Insurance Claim

H.4 Traveling Illness

H.5 Proving Legal Disability Status

Professional Credentials

C.1 Find a Doctor

C.2 Busy Doctor

C.3 Bad University

C.4 New Employer

C.5 Social Authority

C.6 Job Applicant

Legal Identity

L.1 Digital Driver's License

L.2 Seamless Integration

L.3 Speedy Air Travel

L.4 Refugee Crisis

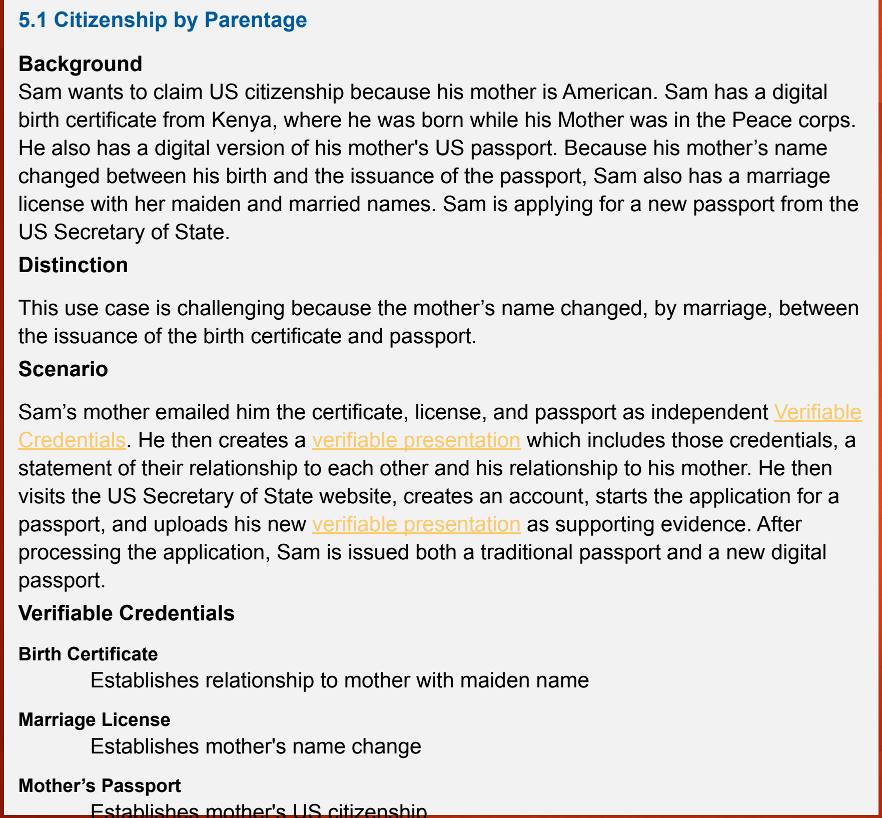
Devices

D.1 Devices During Manufacturing

D.2 Devices During Delivery

D.3 Device setup for operating autonomously

Example – Focal Use Case



5.1 Citizenship by Parentage

Background

Sam wants to claim US citizenship because his mother is American. Sam has a digital birth certificate from Kenya, where he was born while his Mother was in the Peace corps. He also has a digital version of his mother's US passport. Because his mother’s name changed between his birth and the issuance of the passport, Sam also has a marriage license with her maiden and married names. Sam is applying for a new passport from the US Secretary of State.

Distinction

This use case is challenging because the mother’s name changed, by marriage, between the issuance of the birth certificate and passport.

Scenario

Sam’s mother emailed him the certificate, license, and passport as independent Verifiable Credentials. He then creates a verifiable presentation which includes those credentials, a statement of their relationship to each other and his relationship to his mother. He then visits the US Secretary of State website, creates an account, starts the application for a passport, and uploads his new verifiable presentation as supporting evidence. After processing the application, Sam is issued both a traditional passport and a new digital passport.

Verifiable Credentials

Birth Certificate

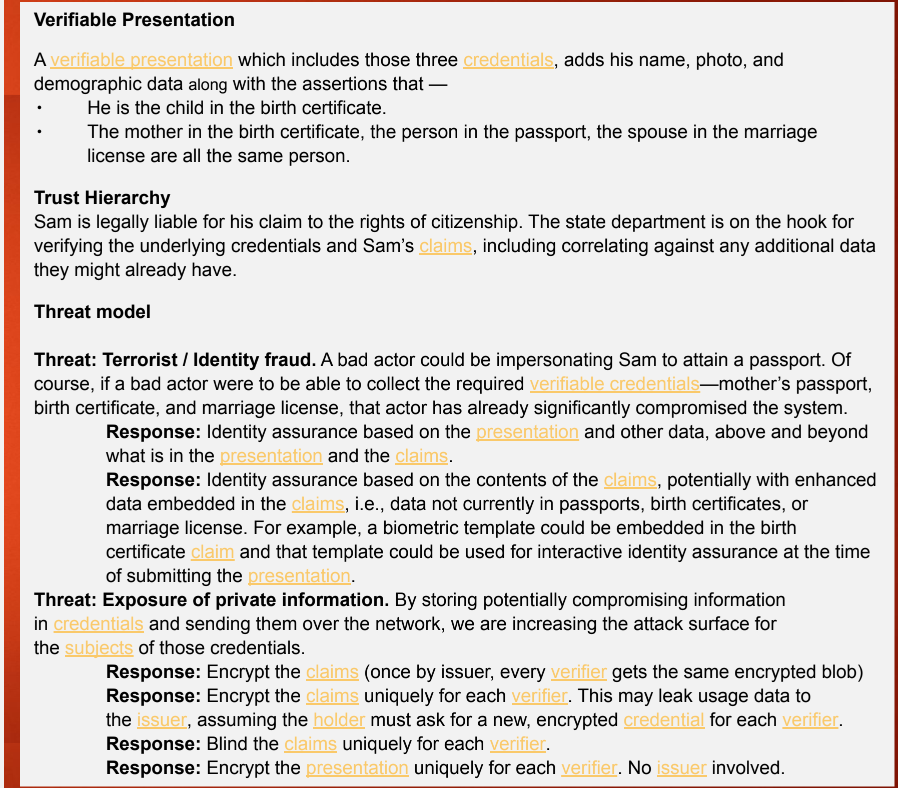
Establishes relationship to mother with maiden name

Marriage License

Establishes mother's name change

Mother’s Passport

Establishes mother's US citizenship



Verifiable Presentation

A verifiable presentation which includes those three credentials, adds his name, photo, and demographic data along with the assertions that —

- He is the child in the birth certificate.
- The mother in the birth certificate, the person in the passport, the spouse in the marriage license are all the same person.

Trust Hierarchy

Sam is legally liable for his claim to the rights of citizenship. The state department is on the hook for verifying the underlying credentials and Sam’s claims, including correlating against any additional data they might already have.

Threat model

Threat: Terrorist / Identity fraud. A bad actor could be impersonating Sam to attain a passport. Of course, if a bad actor were to be able to collect the required verifiable credentials—mother’s passport, birth certificate, and marriage license, that actor has already significantly compromised the system.

Response: Identity assurance based on the presentation and other data, above and beyond what is in the presentation and the claims.

Response: Identity assurance based on the contents of the claims, potentially with enhanced data embedded in the claims, i.e., data not currently in passports, birth certificates, or marriage license. For example, a biometric template could be embedded in the birth certificate claim and that template could be used for interactive identity assurance at the time of submitting the presentation.

Threat: Exposure of private information. By storing potentially compromising information in credentials and sending them over the network, we are increasing the attack surface for the subjects of those credentials.

Response: Encrypt the claims (once by issuer, every verifier gets the same encrypted blob)

Response: Encrypt the claims uniquely for each verifier. This may leak usage data to the issuer, assuming the holder must ask for a new, encrypted credential for each verifier.

Response: Blind the claims uniquely for each verifier.

Response: Encrypt the presentation uniquely for each verifier. No issuer involved.

Good Use Cases

- Concrete
- Distinctive
- Illustrate unique features of the technology
- Memorable
- Short

Lessons Learned from Verifiable Credentials

- Titles Matter
- Multiple levels of breadth
  - Domain Map
  - Scenarios
  - Focal Use Cases
- Collect inferred feature requirements
- Track coverage against features

CCG's DID Use Case Document

- https://w3c-ccg.github.io/did-use-cases/

Proposed Content

- Domain Map + Brief Use Cases
  - 20-30 Titles + Scenarios
- 3-5 Focal Use Cases
  - Background
  - Example code
  - Threat Model
- DID Actions
- Derived Feature Requirements
- Coverage Map

Moving forward

- How shall we create the DID Use Cases document?
- Shall we start with the current CCG DID Use Cases as a starting point?
- Who wants to help drive this work?
- How do we want to coordinate?

Afternoon break

Technical Direction (Chairs, 60 min)

Technical direction

- How should we start?
- Should we adopt the CCG “Decentralized Identifiers (DIDs) v0.13 Data Model and Syntaxes” Final report as the official starting point for our spec?
  - If yes, how do we copy the document? (Or can we just copy the repo)
  - If no, what do we start with?

CCG GitHub Issues (Chairs, 60 min)

CCG DID spec Issues

- 52 open issues
- currently triaged as they were by the CCG
- some issues have multiple tags
- Issue Categories we're going to look at (these tags cover all open issues)
  - Questions
  - Clarifications
  - Discussions
  - Editorial
  - Elsewhere

Questions

- 5 issues tagged `question`
- Some of them have been addressed, but are waiting for feedback from OP before being closed.
- Questions:
  - If an existing DID Document has a Service Endpoint fragment, what are the primary keys to be used if the Service Endpoint (or elements of the Service Endpoint) need to be replaced, updated, or deleted?
  - Is method-specific-id supposed to be equivalent to param-char?
  - Is the "contexts" the same "Contexts" defined in section 5.1?
  - Which version of the ABNF specification are we claiming conformance with?
  - Are service endpoints transport layer or application layer specific?

Clarifications

- 11 issues tagged `clarify`
- non-testable normative statements
- requests for greater specificity
- re-phrasings
- document scope questions
- some of them have been responded to, but all need more attention in order to close.

Discussions

- 9 issues tagged `discuss`
- Bigger questions
- possibly more likely to stir disagreement about the best way to respond
- or an invitation to do some bike-shedding
- Some examples:
    - authentication as a mechanism for proving control of a DID
    - Use colon separator or kebab-case for method-specific DID parameter names?
    - Standardize the key revocation list

Editorial

- 17 open issues tagged `editorial`
- Re-wordings
- calls for fact-checking
- corrections
- complaints about the introduction
- questions
    - e.g., do we need to register a mime type for DIDs?
- etc.

Other

- 13 open issues tagged `elsewhere`
- These are issues that may not actually relate to the DID Spec.
  - possibly more of a did resolution question

Questions?

- Where do we go from here?
- Do we pull these issues in?
- If we do:
  - the triage and tagging needs to be verified
  - do we pull them all over or cherry pick?

Dinner Tonight: 19:00-
https://www.jrk-hotels.co.jp/Fukuoka/en/restaurant/
‘Akasaka Umaya’ in the basement of the JR Kyushu Hotel Blossom Fukuoka, Near JR Hakata Station
Fixed Course (see next slide), 5000yen in cash (or in 42 euros) or credit card.
MEET: 18:30 Grandfloor for taxi. (If you’re late, you’re on your own)
k-sako@ab.jp.nec.com

Dinner Course (subject to minor change)/Free drinks

- Seasonal Appetizer
- Tofu
- Sashimi (raw fish; sushi without rice)
- Grilled skewers
- Ham Salad
- Fried Scallops and Eggplant
- Chicken with Miso paste
- Rice/Noodles
- Dessert

Decentralized Identifier WG TPAC Sessions

Day 2: September 17, 2019
Chairs: Dan Burnett, Brent Zundel
Location: Hilton Fukuoka Sea Hawk, 1st Floor, Navis B

Welcome

8:30	Good Morning, Agenda preview	Chairs
9:00	Test Suite	
10:00	Teleconferences, next f2f, etc.	Chairs
10:30	Morning Break	
11:00	Decentralization Rubric	Joe Andrieu
12:00	Adopting Work - Concerns and Requirements	Manu Sporny
13:00	Lunch	
14:00	DID Resolution - which pieces are ours	Markus Sabadello
14:30	WoT joint session	
15:00	Working through issues	
15:30	Break	
16:00	Open Topics	
16:30	PING joint session	
17:00	Open Topics	

Test Suite (60 min)

Test suite

- Should the WG pull in the existing CCG did-spec test suite?
- At what point will the spec be mature enough for active test suite development to make sense?
    - probably now
- If resolution could be added (non-normatively) into the test suite, which DID method should be used?
    - did:peer or did:key may be good candidates
- Will we have active co-development of implementations as the spec matures which would benefit from an active test suite?
    - we expect so
- Could the test suite be a place for non-normative tests written by the DID methods?
    - probably not, but perhaps it could somehow link to the did method test suites
- Is there an active developer (in the WG) who would maintain the test suite?
    - Yes - Digital Bazaar will supply an engineer (more volunteers are welcome)
    - Evernym is also working to secure at least a partial resource to contribute

Teleconferences, next f2f (Chairs, 30 min)

Logistics of further meetings - chairs seeking input

- Teleconferences
  - Expected weekly (by charter), nothing is perfect because we live on a globe
  - Chairs will decide, considering who will contribute, who might contribute, sharing pain, and grou
  - Existing options:
    - VCWG time slot (Tue 11 am-noon Boston / 1700-1800 CET)
    - p dynamics, but we need inputCCG DID time slot (Thu 4-5 pm Boston / 2200-2300 CET)
  - Doodle Poll for gathering info, WAIT FOR INSTRUCTIONS ~~(https://doodle.com/poll/mnru35rtik6mtsxx)~~
  - Based on which time zone? (Determines who is affected by Daylight Savings Time)
- Face-to-face meeting
  - General time frame (Jan-Feb)
  - Events we can tack onto
  - Other events to avoid (FIDO Feb 2-8, RSA Feb 23-28, RWC Jan 8-10)
  - Location/hosting? (Feb/Mar Amsterdam, NJ/Brussels, Redmond/Bay Area/London)

Morning break

Decentralization Rubric (Joe, 60 min)

A Rubric for Decentralization of DID Methods

W3C TPAC 2019
DID WORKING GROUP

JOE ANDRIEU JOE@LEGREQ.COM

DID WG Charter

Provide a rubric of decentralized characteristics for DID Method specifications. This allows the DID Method specifications to self-certify, or independent third parties to evaluate, the DID Method specification's level of adherence to principles of decentralization.

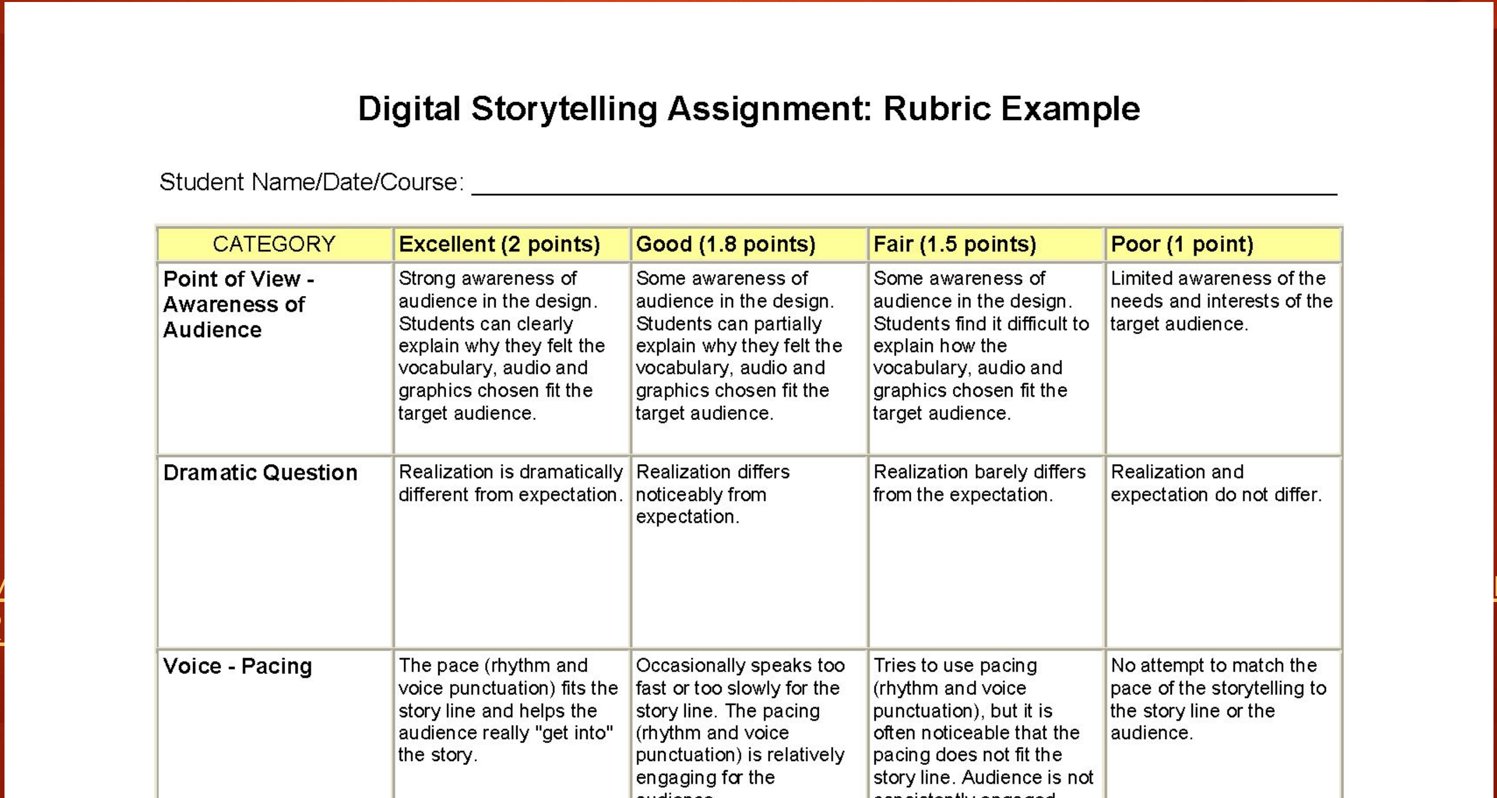
Agenda

- What's a Rubric
- Why a Rubric for decentralization?
- Illustration
- History
- Next Steps

What's a Rubric

- vm a scoring guide used to evaluate performance, a product, or a project.
- Taken from education

Example 1



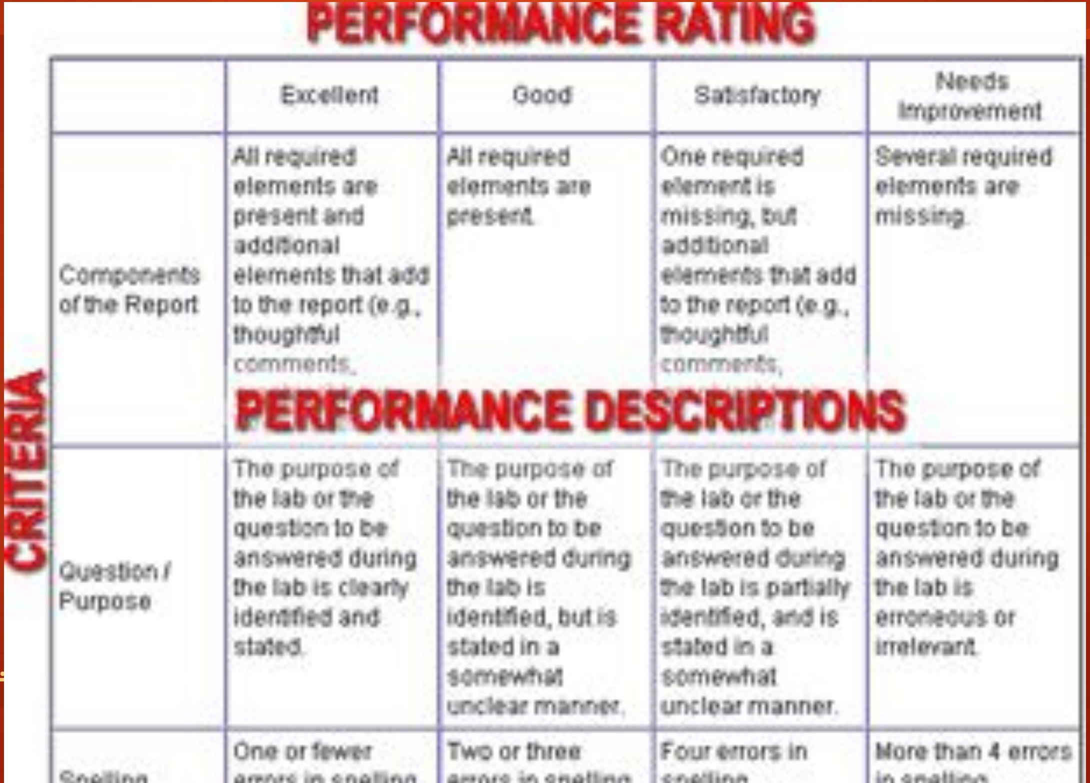
Digital Storytelling Assignment: Rubric Example

Student Name/Date/Course: ______________________________

CATEGORY	Excellent (2 points)	Good (1.8 points)	Fair (1.5 points)	Poor (1 point)
Point of View - Awareness of Audience	Strong awareness of audience in the design. Students can clearly explain why they felt the vocabulary, audio and graphics chosen fit the target audience.	Some awareness of audience in the design. Students can partially explain why they felt the vocabulary, audio and graphics chosen fit the target audience.	Some awareness of audience in the design. Students find it difficult to explain how the vocabulary, audio and graphics chosen fit the target audience.	Limited awareness of the needs and interests of the target audience.
Dramatic Question	Realization is dramatically different from expectation.	Realization differs noticeably from expectation.	Realization barely differs from the expectation.	Realization and expectation do not differ.
Voice - Pacing	The pace (rhythm and voice punctuation) fits the story line and helps the audience really "get into" the story.	Occasionally speaks too fast or too slowly for the story line. The pacing (rhythm and voice punctuation) is relatively engaging for the	Tries to use pacing (rhythm and voice punctuation), but it is often noticeable that the pacing does not fit the story line. Audience is not	No attempt to match the pace of the storytelling to the story line or the audience.

http://v nts/Tut
orials/R

Example 2



PERFORMANCE RATING

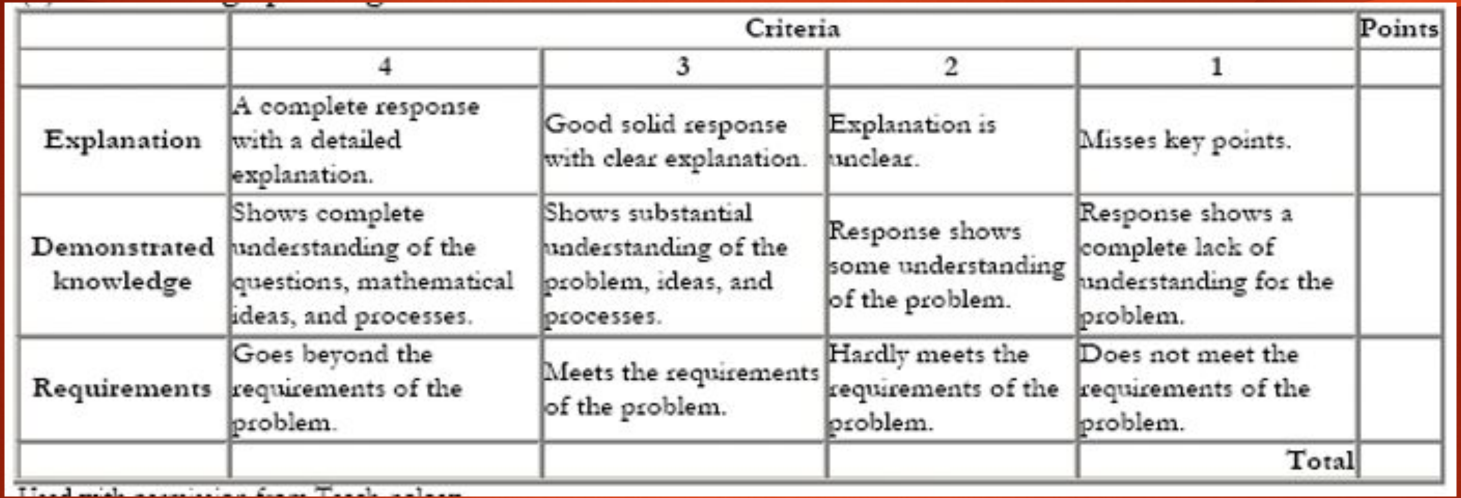
CRITERIA

	Excellent	Good	Satisfactory	Needs Improvement
Components of the Report	All required elements are present and additional elements that add to the report (e.g., thoughtful comments,	All required elements are present.	One required element is missing, but additional elements that add to the report (e.g., thoughtful comments,	Several required elements are missing.
Question / Purpose	The purpose of the lab or the question to be answered during the lab is clearly identified and stated.	The purpose of the lab or the question to be answered during the lab is identified, but is stated in a somewhat unclear manner.	The purpose of the lab or the question to be answered during the lab is partially identified, and is stated in a somewhat unclear manner.	The purpose of the lab or the question to be answered during the lab is erroneous or irrelevant.
Spelling	One or fewer errors in spelling	Two or three errors in spelling	Four errors in spelling	More than 4 errors in spelling

PERFORMANCE DESCRIPTIONS

https://www.uen.
ml

Example 3



	Criteria				Points
	4	3	2	1	
Explanation	A complete response with a detailed explanation.	Good solid response with clear explanation.	Explanation is unclear.	Misses key points.	
Demonstrated knowledge	Shows complete understanding of the questions, mathematical ideas, and processes.	Shows substantial understanding of the problem, ideas, and processes.	Response shows some understanding of the problem.	Response shows a complete lack of understanding for the problem.	
Requirements	Goes beyond the requirements of the problem.	Meets the requirements of the problem.	Hardly meets the requirements of the problem.	Does not meet the requirements of the problem.	
				Total	

Used with permission from Teach-nology

https://www.cbd.int/ibd/2008/Resources/teachers/appendix3.shtml

Why a Rubric for Decentralization of DID Methods?

- "Decentralized" is a quagmire
- Requirements for DID Methods led to passionate, intense debate:
  - The DID community came together with several subtly different meanings of decentralization.
- How can we evaluate DID Methods against the criteria driving this work?

Intentions

- A tool for evaluating DID Methods
- Objective & non-judgmental
  - Minimize bias. Avoid advocacy. Champion characterization.
- Evaluation is in the eye of the beholder
  - Weighting / Selection of criteria based on use case under evaluation
  - Evaluations / Responses up to evaluator
- No summary rating. No universal metric.

A Rubric (structure)

- Set of criteria for evaluation
- Each criteria
  - Question
  - Possible Answers
  - Description of Possible Answers
  - Relevance
  - Examples
- An evaluation is a set of criteria with answers for a specific DID Method, and optional notes explaining each answer.

Illustration – Amusement Park Rides

- Question
  - How tall is the rider?
- Possible Responses
  - Under 3’
  - Between 3’ and 4’
  - Over 4’
- Relevance
  - Height is often a useful indicator for ride safety. For some rides you need to be tall enough for safety devices to work. For other rides, being short is a good proxy for rideability.
- It is up to the ride operator to decide if too tall or too short is an appropriate filter

History

- Passionate Debate in CCG
- Several Sessions at IIW28 Spring 2019
- Initial Draft from IIW Notes
  - https://docs.google.com/document/d/1HXik6hxHfGZR1-nhmQoYO5Ap3eGPNpg8MitCQXdW7Q0/edit?usp=sharing
- RWOT9 (Rebooting the Web of Trust IX)
  - Creative Brief
    - https://github.com/WebOfTrustInfo/rwot9-prague/blob/master/topics-and-advance-readings/rubrics.md
  - Draft Document
    - https://github.com/WebOfTrustInfo/rwot9-prague/blob/master/draft-documents/decentralized-did-rubric.md

Proposed Next Steps

- Finish the RWOT Paper (60 days)
- Propose Initial Draft for DID WG
- Solicit Criteria
- Collect, Collate, Filter
- Solicit Comments & Added Relevance

Adopting Work - Concerns and Requirements (60 min)

What are the concerns?

- Intellectual Property Commitments
- Ability for Working Group to revisit previous decisions
- Continuity - Open issues, PR history, Closed issues, Commit history (and committers)
- Messaging to broader Community
- Maximize Utilization of W3C Infrastructure
- Effort - Editors, W3C Staff

What are the **concrete** requirements?

- Open issues must be available in WG repo
- Open pull requests must be available in WG repo
- There must be a clearly identified point in time at which the Working Group took over.
- PRs from non-WG members must be closed.
- Unclear group opinion on statement “Complete commit history must be available in the WG repo.”
- Closed issues, PRs, must be available in the WG repo.

Lunch

DID Resolution (Markus, 30 min)

DID Resolution

Markus Sabadello
Danube Tech, Decentralized Identity Foundation,
Sovrin Foundation, W3C DID WG + CCG, OASIS XDI TC

https://danubetech.com/
TPAC, Fukuoka, 17th September 2019

DID Resolution

- DID Resolution: DID → DID Document
  - Set of public keys
  - Set of service endpoints
  - Authentication mechanisms
  - Timestamps, proofs
  - Other metadata

- Given a DID, obtain the metadata that is needed for trustable interaction with the DID subject.
- Details are defined by the applicable DID method's "Read" operation.

Example DID Document:

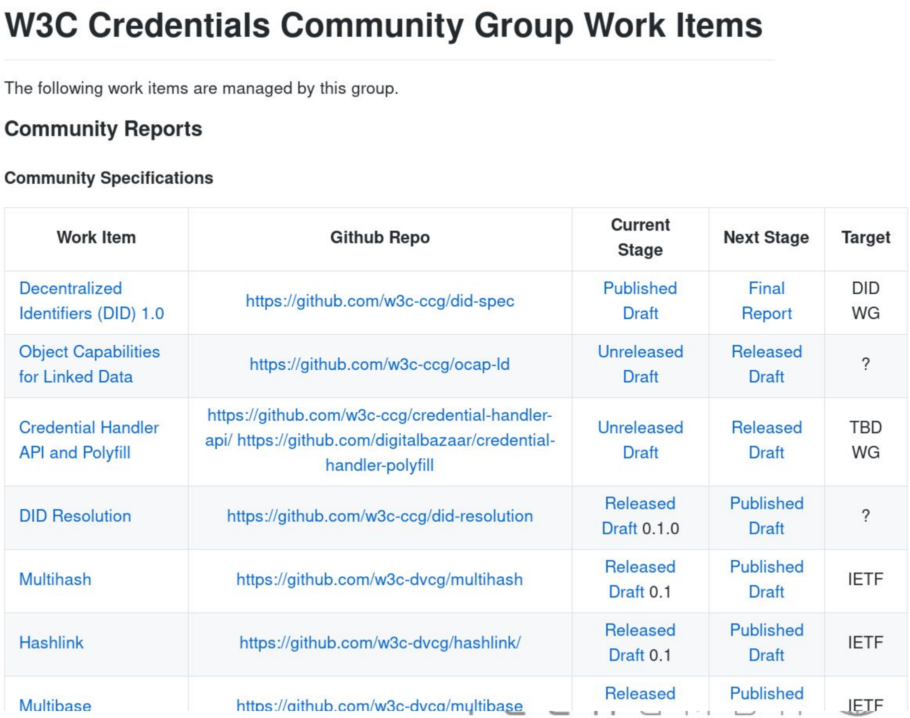
```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:example:456",
  "publicKey": [
    {
      "id": "did:example:456#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjJCwDmqPV"
    }
  ],
  "service": {
    "type": "hub",
    "serviceEndpoint":
        "https://cloud.service.com/hub/did:example:456"
  },
  "authentication": {
    "did:example:456#key-1"
  }
}
```

DID Resolution

- Work Item of the Credentials Community Group
- Currently iterating on v0.2
- Weekly calls, recordings, logs
- Closely related to DID specification

- Out-of-scope for DID Working Group

https://w3c-ccg.github.io/did-resolution/



W3C Credentials Community Group Work Items

The following work items are managed by this group.

Community Reports

Community Specifications

Work Item	Github Repo	Current Stage	Next Stage	Target
Decentralized Identifiers (DID) 1.0	https://github.com/w3c-ccg/did-spec	Published Draft	Final Report	DID WG
Object Capabilities for Linked Data	https://github.com/w3c-ccg/ocap-ld	Unreleased Draft	Released Draft	?
Credential Handler API and Polyfill	https://github.com/w3c-ccg/credential-handler-api/ https://github.com/digitalbazaar/credential-handler-polyfill	Unreleased Draft	Released Draft	TBD WG
DID Resolution	https://github.com/w3c-ccg/did-resolution	Released Draft 0.1.0	Published Draft	?
Multihash	https://github.com/w3c-dvcg/multihash	Released Draft 0.1	Published Draft	IETF
Hashlink	https://github.com/w3c-dvcg/hashlink/	Released Draft 0.1	Published Draft	IETF
Multibase	https://github.com/w3c-dvcg/multibase	Released	Published	IETF

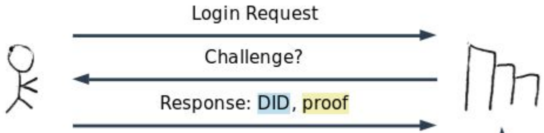
When does DID Resolution happen?

① Verifiable Credentials

```
{
  "issuer": "did:example:456",
  "credentialSubject": {
    "id": "did:example:123",
    "degree": "M.Sc."
  },
  "proof": {
    "jws": "eyJhbGciOiJSUzI1N...",
    ...
  }
}
```

Verifier resolves the issuer's DID,
in order to discover the public key
needed to verify the proof.

② DID Auth



Login Request

Challenge?

Response: DID, proof

Relying party resolves the user's DID,
in order to discover the public key
needed to verify the proof.

③ Service Discovery

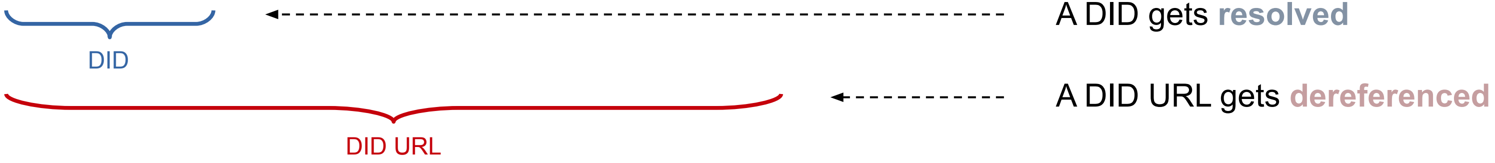
Application resolves a DID,
in order to discover a service for
Interacting via a secure channel.

DANUBE
TECH GMBH

DIDs and DID URLs

```
did                = "did:" method-name ":" method-specific-id
method-name        = 1*method-char
method-char        = %x61-7A / DIGIT
method-specific-id = *idchar *( ":" *idchar )
idchar             = ALPHA / DIGIT / "." / "-" / "_"
did-url            = did *( ";" param ) path-abempty [ "?" query ] [ "#" fragment ]
param              = param-name [ "=" param-value ]
param-name         = 1*param-char
param-value        = *param-char
param-char         = ALPHA / DIGIT / "." / "-" / "_" / ":" / pct-encoded
```

Example: `did:xyz:1234;service=agent/profile?query#frag`

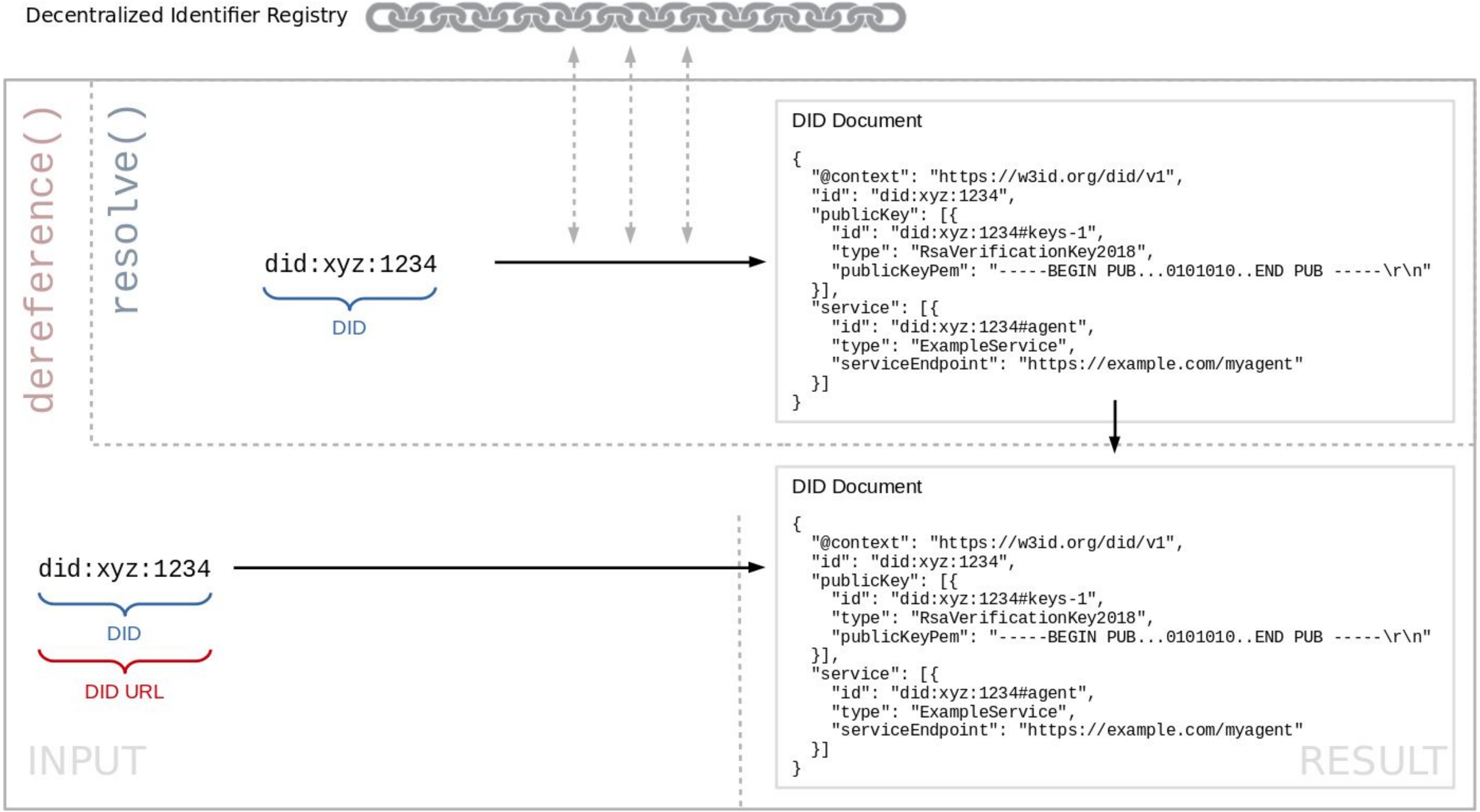


DID

A DID gets resolved

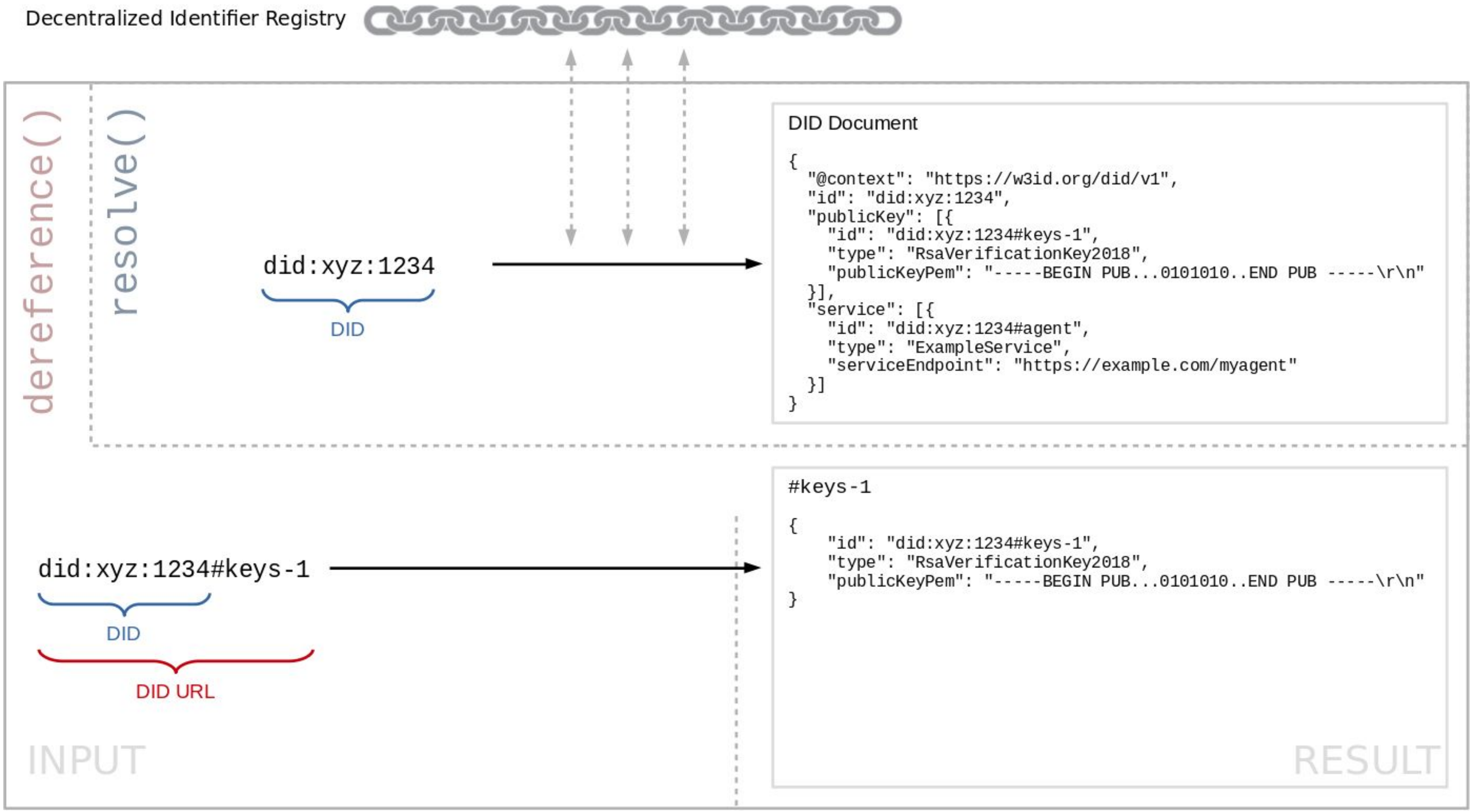
DID URL

A DID URL gets dereferenced



```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

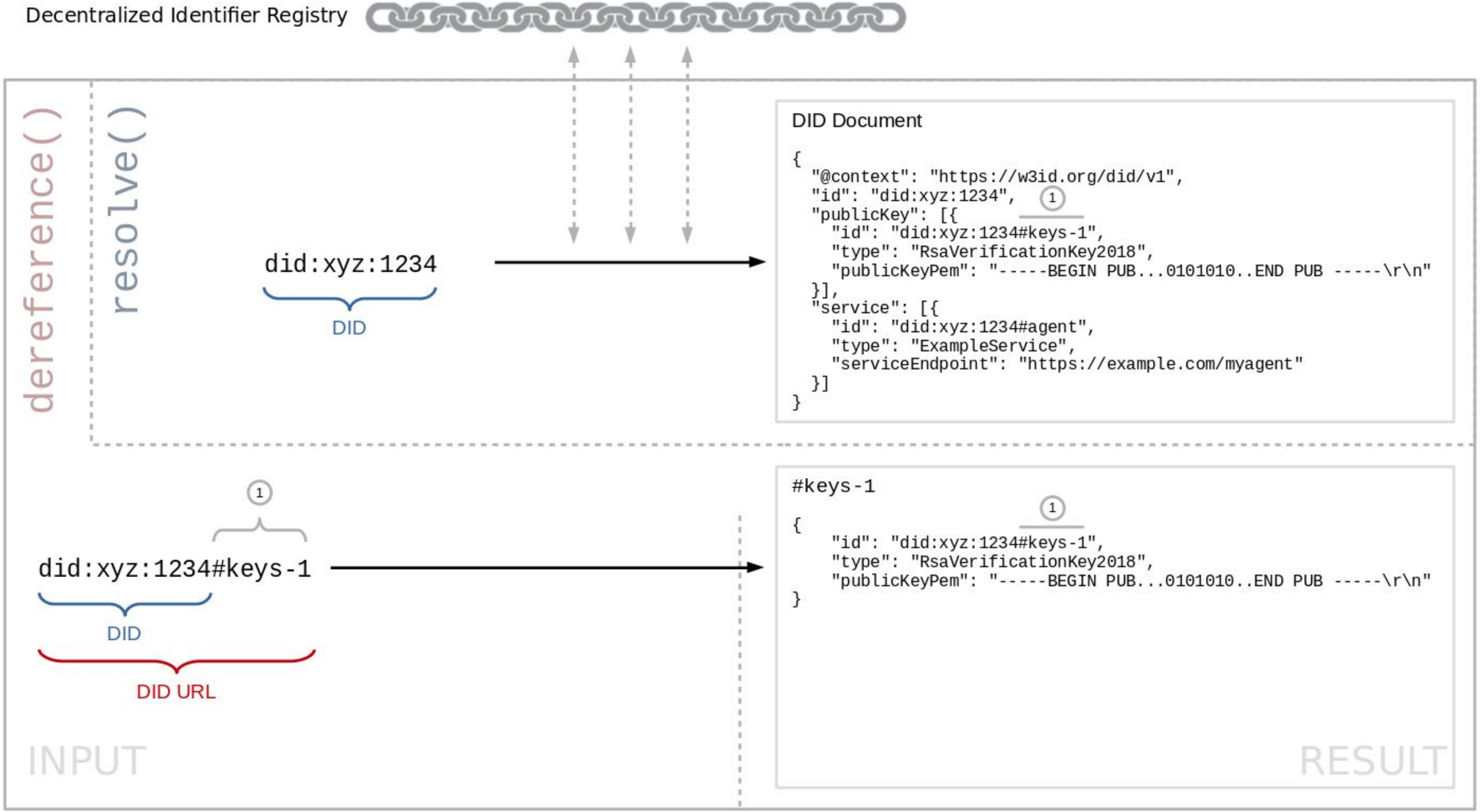


DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

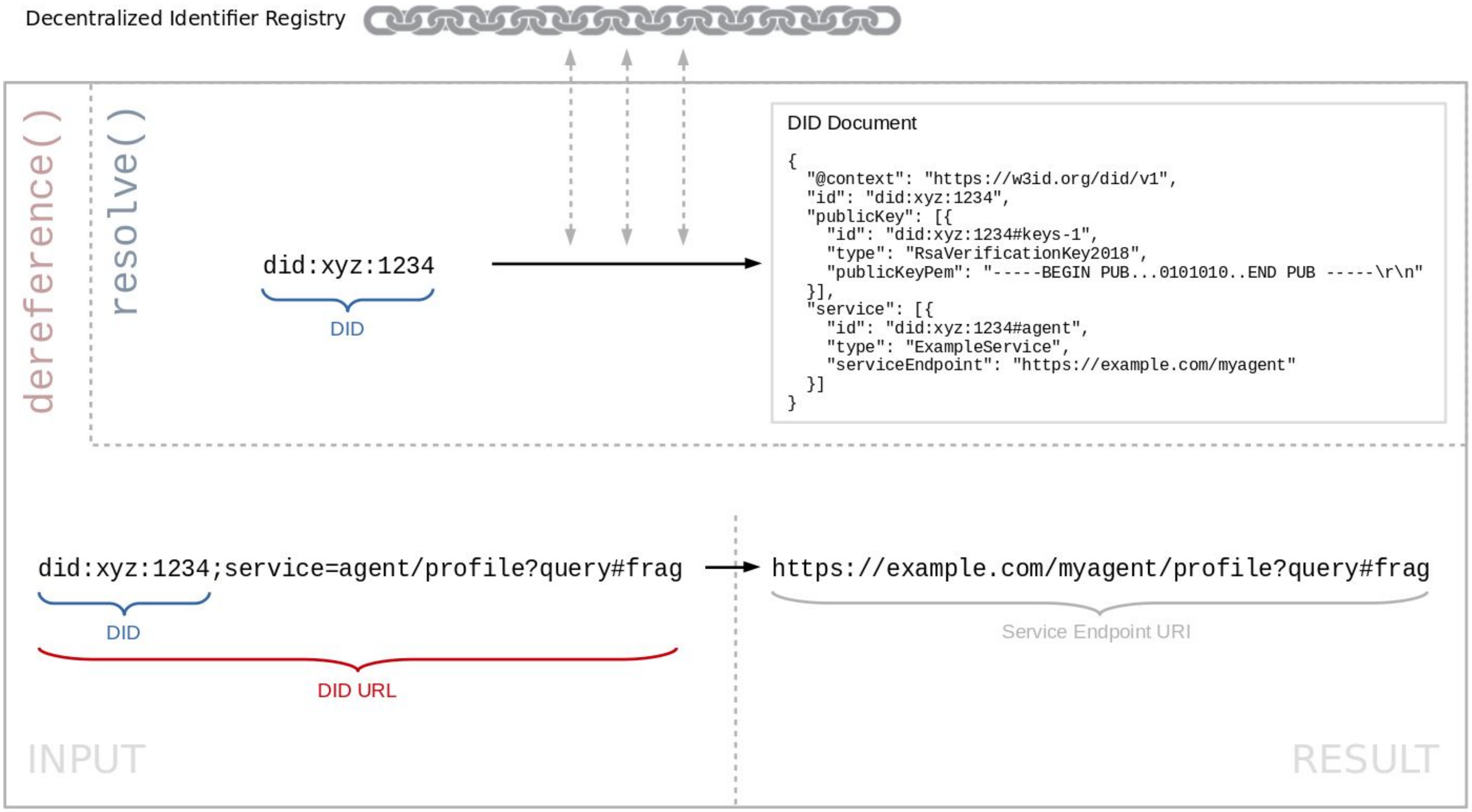
#keys-1

```
{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
}
```

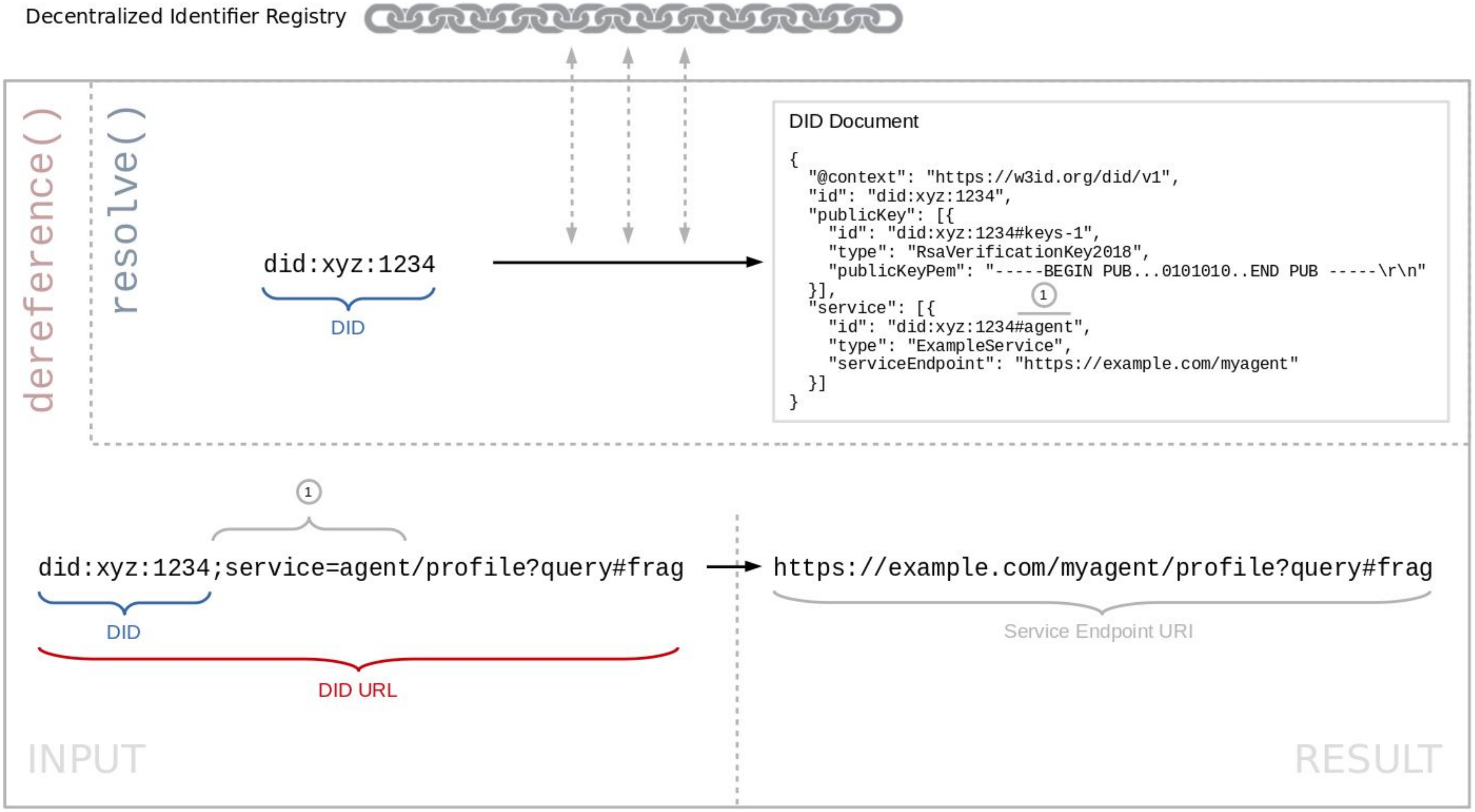


```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

```
{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
}
```

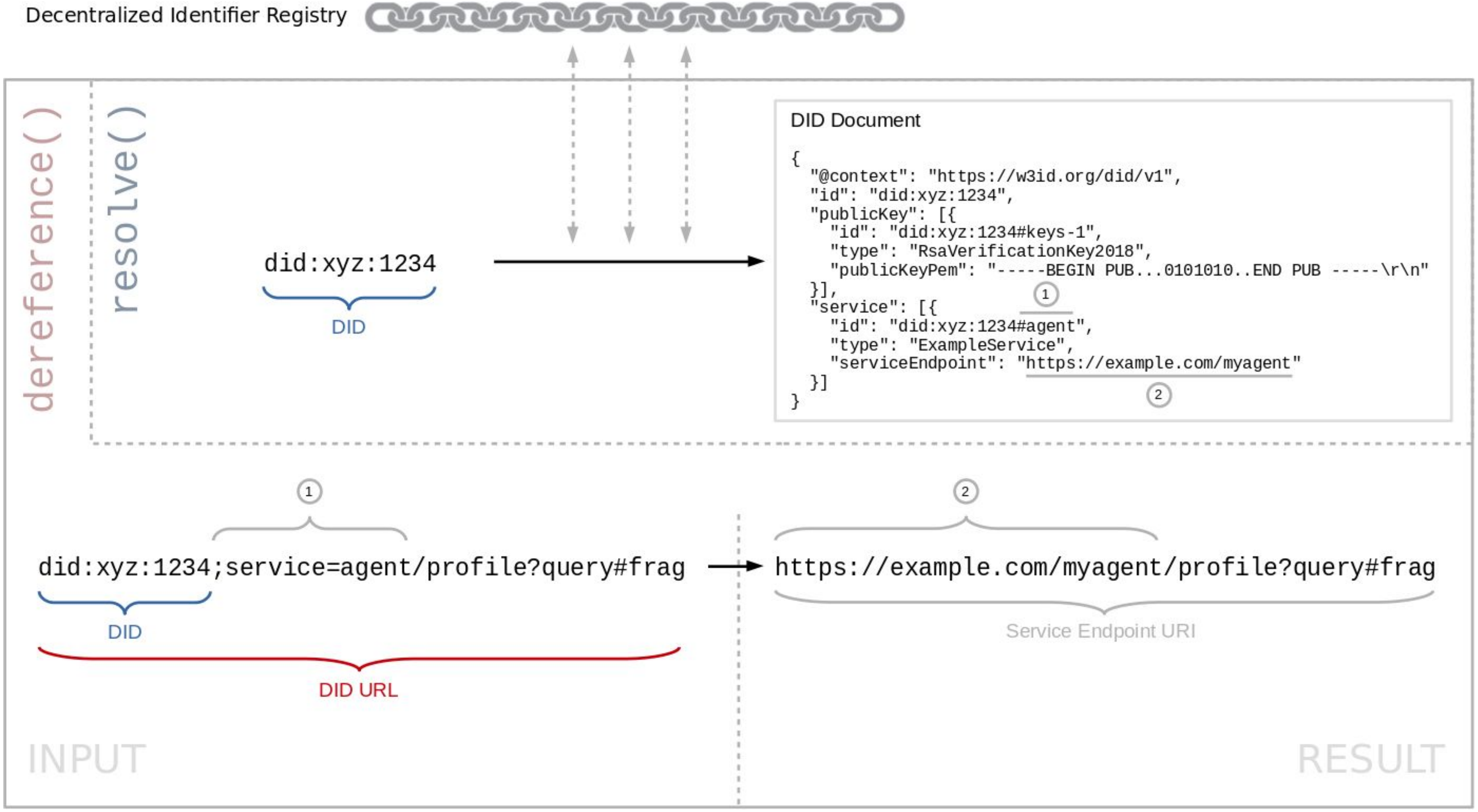

Decentralized Identifier Registry
dereference()
resolve()
did:xyz:1234
DID
DID Document
did:xyz:1234;service=agent/profile?query#frag
DID
DID URL
https://example.com/myagent/profile?query#frag
Service Endpoint URI
INPUT
RESULT

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

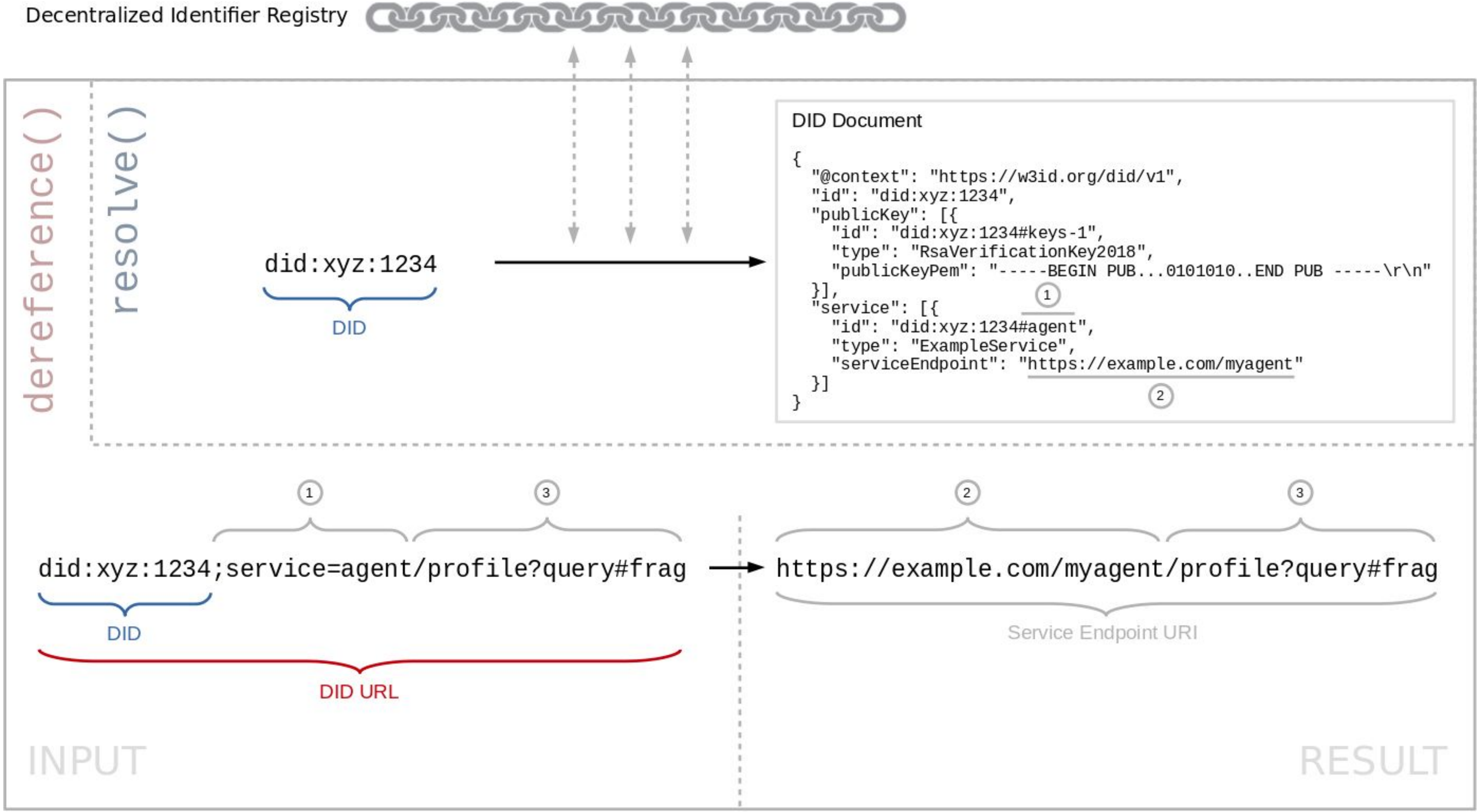


Decentralized Identifier Registry
dereference()
resolve()
did:xyz:1234
DID
DID Document
1
1
did:xyz:1234;service=agent/profile?query#frag
DID
DID URL
https://example.com/myagent/profile?query#frag
Service Endpoint URI
INPUT
RESULT

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

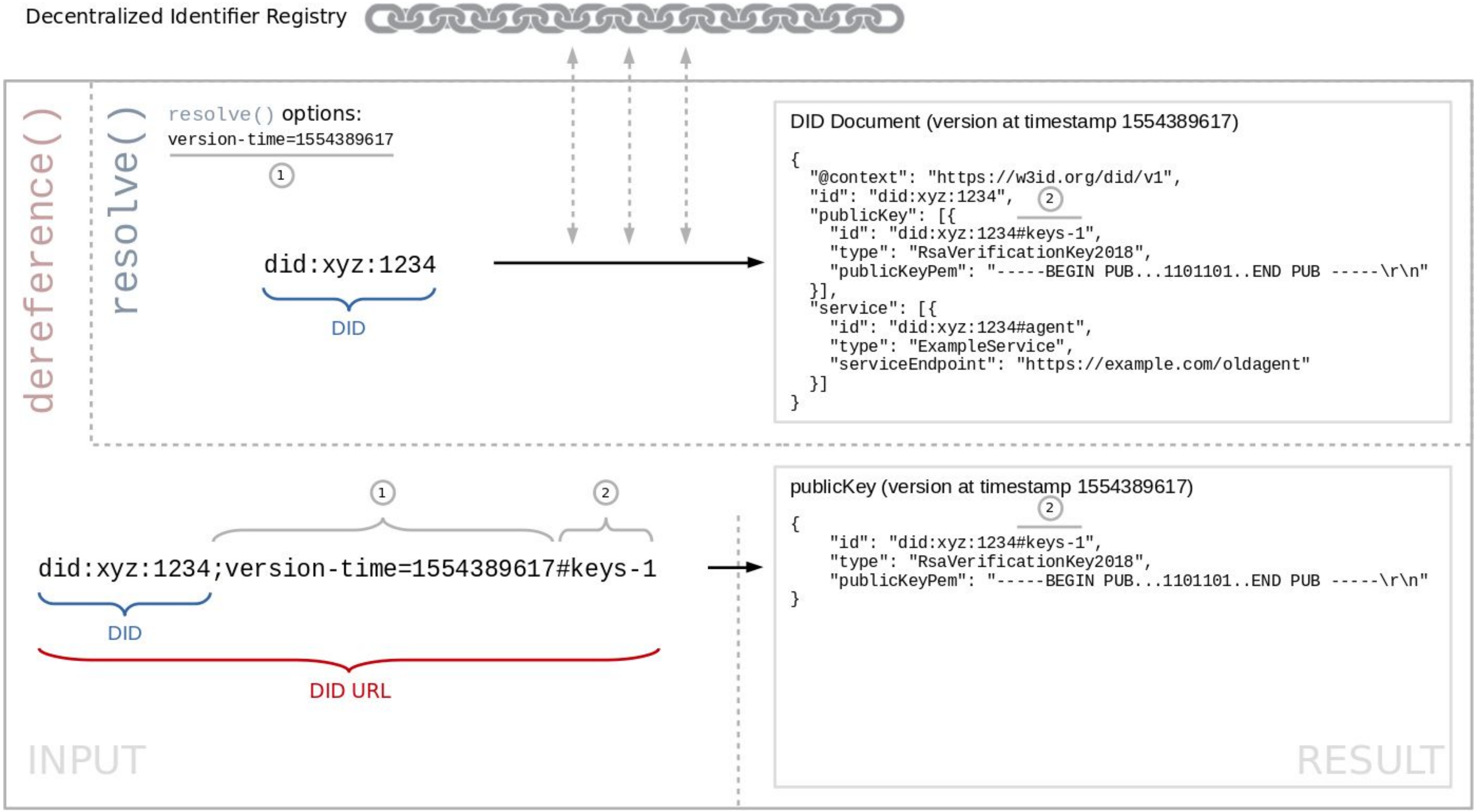

Decentralized Identifier Registry
dereference()
resolve()
did:xyz:1234
DID
DID Document
1
2
1
2
did:xyz:1234;service=agent/profile?query#frag
DID
DID URL
https://example.com/myagent/profile?query#frag
Service Endpoint URI
INPUT
RESULT

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```



Decentralized Identifier Registry
dereference()
resolve()
did:xyz:1234
DID
DID Document
1
2
1
3
2
3
did:xyz:1234;service=agent/profile?query#frag
https://example.com/myagent/profile?query#frag
DID
DID URL
Service Endpoint URI
INPUT
RESULT

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...0101010..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/myagent"
  }]
}
```

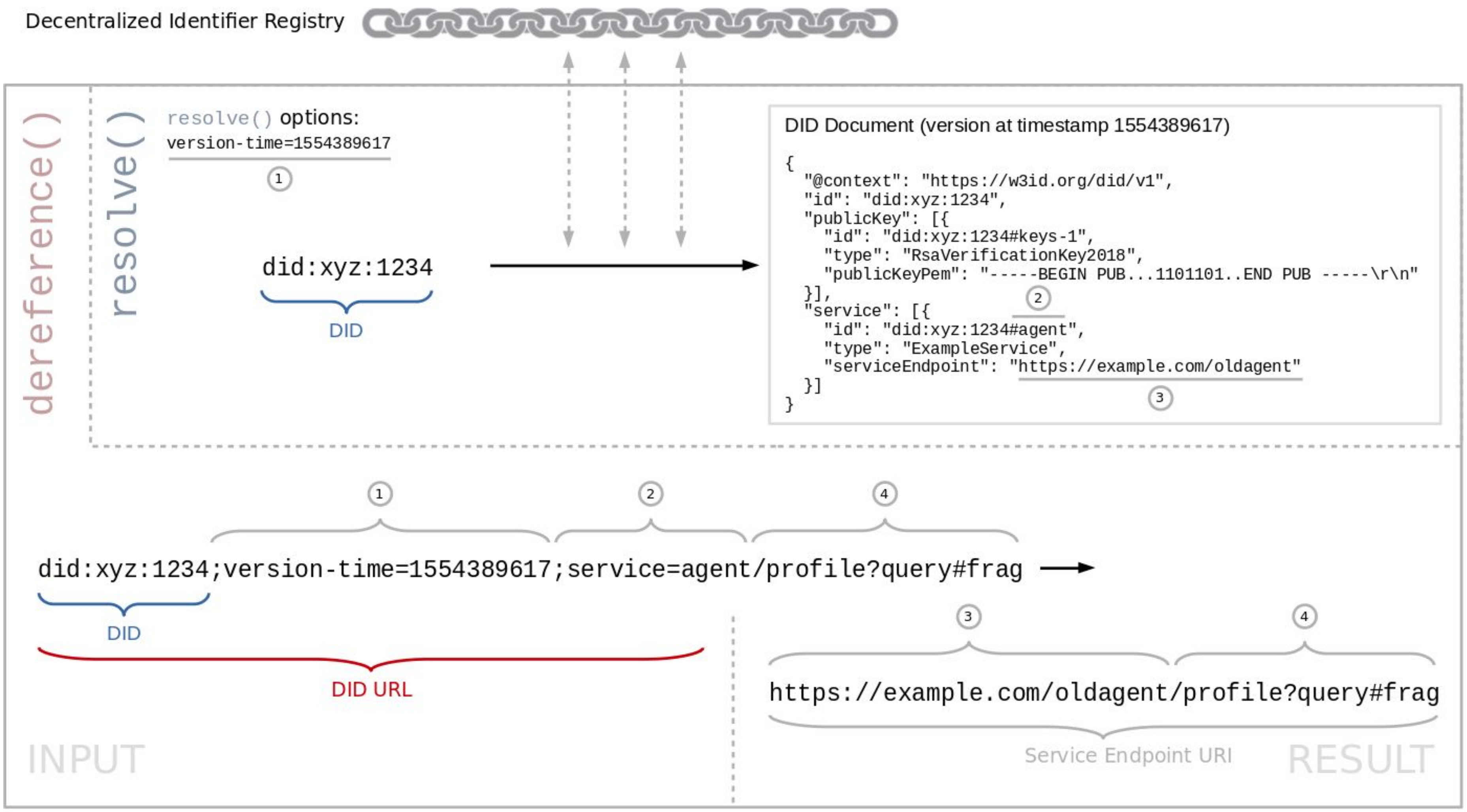


DID Document (version at timestamp 1554389617)

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...1101101..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/oldagent"
  }]
}
```

publicKey (version at timestamp 1554389617)

```
{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...1101101..END PUB -----\r\n"
}
```


Decentralized Identifier Registry
dereference()
resolve()
resolve() options:
version-time=1554389617
1
did:xyz:1234
DID
DID Document (version at timestamp 1554389617)
2
3
1
2
4
did:xyz:1234;version-time=1554389617;service=agent/profile?query#frag
DID
DID URL
3
4
https://example.com/oldagent/profile?query#frag
Service Endpoint URI
INPUT
RESULT

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:xyz:1234",
  "publicKey": [{
    "id": "did:xyz:1234#keys-1",
    "type": "RsaVerificationKey2018",
    "publicKeyPem": "-----BEGIN PUB...1101101..END PUB -----\r\n"
  }],
  "service": [{
    "id": "did:xyz:1234#agent",
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/oldagent"
  }]
}
```

DID URL Matrix Parameters

service - Identifies a service from the DID Document by service ID.

service-type - Identifies services from the DID Document by service type.

key - Identifies a key from the DID Document by key ID.

key-type - Identifies keys from the DID Document by key type.

version-id - Identifies an earlier version of the DID Document.

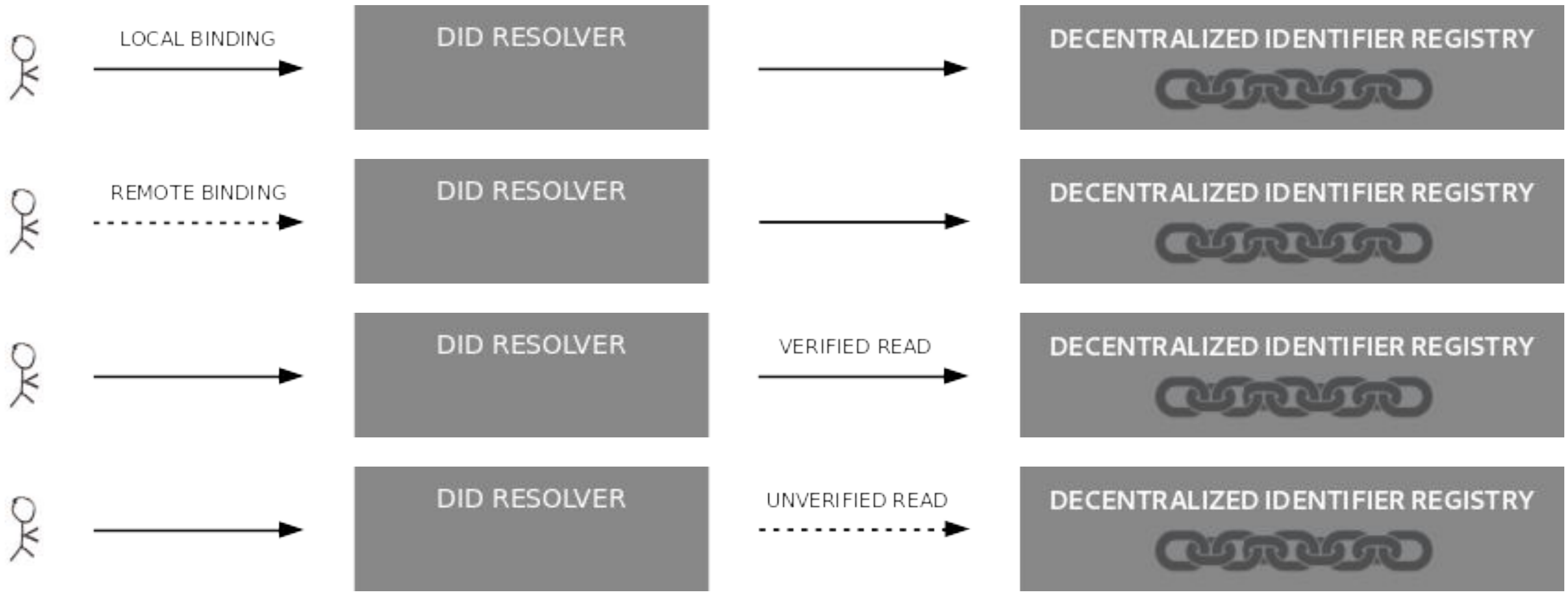
version-time - Identifies an earlier version of the DID Document.

content-id - Identifies content other than the DID Document.

content-type - Identifies content other than the DID Document.

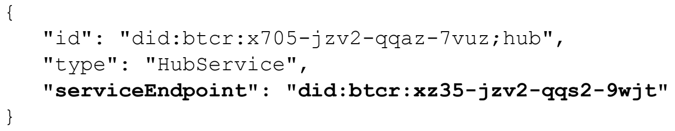
hl - Adds integrity protection to the DID Document.

DID Resolver Architectures


LOCAL BINDING
DID RESOLVER
DECENTRALIZED IDENTIFIER REGISTRY
REMOTE BINDING
DID RESOLVER
DECENTRALIZED IDENTIFIER REGISTRY
DID RESOLVER
VERIFIED READ
DECENTRALIZED IDENTIFIER REGISTRY
DID RESOLVER
UNVERIFIED READ
DECENTRALIZED IDENTIFIER REGISTRY

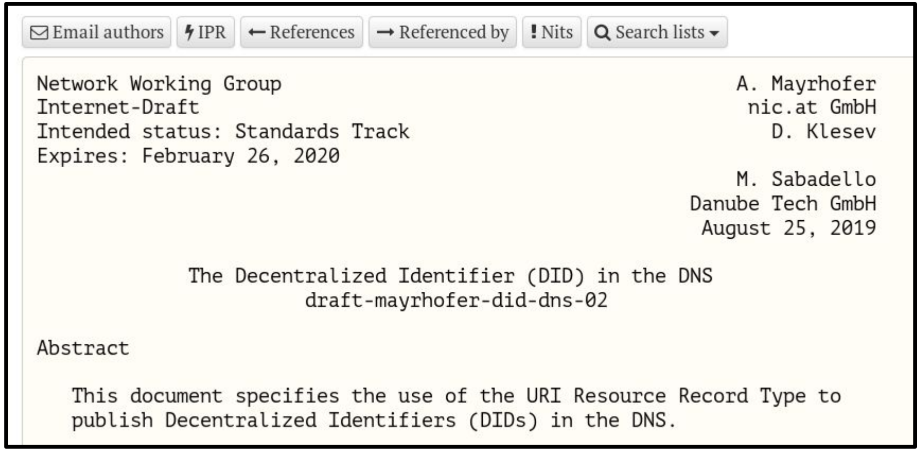
Other DID Resolution Topics

- Versioning:
  - Input parameter to request specific version of DID Document, e.g. by version number, or by timestamp.
  - DID Document can contain version number or timestamp of last update.
- Caching:
  - Input parameter to request specific caching behavior, e.g. force fresh DID Resolution.
  - Controlled by DID Resolver configuration, input options, and DID Document content ("time-to-live").
- Deactivation:
  - DID Resolver can return an error, or a DID Document with a "deactivated" flag.
- Validation:
  - DID Resolver validates DID Documents before returning them.
- Redirects:
  - DID can be used as the value of `serviceEndpoint`.



```
{
   "id": "did:btcr:x705-jzv2-qqaz-7vuz;hub",
   "type": "HubService",
   "serviceEndpoint": "did:btcr:xz35-jzv2-qqs2-9wjt"
}
```

DNS → DID



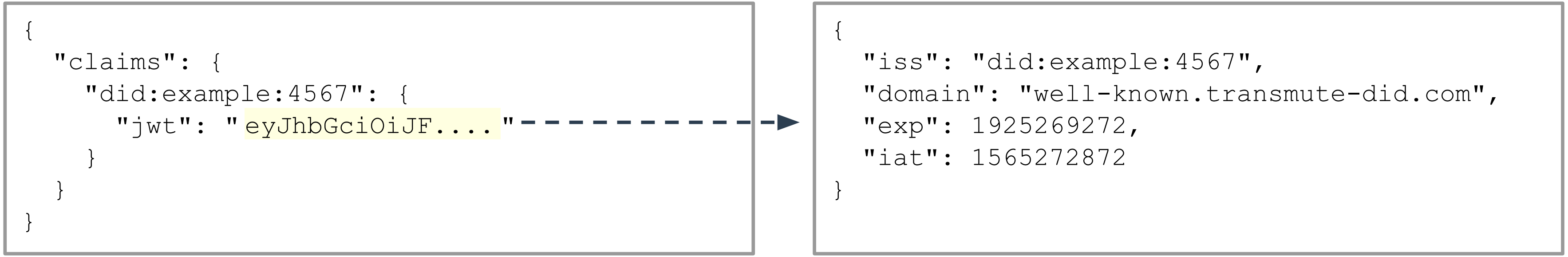
```
Network Working Group                                       A. Mayrhofer
Internet-Draft                                              nic.at GmbH
Intended status: Standards Track                              D. Klesev
Expires: February 26, 2020
                                                           M. Sabadello
                                                       Danube Tech GmbH
                                                        August 25, 2019

          The Decentralized Identifier (DID) in the DNS
                    draft-mayrhofer-did-dns-02

Abstract

   This document specifies the use of the URI Resource Record Type to
   publish Decentralized Identifiers (DIDs) in the DNS.
```

```
~> dig _did.ssi.labs.nic.at uri
;; Warning: Client COOKIE mismatch

; <<>> DiG 9.11.5-P4-5-Debian <<>> _did.ssi.labs.nic.at uri
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50630
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1024
; COOKIE: d79d92fd59041556a1e33daf5d00fdb1774deb5d01034bd3 (bad)
;; QUESTION SECTION:
;_did.ssi.labs.nic.at.           IN     URI

;; ANSWER SECTION:
_did.ssi.labs.nic.at.      292   IN     URI    10 1 "did:sov:stn:r1dwAJxcoG7EPiioGMz7h"

;; Query time: 1 msec
;; SERVER: 192.168.8.1#53(192.168.8.1)
;; WHEN: Wed Jun 12 15:27:21 CEST 2019
;; MSG SIZE  rcvd: 126
```

HTTPS URL → DID

`https://www.mywebsite.com/`**`.well-known/did-configuration`**

- Contains JWTs that claim a list of domain names
- Signed by the DID's private key

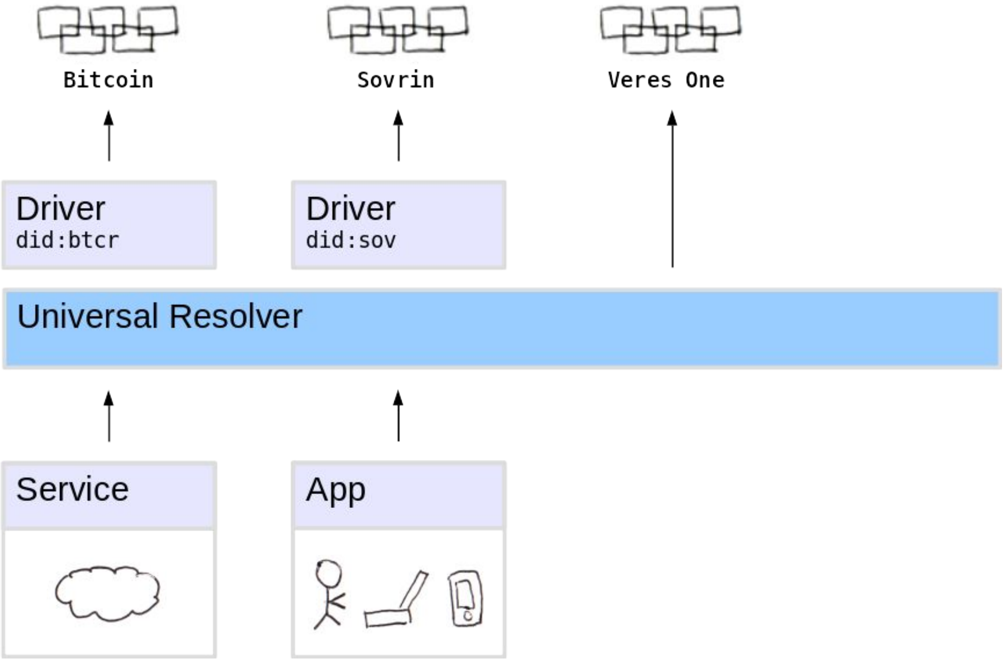


```
{
  "claims": {
    "did:example:4567": {
      "jwt": "eyJhbGciOiJF...."
    }
  }
}
```

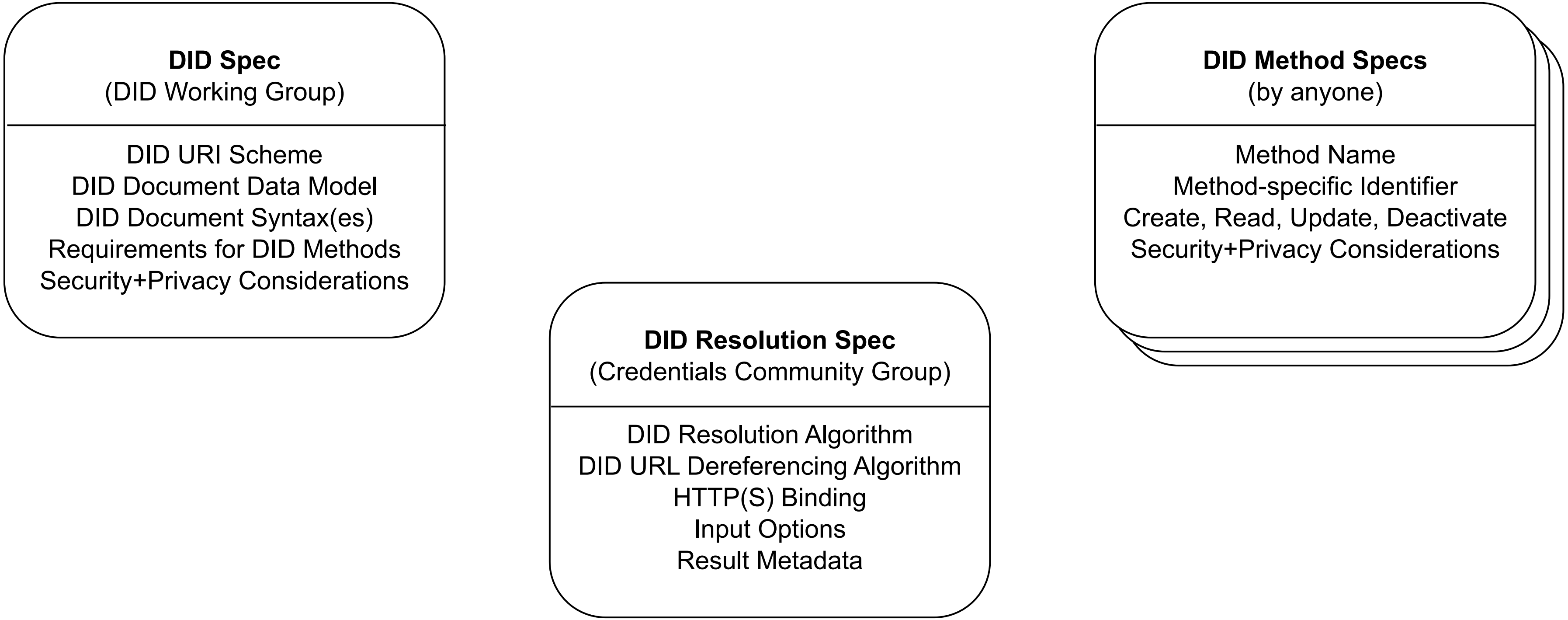
```
{
  "iss": "did:example:4567",
  "domain": "well-known.transmute-did.com",
  "exp": 1925269272,
  "iat": 1565272872
}
```

DID Universal Resolver

- Looks up ("resolves") DID to its DID Document.
- Provides a universal API that works with all DID methods.
- Uses a set of configurable "drivers" that know how to connect to the DID registry.
- **https://uniresolver.io/**



What will happen where?



DID Spec
(DID Working Group)

DID URI Scheme
DID Document Data Model
DID Document Syntax(es)
Requirements for DID Methods
Security+Privacy Considerations

DID Resolution Spec
(Credentials Community Group)

DID Resolution Algorithm
DID URL Dereferencing Algorithm
HTTP(S) Binding
Input Options
Result Metadata

DID Method Specs
(by anyone)

Method Name
Method-specific Identifier
Create, Read, Update, Deactivate
Security+Privacy Considerations

(WoT Joint session, 30 min)

Working through issues (30 min)

Afternoon break

(Open Topics, 30 min)

DID Controller? (Proposed by Joe)

In the spec:

5.4 Authentication

Authentication is the mechanism by which the controller(s) of a DID can cryptographically prove that they are associated with that DID. See Section § 9.3 Binding of Identity . Note that Authentication is separate from Authorization because the **controllers may wish to enable others to update their DID Document** (for example, to assist with key recovery as discussed in Section § 9.8 Key Revocation and Recovery) without enabling them to prove control (and t**hus be able to impersonate the controllers**).

This has triggered a conversation on the CCG email list as the usage here is directly inverted from the meaning of controller and authentication as understood by some of this community. (Hat tip to Sethi Shivam and Daniel Hardman.)

PING (90 min)

Things we can talk about

- We have resolved to select the CCG's Decentralized Identifiers (DIDs) v0.13 - Data Model and Syntaxes as the DID WG's first editor's draft
- We care deeply about privacy
- What is the best way to establish regular review of our work?
  - Is there a cadence that works best for PING without undue burden on them?
- Security and privacy review self-questionnaire
- What scenarios/questions have we already looked at/addressed and how?
-

(Open topics, 60 min)