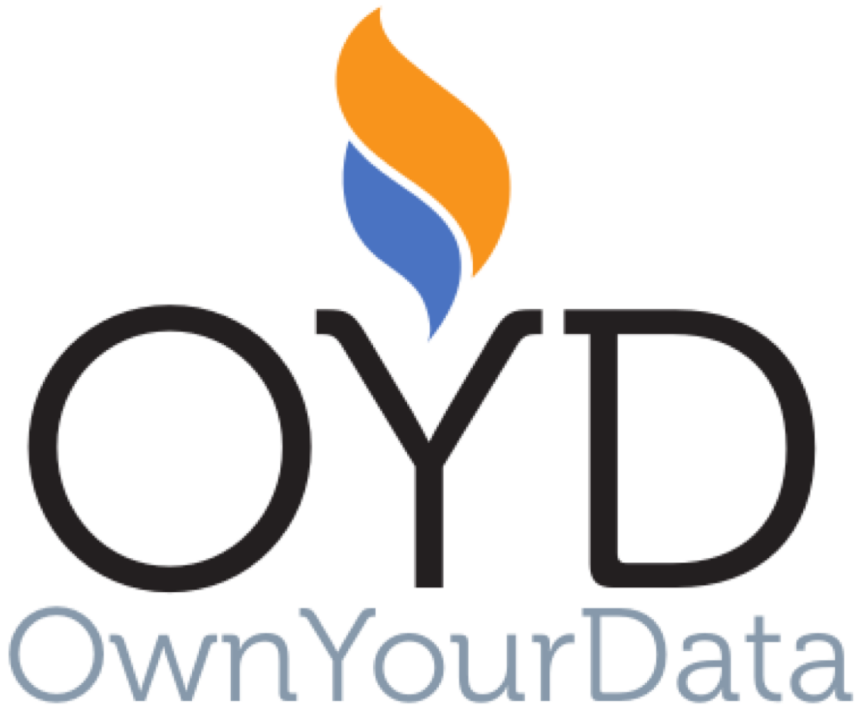

OYD
OwnYourData

ABOUT ME

Job

- Freelance Data Scientist – since 2016
- Senior Lead Scientist at Frequentis AG – since 2006
- Consultant at AI Informatics and Siemens - 2004-2006

Education

- TU Wien: PhD in technical Mathematics
  study abroad at the Innovative Computing Laboratory, UTK
- Danube University: MBA
  study abroad at the Weatherhead School of Management

Private

- Founder & chairman of non-profit organization OwnYourData.eu
- former MyData Global board member (Treasurer)
- married*, 3 kids*

OYD
OwnYourData

SELF SOVEREIGN IDENTITY

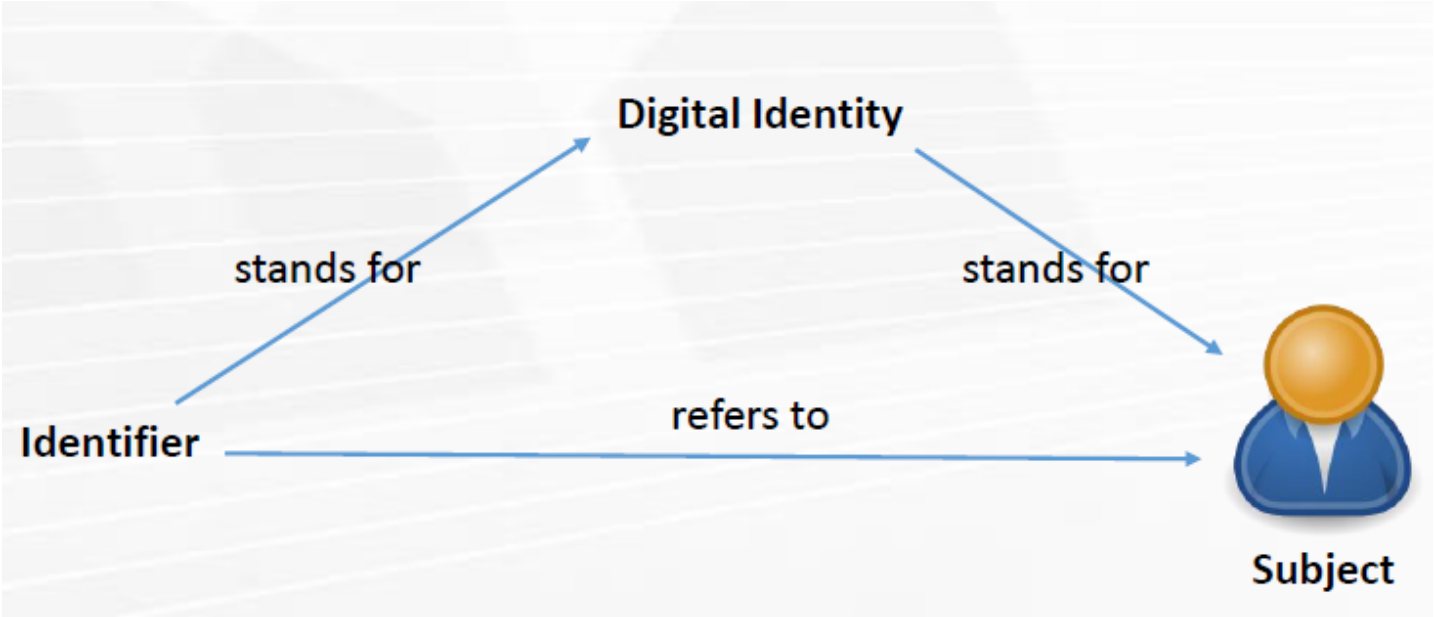
"On the Internet, nobody knows you're a dog."

Fixing the
missing identity layer
on the internet

OYD
OwnYourData

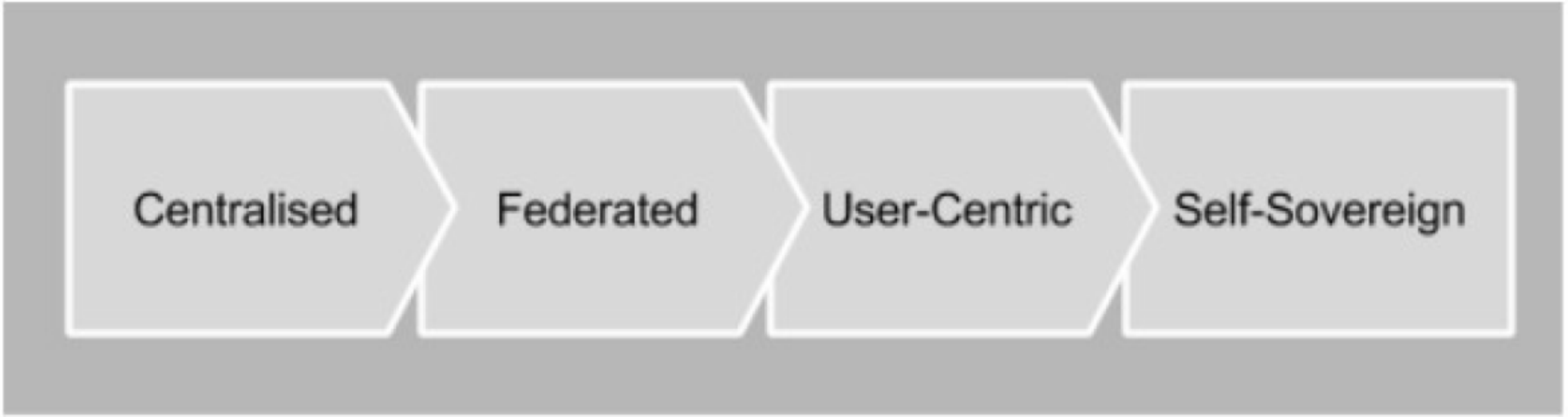
DIGITAL IDENTITY

- the digital represenation of the information known about a specific individual or organization
  - Identifier: e.g., email address
  - Attributes: e.g., name, birth date
  - Credentials: e.g., certificate, password



OYD
OwnYourData

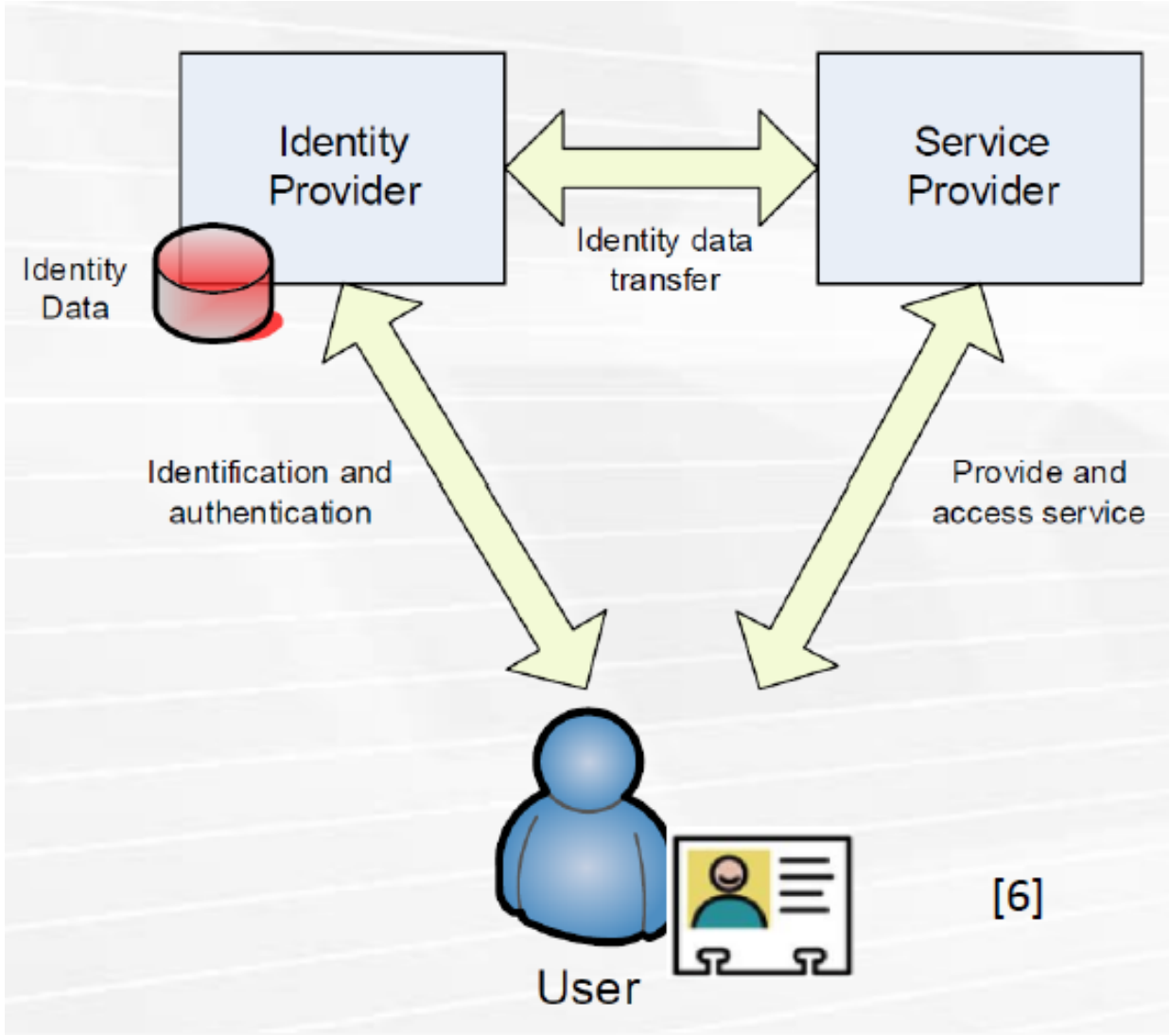
THE EVOLUTION OF ONLINE IDENTITY



OYD
OwnYourData

CENTRALISED IDENTITY MANAGEMENT

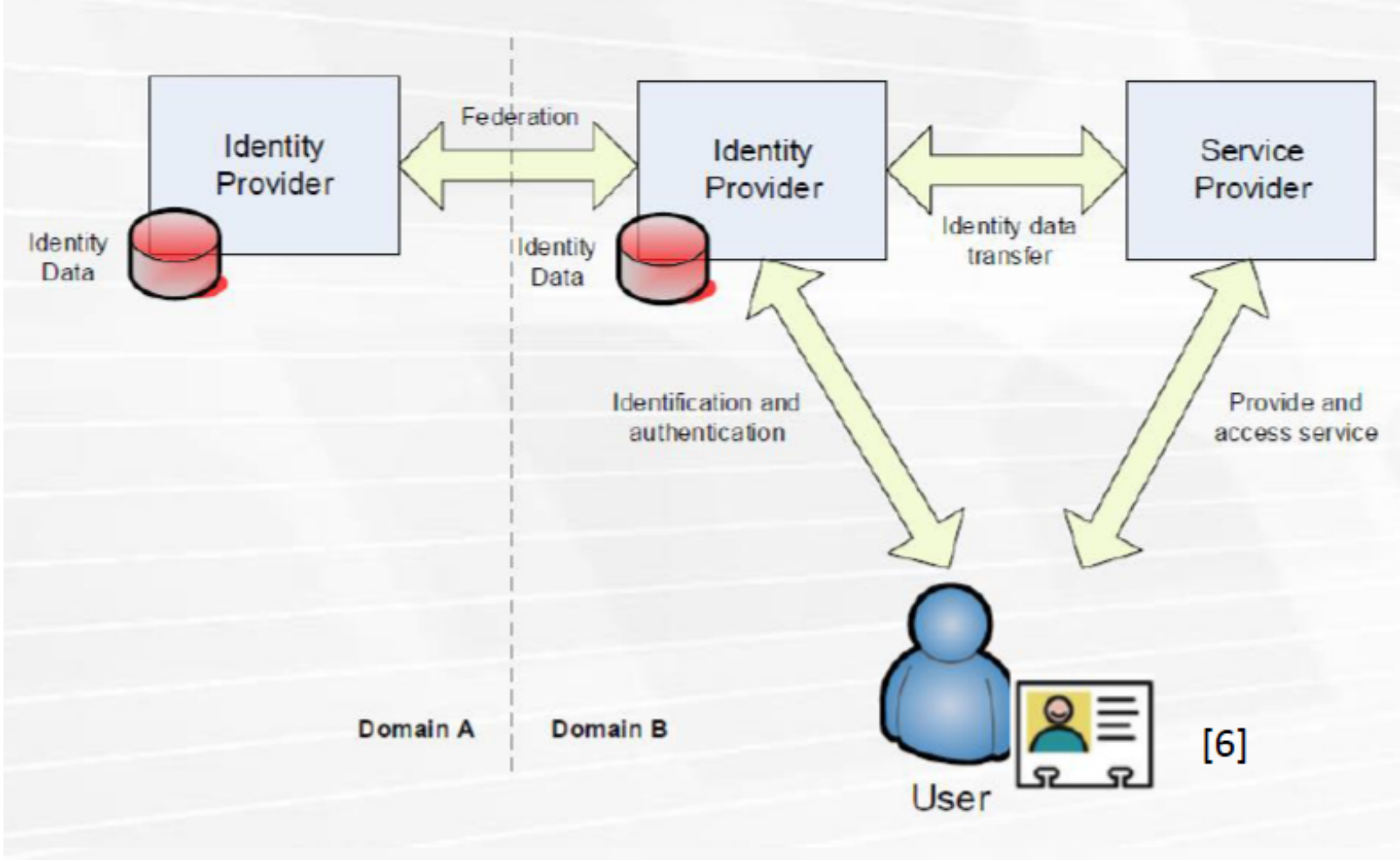
- Identity data stored at Identity Provider
- Service Provider receives identity data from Identity Provider
- User has no control over the actual data transfer



OYD
OwnYourData

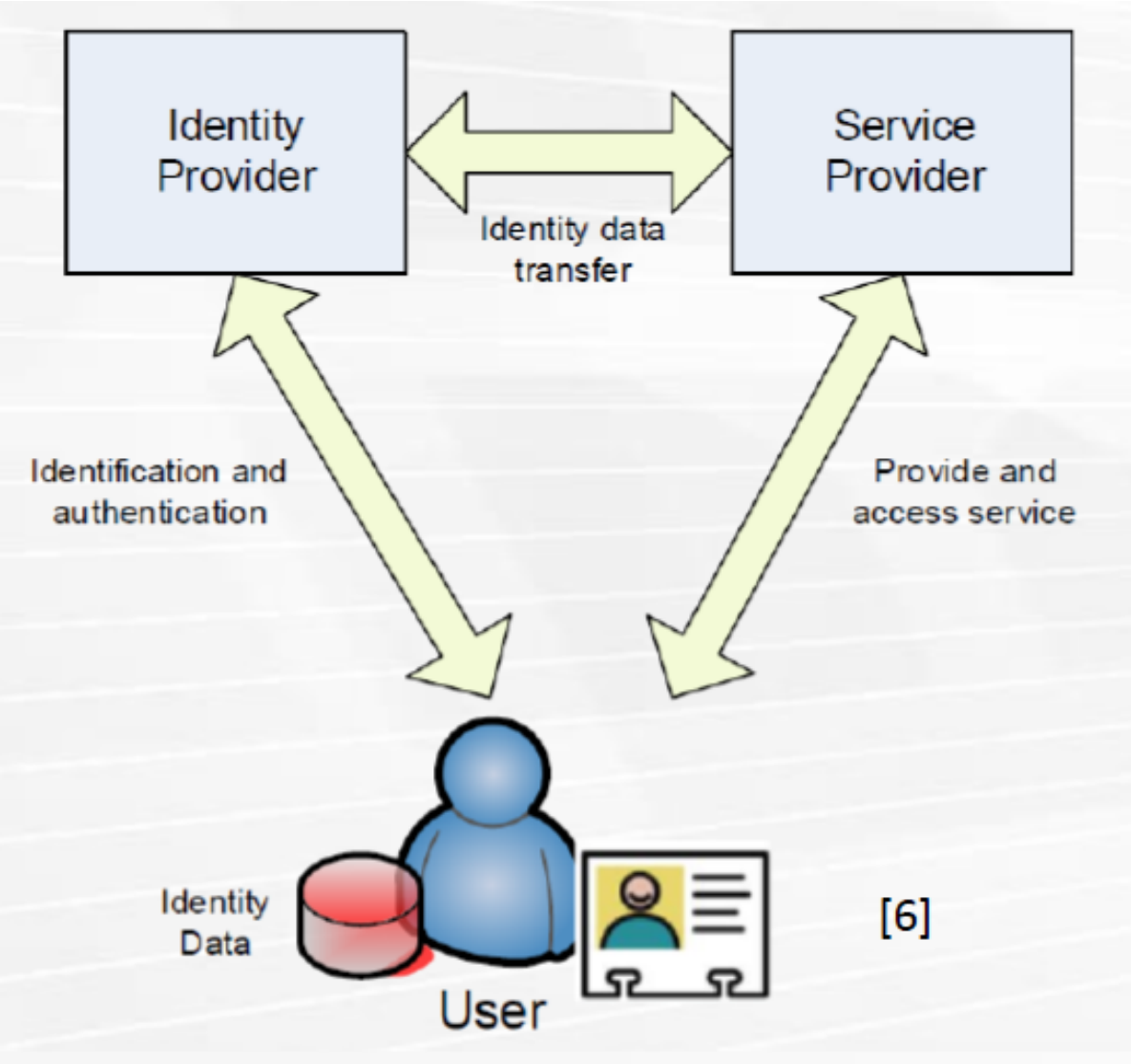
FEDERATED IDENTITY MANAGEMENT

- Identity data is distributed accross several Identity Providers
- Identity data are linked
- Trust relationship between Identity Providers required



USER-CENTRIC IDENTITY MANAGEMENT

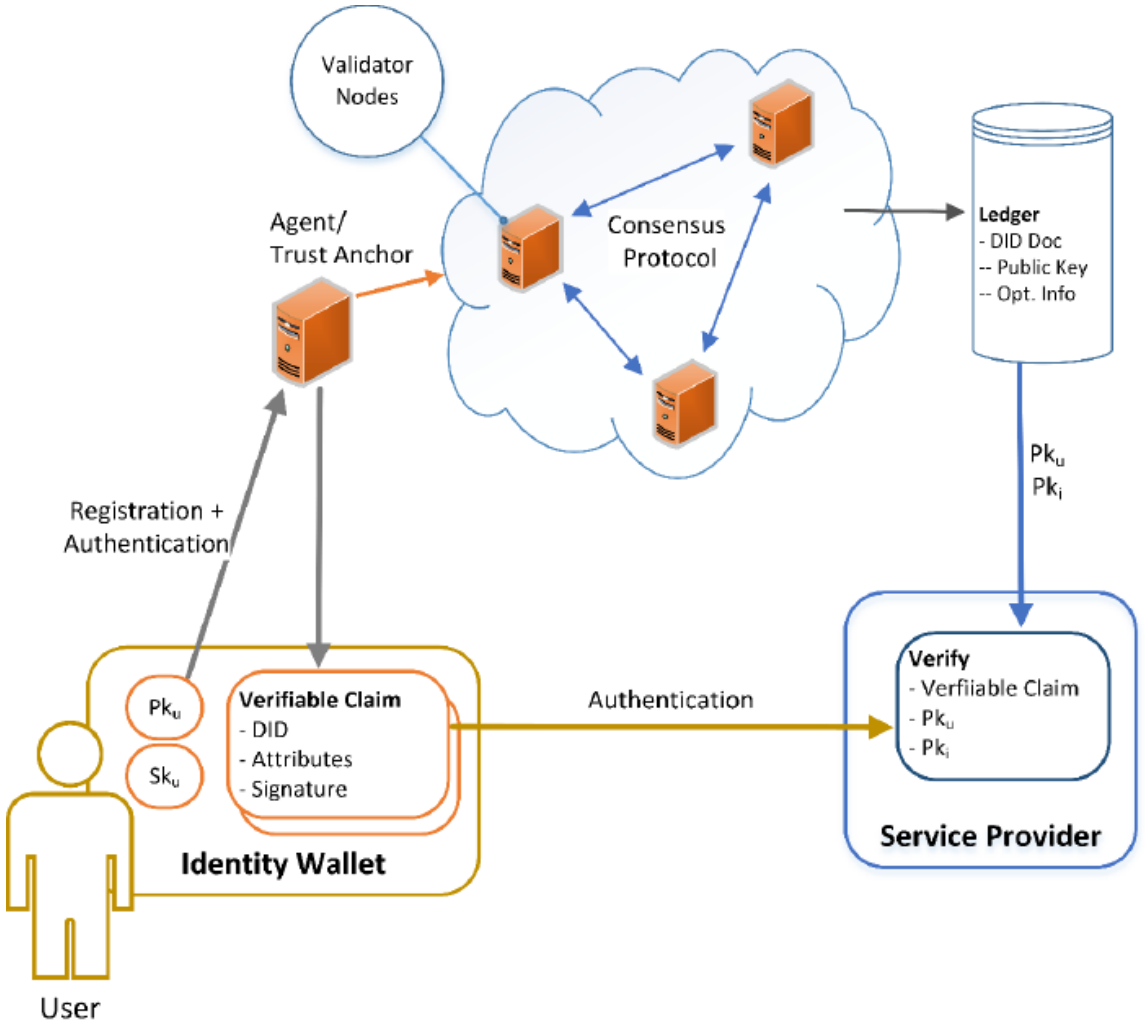
- Identity data are stored in the user‘s domain
- Sharing of identity data requires explicit user consent



OYD
OwnYourData

SELF-SOVEREIGN IDENTITY

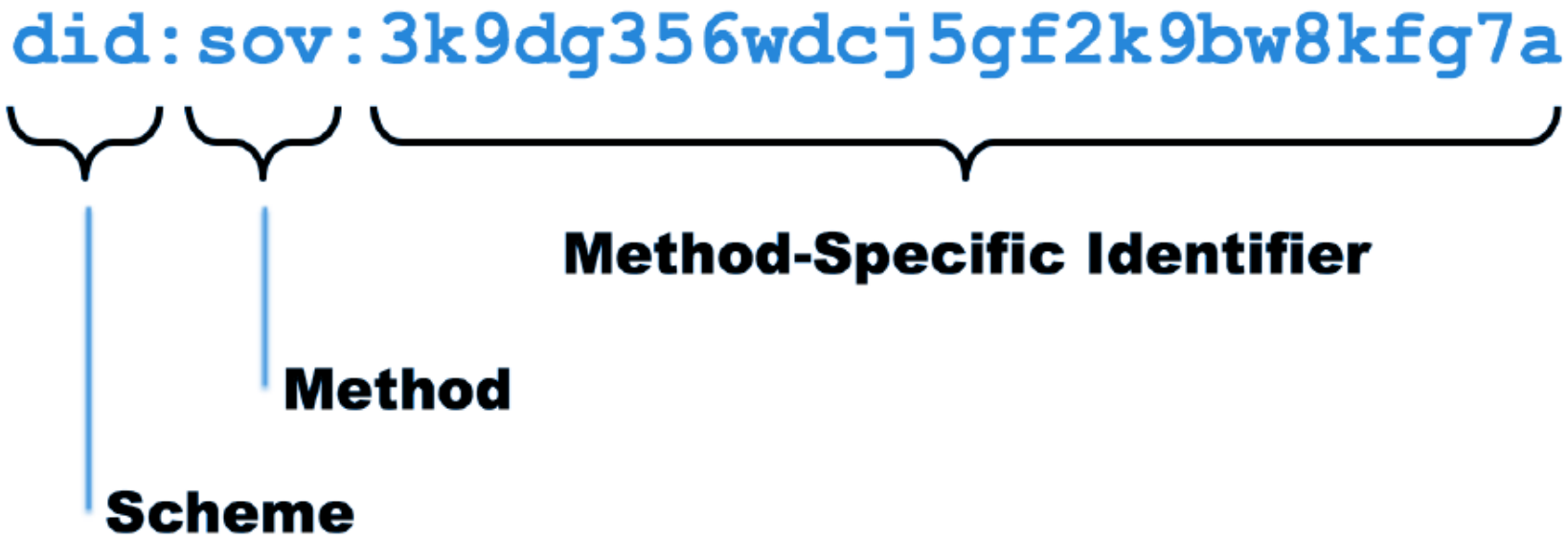
- User fully controls identity data and can create updates as well as delete own identity
- Utilizing distributed ledger technology
- Without trust in a central authority
- Trust is distributed to nodes



OYD
OwnYourData

Decentralized Identifiers (DIDs)

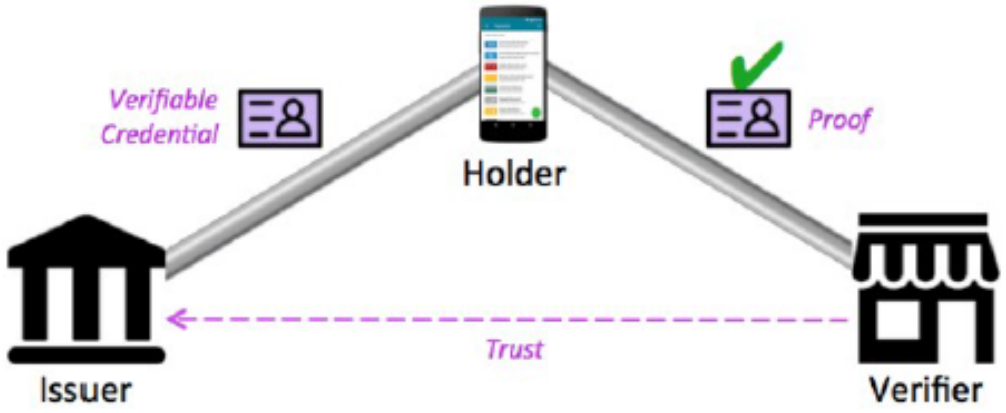
- Self-sovereign identifiers for individuals, organizations, things
- Decentralized, persistent, cryptographically verifiable, dereference-able identifiers
- Registered in blockchain or other decentralized network (ledger-agnostic)
- Resolution: DID → DID Doc
  - Set of public keys, service endpoints, timestamps, proofs & other identifier metadata

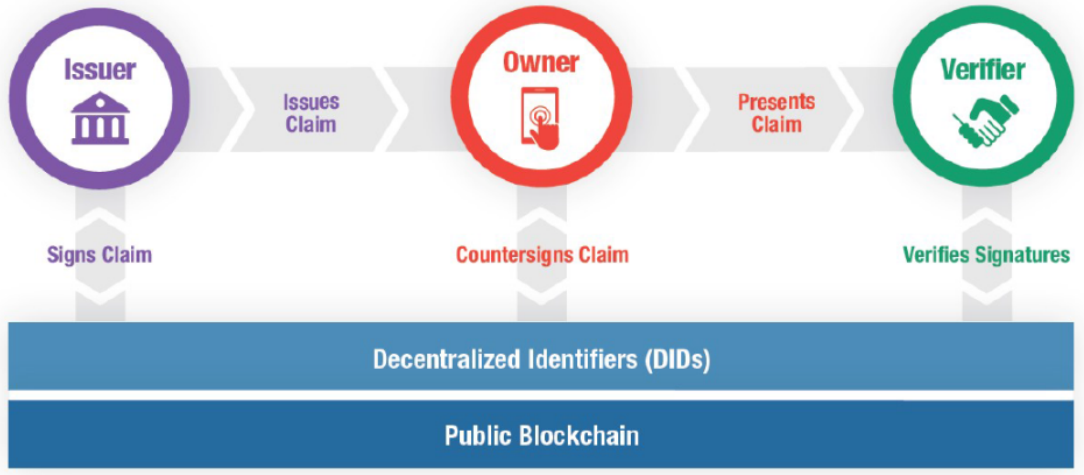


OYD
OwnYourData

Verifiable Claims

- ☐ Identity data, that is “attested” by a trusted party instead of “self-asserted”.
- ☐ Cryptographically verifiable.
- ☐ Semantic statements expressed in JSON-LD / RDF, e.g.:
  - Post attests: I live in 1170 Vienna.
  - University attests: I have a diploma in Computer Science.
  - Bank attests: My credit score is sufficient for a given transaction.
  - Government attests: My name and birthday are ...





OYD
OwnYourData

OYDID: Own Your Decentralized IDentifier

did:oyd:identifier[@location]

- content-based addressing
  - verifiable mapping between DID and DID Document
  - Directed Acyclic Graph for updates
- local-first approach
  - should run locally on own servers
  - decentralised through content based addressing
- low cost
  - independent of 3rd party storage/processing or DLT

OYD
OwnYourData

OYDID

3 Artefacts:

- ☐ DID
- ☐ DID Document
- ☐ Log (actually a directed acyclic graph / DAG)

OYD
OwnYourData

DID

did:oyd:identifier[%40location]

- "did": protocol
- "oyd": specific DID method
- identifier: encoded hash value of DID document
- ":": separator
- "@" / "%40": optional separator between "did:oyd:identifier" and location
- location: optional host that is *recommended* for resolving the DID document

OYD
OwnYourData

DID DOCUMENT

Internal DID Document structure

```
{
        "doc": {JSON object holding payload},
        "key": "strings of encoded public did and public revocation key separated by :",
        "log": "string of encoded hash of termination log entry"
}
```

the DID identifier is calculated from the hashed and encoded DID Document using Multiformats (https://multiformats.io/) for "digest agility“:

- hash function: default is SHA2-256 (RbNaCl::Hash.sha256)
- encoding: default is Base58-btc

Example: `did:oyd:zQmZ8DEGQtJcpoQDMKYJkTiQn9dQLM2QzvmDQXuj8vCfvdj`

OYD
OwnYourData

LOG

#	operation (op)	timestamp (ts)	document (doc)	signature (sig)	previous
1	2 - create	1	DID identifier	sig(doc, private did key)	[ ]
2	0 - terminate	1	$hash_a$(revoke)	sig(doc, private did key)	[ ]
3	1 - revoke	1	$hash_b$(DID Doc)	sig(doc, private rev key)	[#1, #2]

the Log is an array of JSON objects with the following attributes and structure
{"op": int, "ts": int, "doc": string, "sig": string, "previous": array}

- $hash_a$(revoke)
  is the Base58 encoded SHA-256 hash of a subset of the revocation log entry (#3):
  {"op": int, "ts": int, "doc": string, "sig": string}

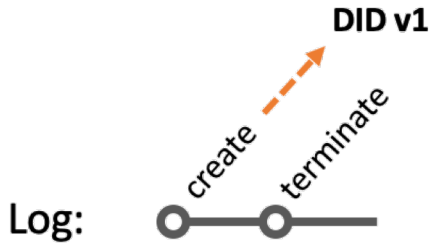
- $hash_b$(DID Doc)
  is the Base58 encoded SHA-256 hash of a subset of the DID Document:
  {"doc": {JSON object}, "key": string}

- signature column (sig)
  holds the Base58 encoded SHA-256 signature hash of the "doc" field using the respective private key; those signatures should be verified upon resolving the DID using the public keys in the DID Document to verify authenticity of the data

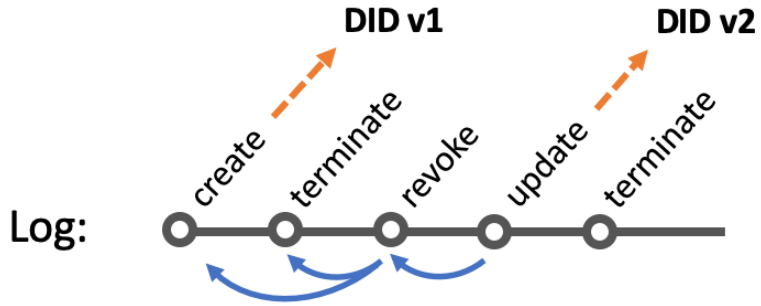
- previous is an array of Base58 encoded SHA-256 hashes of previous entries

OYDID Log

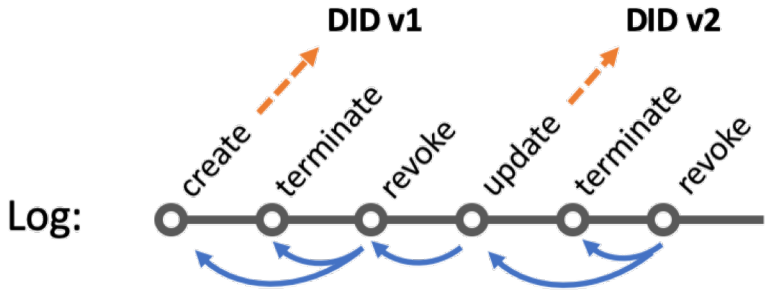
- ☐ Create



- ☐ Update

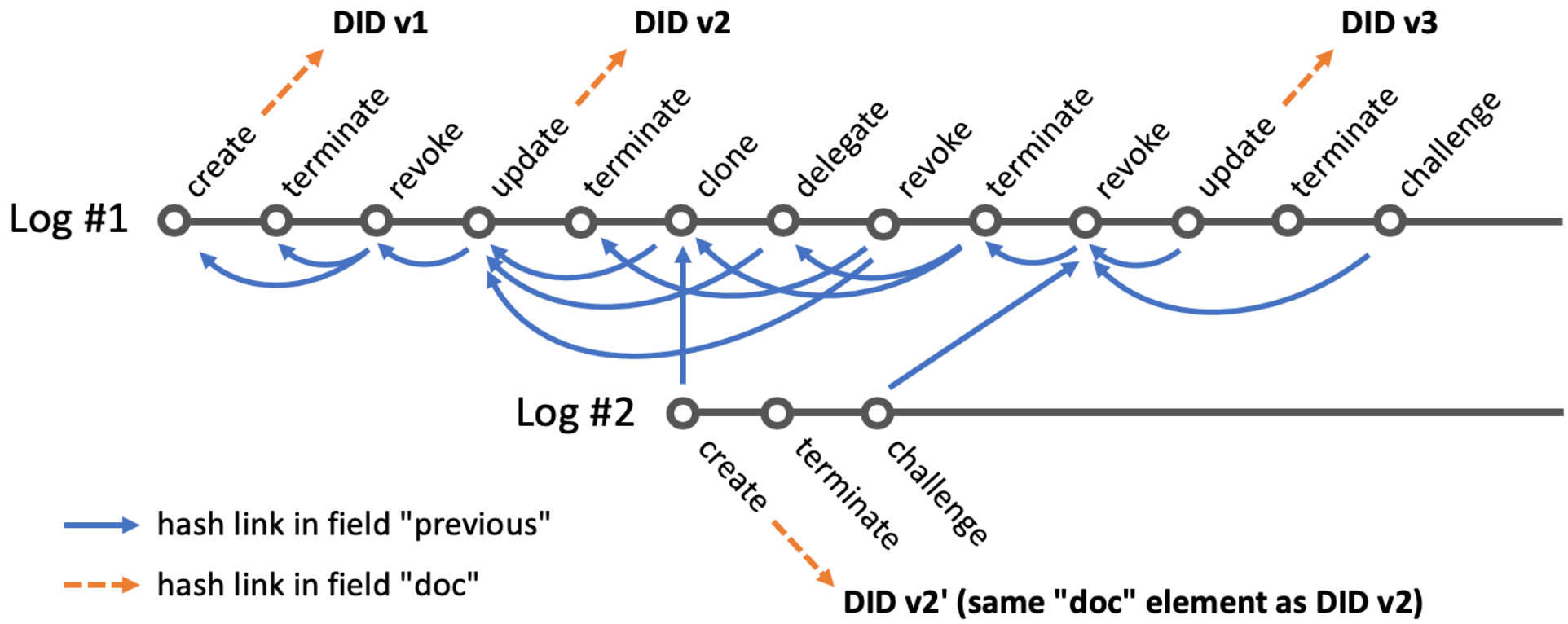


- ☐ Deactivate



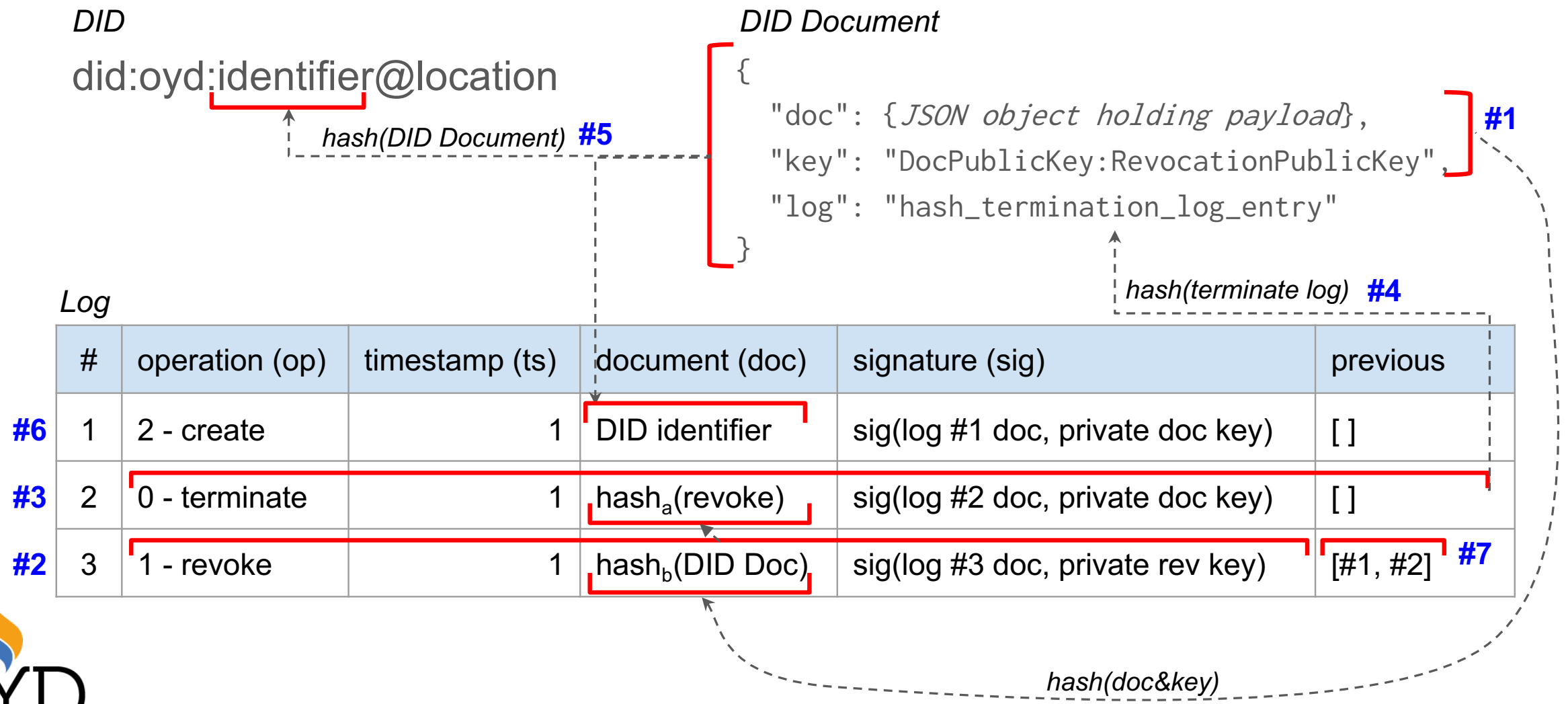
OYD
OwnYourData

Cloning OYDID



OYD
OwnYourData

PROCESS VISUALIZATION



OYD
OwnYourData

USE CASES

- Peer-to-Peer Betting App
  - address question: trust your counterpart
  - participant maintains betting history in DID & VCs
- Ad-hoc Marketplace
  - address question: time & location restricted information sharing
  - settings: pub, class, conference with one-time-QR to check-in/out
- Performance / Application Areas Evaluation
  - address question: 160+ DID methods https://w3c.github.io/did-spec-registries/#did-methods
  - develop evaluation criteria to compare DIDs

OYD
OwnYourData

RESOURCES

- OYDID White Paper
  https://github.com/OwnYourData/oydid/blob/main/docs/OYDIDintro.pdf
- did:oyd W3C-conform DID Spec: https://ownyourdata.github.io/oydid/
- Tools
  - Command line tool: https://github.com/OwnYourData/oydid/tree/main/cli
  - Docker image: https://hub.docker.com/r/oydeu/oydid-cli
  - and repository for hosting, uniresolver plugin, JS library for `did-resolver`
    https://github.com/OwnYourData/oydid
- HackMD for class: https://hackmd.io/wQ4vZuyWTxqbYrhJqOUNXw

OYD
OwnYourData

Dr. Christoph Fabianek

christoph@ownyourdata.eu

@OwnYourDataEU

Your Data is precious.

OWNYOURDATA.EU

OYD
OwnYourData