Decentralized Identifiers (DIDs)

Presentation held at the Fintech event of the W3C Chinese Web IG, on 2020-12-19

Ivan Herman, ivan@w3.org

These slides: https://iherman.github.io/did-talks/talks/2020-Fintech/#/

Introduction

Importance of identifiers in a digital world

- It is increasingly important to identify persons, concepts, things…
  - any reasoning, control, associations, etc., of resources rely on this ability
- The digital economy relies on proper identification to combine information from different sources
  - it is vital that identifiers are unique

Globally unique identifiers are all around us

- They are becoming ubiquitous:
  - persons
  - companies, institutions,…
  - books, magazines,…
  - retail items
  - genes, proteins, viruses,…
  - stars, galaxies,…
  - vehicles, airplanes,…
  - intelligent home devices, Internet/Web of Things,…

What are the problems?

A typical experience

Consider these two scholarly references:

- *Tomislav Strinić, Damir Buković, Ljubomir Pavelić, Josip Fajdić, Ivan Herman, Ivica Stipić, Ivan Palada & Ivana Hirš, "Anthropological and clinical characteristics in adolescent women with dysmenorrhea". Collegium antropologicum, 27(2), (2003).*
- *Ivan Herman, Markus Gylling, "Bridging the Web and Digital Publishing", The Journal of Electronic Publishing, (2015).*

- Only one of the two publications is mine…
- The name is not enough; you need a *unique personal identification* to avoid problems with, in this case, homonyms
- This has become even more important in a networked, digital world

Of course, I do have identifiers

- ivan@w3.org
- ivan@ivan-herman.net
- https://www.w3.org/People/Ivan/
- https://www.ivan-herman.net/
- 0000-0003-0782-2704 (ORCID)
- 89df9321-bf5c-4237-aabc-1f8f202ab5c6 (UUID)

Problems with current identifiers

- *Is it easy to create?*
    - https://www.ivan-herman.net depends on buying a host name
    - ivan@w3.org is not meant to be an identifier, and an email address is not “cheap”
- *Is it decentralized?*
    - https://www.ivan-herman.net depends on a single point of failure; what happens if the hosting site disappears?
    - 0000-0003-0782-2704 depends on the ORCID database. What happens if it is discontinued, hacked, etc?

Problems with current identifiers

- *Is it persistent*
  - When I leave the W3C then ivan@w3.org disappears…
  - If I do not pay for the `ivan-herman.net` domain any more, the URL disappears…
- *Is it resolvable to some reasonable information?*
  - How can I get more information on what 0000-0003-0782-2704 identifies?
- *Is it (cryptographically) verifiable?*
  - What about https://www.ivan-herman.com? How can one prove that this domain is ***not*** referring to me?
  - What happens if I stop paying for the domain and somebody else buys it?

No identifiers display *all* those requirements!

A DID is a self-sovereign identity, i.e., lifetime, portable, and verifiable digital identity that does not depend on any centralized authority

Goals of DIDs

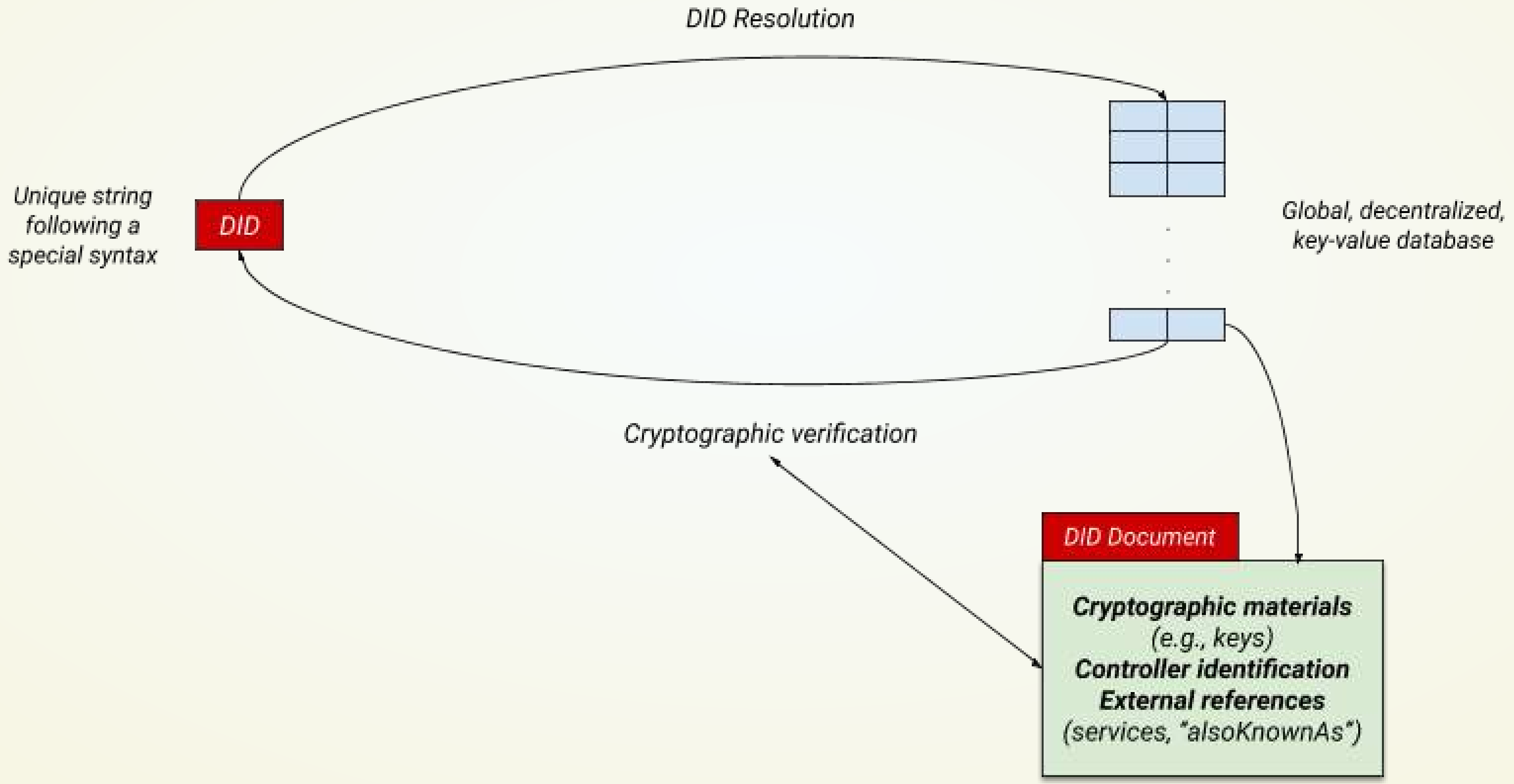
- *Ease of creation*
  - it should be quick and “cheap” to create possibly thousands of DIDs
- *Decentralized*
  - do not depend on centralized registries, identity providers, authorities, etc.
  - the DID has a sovereign controller, an the entity identified explicitly
    - the “subject” of the identifier may be different (e.g., the owner of a dog “controls” the DID identifying the dog)

Goals of DIDs

- *Persistent*
  - once created, it is permanently assigned to the subject
- *Resolvable*
  - it is possible to find out basic set of information on the subject
- *Cryptographically verifiable*
  - there is a mechanism to cryptographically prove that it indeed identifies a specific subject (possibly controlled by a separate controller) and nothing else

High level view on DIDs

High level view: DIDs and DID Documents



DID Resolution

Unique string following a special syntax

DID

Global, decentralized, key-value database

Cryptographic verification

DID Document

Cryptographic materials
(e.g., keys)
Controller identification
External references
(services, "alsoKnownAs")

“Global, distributed, key-value database”

- Also known as ”Verifiable Data Registry”
- There may be several of those!
  - in the DID world, the term *method* is used for the different approaches and/or implementations
- Different methods can have very different characteristics
  - May be based on distributed ledgers (generic or specialized)
  - DID documents stored on specialized sites (e.g., github)
  - May be ephemeral DIDs with lighter requirements (e.g., on an intelligent device)
- The choice depends on the relative importance of the various requirements for a specific usage

DID Documents

- Contain reference to the “controllers”, i.e., entities that may make changes on the DID Document
- Include cryptographic information related to the DID subject
  - RSA, secp25519 elliptical curve keys, bitcoin elliptic curve keys, etc.
  - can be expressed using JWK, or with a DID specific terms
  - can be used for
    - authentication;
    - assertions (e.g., of credentials);
    - key agreement (e.g., to establish secure communication);
    - capability invocation (e.g., authorization to access an API);
    - capability delegation (e.g., delegate an API access to another authority);
    - …

DID Documents (cont.)

- May contain other types of data related to the subject
  - reference to alternative identities (“`alsoKnownAs`”)
  - various service references (e.g., access to a service providing credentials)
  - etc.
- May or may not physically “exist” somewhere in the database
  - some methods generate them on-the-fly

DIDs and DID Documents are closely coupled

- DIDs have the right characteristics through the DID Document
  - DID documents are the “representation data and metadata” of a DID in the Web architecture
- A DID Document is tightly bound to the DID it “describes”
- DID+DID Document may be also used as a decentralized cryptographic keychain for various cryptography applications

Serialization of DID Documents

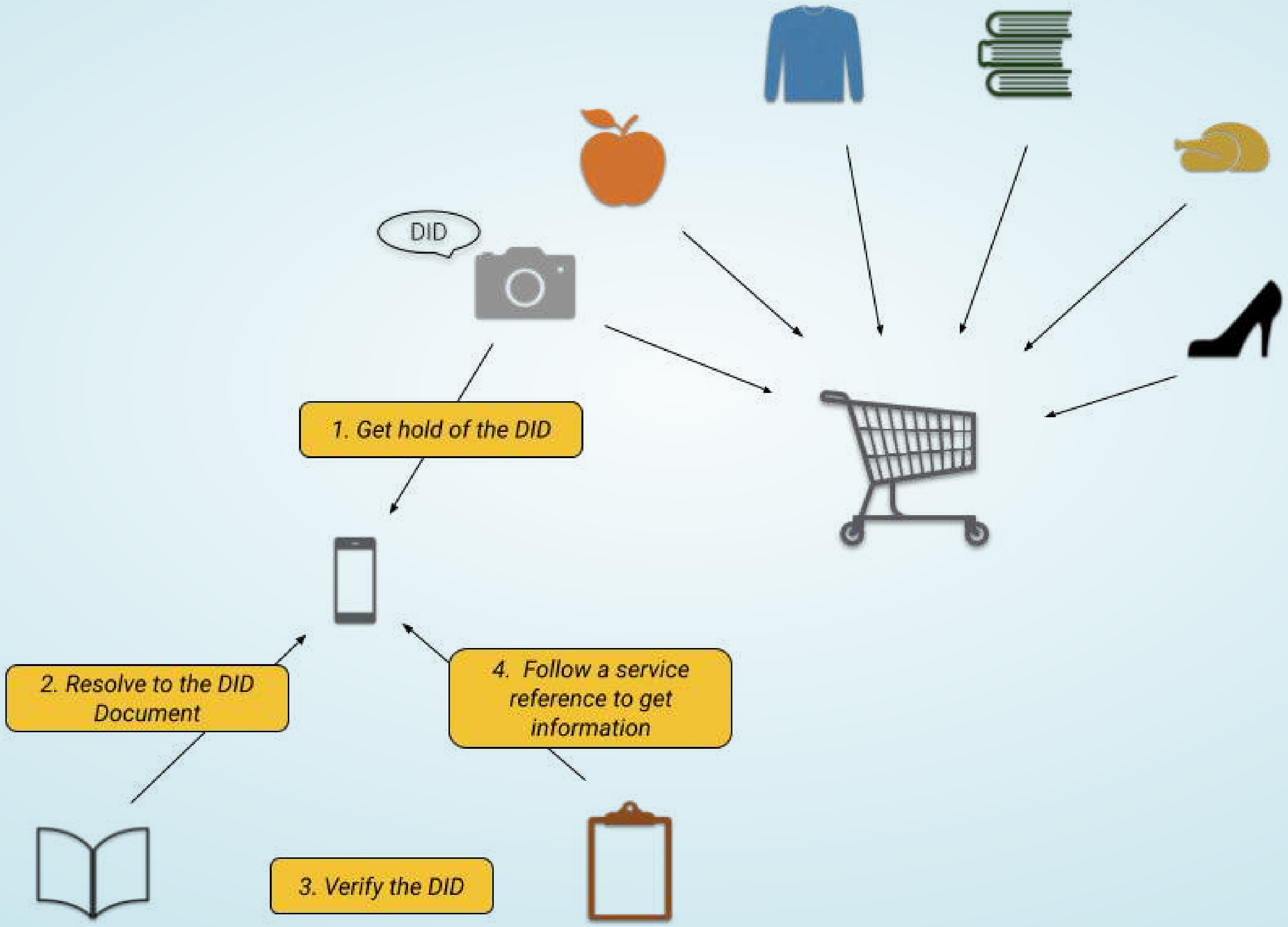
- DID Documents are defined via an abstract data model
- Can be serialized as:
  - JSON
  - JSON-LD
  - CBOR
  - other serializations may come to the fore

Some use cases

There are simple ones

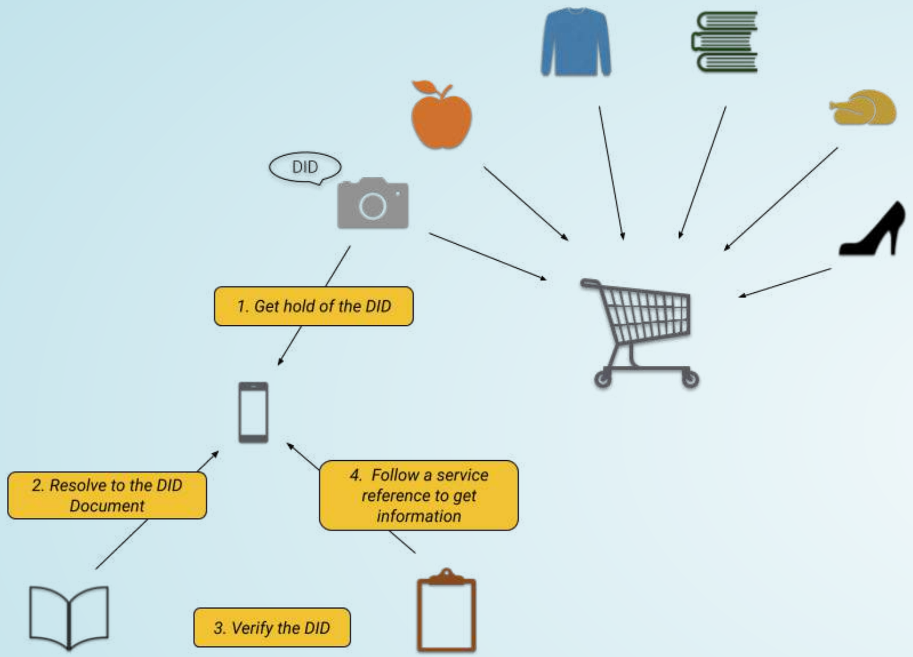
- Securing unique, secure, etc., identities for persons, animals, objects, abstract concepts…
  - unique and unambiguous literature references
  - consistent semantic statements on “resources”
  - identify objects in an internet of “things”
  - etc.
- DID usage is often bound to Verifiable Credentials
  - e.g., life-long credential proving a University Degree, identified with a DID

Find information on purchased goods



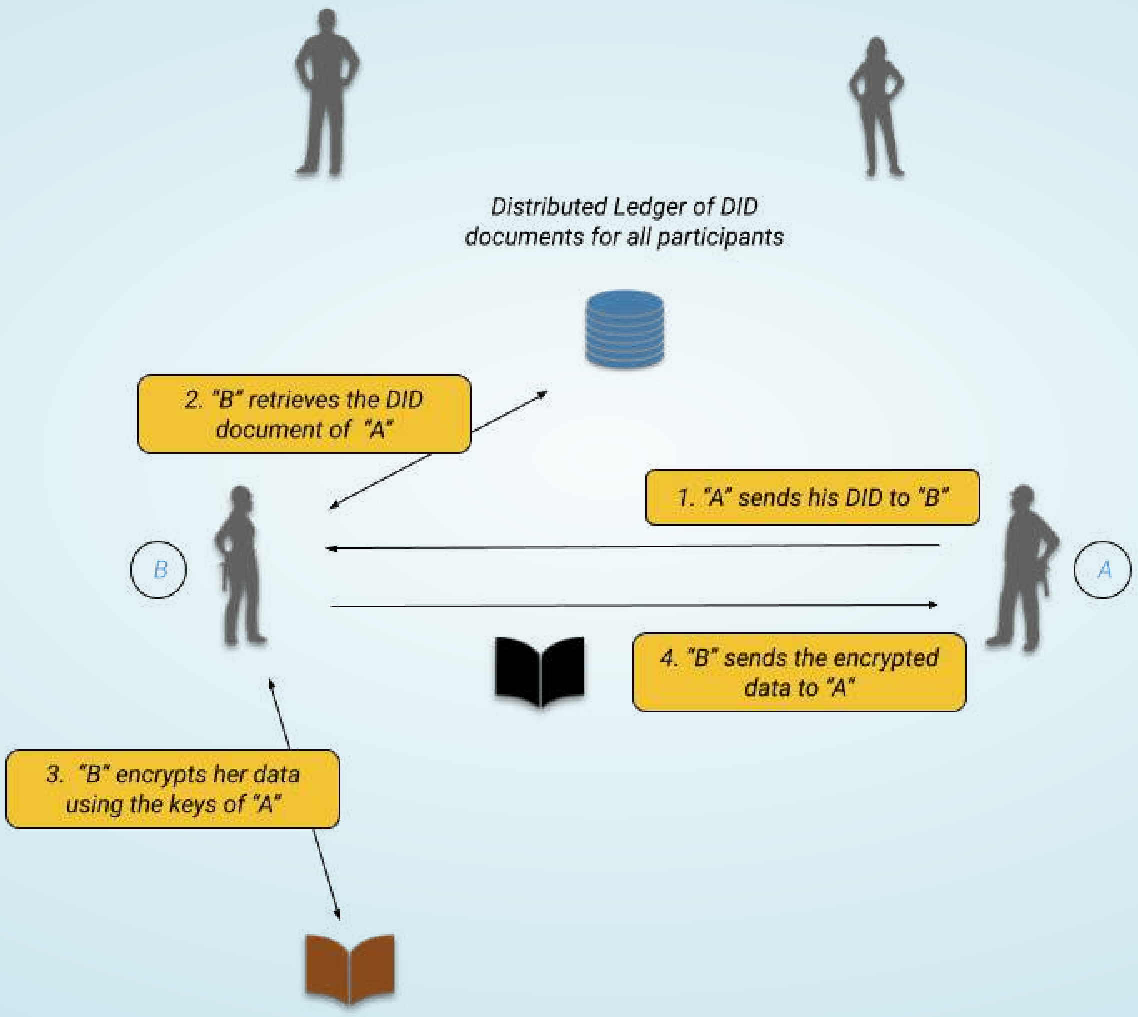
DID
1. Get hold of the DID
2. Resolve to the DID Document
3. Verify the DID
4. Follow a service reference to get information

Find information on purchased goods


DID
1. Get hold of the DID
2. Resolve to the DID Document
3. Verify the DID
4. Follow a service reference to get information

- It is very important that:
  - the identification is unique and persistent
  - the information has not been tampered with
- The same mechanism can be used for constituent parts of goods but the information can remain fully decentralized

Pool of relationships



Distributed Ledger of DID documents for all participants

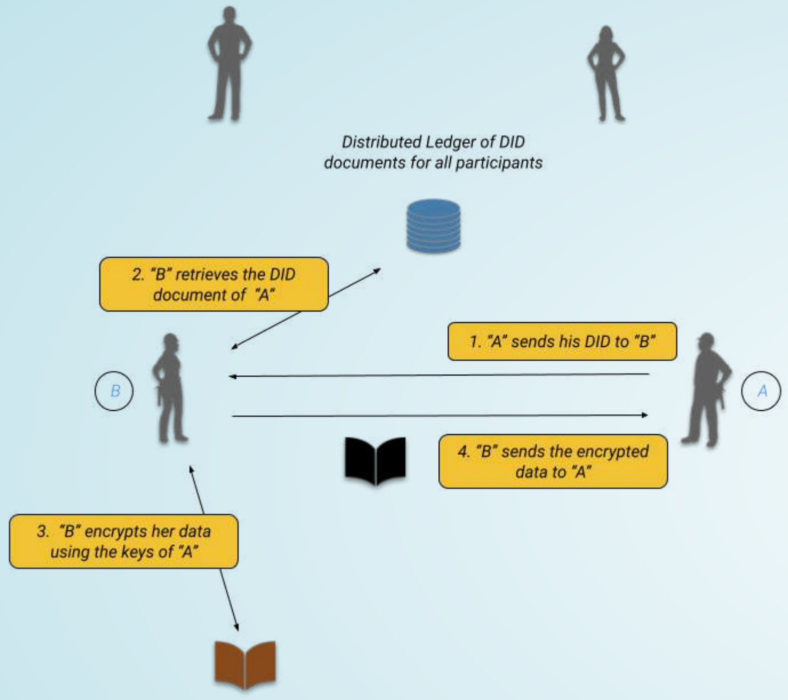
2. "B" retrieves the DID document of "A"

1. "A" sends his DID to "B"

4. "B" sends the encrypted data to "A"

3. "B" encrypts her data using the keys of "A"

Pool of relationships



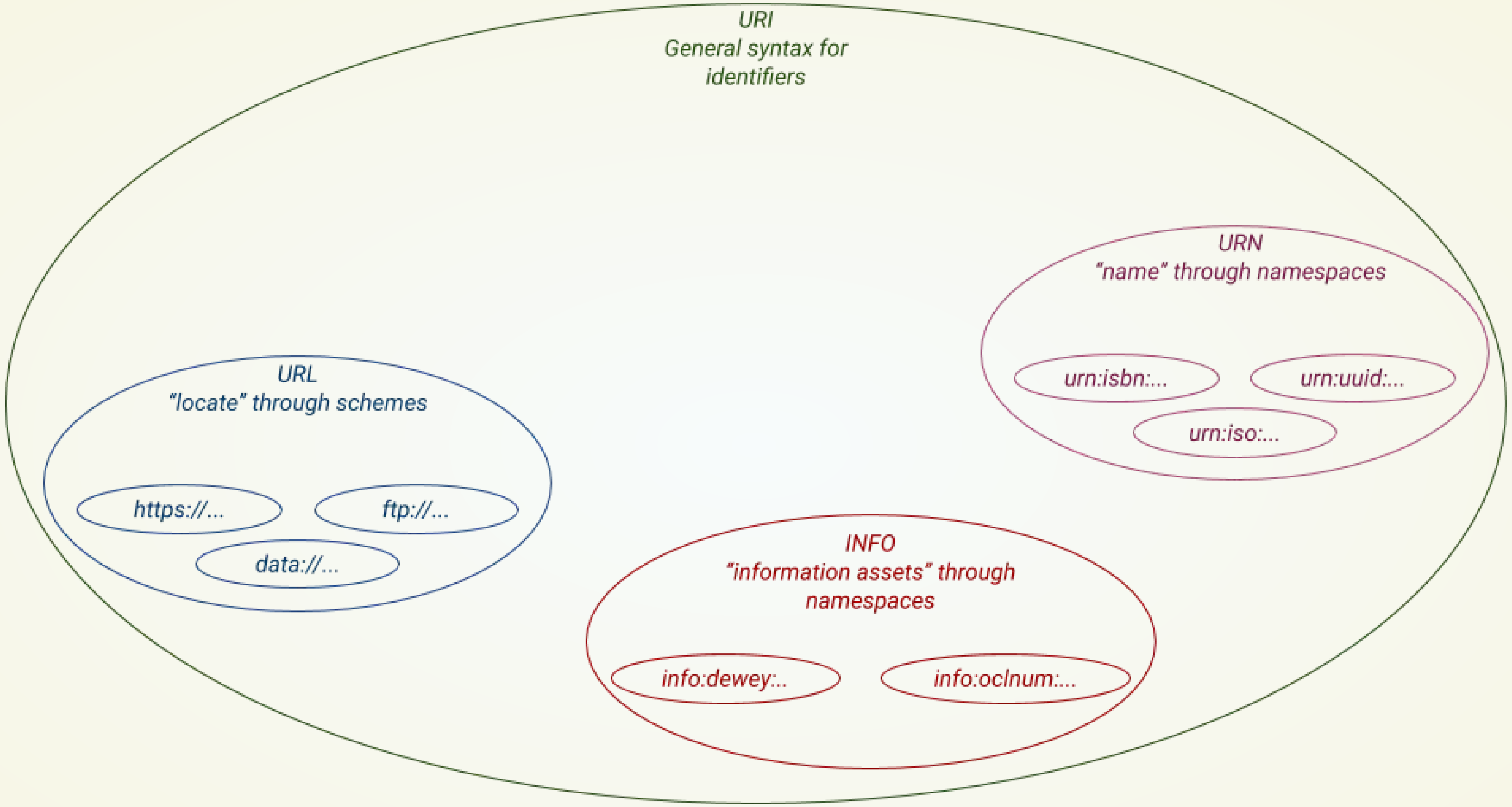
Distributed Ledger of DID documents for all participants
2. "B" retrieves the DID document of "A"
1. "A" sends his DID to "B"
B
A
4. "B" sends the encrypted data to "A"
3. "B" encrypts her data using the keys of "A"

- No need for a complex and centralized key management system
  - there may be different ledgers for the various participants
- Both "A" and "B" may remain anonymous

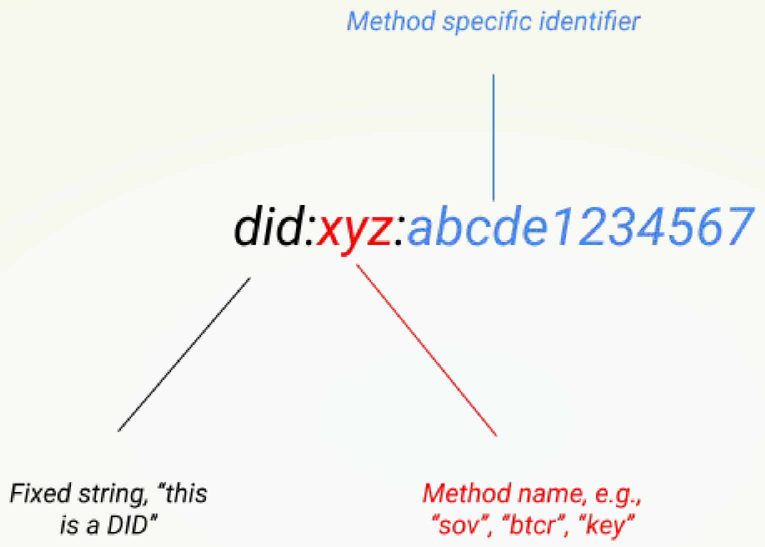
Some technical details

How do DIDs look like?

Reminder: URIs, URNs, URLs, …



DID is a new type of URI



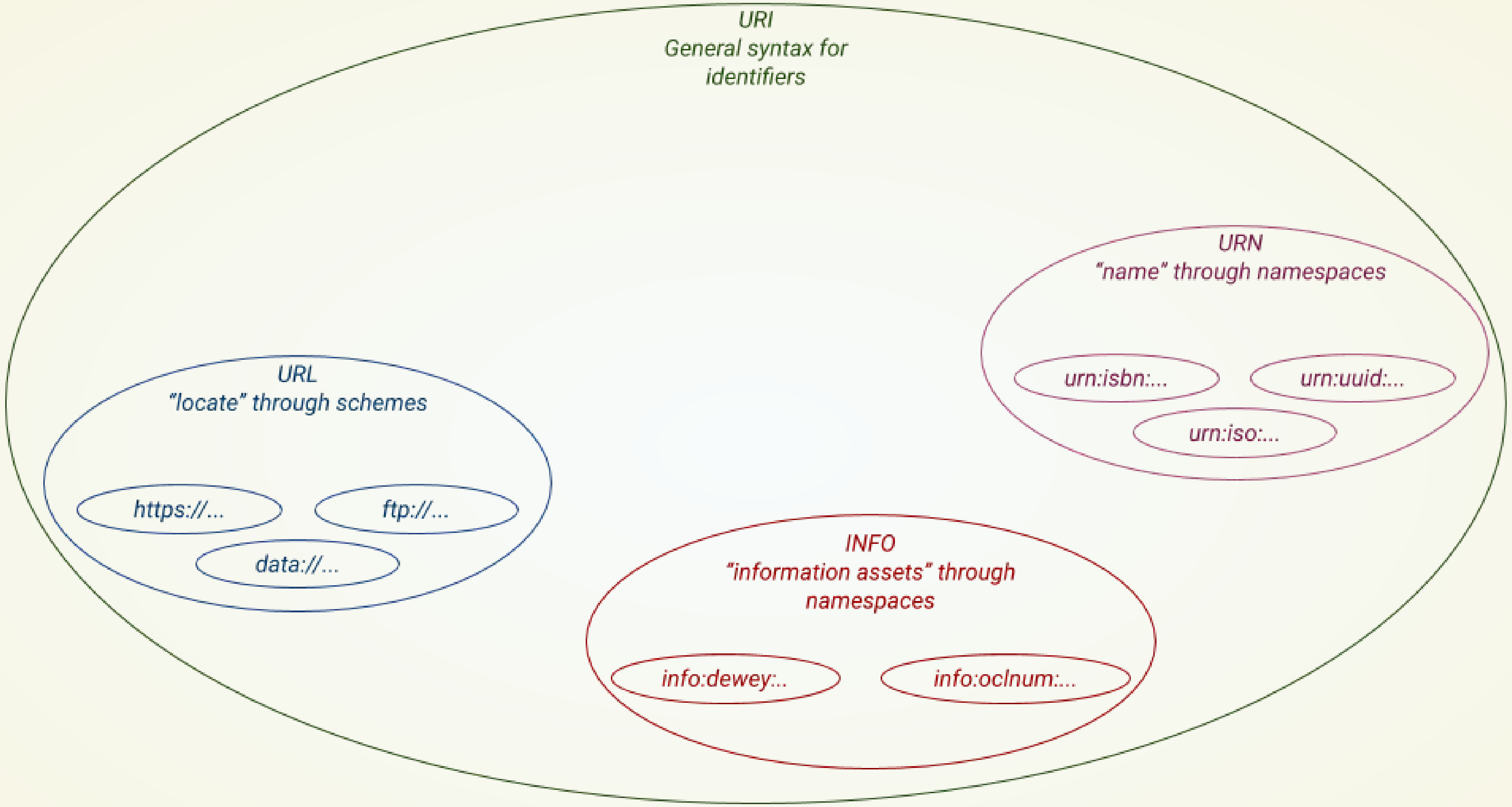
Method specific identifier

did:xyz:abcde1234567

Fixed string, “this is a DID”

Method name, e.g., “sov”, “btcr”, “key”

DID is a new type of URI



Why is it important that it is a URI?

- DID are within the IETF/W3C world
  - tools, libraries may be used to manage them
  - existing specifications automatically apply to DIDs:
    - "`<a href='did:btcr:xyv2-xzpq-q9wa-p7t'>abcd</a>`" is valid HTML
    - "`<did:btcr:xyv2-xzpq-q9wa-p7t> a rdf:Class.`" is a valid RDF Turtle statement
- *DIDs are part of the Web*

Examples for ledger based DIDs

- `did:btcr:xyv2-xzpq-q9wa-p7t`
  - built “on top” of the Bitcoin blockchain
  - the method specific identifier is generated from the bitcoin transaction position reference
- `did:sov:mnjkl98uipsndg2hdjdjuf7`
  - based on a dedicated distributed ledger (Sovrin)
  - the method specific identifier generated from either a simple UUID or the subject’s public keys

Ledger based DIDs

- There are other methods based on generic (e.g., Ethereum), or dedicated (e.g., Veres One) ledgers
- They are generally meant to be general solutions for identity, usable by various applications

Non-ledger based DIDs

- General solutions for identity storage but based on other technologies:
  - github method, based on the user’s Github presence, with DID documents stored in a dedicated (per-user) repository
  - methods looking into the usage of IPFS
  - etc.

Examples for special purpose methods

- `did:key:z6Mki7KaCeTufKQ6…NEv28PhP1PHF35btNN`
  - can be used for single, ephemeral interactions (e.g., IoT)
  - the method specific identifier is an encoded public cryptographic key
  - the DID documents aren't stored; they are generated on demand
- `did:peer:1zQmZMygzYqNwU6Uhmewx…LSwwgf2aiKZuwa`
  - interaction among a fixed number of "peers", e.g., business relationships
  - the method specific identifier is generated from the DID document
    - the document stores the user's public key(s)
  - all participants have access to the DID documents
  - information in the DID documents are used to exchange encrypted messages among peers

Methods in general

- Lots of experimentation is happening, exploring different methods
  - there is also a need to develop proper user interfaces, applications, etc, to store DIDs in personal wallets, for example
- We can expect to see a convergence of methods to only a few in the coming years

How do DID Documents look like?

Abstract model of a DID document

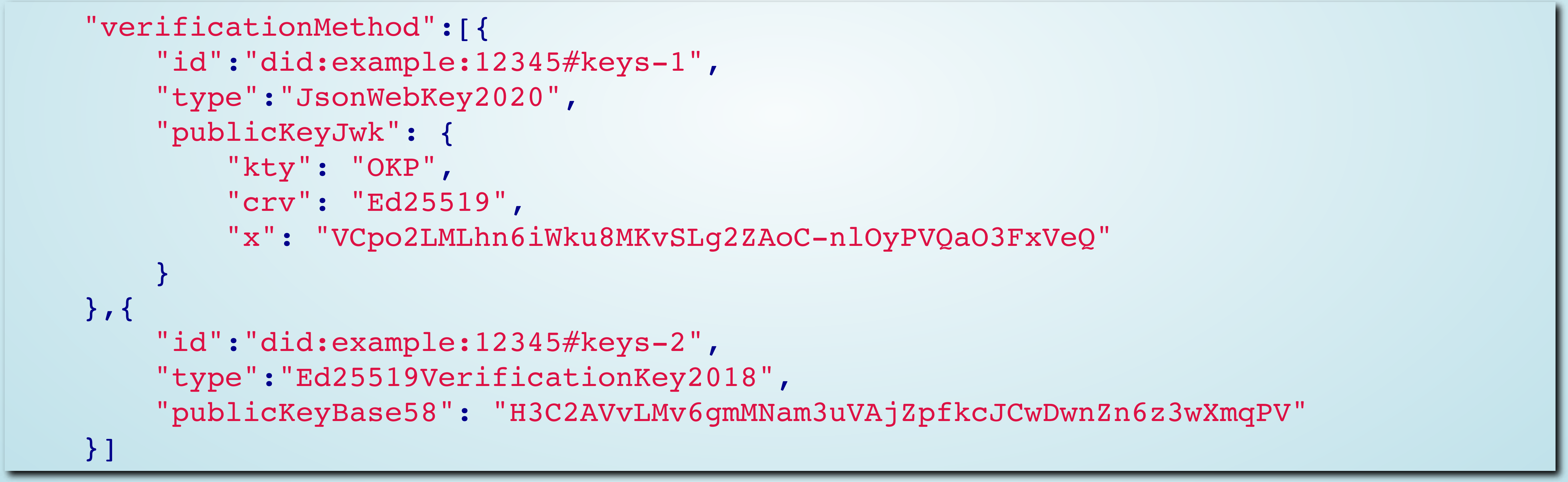
- Uniquely related to the DID *subject*, i.e., the entity identified by the DID
  - the document must contain the DID itself
- Includes a separate DID for the *controller*
  - identifies an entity that “in charge” of the DID document
- Expresses public cryptographic keys and other verification methods
- May be extended to include application or method specific information
- Serialized in JSON, JSON-LD, or CBOR

Typical DID document structure

```
{
    "id":"did:example:abcedefgh",
    "controller":"did:example:xyzwvy",
    "verificationMethod":[{ ... }],
    "authentication":[{ ... }],
    "assertionMethod":[{ ... }],
    "service":[{ ... }]
}
```

Verification methods

- List of various public keys
- Their usage is not specified: can be used for DID verification but also for any other application



```
"verificationMethod":[{
    "id":"did:example:12345#keys-1",
    "type":"JsonWebKey2020",
    "publicKeyJwk": {
        "kty": "OKP",
        "crv": "Ed25519",
        "x": "VCpo2LMLhn6iWku8MKvSLg2ZAoC-nlOyPVQaO3FxVeQ"
    }
},{
    "id":"did:example:12345#keys-2",
    "type":"Ed25519VerificationKey2018",
    "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
}]
```

Authentication

- Keys that can be used for the *authentication* of the controller:
  - can refer to a key listed separately in `publicKey` (via a DID URL)
  - can include a full key that can be used for authentication only

```
"authentication":[
"did:example:12345#keys-1",
{
    "id":"did:example:12345#keys-3",
    "type":"X25519KeyAgreementKey2019",
    "publicKeyBase58": "9hFgmPVfmBZwRvFEyniQDBkz9LmV7gDEqytWyGZLmDXE"
}]
```

Service endpoints

- Discovering any service endpoints the subject wants to advertise

```
"service":[{
    "type":"IdentityHub",
    "verificationMethod":"did:example:12345#keys-2",
    "serviceEndpoint":"https://example.org/identityservice"
},{
    "type":"MessagingService",
    "serviceEndpoint":"https://example.org/photos/34567"
}]
```

Documents to read

Use cases and requirements

https://www.w3.org/TR/did-use-cases/

Core spec

https://www.w3.org/TR/did-core/

DID Specification Registries

https://www.w3.org/TR/did-spec-registries/

These slides

https://iherman.github.io/did-talks/talks/2020-Fintech/

Some more documents to come

Rubric

Documenting what criteria to look for when choosing a specific method

Implementation guide

Today’s Status

- Draft specification was developed in a W3C CG
- Working Group started in September 2019
- Plan is to be technically ready (i.e., Candidate Recommendation) in January 2021
- Recommendation should be available by by the end of 2021

Thank you for your attention!

ivan@w3.org

These slides: https://iherman.github.io/did-talks/talks/2020-Fintech/#/