This document is an “Explainer” as requested by the Technical Architecture Group at the W3C. An explainer for explainers can be found here: https://github.com/w3ctag/w3ctag.github.io/blob/master/explainers.md

We’ve populated this document with the template provided in that link. As the document evolves, there will be a period with a confusing mix of references to "your feature" and concrete explanations about DIDs.

NOTE: In essence, every sentence should support the delineation of what is in scope and out of scope.

Decentralized Identifiers (DIDs) Explained

(Community Rough Draft)

February 6th, 2019

What’s all this then? (Dan)

A brief, 4-5 paragraph explanation of the feature’s value. Outline what the feature does and how it accomplishes those goals (in prose). If your feature creates UI, this is a great place to show mocks and user flows.

A *decentralized identifier*, or DID, is a new URI scheme. Unlike existing HTTP URLs, email addresses, commercial account identifiers, most government identifiers, etc., the DID was purpose-designed to support cryptographically-controlled identifiers backed by decentralized ledgers such as Bitcoin and Ethereum, among others.

https://tools.ietf.org/html/rfc3986#page-7

Syntactically, a DID looks like this: did:btcr:blahblah. The three parts of a DID, separated by colons, are:

1. The string “did”, representing the “did” URI scheme.
2. A DID *method*, in this case “btcr”, that must be one of the values in the to-be-created DID Registry
3. A string that is interpreted according to the rules of the specific DID method.

For some DID methods the URI string itself is sufficient to understand how to verify control over the DID, including key rotation and revocation. An example is the “btcr” method, which treats the third part of the DID as a Bitcoin address. To support other, more complex, DID methods, the concept of a *DID document* is introduced. A DID method is required to define how a DID is to resolve to a DID document. That document contains, at a minimum, blah blah blah. Additionally, a DID document may provide other details regarding how to authenticate control over the DID. The standard will not define new authentication schemes, but rather use only ones that already exist. Here is an example DID document:

<example DID document>

It is important to note that a DID is a decentralized *identifier*, and not a decentralized *identity*. While a DID could be used as an identifier in an identity system, the standard itself is expected to only define the identifier itself, making no claims about interpretability as an identity.

Although the standard will not expressly enforce privacy, it is a strong goal of the work that DIDs, and specific DID methods, can be used in ways that strongly reduce correlatability of personally-identifiable information. The proof of this is in the existing preliminary implementations for whom such privacy is a strong moral and/or commercial motivator.

What is your Feature? (Dan)

Your [DIDs used in Verifiable Credentials] feature is distinguished from existing [Web URLs and centralized checks of credentials] solutions, by: [...]

- Designed for decentralized access and control by as little as one individual
- Designed to be lightweight, potentially low-cost to create and maintain

Goals

How will the web be better when this feature launches? And who will it help?

- Key management flexibility
    - Indirection
    - Auditability
    - Management without dependence on external authority
- More flexibility for credential issuance and use
- More flexibility for cryptographically anchored transactions
    - Signatures
    - Cryptocurrency
    - Authentication
- Key discovery
- Permanent identifiers, that can be used for various use cases:
    - Issue of long term documents

    - Unique representation of living human individuals
- Transient identifiers, that can be created and destroyed at ease
    - Low cost
    - Minimal or zero administrator
- Service discovery
- Why do existing protocols not already offer this feature?

Limitation of federation? This outline doesn't have pain points, but maybe we should add it as a form of Goals.

Non-goals

You're not going to solve every problem so enumerate the attractive, nearby problems that are out of scope for this effort. This may include details on the tradeoffs made due to architectural limitations made due to implementation details, and features left out either due to interoperability concerns or other hurdles, and how you plan to improve on this. This can often be the single most important part of your document, so give it careful thought.

- Not decentralized identities
- Not for complete disruption of existing systems, can help transition
- Not Replacing DNS
- Not Identity assurance
- Not content addressed data stores
- Not human readable identifiers
- Not defining DID-Auth, but enabling possible authentication methods
- Not for PII, but may lead to services that
- Maybe see also SSI Myths:
    - https://medium.com/evernym/7-myths-of-self-sovereign-identity-67aea7416b1
    - https://medium.com/evernym/7-myths-of-self-sovereign-identity-b16648c3090d
        - Self-sovereign means self-attested.
        - SSI attempts to reduce government's power over an identity owner.
        - SSI creates a national or "universal ID" credential.
        - SSI gives absolute control over identity.
        - There's a "main" issuer of credentials.
        - There's a built-in method of authenticating.
        - User-centric identity is the same as SSI.

Getting started / example code (Dave Longley?)

Provide a terse example for the most common use case of the feature. If you need to show how to get the feature set up (initialized, or using permissions, etc.), include that too

[here]


```
{
  "@context": [
    "https://w3id.org/did/v0.11",
    "https://w3id.org/veres-one/v1"
  ],
  "id": "did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ",
  "authentication": [
    {
      "id":
"did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ#z6MkvHzs56yEvkL5BLeVYLcxSTZ6jDnuvWs5W32B9BoJ1fmL",
      "type": "Ed25519VerificationKey2018",
      "controller": "did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ",
      "publicKeyBase58": "GqjpUriobCqc4qonrmf7bN16ueX4Wdcip27FJuqH6Syx"
    }
  ],
  "capabilityDelegation": [
    {
      "id":
"did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ#z6MkqjDMkavJHgS258Z9kGXTp4QemziQv6YBAHVgNwodLH9z",
      "type": "Ed25519VerificationKey2018",
      "controller": "did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ",
      "publicKeyBase58": "CGxKALfrx8wYxdiT4hZcxxrexRSZWDHpUGakYfqcR4Nc"
    }
  ],
  "capabilityInvocation": [
    {
      "id":
"did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ#z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ",
      "type": "Ed25519VerificationKey2018",
      "controller": "did:v1:nym:z6MkozMi3FAww3ZaXjtXuf6Q8RPbisfiemthzPXxMtowVMGQ",
      "publicKeyBase58": "AY6fSzvWbW57RF3qE68ZHKqbuJPsEteMJNd2Xcqva8V2"
    }
  ]
}
```


Key scenarios (JOE)

Next, discuss the key scenarios which move beyond the most canonical example, showing how they are addressed using example code:

Possible Scenarios (just list without details)

Scenario 1 -- International job applicant

When Sally earned her master’s degree at Oxford, she received a digital diploma which contained a decentralized identifier she provided. Over time, she updates the cryptographic material associated with that DID to use her latest hardware wallet. A decade after graduation, she applies for a job in Japan, for which she provides her digital diploma by uploading it to the

prospective employee's website. To verify she is the actual recipient of that degree, she uses the decentralized identifier to authenticate, using her current hardware wallet (with rotated keys). In addition to the fact that her name matches the name on the diploma, the cryptographic authentication provides a robust verification of her claim, allowing the employer to rely on Sally's assertion that she earned a master's degree from Oxford.

Oxford had no need to provide services for resetting or updating Sally's username or password; they had no role in managing Sally's changes to her authentication credentials. The potential employer did not need to contact Oxford to verify Sally's claim of a master's degree; they were able to verify the credential and authenticate Sally's identity with information retrieved over the Internet.

This scenario uses the proposed Decentralized Identifiers, a digital credential such as a Verifiable Credential, and a strong authentication protocol such as DID-Auth or WebAuthn.

DIDs

Subject DID (referring to Sally)

did:btcr:xkyt-fzgq-qq87-xnhn

Diploma DID (referring to Digital Diploma)

did:btcr:xkyt-fzgq-qq87-xnhn

DID Document

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:btcr:xkyt-fzgq-qq87-xnhn",
  "authentication": [{
    // this key can be used to authenticate as DID ...9938
    "id": "did:btcr:xkyt-fzgq-qq87-xnhni#keys-1",
    "type": "EdDsaSAPublicKeySecp256k1",
    "owner": "did:btcr:xkyt-fzgq-qq87-xnhn",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"

  }],
  "service": [{
    "type": "ExampleService",
    "serviceEndpoint": "https://example.com/endpoint/8377464"
  }]
}
```

Verifiable Credential

```
{
  "@context": [
    "https://w3id.org/credentials/v1",
    "https://w3id.org/openbadges/v2"
  ],
  "id": "https://some.university.edu/credentials/9732",
  "type": ["VerifiableCredential", "OpenBadgeCredential"],
```

```
  "issuer": "did:example:issuer_did",
  "issuanceDate": "2018-02-28T14:58:57.461422+00:00",
  "credentialSubject": {
    "id": "urn:uuid:437fc6ff-bb3c-4987-a4b7-be8661ff6f21",
    "type": "Assertion",
    "recipient": "did:btcr:xkyt-fzgq-qq87-xnhn",
    "badge": {
      "type": "BadgeClass",
      "id": "urn:uuid:7aad3c57-3bfb-45ea-ae79-5a6023cc62e4",
      "name": "Oxford University Master's Degree",
      "description": "Recipient has received a Master's Degree in Economics",
    }
  },
  "proof": {
    "type": "RsaSignature2018",
    "created": "2010-03-07T19:22:15Z",
    "verificationMethod": "did:example:issuer_did/0#signing-key",
    "jws":
"eyJhbGciOiJQUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0Il19..p9RDJqwzXyxfC69dMEz503_ZZ_af1e_hV931dPlIdrofC6p2y_dcjjq
IDysReJy6W_fnN_dZGVoXqFAg2OD_SmbDi5dNMOZILot-zJdDJCxXWuwZtiCFlt29KfLmJs6me0bD5pU4RbknXDoyBhA8muMby8j1fUeBDo3Ienmzv
5UlB3v0f0-w5l6-z_cswHB_UXIlWw4EzcsmLvHzjB7TI76QLwq3KeVPSB3U9aM3o2Ejkq6Ygh5XxUGkXiZUQ5ungQ9Psy_VicjZyOc19LoBPoiPxDH
QodTrqCFNH2qCNhDc4lg2zE8S9KNlQhUUFatzkTN70s23fhWBMKz2a5DWgQ"
  }
}
```

Authentication

This specification does not itself define any methods of authentication. It does provide a way for DID methods to specify the authentication supported by that method.

Although these standards are still in development, there are several implementations that demonstrate what is possible. Sally uses a hardware wallet to securely authenticate over the network. The wallet maintains one or more private keys with which she can respond to challenges associated with the public credential stored in the DID Document. The DID Document is retrieved according to the DID method, likely using a universal resolver such as that under development by the Decentralized Identity Foundation[1]. The DID Document may be stored anywhere, such as on a decentralized file system or on a server Sally controls. Once retrieved, the potential employer has the cryptographic material required to authenticate Sally.

[1] https://medium.com/decentralized-identity/a-universal-resolver-for-self-sovereign-identifiers-48e6b4a5cc3c

Outline the scenario, then provide:

[sample code that demonstrates the feature]

- Educational scenario
    - bearer credential
    - potentially long-lived
    - potentially need for time-stamp in addition to signature

- peer to peer examples also useful (ChristopherA states Kim writes useful Javascript hacks)
- subject may have key rotation
- International acceptance of credential?
- Applying for a job?
-

Scenario 2 (Sam)

Outline the scenario, then provide:

Sick of managing all her identities online and concerned with the news of how much data is collected on her by online platforms and third parties; Sam signed up to lifescope, a digital wallet that pulls her personal data already collected by her favourite applications into her own easy to understand searchable database. This wallet acts as her central sign on but doesn't operate as a master Identity. Because she has all of her own data easily managed she can self issue and selectively share preferences, credentials and verifiable claims about herself. She can 'turn on' her assertions whenever she is online. Her sign on acts like a cookie that brokers her interests. Because this cookie is available now by a sovereign individual representing their own interests it offers a much more valuable target for an advertiser than the 0.1% click through offered by the nefarious guesswork of data brokers. When she goes shopping for furniture she uses her Pinterest as the governing sign on from her Lifescope wallet and attaches her favourite pinterest boards to it. Now she sees what she's interested in on the site, and the ads that follow her are pieces she would have normally pinned to her own board.

When she's searching for jobs she uses her linkedin as her governing sign on, she selected her relevant experience and education to assert as she begins her job search.

Post GDPR Consent management

Euro citizen consent and participation... can respond to an overlay request, a request for specific data types

[sample code that demonstrates the feature]

Detailed design discussion (Manu?)

Tricky design choice #1

Talk through the tradeoffs in coming to the specific design point you want to make, hopefully:

[illustrated with example code]

…

- DIDs are not intended to be human memorable / real / reserved names / not a replacement for DNS
- Avoiding vendor lock-in, tech lock-in (e.g. DLT/blockchain choice), future proofing
- Good answer for “why blockchain”?
- Rotatable keys are a key consideration
    - Update/Delete required?
- Interoperability?

Tricky design choice N

-

…

Considered alternatives (Christopher?)

One of the most important things you can do in your design process is to catalog the set of roads not taken. As you iterate on your design, you may find that major choices in your approach or API style will be revisited and enumerating the full space of alternatives can help you apply one (or more) of them later, may serve as a “graveyard” for u-turns in your design, and can give reviewers and potential users confidence that you’ve got your ducks in a row.

Pain Points

- Lost passwords
- Transient authorities (schools going out of business)
- Cryptographic end-of-life (quantum sci-fi)

Alternatives

- Why not Federation? Or even Centralized?
- Why not ENS (Ethereum Name System), Blockstack, Namecoin, IPNS, etc.
- Items mentioned by David Challener via Snorre in this thread
- OpenID Connect
- Hardware Wallets (& AuthN)

References & acknowledgements

Your design will change and be informed by many people; acknowledge them in an ongoing way! It helps build community and, as we only get by through the contributions of many, is only fair.