JSON-LD

```json
{
  // set the context, which establishes the special terms we will be using
  // such as 'issuer' and 'alumniOf'.
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://www.w3.org/2018/credentials/examples/v1"
  ],
  // specify the identifier for the credential
  "id": "http://example.edu/credentials/1872",
  // the credential types, which declare what data to expect in the credential
  "type": ["VerifiableCredential", "AlumniCredential"],
  // the entity that issued the credential
  "issuer": "https://example.edu/issuers/565049",
  // when the credential was issued
  "issuanceDate": "2010-01-01T19:23:24Z",
  // claims about the subjects of the credential
  "credentialSubject": {
    // identifier for the only subject of the credential
    "id": "did:example:ebfeb1f712ebc6f1c276e12ec21",
    // assertion about the only subject of the credential
    "alumniOf": {
      "id": "did:example:c276e12ec21ebfeb1f712ebc6f1",
      "name": [{
        "value": "Example University",
        "lang": "en"
      }, {
        "value": "Exemple d'Université",
        "lang": "fr"
      }]
    }
  },
```

```json
    "alumniOf": {
      "id": "did:example:c276e12ec21ebfeb1f712ebc6f1",
      "name": [{
        "value": "Example University",
        "lang": "en"
      }, {
        "value": "Exemple d'Université",
        "lang": "fr"
      }]
    }
  }
,
// digital proof that makes the credential tamper-evident
// see the NOTE at end of this section for more detail
"proof": {
  // the cryptographic signature suite that was used to generate the signature
  "type": "RsaSignature2018",
  // the date the signature was created
  "created": "2017-06-18T21:19:10Z",
  // purpose of this proof
  "proofPurpose": "assertionMethod",
  // the identifier of the public key that can verify the signature
  "verificationMethod": "https://example.edu/issuers/565049#key-1",
  // the digital signature value
  "jws": "eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0Il19..TCYt5X
    sITJX1CxPCT8yAV-TVkIEq_PbChOMqsLfRoPsnsgw5WEuts01mq-pQy7UJiN5mgRxD-WUc
    X16dUEMGlv50aqzpqh4Qktb3rk-BuQy72IFLOqV0G_zS245-kronKb78cPN25DGlcTwLtj
    PAYuNzVBAh4vGHSrQyHUdBBPM"
```

Decentralized Identifiers (DIDs)

- Globally unique URI-like scheme

- Verifiable, persistent, and does not require a central registry

- Resolves (points) do a DID Document

- Did method specifies how to CRUD the DID Document

Scheme

`did:example:123456789abcdefghi`

DID Method   DID Method-Specific Identifier


```
# Parse a did:web DID
did_string = 'did:web:identity.foundation'

did = DIDX(did_string)
=> #<DIDX::Web:0x000000010bb1ad30 @id="did:web:identity.foundation", @method="web",

# Resolve to a did:web's DID Document
did.document
=> #<DIDX::Document:0x000000010aa597d0 @assertion_method=nil, @authentication=nil,
```