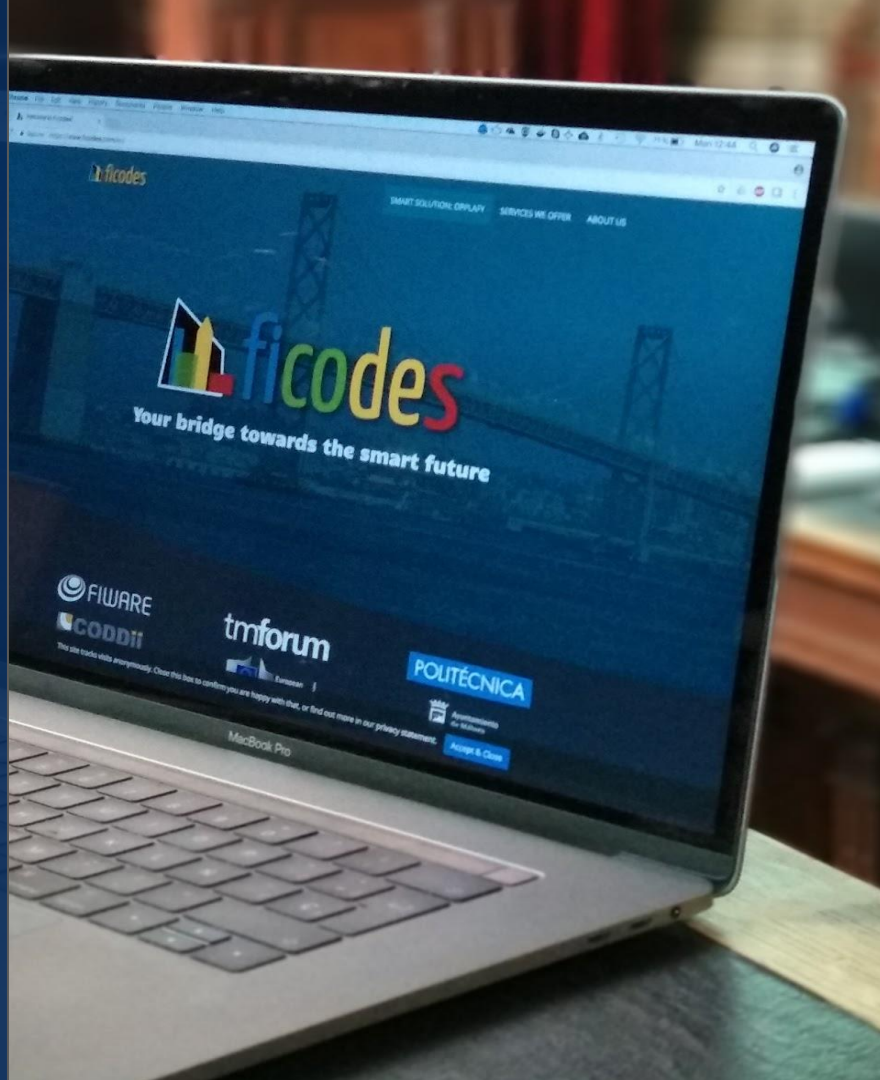
Aufbau eines dezentralen Datenraumes mit dem FIWARE Data Space Connector

Stefan Wiedemann

Senior Software Engineer FiCodes & Seamware
Member of the FIWARE TSC

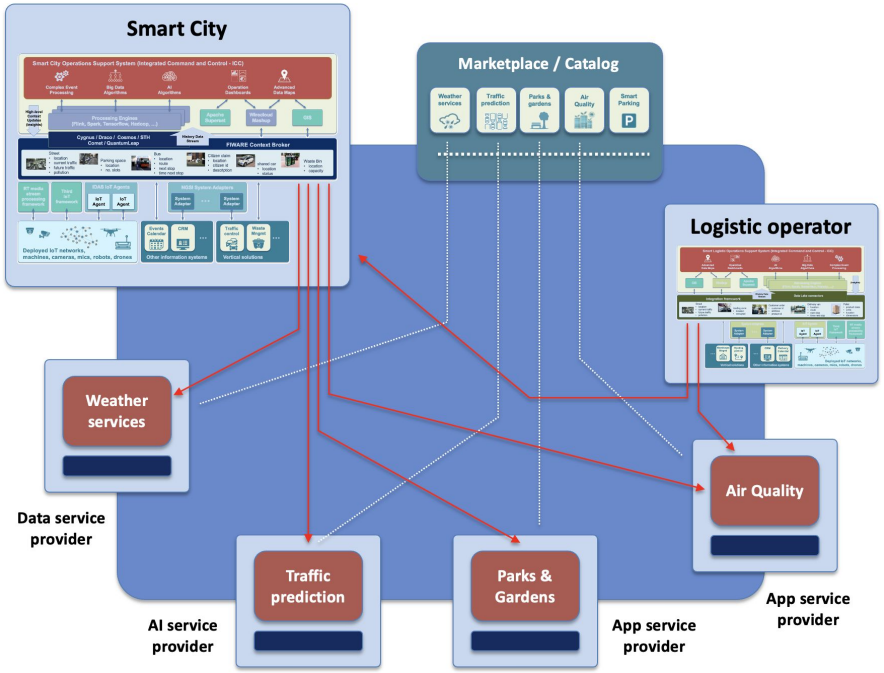
Future Internet Consulting and Development Solutions S.L.
C/ Hespérides 5, 28232, Las Rozas de Madrid, Madrid, España



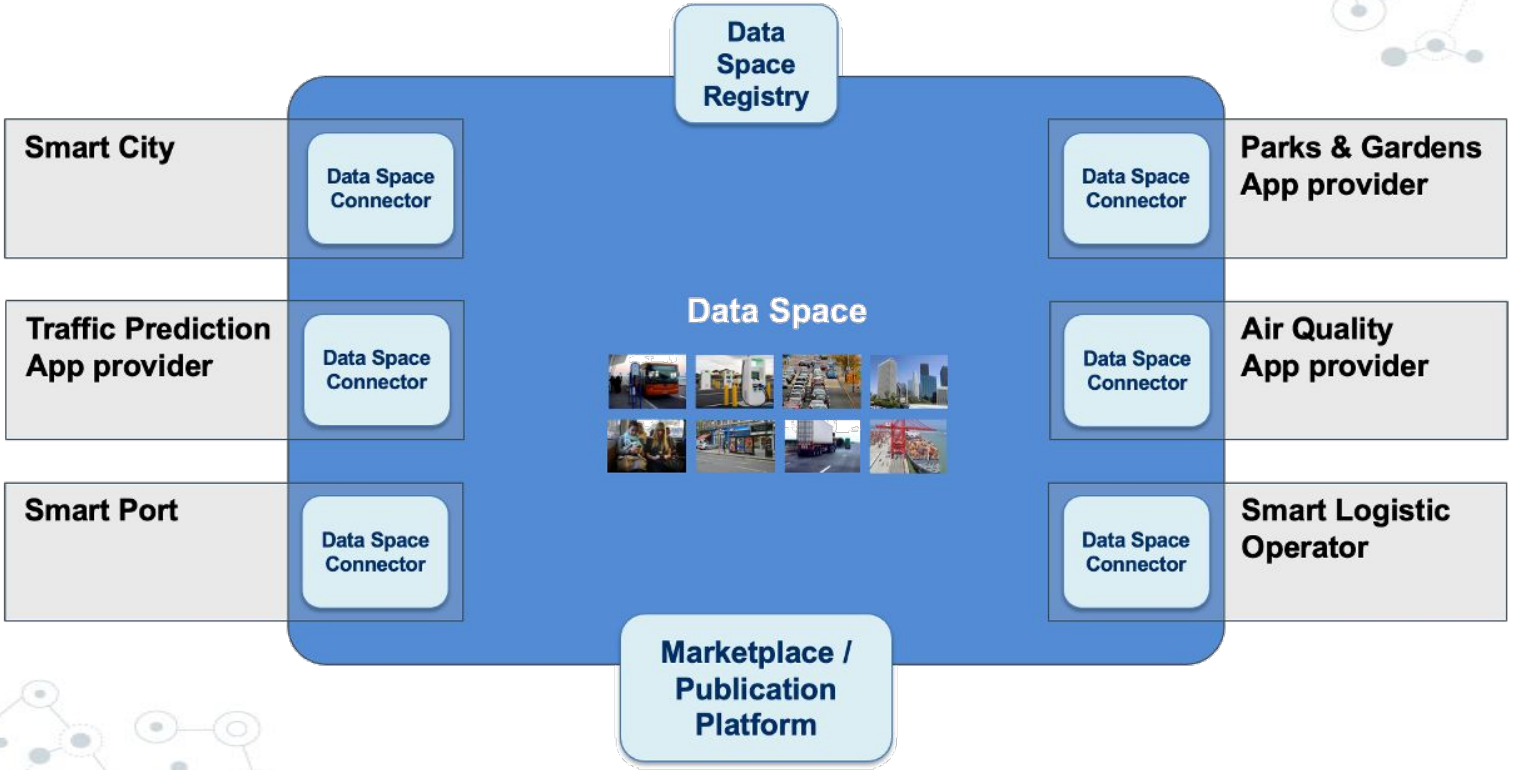
FIWARE Vision

- Datenräume sind nicht auf peer-to-peer Austausch von Daten beschränkt
- ermöglichen Organisationen:
  - nahtlose Erweiterung der Systeme durch 3rd Party Anwendungen
  - einfache Bereitstellung von Services für andere Organisationen
- jedes System bietet natürlich:
  - Daten Services (zum Zugriff auf Daten)
  - Daten Verarbeitungs Services(um Daten zu empfangen, verarbeiten und generieren)

➢ Deswegen sind Daten in Datenräumen zentral



Der Datenraum



ficodes SEAMWARE

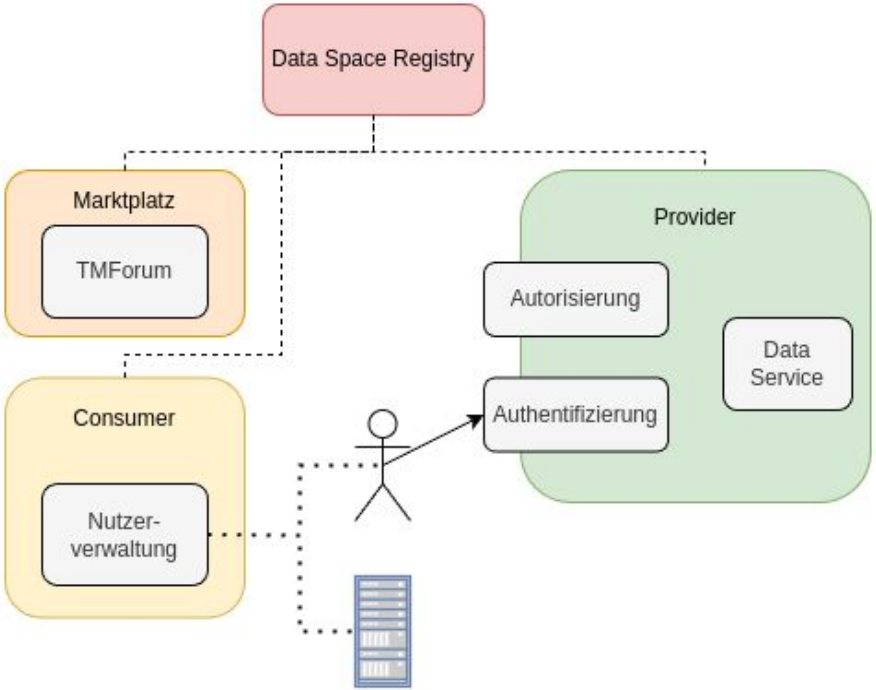
FIWARE Data Space Connector

Der FIWARE Data Space Connector besteht aus Komponenten die eine sichere Teilnahme an dezentralen Datenräumen ermöglicht.

- Trust & Identity Management basierend auf W3C DID und den OID4VC Protokollen
- Anbindung an Trust Anchor Services von EBSI, Gaia-X und Qualified Trust Providers nach eIDAS
- Autorisierungsverwaltung basierend auf ODRL
- Marktplatz-Funktionalitäten basierend auf den TMForum APIs
- Unterstützung des IDSA/Eclipse Data Space Protocol

FIWARE Data Space Connector

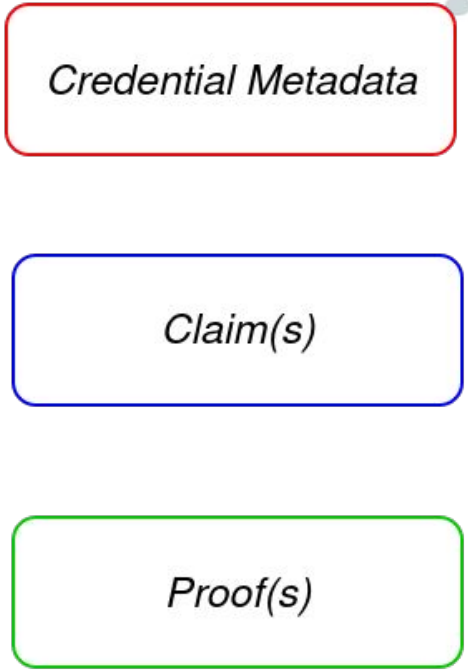
- Teilnehmer können unterschiedliche Rollen einnehmen
- unterschiedliche Services zur Erfüllung der Rollen
- Teilnehmer sind bei der Data Space Registry angemeldet
- menschliche und technische Nutzer interagieren mit den Teilnehmern



Standards

W3C Verifiable Credentials


```
{
        "@context": [
                "https://www.w3.org/2018/credentials/v1",
                "https://ec.europa.eu/eudi/v1"
        ],
        "type": ["VerifiableCredential", "NationalID"],
        "issuanceDate": "2024-01-01T12:00:00Z",
        "expirationDate": "2034-01-01T12:00:00Z",
        "issuer": "did:elsi:issuer-identifier",
        "credentialSubject": {
                "id": "did:eu:country-code:user-identifier",
                "givenName": "Maria",
                "familyName": "Musterfrau",
                "dateOfBirth": "1983-12-01",
                "role": "User"
        },
        "proof": {
                "type": "Ed25519Signature2020",
                "created": "2024-01-01T12:00:00Z",
                "proofPurpose": "assertionMethod",
                "verificationMethod": "did:elsi:issuer-identifier",
                "jws": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
        }
}
```

Dezentrale Identität

did:<method-name>:<method-specific-id><additional-components>

- erlauben dem Besitzer der Identität den Nachweis der Identität, ohne Einfluss Dritter
- verschiedene Arten von Dezentralen Identitäten:
    - did:key:zDnaeTTC781gfJKRjjShQV5sjNxX…
        - Public Key ist der Identifier, erlaubt sehr einfachen Nachweis der Identität ohne zusätzliche Services
        - häufig genutzt für Personen oder Services
    - did:web:demo-domain.org
        - basiert auf Standard Web Protokollen, kann zu einem did-Dokument aufgelöst werden
        - benötigt zusätzliche (Web) Services, bietet mehr Möglichkeiten als did:key
        - häufig genutzt für Organisationen(bspw. in Gaia-X)

DID:ELSI

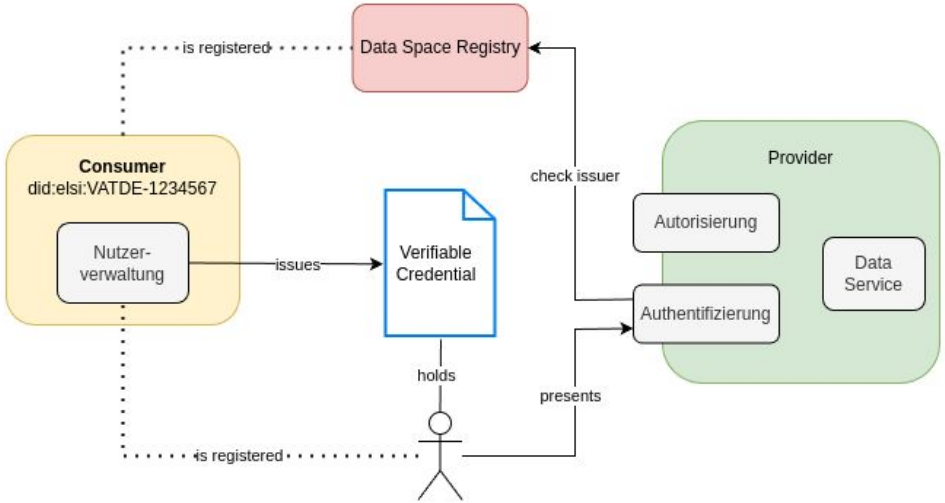
did:elsi:VATDE-309937516

- Methode für juristische Personen(e.g. Organisationen) als Brücke zwischen der eIDAS-Regulierung und Verifiable Credentials
- nutzt JAdES Signaturen für VerifiableCredentials um die eIDAS Anforderungen zu erfüllen
- Identifier basieren auf bereits vorhandenen eIDAS Zertifikaten für etablierte Organisations-Identifikationen, bspw:
    - VAT - Umsatzsteuernummer
    - LEI - Legal Entity Identifier
    - PSD - Zulassungsnummer für Zahlungsdienstleister
    - NTR - Nationale Handelsregister Identifikation

Umsetzung

Dezentrale Authentifikation

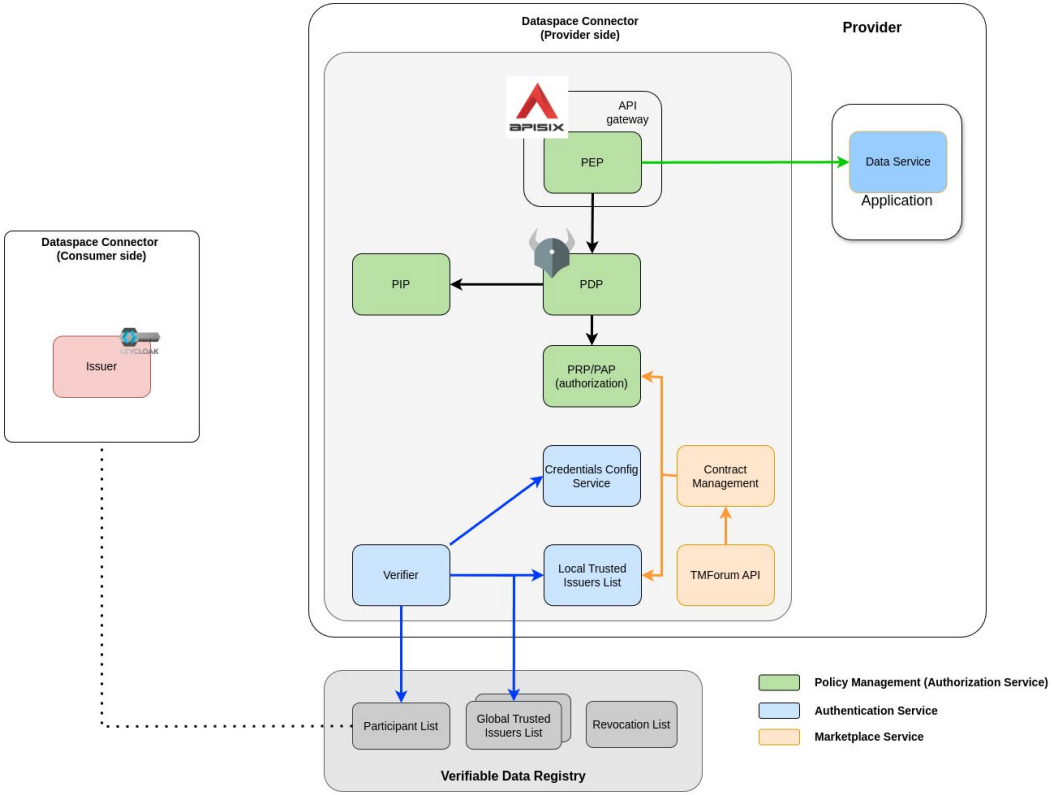
- registrierte Nutzer erhalten VerifiableCredentials
- Credentials sind signiert mit der Identität des Consumers
- der Nutzer authentifiziert sich mit dem Credential beim Provider
- der Provider prüft die Signatur und den Issuer an der Data Space Registry



Zugriffskontrolle

- Open Digital Right Language zur Spezifierung von Zugriffsberechtigungen
- erlaubt die Spezifizierung feingranularer Berechtigungen für digitale Ressourcen
- im FIWARE Data Space Connector:
    - Übersetzung der ODRL-Policies in Rego
    - Evaluation der Policies durch den Open Policy Agent

Architektur



Demo

- Aufbau eines Datenraumes mit 2 Teilnehmern und einem Trust Anchor
  - ein Teilnehmer in der Rolle “Consumer” - *Fancy Marketplace Co.*
  - ein Teilnehmer in der Rolle “Provider” mit zusätzlicher Marktplatzfunktionalität - *M&P Operations Inc.*
- Erstellen einer did:elsi für den “Consumer” und Registrierung im Datenraum
- Erstellen von VerifiableCredentials mit unterschiedlichen Zugangsrechten für registrierte Nutzer des “Consumer”
- Angebotserstellung für einen Service mit den notwendigen Zugangsberechtigungen durch den “Provider”
- Angebotsannahme und Service Nutzung durch Nutzer des “Consumer”

https://github.com/wistefan/workshop-authenticon

Links

- Slides:
    - https://github.com/wistefan/presentations
- FIWARE Data Space Connector:
    - https://github.com/FIWARE/data-space-connector
- Workshop Material:
    - https://github.com/wistefan/workshop-authenticon

contact@ficodes.com | www.ficodes.com | +34 614 20 74 47 | C/ Hespérides 5 (28232) | Las Rozas de Madrid

ficodes SEAMWARE

VC - Metadata


```
{
        "@context": [
                "https://www.w3.org/2018/credentials/v1",
                "https://ec.europa.eu/eudi/v1"
        ],
        "type": ["VerifiableCredential", "NationalID"],
        "validFrom": "2024-01-01T12:00:00Z",
        "validUntil": "2034-01-01T12:00:00Z",
        "issuer": "did:eu:country-code:issuer-identifier",
        "id": "did:example:ebfeb1f712ebc6f1c276e12ec21"
}
```


- Provides contextual and technical information about the credential itself
- Helps the verifier to understand validity, provenance and usage context

- @context: Provides the context for interpreting the data within the credential, it's often used to link schemas and standards
- type: Defines the types of the credential to help verifiers interpret its purpose
- issuer: Information about the issuing entity. It can be used to verify the document.
- optional metadata like id, names, descriptions, validFrom, validUntil or credentialStatus supported

VC - Claims


```
{
        "credentialSubject": [{
                "id": "did:eu:country-code:user-identifier",
                "givenName": "Maria",
                "familyName": "Musterfrau",
                "dateOfBirth": "1983-12-01",
                "nationality": "AU",
                "address": {
                        "street": "Kärtner Straße",
                        "city": "Wien",
                        "postalCode": "1010"

                }
        }
        {
                ...
        }]
}
```


- Section where the actual information or assertions are stored
- Property “credentialSubject” contains a set of objects, with the specific attributes or data pieces the issuer is attesting about the subject
- The claims usually should follow specific data schemas to ensure consistency and interoperability
- The subjects often contain id's, linking the claims to specific holders. It helps verifiers to confirm that a claim truly applies to the presenting subject.

VC - Proofs


```
{
        "proof": {
                "type": "Ed25519Signature2020",
                "created": "2024-01-01T12:00:00Z",
                "proofPurpose": "assertionMethod",
                "verificationMethod":
        "did:eu:country-code:issuer-identifier#keys-1",
                "jws":
        "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
        }
}
```


- The proof contains details about the digital signature on the credential
- “type” contains the proof type. It determines what other fields are required to secure and verify the proof
- “proofPurpose” contains the reason the proof was created for.
- “verificationMethod” provides the information required to verify the proof
- “jws” contains the actual digital signature, which ensures the credential has not been tampered with

Verifiable Presentations


```
{
        "@context": [... ],
        "type": ["VerifiablePresentation"],
        "verifiableCredential": [
                "@context": [ ... ],
                "type": ["VerifiableCredential", "NationalID"],
                "issuanceDate": "2024-01-01T12:00:00Z",
                "expirationDate": "2034-01-01T12:00:00Z",
                "issuer": "did:eu:country-code:issuer-identifier",
                "credentialSubject": {
                        ...
                },
                "proof": {
                        ...
                }],
        "proof": {
                "type": "Ed25519Signature2020",
                "created": "2024-01-01T12:00:00Z",
                "proofPurpose": "assertionMethod",
                "verificationMethod": "..-",
                "jws": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
        }

}
```


- A package of one or more Verifiable Credentials, created by the holder, to share specific information with a verifier
- Can combine multiple credentials, which is useful when a verifier required proof of several attributes
- The proof ensures integrity of the data and confirms that it was actually generated by the holder
- Enables the verifier to ensure a credential is provided by the actual holder, by checking the holder information in the credential with the presentations' proof