UNIVERSITÀ DI TRENTO

Master’s thesis presentation

Design and implementation of a verifiable credentials service for a data marketplace

Supervisor:
Silvio Ranise

Co-supervisors:
Giada Sciarretta
Alessandro Tomasi

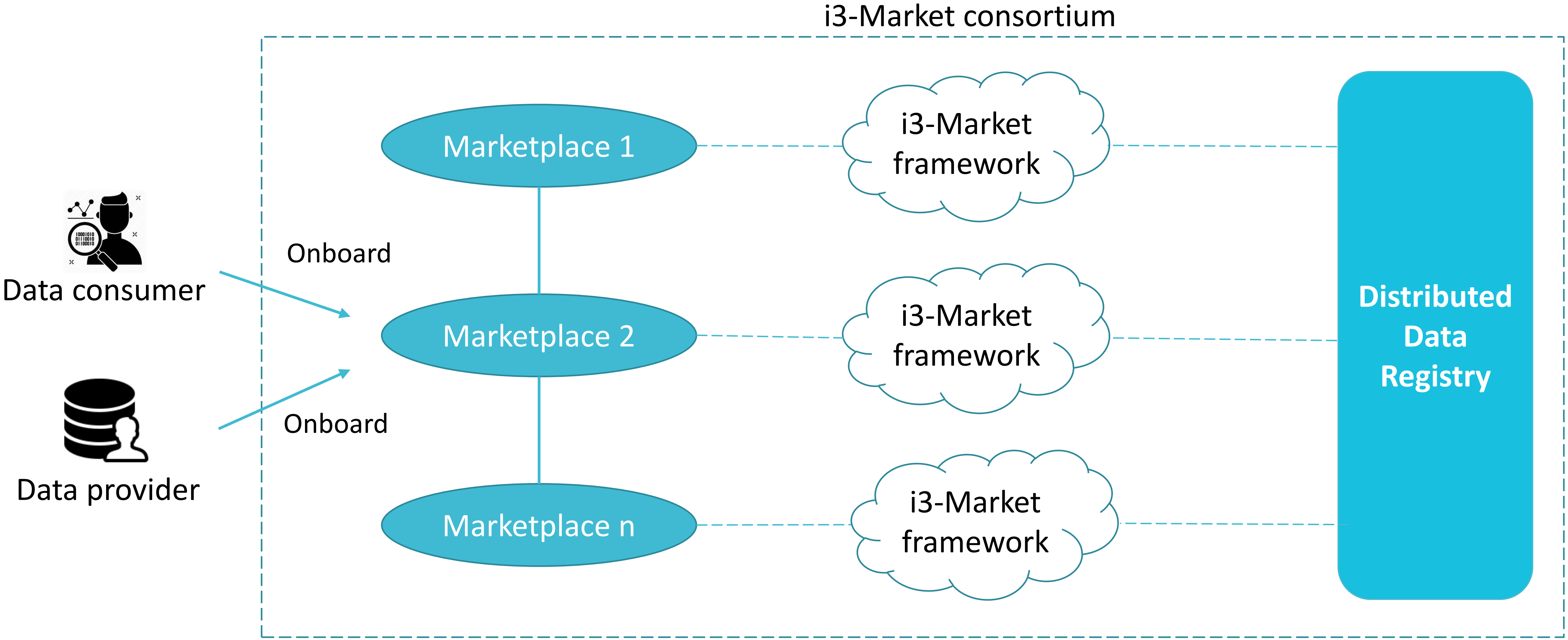
Student:
Rupert Gobber

AY. 2020/2021

Plan of the talk

- i3-Market project, www.i3-market.eu
- Challenges
- Solution design description of the verifiable credential service
- Integration of the solution with others component of i3-Market
- Conclusions and contributions to the state-of-the art

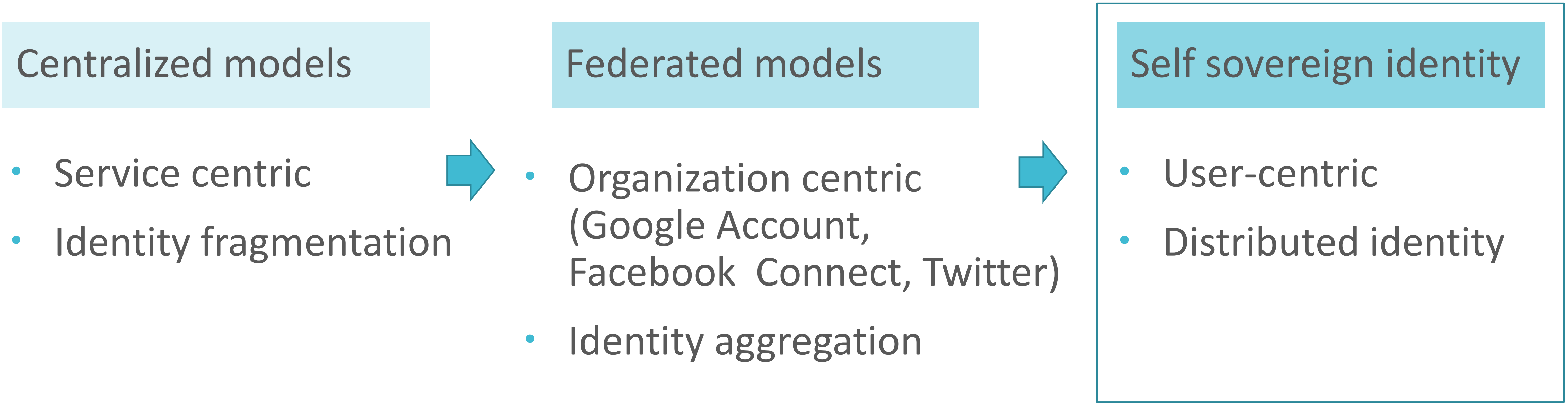
i3-Market ecosystem



Challenges

- Allow the use of a single digital identity shared among marketplaces
- Prevent the sharing of personal information across marketplaces
- Prevent redundancy of user information
- Prevent user information data breaches

Identity and access management models



Centralized models

- Service centric
- Identity fragmentation

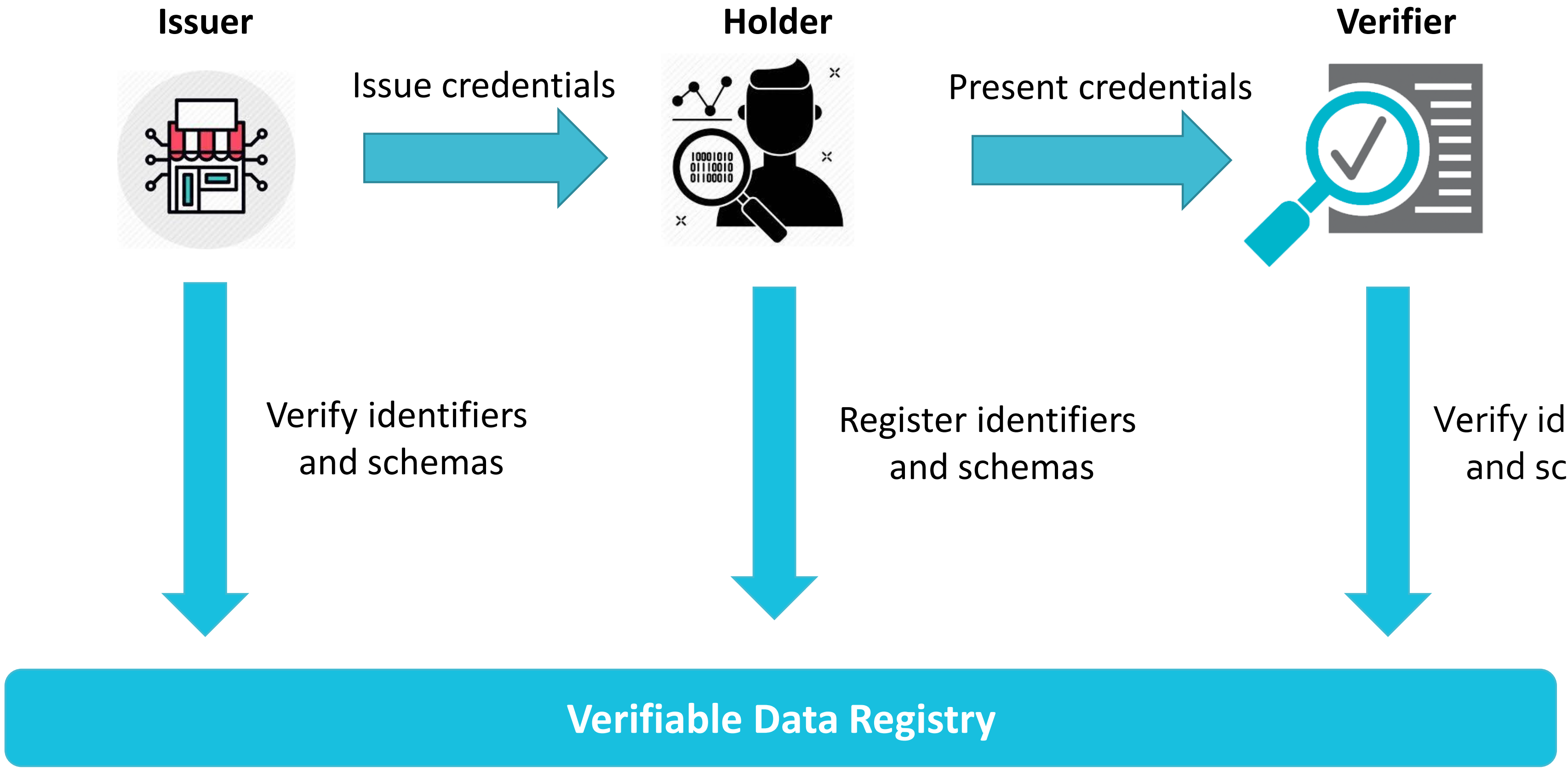
Federated models

- Organization centric (Google Account, Facebook Connect, Twitter)
- Identity aggregation

Self sovereign identity

- User-centric
- Distributed identity

Verifiable Credentials in Self-Sovereign Identity ecosystem



Issuer

Holder

Verifier

Issue credentials

Present credentials

Verify identifiers and schemas

Register identifiers and schemas

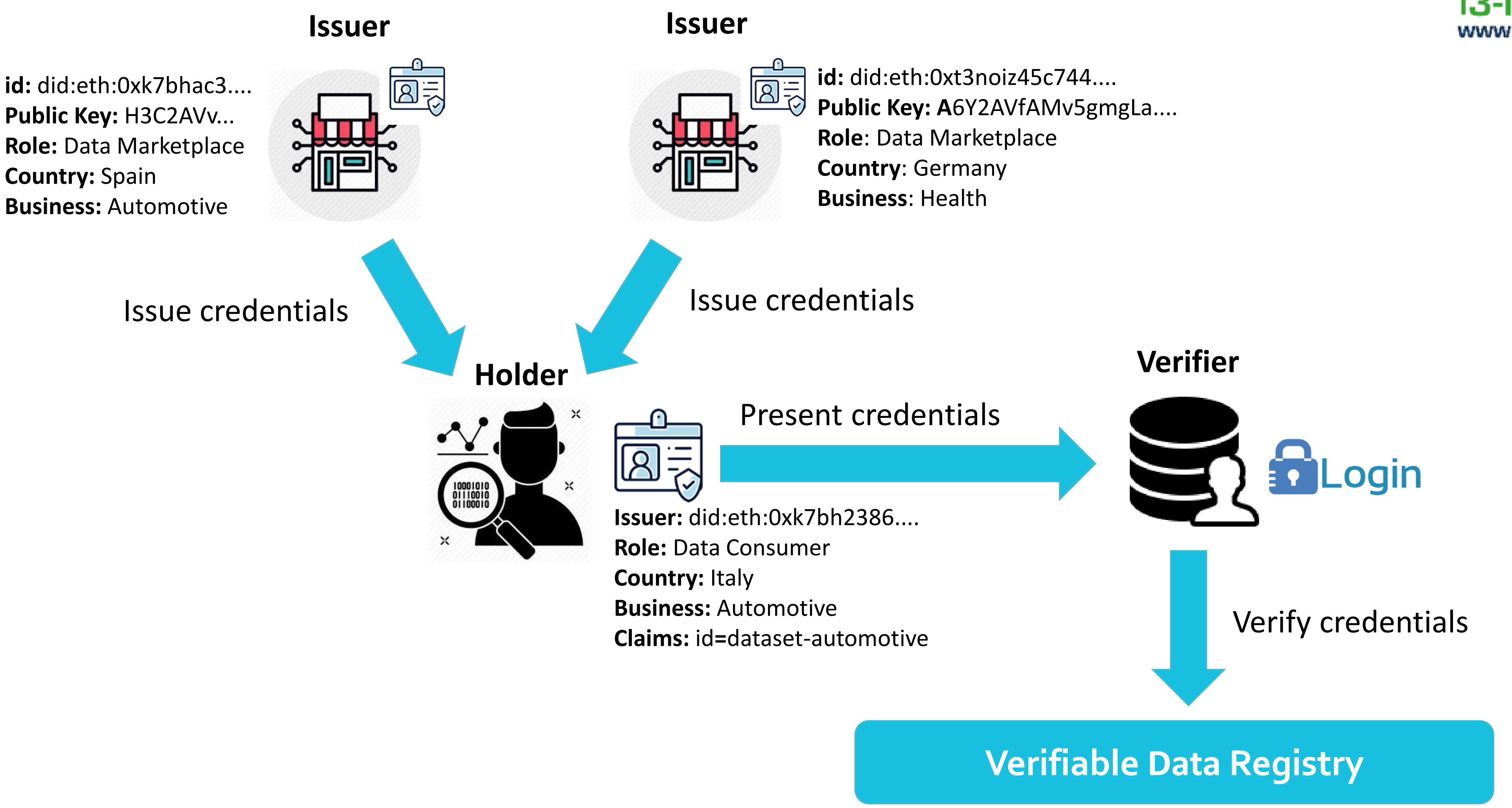
Verify identifiers and schemas

Verifiable Data Registry

Source: W3C Recommendations for Verifiable Credentials

Self-Sovereign Identity in i3-Market

i3-MARKET
www.i3-market.eu



Issuer

id: did:eth:0xk7bhac3....
Public Key: H3C2AVv...
Role: Data Marketplace
Country: Spain
Business: Automotive

Issuer

id: did:eth:0xt3noiz45c744....
Public Key: A6Y2AVfAMv5gmgLa....
Role: Data Marketplace
Country: Germany
Business: Health

Issue credentials

Issue credentials

Holder

Issuer: did:eth:0xk7bh2386....
Role: Data Consumer
Country: Italy
Business: Automotive
Claims: id=dataset-automotive

Present credentials

Verifier

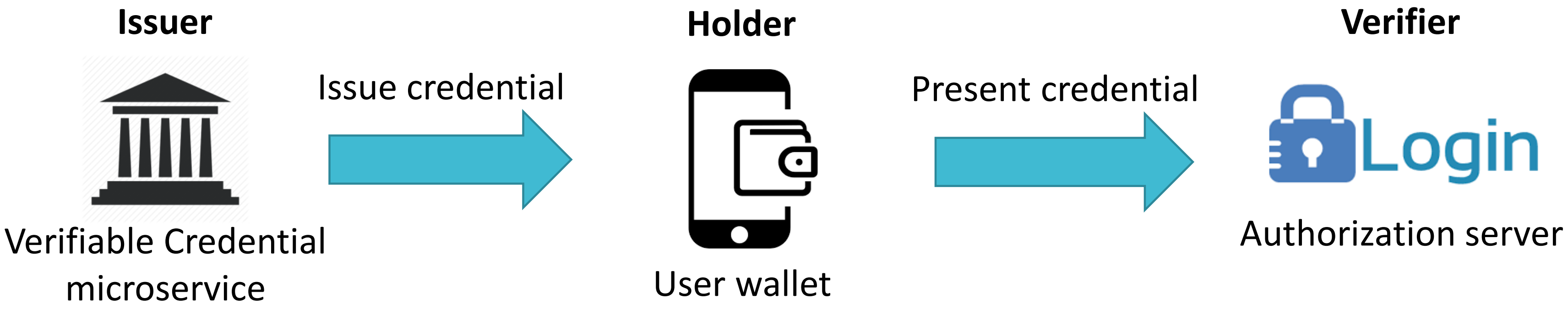
Login

Verify credentials

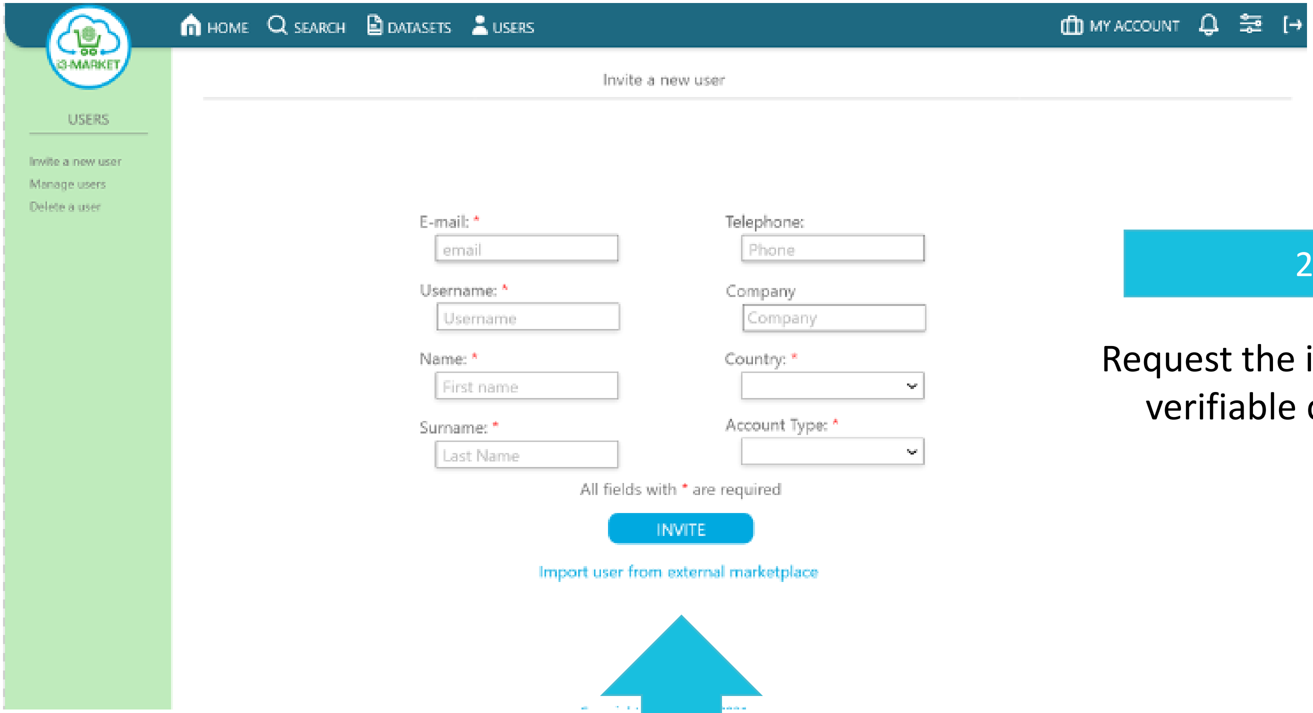
Verifiable Data Registry

Thesis contribution – Verifiable Credential microservice

- Development of a microservice for managing the «Issuer» actor of SSI ecosystem:
  - Issuance of Verifiable Credentials (VC)
  - Verification of VC
  - Revocation of VC
- Integration with an Open ID Connect identity provider for authentication and authorization through verifiable credentials («Verifier» actor of SSI)
- Integration with the user wallet («Holder» actor of SSI)



Credential issuance (Registration use case)



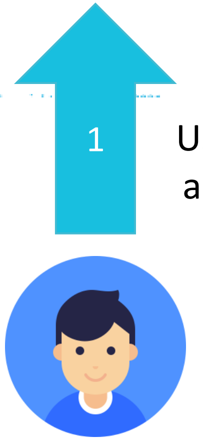
Invite a new user

E-mail: * email	Telephone: Phone
Username: * Username	Company Company
Name: * First name	Country: *
Surname: * Last Name	Account Type: *

All fields with * are required

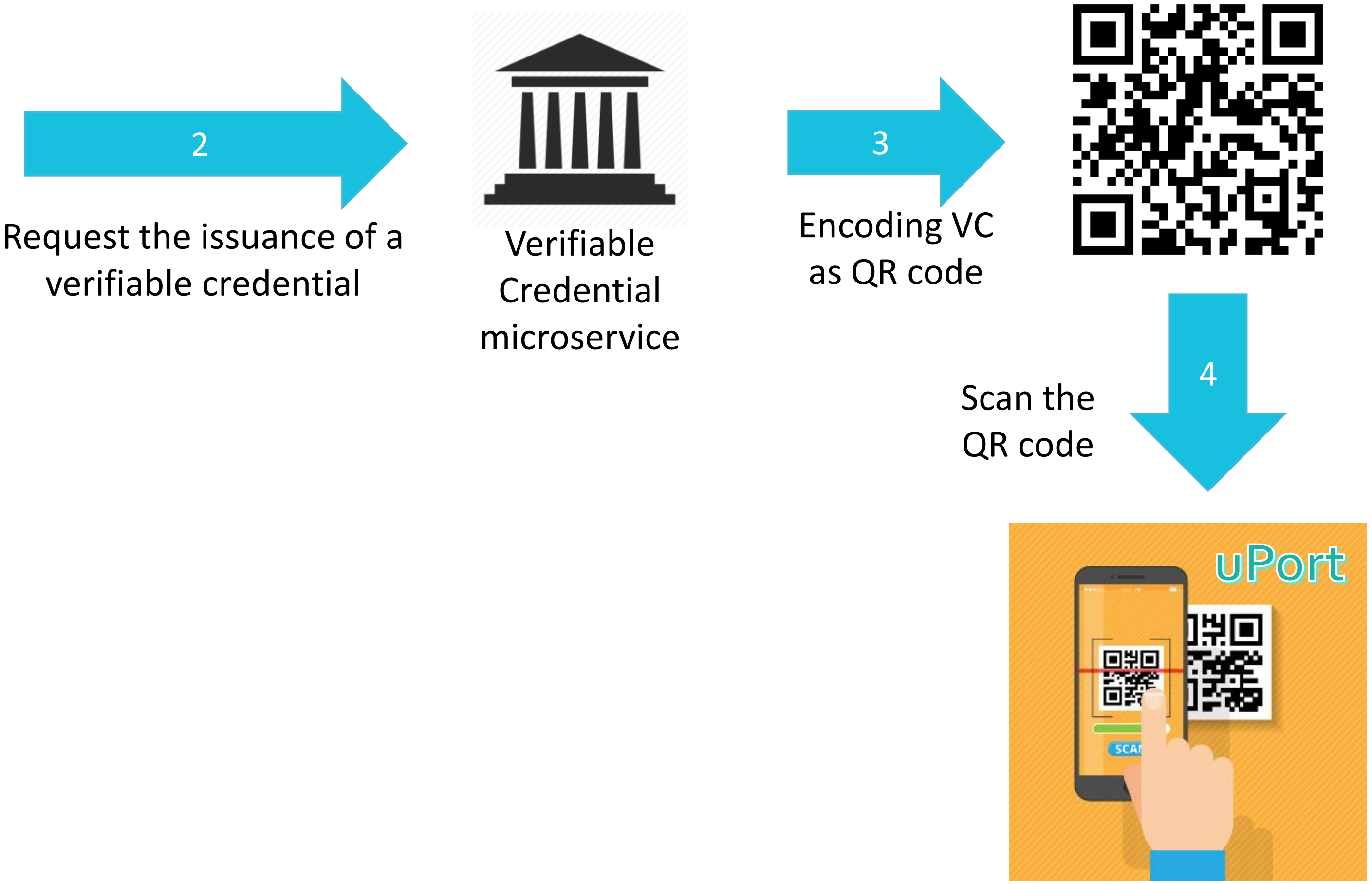
INVITE

Import user from external marketplace



1 User sign-up in a marketplace

Data consumer / Data provider



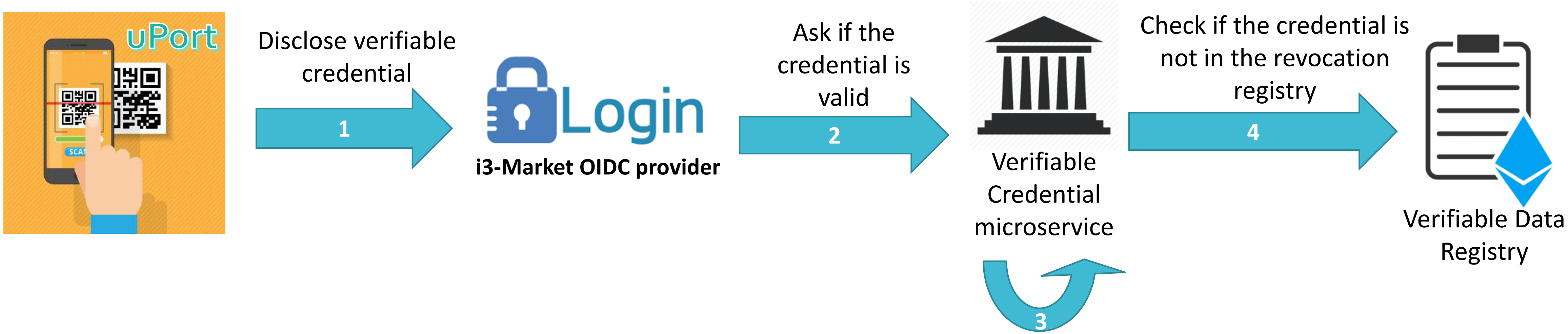
2 Request the issuance of a verifiable credential

Verifiable Credential microservice

3 Encoding VC as QR code

4 Scan the QR code

Credential verification



1 Disclose verifiable credential

i3-Market OIDC provider

2 Ask if the credential is valid

Verifiable Credential microservice

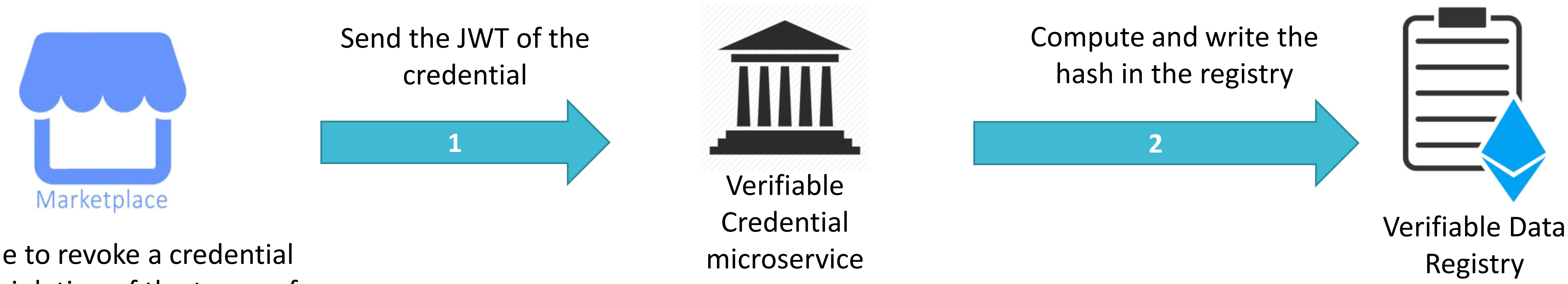
3

4 Check if the credential is not in the revocation registry

Verifiable Data Registry

- Check signature
- Check if is not expired
- Check if the issuer is trusted

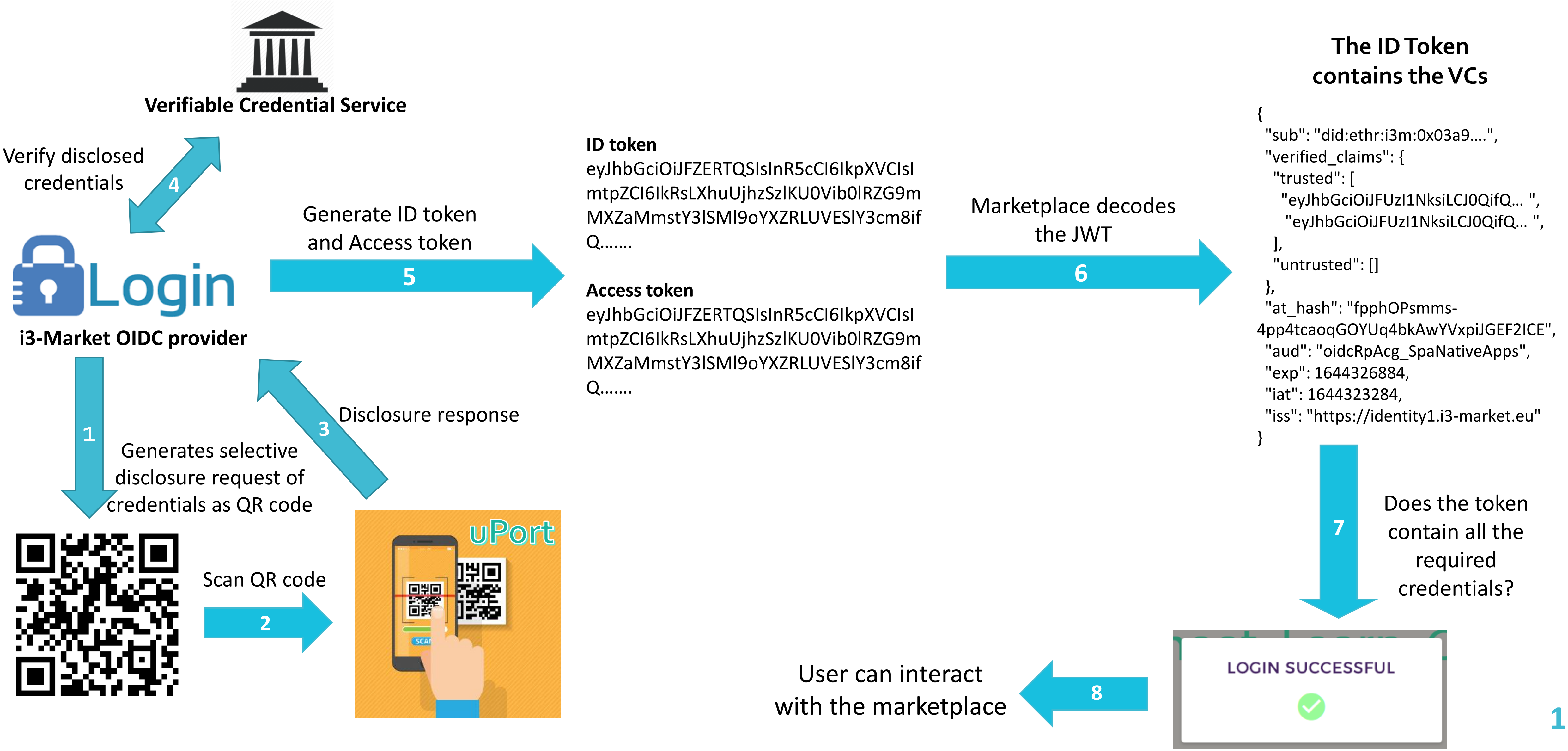
Credential revocation


Marketplace
Send the JWT of the credential
1
Verifiable Credential microservice
Compute and write the hash in the registry
2
Verifiable Data Registry

Decide to revoke a credential (e.g., violation of the terms of use, dismissing service relate to that credential, …)

- Once a credential is marked as revoked, it can no longer be trusted

Integration with the Open ID Connect identity provider



Verifiable Credential Service

4 Verify disclosed credentials

i3-Market OIDC provider

1 Generates selective disclosure request of credentials as QR code

2 Scan QR code

3 Disclosure response

5 Generate ID token and Access token

ID token
eyJhbGciOiJFZERTQSIsInR5cCI6IkpXVCIsI
mtpZCI6IkRsLXhuUjhzSzlKU0Vib0lRZG9m
MXZaMmstY3lSMl9oYXZRLUVESlY3cm8if
Q…….

Access token
eyJhbGciOiJFZERTQSIsInR5cCI6IkpXVCIsI
mtpZCI6IkRsLXhuUjhzSzlKU0Vib0lRZG9m
MXZaMmstY3lSMl9oYXZRLUVESlY3cm8if
Q…….

6 Marketplace decodes the JWT

The ID Token contains the VCs

```
{
  "sub": "did:ethr:i3m:0x03a9….",
  "verified_claims": {
   "trusted": [
     "eyJhbGciOiJFUzI1NksiLCJ0QifQ… ",
      "eyJhbGciOiJFUzI1NksiLCJ0QifQ… ",
   ],
   "untrusted": []
  },
  "at_hash": "fpphOPsmms-
4pp4tcaoqGOYUq4bkAwYVxpiJGEF2ICE",
  "aud": "oidcRpAcg_SpaNativeApps",
  "exp": 1644326884,
  "iat": 1644323284,
  "iss": "https://identity1.i3-market.eu"
}
```

7 Does the token contain all the required credentials?

8 User can interact with the marketplace

Results & Contributions to the state-of-the-art

- There are no such open source components in the state of the art for the “Issuer” (Verifiable Credential microservice) and the “Verifier” (OIDC provider). However, there are multiple frameworks to implement them
- As parts of H2020 funding research program, the developed components of the i3-Market project will be released as open source software in a public Github organization repository
- For this actors, it is possible to integrate the communication with other wallets that follow the W3C recommendation for verifiable credentials. The supported wallet for i3-Market IAM are:
    - uPort mobile wallet application
    - i3-Market desktop wallet application, using the Veramo framework

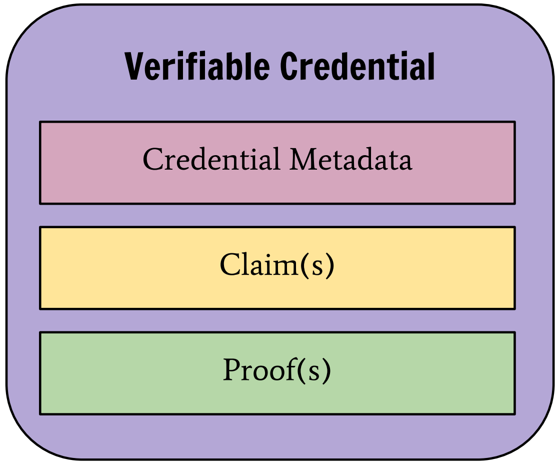
Thanks for your attention

References

- I3Market project, https://www.i3-market.eu/
- Verifiable Credentials Data Model v1.1, https://www.w3.org/TR/vc-data-model/
- Decentralized Identifiers (DIDs) v1.0, https://www.w3.org/TR/did-core/
- uPort, https://developer.uport.me/#platform
- Veramo, https://veramo.io/

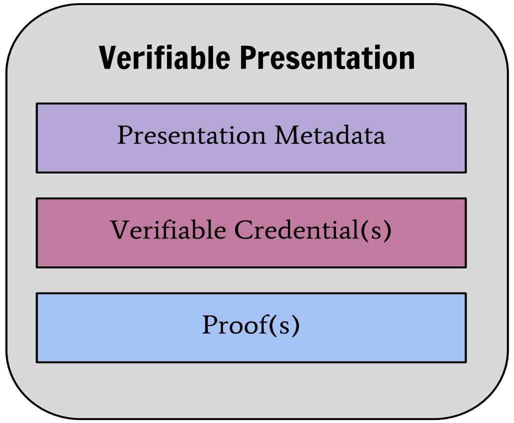
Appendix 1: Verifiable Credential Data Model

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1"
  ],
  "id": "http://example.edu/credentials/1872",
  "type": ["VerifiableCredential", "AlumniCredential"],
  "issuer": "https://example.edu/issuers/565049",
  "issuanceDate": "2010-01-01T19:23:24Z",
  "credentialSubject": {
    "id": "did:example:ebfeb1f712ebc6f1c276e12ec21",
    "alumniOf": {
     "id": "did:example:c276e12ec21ebfeb1f712ebc6f1",
     "name": [{
       "value": "Example University",
       "lang": "en«
     }, {
       "value": "Exemple d'Université",
       "lang": "fr"
    }]
   }
 },
 ....
```



```
....
"proof": {
  "type": "RsaSignature2018",
  "created": "2017-06-18T21:19:10Z",
  "proofPurpose": "assertionMethod",
  "verificationMethod":
"https://example.edu/issuers/565049/keys/1",
  "jws":
"eyJhbGciOiJSUzI1NiIsImI6ZmFsc2UsImNyaXQiOlsiYjY0Il19.. «
}
```

Appendix 2: Verifiable Presentation Data Model



Verifiable Presentation

Presentation Metadata

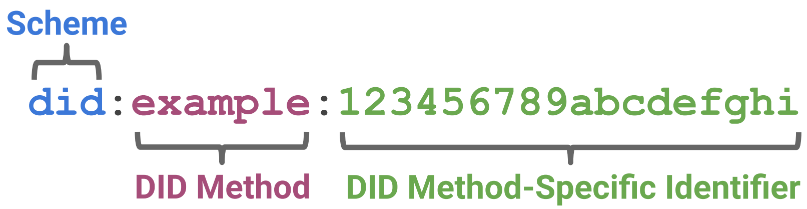
Verifiable Credential(s)

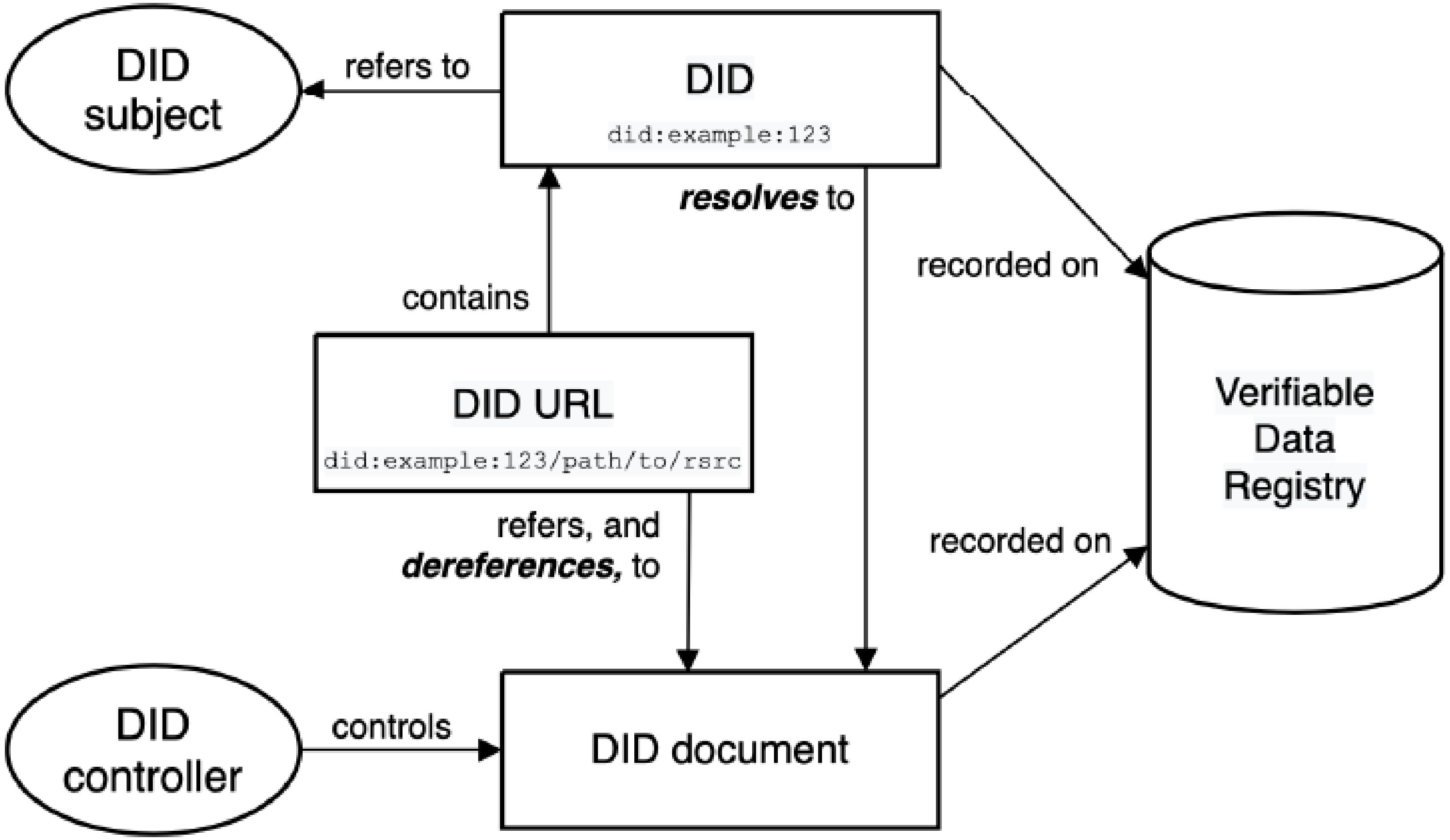
Proof(s)

```
{
 "@context": ["https://www.w3.org/2018/credentials/v1"],
 "type": "VerifiablePresentation",
 "verifiableCredential": [{
     "@context": ["https://www.w3.org/2018/credentials/v1"],
     "id": "http://example.edu/credentials/1872",
     "type": ["VerifiableCredential", "AlumniCredential"],
     "issuer": "https://example.edu/issuers/565049",
     "issuanceDate": "2010-01-01T19:23:24Z",
     "credentialSubject": {
        "id": "did:example:ebfeb1f712ebc6f1c276e12ec21",
        "alumniOf": {
           "id": "did:example:c276e12ec21ebfeb1f712ebc6f1",
           "name": [{
              "value": "Example University",
              "lang": "en"
           }, {
              "value": "Exemple d'Université",
              "lang": "fr"
           }]
        }
     },
  ....
```

```
....
 "proof": {
    "type": "RsaSignature2018",
    "created": "2017-06-18T21:19:10Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod": "https://example.edu/issuers/565049/keys/1",
    "jws": "eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0.. "
  }
}, { … } ],
"proof": {
 "type": "RsaSignature2018",
 "created": "2018-09-14T21:19:10Z",
 "proofPurpose": "authentication",
 "verificationMethod": "did:example:ebfeb1f712ebc6f1c276e1221#keys-1",
 "challenge": "1f44d55f-f161-4938-a659-f8026467f126",
 "domain": "4jt78h47fh47",
 "jws": "eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0Il19.. "
}
}
```

Appendix 3: Decentralized Identifiers (DIDs)


Scheme
did:example:123456789abcdefghi
DID Method
DID Method-Specific Identifier

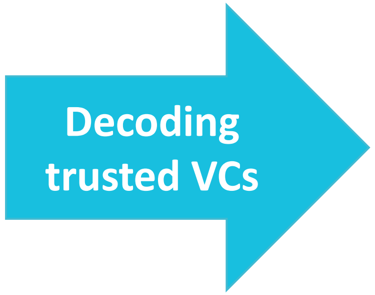


Appendix 4: Decoding ID tokens

Example of ID token decoded

```
{
  "sub":
"did:ethr:i3m:0x03a98f6055a418be6d73b…",
  "verified_claims": {
    "trusted": [
      "eyJhbGciOiJFUzI1NksiLCJ0eXAiOiJKV1QifQ….."
    ],
    "untrusted": []
  },
  "at_hash": "fpphOPsmm-4pp4tcaoqGxp…",
  "aud": "oidcRpAcg_SpaNativeApps",
  "exp": 1644326884,
  "iat": 1644323284,
  "iss": "https://identity1.i3-market.eu"
}
```

The marketplace decodes the array of trusted credentials, in order to see claims about the subject



Example of trusted Verifiable Credential decoded

```
{
  "vc": {
     "credentialSubject": {
       "role": "provider",
       "profile": { "name": ….., "surname": …..,  }
     },
     "@context": [ https://www.w3.org/2018/credentials/v1 ],
     "type": [ "VerifiableCredential" ]
  },
  "sub": "did:ethr:i3m:0x03a98f6055a418be6d73b6caf6…..",
  "nbf": 1637597151,
  "iss": "did:ethr:i3m:0x2e3592788eb9154914f87e4bb820110…."
}
```

NB 2:

- "iss" in this case the issuer is the instance of the VC service that had generated the credential

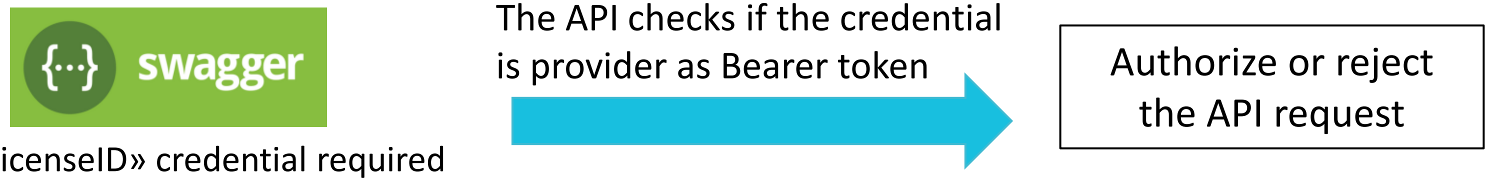
NB:

- "trusted" is the array of VCs disclosed via OIDC provider and i3Market wallet
- "iss" is the issuer of the token, i.e., the OIDC provider that generate it (in this case the instance running on the identity1 server)
- "sub" is the subject of the token, i.e., the wallet identity that disclose the credentials (could be a data provider or data consumer or any entity that own an i3Market identity and had disclosed the credential)

Appendix 5: API Authorization

Components of i3-Market framework may need information from verifiable credentials

- Case 1: an API needs a specific credential to be triggered

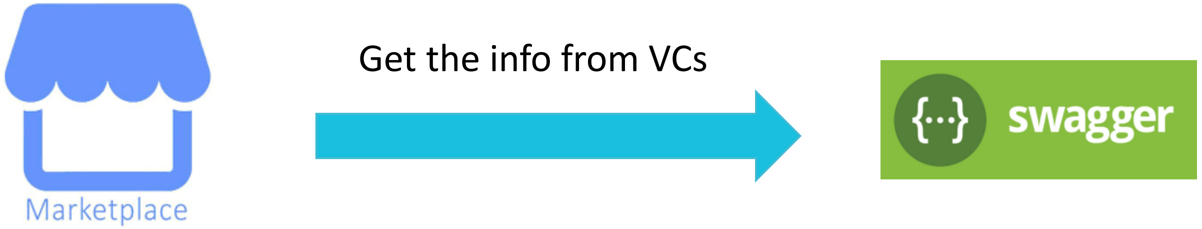


E.g., «LicenseID» credential required

The API checks if the credential is provider as Bearer token

Authorize or reject the API request

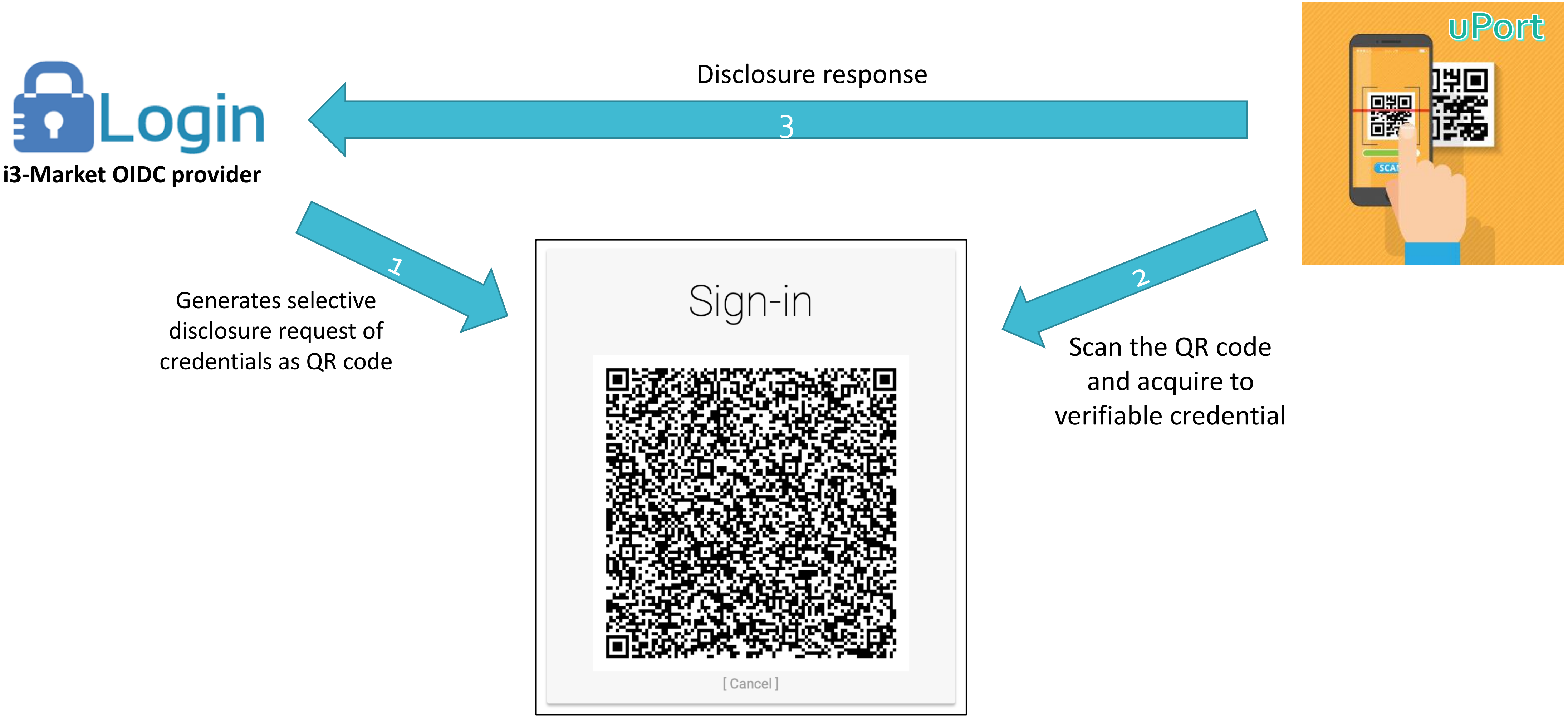
- Case 2: an API is not protected and takes as input information provided via VCs



Get the info from VCs

The marketplace decodes the VCs and use the information for API requests

Appendix 6: Open ID Connect Provider authentication GUI



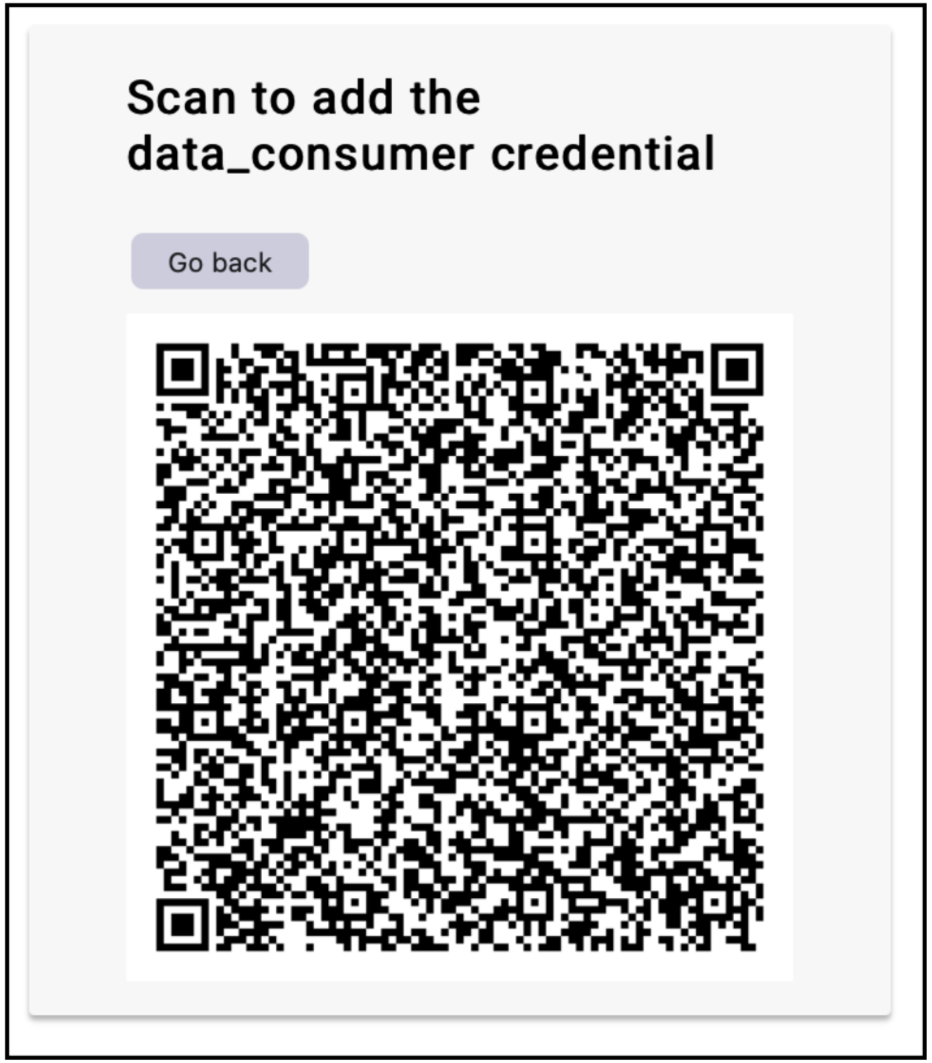
i3-Market OIDC provider

Disclosure response

Generates selective disclosure request of credentials as QR code

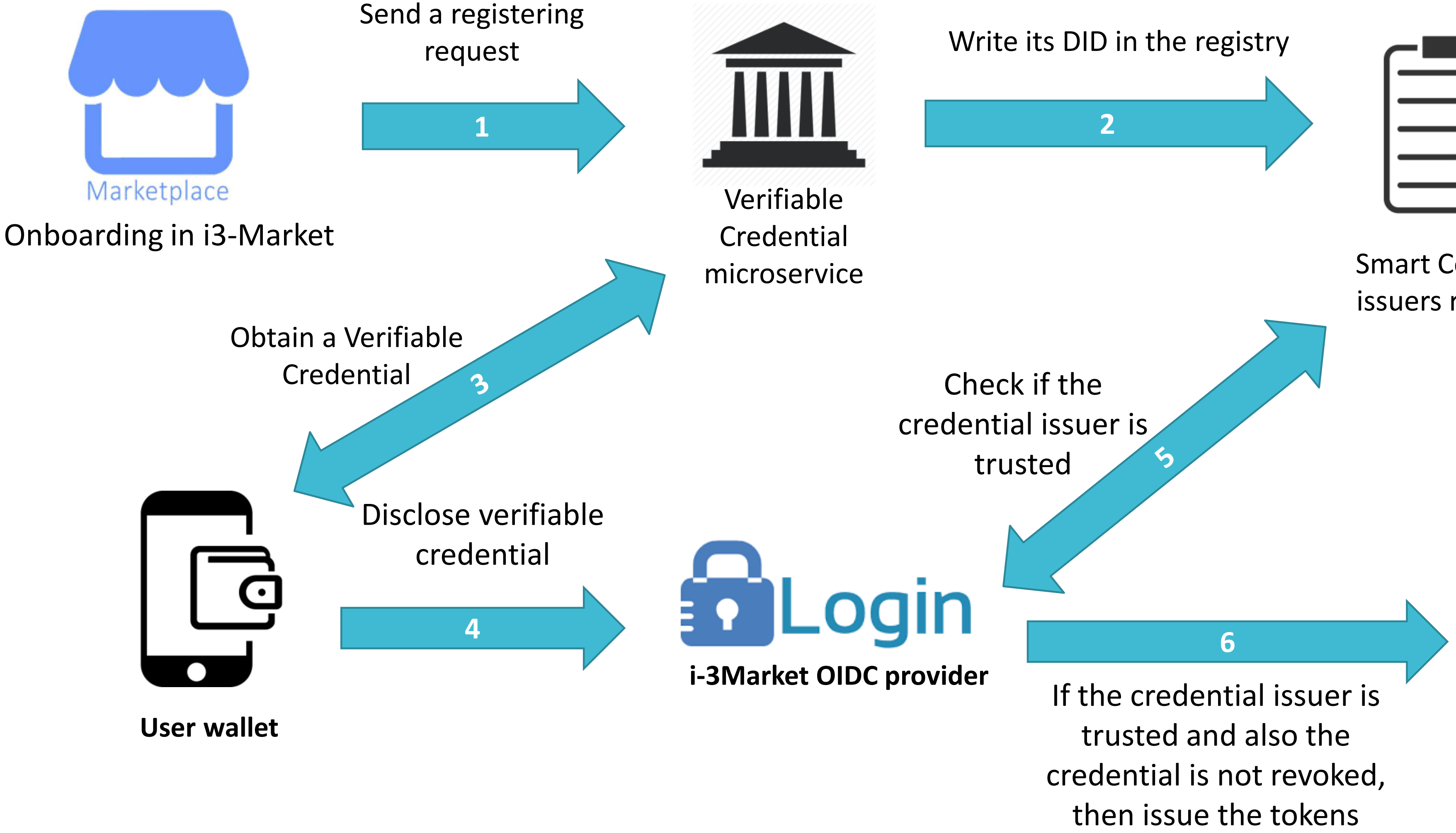
Scan the QR code and acquire to verifiable credential

Appendix 7: Verifiable Credential microservice GUI



Scan to add the data_consumer credential

Appendix 8: Trusted Issuers management



Marketplace

Onboarding in i3-Market

Send a registering request

1

Verifiable Credential microservice

Write its DID in the registry

2

Smart Contract issuers registry

Obtain a Verifiable Credential

3

User wallet

Disclose verifiable credential

4

Login

i-3Market OIDC provider

Check if the credential issuer is trusted

5

6

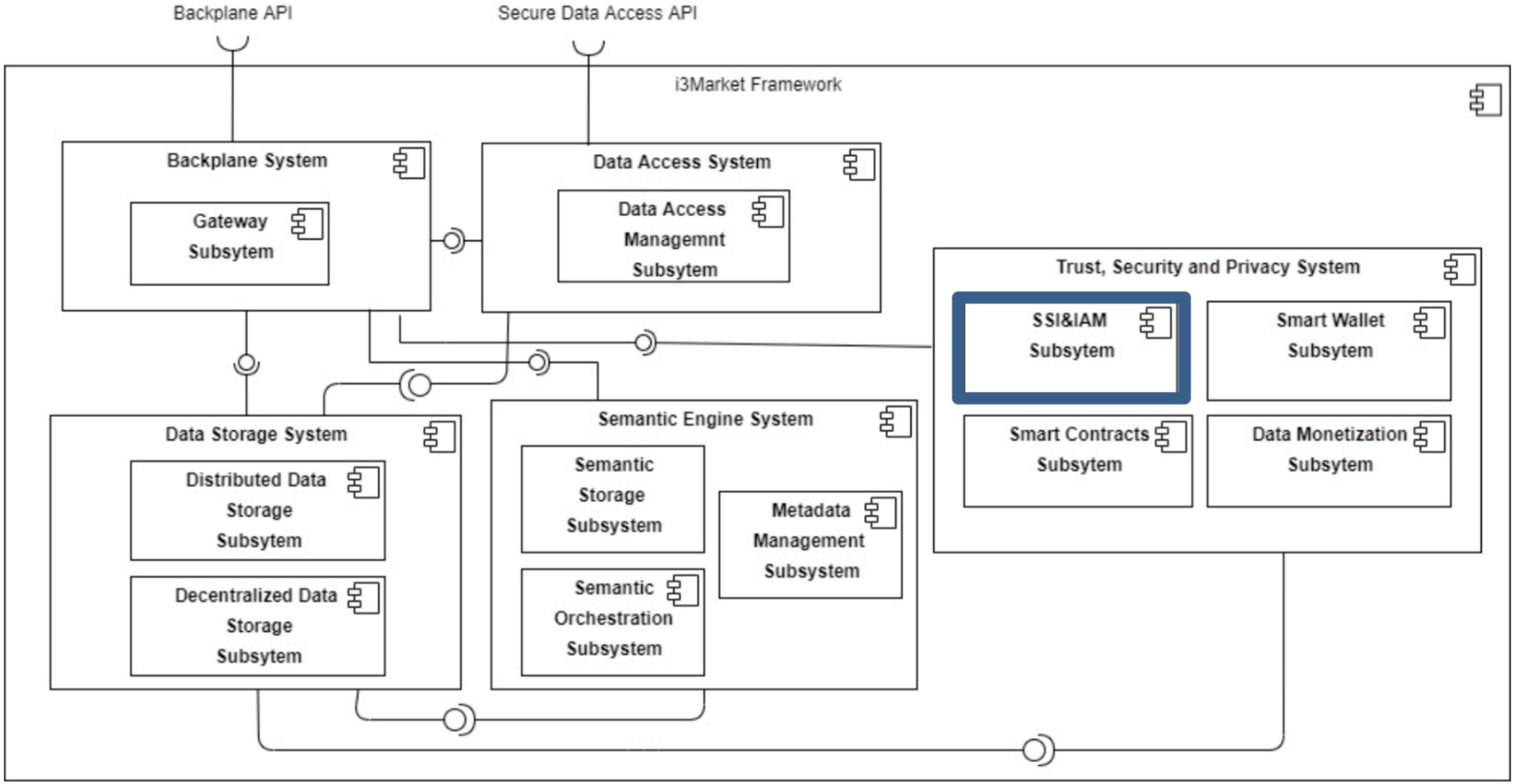
If the credential issuer is trusted and also the credential is not revoked, then issue the tokens

Once a issuer is marked as trusted in the registry, their credentials are considered trusted within i3-Market

ID token
eyJhbGciOiJFZERTQSIsInR5cCI6IkpXVCIsI
mtpZCI6IkRsLXhuUjhzSzlKU0Vib0lRZG9m
MXZaMmstY3lSMl9oYXZRLUVESlY3cm8if
Q.......

Access token
eyJhbGciOiJFZERTQSIsInR5cCI6IkpXVCIsI
mtpZCI6IkRsLXhuUjhzSzlKU0Vib0lRZG9m
MXZaMmstY3lSMl9oYXZRLUVESlY3cm8if
Q.......

Appendix 9: i3-Market Framework reference architecture


Backplane API
Secure Data Access API
i3Market Framework
Backplane System
Gateway Subsytem
Data Access System
Data Access Managemnt Subsytem
Trust, Security and Privacy System
SSI&IAM Subsytem
Smart Wallet Subsytem
Smart Contracts Subsytem
Data Monetization Subsytem
Data Storage System
Distributed Data Storage Subsytem
Decentralized Data Storage Subsytem
Semantic Engine System
Semantic Storage Subsystem
Metadata Management Subsystem
Semantic Orchestration Subsystem