Output

1. DLT evaluation

The general suitability evaluation of DLT for this use case states: DLT suitable

More subjective fitting criteria indicate for DLT: Likely use case

2. DLT elicitation

After detailed elicitation of requirements, the following recommendations for most appropriate, also viable and least recommended DLT type are:

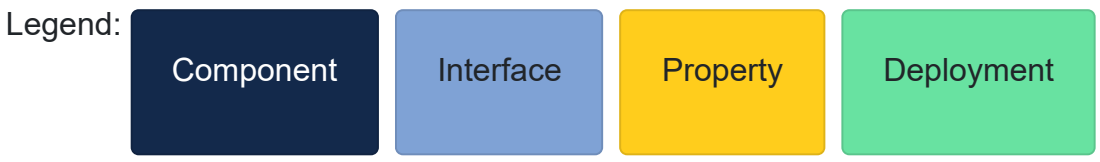
- Most appropriate: hybrid DLT
- Also viable: public / hybrid DLT
- Not recommended: private DLT

3. DLT design

After re-evaluating the DLT type in the DLT design stage, the tool concludes the following DLT type as best fit: hybrid / private DLT

The following design patterns are recommended for the use case:

- On-/off-chain connection
  - Off-chain signature pattern
  - Content addressable storage pattern
  - Delegated computation pattern
- Encryption
  - Hash on-chain and raw data in external storage (database or decentralized content addressable storage)



Central backend service

Own full DLT node

3rd party DLT node service

Key management / wallets

3rd party key management service

Central backend service

Companies

Stability / standardization of business logic

Stable

Token

not required

Database

required

File system

required

Infrastructure layer

Network layer

Deployment

Frontend

On-premise

(Private / hybrid) DLT component / node

On-premise

Remaining system (central backend service, IAM, logging, monitoring)

On-premise

Governance

hybrid / private DLT

Consensus

Recommended algorithm

Processing layer

DLT Business logic execution

Smart contracts

Usage

Security (e.g. verification, identity management, access control), Real-life goods (e.g. transfer of ownership, trade of goods and data, proof of existence, tracking of goods), Ledger (e.g. data storage, accounting, digital rights and IP management)

Non-DLT Business logic execution

Storage layer

On-chain-data

Data type

Reporting data

Data source

User input, System itself

Encryption

not required

Off-chain data

Central file system

Data type

Transactional data

Encryption

required

PoA

Recommended algorithm

PBFT

Recommended algorithm

IBFT

Recommended algorithm

Raft

Scalability

Consistency

Strict consistency

Throughput

tps > 1k important