I-DELTA

Interoperable Distributed Ledger Technology

Deliverable 2.2-2.3

Use Cases and Requirements Definition

Deliverable type:	**Document**
Deliverable reference number:	**ITEA| D2.2-D2.3**
Related Work Package:	**WP2**
Due date:	**2020-10-28**
Actual submission date:	**2020-10-15**

Responsible organisation:	
Editor:	
Dissemination level:	Public
Revision:	Final | Version 2.0

Abstract:	
Keywords:	

Table_head	Name 1 (partner)	Name 2 (partner)	Approval date (1 / 2)
Approval at WP level			
Approval at PCC level			

Editor

Contributors

Table of Content

1. Introduction

This document details the requirements of the I-DELTA project use cases. There are three use cases included in the project: the first Turkish use case, the Energy Wallet use case made up of Turkey, Canada, and Spain, and finally, the Czech Republic use case.

As it is a large project and with so many different partners, it is essential to define general requirements that all parties have to adhere to them. These general requirements of I-DELTA are group into framework requirements and architecture requirements.

Once the general requirements are defined, it is necessary to define also the requirements for each of the three use cases. In the Energy Wallet use case, besides defining the shared requirements for the partners in the three countries, it is necessary to establish the specific ones for each country.

With this, all the requirements of the project's use cases are framed, to which the partners must adapt.

2. Use case descriptions

2.1. Turkey, Loyalty Use Case

In today’s ever increasing mobile work environment, businesses continue to find it challenging to maintain high levels of employee satisfaction. One of the tools they have at their disposal are loyalty programs, through which they give extra benefits to their employees in addition to their regular salary. These benefits create tax advantages and increase the loyalty of their employees. However, some of the benefits given to employees are not utilized and are wasted. Based on recent studies, 95% of employees prefer to select their own benefits instead of having their employers allocate benefits for them (Liazon). Furthermore, 80% of organizations state that employees have low knowledge about the benefits that are provided (IFEBP). Companies therefore want to maximize the effectiveness of the benefits of their employees without wasting unnecessary funds. A single enterprise by itself does not offer enough of a variety of benefits to maintain employee satisfaction. Furthermore, making these benefits exchangeable between firms is not easy. Using digital ledgers (DLTs) and blockchains as a bridge across companies will aid in establishing a baseline of trust in the loyalty network. Additionally, by deploying their benefits program on the I-DELTA platform, companies will be able to give their employees access to a diverse and wide-ranging benefit pool where they can choose their benefits transparently instead of having a limited set of benefity involuntarily allocated to them by default.

The objectives are as follows:

O1) The benefits given to employees must be exchangeable, transferrable and distributed between companies.

O2) The system will maximize the use of benefits across companies, increasing employee satisfaction and reducing the waste of unused and involuntary benefits.

O3) Every benefit that is offered in the system will be available to the entire employee network spanning across companies.

Technical foundations supporting the use case:

T1) A type of loyalty token crypto currency to make the exchange of benefits easier and more secure will be created by each company. By using a DLT, digital assets and loyalty tokens can be stored and exchanged.

T2) A Digital Identity that we will call Employee Identity will hold the ownership of the assets and tokens on the DLT. People who have Employee Identity on their personal devices will manage all their assets.

2.1.1. Context Scenario - 1

Goal: Initializing the loyalty network by distributing tokens and adding benefits to the loyalty network

Actors: CA_1: Company 1 Admin, CA_2: Company 2 Admin, CS_1: Company 1 DLT platform service, CS_2: Company 2 DLT platform service

CA_1 logs into the I-DELTA platform and gives order to create Company A loyalty tokens on the blockchain

CS_1 creates Company A loyalty tokens and places them in Company A’s wallet on the blockchain

CA_1 distributes loyalty tokens to wallets of employees of Company A

CA_2 and CS_2 perform identical operations for Company 2 in parallel.

CA_1 and CA_2 independently create benefits and submit them to their own DLT platforms respectively.

CS_1 and CS_2 independently register the benefits created by their own administrators on their own company blockchain

CS_1 and CS_2 independently use the I-DELTA platform to access the benefits registered on the other company's blockchain.

CA_1 and CA_2 then independently query the benefits available on their system and are able to view the benefit pool consisting of all the benefits from every company connected to the loyalty network.

2.1.2. Context Scenario - 2

Goal: Employee from one company purchasing benefits created by another company

Actors: CE_1: Company 1 Employee, CS_1: Company 1 DLT platform service, CS_2: Company 2 DLT platform service

CE_1 logs into Company 1's loyalty app and views benefit pool

CS_1 uses the I-DELTA platform to access the benefits registered on the other companies' blockchains that are part of the loyalty network.

CE_2 selects a benefit to purchase created by Company 2

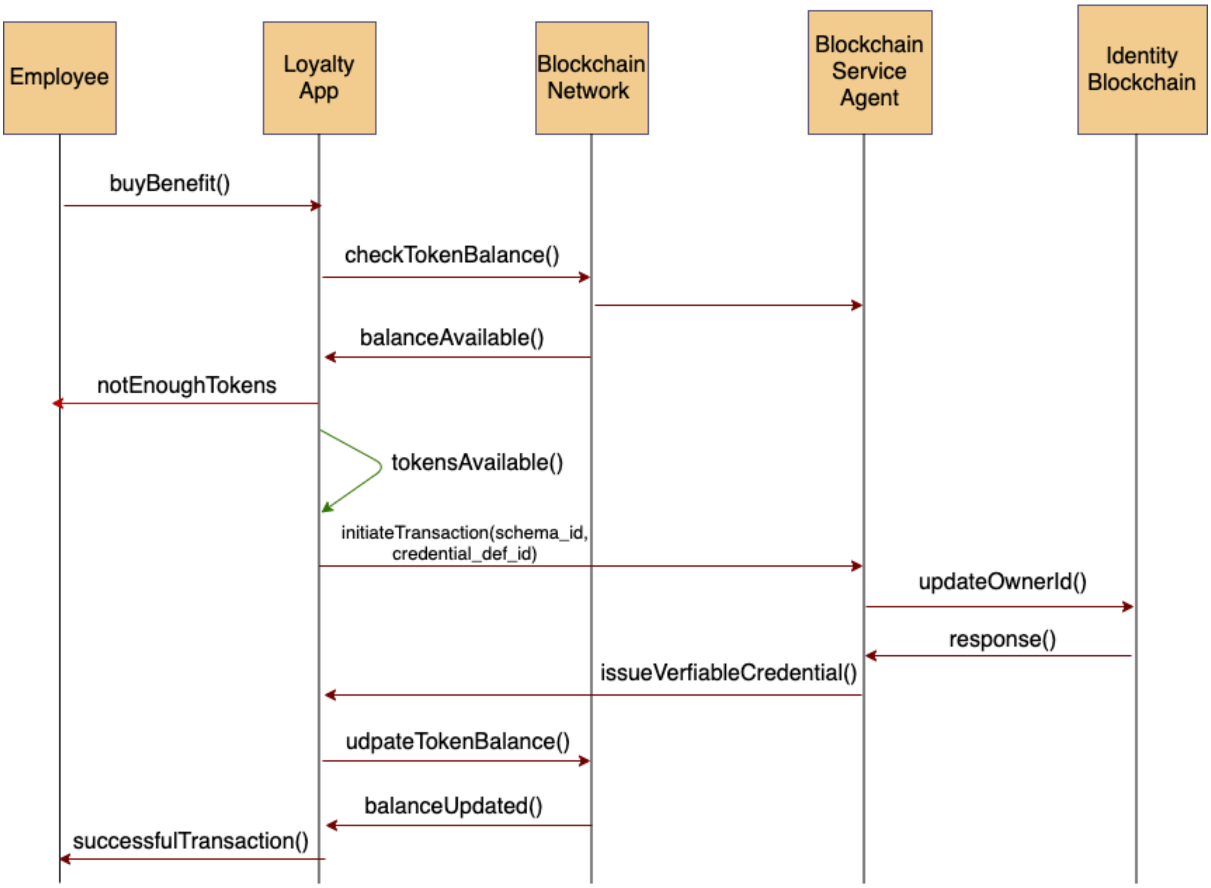
CS_1 uses the I-DELTA platform to submit a request to purchase the benefit on behalf of CE_1.

CS_2 uses the I-DELTA platform to verify that CE_1 is a approved employee of Company 1

The benefit is transferred to CE_1 and the corresponding loyalty tokens are transferred from CE_1's wallet to Company 2.

Usage scenario	Employee of one company purchases a benefit created by another company
Description	A scenario that requires blockchain interoperability between the DLTs of two or more companies
Actors	· Employee of Company 1 · Company 1 DLT platform service · Company 2 DLT platform service
Assumptions	· Actors must exist in the system and be defined · Both companies must be registered to the I-DELTA loyalty network · Employee must have sufficient loyalty token in their wallet for the purchase · The other company must have registered benefits on its blockchain
Required Fields	· Employee ID · Benefit ID · Benefit Price · Company 1 ID · Company 2 ID
Potential Risks and Challenges	· Unauthorized users accessing benefit information · Improper handling of loyalty token transfer and currency settling between companies

Purchase and transfer of benefit sequence:



2.2. Turkey, Energy Use Case

With the advancement of the reform of the power market, the development of independent negotiation transactions by market entities has become more frequent. Data such as electricity information, user identity information, and enterprise information during the power transaction process are uploaded to the chain, and distributed transactions are stored in the distributed network through distributed shared ledgers to ensure that each transaction is authentic, reliable, and traceable. Perform quality tracing and operation monitoring to achieve accurate management of participants, equipment and settlement. Transaction smart contracts are set up between transaction entities on the blockchain. Trusted identity authentication technology is used to determine the identities of both parties to the transaction, and the corresponding smart contracts are matched to achieve efficient electricity bill settlement and settlement. In the process, the use of blockchain technology to achieve the identity verification of the parties to the transaction; the application of blockchain technology in electronic invoices to solve problems such as repeated reimbursement, bill falsification; the entire process is transparent and traceable, and resolves transaction repudiation, denial, fraud, etc.

2.2.1. Context Scenario - 1

Actors **-** SP: Service Provider, L: Local Distributor, Prosumer_1, Consumer_1

Prosumer_1 has solar panels on his rooftop. He buys a metering device. He downloads the mobile app that lets him join his local micro-grid community. The app is enabled by his Participant Id authentication. He earns a currency in simulation for selling his power to Consumer_1 in the same grid on the local market.

Consumer_1 downloads the app because he wants to buy power from Prosumer_1 in the same grid. He sets his budget for local renewable energy in the app and pays in a currency.

SP creates the rules of the marketplace (inside micro-grid), is paid to run settlements between prosumers and consumers, and allows local value added services into the marketplace.

2.2.2. Context Scenario - 2

Actors: Prosumer_1@L2, Consumer_1@L1, DSO@L3 – Distribution System Operator, MSP – Microgrid Service Provider

The local market consists of two micro-grid, one distributor (DSO) and one service provider (MSP).

Prosumer_1@L2 has solar panels on his rooftop. He buys a metering device. He downloads the mobile app that lets him join his local micro-grid community. The app is enabled by his Participant Id authentication. He earns a currency in simulation for selling his power to Consumer_1 in a different grid on the local market.

Consumer_1@L1 downloads the app because he wants to buy power from Prosumer_1 in a different grid. He sets his budget for local renewable energy in the app and pays in a currency.

DSO@L3 se

ll the service of electricity transport into the marketplace. DSO@L3 receives payment for physical transfer of electricity across the network between Microgrid_1 and Microgrid_2.

SP creates the rules of the marketplace (between micro-grids), is paid to run settlements between prosumers and consumers, and allows local value-added services into the marketplace.

Usage scenario	Purchase order is given
Description	Scenario that belongs to mutual contract depending on match of purchase and sell order in the market place
Actors	· Panel owner · Battery
Assumptions	· Actors must exist in the system and defined · Battery must be on and working · Battery must be connected to consumer end points · Battery must have sufficient energy stored · Buy Order – Sell Order N-N relation (selling transctions may be completed as a whole or partially) · Date must be in format dd MM YYYY HH:MM , transfer is subject to within one hour of release date
Steps	1. Through which asset selling is going to take place is chosen (in this case battery) 2. Order date and amount and validity duration is entered 3. Selling amount option is selected 4. System must allow more than one buying order associated with selling order 5. Buyer can select amount of purchase option or buying as a bulk in the same order

Required Fields	· Selling order amount · Order validity period · Date of record entry · price · Prosumer identity
Potential Risks and Challenges	· Panel failure · Connection failure · Smart meter problems · Grid maintenance

2.3. Spanish use case

The Spanish Use Case for iDELTA is built on Digital Governance and city management, one of the major aspects of e-democracy, is one of the most outstanding society disruptors at present. The combination of Internet of Things (IoT) technologies with Smart Contracts based on cutting-edge Blockchain platforms could significantly change Smart Cities.

2.3.1. Introduction

Digital democracy is the combination of information and communications technologies with politics which allows to generate spaces for dialogue and social reflection, strongly involving the private, political and institutional sectors. The platform described in this deliverable has as its fundamental objective the analysis, design and prototyping of a Digital Governance and City Management Platform where different members of a community (or city) can effectively change and revolutionize how cities, regions and town hall are currently managed and governed, changing society by itself.

The platform intends to obtain the following objectives: implement a platform for digital governance, which allows effective control of the actions of public authorities; reach a scope of action in the platform which is not generalist but is carefully selected and oriented; build a tool where journalistic work is perceptible, providing the platform with a component of professionalism and reflection that is expected to be distinctive with respect to other platforms of social activism; allow intensive and constant collaboration with certain traditional media; obtain public information, backed by the different transparency regulations of the different public powers; develop and generalize codes of good practices that allow to govern interactions with public institutions; prepare metrics that allow evaluating the social, economic and / or political impact of the platform; become a catalyst and a meeting point for numerous other initiatives to monitor political action; analyze the different receptivity of certain social segments, and to guarantee the impartiality of the platform's political orientation.

2.3.2. Context Scenario

Digital Governance can be traced over three major aspects:

1) Incidences: Smart City Management is mostly based in finding problems or situations where a reaction is necessary to come back to a previous stage: damage in the city infrastructure, potential fires or other type of disasters, included natural ones that might arise encompass some of the most significant.

2) Voting and Decision-making: It is generally assumed that the emergence of phenomena of institutional nature as well as corruption in the same context is retributed to the opacity, or lack thereof, of certain public institutions in the eyes of an average-going citizen. Votings and open decision-making can change this situation and turn it around.

3) IoT Infrastructure Interaction: Current IoT Infrastructure deals with the integration and interoperability among the different ways of devices in IoT such as RFID, sensors and actuators (located on the bottom of the architecture) with the citizen devices (normally, smartphones) which could empower their interaction.

Our objectives are as follows:

O1) We will create a special token, the SABINA token, based on Blockchain Technologies as described in the previous section.

O2) We will build an Incidences Management platform, where citizens can find Incidences, report them and, potentially, solve them by themselves, getting SABINAs in retribution.

O3) We will create an eVote infrastructure where actively involving in voting and decision-making processes will imply retribution by means of the SABINAs token for the citizen. But decision recipients (such as service providers: utilities, garbage management, construction companies) will also be rewarded through SABINAs. Hence, if the value of the token increases together with the city uprising, reward is higher and transparent.

O4) We will develop a Smart Contract based IoT Infrastructure which allows paying every service (electric car, water and city taxes, theatres, etc) through the Smart Contract execution which uses SABINA tokens as payment.

2.4. Canada, Energy Use Case

The focus of supply chain management (SCM) for the past few decades has been centered around investments to reduce inventory costs, outsource logistic services and implement lean manufacturing processes. While these strategies are effective at optimizing operations, they increase the risk exposure of supply chains in the case of fortuitous events. The Covid-19 pandemic has demonstrated the lack of responsiveness and flexibility of global supply chains, as evidenced by the spike in closures of manufacturing facilities and the unmet demand for basic commodities. An added concern for international trade is the rise in counterfeiting, which exceeds US$ 900 bn annually[1]. These threats to the security and stability of global supply chains, result in four main pain points that are felt across industries:

[1] https://gfintegrity.org/report/transnational-crime-and-the-developing-world/

- Traceability: capability to monitor events and metadata associated with a product and its components in order to understand origination, sourcing, associated documentation, etc.
- Auditability: in case of failure or a larger catastrophic event, running forensics is extremely complicated due to data fragmentation. Forensics usually takes months if not years.
- Regulatory and Tariff Compliance: standards and controls to provide evidence that regulatory conditions are met.
- Flexibility: the ability to adapt rapidly to events or issues, run various scenarios, without significantly increasing operational costs.
- Stakeholder Management: effective governance in place to enable communication, risk reduction and trust among the involved parties.

The Canadian use case aims to address these pain points by creating a standardized digital fabric that enables different networks and systems, within global supply chains, to communicate seamlessly and operate based on a trusted source of end-to-end supply chain data. Currently, organizations have implemented digital systems to optimize internal operations; however, supply chains of multiple stakeholders are still fragmented and siloed in terms of data exchange. In this way, we will focus on connecting six information layers: (1) identity management, (2) events, (3) payments, (4) analytics, (5) IoT and (6) privacy. This will ultimately result in supply chains that are securely and privately interconnected to reduce redundancies in reporting, provide unforgeable proof of origin of materials, deliver end-to-end traceability of products and make better decisions with faster, more accurate data.

After conducting a State-of-the-Art Analysis on SCM, we have identified a need to build a blockchain-based platform for management of complex supply chains, with the focus on a common set of standards and interoperability with existing IT and future blockchain platforms. This would result in the aforementioned digital fabric that enables seamless information interchange. The following objectives to bridge the gap between the desired state of seamless system interoperability and the current state of practice:

1) Development of interface standards to enable data to be easily transferred, stored and reported amongst stakeholders' systems and device-to-platform communication. These negotiated shared formats should be governed by a set of access and privacy requirements.

2) Definition of a comprehensive governance model for distributed systems to achieve wide industry adoption and ensure the credibility of data input into the blockchain (i.e. decisions of data storage, control and methods of gathering).

3) Implementation of identifiers and credentials that can be accurately traced back to the real-world entity and/or product, effectively creating a digital twin that is updated in real-time throughout a product's lifecycle. These identifiers should marry various systems, in order to exchange data without introducing intermediaries or reducing trust.

4) Data harmonization – methods to gather heterogenous data from different sources into a cohesive set – is paramount to realizing all the benefits of IoT devices and analytics.

5) Design an architecture of networks with different consensus mechanisms for their appropriate use case, to manage capacity and throughput needs.

6) Build cryptographic solutions that address privacy needs and can also be deployed at a large-scale (i.e. zero-knowledge proofs, group signatures, multi-party computation and homomorphic encryption).

2.4.1. Context Scenario

In order to demonstrate the functionality of a macro supply chain that interoperates with multiple systems, we will develop a commodity traceability platform that will initially focus on the steel and auto industry (after an MVP is demonstrated, the goal would be to scale the platform to multiple types of products). Raw materials will be traced from their origin, through transformation into finished materials, fabrication into parts or components and assembly into end-products. Further, we will include services that occur in parallel to the physical supply chain, such as trade finance and contract management. The following scenario describes the key steps in the process:

1) Raw materials (i.e. aluminum, iron ore, limestone, coal) are extracted and processed by **mining companies**. These materials are then transported by **carriers** (outsourced or insourced) to manufacturing facilities.

2) **Manufacturers** transform raw materials into semi-finished or finished materials (i.e. steel).

3) **Manufacturers** then supply the materials to original equipment manufacturers (OEMs) for fabrication into car parts and/or components. The materials are delivered through varying methods of transportation and could be exported or sold within the country.

4) **OEMs** produce the finished products and supply them to car assembly plants. This process would vary depending on the level of company integration.

5) **Car assembly plants** produce cars and other automotive vehicles ready for use.

<u>Processes that occur in parallel</u>

1) If the commodities are purchased from foreign suppliers, importers would submit an import declaration through a **customs broker** (3rd party).

2) **Customs and border control agencies** assess the import declarations and apply trade benefits depending on the rules of origin, composition, etc.

3) Sales and purchases contracts are executed throughout the process between **buyers and suppliers**.

4) Letters of credit are issued between **banks**, following an application, presentation of documents, risk and creditworthiness assessment.

5) Once the contract(s) is met, **buyers and suppliers** settle transactions among themselves and with regulatory agencies if applicable.

6) Throughout the shipment process and during the post-trade, **companies** submit reports to **regulatory agencies** that oversee raw material extraction, international trade and environmental footprint, among others.

We will develop a blockchain platform that tracks the composition of car parts and integrates with a commodity traceability platform. This platform will in turn integrate with stakeholders' existing IT systems and other blockchain networks (e.g. multi-modal logistics, supply chain finance), where different processes such as financial services and border clearance are recorded.

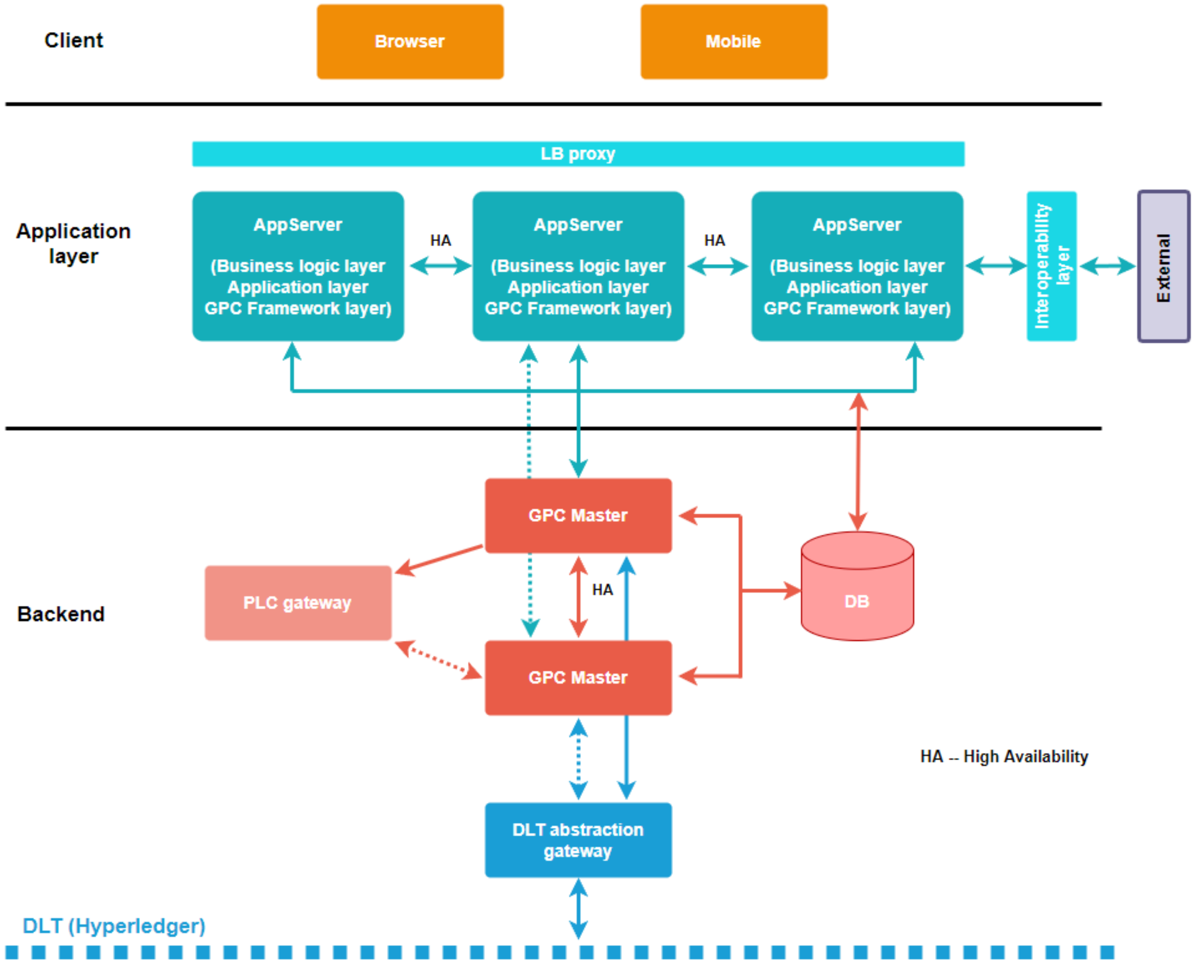
2.5. Czech Republic, Digitalization of Legal Agendas Use Case

Our application scenario covers voting for both citizens of a city and special use at general meetings using digital technologies based on DLT. The voting process is inherently regulated by a number of rules, some of them legally required to be verified by a notary record. A notary functions as a 3rd party arbiter validating votes and decisions made by legitimate voters.

Important decisions might include public polls for city hall decisions or in case of companies change of a management, structure or distribution of profit. Using the GPC framework, embedded devices and distributed ledger technology, an external arbiter is no longer required. Notary is substituted by a DLT register receiving data from embedded devices.

A wearable or easy to transport HW module which will perform authorization of the legal act and save the transaction record into DLT is an integral part of our use case. The reason for this approach is to make utilization of DLT transparent to non IT specialists. For end users like lawyers who are not professionals in IT it is necessary to provide an easy to use solution, not requiring knowledge of used technology like DLT, digital signing, encryption algorithms, etc. The HW module will provide some authentication mechanisms like fingerprint scanner or keypad for input of PIN, push-buttons for voting, secure storage of PKI keys and communication interface.

2.5.1. Components diagram


Client
Browser
Mobile
LB proxy
Application layer
AppServer (Business logic layer Application layer GPC Framework layer)
HA
AppServer (Business logic layer Application layer GPC Framework layer)
HA
AppServer (Business logic layer Application layer GPC Framework layer)
Interoperability layer
External
Backend
PLC gateway
GPC Master
HA
GPC Master
DB
HA – High Availability
DLT abstraction gateway
DLT (Hyperledger)

Application layer

Individual sub-applications, functionalities and support services operated within the platform or offered by external providers / service providers with which the platform is connected, etc.

Middleware

At this level, the operational management of the entire IoT ecosystem takes place. It includes the means for managing the central and end infrastructure, integration with native and external applications and functionalities, monitoring the status of sub-components, implementation of event management processes, incident management, problem management and change management (or Service Operation in general).

Key functionalities at this level are mainly adding/removing/replacing individual platform components, i.e. devices, applications, service providers, etc. There is also support for technologies including distributed systems, cloud computing, edge computing, temporal and network relational databases and computations/operators over a generic data model.

In the area of security, support for encryption, data anonymization and pseudonymization, authentication, authorization, security model and role-based access control (RBAC). Resources for support for security functions will also be implemented at the layers to the extent necessary for lower layers.

Communication and interconnection layer

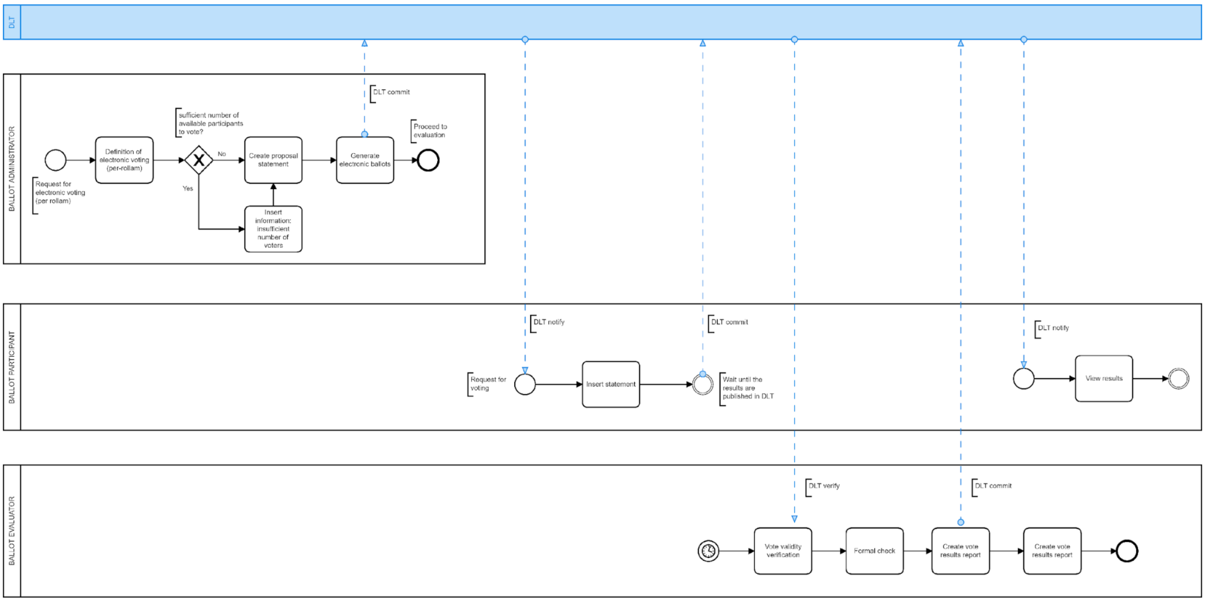
This layer contains both hardware and software resources and is used to ensure mutual communication (whether permanent, intermittent, on-demand, etc.) between the elements that make up the IoT ecosystem.

Brokerage layer – PLC Gateway

This layer contains concentrators for connecting several hardware modules (sensors, actuators, display devices, etc.). Individual brokers are globally addressable throughout the ecosystem and it is the smallest directly queryable unit offering certain (service catalog defined) services. In addition to collecting data and executing defined commands over connected end-elements, devices in this layer can provide other, more complex services towards the IoT ecosystem – typically Edge Computing support (e.g. for input aggregation, data preprocessing, etc.), temporary data storage and caching, custom control logic in case of unavailability of parent control nodes (either due to failure or planned in areas with non-guaranteed permanent network connectivity).

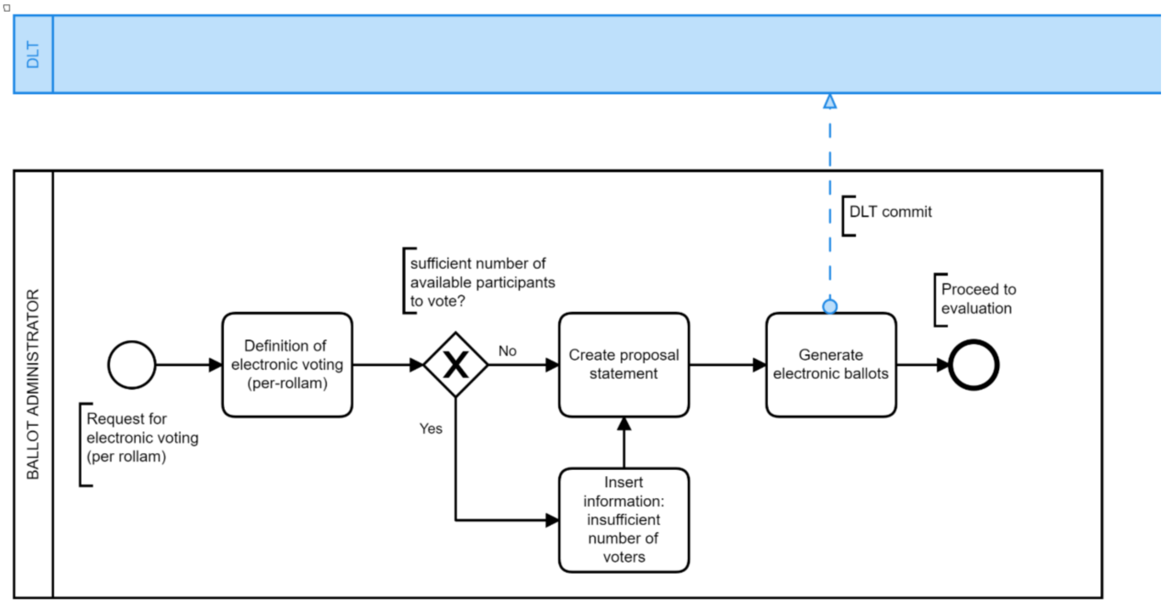
2.5.2. Workflow Actions

Overview


DLT
BALLOT ADMINISTRATOR
Request for electronic voting (per rollam)
Definition of electronic voting (per-rollam)
sufficient number of available participants to vote?
No
Yes
Create proposal statement
Insert information: insufficient number of voters
DLT commit
Generate electronic ballots
Proceed to evaluation
BALLOT PARTICIPANT
DLT notify
Request for voting
Insert statement
DLT commit
Wait until the results are published in DLT
DLT notify
View results
BALLOT EVALUATOR
DLT verify
DLT commit
Vote validity verification
Formal check
Create vote results report
Create vote results report

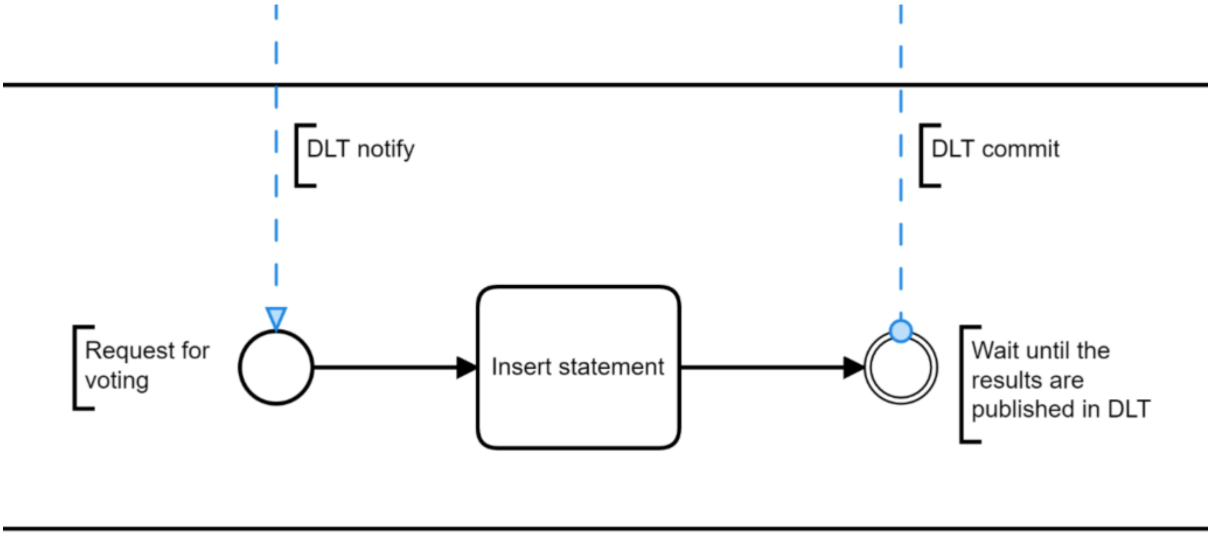
The diagram reveals administration, participant and evaluator workflow, all communicating through commits written to and received from DLT (Hyperledger Fabric in our case). Let's have a closer look at each step below.

Ballot administrator workflow


DLT
BALLOT ADMINISTRATOR
DLT commit
sufficient number of available participants to vote?
Proceed to evaluation
Definition of electronic voting (per-rollam)
No
Create proposal statement
Generate electronic ballots
Request for electronic voting (per rollam)
Yes
Insert information: insufficient number of voters

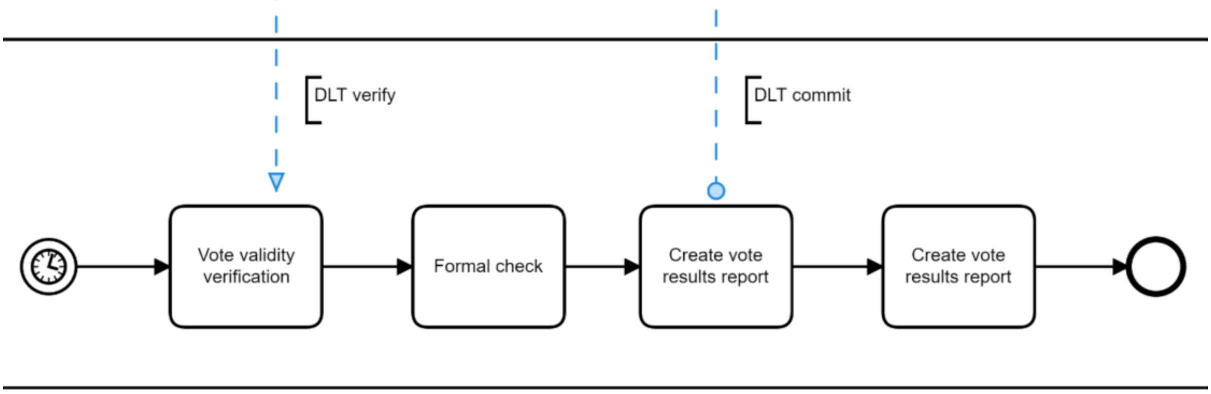
By signing into the application a ballot administrator creates a request for remote voting (per rollam). For the subsequent step the user sets properties of the voting (definition of the voting). These properties include how big the majority of voters must be in order to have a valid result, voting period, description, additional conditions for valid vote etc. If the number of available participants is sufficient a proposal statement with generated electronic ballots are created. Ballots are then committed into our Hyperledger Fabric, making it available for eligible voters for a defined time period.

Ballot participant workflow



The voter (end-user of our application) receives notification for the voting. Now he can insert his vote through **either an application** or a dedicated voting device, which we have developed for cases when the voting **device** is required to remain at an office or public place. Both of these choices **possess** authentication tools. The vote is committed into DLT. As soon as the voting results are available (processed in the evaluator layer) the user receives a notification.

Evaluator workflow


DLT verify
DLT commit
Vote validity verification
Formal check
Create vote results report
Create vote results report

When the time period defined at the beginning (definition of electronic voting) passes the received results are put under validity verification process and formal check. If all data met the required conditions a result is reported to both users and the administrator.

2.5.3. Development results during the reported period

Software for performing and controlling the execution of legal actions described by process-object means. The software for the execution of legal acts described in a XML document and derived from the process-object model has been implemented and tested. Including a set of elementary activities for the execution of legal actions and implementation of validity check of the XML document describing the legal action.

Software for description, formalization, validation and visualization of the legal process by process-object means. Basic technological components for general user interface have been implemented as well as means for advanced data modeling, means for modeling the process describing the legal action, implemented means for generating a process from a document and a document from process, designed means for ensuring consistency and validity of the individual blocks describing a legal action during its life cycle (implementation, modification and update).

Hardware module for identification of a person and execution of a legal action - basic variant. This module was used in particular to check the suitability of the different types of controls and displays in terms of user comfort. The module includes a fingerprint reader, rotary encoder, push-button switches, signaling LEDs and an OLED display. The module is controlled by a RockPi microcontroller and is powered by an external power supply.

Hardware module for identification of a person and execution of a legal action - advanced version. This module is based on the basic variant module. The location of individual control elements has been optimized as well as the addition of a switch equipped with a key, the replacement of the fingerprint reader module with a module with better ergonomics, the encapsulation of the module and the addition of cryptographic security of operations. The cryptographic operations are implemented using a Microchip ATECC608A chip.

3. Use case requirements

3.1. I-DELTA general requirements

3.1.1. Framework requirements

Identity

1. I-DELTA framework must support external DID-based Identity for organizations, users, and digital assets.
2. I-DELTA framework must define a standard to map multiple DLT-specific Identities to a unique DID that represents a digital asset uniquely across various DLTs.
3. I-DELTA framework must prevent DID to be used for unwanted/malicious entry and usage by providing necessary authentication and authorization.

Security & Privacy

1. I-DELTA must have a mechanism for public and private transactions between different DLTs.
2. I-DELTA must have a robust and secure mechanism as a building block for atomic transactions and to prevent them from being used or taken over by malicious actors. (counterparty signing)
3. I-DELTA must have a signature layer that will be used to (re-)register assets, contracts , etc. This layer will also be used to inform all the joining DLTs in case the functionality or the state of these objects change.
4. I-DELTA platform should provide robust auditability features to investigate the causes and to detect malicious behaviour in case of catastrophic failure or security breaches.

5. I-DELTA platform should provide a mechanism for certificate/authorization revoke operations.

6. I-DELTA may provide identity recovery options using threshold cryptography or other cryptographic ways to rebuild the identity on the user side.

Interoperability

1. I-DELTA framework must define a standard for connecting with external DLTs.

2. I-DELTA framework must define a standard for integrating with IoT devices.

3. I-DELTA framework must define a standard for integrating with traditional IT systems.

4. I-DELTA must provide a mechanism for all parties to commit their transactions as an atomic operation. Like distributed transactions. (one party)

5. I-DELTA may provide a mechanism for creating reverse operations to provide rollback on non-transactional DLTs. Providing Command Pattern by allowing parties to implement DO and UNDO (Reverse Operation) operations.

6. I-DELTA must provide a mechanism for one-way transactions if it is unnecessary for several parties to commit together.

7. I-DELTA must adapt itself to support at least the maximum consensus latency of the joining DLTs, as well as minimum throughput, to ensure a healthy interoperable functionality.

8. I-DELTA DLT client software must be able to run on embedded devices - full functionality is not required, but an embedded device must be able to interact with the DLT.

Data & Processes

1. I-DELTA framework must define a standard conceptual model for the exchange of data between DLTs. (ALL)

2. I-DELTA framework must define a standard protocol for reliable orchestration of tasks, actions, and logic (e.g. smart contracts or chain code) between DLTs. (ALL)

3. I-DELTA framework must define a standard mechanism for performing analytics/reporting on data stored across multiple DLTs while preserving applicable data privacy.

4. I-DELTA framework should provide protocol for distributed voting.

5. Additional constraints (processing power consumption, connectivity), seldom-connected parties.

6. Every transaction must contain different timestamps created by DLT (create, commit, publish…).

7. Language for specifying the filter (query) for the DLT transactions for verification or notification functionality – e.g. an embedded device wants to verify that a particular transaction exists in the ledger.

3.1.2. Architecture requirements

1. I-DELTA DLT Architecture must support private, public and permissioned deployment models.

2. Every DLT should query a single Discovery Service to discover endpoints for participating DLTs.

3. The Discovery Service should implement health checks for endpoint management and failover.

4. Adding/Removing/Modifying endpoints of DLTs should be seamlessly facilitated by the Discovery Service.

5. I-DELTA must include a set of standard interfaces for external DLTs that accomplish specific business functions (e.g. payment, logistics, insurance, finance).

6. I-DELTA needs to support synchronous and asynchronous communication with appropriate ordering mechanisms.

7. I-DELTA should include notifications across DLTs.

8. I-DELTA platform infrastructure should support on-premise, cloud or container technology (docker, Kubernetes) deployment options; without requiring vendor lock-in.

3.2. First Turkey pilot/use case specific requirements

3.2.1. Functional Requirements

1. The marketplace hosted on I-DELTA should allow users, companies and SMEs to transfer and / or exchange uniquely identifiable loyalty tokens on the owners behest. Therefore, the I-DELTA platform should keep track of and update the state of ownership of all such tokens instantaneously and simultaneously.
2. Allows all participating actors in the marketplace to have the ability to buy, sell, store, exchange or transfer digital assets between them.
3. Allow for searching/filtering digital assets/benefits listed on the marketplace.
4. The platform must provide a standard mechanism for analysis of digital assets placed on the marketplace so as to allow companies/organizations responsible for their publication to study spending patterns, the popularity of digital assets, frequency of purchases/trades so as to provide more effective and robust loyalty programs.

3.2.2. Non- Functional Requirements

1. The system must comply with KVKK.

3.2.3. Privacy/Security

1. I-DELTA will provide a Digital Identity for authorized access to the digital marketplace and as proof of ownership of tokens and digital assets.
2. Each marketplace should be mapped to the organization/company responsible for its management.
3. The I-DELTA platform will generate uniquely identifiable, cryptographically secure loyalty tokens on the behest of employers/companies for employees and/or SMEs that can be used in the marketplace for various transactions related to the loyalty program.
4. Will not generate tokens that are not assigned to an employee/employer.
5. All listed digital assets must provide transparency to allow employees and other actors to gain information regarding the asset/benefit in question.

3.2.4. Interoperability

1. I-DELTA should allow for the building and/or integration of DLT based marketplaces.

3.3. Energy wallet use case specific requirements (Turkey, Canada and Spain use case)

3.3.1. Second Turkey pilot/use case specific requirements

Functional requirements

1. **Actors**
    1. Platform must support define user role as producer/consumer/prosumer. Must define a user with an unique identifier.
    2. Certificate issuer (Canada), payment actor (Spain)?
    3. Platform must support define assets with their ownership, location information and each asset must have an unique identifier.
    4. Each organization (auditor institution), must define in the platform and must have an identity.
2. **Wallet**
    1. The system must store user balances and after trading system must have calculated user' new balance.
    2. There must be a mechanism that suggests to consumers available energy with their costs. ? (Spain).
    3. The system must integrate with electricity grid management and smart grid platforms.
    4. The system must provide a middleware API that enables electricity trade. (Mobile, web etc.).

Privacy/Security

1. The access authorization of organizations to the ledger should be controlled by the system. Some organizations should only have read authorization, and some organizations should have both read and write authorization.

Interoperability

1. The system must be able to integrate with industry specific DLTs (finance, insurance etc.).

3.3.2. Canada pilot/use case specific requirements

Functional requirements (General Energy)

1. System must provide a standard mechanism to define, store, and exchange asset lifecycle information. The platform must support storing asset origin, ownership, composition and key lifecycle events in an immutable data structure.

2. Platform must support storing asset lifecycle events and data in real-time.

3. Platform must support storing asset lifecycle events and data in an immutable fashion.

4. Each asset represented on the platform must have an identifier that is unique across systems.

5. Asset data should include physical origin, digital origin and asset composition information. This information may be gathered from and communicated to other systems.

6. Platform must support tracking asset parent-child relationships, including scenarios where parent and child are tracked in different systems.

7. The system must support both non-fungible (e.g. metals, some types of crude oil) and fungible (e.g. natural gas, electricity) assets.

8. The system must support tracking production and consumption of assets and record “liveliness” status to prevent asset misrepresentation or double-spend.

9. The system must be configurable to allow for nuanced business requirements of specific industry verticals. Configuration must include asset attributes, asset events

and event attributes, asset and event validation rules. Configuration may include rules of origin, tariff rules.

Functional requirements (Electricity Pilot)

1. The system must support issuing and tracking renewable energy credits based on the attributes of electricity produced.
2. The system must support calculating and storing environmental impact of an asset based on the asset lifecycle (production, transportation, transformation).
3. The system must expose an API specific to the electricity domain.
4. The system must integrate with electricity grid management and smart grid platforms.

Privacy and Security

1. Access to data should be on a least-privilege basis.
2. The system must implement organization registration, user registration, onboarding, RBAC based on a user’s role within an organization and organization’s relationship with specific digital asset supply chain.

Interoperability

1. The system must be able to rely on and integrate with multiple external system providers for auxiliary services (e.g. logistics, finance, insurance).
2. Input from different data sources must be sanitized and validated before being committed to the DLT.
3. Analytics must be real-time with the ability for reporting, AI or any other analysis on the raw ledger data available to the specific stakeholder.

Non-functional

1. The system must rely on the platform’s Identity and Discovery Services, and be built to utilize other Shared Services.
2. The system must comply with NIST.

3.4. Czech Republic use case specific requirements

3.4.1. Functional Requirements

1. DLT_JOIN by which any subject, system, or participant can join the DLT.
2. DLT_COMMIT by which any member of the DLT ecosystem can issue a confirmation that a certain operation, event, transaction etc. has occurred.
3. DLT_VERIFY by which any member of the DLT ecosystem can verify information.
4. DLT_NOTIFY by which any member can be informed about a particular event occurring in the DLT (e.g. a new transaction).

3.4.2. Non-Functional Requirements

1. Immediate response for successful/unsuccessful transactions
2. The system must comply with GDPR

3.4.3. Privacy/Security

1. Data are available only to authorized parties (permissioned model of DLT).
2. Digital Identity must be protected to prevent unwanted/malicious entry and usage.

3.4.4. Interoperability

1. Access to public information service registry implemented as another DLT.