cssf

Commission de Surveillance
du Secteur Financier

Distributed Ledger Technologies & Blockchain

TECHNOLOGICAL RISKS AND RECOMMENDATIONS FOR THE FINANCIAL SECTOR

Commission de Surveillance du Secteur Financier

Distributed Ledger Technologies & Blockchain

Technological risks and recommendations for the financial sector

CONTENTS

Commission de Surveillance du Secteur Financier

Distributed Ledger Technologies & Blockchain

Preliminary remarks

This document is not binding. It is a supporting document aimed at guiding professionals in their due diligence process related to the use of a DLT.

The CSSF applies a principle of technology neutrality and acknowledges that innovative processes and technologies can contribute to improvement of the provision of financial services. When properly used, a DLT, like other technologies, can provide benefits for the financial sector. However, institutions must demonstrate that prudential and regulatory requirements are met when using a DLT.

It is essential that professionals conduct a proper risk assessment when developing, providing, using or implementing a DLT. These risks must be clearly identified, mitigated and monitored throughout the entire life cycle of the DLT use.

It shall be noted that the present white paper does not purport to provide all the technical explanations on the functioning of a DLT. There is an extensive literature explaining the principles of a DLT[1]. If the document recalls the commonly accepted definitions of a DLT, readers should have a minimum level of prior knowledge of it.

[1] *Refer to 7 - Bibliography*

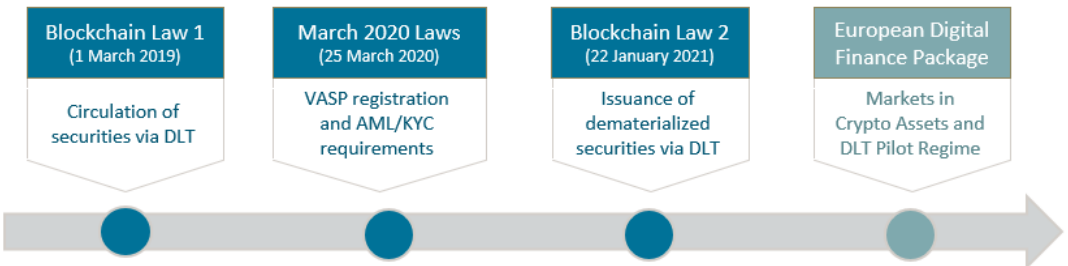
1. Introduction

1.1. Context: Opportunities for the development of DLT use in the financial sector

Distributed Ledger Technology (DLT) is a technology that has been used for many years. Its potential has been emphasized in 2008, through the development of the blockchain (which is a particular type of DLT) on which the crypto-currency referred to as Bitcoin still relies. Focusing on the technical aspects only[1], 13 years after its implementation, the blockchain on which Bitcoin relies has proven its worth in terms of security and robustness.

Nowadays, the DLT is seen, by some, as the next step towards the digital transformation. The DLT, with its disruptive potential, may have a significant impact on the financial sector in the decade to come. This technology has been recently considered as potentially revolutionary as many practical applications have been identified or developed. The potential opportunities offered by the DLT resulted in an increase of interest from the financial sector (in areas such as initial coin offering, KYC and counterparty or customer identification, collateralization, fund distribution, payment systems, etc.). The financial sector is thus seeing the emergence of more and more DLT applications and use-cases to streamline and digitize business processes by limiting or eliminating the need for reconciliations or intermediaries with the help of the DLT.

Important legal uncertainties around the potential use of DLT or DLT applications have been or might be cleared in the years to come, both on National and European levels, with a view to allow the uptake of DLT.



[1] *And without taking position on the cryptocurrency Bitcoin itself.*

On a national level, the law of 1 March 2019[1] and the law of 22 January 2021[2] (also often referred to as the 'Blockchain laws' 1 and 2) respectively allow (i) the maintenance of securities in distributed form, thus enabling the circulation of securities by way of inscription in a distributed ledger and (ii) the issuance of dematerialized securities and maintenance of the securities issuance account, within or through secured electronic registration mechanisms, including DLT or distributed databases. The law of 12 November 2004 on the fight against money laundering and terrorist financing has been amended[3] to introduce a new status of virtual asset service provider, subject to a registration with the CSSF and an AML/CFT supervision by the latter.

On a European level, on 24 September 2020, the European Commission adopted an ambitious digital finance package[4], including 3 proposals for regulations on markets in crypto-assets, on digital operational resilience for the financial sector and on a pilot regime for market infrastructures based on DLT.

Both the legislator and the financial sector are taking up the challenge of integrating the DLT in a fast-moving environment.

In this context, over the past few years, the CSSF has been increasingly solicited by financial and non-financial institutions, incumbents and start-ups, wishing to present a large diversity of applications and use-cases of DLT, in various sectors.

Several countries are working on projects related to the issuance of their own central bank digital currency (CBDC) projects. The European Central Bank is analysing the possibility of the launch of a digital Euro and has recently published a study[5] that highlights "the risks to stability that might arise if a central bank does not offer a digital currency". China has been working and testing the concept of a digital yuan in 4 cities. In a recently published white paper[6], the People's Bank of China (PBoC) announced that the digital yuan will be programmable with smart contracts. Successful completion of these initiatives would have a boosting effect on the development of DLT-based projects.

[1] *Law of 1 March 2019 amending the law of 1 August 2001 on the circulation of securities*

[2] *Law of 22 January 2021 amending the law of 5 April 1993 on the financial sector and the law of 6 April 2013 on dematerialised securities*

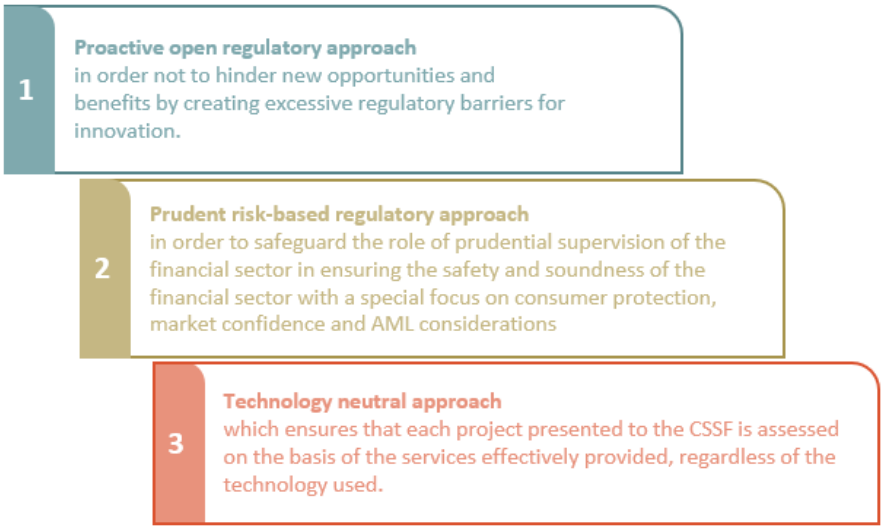
[3] *Laws of 25 March 2020: https://legilux.public.lu/eli/etat/leg/loi/2020/03/25/a193/jo and https://www.legilux.public.lu/eli/etat/leg/loi/2020/03/25/a194/jo*

[4] *https://ec.europa.eu/info/publications/200924-digital-finance-proposals_en*

[5] *https://www.ecb.europa.eu/pub/ire/html/ecb.ire202106~a058f84c61.en.html*

[6] *http://www.pbc.gov.cn/en/3688110/3688172/4157443/4293696/2021071614584691871.pdf*

The integration of evolving technology-based innovation in financial services and markets is a continuing challenge for regulators such as the CSSF, calling for:



1.2. Purpose and scope of the white paper

The DLT is based on complex cryptographic mechanisms, consensus mechanisms and concepts such as smart contracts. Understanding such a technology and its concrete applications requires a high-level of knowledge of and familiarity with various technical notions.

Institutions seeking to offer services based on DLT, which do not necessarily have all the technical skills required for its implementation, may tend to overlook the risks inherent with this technology. Like any other technology, the DLT entails specific risks that must be understood, mitigated and monitored.

The present white paper is a document that is both educational and thought-provoking. It primarily targets professionals being financial and non-financial institutions providing or intending to provide services to the Luxembourg financial sector.

This non-binding document invites any stakeholder to consider the concrete implications of the use of a DLT in the provision of its services. It seeks to encourage stakeholders to conduct a proper assessment of the risks related to the DLT and its use in the provision of services in the financial sector.

In line with the technology-neutrality principle applied by the CSSF, this white paper does not by any means whatsoever constitute a positive or negative assessment of the DLT itself which, whilst it entails specific risks, may offer important advantages when properly implemented and used. The present white paper's purpose is to ensure that both risks and advantages are adequately and appropriately taken into consideration by the financial sector.

To that extent, it aims to:

- **Identify the key components of a DLT and the different types of DLT available;**
- **Highlight the roles and responsibilities of the different actors in the use of a DLT (i.e. DLT developer, infrastructure provider, solution provider and users); and**
- **Emphasize some of the main risks related to the DLT, both in terms of governance and technical risks.**

2. Definitions

In Luxembourg, the law of 1 March 2019 does not give a definition *per se* of the DLT but simply refers to "secure electronic recording devices, including distributed electronic registers or databases".

At the European level, the European Banking Authority ("EBA") first, in its July 2018 report[1], defined the Distributed Ledger Technology (DLT) as "the term used to refer to those technologies that allow a common ledger to be shared across networks of computers."

In September 2020, the European Commission proposed its own definition in its draft Pilot regime for DLT market infrastructures[2], the article 2 defining "DLT" as "a class of technologies which support the distributed recording of encrypted data".

Actually, developments around DLT are numerous and to elaborate a definition of DLT is a moving target. For the purpose of this white paper, we propose nevertheless to define Distributed Ledger Technology as follows:

> *DLT is a technology allowing a network of independent and often geographically dispersed computers to update, share and keep a definitive record of data (e.g. information, transactions)[3] in a common decentralised database in a peer-to-peer way, without the need for a central authority.*

The growing interest for the technology has motivated people to develop various types of DLTs and adapt them to a vast number of domains depending on the developers' goals. However, some key characteristics can be identified as common to DLTs. The key common characteristics and the main types of DLT are respectively presented in sections 2.1 and 2.2 below.

[1] *EBA Report on the prudential risks and opportunities arising for institutions from Fintech, July 2018.*

2 Proposal for a Regulation of the European Parliament and of the Council on a Pilot Regime for market infrastructures based on distributed ledger technology - COM(2020)594: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52020PC0594

[3] *In this white paper, when used outside a specific use-case, the term "transaction" is equivalent to "data" or "information".*

2.1. Key common characteristics of DLTs

The fact that the database is distributed over multiple nodes does not constitute the differentiator of the Distributed Ledger Technology. Indeed, distributed databases have been known for decades and have come along the rise of Cloud computing and virtualization. Most websites run on distributed databases with multiple clusters of nodes geographically distanced.

Actually, DLT can be implemented with a wide variety of characteristics (refer to section 2.2). However, without giving a strict definition of DLT, the CSSF considers that two key elements may really differentiate DLT from traditional databases and qualify a technology as DLT, including, namely: the use of a consensus mechanism to validate the transactions added to the ledger; and the use of cryptography means to guarantee the immutability, non-repudiation and authorisation of transactions.

- **Use of a consensus mechanism through the network of nodes.** The nodes must reach a consensus amongst each other to validate new data entries by following a set of predefined rules. The consensus mechanism is specified in the algorithm that defines the distributed ledger[1]. As there are various types of DLTs, this consensus mechanism can vary depending on the nature, goal and underlying asset of the technology.

  The aim of the consensus mechanism is to determine whether a new transaction on the DLT is legitimate or not. Every local addition to the ledger by a network participant is propagated to all nodes in a peer to peer manner. Once every node involved in the consensus has validated the transaction independently and have collectively agreed on the valid transaction, it is added to the DLT, and only when this specific condition is met (i.e. the consensus is reached between the participants). The validation of transactions is therefore decentralised to a network of nodes sharing control of the ledger, removing the need to rely on a central authority. The information is then synchronised across the network in every node, which ensures data consistency.

  In an environment without a central authority, the consensus mechanism prevents the network from being hacked or misused and allows for trust in a non-trusted environment. With DLT being originally used for cryptocurrency-related use-cases, the consensus was the solution against the double-spending problem – guaranteeing that a transaction cannot be performed twice within the network.

[1] *Note however that one ledger could support different consensus mechanisms.*

- **Usage of cryptography means to ensure immutability, non-repudiation and authorisation:**
    - **Immutability**: Once a transaction has been validated by the nodes and added to the DLT, it can no longer be retroactively altered or changed. In cases of blockchains, hashing functions are used to prevent tampering of the data stored in the ledger. Immutability can also be achieved for non-blockchain DLTs through the use of other techniques such as digital signatures[1]. This is a clear difference to conventional databases where a database administrator has the possibility to modify records stored in a database.
    - **Non-repudiation**: Non-repudiation means that participants of the DLT cannot deny the authenticity of the transaction or message accepted in the DLT. The purpose of non-repudiation service is to collect, maintain, provide, and verify the undeniable evidence about messages or transactions between participating actors. The non-repudiation has two aspects: on the one hand, the information or transaction sent cannot be denied and, on the other hand, the recipient(s) cannot claim that they did not receive the message or transaction.
    - **Authorisation**: Asymmetric cryptography is used to provide users of the DLT with public and private key pairs. Each transaction is signed to be acknowledged as valid during the nodes' validation process and to prove that the transaction has been initiated by the authorized user.

Those key components of the Distributed Ledger Technology enable transactions and data added to it to be recorded in an immutable way, shared and synchronised instantly across its distributed network.

[1] *https://docs.corda.net/docs/corda-os/4.6/key-concepts-transactions.html*

2.2. The different types of distributed ledgers

Still at its early-stage of development due to its novelty, the innovative characteristics of DLT have contributed to its exponential popularity. As previously mentioned, various types of DLTs have been developed and adapted to a vast number of domains depending on the developers' goals.

The figure below proposes a classification of the main types of DLT according to their key properties, which are further explained thereafter.

Access Data ('read' data)	Public			Private	
Submit Transactions	Unrestricted			Restricted or Unrestricted	
Validate Transactions ('write' data)	Permissionless	Semi-Permissioned		Permissioned	
Consensus	Proof-based	BFT-based & others (PBFT/FBA/PoS...)		BFT-based & others (PBFT/FBA/PoS...)	
Data Structure	Blockchain	Blockchain	Non-Blockchain	Blockchain	Non-Blockchain
Example	Bitcoin, Ethereum	Stellar	Ripple	Quorum, Hyperledger	Corda

Figure 1: Different types of distributed ledgers[1]

2.2.1. Access rights

Public vs Private

Distributed ledgers are often categorized as public or private depending on who can access the ledger. By default, public ledgers are typically open to anyone whereas private ledgers are only accessible to authorised users.

Unrestricted vs restricted

It is also possible to restrict further the access rights of a participant on an individual basis, for instance by authorising only to read the ledger. With regards to creation rights, a ledger is said restricted when a specific access right is required to create transactions in the ledger or unrestricted when any user of the private ledger can submit transactions for inclusion in the ledger.

[1] *"BFT" stands for Bizantine fault tolerant, "PBFT" for Practical Byzantine Fault Tolerance, "FBA" for Federated Byzantine Agreement and "POS" for Proof of Stake*

2.2.2. Validation rights

Permissioned vs permissionless

Distributed ledgers may be permissioned or permissionless. In permissionless ledgers, anyone can join the network as a validator node and validate transactions. The validators do not know each other.

On the other hand, permissioned ledgers rely on transaction validators who are identified and authorized to act as such and who trust each other.

Usually **public ledgers** i.e. for which anyone can join the network and access the ledger **are also unrestricted** (anyone can submit a transaction) **and permissionless** (anyone can validate / verify transactions). **Private ledgers** i.e. for which only authorized users can join the network/access the ledger **may be unrestricted** (all users authorized to access the private ledger can submit a transaction) **or restricted** (only a subset of authorized users can submit a transaction) **and are permissioned** (only a subset of authorized users can validate transactions).

"Semi-permissioned ledgers"

Finally, we have identified a category of ledgers that we qualify as "semi-permissioned" ledgers. Such ledgers offer the possibility for anyone to run a validator node but each validator node does not have the same "validation capacity". Indeed, each validator node maintains a list of nodes it knows and trusts among all existing validator nodes. The more a node is selected on a list, the more weight it has in the validation process; it becomes a "trusted validator node". Even if each validator node can be part of the consensus process, including potentially malicious, unknown or unproven nodes, this concept of trusted validator nodes prevents them from having any validation power on the network.

Therefore, in what we call a "semi-permissioned ledger", anyone can join the network of validators as in a permissionless ledger but only trusted and identified nodes (as in permissioned ledgers) really have any weight in the transaction validation process.

2.2.3. Consensus methods

As explained previously, the aim of the consensus mechanism is to determine whether a new transaction or record on the DLT is legitimate or not and can therefore be added to the distributed ledger or not.

In a permissionless distributed ledger, anyone can join the network so the risk of bad behavior from a participant is typically higher than in a permissioned distributed ledger where participants are known and could face consequences in case of malicious behavior. In order to validate transactions over a network of untrusted participants, “proof-based” consensus methods are generally used (e.g. PoW, Proof-of-Work). In public blockchain, participants validating transactions, also known as miners, are rewarded for contributing to the validation process. A defined number of token or cryptocurrencies are created with each new block of transactions and are rewarded to the miners. It acts as an incentive mechanism to attract miners in securing the network.

As Proof-of-Work has some known drawbacks (i.e it requires a large amount of computing power and does not scale very well), new generation of public blockchains are arising and are based on Proof of Stake (PoS). In PoS based public blockchains, owners of the blockchain-native virtual currency can put some of their currencies at stake to participate in the consensus. They chose one or many validator nodes based on the trust they put in them and their stakes are linked to these validator nodes. When the validator node validates a block, stakers linked to this node are rewarded proportionally of the amount they have staked. The validator node is also rewarded. These new generations of blockchains also come with a new topology as it will be explained in the next section.

Permissioned distributed ledgers are typically based on byzantine fault tolerance methods to generate the consensus between the participants. As the participants are identified and have been authorized to participate in the network, consensus methods requiring less computing resources can be used to reach an agreement on what they believe is the correct data to be added in the ledger. As a result, it generally offers better performance in terms of transaction processing than the permissionless distributed ledgers.

2.2.4. Data structure and the particular case of blockchain

A blockchain is a particular type of data structure used in some DLTs. As its name describes it, blockchain transmits and stores data in “blocks”, and connects these blocks to each other in a “chain”, using (a set of) cryptographic techniques[1] in order to form a tamper-resistant[2] chain of transaction blocks. Thus, blockchain creates a continuously growing data structure – to which data can only be appended and from which existing data cannot be removed – that functions as a distributed ledger[3]. This is why the blockchain is often used in public ledger as a way to maintain the integrity/immutability of the ledger in an untrusted environment.

Every blockchain is a distributed ledger, but not every distributed ledger uses blockchain technology to run its system.

New generation of blockchains (e.g. Polkadot) come with a more advanced and complex topology. There is a main chain which is only dedicated to validating the transactions and storing the proofs of the validations. Other blockchains (shards or parachains) based on the same technology are connected to the main chain and are dedicated to application purpose. This topology allows a better scalability and more advanced features (like privacy on public blockchain).

As we have seen, a DLT can be declined in various forms, implying important differences in key features and in its governance model. An entity that is willing to use the DLT technology needs to clearly identify the roles and responsibilities of the various actors being part of its DLT project ecosystem. The next chapter presents the main roles we have identified in a DLT project along with use-case examples.

[1] *i.e. Hashing algorithm or asymmetric cryptography, but not limited to.*

[2] *This is based on current technology and does not presume of future evolutions that could question this characteristic.*

[3] *Cryptocurrencies and blockchain – Legal context and implications for financial crime, money laundering and tax evasion. European Parliament. 2018.*

3. What type of DLT and ecosystem for what project?

Depending on the type of DLT chosen and/or the configuration used, important properties are impacted, which also implies appropriate governance and processes in terms of the roles and responsibilities of all parties involved.

And the other way around, depending on the business use-case and its ecosystem, the choice of a particular type of DLT rather than another will be relevant.

This chapter also presents some use-cases observed in the financial sector for illustration purposes.

3.1. Impacts of the type of DLT chosen

The state-of-the-art DLTs support various configurations of the characteristics mentioned in the previous chapter of this white paper. For example, a permissioned ledger could be either private or public. A blockchain-backed financial service that constrains network participation as well as access to records is both permissioned and private. In contrast, a blockchain-backed financial service that allows anyone to submit transactions but controls the identities that could be part of the network is permissioned and unrestricted.

The choice of these characteristics impacts important properties such as transparency, performance and governance.

Thus, a private ledger is often governed and hosted by a single organization and allows for a relatively flexible configuration as compared to a public ledger that aims to provide equal rights, greater transparency and auditability. Similarly, permissioned ledgers allow recording larger amount of transaction details, and allow specifying fine-grained policies wherein some participants may view only abstract information while others, such as auditors, have access to broader range of transactions. Such granular permissions management may not be feasible in case of permissionless blockchain.

On the contrary, certain design choices made in permissionless blockchain become non-essential for scenarios using permissioned blockchain.

For instance, Proof-of-Work is suitable for bitcoin (permissionless blockchain) to counter Sybil attacks[1]; but, in case of permissioned blockchain, since the identity of each node is known, Sybil resistance becomes superfluous and a more cost-effective alternative such as a threshold signature scheme may suffice.

The risk assessment of the provision of financial services by a regulated entity through a DLT needs to be adapted to these different types of DLT architectures since each configuration entails a different set of risks, although some risks remain inherent to the use of DLT itself. The risks to be considered by entities looking to use a DLT are discussed in chapter 4 of this white paper.

3.2. Roles and responsibilities in a DLT

Participants in a DLT ecosystem take different roles and provide varying sets of functionalities.

	DLT Developer	Infrastructure Service Provider	Solution Provider	User	End-User
Role	Develops the application code for running the DLT	Delivers the infrastructure on which the DLT runs	Designs the business solution which is based on the DLT	Uses the software developed by the Solution Provider in order to provide services to the End-User	Uses the software developed by the Solution Provider in order to access the services provided by the User
Responsibilities	Provides updates, fixes the code	Operates and maintains the infrastructure	Writes the code of the business software solution	Input data in the software provided by the Solution Provider. Optional: Secures the credentials (passwords, private keys)	Inputs data in the software provided by the Solution Provider
Contractual Relationship	Provides services to the Solution Provider	Provides service to the Solution Provider	Provides services to the User/End-User	Provides services to the End-User of the solution.	n/a

Technical role Business role

Figure 2: Main roles in a DLT ecosystem[2]

[1] *A Sybil attack is an attempt to control a peer network by creating multiple fake identities. To outside observers, these fake identities appear to be unique users. However, behind the scenes, a single entity controls many identities at once. As a result, that entity can influence the network through additional voting power in a democratic network, or echo chamber messaging in a social network. (Bennett Garner, Coincentral.com)*

[2] *Icons from www.flaticon.com*

The CSSF has identified the following main roles in the implementation of a DLT-based solution.

The DLT developer

The DLT developer develops the application code for running the DLT. These are often open source developments.

- **Responsibilities**: The DLT developer can be compared to the developer of an operating software. As such, he has to provide updates and fix the vulnerabilities or bugs when they are detected. He is in charge of optimizing the code and developing new functionalities that can be useful for the solution provider.
- **Contractual relationship**: the developer provides a service to the Solution Provider. In the case of a public blockchain, there is no formal contractual relationship as the DLT software is open-source and can be used or modified by anyone.

The infrastructure service provider (ISP)

The infrastructure service provider (ISP) delivers the infrastructure on which the DLT runs. A typical role of the ISP will be to provide nodes on which the ledger is distributed.

- **Responsibilities**: the ISP responsibilities are to operate and maintain the infrastructure on which the DLT components run.
- **Contractual relationship**: In a private DLT, the ISP typically provides a service to the solution provider by operating a node. In the case of a public DLT, service providers are free to join and leave the network so there is no contractual relationship possible; service providers such as miners are participating to the network validation based on the incentive of earning block rewards.

The solution provider

The solution provider (or software designer) is the designer of the "business" solution which is based on the DLT. He develops applications for (end-)users to access the distributed ledger and use the business solution.

- **Responsibilities**: The solution provider writes the code of the business software solution that uses the DLT technology developed by the DLT developer. Other solutions developed can allow to store tokens and send transactions to the network (wallet software), use smart contracts for automation if the DLT offers this possibility and interact with other software through API and/or oracles[1].
- **Contractual relationship**: the solution provider provides a service to the users (and/or end-users) of the solution. The applications developed by the solution provider can be hosted and managed by himself, by his clients (e.g users or end-users), by third parties (e.g Cloud hosting, servers, etc..) or by ISP (e.g applications running through smart contracts).

The users

This participant is the user of the software developed by the solution provider.

In the case of a fund distribution platform developed on a distributed ledger, the users would be regulated entities such as management companies, fund accountants or transfer agents which contribute to the lifecycle of a fund.

- **Responsibilities**: The users are responsible for the accuracy of the data they input in the software offered by the solution provider. Depending on the platform model, they could also be responsible for the security of the credentials (safe storage of passwords, private keys).
- **Contractual relationship**: the users provide a service to the end-users of the solution.

The end-users

In the previous fund distribution platform example, the end-users would be the investors of a fund managed on the DLT-based fund distribution platform.

- **Responsibilities**: The end-users are responsible for the accuracy of the data they input in the software offered by the solution provider. Depending on the platform model, they could also be responsible for the security of the credentials (safe storage of passwords, private keys).

[1] *Refer to the definition of Oracle in part 4.2.3 Smart Contracts (question Q14)*

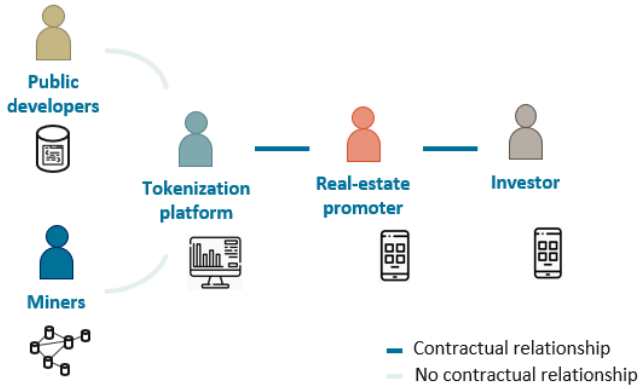


Figure 3: One of the many models of contractual relationships in a public DLT ecosystem

The above-mentioned roles are not mutually exclusive. As such an entity could cumulate the role of solution provider and infrastructure service provider. There can also be multiple providers for the same role: in a typical distributed environment, it is expected to have multiple nodes operated by various infrastructure service providers. Also, an entity could cumulate the role of infrastructure service provider by running a node and the role of a user by using the software based on the distributed ledger it stores on its node (e.g. a wallet).

When assessing the risks in the provision of financial services through a DLT, **it is therefore essential first to identify the participants as well as their role(s) and responsibilities including potential conflicts of interest that may arise when a participant cumulates multiple roles.**

3.3. Examples of use-cases

The purpose of the present section is to give a general overview of a sample of DLT applications observed over the past few years, to allow the reader to have a concrete view on how the DLT is or may be used. The proposed sample is not exhaustive and the CSSF acknowledges the fact that many more DLT applications have been developed or could be developed in the future.

It shall further be noted that the CSSF is not hereby providing a positive assessment of the below-listed use-cases in any way whatsoever. It only aims to draw the reader's attention to the way the technology itself is used. As raised earlier in the present document, in line with its constant application of the technology neutrality principle, the CSSF focuses on the services provided and ensures that the way the proposed technology is implemented fits with the relevant security and regulatory requirements.

Each business model is unique and must be assessed on a case by case basis. The development of a business model similar to any of those listed below does not exempt the institution from seeking the prior assessment/approval of its activities, services and products by the CSSF nor, as the case may be, the appropriate license or registration. Finally, the institutions should also ensure compliance with legal and regulatory requirements on outsourcing (including CSSF circulars) when applicable.

Use-case 1 – KYC

General description: Professionals subject to the AML/CFT obligations as well as customers required to be identified can benefit from KYC data management operated through the use of a DLT. Customer identification checks can be performed by authorized entities and the verified status of customers' digital identity can be shared between institutions.

Purpose: The objective is to have a verified digital identity accessible to various third parties thereby avoiding duplication of data collection and verification efforts by various authorized professionals. It is recommended to not store sensitive documents or data used to verify the identities in the DLT but to maintain their verified status in the DLT.

DLT use: The DLT is used to confirm an identity claim through cryptographic proof.

Configuration example: A conglomerate of banks decides to run a Corda permissioned DLT in which their customers' digital identities can be checked and maintained by the different banks. The conglomerate outsources the development of the software to an established ICT (Information and Communication Technology) service provider and the DLT nodes are hosted in the Cloud and managed by the banks' IT department. In this example, banks have at the same time the roles of ISP and (end-) users. The solution provider is the ICT provider to which the software development is outsourced and the DLT developer is R3 (developer of the Corda DLT).

Use-case 2 – Transfer of funds and assets

General description: Processing payments and transferring funds and other cryptographic assets cross border is another possible application of the DLT. Transfers are settled directly from the originator to the beneficiary without requiring the use of intermediaries. Funds could be transferred without going through clearing and settlement houses and the traditional correspondent banking network.

Purpose: The use of a DLT can improve the overall efficiency of the payment systems, allowing the fast, automatic and secure processing of payments, 24 hours a day, 7 days a week at a low cost.

DLT use: Debits and credits are recorded in real time in a DLT. Payments can be made using crypto-assets such as stable-coins denominated in an official currency. The DLT offers the possibility to include more parameters and conditions to the execution of the payments through the use of smart-contracts. Transfers can be executed automatically based on pre-agreed conditions or settled according to the execution pattern of the contractual agreement underlying the payments.

Configuration example: Bank A and Bank B use a stablecoin issued on the Ethereum public blockchain to settle transactions. In this example, Banks A and B are the users. The solution provider is the stablecoin issuer and the DLT developer is the Ethereum Foundation. The ISP are the miners/nodes owners of the Ethereum public blockchain. In this specific example relying on a public DLT, it has to be noted that there is no contractual relationship between the users and the ISP and the DLT developer.

Use-case 3 – Fund distribution platform

General description: A DLT-based distribution platform allows the tokenization of investment funds in which investors can subscribe and redeem their fund shares through a web or mobile application. The DLT distribution platform allows a direct access to the management company's products for the investors.

Purpose: The use of a DLT can reduce time-consuming tasks such as reconciliation with automation, reduce the intermediation costs between investors and management companies, mutualize costs amongst the fund distribution participants, increase the resiliency of the platform and offer better transparency of transactions.

DLT use: An investor account would be assimilated to a wallet on a DLT and the funds he is investing in would be tokens stored in this wallet. Subscriptions and redemptions could be submitted directly by the investors in a digitalized way. The investor would be authenticated and would validate transactions through cryptographic means. Processing of these subscriptions and redemptions in funds would be automated with the help of smart contracts. History of the transactions would be immutable and accessible to authorized parties (for instance the regulator, auditors, the investors or the Management company).

Configuration example: A company offers a platform on which fund shares are tokenized and where investors can buy them directly. On the other hand, management companies have the possibility to issue tokenized funds on this platform. The platform is powered by smart contracts running on a permissioned Quorum DLT. Quorum nodes are hosted in the Cloud and operated by the company developing the fund distribution platform.

In this example, the DLT designer is Consensys (developer of the Quorum DLT). The company developing the fund distribution platform has the roles of solution provider and ISP (e.g nodes provider), the users are management companies issuing their funds on this platform and the end-users are the investors.

As explained in this chapter, a DLT project involves different actors with different roles and responsibilities. Depending on the setup, an actor can cumulate roles and responsibilities. We have also seen that the type of DLT chosen has an impact on the governance and the performance of the DLT. In the next chapter, we will analyse the various risks that a regulated entity that plans to use the DLT to provide its services needs to consider: we will focus first on DLT specific risks, then as a second step we will review traditional ICT risks that also need to be considered by entities looking to launch a DLT project.

4. Risks to be considered by entities looking to use a DLT

As previously mentioned, the risk assessment of the provision of financial services by a regulated entity through a DLT needs to be adapted to these different types of DLT architectures (e.g. a public - permissionless DLT does not entail the same risks than a private - permissioned DLT), even if some risks are inherent to the use of DLT itself, regardless of its architecture.

When assessing the risks in the provision of financial services through a DLT, the regulated entity should especially focus on the elements presented below. Whatever the different actors who are part of the DLT solution are, the regulated entity should understand these elements and should ensure their appropriate coverage, either directly or indirectly with its subcontractors / partners (solution provider, infrastructure service provider, DLT designer, etc…).

4.1. Governance aspects

4.1.1. DLT strategy

Q1 – Is the use of a DLT justified?

The choice to use a DLT technology to support the provision of financial services is first a strategic decision to adopt a new business model which redistributes roles (disintermediation) and relies on higher collaboration (mutualization, reliance on a trusted network, etc.).

This business strategic decision should also be taken with due consideration of the risks linked to distributed ledger technologies. DLT is relatively complex and poses challenges that should not be underestimated. **An entity should weight the risks that involved the use a DLT against the benefits**. The risk analysis should at least cover strategic risks, legal and regulatory risks, security risks, performance risks, confidentiality risks linked to the use of a DLT.

The use of DLT without the adequate governance and prior study and testing could have a negative impact leading to reputational risks, and expose institutions to potential poor service, poor user experience and sanctions for non-compliance that could negatively affect the overall reputation of the financial sector. In some cases, the lack of Service Level Agreements (SLAs), the breach of agreed service levels, or improperly managed recovery mechanism, among others, could potentially damage the reputation of the institution.

Q2 – What DLT model to choose?

A public permissionless DLT has a different risk profile than a permissioned private DLT. For instance, in a public DLT, an entity will face challenges in regards to key management, confidentiality, variable cost of transactions, smart contract management and transaction throughput that will not be as critical in private permissioned DLT.

The DLT model will need to fit with the business needs and the regulatory requirements applicable to that entity.

An important decision one needs to make while designing or choosing a DLT model concerns its access rights, creation rights and validation rights. This decision depends on the specific financial services implemented and various other parameters such as the stakeholders' goals, constraints and trust requirements, access control requirements, as well as the types and sensitivity of the transactions recorded in the distributed ledger.

Another key feature is the consensus mechanism that should be carefully assessed during the analysis of the DLT model. Different consensus mechanisms lead to different balances among security, scalability or decentralisation[1].

4.1.2. Governance framework

Q3 - What is the governance framework in place to manage changes at the DLT level?

There are many different change possibilities for DLT such as updating blockchain parameters/rules (e.g. block size), changing consensus mechanism (e.g. from PoW to PoS). **An entity should assess how those changes at the DLT level can potentially impact the continuity and the validity of its business**.

- In case of a public DLT, a solution provider has to be aware that new rules or changes to the DLT and decided by the DLT developer, which is outside its control, can impact the way it operates on the DLT.

[1] *Often referred to as the 'Blockchain Trilemma'*

https://www.gemini.com/cryptopedia/blockchain-trilemma-decentralization-scalability-definition#section-what-is-the-blockchain-trilemma

- In case of private DLT, a governance model should be defined to frame how DLT changes are managed taking into consideration the business continuity needs of the solution provider. The governance model should cover how infrastructure service providers are supposed to implement these changes, notably by upgrading software versions of the nodes they maintain.

4.1.3. Legal and contractual points

Q4 – Do the planned DLT-based activities, services or products require a license or registration from the CSSF?

Irrespective of the technology used, the nature of the activities, services or products offered may imply the obligation for the entity to seek a license or registration from an authority. Also, a financial institution already licensed and supervised by the CSSF may decide to evolve its business model with the use of DLT, with the possible consequence of having to obtain an additional or different license required for the new services offered.

For instance, entities established in Luxembourg or providing services in Luxembourg may not provide virtual asset services without being registered with the CSSF[1].

Depending on the project, one or several licenses may be necessary. The institution developing a DLT project will have to assess its compliance against current and future regulations and ensure that it has the necessary approvals to operate at all times.

Q5 – Are the liabilities in case of malfunction and dispute resolution mechanisms clearly defined?

In the dispute settlement for issues arising from the use of DLT, the identification of the person responsible of the proper functioning of the DLT and of the solution running on it is not always clear. This can constitute a risk for the DLT participants. Uncertainty around such liability may also endanger the trust in the system.

Therefore, it is recommended **to define a person in charge of any claims related to the malfunctions of the DLT** at the solution provider, infrastructure service provider and at the DLT developer levels when possible. In the case of a public DLT, the entity has to be aware that it is often impossible to designate a person in charge at the infrastructure service provider and at the DLT developer levels or to contract the relationship contrary to a typical outsourcing relationship.

[1] https://www.cssf.lu/en/registration-vasp/

If there is a contractual relationship between all participants, the liabilities, **the dispute resolution mechanisms and the choice of the applicable jurisdiction** in case of dispute should be formalized.

Q6 – Does the entity have the capacity to comply with laws and court decisions?

An entity must **consider the risk of having to enforce court decisions** and block access to assets stored in a DLT. In the same vein, an entity should have **procedures and follow the laws for unexpected events** such as the death of a client to be able to recover the client's assets stored in the DLT.

This can be impossible to do in the case of a public DLT where the end-user stores himself the private key allowing access to his assets. This risk linked to private keys' unavailability / inaccessibility is further discussed in section 4.2.4. on key management.

Q7 – What is the legal certainty of the use of DLT and smart contracts?

The legal and regulatory framework for the use of DLT can vary between jurisdictions and should be analysed. Also, if smart contracts are used, **their legal effect and interpretation should be clear, recognised and formalised**. The responsibility for their proper functioning should be clearly assigned (refer to Q5).

4.2. DLT-specific technical risks to consider

4.2.1. Distributed ledger design

All distributed systems use consensus algorithms. The goal of a distributed consensus algorithm is to allow a set of computers to all agree on a single data value that one of the nodes in the system proposed. The challenge in doing this in a distributed system is that messages can be lost, machines can fail or participants can cheat.

Q8 - Has the consensus algorithm been formally tested for correctness of operation and the management of shortcomings?

In a DLT environment, a consensus algorithm is used to validate transactions from the participants. From its quality depends the success or failure of the consensus and, then, the DLT itself. The solution provider should **verify whether this consensus algorithm has been formally tested for correctness of operation and how shortcomings are handled** by this consensus mechanism (for instance in the case of node failures, faulty or malicious nodes).

The level of risk is different between a well-documented and proven algorithm and a proprietary algorithm:

- When a well-documented and proven algorithm is used, the information could be made available.
- However, when a proprietary algorithm is used, additional checks may need to be done:
    - Formal proof of the correctness in operation of the algorithm;
    - Convincing and reproducible experiments for the performance of the system under realistic failures (node failures, faulty nodes).

Q9 - What are the mechanisms to assure the sharing of capacity and the quality of service between actors?

The solution provider should **consider how the DLT is designed in terms of transaction throughput** and that the DLT capacity (in terms of volume of transactions) fits with the business needs. It is especially important in the case of public DLT in which the transaction processing priority is often associated with transaction fees. In case of heavy demand on public DLT or when the underlying cryptocurrency sees its price raising quickly, the transaction fees can become prohibitively high[1].

Quality of service in a DLT is defined by the assurance that a transaction will be executed in a certain timeframe. A DLT network delivers a certain capacity of transactions per second. If the DLT is used at full capacity, **is there any mechanism that assures the equitable sharing of capacity** and that no actor will abuse the capacity and thus block the other actors (denial of service)? As an example, in a public DLT the competition for blockchain capacity can be assured by the fact that there is an economic competition between the miners and all transactions need to pay a fee to be processed. A higher fee per transaction assures a higher priority for a transaction to be included in the DLT. A natural competition assures quality of service between parties and a sharing of the capacity. A downside of the PoW model is that the cost per transaction will increase in case of speculation on the token used to pay for capacity or if the demand for transactions is higher than the available capacity.

[1] *https://www.coindesk.com/ethereum-developers-consider-new-fee-model-as-gas-costs-climb*

The DLT developer should explain and the entity should know and understand:

- The **transaction acceptance management mechanisms** and how the DLT is supposed to behave in case of overload (i.e. when the number of transactions exceeds the capacity of the DLT); the **impact of network latency** on transaction integration performance;
- What are the mechanisms available to **increase the performance of the DLT** if needed in the future;
- How does the consensus method **prevent major players from monopolising the validation** of transactions;
- How **changes and new functionalities** are chosen, tested and added to the DLT.

4.2.2. Nodes management

In a network of DLT nodes, **it is essential to have a proper governance and topology for node distribution and node management**. Nodes are needed to participate in the network of the DLT and to be able to write and read data to/from the DLT. All - or only a subset of nodes - participate in the consensus mechanism. The topology of the network should be well designed, configured and tested to assure the continuity of service for a particular actor (i.e node owner) or for the whole DLT network, in case of a failure of infrastructure elements or of a whole actor.

As an example, in the case of permissioned DLT the number of nodes participating in the consensus is limited due to performance but needs to be sufficient to guarantee continuity. If the number of nodes falls under a certain minimum due to infrastructure or configuration problems the whole DLT will stop working.

In a public blockchain, an example would be a denial of service attack due to errors in the DLT software, which can slow tremendously the execution of transactions and need immediate update of software and resynchronization of the nodes' ledger.

In 2013, the Bitcoin network almost unexpectedly forked because of divergent node software.

A solution provider should be able to demonstrate a sufficient level of understanding of how the DLT works and therefore be able to identify risks related to nodes and to address them properly.

Q10 – How a sound governance of the nodes is ensured in a private DLT?

In the case of private permissioned DLT, procedures and rules should be formalized in order to ensure a sound governance of the nodes. As such, the following points should be covered to ensure appropriate controls and liability over key nodes management actions:

- **Adding** nodes, **deleting** nodes, **rebalancing** after market operation (merger and acquisition);
- **Updating node software**, by ensuring that all the nodes in the network run with a compatible version of the software and no fork of the chain is inadvertently created;
- **Identification of the participants** to the node network;
- The solution provider should also **know the minimum number of nodes needed** for the consensus mechanism to work and also have measures in place to ensure that this minimum number of nodes is in place at all times.

Q11 – What are the mechanisms in place in a public DLT to address risks linked to nodes management?

Having procedures and governance over nodes in a public permissionless DLT is by definition impossible as a public DLT is a decentralized network that anyone can join. However, the solution provider should still assess and cover the following risks:

- **What is the mechanism to manage a split-brain** and avoid double spending and unwinding of valid transactions and how is this managed in the use-case presented?

  A split-brain of the DLT network is when a network failure will result in two independent networks of nodes without any possibility to communicate between both resulting networks. On a public blockchain, the chain would split in a case of a split brain and when the networks re-connect the longer chain would take precedence. As a same transaction may be present in multiple branches of the chain, all transactions on the orphaned chain from the time of the split would be invalidated and are either put back in the pool of transactions that have to be validated, or deleted if they have already been validated.
- What are the procedures or mechanisms in place to **detect and isolate malicious nodes**?
- What are the procedures or mechanisms in place to **detect and isolate a denial of service** from one of the nodes? For instance, nodes could be overloaded with read accesses, preventing the processing of transaction requests in a timely manner.

4.2.3. Smart contracts

Developing a smart contract is very different than developing code for a back-end application. A bug in a smart contract can result in the loss of assets by end users. **The main questions around smart contracts are related to the validation and auditing of the code before its deployment**. How is a smart contract validated? What are the frameworks and standards used to ensure the integrity and security of smart contracts?

Depending on the type of the blockchain (permissioned or permissionless), the way smart contracts' deployment is handled can be quite different.

On a public permissionless blockchain, anybody can deploy a smart contract and the code is visible to anyone including hackers looking for a potential exploit. A conscious attention should be given to the access control to the smart contract and/or the proxies in the case of upgradeable smart contracts.

Each blockchain has its own vulnerabilities. The least the blockchain is used, the more vulnerabilities can be encountered. On the other side, the least it is used, the least probability of attack it has.

In order to improve the security of smart contracts, **multiple resources and best practices have been developed**:

- Frameworks for smart contract development, such as the one from Consensys[1].
- Some tools to perform static analysis against the most common vulnerabilities. An example is Slither[2].
- Standard design patterns, that should also be used to allow among other capabilities whitelisting, upgradeable smart contract, ERC20-Token-Vault or Role Based Access Control. Libraries like OpenZeppelin[3] or DAppsys[4] provide standard, tested and community reviewed reusable code for that.

[1] *https://consensys.github.io/smart-contract-best-practices/*

[2] *https://github.com/crytic/slither*

[3] *https://openzeppelin.org/*

[4] *https://dappsys.readthedocs.io/en/latest/*

On permissioned blockchains, the same precautions, best practices and tools should be used. In addition, a governance layer should be put in place to define a standard process to allow any new smart contract to be deployed on the blockchain. A layer of security should also be set up to only allow an authorized person to deploy its smart contract.

Q12 - Smart contract: how is the code properly developed, audited and validated before deployment?

When a regulated entity intends to use, either directly or indirectly (e.g. through a third-party provider), an application deployed with smart contracts on a DLT, it is important that the following points are assessed by the regulated entity as part of its risk assessment.

- Describe the **tools and standard of development** that are used to code the smart contracts and limit the risks of vulnerabilities in the code.
- Describe the **practices in terms of code review**. Is the code quality of the smart contracts automatically scanned? Has the code of smart contracts been audited by independent experts to eliminate all security flaws?

Q13 - Smart contract: Does the code deployment process allow to ensure continuity and quality of service?

- Explain the best practices used for ensuring **secure and authorized deployment** of smarts contracts in a DLT.
- Describe whether the employees from the business side have been involved in the **testing** to ensure that smart contracts fulfill their purposes.
- Describe the **controls and processes in place to ensure continuity of service** and transaction performance of existing services when deploying or updating smart contracts:
    - Participants may have the right to deploy smart contracts with the consequence that their execution times penalize the performance of the other participants and make the service unusable;
    - The deployment of new versions of existing smart contracts in an active network and under load should be done in a timely manner to ensure continuity of service. Indeed, incoherence in data could lead to a stalled DLT (nodes do not apply the same smart contract and therefore are unable to agree on the data to add to the ledger).
- Describe the **resolution process** in case of incidents or with smart contracts' malfunctions.

Q14 - Smart contracts – How are risks linked to Oracles managed?

An **oracle** can be defined as "a node of the DLT network that certifies to other nodes the occurrence of specific events outside the network (e.g. change in asset prices, weather conditions, etc.)[1]. The use of an oracle is necessary for DLT to communicate with external sources, whether retrieving data from outside the DLT or sending out data from the DLT to external resources. Oracles are essential for dApps[2] such as decentralized exchanges which use them to get the token prices during swaps or prediction markets to determine the outcome of an event and pay out the winners[3].

- **How are the "oracles" chosen**, if they are used in smart contracts? In some cases, the execution of smart contracts needs input from oracles, which are assumed to be trusted data sources. If something goes wrong with the oracles, then the execution of smart contracts will also be jeopardized.
- **What are the controls or remediation methods available in case of problems with an oracle** (wrong data entered by the oracle, oracle not entering any data unexpectedly, etc…)? An oracle can become a single point of failure as smart contracts rely on the data they post in the blockchain to function properly. A solution provider should be prepared for scenario where oracle is not working as expected.

4.2.4. Key management

In both public and private DLTs, a user account is defined by a pair of cryptographic keys where the public key is the username and the private key is the password. Key pairs are essential for identification, signing transactions and ultimately proving ownership of assets recorded on the DLT. In case of loss of the private key, the assets will remain on the blockchain but will be unreachable and unrecoverable by the owner or anyone else, which is a **very high-risk** unknown in the conventional financial sector.

[1] *Definition from "The use of a DLT in post-trade processes", published by the ECB in April 2021*

[2] *Decentralized applications (dApps) are applications that run on a blockchain network.*

[3] *https://ethereum.org/en/developers/docs/oracles/*

Q15 – What are the processes for key generation and delivery to an identified user?

The solution provider should describe the mechanisms to generate and safely deliver key pairs to the users of the DLT solution.

The solution provider should ensure a strong relationship between a customer's keys and his identity. **The solution provider should demonstrate measures and controls in place to protect the information about the ownership of the assets, i.e. the relationship between public key and real identity**.

It is also reminded that AML/CFT regulation should be fully respected at all times when using DLT. A proper identification of the users of the DLT solution and especially the end-users (investors, customers) is required. An entity using a DLT solution should be able to describe the KYC processes at client onboarding but also on an on-going basis in relation to KYC and transaction data.

Q16 – How are the keys stored & managed?

How are private keys managed? A solution provider in charge of customers' private keys' storage should describe the **key storage mechanisms** and tools used for this purpose (smart contracts, multisig wallets, cold wallets, etc...).

An entity should describe the **type of wallet** used (software/hardware), its **mode of operation** and the **security mechanisms** to prevent theft/corruption/loss of the private keys stored in the wallet.

- When a solution provider stores its customers' private keys and receives transaction instructions from its customers through an interface (API, mobile application, web interfaces, etc.), the solution provider should implement a **strong customer authentication** mechanism to verify the customer identity linked to the private keys. An entity should be able to describe the authentication mechanisms (2FA, certificates, behavior analysis, geofencing, etc…) and processes (recovery of password, accounts) in place **to prevent unauthorized access** to the solution.
- When the end-user is responsible to save his private key, the service provider should describe the mechanisms, software or procedures in place to **help the end-user managing and storing this private key safely**.

Q17 - What are the procedures and tools in place in case of lost keys or stolen keys?

The service provider should describe the procedures and tools in place in case of lost keys and **whether there is a possibility to recover** the lost keys.

The service provider should describe the procedures and tools in place in case keys get stolen.

Providers of hardware wallet such as Trezor or Ledger recommend to have **multiple hardware wallets** that will act as duplicates of each other in order to have a backup of the hardware wallet.

When possible, the assets should not be directly owned by the customer's address which incurs the risk of losing access to the assets if the customer's private key is lost. Using a **multi-signature wallet** is a solution that allows multiple signers to access the assets and it offers the possibility of asset recovery if one key is lost.

4.2.5. Privacy & user identity

Q18 – How are privacy rights and needs addressed?

Privacy concerns in a DLT should be carefully analysed with regards to the applicable laws.

For instance, public blockchains allow anyone to consult the transactions performed by anyone on the network and this history is stored forever in the ledger due to the immutability principle of a blockchain. As such, being able to link a public key with the real identity of the participant would allow to scrutinize the complete transaction history of this participant. Due to the **lack of unlinkability**[1], blockchain such as Bitcoin can be qualified of being pseudonymous rather than anonymous. As discussed in previous sections, strong controls can be implemented to avoid that such privacy and confidentiality risks materialize.

However, a regulated entity should perform its own risk assessment and:

- Define **what data to store** inside and outside the DLT;
- Assess **legal implications** including those related to GDPR;
- Take the adequate measures to **protect customer data including elements** that would make possible **to link a customer identity to a public address**.

A summary of all the DLT-specific key questions & considerations explained in this section can be found in part 6 – Appendix.

[1] *https://arxiv.org/pdf/1903.07602.pdf*

4.3. Other traditional ICT (Information and Communication Technology) risks

Some risks are not specific to the use of a DLT but should still be covered during the risk analysis. Thus the regulated entity should also assess and describe how general IT controls such as the ones described below (non-exhaustive list) are addressed in the specific context of their DLT project.

4.3.1. Governance

- **Outsourcing relationships** should be formalized with SLAs in line with CSSF's requirements and outsourcing relationships should be monitored through regular KPIs and meetings.
- **Outsourcing and concentration risks** should be assessed. Identified single point of failure should be mitigated.
- Procedures should be formalized to ensure a sound governance especially for **dispute resolution**. In terms of organization, a person should be designated as a responsible for disputes' settlement.
- An **exit strategy** should be formalized and cover various scenarios such as a user leaving the DLT solution or changes in the regulation. Issues that might arise from the immutability principle of a blockchain (impossibility of erasing data) should be assessed upfront.

4.3.2. Continuity & Resiliency

- **Resiliency measures** should be taken to ensure that the IT infrastructure supporting the DLT solution is resilient:
    - At the data centre level;
    - At the server level;
    - At the network level.
- In public DLT the resilience is assured by the high number of nodes participating in the network. A subset of nodes can be down without impact on the DLT. Still the software using the DLT needs to be able to have access to a healthy node to assure the service.
- In the case of a private network the number of nodes is limited and a minimum of nodes are required, to ensure that the services provided can still access a node and that a sufficient number of nodes are operational to ensure the quality of service.

- The resiliency and availability measures should allow to **minimize the impact on the DLT and the auxiliary applications in case of a major system breakdown**.
    - The resiliency measures should cover all systems involved and not only the servers hosting the nodes that are supporting the DLT. Most DLT solutions require web servers, API, middleware and database servers for the DLT solution to be operational.
    - As such a **BCP plan** should be formalized and cover the risks of major incidents such as failures at network level, validator nodes' unavailability, unavailability of one of the datacentres hosting the DLT nodes or impacting auxiliary services required for proper business operation.
    - In private DLT, it is recommended to **spread the nodes** in multiple locations/geographic regions to avoid a risk of single point of failure (i.e if all nodes are hosted in the same data center). The entity should take the necessary measures to ensure that decentralizing the nodes will not impact the performance of the DLT in terms of latency and transaction execution times.
    - **Recovery point objectives (RPO) and Recovery time objectives (RTO)** should be defined for each system supporting the DLT solution. High availability setup is needed for critical services.
- Tools and mechanisms should be in place to monitor hosts, software, network, DLT, data centres and to measure service availability and quality. Mechanisms should also be implemented to detect a corruption of the DLT database (due to a hardware failure or software bug) and repair them in a timely manner to limit their impact on the use-case.
- The solution provider writes the code of the business software solution that uses the DLT technology developed by the DLT developer. The solution provider should **consider the risk of lock-in**: how can the solution provider guarantee the continuity of the service if the DLT developer stops supporting/developing the DLT technology/software?
- An entity should explain the mechanisms and **fallback solutions** to ensure access to transactions and asset ownership information if the DLT is no longer operational.

4.3.3. Security & cybersecurity

- A solution provider should have **mitigation strategies and measures to prevent cybersecurity risks** such as phishing, ransomware, man in the middle, social engineering attacks, etc… In a DLT project, a special attention has to be given to the storage of private keys that allow to perform transactions on the DLT.
- **Mobile applications**: A solution provider should describe the security management of mobile applications, authentication and communications between the application and the web services servers/DLT components.
- A solution provider and/or service provider should describe the **hardening of the servers** (including the ones hosting nodes), security management, configuration management and monitoring of the servers.
- A solution provider should describe security management of **software components**, configuration management and communications between software components.
- Solution provider and ISP should describe the controls and tools they implement to **prevent unauthorized access** to IT resources. Additional caution should be used to protect high privileged accounts and accesses to critical services.
- **Services exposed to Internet** should be subject to additional scrutiny and monitoring.
- A solution provider and/or ISP should describe security of **network communications** and monitoring, interfaces with external systems. What data is exchanged? How is data securely exchanged? How are the integrity and confidentiality needs addressed?
- A solution provider should formalize a **patching** policy and patch management software.
- **Datacenters** hosting servers for the DLT solutions should have security measures in place to prevent unauthorized access, theft and data loss.
- **Encryption**: for data in transit, what are the encryption algorithms/certificate standards and key sizes used? A solution provider should be able to describe the encryption mechanism in place from end to end for transaction management and the keys/certificates life cycle.
- In private DLTs, a **security governance** should be established and should apply to all participants hosting DLT nodes. Notably, **security incidents** on nodes need to be collected and notified as appropriate to each node to guarantee a continuous enhancement of security policies and procedures.

4.3.4. Change management

- The solution provider and ISP should implement a sound **software delivery process and tools**. This software delivery process should cover new versions of any software in the DLT ecosystem (node, auxiliary services, wallets, etc).
- When developing a new version of a software, a formal **testing process** should be followed which involves the business users.

5. Conclusion

The DLT combines different existing technologies and can be developed with various set-ups and functionalities, depending on the needs arising from the project relying on it. Each DLT has its own features and specificities. As a result, the DLT can be a complex technology to apprehend.

Professionals are therefore invited to make a proper assessment of the risks related to the development, implementation and use of a DLT. The list of risks detailed in the present white paper is not, and does not intend to be, exhaustive and may not perfectly fit with all types of activities. However, the present white paper raises questions that may, at the very least, be addressed by the professionals and may help them in the design and development of their DLT project.

The CSSF acknowledges that, when properly used, a DLT, like any other innovation, can bring advantages and opportunities to the financial sector. Some, even, talk about a future DLT revolution. However, its integration to a constant-moving environment constitutes a real challenge, and not only for the institutions. Integration of evolving technology-based innovation in financial services and markets is a continuous challenge for regulators such as the CSSF.

Within this backdrop, in order to gain the best possible understanding of innovative developments and expectations of the industry and to address the forthcoming challenges, the CSSF is in permanent contact with market players. The CSSF is thus promoting a constructive and open dialogue with the financial sector.

In that view, the CSSF remains open to consultation and exchange regarding the development of DLT-based projects for the financial sector and the application of regulation and encourages market players to contact it in order to either present an innovative project, request information on the regulatory framework applicable to a project or to initiate a dialogue on new technologies or regulation that may impact the financial sector. Any contact request must be made via the following e-mail address: innovation@cssf.lu.

For further details on the CSSF's approach to financial innovation, we invite you to read our Communication on Financial Innovation, using the following link: https://www.cssf.lu/wp-content/uploads/C_Financial-innovation_February-2021.pdf.

6. Appendix: Summary of DLT-specific key questions & considerations

Governance aspects

DLT strategy	
Q1 – Is the use of a DLT justified?	- Perform a general risk & reward analysis from using a DLT Identify the key risks to mitigate and the key benefits to achieve.
Q2 – What DLT model to choose?	- Compare the different DLT models (key features and risk profiles) and select the most appropriate to the project.
Governance framework	
Q3 - What is the governance framework in place to manage changes at the DLT level?	- Assess to which extent a control on the DLT parameters & rules is required in order to define the governance model. - Assess the impact of a change of governance on the service delivery, especially in case of a permissionless environment.
Legal and contractual points	
Q4 – Do the planned DLT-based activities, services or products require a license or registration from the CSSF?	- Assess whether any licensing or registration requirements apply to the proposed activity, service or product and if so, take immediate steps to comply.
Q5 – Are the liabilities in case of malfunction and dispute resolution mechanisms clearly defined?	- Identify the persons in charge of claims or malfunctions and the applicable jurisdiction.
Q6 – Does the entity have the capacity to comply with laws and court decisions?	- Take into account the risks (e.g. inability to freeze/recover assets) during the definition of the private key management strategy.
Q7 – What is the legal certainty of the use of DLT and smart contracts?	- Analyse and disclose the legal effects from the use of DLT and smart contracts.

DLT-specific technical risks to consider

Distributed ledger design	
Q8 - Has the consensus algorithm been formally tested for correctness of operation and the management of shortcomings?	- Obtain this information about the algorithm used. - In case of a proprietary algorithm, obtain a formal proof of the correctness using test cases under realistic conditions.
Q9 - What are the mechanisms to assure the sharing of capacity and the quality of service between actors?	- Analyse the transaction management in a stress test environment (overload, latency) and document any preventive measures. - Identify the mechanism to prevent monopoly of transactions' validation.

Nodes management	
Q10 – How is a sound governance of the nodes ensured in a private DLT?	- Define a node selection & management policy. - Define a node software management policy.
Q11 – What are the mechanisms in place in a public DLT to address risks linked to nodes management?	- Define a split-brain event policy. - Define mechanism to detect/prevent/remedy malicious nodes.
Smart contracts	
Q12 - Smart contract: how is the code properly developed, audited and validated before deployment?	- Use preferably recognized standards and frameworks for smart contract development. - Submit the contracts to an audit review and document the smart contract processes.
Q13 - Smart contract: Does the code deployment process allow to ensure continuity and quality of service?	- Define a contract deployment strategy taking into account the authorisation process, the continuity of service, the disputes' resolution.
Q14 - Smart contracts – How are risks linked to Oracles managed?	- Define the process for the selection and monitoring of oracles. - Define a remediation process and dispute resolution mechanism in case of failure/error/issue.
Key management	
Q15 – What are the processes for key generation and delivery to an identified user?	- Define a process to generate and deliver the encryption key pairs to the customer. - Define a process to protect the data linking the customer real identity and its public key.
Q16 – How are the keys stored & managed?	- Describe the private keys storage mechanisms and tools. - Describe the wallet solution and the security measures to prevent theft/corruption/loss of the private keys stored in the wallet.
Q17 - What are the procedures and tools in place in case of lost keys or stolen keys?	- Describe the procedures and tools in place in case of lost/stolen keys and whether the keys can be recovered. - Assess the appropriate use of backup wallets, multi-signatures or other private key security practices.
Privacy & user identity	
Q18 – How are privacy rights and needs addressed?	- Define what data is stored inside and outside of the DLT. - Assess legal implications including those related to GDPR. - Take adequate measures to protect customer data, including elements that would make possible to link a customer identity to a public address.

7. Bibliography

- World Bank document published in 2017:
  - Distributed Ledger Technology (DLT) and Blockchain, FinTech Note | No. 1, http://documents.worldbank.org/curated/en/177911513714062215/Distributed-Ledger-Technology-DLT-and-blockchain
- White paper issued by ILNAS
  - White Paper Blockchain and Distributed Ledgers – June 2018, https://portail-qualite.public.lu/dam-assets/publications/normalisation/2018/white-paper-blockchain-june-2018.pdf
- ISACA:
  - Blockchain Framework and Guidance - https://www.isaca.org/bookstore/bookstore-misc-ebook/wbfg
- ECB:
  - The use of DLT inpost-trade processes - https://www.ecb.europa.eu/paym/intro/news/html/ecb.mipnews210412.en.html
- European Parliamentary Research Service - Can distributed ledgers be squared with European data protection law?
  - https://www.europarl.europa.eu/RegData/etudes/STUD/2019/634445/EPRS_STU(2019)634445_EN.pdf

8. Acknowledgments

We would like to thank the following persons for their contributions to this white paper:

- Bernard Simon, FundsDLT's CIO & Member of Executive Committee,
- Dr. Qiang Tang, Senior Researcher at Luxembourg Institute of Science and Technology (LIST),
- Fabrice Croiseaux, CEO of InTech and President of Infrachain.
- Prof. (hon.) Jean-Louis Schiltz, Senior Partner Schiltz & Schiltz law firm,
- Emilie Allaert, Project Lead at the Luxembourg Blockchain Lab & Head of Operations and Projects at the LHoFT,
- Christian Dohmen of the Luxembourg Stock Exchange.

Commission de Surveillance du Secteur Financier
283, route d'Arlon
L-2991 Luxembourg (+352) 26 25 1-1
direction@cssf.lu
www.cssf.lu