International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

F.751.1

(08/2020)

SERIES F: NON-TELEPHONE TELECOMMUNICATION SERVICES

Multimedia services

Assessment criteria for distributed ledger technology platforms

Recommendation ITU-T F.751.1

ITU-T F-SERIES RECOMMENDATIONS

NON-TELEPHONE TELECOMMUNICATION SERVICES

TELEGRAPH SERVICE	
Operating methods for the international public telegram service	F.1–F.19
The gentex network	F.20–F.29
Message switching	F.30–F.39
The international telemessage service	F.40–F.58
The international telex service	F.59–F.89
Statistics and publications on international telegraph services	F.90–F.99
Scheduled and leased communication services	F.100–F.104
Phototelegraph service	F.105–F.109
MOBILE SERVICE	
Mobile services and multidestination satellite services	F.110–F.159
TELEMATIC SERVICES	
Public facsimile service	F.160–F.199
Teletex service	F.200–F.299
Videotex service	F.300–F.349
General provisions for telematic services	F.350–F.399
MESSAGE HANDLING SERVICES	F.400–F.499
DIRECTORY SERVICES	F.500–F.549
DOCUMENT COMMUNICATION	
Document communication	F.550–F.579
Programming communication interfaces	F.580–F.599
DATA TRANSMISSION SERVICES	F.600–F.699
MULTIMEDIA SERVICES	**F.700–F.799**
ISDN SERVICES	F.800–F.849
UNIVERSAL PERSONAL TELECOMMUNICATION	F.850–F.899
ACCESSIBILITY AND HUMAN FACTORS	F.900–F.999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T F.751.1

Assessment criteria for distributed ledger technology platforms

Summary

Recommendation ITU-T F.751.1 specifies assessment criteria for distributed ledger technology (DLT) platforms. The purpose of Recommendation ITU-T F.751.1 is to assist implementers in the evaluation and comparison of different platforms. The assessment framework includes a set of criteria for function, performance and other aspects. It can be used as a guideline for DLT platform assessment, as well as for information disclosure of a given DLT platform product.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T F.751.1	2020-08-13	16	11.1002/1000/14333

Keywords

Assessment, blockchain, criteria, distributed ledger technology, DLT, ledger, test.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T F.751.1

Assessment criteria for distributed ledger technology platforms

1 Scope

This Recommendation specifies an assessment framework for distributed ledger technology (DLT) platforms, which includes a set of criteria for function, performance and other aspects. The framework is recommended for use as a guideline for DLT platform assessment, as well as information disclosure of a given DLT platform product.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

None.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 asset [b-ITU-T F.751.0]: A representation of value. It can be a diamond, a unit of currency, items inside a shipping container, etc. An asset can be physical or virtual.

3.1.2 blockchain [b-ITU-T F.751.0]: A type of distributed ledger that is composed of digitally recorded data arranged as a successively growing chain of blocks with each block cryptographically linked and hardened against tampering and revision.

3.1.3 consensus [b-ITU-T F.751.0]: Agreement that a set of transactions is valid.

3.1.4 entity [b-ITU-T F.751.0]: Anything that has a separately identifiable existence (e.g., organization, person, group, smart contract or device). An entity uses distributed ledger technology to solve the problem of its business or information systems.

3.1.5 entity address [b-ITU-T F.751.0]: Identifier for one or more entities performing transactions or other actions in a blockchain or distributed ledger network.

3.1.6 ledger [b-ITU-T F.751.0]: Information store that keeps final and definitive (immutable) records of transactions.

3.1.7 node [b-ITU-T F.751.0]: Device or process that participates in a distributed ledger network.

3.1.8 permissionless [b-ISO 22739]: Not requiring authorization to perform any particular activity.

3.1.9 permissionless distributed ledger system [b-ITU-T F.751.0]: Distributed ledger system where permissions are not required to maintain and operate a node.

3.1.10 private distributed ledger technology system [b-ISO 22739]: DLT system that is accessible for use only to a limited group of DLT users.

3.1.11 public DLT; public distributed ledger system; public distributed ledger technology system [b-ISO 22739]: DLT system which is accessible to the public for use.

3.1.12 smart contract [b-ITU-T F.751.0]: Program written on the distributed ledger system that encodes the rules for specific types of distributed ledger system transactions in a way that can be validated, and triggered by specific conditions.

3.1.13 token [b-ITU-T F.751.0]: A digital representation of value on a shared distributed ledger that is owned and secured using cryptography to ensure its authenticity and prevent modification or tampering without the owner's consent.

3.1.14 transaction [b-ITU-T F.751.0]: An incident or an operation that leads to a change in the status of a ledger, such as adding a record or equivalent exchange based on currency.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 account: Representation of an entity whose data is recorded on a distributed ledger.

NOTE – Originally published in [b-ITU-T TS FG DLT D1.1].

3.2.2 block: Individual data unit of a blockchain, composed of a collection of transactions and a block header.

NOTE – Originally published in [b-ITU-T TS FG DLT D1.1].

3.2.3 Byzantine fault tolerance (BFT): Property that enables a system to continue operating properly even if some of its components fail or existence of intentional bad actors.

NOTE – Adapted from [b-ITU-T TS FG DLT D1.1].

3.2.4 consensus mechanism: Rules and procedures by which consensus is reached.

NOTE – Originally published in [b-ITU-T TS FG DLT D1.1].

3.2.5 crash fault tolerance (CFT): Property that enables a system to continue operating properly even if some of its components fail.

NOTE – Adapted from [b-ITU-T TS FG DLT D1.1].

3.2.6 decentralized autonomous organization (DAO): A digital entity that manages assets and operates autonomously in a decentralized system, but also relies on individuals tasked to perform certain functions that the automaton itself cannot.

NOTE – Originally published in [b-ITU-T TS FG DLT D1.1].

3.2.7 distributed ledger technology (DLT): Technology enabling large groups of nodes in distributed ledger networks to reach agreement and record information without the need for a central authority.

3.2.8 token ecosystem: Digital system or digital space where participants and users interact and co-ordinate with each other using tokens.

NOTE – Originally published in [b-ITU-T TS FG DLT D1.1].

3.2.9 tokenomics (token economic): Economics of a DLT-based token.

NOTE – Adapted from [b-ITU-T TS FG DLT D1.1].

3.2.10 wallet: Software or hardware used to generate, manage and store both private and public keys and addresses, which enable DLT users to transact. Some wallets are recommended to interact with smart contracts and allow single or multi-signature.

NOTE – Adapted from [b-ITU-T TS FG DLT D1.1].

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

API Application Programming Interface

ASIC Application-Specific Integrated Circuit

BFT Byzantine Fault Tolerance

CFT Crash Fault Tolerance

CPU Central Processing Unit

DAO Decentralized Autonomous Organization

DDoS Distributed Denial of Service

DLT Distributed Ledger Technology

ID Identifier

SDK Software Development Kit

SPV Simplified Payment Verification

TPS Transaction Per Second

5 Conventions

This Recommendation uses the following conventions:

– The keywords "**is required to**" indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

– The keywords "**is recommended to**" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement needs not be present to claim conformance.

– The keywords "can optionally" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator or service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with this Recommendation.

6 Overview

The assessment criteria framework established in this Recommendation consists of 27 assessment items. They are recommended for classification into three domains: core functions; application support functions: and operation functions of a DLT platform.

This framework also includes a performance domain and an ecosystem domain, See Figure 1.

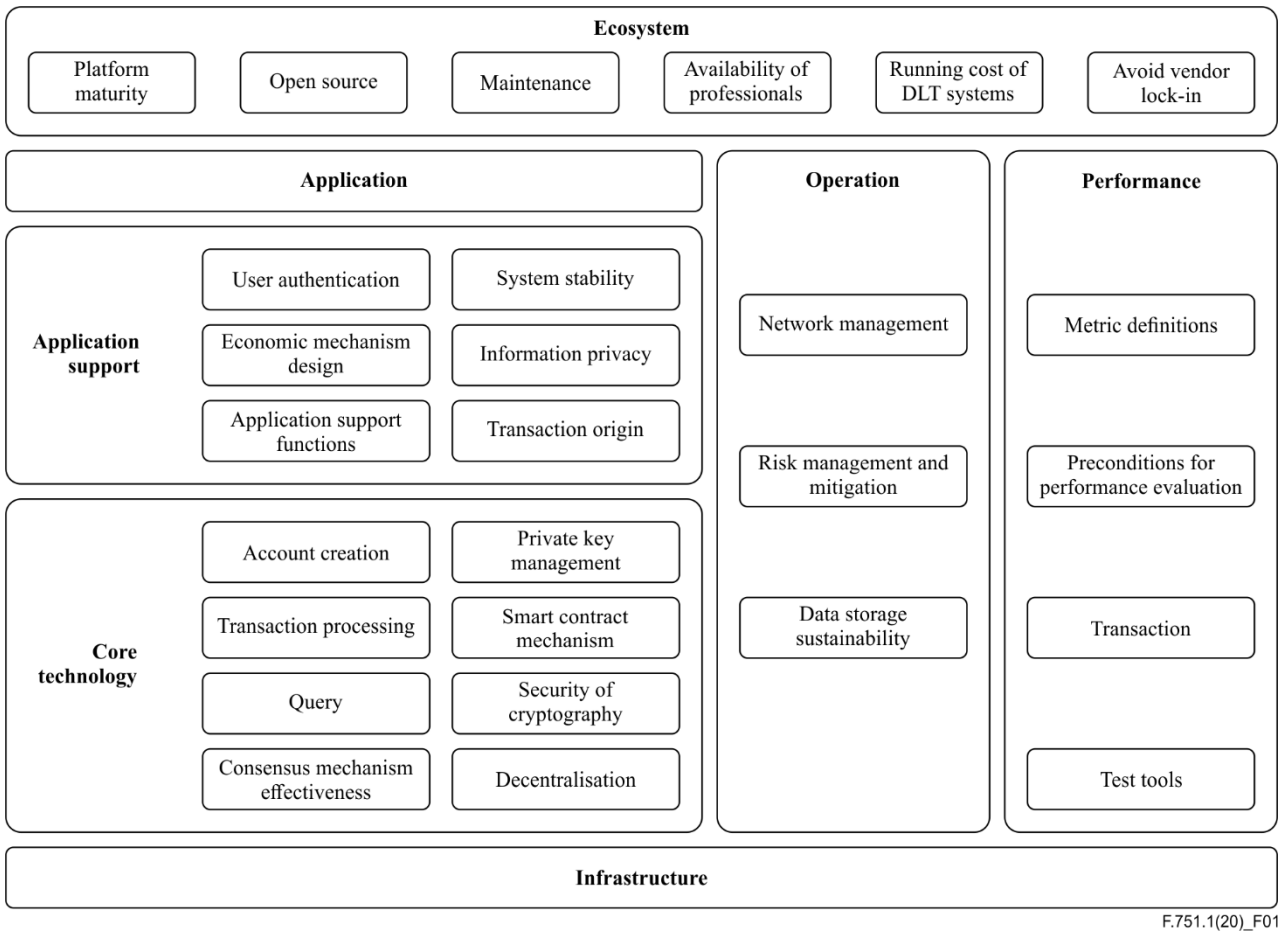

Figure 1 – Overview of the DLT assessment framework

Vendors are required to reveal details about these metrics, which are recommended for verification by document review or functional testing.

7 Criteria for DLT core functions

7.1 Account creation

This relates to the ability to create user accounts. Accounts contain public and private key pairs. The creation is recommended for launch by a client or by a smart contract or other automated functionality components. If an account name is recommended for customization, the account name is required to be unique in the system.

7.2 Transaction processing

This relates to the ability to process transaction(s). There are two types of transaction, but it is not necessary for a DLT platform to support both. The system is required to have all transactions timestamped.

– An **asset transfer transaction** relates to transfer of a certain amount of an asset between accounts, ensuring the asset in the ledger is balanced.

– A **non-asset transfer transaction** (such as changing the configuration parameter of an account, modifying the status of a smart contract and other status modification operations within an account) relates to a transaction without transfer of any asset.

Users can check the result from any node in the DLT system after a successful transaction.

7.3 Query

Users are recommended to get result(s) by information request(s).

7.3.1 Balance query

Users are recommended to acquire their account balance with a searching condition. A DLT platform without an asset transfer function does not need to support this function.

7.3.2 Conditional query

Users are recommended to search their historical information on a DLT platform with a searching condition, such as a time period specification or specified user account.

7.4 Consensus mechanism effectiveness

A consensus mechanism is a set of rules and procedures by which mutual agreement is reached. A data consistency mechanism is used on a DLT platform to achieve the necessary agreement on a single data value or a single state of the network among distributed processes or multi-agent systems.

To ensure the effectiveness of the consensus mechanism, sufficient nodes to participate in the consensus process are required, aligned with the objectives of the DLT platform, e.g., some DLT platforms only require a specific number of masternode participants, while others require all token holders.

7.4.1 Data consistency

The data synchronization module ensures that the distributed ledger is consistent. The synchronization module also validates the synchronized data to ensure its correctness and consistency no matter how immediate or prolonged is the achievement of finality.

7.4.2 Byzantine fault tolerance/crash fault tolerance (BFT/CFT)

DLT systems are required to continue to function in spite of some nodes taking malicious action or system failure. The tolerance threshold for malicious or crashed nodes is determined by the consensus mechanism selected, as well as the economics design of the DLT platform.

7.5 Private key management

Public-key cryptography is a cryptographic system that uses pairs of keys: public keys (which are recommended for wide dissemination); and private keys (which are known only to the owner).

Private key management for DLT use is an important function for user experience and security measures. It provides a reliable and safe way to keep users' keys private. Storage and usage of private keys are recommended to be kept separate. Users are required to have full control of their private key usage. There are two common wallet methods to store private keys: software; and hardware.

NOTE – Separate private key storage and usage are recommended to lead to many other applications, such as that of custodian.

7.5.1 Software wallet

A software wallet is recommended to be a software application (i.e., client application, mobile application) or a service (i.e., digital asset exchanges) to store the private key and to track asset ownership.

In a deterministic wallet, a mnemonic sentence or a word seed is generated and that single root key is safe enough. In a non-deterministic wallet, automatic random generation of each key is recommended. Therefore, any backup of the wallet is required to store each private key used as an address or account password.

7.5.2 Hardware wallet

A service is recommended to create a private key offline and provide the user with a device or physical medium to store it. This service, called a hardware wallet, is required to have the ability to execute basic operations such as signing transactions.

7.6 Smart contract mechanism

A DLT is recommended to support more complex transactions as the technology evolves. Some complex transactions are stored in DLT systems in the form of source code or bytecode programs, and are recommended for execution to deal with different business logics. These programs are called smart contracts.

The smart contract mechanism includes language specification, compilation and execution of the code. Smart contracts for different DLT systems are recommended for implementation using simple interpreted scripts or programming languages.

7.6.1 Monitor ability of participants' status

This relates to a system's ability to monitor the status of nodes (i.e., allocation of computing resources, computing resources consumed, network resources used) that are participating in the execution of a smart contract.

7.6.2 Lifecycle management of smart contract

This relates to the availability of a smart contract state identifier (ID) and the availability of functions for its lifecycle management, such as create, deploy, activate, suspend and destroy.

7.6.3 Security of smart contract

This item relates to the capability to write high-quality smart contracts (i.e., low bug rate) by using a programming language, either from scratch or by using a template.

7.6.4 Smart contract data access control

A DLT system is required to disclose how developers specify the authorization and confidentiality of their smart contract from a technical viewpoint.

Sharing of smart contract data is required among parties to a smart contract, e.g., via interfaces to represent smart contracts and query them.

7.7 Security of cryptography

Security includes encryption, cryptography, crash tolerance and hack tolerance. A DLT system is required to ensure the highest security for the system and disclose its security measures.

7.7.1 Encryption declaration

A DLT platform is required to specify whether the encryption, is derived from an open source solution or through regulatory compliance.

7.7.2 Pluggable encryption algorithm

A DLT platform is recommended to use pluggable modular encryption and to switch to a specified encryption algorithm online or offline as required.

7.7.3 Efficiency of encryption algorithm

A DLT platform is required to use and to make available for use secure, sufficiently strong encryption with acceptable efficiency, depending on the objectives of the system.

7.7.4 Strength of encryption

A DLT platform is required to declare the security level of the cryptographic schemas used. Category and cipher strength of the encryption can be taken as metrics. In addition, quantum-resistant encryption algorithms can be taken into account.

7.8 Decentralization

The system is required to have means to create a distributed and decentralized architecture. It is important to analyse whether there are recommended special node types or limitations that are likely to compromise the decentralization. It is also important to verify whether there is support for decentralized management of the network.

8 Criteria for DLT application functions

8.1 User authentication

A DLT platform is required to have modules for user authentication and user access control management. An electronic signature is an effective way to authenticate a user. The platform is recommended to allow the creation of smart contracts to authenticate and control the access of users.

8.1.1 User account verification

This is the validation of information, such as key-store and password, or two-step verification.

8.1.2 Login status management

The platform is required to update the status after user login.

8.1.3 User classification and user management

This is to assign users into one of several types and manage their permissions.

8.1.4 Authorization

Users are recommended to grant authority to others to access or modify their private data.

8.2 System stability

The platform is required to satisfy at least the requirements of clauses 8.2.1 to 8.2.6.

8.2.1 Stability for node management

The system is required to grant normal operations when some nodes join, leave or upgrade.

8.2.2 Stability for cross-chain operation

The system is required to grant normal operations when co-operating with another DLT or cloud system.

8.2.3 Network latency

The system is required to remain stable after running 7 × 24 h with network latency. The tolerable extent of latency is based on the design of the system.

8.2.4 Memory utilization

The system is required to remain stable after running 7 × 24 h without memory exceptions.

8.2.5 Central processing unit utilization

The system is required to remain stable after running 7 × 24 h without central processing unit (CPU) exceptions.

8.2.6 Stability for concurrency

The system is required to remain stable with bursts of concurrent transactions.

8.3 Economic mechanism design

Economic mechanisms are required to be in place to incentivize user participation. These include, but are not limited to, consensus protocol, resolution mechanisms, voting protocol, allocation mechanism, bargaining protocol, monetary policy of the token and transaction fees.

Economics design focuses on both the economics of the DLT platform, as well as the economics of the tokens that are produced in the DLT system (when applicable).

8.3.1 Incentive mechanisms

Incentive mechanisms includes both financial and non-financial incentives. Non-financial incentives are recommended to include voting protocol, reputation mechanism and allocation mechanisms. Financial incentives are a more direct form of reward, like monetary policy of the token, transaction fees or platform activities (i.e., block rewards when a block is mined). The last is more applicable in permissionless DLT.

Incentive mechanisms are ways to co-ordinate activities among decentralized participants, to achieve the objectives of the DLT system. The incentives (financial and non-financial) are in place to align the behaviours of decentralized participants with the DLT ecosystem as a whole.

For example, Bitcoin's proof-of-work mechanism rewards users in financial ways (diminishing returns of block rewards and transaction fees) in return for the investment of electricity usage and special application-specific integrated circuits (ASICs). This incentive mechanism is required to be described and published.

8.3.2 Token economics disclosure

The token is a digital representation of value. Tokens are recommended to be fungible or non-fungible in nature. Fungible tokens are recommended to have different functions, namely security, utility and money. Depending on the token function, a tokenomics report describing token economics is required to be published.

This is required to include, but not limited to, token policy (monetary policy of the token), valuation policy (for security tokens), platform activities (e.g., block rewards for miners), transaction fees, property rights and distribution of tokens.

8.3.3 Token lifecycle

If tokens are used, the distributed ledger system is required to support token issuance, token transfer, token withdrawal, token settlement and clearance, as well as token balance enquiries.

A DLT platform is required to use a standard protocol for its tokens. The system is required to have functionalities to facilitate cross-chain or cross-DLT system operations, when applicable or required by regulations.

8.4 Information privacy

Privacy of information is a key requirement for DLT platforms, useful in sectors like finance and healthcare. It relates to confidential generation, storage and transmission of data in a DLT system and safe storage of user private information.

8.4.1 Secure transmission

Transfer over a secure channel of confidential or proprietary information is required, which is achievable by a specified secure transmission protocol.

8.4.2 Restricted data access

Any confidential or personal information that is protected by law or policy requires the appropriate level of differential access control and security protection, whether in storage or in transit.

8.4.3 Privacy protection

A DLT platform is recommended to use privacy protection algorithm(s) such as zero knowledge proofs, ring signature, secure multi-party computation and homomorphic encryption to avoid privacy disclosure.

8.5 Application support functions

A DLT platform is recommended to implement application support functions to improve user-friendliness (i.e., user experience and user interface).

8.5.1 User interface for query

A DLT platform is recommended to provide functionality (web page, app, browser plug-in, etc.) to perform a query, visualize the query result and show ledger status.

8.5.2 User interface for smart contract

A DLT platform is recommended to provide functionalities for the visualization of the deployment and invocation of smart contracts, as well as queries about smart contract data.

8.5.3 Multi-language software development kits

A DLT platform is required to provide at least one software development kit (SDK) and is recommended to translate it into other programming languages.

8.6 Transaction origin

A DLT system is required to provide mechanisms to identify the origin of a transaction.

8.6.1 Transaction origin – node

A DLT system is required to provide mechanisms to identify the original node to dispatch a transaction to the network.

8.6.2 Transaction origin – account

A DLT system is required to provide mechanisms to segregate the account signing a transaction from the account dispatching the transaction to the network.

9 Criteria for DLT operation functions

9.1 Network management

The management and monitoring of nodes within a DLT system, including status, configuration, node type and behaviour, are described in clauses 9.1.1 to 9.1.4.

9.1.1 Node status monitoring

A DLT system is required to have the ability to monitor overall node status, such as number of nodes on- or offline, synchronization status and client version.

9.1.2 Multi type nodes

A DLT system is required to have the ability to classify nodes. For instance, a node is recommended for classification in two categories, either full or lightweight, according to whether a complete ledger copy is stored within it. With lightweight nodes, the simplified payment verification (SPV) method is required for verification of the correctness of shared ledgers.

9.1.3 Node configuration modification

A DLT system is required to have the ability to support hot or cold modification of the node's configuration parameter, such as the block size and node type.

9.1.4 Network fairness

The system is required to support a mechanism to balance the sharing of use of the network among nodes.

9.2 Risk management and mitigation

A DLT system is required to have the ability to resist distributed denial of service (DDoS) attacks, Sybil attacks or dishonest nodes. If failure occurs after an attack, the DLT system is required to have the ability to recover its previously known state.

9.2.1 Recovery mechanisms

A DLT system is required to recover from failure by downgrading recovery, security service, etc. A recovery solution is required to be flexible and problem oriented.

9.2.2 Trouble shooting

A DLT system is required to have the ability to execute rapid trouble shooting and automatically send failure notifications.

9.2.3 Avoid single point of failure

The DLT system is required to be independent of any centralized system that might cause a single point of failure.

9.3 Data storage sustainability

A DLT system is an append-only trusted ledger. However, mass data stored in DLT systems is likely to degrade the query performance. The system is required to provide sustainable ways to deal with data volume growth.

9.3.1 Alternative solutions to storage

The system is required to provide a way to minimize the total size of data stored by the whole system. It is recommended to store data no longer used or whose activity level is below a threshold. Some examples are the transfer of data to independent storage or to store the data only on specific types of node.

9.3.2 Data query

A DLT system is required to have ability to query archived data by providing application programming interfaces (APIs) or tools.

9.3.3 Data recovery

A DLT system is required to have ability to recover archived data in some ways, and the ledger(s) is required to keep the same status as before it was archived.

10 Performance

The throughput and resource usage of processing standard transactions. Environment and deployment reasons is recommended to affect performance, such as network topology and test environment (CPU, memory, disk, network). The number of transactions per second (TPS) is a standard performance indicator. When evaluating performance by TPS, indications of topology deployment and test environment are required.

10.1 Metric definitions

The number of TPS is given by

$$\frac{n_{\text{proc}}}{t_{\text{proc}}}$$

where n_{proc} is the number of processed transactions and t_{proc} is the time taken to process transactions.

The data used to calculate TPS needs to be gathered when the system is stable. Both system booting or shutting down and test environment instability are likely to lead to inaccuracy.

The maximum transaction delay, $t_{\text{max del}}$, is given by

$$t_{\text{max del}} = \max\left(t_{\text{con}} - t_{\text{com}}\right)$$

where t_{con} is the transaction confirmed time and t_{com} is the transaction committed time.

The average transaction delay, $\bar{t}_{\text{del}}$, is given by

$$\bar{t}_{\text{del}} = \frac{\sum(t_{\text{con}} - t_{\text{com}})}{n}$$

where n is the number of transactions.

10.2 Preconditions for performance evaluation

10.2.1 Test environment

When evaluating performance of a DLT system, indications of the hardware such as CPU, memory, hard disk, network are required.

10.2.2 Topology of the network

When evaluating performance of a DLT system, an indication of network topology is required.

10.2.3 Test system deployment

When evaluating performance of a DLT system, an indication of the type of deployment is required.

10.3 Transaction

When evaluating performance of a DLT system, indications of the numbers of transactions is required.

10.4 Test tools

There are open source tools designed especially for DLT system to test performance, such as Hyperleder Caliper [b-Caliper] or TrustedBench [b-TB]. Script is also recommended for use to test performance.

11 Criteria for the DLT ecosystem

11.1 Platform maturity

Platform maturity of DLT includes many factors, such as year of creation, launch of production version, deploying networks, numbers of production networks, numbers of applications and team expertise. Whether a network is permissioned or permissionless, a DLT vendor is required to disclose this information to all consumers.

11.2 Open source

DLT platforms are required to be open sourced (to users) and to announce the licence it is using.

11.3 Maintenance

DLT platforms are required to be well maintained by either individuals, companies or non-profit organizations. Inclusion of regular updates on source repository, active discussion by the community regarding the DLT platform and the ease of updating it with respect to the decentralized applications and DAOs that exists on it are recommended.

11.4 Availability of professionals

Different DLT platforms require expertise from various technical backgrounds. This indirectly limits the number of professionals available in the market. Sufficient talent supply is an important factor in the evaluation of DLT platforms.

11.5 Running cost of DLT systems

It is essential for DLT systems to evaluate running costs, which include transaction fees payable, transaction confirmation time and cost of writing, as well as the reading and execution of smart contracts. In the future, with the importance of smart contract security, audit fees for smart contract and code review, as well as many other intermediate fees, will also be required to be taken into consideration.

11.6 Avoid vendor lock-in

DLT system is required to have standardized APIs, such as service access, data dictionary, communication protocol, encryption algorithm and system testing, so that there is recommended to be multiple vendors that provide similar services for the customers to avoid vendor lock-in.

Annex A

How to use the assessment criteria

(This annex forms an integral part of this Recommendation.)

Assessment criteria concerning the economic mechanism (clause 8.3) and ecosystem (clause 11) are required to undergo testing by information disclosure. The tester conducts a comprehensive assessment of information about the DLT system and market.

Criteria related to application (clause 8) and operation (clause 9) functions are required to undergo testing on DLT platforms in a production environment.

Performance criteria (clause 10) are required to undergo assessment in a laboratory environment, i.e., a quality assurance environment, which is required to be stable and controllable [b-ISO/IEC 25000].

DLT core function criteria (clause 7) are required to undergo assessment in a test environment. These tests are recommended to ensure normal operation of a DLT platform.

Bibliography

[b-ITU-T F.751.0] Recommendation ITU-T F.751.0 (2020), *Requirements for distributed ledger systems*.

[b-ITU-T TS FG DLT D1.1] ITU-T Technical Specification FG DLT D1.1 (2019), *DLT terms and definitions.*

[b-ITU-T FG DLT 3.1] ITU-T Technical Specification FG DLT D3.1 (2019), *DLT reference architecture.*

[b-ISO 22739] ISO 22739:2020, Blockchain *and distributed ledger technologies — Vocabulary*.

[b-ISO/IEC 25000] ISO/IEC 25000:2014, *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE*.

[b-Caliper] Linux Foundation Projects (Internet). *Hyperledger – Caliper*. San Francisco, CA: Linux Foundation. Available [viewed 2020-10-09] at https://www.hyperledger.org/projects/caliper.

[b-TB] Github (Internet). *TrustedBlockchain/TrustedBench*. San Francisco, CA: Github. Available [viewed 2020-10-09] at: https://github.com/TrustedBlockchain/TrustedBench.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	**Non-telephone telecommunication services**
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2020