ETSI GR PDL 010 V1.1.1 (2021-08)

PDL Operations in Offline Mode

Disclaimer

The present document has been produced and approved by the Permissioned Distributed Ledger ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGR/PDL-0010_Offline_Mode

Keywords
accountability, ledger, PDL, smart contract

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.
Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Permissioned Distributed Ledger (PDL).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document describes the current challenges related to data storage and ledger operations when a single PDL node or several PDL nodes are offline. This may happen because the nodes are duty cycled, or go offline because of an operational failure or a cyber attack. Operational problems are identified, and possible solutions discussed. Procedures and architecture designs are also presented.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GR PDL 002 (V1.1.1): "Permissioned Distributed Ledger (PDL); Applicability and Compliance to Data Processing Requirements".

[i.2] Capability Hardware Enhanced RISC Instructions (CHERI); University of Cambridge.

NOTE: Available at https://www.cl.cam.ac.uk/research/security/ctsrd/cheri.

[i.3] ETSI GS NFV-MAN 001: "Network Functions Virtualisation (NFV); Management and Orchestration".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

client node: physical node belonging to a client of the PDL, which may or may not be consortium member of the PDL

ledger node: physical node belonging to the PDL which stores data and execution logic in form of Smart Contracts in a distributed and secure manner

offline mode: PDL with at least one element losing connectivity to one, several or all connected elements

online mode: PDL with all of its underlying nodes and protocols being operational and reachable at all times

orchestration: logical capability to provide operational and governance support to the PDL

Permissioned Distributed Ledger (PDL): distributed ledger which is not public, i.e. access permissions are restricted and governed by prior established principles

validator node: physical node belonging to the PDL which is responsible for validating the content provided by the Client nodes before passing it on to the ledge nodes

3.2 Symbols

For the purposes of the present document, the following symbols apply:

N	validated block of the main PDL at time moment N
N'	validated block of the side-chain sPDL at time moment N

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

3GPP	3rd Generation Partnership Project
AI	Artificial Intelligence
CHERI	Capability Hardware Enhanced RISC Instructions
CP	Control Plane
CU	Central Unit
dApp	distributed Application
DDoS	Distributed Denial of Service
DLT	Distributed Ledger Technology
ETSI	European Telecommunications Standards Institute
GDPR	General Data Protection Regulation
HTTP	HyperText Transfer Protocol
IoT	Internet of Things
IP	Internet Protocol
IT	Information Technology
JSON	JavaScript Object Notation
MANO	Management and Orchestration
ML	Machine Learning
OFF	Offline mode
ON	Online mode
OOB	Out Of Band
OS	Operating System
pBFT	practical Byzantine Fault Tolerance
PDL	Permissioned Distributed Ledger
PDLF	PDL Function
PoC	Proof of Concept
PoET	Proof of Elapsed Time
PoS	Proof of Stake
PoW	Proof of Work
RAN	Radio Access Network
RISC	Reduced Instruction Set Computer
RTT	Round Trip Time
SBI	Service-Based Interface
SDK	Software Development Kit
SHA	Secure Hash Algorithm
sPDL	sidechain PDL
SSH	Secure Shell (Protocol)
TCP	Transmission Control Protocol
TEE	Trusted Execution Environment
UK	United Kingdom
VM	Virtual Machine
VPN	Virtual Private Network

4 Introduction To PDL Offline Mode

4.1 Introduction

This clause gives a high-level introduction to Permissioned Distributed Ledgers (PDL) in Offline mode (OFF). To this end, the operational mechanisms of PDL are reviewed first. Then, the present document illustrates a PDL usage scenario and discusses reasons for a PDL going into OFF. Finally, the resulting implications are discussed which lay the foundations for the work carried out in the present document.

4.2 PDL Overview

In the most general case, a ledger is a database. A distributed ledger is thus a distributed database that is consensually shared and synchronized across multiple sites or nodes. The database is typically used through three main interactions:

1) **submit** (content from a client/user onto the ledger);

2) **validate** (and write onto the ledger the submitted content through a consensus protocol); and

3) **read** (the stored content from the ledger).

The content itself has evolved over past years. Initially, only ledger entries could be stored, such as financial transactions, ownership association, etc. However, with the introduction of Ethereum in 2015, execution logic in form of programming code could be stored too. This has led to the emergence of **Smart Contracts** and, as of late, distributed Applications (dApps). The role of distributed ledgers is thus evolving from distributed databases that only store data, to distributed contracts which can take programmatic action on stored or submitted data, to distributed applications that can interface with the clients/users when submitting/reading data.

The **writing of content** onto the ledger typically happens in blocks which are cryptographically linked and, once validated, distributed to all nodes across the entire ledger. The cryptographic linkage and spatial distribution, along with a properly designed validation protocol, make the data written onto the ledger immutable. The distributed nature of the ledger ensures that no central authority can alter content, thus making this technology useful in the context of non-trusted parties interacting with each other.

At the core of each Distributed Ledger Technology (DLT) is the **consensus** protocol, which is being carried out by the validators. It has many roles but mainly ensures that a specific ledger entry cannot appear more than once (thus e.g. preventing the double-spend problem). Different consensus protocols have emerged over past years, such as Proof of Work (PoW); Proof of Stake (PoS), Proof of Elapsed Time (PoET) or practical Byzantine Fault Tolerance (pBFT). They differ in energy efficiency, scale, speed of transactions, among other factors.

Another important aspect is the notion of **public vs. private** ledger. It commonly refers to the degree of anonymity of the validators but also typically extrapolates to the access rights in general, i.e. who can write to and read from the DLT. In the case of a public ledger, validation and access can be done anonymously and by anybody wishing to participate in the ledger. In the case of a private ledger, validation and/or access is restricted to a closed user group such as a consortium (e.g. 10 companies).

Another point to consider is the difference between **permissionless vs permissioned** DLTs. It defines the degree of trust in the validators which execute the consensus protocol. In a permissionless ledger, anyone can participate in the consensus mechanism; whereas, in a permissioned ledger, only those fulfilling certain requirements can take part in the consensus mechanism. Not all consensus protocols are suitable to all scenarios; e.g. permissionless ledgers (such as Bitcoin) would use protocols such as PoW while permissioned ledgers (such as HyperFabric) may use more protocols such as pBFT.

In the present document, solely Permissioned Distributed Ledgers (PDLs) are considered.

4.3 PDL Application Example

Figure 4-1 illustrates an example of a PDL reference use-case scenario, which had been introduced in ETSI GR PDL 002 [i.1]. The scenario pertains to an agricultural application, which is explained in more detail below.

Consider a farmer of a large set of disaggregated land claiming to only be using natural and organic substances, without any chemical and/or genetically modified substances. To prove these credentials, and thus boosting sales, the farmer decides to join a Bio Certification Alliance. The alliance offers bio certification using a PDL, so as to increase transparency to its alliance governance players, to its farmers and to the end consumer wishing to validate the truthfulness of the bio certificate.

At the farmer's side, this is enabled through a set of Internet of Things (IoT) sensors measuring chemical and other pollution throughout the growth and production process. These sensors have their trusted certification and unique digital identity. They constitute the **Client nodes** which transmit information into the PDL for validation. Said validation is done by means of **Validator nodes**. Once validated, the information is immutably written onto the ledger and stored by means of the **Ledger nodes**.

The accuracy of the measurements, and thus the credibility of the bio credentials, depends on the quality and accuracy of the IoT devices and the sampling process. This is referred to as "the last mile" problem. It is beyond the scope of the present document to discuss or solve the last mile problem, but embedding the trusted certification of such IoT devices in the PDL may increase users' trust in the data these devices collect and store in the PDL.

In the context of a PDL, the Validator and Ledger nodes typically belong to a consortium where each member may own a prior agreed set of these nodes. Furthermore, each member or a sub-set of members may offer a set of applications. For instance, a part of the alliance members jointly offers the bio certificate, as long as the sensors in the field support the bio credentials. Another alliance member may offer a smart irrigation service which controls the irrigation system in each of the disaggregated land areas.

Above is enabled through **Smart Contracts** residing in the PDL. Notably, the logic of the Smart Contract will issue a positive certification flag only when all sensors from each of the fields report adherence to bio credentials. The logic can be programmed to perform that check at regular intervals, or be updated when new data from the sensors in the field arrives. Equally, the irrigation Smart Contract will trigger the water valves to be opened when moisture falls below a certain level across a prior agreed set of nodes. Other interesting conditions can be baked in, such as only switching on irrigation when the water price is below a certain threshold (unless irrigation is critical to the survival of the crop). Note that such conditions may be specific for bio credentials (e.g. detection of chemicals) while others may be general (e.g. water cost optimization).

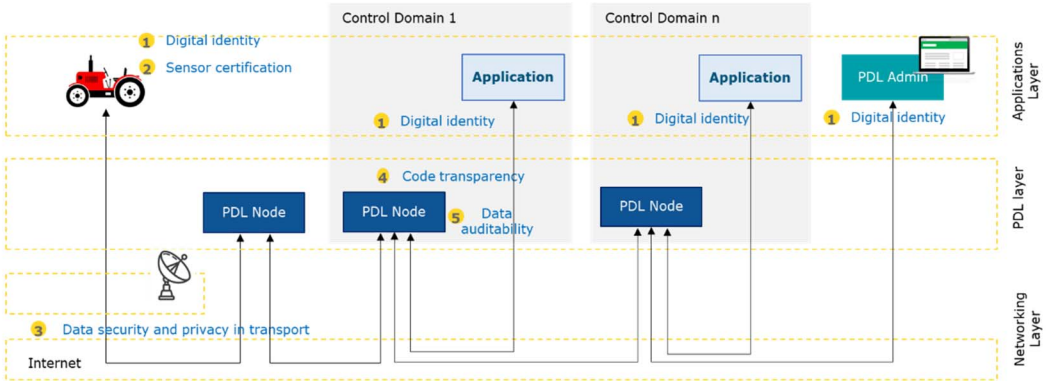

Figure 4-1: Example of a PDL reference use-case scenario from ETSI GR PDL 002 [i.1]

4.4 Reasons For PDL Going Offline

From above example, it is clear that underpinning technology elements of a distributed ledger may occasionally go offline. The reasons are discussed here:

- **By Design:**
  - **Maintenance:** Part of the ledger may require maintenance, such as hardware, networking or software maintenance. Whilst maintenance occurs, the normal operations of the ledger might be impacted which leads to vulnerabilities that need to be dealt with.
  - **Duty Cycle:** To minimize energy consumption of the entire ledger, some of its elements might be duty cycled as per a given and prior agreed schedule. Whilst nodes are offline, possible operational issues as well as vulnerabilities may occur which ought to be dealt with.
- **By Circumstance:**
  - **Temporary Unavailability:** Elements of the ledger may unexpectedly become temporarily unavailable. This could be, for instance, due to an end point using wireless communications moving away from a base station and temporarily losing access to the ledger. Another example occurs in very remote locations where the ledger might be connected via intermittent fixed-line or satellite network.
  - **Congested Network:** In the case of bandwidth-limited or highly congested networks, intra and inter ledger data may not be delivered in time (or delivered at all) and thus compromise the integrity of the ledger operations. That is, all elements are in principle connected but data is not distributed properly. The communication or PDL protocols may have mechanisms in place to resend missing data but that may not happen within the timeframe requirements by the PDL.
- **By Incident:**
  - **Disaster and natural phenomena:** Certain elements of the ledger might be powered by power sources which go out of operations, e.g. due to unforeseeable natural phenomena, such as a lightning strike. The loss of power can cause serious issues to the proper operations of the ledger.
  - **Attack:** In the case of malicious attacks, elements of the ledger can be compromised and rendered intact or offline; in the worst case, the majority quorum capabilities are compromised. This poses serious threats to the integrity and operations of the ledger.

4.5 Offline Challenges

In the case that a PDL or some nodes of a PDL or certain functionality of a PDL are rendered offline, a series of challenges arise which are not normally encountered in fully operational PDLs. These are summarized below and form the rational for the technical approach outlined in the later part of the present document.

From a high level technical and operational point of view, the offline challenges can be categorized as follows:

1) **Security:** In terms of security of the PDL, the following issues arise:
   a) **Consensus Capabilities**, i.e. the consensus approach underpinning the very essence of the PDL may get compromised.
   b) **Weakening Security**, i.e. cryptographic primitives may become unavailable and thus certain processes become unverifiable.
2) **Availability:** In the most general terms, availability is impacted as follows:
   a) **Reconciliation Time**, i.e. when nodes come back online after being offline, it may take some time for them to catch up with the PDL. In the case of nodes suffering intermittent connectivity, it may be possible that by the time they reconcile, the service is interrupted again and they go offline again.
   b) **Side-Chains & Chain Merging**, i.e. the offline mode may trigger the emergence of side chains which has an impact on integrity and availability; and certainly will influence the approach to chain merging. The latter is typically a rare event but may happen very frequently in the discussed scenarios of offline operations.

c) **Stale Transactions**, i.e. if chain merging is not successful, some transactions may become stale which ought to be dealt with by the offline ledger design.

3) **Integrity:** Important issues arise in the context of data integrity:

a) **Control Data**, i.e. data a previously offline node needs to have in order to be accepted back into the PDL network. This typically pertains to authentication data, cryptographical keys, connectivity data, etc. The data related with the content of the ledger itself is out of scope here.

b) **Software Code**, i.e. before being accepted back into the PDL network, checks will need to be performed to ensure that the software/program running on it have not been compromised.

5 PDL Offline Scenarios

5.1 Introduction

This clause discusses possible scenarios resulting from ledgers going offline, as discussed in clause 4. To aid technical understanding, a high-level technical reference architecture is introduced. Thereupon, the role of the different types of nodes is discussed. It is then possible to construct and discuss the different PDL offline scenarios. Last, temporal and spatial characteristics emerging from the different offline scenarios which is unique to a permissioned ledger having offline constituents are discussed.

5.2 High-Level Node Reference Architecture

The application example given in clause 4.3 can be abstracted to a high-level reference Ledger node architecture as illustrated in figure 5-1.

Notably, a set of Client nodes collects data which needs to be written onto the PDL. The data from the Client nodes is transmitted via a fixed or wireless network to the Validator nodes.

The Validator nodes prepare the data for the specific ledger, i.e. sort the data, cast it into a specific format, check for initial consistency, etc. They then perform the PDL-specific consensus protocol to validate the content of the information provided by the Client nodes.

Once validated, the content is written onto the Ledger nodes which store the content for perpetuity. Ledger and Validator nodes belong to several parties.

The end-to-end ledger is configured and maintained by means of PDL-orchestration. Said orchestrator may be implemented on one or several Ledger nodes or on special-purpose orchestration nodes.

The links between the client and Validator nodes and between validator and Ledger nodes might be volatile due to intermittent connectivity or congested networks.

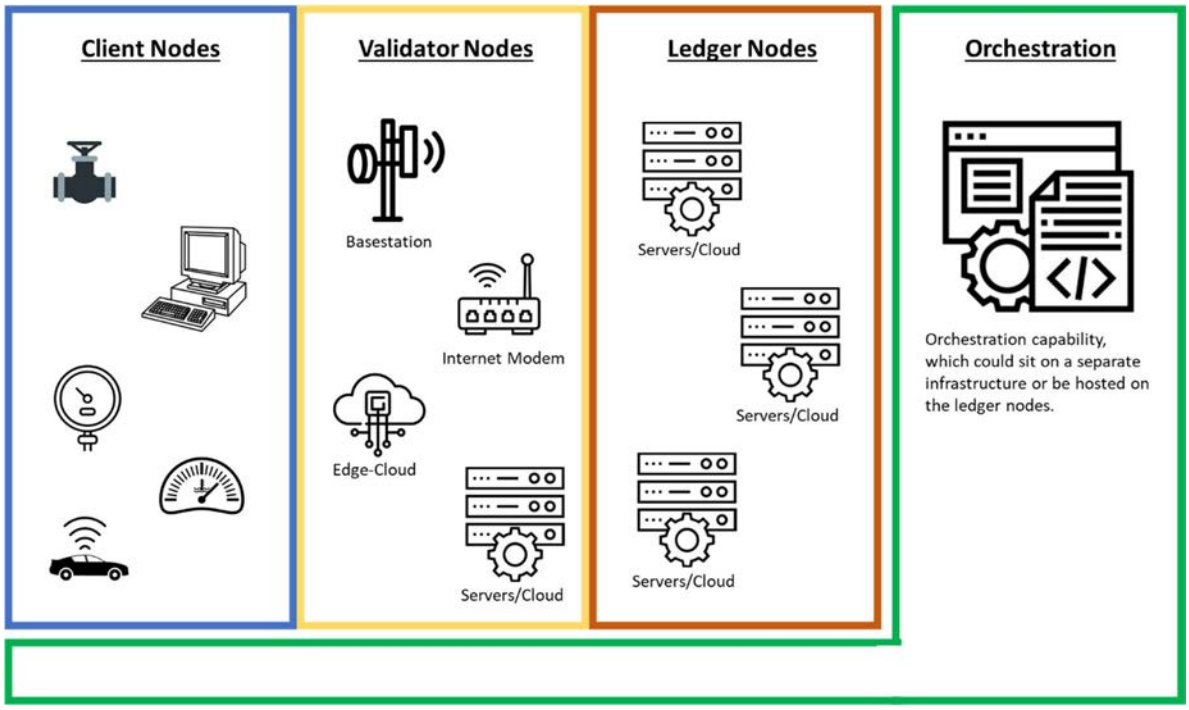

Figure 5-1: High-level node reference architecture comprised of client, validator and Ledger nodes; the system is overseen by orchestration capabilities

5.3 Type of Nodes

As discussed in clause 4.4, there can be several reasons for a node to go offline, such as an engineered duty cycle or an unexpected outage of the wireless network. In this clause, the role of each node type is explained and the likelihood of them going offline estimated.

- **Client nodes:** These are nodes which belong to a client collecting, storing and/or transmitting data. From the ledger's perspective, these are temporary and transient nodes: temporary because a specific Client node may lose connection with the ledger (e.g. a vehicle is going through a patch of poor connectivity) and transient because clients may sign onto the specific PDL service and after a few months/years sign off. Since their presence cannot be guaranteed, they do not take part in the consensus process nor do they form part of the storage ledger. They typically only update their state by sending transactions to Validator nodes.

- **Validator nodes:** These nodes accept transactions from the Client nodes, check the validity and send them to Ledger nodes. Whilst Validator nodes are typically hosted in operationally reliable locations, they can go offline when e.g. there is a network congestion or poor mid/backhaul. For that reason, a strong governance model is required which ensures the viability of the consensus approach when some of the Validator nodes are OFF. For example, a requirement could be that consensus can only be reached if two-thirds of the Validator nodes are online.

- **Ledger nodes:** Ledger nodes are permanent nodes of the ledger. This means that all of the nodes are generally available and online, unless of course in the case of force majeure or a cybersecurity attack. Depending on the design requirements these nodes can or cannot be Validator node. Some architecture models allow Ledger nodes to also serve as Validator nodes and vice versa. The state of the Ledger nodes is being updated by the Validator nodes.

- **PDL-Governance/Orchestration:** Governance is important as it ensures the proper monitoring and execution of the PDL. It is particularly important in the offline scenario as the orchestration capabilities will have to maintain viability of the consensus protocol and ledger operations, among others. It may include roles like the verification of certificates or the revocation of access rights.

Each of these nodes requires trusted digital identities and trusted certificates. Operationally, they can be hosted on bare-metal servers, Virtual Machines (VMs) or containers. Depending on policy and regulation they may be required to be hosted on-premises of the node operator or may be hosted on a public or private cloud.

5.4 Offline Scenarios

The possible scenarios arising due a subset of certain node types becoming unavailable are summarized in table 5-1. Note that when nodes are offline, they may or may not be functional. If they are functional, then they can continue executing tasks locally. Note further that OFF refers to at least one of the node types not being reachable, and not necessarily all of them. The table is sorted from the most likely scenario to the least likely one:

- **Scenario #1:** Is the reference scenario where all nodes are functional and reachable. This should be the modus operandi of the PDL.
- **Scenario #2:** Occasionally, one, several or a cluster of the Client nodes may go offline due to reasons discussed in previous clauses. When this occurs, in most cases the nodes will remain functional, but unable to communicate with other PDL nodes. Situations where the nodes have been rendered dysfunctional or where an entire cluster of nodes has gone offline should also be catered for. It is important to note that when a node goes OFF the reason and functionality of such node are unknown to the other nodes.
- **Scenarios #3 & #4:** Rarely, one or more Validator nodes goes offline for reasons discussed in previous clauses. Again, one needs to cater for situations where the node has been rendered dysfunctional, and not just temporarily gone offline.
- **Scenarios #5-#8:** Extremely unlikely but plausibly, one or more Ledger nodes goes offline for reasons discussed in previous clauses.

The likelihood of any of the scenarios to occur depends on the spatial distribution of the nodes as well as on redundancy/diversity of power and communication utilities, which also impacts the temporal behaviour of the system. Unlike always-on PDLs, Offline PDLs thus suffer from breaks in chain causality. This, in turn, jeopardizes the very essence of distributed ledgers and thus warrants appropriate design attention.

Table 5-1: Possible operating scenarios due to different node types being reachable or offline

	Client nodes	Validation Nodes	Ledger nodes	*Likelihood*
Scenario #1	ON	ON	ON	*very likely*
Scenario #2	OFF	ON	ON	*occasional*
Scenario #3	ON	OFF	ON	*rare*
Scenario #4	OFF	OFF	ON	*rare*
Scenario #5	ON	ON	OFF	*unlikely*
Scenario #6	OFF	ON	OFF	*unlikely*
Scenario #7	ON	OFF	OFF	*unlikely*
Scenario #8	OFF	OFF	OFF	*unlikely*
NOTE: Nodes that are offline may or may not be functional. Furthermore, OFF refers to at least one of the node types not being reachable. The table is sorted from most likely to least likely.				

5.5 Operational Characteristics

The likelihood of a node or set of nodes going offline depends on spatial and operational characteristics of the system, i.e. the node location and operational provisioning.

The spatial composition of a typical PDL is likely as follows:

- **Client nodes:** These are typically lightweight nodes with an embedded Operating System (OS) and limited processing power, onboard memory and battery power. Depending on the PDL design, they could form part of the PDL or not. In any case, Client nodes write data onto the ledger and may receive instructions from the ledger via Smart Contracts. They are located at the very edge of the network, are typically mobile and often untethered.
- **Validator nodes:** These are typically placed physically close to the Client nodes but within the networking infrastructure. For instance, in a 5G system, Validator nodes could sit in the Central Unit (CU) of the Radio Access Network (RAN), or any other edge-cloud location. They would typically be virtualized via VMs or containers, and have sufficient processing power. Whilst spatially placed in a managed environment, the edge is often connected via unreliable backhaul which makes the operational viability of these nodes volatile.