ETSI GS PDL 031 V1.1.1 (2025-11)

GROUP SPECIFICATION

Permissioned Distributed Ledger (PDL); Energy Consumption Data Sharing based on PDL Service

Reference

DGS/PDL-0031_Energy_CDS

Keywords

data sharing, distributed ledger,
energy consumption

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

Contents

Intellectual Property Rights 5

Foreword 5

Modal verbs terminology 5

1 Scope 6

2 References 6
2.1 Normative references 6
2.2 Informative references 6

3 Definition of terms, symbols and abbreviations 7
3.1 Terms 7
3.2 Symbols 7
3.3 Abbreviations 7

4 Fine-grained EC metering in mixed deployment environments 9
4.1 EC measurement in 6G networks 9
4.1.1 Importance of fine-grained EC measurement 9
4.1.2 Key features of fine-grained EC measurement 9
4.1.3 Techniques and approaches enabling fine-grained EC metering 10
4.1.4 Service types impacting EC 12
4.2 Use cases and metrics 13
4.2.1 Use cases of EC measurement 13
4.2.1.1 Use case on media streaming carbon footprint transparency 13
4.2.1.2 Use case on digital sobriety 13
4.2.1.3 Use case on economic incentives for digital sobriety 13
4.2.1.4 Use case on behavioural incentives for digital sobriety 13
4.2.1.5 Use case on watch TV over 5G 13
4.2.1.6 Use case on any Service provider 13
4.2.1.7 Use case on carbon certificate as a service 13
4.2.1.8 Use case on energy profiling on network nodes 14
4.2.2 Key Metrics for EC measurement 14
4.2.2.1 Metric of EC of a single user end-to-end session 14
4.2.2.2 Metric of EC of a HW component/device 14
4.2.2.3 Metric of EC measurement of a software component 14
4.2.2.4 Metric of EC measurement of a software processes 15
4.2.2.5 Metric of EC measurement of a complete service 15
4.2.2.6 Metric of EC of virtualized environment 15
4.2.2.7 Metric of residual EC of a service 16
4.2.2.8 Metric of residual EC of a user session 16
4.2.2.9 Metric of energy production of local green sources 16
4.3 Deployment of EC measurement techniques 17
4.3.1 Measurement points in the network 17
4.3.2 Data reporting and feedback requirements 18

5 Architecture enhancement with PDL service capability for supporting End-to-End (E2E) EC data collection 18
5.1 Background 18
5.2 PDL Overview 19
5.3 Architecture enhancement with PDL service capability 19
5.3.1 Overview of existing telecom network architecture (5G) 19
5.3.2 Architecture enhancement to telecom networks 20
5.3.2.1 General architecture 20
5.3.2.2 Reference point representation 20
5.3.2.2.1 Option#1: EIF directly interfacing with cUPFs 20
5.3.2.2.2 Option#2: EIF indirectly interfacing with cUPFs via network domain CP NFs 21
5.3.2.3 Additional functionalities 22
5.3.2.3.1 Access control 22
5.3.2.3.2 Policy enforcement with smart contracts 22

5.3.2.3.3 Zero-Knowledge Proof (ZKP) 22
5.3.2.3.4 On-Chain/Off-Chain Data Storage 22
5.4 Summary 23

6 Distributed consensus mechanisms for EC metering data post-verification and service enforcement with smart contracts 23
6.1 Post-verification of EC Metering Data 23
6.1.1 General introduction 23
6.1.2 EC data distributed verification 24
6.1.2.1 Single-domain EC data aggregation 24
6.1.2.2 Cross-domain EC data verification 25
6.1.3 Real-time auditability and monitoring 25
6.1.4 Robustness through multi-tool attestation 26
6.2 Smart contracts for service enforcement 26
6.2.1 Smart Contract Design 26
6.2.2 Usage Scenarios and Invocation Mechanisms 27
6.2.3 Deployment and Governance 27

7 Conclusion 27

History 28

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 IPR online database.

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™**, **LTE™** and **5G™** logo 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 Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Permissioned Distributed Ledger (PDL).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**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

Energy Consumption (EC) is a key metric to assess the energy efficiency of a service. When a service is operated over multiple domains under different authorities, sharing the EC data is challenging without a centralized authority. Given that, the present document aims to study how EC data can be trustworthily shared with PDL service. Specifically, the present document targets to address the following problems:

- Study the existing methods for fine-grained EC metering in physical and virtualized environments; study PDL service architecture enhancement for supporting E2E EC metering data collection.
- Study distributed consensus mechanisms for EC metering data post-verification and service enforcement with smart contracts.

2 References

2.1 Normative 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.

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

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 necessary for the application of the present document.

[1] ETSI GS PDL 012: "Permissioned Distributed Ledger (PDL); Reference Architecture".

[2] ETSI GS PDL 011: "Permissioned Distributed Ledger (PDL); Specification of Requirements for Smart Contracts' architecture and security".

[3] ETSI GS PDL 013: "Permissioned Distributed Ledger (PDL); Supporting Distributed Data Management".

[4] ETSI GS PDL 024: "Permissioned Distributed Ledgers (PDL); Architecture enhancements for PDL service provisioning in telecom networks".

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 may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] Weaver, V.: "Reading RAPL energy measurements from Linux®", University of Maine.

NOTE: Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

[i.2] ARM Ltd.: "Energy Probe overview", Arm Developer Documentation.

[i.3] NETIO Products a.s.: "PowerBOX 4Kx", Smart PDU with Electrical Measurement, NETIO Official Product Documentation.

[i.4] NETIO Products a.s.: "PowerPDU 8KF", Professional Smart PDU with Metered & Switched Outlets, NETIO Official Product Documentation.

[i.5] NETIO Products a.s.: "PowerDIN 4PZ", DIN Rail Power Meter and Smart Switch, NETIO Official Product Documentation.

[i.6] NETIO Products a.s.: "Official Website of NETIO Smart Power Monitoring Solutions".

[i.7] Amaral, Marcelo, et al.: "Kepler, A framework to calculate the energy consumption of containerized applications", 2023 IEEE™ 16th international conference on cloud computing (CLOUD).

[i.8] van Kemenade, Tim: "Real-time Scaphandre Energy Metrics Pipeline Integrated with Escheduler". Diss. Vrije Universiteit Amsterdam, 2024.

[i.9] ETSI GR PDL 009: "Permissioned Distributed Ledger (PDL); Federated Data Management".

[i.10] ETSI GR PDL 004: "Permissioned Distributed Ledgers (PDL); Smart Contracts; System Architecture and Functional Specification".

[i.11] ETSI ES 202 336-12: "Environmental Engineering (EE); Monitoring and control interface for infrastructure equipment (power, cooling and building environment systems used in telecommunication networks); Part 12: ICT equipment power, energy and environmental parameters monitoring information model".

[i.12] ETSI TS 128 554: "5G; Management and orchestration; 5G end to end Key Performance Indicators (KPI) (3GPP TS 28.554 Release 18)".

[i.13] 4E EDNA.(2019): "Total Energy Model for Connected Devices".

[i.14] JRC135926_01: "Energy Consumption in Data Centres and Broadband communication networks in the EU", Kamiya & Bertoldi, 2024.

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

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

3GPP	3rd Generation Partnership Project
5G	Fifth Generation
6G	Sixth Generation
ABAC	Attribute Based Access Control
AI	Artificial Intelligence
AMF	Access and Mobility management Function
API	Application Programming Interface
ARM	Advanced RISC Machines
BMC	Baseboard Management Controller
CO2	Carbon Dioxide
CP	Control-Plane
CPU	Central Process Unit

CU	Centralized Unit
cUPF	computing User Plane Function
DLE	Distributed Ledger Enabler
DLE-Peer	Distributed Ledger Enabler -Peer
DU	Distributed Unit
E2E	End-to-End
EC	Energy Consumption
EE	Energy Efficiency
EIF	Energy Information Function
GDPR	General Data Protection Regulation
GHG	Greenhouse Gas
GPU	Graphic Process Unit
GR	Group Report
HW	Hardware
I/O	Input/Output
ICT	Information and Communication Technologies
IoT	Internet of Things
IP	Internet Protocol
IPMI	Intelligent Platform Management Interface
NEF	Network Exposure Function
NF	Network Function
OEF	On-site Energy Fraction
OTLP	Open Telemetry Protocol
OTT	Over-The-Top
PBFT	Practical Byzantine Fault Tolerance
PCF	Policy Control Function
PDL	Permissioned Distributed Ledger
RAM	Random-Access Memory
RAN	Radio Access Network
RAPL	Running Average Power Limit
RF	Radio Frequency
SC-ITM	Smart Contract of Incentive & Token Management
SC-MV	Smart Contract of Measurement Validator
SC-PE	Smart Contract of Policy Enforcement
SDIA	Sustainable Digital Infrastructure Alliance
SLA	Service Level Agreement
SMF	Session Management Function
TS	Technical Specification
TV	Television
UDM	Unified Data Management
UE	User Equipment
UPF	User Plane Function
VM	Virtual Machine
ZKP	Zero Knowledge Proof

4 Fine-grained EC metering in mixed deployment environments

4.1 EC measurement in 6G networks

4.1.1 Importance of fine-grained EC measurement

In the context of 6G networks, Energy Consumption (EC) measurement is critical to ensuring that the deployment and operation of the network are sustainable, efficient, and cost-effective. In the transition to 6G, Energy Efficiency (EE) will become even more important due to the expected exponential increase in the number of connected devices, higher data rates, and more complex network services.

EC measurement has long been a critical aspect of managing network resources and optimizing EE. Traditional methods of EC measurement often provide a broad, system-level view of power usage, typically aggregated across entire systems or network components. However, as modern networks, particularly 6G, evolve to support increasingly complex and dynamic services, a higher level of precision is required to capture the nuances of EC.

Fine-grained EC metering, which provides detailed, process-level, or even thread-level measurement, represents a paradigm shift towards more sophisticated energy management. This approach enables the identification of energy usage patterns at a granular level, offering insights into the exact power consumption of specific processes, applications, or components within the system.

4.1.2 Key features of fine-grained EC measurement

For fine-grained EC metering to be effective, several key features shall be present.

First, granularity of measurement is crucial. The EC meter shall be able to track EC down to the process, thread, or program level. This includes per-process metering, where the device identifies and monitors EC of each individual process on the system. This involves measuring CPU consumption per process, where the power used for CPU cycles is tracked, as well as memory access power consumption, which involves monitoring energy used by memory operations (RAM reads and writes) for each process.

Second, Input/Output (I/O) operations shall also be tracked, as they consume energy when interacting with disks, networks, or peripherals. Beyond per-process metering, the device needs to offer thread-level precision to differentiate between energy used by various threads within a process. Integration with hardware-level energy meters, such as Intel RAPL [i.1] or ARM energy monitors [i.2], is essential for capturing EC of specific hardware components like CPUs, GPUs, or network devices.

Third, the sampling rate of the EC metering shall be high to ensure the instrument can capture fine-grained data. A high-frequency sampling rate, ideally in the range of microseconds (µs) or milliseconds (ms), ensures that even short-lived processes are captured accurately. For instance, in high-performance computing environments, rapid processes should not go unnoticed. The EC meter shall be capable of real-time monitoring, where short-term EC spikes (such as during I/O bursts or GPU operations) are captured instantly. Additionally, adaptive sampling could be employed, where the sampling interval is adjusted based on process duration, allowing for detailed monitoring of shorter processes while minimizing data logging overhead for longer tasks.

Fourth, support for virtualized and distributed environments is critical for modern cloud and distributed systems. Many applications now run across Virtual Machines (VMs) or containers. The metering shall therefore be able to monitor EC in virtualized environments by interfacing with hypervisors such as VMware, KVM, or Hyper-V, and attribute power usage to each individual VM instead of merely the physical host. For containerized applications like those running in Docker or Kubernetes environments, the EC meter shall track energy used by individual containers, even though they share underlying hardware resources. In distributed systems, the EC meter shall gather and consolidate energy data from multiple physical or virtual nodes, including edge and cloud nodes, ensuring accurate attribution of power consumption. When shared resources like CPU cores or network interfaces are used by multiple processes, the instrument should allocate energy usage accordingly based on their utilization.

4.1.3 Techniques and approaches enabling fine-grained EC metering

Fine-grained Energy Consumption (EC) metering methods refer to techniques used to measure EC with a high level of granularity. These methods aim to provide detailed and accurate measurements of energy usage in various environments, including both physical and virtualized settings.

Traditional approaches to EC metering in physical environments have been foundational in managing and recording electricity usage. Direct metering techniques involve the use of physical meters to measure EC directly at the point of use. These methods are fundamental to energy management in various settings and have evolved over time, starting with analog meters and progressing to more sophisticated digital meters.

Analog meters have been the traditional tool for measuring electricity consumption. They operate using a mechanical rotating disc that moves in proportion to the amount of electricity consumed. The simplicity and robustness of analog meters make them reliable; they do not require an external power source and can operate for extended periods with minimal maintenance. However, these meters come with significant drawbacks. They require manual reading and recording, which is labour-intensive and prone to human error. The infrequency of readings, often limited to monthly intervals, means that the data collected lacks granularity, making it difficult to analyse consumption patterns or identify opportunities for efficiency improvements.

Digital meters represent a more advanced approach to direct metering. Unlike their analog counterparts, digital meters provide electronic readings of EC. These meters can store data at much higher frequencies, allowing for more detailed and accurate records. Digital meters can transmit data automatically to central systems via communication networks, reducing the need for manual readings and minimizing errors. This capability enhances the granularity of the data, enabling more precise monitoring and analysis of energy usage. However, digital meters can be more expensive to install and maintain than analog meters and require a reliable power source and communication infrastructure.

Sensor-based monitoring systems offer a more granular and dynamic approach to EC metering. These systems use various sensors to continuously measure different parameters of energy usage, providing real-time data that can be analysed for immediate insights and long-term trends.

Smart sensors are at the core of these systems, capable of measuring voltage, current, power, and other relevant metrics with high precision. These sensors are often installed at various points within an electrical network, including at individual appliances, circuits, or distribution panels. The real-time data collected by these sensors is transmitted to a central monitoring system, where it is aggregated and analysed. This continuous stream of data enables the detection of anomalies, peak demand periods, and inefficient energy use, allowing for proactive management and optimization of EC.

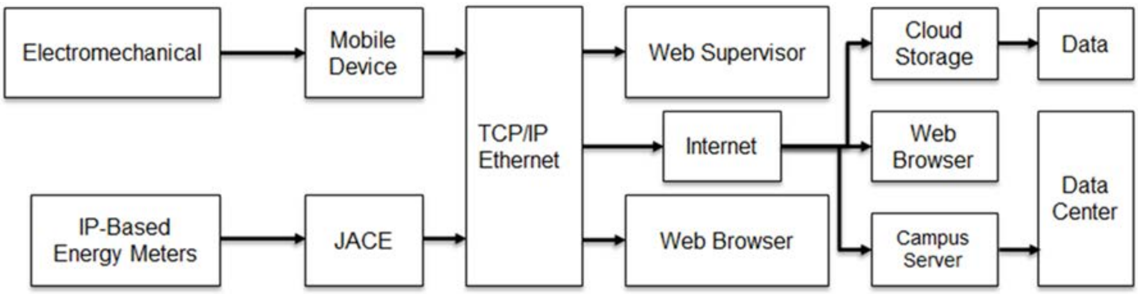

Figure 1: A metering infrastructure for campus distribution system

The integration of Internet of Things (IoT) technology enhances the capabilities of sensor-based monitoring systems. IoT-enabled sensors can communicate wirelessly, facilitating easy installation and scalability. They can also be integrated with advanced analytics platforms and machine learning algorithms to provide predictive insights and automated responses to identified issues. For instance, a sensor detecting an unusual spike in energy usage can trigger an alert or automatically shut down a malfunctioning device to prevent energy waste or damage.

In virtualized environments, where multiple Virtual Machines (VMs) or containers share underlying physical resources, accurate and fine-grained metering of EC is crucial for optimizing resource allocation, improving energy efficiency, and enabling fair billing practices. Two common approaches for EC metering in virtualized environments are hypervisor-based monitoring and application-level instrumentation.

Hypervisor-based monitoring leverages the virtualization layer, known as the hypervisor, to collect EC metrics from the underlying physical hardware and the virtual entities (VMs or containers) running on top of it. The hypervisor has a privileged view of the entire system and can gather detailed information about resource utilization, which can be correlated with EC.

The hypervisor-based approach typically involves the following techniques:

- Hardware performance counters: Modern processors and other hardware components provide performance counters that can be accessed by the hypervisor. These counters capture low-level metrics such as CPU cycles, cache misses, and memory accesses, which can be used to estimate EC based on predetermined models or calibration data.

- Power modelling: The hypervisor can employ power models that map resource utilization metrics (CPU, memory, disk, network) to EC values. These models can be derived from empirical measurements or vendor-provided data for specific hardware configurations.

- Direct energy measurement: In some cases, the hypervisor may have access to hardware sensors or instrumentation that directly measures the EC of physical components, such as CPU packages, memory modules, or entire server nodes.

Advantages of hypervisor-based monitoring include a comprehensive view of resource usage across all virtual entities, minimal overhead as monitoring is performed at the hypervisor level, and the ability to monitor EC even for opaque or closed-source applications.

Disadvantages include limited visibility into application-level metrics and behaviour, potential security concerns due to the hypervisor's privileged access, and the need for accurate power models or calibration data for reliable energy estimation.

Application-level instrumentation involves embedding monitoring code or agents within the applications running inside the virtual machines or containers. These agents collect resource usage metrics and application-specific performance indicators, which can be correlated with EC patterns.

The implementation of application-level instrumentation can take various forms, such as:

- Code instrumentation: Modifying the application's source code to include energy monitoring hooks or probes that capture resource usage metrics and report them to a centralized monitoring system.

- Agent-based monitoring: Integrating a separate monitoring agent or library with the application, either through dynamic linking or by running alongside the application process. The agent collects resource usage metrics and sends them to an energy monitoring system.

- Energy-aware profiling tools: Utilizing profiling tools specifically designed for EC analysis, which can instrument the application code or integrate with the runtime environment to capture energy-related metrics as in Figure 2.

Application-level instrumentation can collect various metrics relevant to EC, such as:

- CPU usage: Tracking CPU cycles, instruction counts, and CPU-bound workloads.

- Memory usage: Monitoring memory footprint, memory access patterns, and memory-bound workloads.

- Disk I/O: Capturing disk read/write operations, disk throughput, and disk-bound workloads.

- Network I/O: Tracking network traffic, bandwidth utilization, and network-bound workloads.

- Application-specific metrics: Collecting application-level performance indicators, such as transaction rates, response times, or custom metrics relevant to EC patterns.

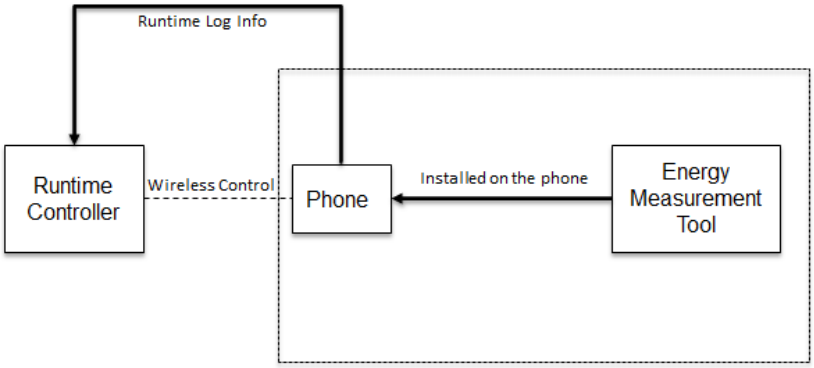

Figure 2: Runtime monitoring

Advantages of application-level instrumentation include granular insights into application-level EC patterns, the ability to correlate energy usage with specific application behaviours or workloads, and fine-grained optimization opportunities.

Disadvantages may include the need to modify or instrument the application code, potential overhead introduced by the monitoring agents, and the challenge of accurately mapping application-level metrics to EC without hardware-level calibration or modelling.

In practice, a combination of hypervisor-based monitoring and application-level instrumentation may be employed to achieve comprehensive and accurate EC metering in virtualized environments. The hypervisor-based approach provides a system-wide view, while application-level instrumentation offers granular insights into application behaviour and EC patterns, enabling targeted optimization and energy-efficient resource allocation strategies.

4.1.4 Service types impacting EC

In 6G networks, services can be categorized into distinct types, each with different Energy Consumption (EC) characteristics and computational requirements. These service types include transmission services, computation services, and dynamic services:

- **Transmission Services:** These services primarily focus on data transfer between network nodes and end devices. EC for transmission services is closely related to the volume of data being transmitted, the distance between devices, and the network infrastructure (e.g. base stations or edge nodes). Power usage in these services tends to scale with bandwidth demands and network traffic, requiring efficient protocols to optimize energy use during data transfer.

- **Computing Services:** In contrast to transmission services, computation services are more resource-intensive, involving data processing, analytics, and decision-making tasks at both the edge and cloud layers. These services require significant computational power and are heavily dependent on the performance and efficiency of the underlying hardware (e.g. GPUs or specialized processors). EC in computation services is influenced by factors such as processing complexity, resource allocation, and task scheduling.

- **Dynamic Services:** Dynamic services in 6G networks are characterized by their ability to adapt to real-time changes in network conditions and user requirements. These include services like network slicing, on-demand resource provisioning, and real-time edge computing. The EC of dynamic services is variable, as it is affected by factors like the level of service scaling, load balancing, and the agility of the network in adjusting to varying demand. Efficiently managing the energy use of dynamic services requires advanced algorithms and predictive models that can anticipate demand spikes and optimize resource allocation.

Each service type has distinct EC patterns, requiring different methods for accurate energy metering and management.

4.2 Use cases and metrics

4.2.1 Use cases of EC measurement

4.2.1.1 Use case on media streaming carbon footprint transparency

Energy Consumption (EC) measurement plays a critical role in a variety of use cases across different sectors, particularly in environments where sustainability and operational efficiency are key priorities. These use cases highlight the need for precise and granular tracking of energy usage to optimize performance, minimize environmental impact, and ensure compliance with regulatory standards. By integrating EC measurement into these scenarios, organizations can make data-driven decisions that not only reduce EC but also contribute to broader efforts to mitigate climate change. This clause explores the specific use cases that require accurate EC measurement to achieve these objectives.

This use case focuses on providing users with visibility into the carbon footprint of their video streaming activities, including OTT platforms and video conferencing. Metrics such as "Instant Carbon Footprint" or "Total Daily CO_2" are displayed to the user during streaming. This transparency allows users to make informed choices about the trade-off between service quality and environmental impact, offering them the option to adjust settings that minimize their carbon footprint.

4.2.1.2 Use case on digital sobriety

Building on UC introduced in clause 4.2.1.1, this use case empowers users to take active steps to reduce their carbon footprint by providing alternative delivery modes such as lower video resolution or data rate options. These alternatives are accompanied by their respective carbon footprint impacts, enabling users to select a service option that aligns with their environmental goals.

4.2.1.3 Use case on economic incentives for digital sobriety

This use case adds an economic layer to UC introduced in clause 4.2.1.2 by offering users incentives for reducing their carbon footprint. Users are rewarded with "environmental points" for selecting carbon-reducing alternatives, and at the end of each month, those points contribute to a lottery, with the user's potential reward being proportional to their accumulated points.

4.2.1.4 Use case on behavioural incentives for digital sobriety

Similar to UC introduced in clause 4.2.1.3, this use case seeks to motivate users to reduce their carbon footprint, but through behavioural incentives. Peer pressure is used as a motivational tool, where users are privately informed of their relative performance in terms of carbon footprint reduction, with comparisons to others in the same timeframe (e.g. weekly performance).

4.2.1.5 Use case on watch TV over 5G

This use case introduces carbon footprint tracking in the context of watching TV over 5G networks. Similar to UC1, metrics are provided to the user, but this use case highlights differences in connectivity types (specifically 5G) that may affect the carbon footprint of the service session.

4.2.1.6 Use case on any Service provider

While UC introduced in clause 4.2.1.1 visualizes energy and carbon footprint data to the user, this use case extends this functionality to all service providers involved in the service delivery chain. This ensures that carbon footprint transparency is shared across the entire ecosystem, allowing each provider to assess and adjust their environmental impact.

4.2.1.7 Use case on carbon certificate as a service

This use case envisions a carbon trading market where users receive carbon usage reports at the end of their billing period. Based on these reports, users can trade their carbon credits, receiving compensation for unused carbon or purchasing additional credits if their emissions exceed their allowance.

4.2.1.8 Use case on energy profiling on network nodes

This use case involves the monitoring and aggregation of energy-related data at each node in the network infrastructure. The collected data is analysed to create an energy profile for each node, with an efficiency index assigned to represent the node's energy performance. This dynamic energy profiling sets the stage for optimizing load distribution across the network, helping reduce overall EC.

4.2.2 Key Metrics for EC measurement

4.2.2.1 Metric of EC of a single user end-to-end session

Description: evaluation of EC generated by a single user session. EC contributions of every component included into the end-to-end (multidomain) service path should be considered. The metric includes EC contributions of service components required for the service to run (see EC#3, EC#5) and energy consumed at the client side (see EC#3). The mathematical definition is:

$$E_{EC\#1} = \int_t P_{UserSession}\, dt = \sum_T P_{UserSession}\, \Delta T \quad (1)$$

Formula 1: EC of a single user session based on integration or discrete summation of power usage over time:

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.
- Time resolution: would depend on input metrics time resolution.

4.2.2.2 Metric of EC of a HW component/device

Description: measuring EC of a HW device, i.e. the measurement would normally take place at the power outlet, resulting in measurement of total EC. In case the device has multiple power inputs, EC of all of them shall be measured, while total EC is sum of all partial measurement results. The mathematical definition is:

$$E_{EC\#2} = \int_t P_{device}\, dt = \sum_T P_{device}\, \Delta T \quad (2)$$

Formula 2: EC of a hardware device as the integral or discrete summation of power consumption over time:

- Unit: Joule (can be transformed to kWh).
- Accuracy: metric definition does not specify accuracy by itself, while general recommendation for practical implementations would be, case by case, to first identify optimal accuracy needed, i.e. costs of measurement would normally increase with higher accuracy, which might not be needed for every case.
- Time resolution: although the metric can be defined in continuous time, in general, the implementation should look for optimal time distance between the two adjacent samples.
- Potential implementation tool: power sockets/outlets with EC measurement (or power measurement) ability such as Shelly, Netio (e.g. PowerBOX [i.3], PowerPDU [i.4], PowerDIN [i.5]), etc.

4.2.2.3 Metric of EC measurement of a software component

Description: evaluation of EC of a software component. A software component can consist of a single atomic process (see EC#4) or could be composed of multiple atomic processes. EC of the software component is therefore the sum of its atomic components' EC. The mathematical definition is:

$$E_{EC\#3} = \int_t P_{component}\, dt = \sum_T P_{component}\, \Delta T = \sum_i E_{EC\#4(i)} \quad (3)$$

Formula 3: EC of a software component calculated by direct power integration or as the sum of its constituent atomic processes:

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.

- Time resolution: would depend on input metrics time resolution.

4.2.2.4 Metric of EC measurement of a software processes

Description: measuring EC of an atomic process, i.e. the most basic process (executable/binary) that can be measured in practice, e.g. utilizing mechanisms such as RAPL. This level of granularity allows for evaluating EC of more complex software components (i.e. composed of multiple processes) simply by summing EC of its basic processes. The atomic process itself should be perceived as a general process which is not limited to any specific environment such as virtual or non-virtual environment. As well, the metrics refers to any atomic process, including those required for the system to run properly, e.g. processes enabling virtual environment. The mathematical definition is:

$$E_{EC\#4} = \int_t P_{SWprocess}\, dt = \sum_T P_{SWprocess}\, \Delta T \qquad (4)$$

Formula 4: EC of an atomic software process computed via time integration or discrete power sampling:

- Unit: Joule (can be transformed to kWh).
- Accuracy: metric definition does not specify accuracy by itself, while general recommendation for practical implementations would be, case by case, to first identify optimal accuracy needed, i.e. costs of measurement would normally increase with higher accuracy, which might not be needed for every case.
- Time resolution: although the metric can be defined in continuous time, in general implementation should look for optimal time distance between the adjacent samples.
- Potential implementation tool: software tools such as Scaphandre, Kepler.

4.2.2.5 Metric of EC measurement of a complete service

Description: evaluation of EC of a complete (end-to-end) service (may span over multiple domains), consisting of multiple components, i.e. each component's contribution should be identified and considered. The mathematical definition is:

$$E_{EC\#5} = \sum_i E_{EC\#3(i)} \quad \forall\, i \in service \qquad (5)$$

Formula 5: aggregated EC of a complete service based on its constituent software components (EC#3):

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.
- Time resolution: would depend on input metrics time resolution.

4.2.2.6 Metric of EC of virtualized environment

Description: evaluation of EC of virtualized environment. This would include total EC of all software components/processes running in the environment, including processes necessary only for the virtual environment to exist and operate. Another perspective of the metric would be EC of HW device(s) enabling virtual environment reduced by the HW's own EC (see EC#2). The mathematical definition is:

$$E_{EC\#6} = \sum_i E_{EC\#4(i)} \quad \forall\, i \in process\ of\ enabling\ virt.\ env \qquad (6a)$$

or

$$E_{EC\#6} = \sum_j E_{EC\#3(j)} \quad \forall\, j \in component\ of\ enabling\ virt.\ env. \qquad (6b)$$

Formula 6a and 6b: total energy consumed by processes or components enabling a virtualized environment:

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.
- Time resolution: would depend on input metrics time resolution.

4.2.2.7 Metric of residual EC of a service

Description: based on the fact that there is common EC required for the services to run, i.e. own EC of HW and virtual environment, this residual EC should be divided and added to the EC of each service running in the environment. The mathematical definition (w - weight, i.e. share of specific service in common EC that correspond to residual EC) is:

$$E_{EC\#7} = \sum_{i,j} E_{EC\#2(i)} \cdot w_{i,j} \quad \forall\, i \in \textit{devices enabling service},\ j \in \textit{service} \qquad (7)$$

Formula 7: residual EC of a service based on weighted allocation from shared hardware:

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.
- Time resolution: would depend on input metrics time resolution.

4.2.2.8 Metric of residual EC of a user session

Description: similarly to EC#7 (Residual EC of a service), this metric evaluates common EC required for the services to run, divided per each single user (end-to-end) session which may span over multiple domains. The mathematical definition (w - weight, i.e. specific's user share of service residual EC) is:

$$E_{EC\#8} = \sum_{i,j} E_{EC\#7(i)} \cdot w_{i,j} \quad \forall\, i \in \textit{services included in user sess.},\ j \in \textit{user sess.} \qquad (8)$$

Formula 8: residual EC of a user session based on its share across contributing services:

- Unit: Joule (can be transformed to kWh).
- Accuracy: would depend on input metrics accuracy.
- Time resolution: would depend on input metrics time resolution.

4.2.2.9 Metric of energy production of local green sources

Description: measurement of green energy produced by local sources (e.g. solar cells) powering network equipment, servers, user devices, etc. The metric is utilizable with EE#1 (Renewable energy factor - REF) metric. The metric can be also applied to any remote green source providing energy for the network/environment considered. The mathematical definition is:

$$E_{EC\#9} = \sum_i \int_t P_i\, dt \quad \forall\, i \in \textit{local source} \qquad (9)$$

Formula 9: total energy production of local renewable sources over time:

- Unit: Joule (can be transformed to kWh).
- Accuracy: metric definition does not specify accuracy by itself, while general recommendation for practical implementations would be, case to case, to first identify optimal accuracy needed, i.e. costs of measurement would normally increase with higher accuracy, which might not be needed for every case.
- Time resolution: although the metric can be defined in continuous time, in general implementation should look for optimal time distance between the adjacent samples.

Considering time resolution, the above-mentioned documents deal with generic time interval, i.e. specifying only the total EC is sum of discrete average values over the observed period, multiplied by the time interval duration where necessary.

Table 1: Summary of EC measurement metrics

Metric Index	Time Resolution	Accuracy Resolution	Unit	Reference
#1	Depends on input data res.	Depends on input data res.	Joule	ETSI ES 202 336-12 [i.11], ETSI TS 128 554 [i.12]
#2	no specific req.	no specific req.	Joule	ETSI ES 202 336-12 [i.11], ETSI TS 128 554 [i.12]
#3	depends on input data res.	depends on input data res.	Joule	ETSI ES 202 336-12 [i.11], ETSI TS 128 554 [i.12]
#4	no specific req.	no specific req.	Joule	ETSI TS 128 554 [i.12]
#5	depends on input data res.	depends on input data res.	Joule	ETSI TS 128 554 [i.12]
#6	depends on input data res.	depends on input data res.	Joule	ETSI TS 128 554 [i.12]
#7	no specific req. depends on input data res.	depends on input data res.	Joule	DIMPACT, GHG Protocol ICT sector guidance, [i.10], [4], [i.13], [i.14]
#8	depends on input data res.	depends on input data res.	Joule	DIMPACT, GHG Protocol ICT sector guidance, [i.10], [4], [i.13], [i.14]
#9	no specific req.	no specific req.	Joule	SDIA (On-site Energy Fraction - OEF), ETSI TS 128 554 [i.12]

Table 1 summarizes the key details of these EC measurement metrics, outlining their time and accuracy resolution requirements, and provides a comprehensive overview of the standards guiding EC measurements in 6G and related technologies.

Each metric is designed to capture the EC in discrete intervals, depending on the resolution of the input data. For most metrics, time resolution and accuracy resolution depend on the characteristics of the input data, meaning that the granularity of measurement varies accordingly. In cases where no specific requirements are defined, the measurement is typically dependent on the resolution of the data provided. Most of these metrics measure EC in Joules, which is the standard unit for energy. The references in the table include various standards and protocols such as ETSI ES 202 336-12 [i.11], ETSI TS 128 554 [i.12], and DIMPACT, which guide the implementation of energy measurement practices in telecom and ICT sectors.

4.3 Deployment of EC measurement techniques

4.3.1 Measurement points in the network

To accurately measure EC across 6G networks, measurement points need to be strategically deployed at various levels of the network infrastructure. These points should cover key components where energy usage occurs, enabling comprehensive EC monitoring:

- Edge Devices and User Equipment: At the edge of the network, User Equipment (UE) and edge devices are critical measurement points, as they contribute directly to EC, especially in computational services. Monitoring the energy used by these devices provides insights into the real-time energy demands of end users and application-specific tasks.

- Base Stations and Access Nodes: These components serve as central hubs for data transmission between devices and the core network. Measuring EC at base stations and access points can provide valuable data about the energy required for data transmission, signal processing, and network resource management. These are primary measurement points for transmission services.

- Core Network and Data Centres: The core network, including routers, switches, and data centres, plays a significant role in computation services. Measuring energy at these points helps capture the computational load and the energy required for data processing, routing, and storage. Data centres, where cloud-based services and AI computations occur, are particularly important for understanding the EC of large-scale computational tasks.

- Network Slicing and Virtualized Environments: For dynamic services, EC measurement points should also be included in virtualized network environments, such as within network slices. These points enable monitoring of the energy consumed during on-demand resource allocation and real-time adjustments made in response to varying network traffic and service demands.

4.3.2 Data reporting and feedback requirements

Once EC data is collected from the various measurement points, effective reporting and feedback mechanisms shall be in place to ensure actionable insights. The data should be continuously monitored, aggregated, and communicated to enable dynamic service optimization and ensure sustainability. Specifically, the following data have to be collected:

- Power Consumption Data: This is the most immediate and direct form of energy data. Power consumption shall be monitored in real-time and reported with high granularity, especially in areas with high traffic or computational demand. Feedback on power usage should trigger alerts if power thresholds are exceeded or if inefficient consumption patterns are detected.

- EC Data: EC over time should be aggregated to understand overall usage patterns and trends. It is critical for long-term energy efficiency strategies and for understanding the total energy footprint of the network. Energy data should be reported periodically to allow for the analysis of load balancing, energy optimization, and carbon footprint management.

- Carbon Emission Data: The carbon emissions related to energy usage in the network shall also be calculated and reported. This involves using the energy data and applying carbon intensity factors based on the energy source (e.g. renewable vs. non-renewable). Carbon emissions data should be provided to relevant stakeholders, such as operators and regulatory bodies, and should inform decisions on optimizing energy sources and adopting green energy initiatives.

- Feedback for Dynamic Service Adjustment: The reported data should feed into a feedback loop for dynamic service adjustment. For example, if EC exceeds predefined thresholds, the network should automatically trigger actions like load shedding, resource scaling, or adjusting computational tasks to less energy-intensive processes. Additionally, penalties or adjustments for high-carbon activities can be implemented using smart contracts.

By deploying measurement points at key network components and ensuring real-time reporting and feedback, operators can gain complete visibility into EC across the network. This enables better optimization of energy use, reduces carbon emissions, and improves overall network sustainability.

5 Architecture enhancement with PDL service capability for supporting End-to-End (E2E) EC data collection

5.1 Background

EC is a fundamental metric for assessing EE in digital and physical infrastructures. Its importance is particularly evident in multi-domain environments, where services span across different administrative authorities, infrastructures, or technological domains. In such decentralized settings, the ability to collect, share, and verify EC data becomes significantly more complex, and traditional EC metering architectures often prove inadequate.

The lack of centralized management in multi-domain scenarios introduces several challenges:

- Interoperability limitations arise when heterogeneous systems - ranging from industrial IoT devices using legacy protocols to cloud-native platforms relying on modern APIs - attempt to exchange EC data. This leads to inconsistencies in data formats, time resolution, and collection methodologies.

- Data fragmentation and silos naturally form as each domain retains EC metrics in isolated repositories. Without a unified coordination layer, cross-domain energy optimization, global demand response, or unified analytics become impractical.

- Trust and integrity are difficult to maintain in the absence of a centralized authority. Malicious manipulation or accidental inconsistencies in distributed systems can compromise the accuracy of EC measurements, undermining decisions related to energy savings, carbon accounting, or regulatory compliance.

- Scalability and performance bottlenecks become apparent as the volume of metering data increases, especially with the proliferation of IoT nodes, edge devices, and virtualized workloads. Real-time collection and processing place additional stress on architectures, particularly when latency, bandwidth, and synchronization differ across domains.

- Security and auditability gaps are amplified in decentralized environments, where encryption, access control, and logging mechanisms may vary across stakeholders. This raises privacy and compliance risks, particularly under stringent regulatory regimes such as GDPR.

These systemic issues highlight the need for a structured yet decentralized approach to EC data management. A Permissioned Distributed Ledger (PDL) technology offers a viable solution by enabling:

- Trusted, multi-party validation of EC records.

- Interoperable data exchange across administrative boundaries.

- Tamper-proof audit trails and traceability.

- And fine-grained, policy-based access control for secure EC data sharing.

In this context, PDL serves not only as a decentralized storage substrate, but also as an orchestration and trust layer for scalable, transparent, and verifiable EC data management across domains.

5.2 PDL Overview

Architecture for secure and efficient EC data sharing across multiple domains is developed with extensive reference to the work of the ETSI Industry Specification Group on Permissioned Distributed Ledgers (ISG PDL). The foundational concepts and system models outlined in ETSI GS PDL 012 [1] provide a comprehensive reference architecture that informs the design.

Federated and distributed data management challenges are addressed by incorporating principles from ETSI GR PDL 009 [i.9] and ETSI GS PDL 013 [3], which discuss the application of the PDL reference architecture and define platform services and architectural requirements for distributed data management. These documents outline architectural requirements and define extended platform services essential for the implementation.

Given the critical importance of data integrity and trust in cross-domain energy data sharing, the architecture leverages the security and smart contract frameworks detailed in ETSI GR PDL 004 [i.10] and ETSI GS PDL 011 [2]. These specifications provide insights into implementing non-repudiation mechanisms and secure smart contract execution, ensuring the reliability and authenticity of shared data.

In scenarios involving telecom networks, the architecture enhancements proposed in ETSI GS PDL 024 [4] are considered. This document offers guidance on integrating PDL services within telecom infrastructures, which is pertinent when the architecture interfaces with such networks. By aligning the architecture with these established ETSI ISG PDL specifications, robustness, interoperability, and adherence to industry best practices for secure and efficient EC data sharing across multiple domains are ensured.

5.3 Architecture enhancement with PDL service capability

5.3.1 Overview of existing telecom network architecture (5G)

In the current network architecture, each operator collects EC metrics from Radio Access Network (RAN) nodes (DU/CU), User Plane Functions (UPFs), and Control-Plane (CP) network functions (e.g. AMF, SMF, PCF, UDM) into a locally managed database. Access policies often rely on static configurations (e.g. IP-based whitelists), while the cryptographic and authentication infrastructure rests with a single authority. Although sufficient for single-operator scenarios, this design exhibits limited transparency, lacks robust auditability, and presents potential single points of failure when multiple stakeholders - such as other operators, third-party service providers, or regulatory bodies - need verifiable insights into the network's energy usage.

5.3.2 Architecture enhancement to telecom networks

5.3.2.1 General architecture

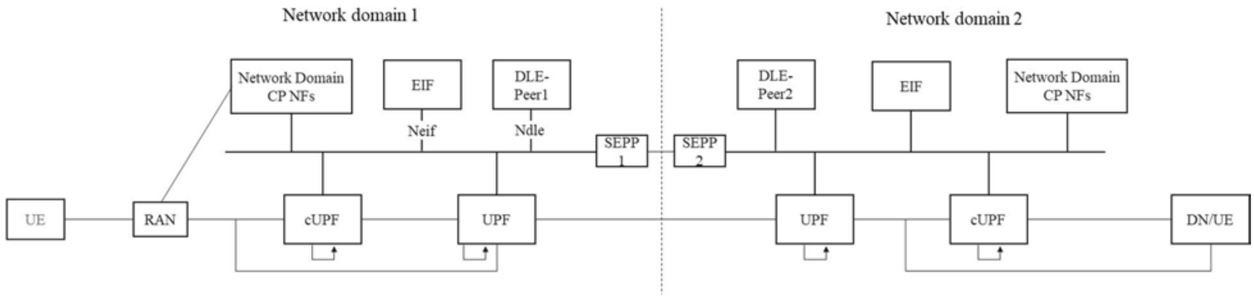

Figure 3: Architecture enhancement with PDL capability for multi-domain EC data sharing

Figure 3 presents the enhanced telecom network architecture based on 5G network architecture. The enhanced architecture incorporates DLE-peer nodes and Energy Information Functions (EIFs) to manage EC data across multiple network domains. In each domain, EC data are collected from entities that are involved with a particular service, in which an EC information consumer is interested. Given a particular service, the relevant entities can include User Equipment (UE), Radio Access Networks (RAN), UPF, computing UPF, and Control Plane Network Functions (CP NFs).

To address the limitations of traditional 5G architectures in EC data management, the enhanced architecture introduces a series of structural innovations designed to enable secure, transparent, and scalable handling of EC metrics. Specifically, two new network functions are introduced:

- Energy Information Function (EIF): A specialized NF is introduced to collect and pre-process all EC data from relevant entities participating a service offered through the current network domain. By unifying data flows in a single function, the architecture centralizes energy metrics collection and ensures consistent formatting, filtering, and quality checks before submission to the distributed ledger.

- Distributed Ledger Enabler (DLE)-Peer: Each network domain uses DLE-Peer function, which could be instantiated with one or multiple network nodes to share the collected EC data for different granularities. DLE-Peer offers PDL service capability such as consensus protocols (e.g. Raft, PBFT, or a hybrid) and shares the EC data with other DLE-Peers in other network domains (e.g. Network domain 2 in Figure 3). DLE-Peer nodes together validate, store, and share EC records across administrative boundaries, guaranteeing tamper-proof data and avoiding the vulnerabilities of a centralized store. In cross-operator or multi-domain use cases, DLE-Peer nodes coordinate to achieve consensus on new EC entries via threshold signatures or majority voting.

5.3.2.2 Reference point representation

5.3.2.2.1 Option#1: EIF directly interfacing with cUPFs

To support real-time collection, validation, and governance of EC metrics, the enhanced 5G architecture introduces a refined set of data exchange interfaces. As illustrated in Figure 4, energy data from various sources - including User Equipment (UE), RAN nodes, UPF components, and Control Plane Network Functions (CP NFs) - is transmitted to the Energy Information Function (EIF) for unified pre-processing and verification. This is achieved through the PN-1, PN-2, PN-3, and Neif interfaces. Once validated, the data is forwarded to the Distributed Ledger Enabler (DLE) node via the Ndle interface, enabling secure on-chain anchoring and distributed consensus:

- **PN-1:** This interface collects EC metrics from UE, including device-level energy profiles such as battery usage, transmission power, and energy efficiency statistics during data sessions. PN-2 captures energy telemetry from the Radio Access Network (RAN), specifically from DUs and CUs, encompassing real-time values such as RF power output, CPU load, and beam-level consumption.

- **PN-2:** This interface gathers energy consumption data from RAN components, such as DUs and CUs. Metrics typically include radio transmission power, processing load, and beam-level energy utilization.

- **PN-3:** This facilitates the transmission of EC data from both computing UPFs (cUPFs) and standard UPFs. Metrics reported by cUPFs include power consumption of AI tasks, content caching engines, and service-specific compute functions. Standard UPFs provide energy data related to packet forwarding, interface-level activity, and application-layer services such as streaming or analytics.
- **Neif:** This interface links Control Plane Network Functions - namely AMF, SMF, PCF, UDM, and NEF - to the EIF, interacting with metadata and control-plane telemetry relevant to EC monitoring.
- **Ndle:** This interface links the EIF to the domain's DLE-Peer node, supporting ledger update and consensus execution within the same domain.
- **PN-4:** To enable secure collaboration across different administrative domains, this interface is defined between DLE-Peer nodes of different networks, facilitating cross-domain data exchange, consistency verification, and distributed consensus. This modular interconnect structure ensures that each domain can contribute and validate EC data in a decentralized yet harmonised fashion, supporting scalable and auditable energy data governance.

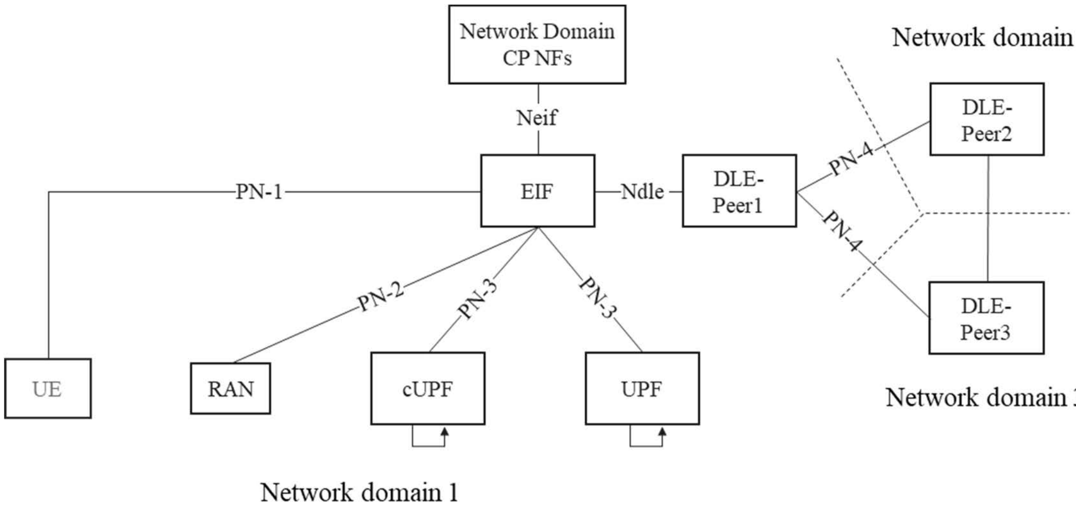

Figure 4: Reference points of the enhanced architecture with PDL capability for multi-domain EC data sharing

Figure 4 illustrates a reference architecture where the EIF is deployed as a network entity that directly interfaces to UPFs or cUPFs. In this architecture, all upstream energy data - collected from UE, RAN, and UPF components over PN-1 and PN-2 and PN-3 - is directed to the EIF. Control-plane telemetry from CP NFs is relayed through the Neif interface to the EIF, consolidating all EC-related signalling and metadata at a higher orchestration layer. Validated records are then transmitted to the domain-local DLE-Peer using the Ndle interface. Cross-domain synchronization is again achieved via PN-4, maintaining distributed consistency of the distributed ledger.

5.3.2.2.2 Option#2: EIF indirectly interfacing with cUPFs via network domain CP NFs

Alternatively, Figure 5 illustrates a reference architecture where the EIF is deployed as a supervisory entity interfacing to CP NFs only rather than cUPFs/UPFs. In this architecture, all upstream energy data - collected from UE, RAN, and UPF components via PN-1 to PN-3 - is directed to the EIF for centralized pre-processing. Control-plane telemetry from CP NFs is relayed through the Neif interface to the EIF, consolidating all EC-related signalling and metadata at a higher orchestration layer. Validated records are then transmitted to the domain-local DLE-Peer using the Ndle interface. Cross-domain synchronization is again achieved via PN-4, maintaining distributed consistency of the distributed ledger.

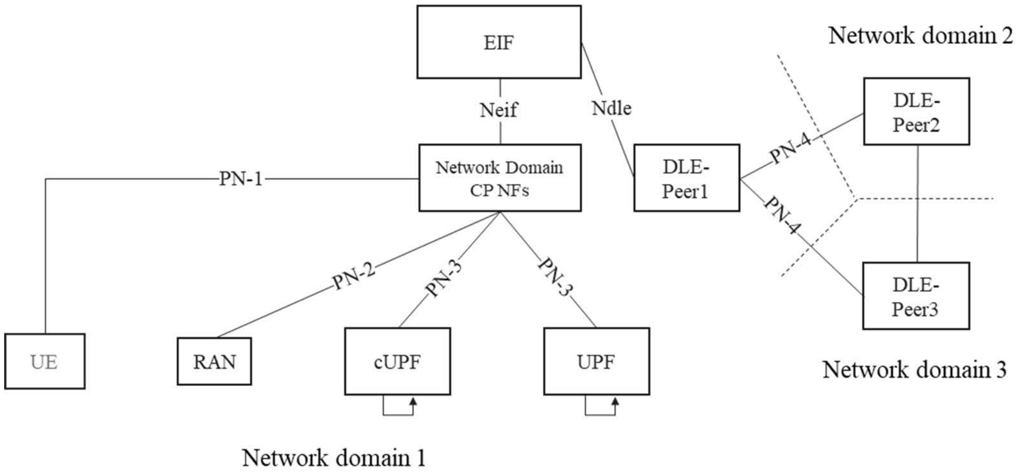

Figure 5: Reference points of the enhanced architecture (Option 2)

Both deployment options maintain compatibility with the distributed ledger layer and support scalable, decentralized, and privacy-preserving EC data handling across administrative boundaries. The choice between these configurations depends on operator preference, network modularity requirements, and energy management orchestration strategies.

5.3.2.3 Additional functionalities

5.3.2.3.1 Access control

NEF, which is interfaced via Neif, shall be responsible for managing fine-grained data access based on Attribute Based Access Control (ABAC). It enforces on-chain access policies defined by cryptographic attributes such as stakeholder identity, role, and authorization level. This replaces traditional IP-based whitelisting and is anchored in the PDL through interaction with the UDM.

5.3.2.3.2 Policy enforcement with smart contracts

Smart contracts shall be deployed on distributed ledger on the DLE-Peers, which is connected via the Ndle interface. These contracts encode operational energy policies, such as consumption thresholds or slice-level carbon budgets. When validated EC data flows through EIF and is pushed to the ledger, smart contracts automatically trigger policy actions (e.g. alerts, carbon penalty, or resource reallocation), ensuring real-time compliance enforcement across domains.

5.3.2.3.3 Zero-Knowledge Proof (ZKP)

Privacy-preserving access to EC metrics shall be handled through the NEF via Neif. ZKP allow external parties (e.g. regulators or auditors) to validate compliance or claim entitlement to energy credits without accessing raw energy data. This is especially critical for scenarios involving user session-level EC (collected via PN-1) or application-specific energy traces from UPFs (PN-3).

5.3.2.3.4 On-Chain/Off-Chain Data Storage

A hybrid storage mechanism shall be available for archiving the ledger data when an offline storage requirement needs to be fulfilled. It coordinates on-chain metadata anchoring (hashes, timestamps, access logs), while the offline storage system stores raw EC datasets off-chain using erasure coding. This division ensures ledger efficiency and scalability while maintaining auditability. Data routed from the EIF (Neif and Ndle) is hash-validated before being linked to its off-chain storage entry.

By assigning each function to a defined architectural entity and linking it to a corresponding reference point, the architecture ensures that these advanced capabilities are not abstract design goals but are implementable, verifiable, and aligned with ETSI standard practices.

5.4 Summary

The architecture enhancement with PDL service capability transforms the architecture of the conventional telecom network systems (e.g. 5G) into a collaborative, trustworthy ecosystem for energy data. Each domain's RAN and UPF functions report power consumption via PN-X interfaces to a centralized EIF, which prepares and dispatches consistent EC records on-chain. By relying on a permissioned ledger (DLE-Peers) and hybrid storage (on-chain hashes, off-chain UDM/UDR data), the design preserves performance while ensuring end-to-end traceability and immutability. With ABAC and zero-knowledge tools enforcing selective data visibility, plus automated triggers via smart contracts, the framework meets stringent regulatory demands and enables dynamic, multi-stakeholder energy governance establishing a robust foundation for sustainable 5G/6G networks.

6 Distributed consensus mechanisms for EC metering data post-verification and service enforcement with smart contracts

6.1 Post-verification of EC Metering Data

6.1.1 General introduction

As established in clause 5, the enhanced architecture leverages Energy Information Functions (EIFs) and DLE-Peer nodes to achieve secure and fine-grained metering of EC data. However, trustworthy service-level energy governance not only depends on the collection of data, but also on its post-verification and enforcement mechanisms, particularly when the data is shared across multiple domains consisting of different participating service providers.

With support for external power meters (e.g. Netio), internal BMC telemetry (e.g. IPMI, Redfish), and software-level tools (e.g. KEPLER, ALUMET, SCAPHANDRE), EC data can now be captured with high accuracy, granularity, and provenance. Each data packet is signed, timestamped, and linked with its measurement context (method, location, component) to form a tamper-proof tuple, ready for distributed validation.

Distributed consensus is thus necessary not just for storing this data, but for ensuring that it has passed cross-domain post-verification, enabling thrustless enforcement of energy-aware policies:

- an enhanced post-verification design is defined that differs from traditional energy data collection models by introducing a verifiable, multi-source validation layer embedded within the distributed ledger workflow. This enhancement is characterized by two major architectural improvements:
    - i) decoupling of validation from centralized logic by enabling peer-level mutual verification among DLE-Peer nodes; and
    - ii) extending support for multi-modal measurement sources - such as Netio [i.6], KEPLER [i.7], and Scaphandre [i.8] - whose outputs are reconciled through a structured consensus process.

Compared to conventional single-source logging, the enhanced mechanism enables more reliable, tamper-resistant validation of EC records across domains with heterogeneous metering capabilities.

Post-verification refers to the process by which EC data - after being prepared by the EIF - is validated across different domains and tools before it is committed to the local DLE node. This design ensures:

1) **Cross-validation across heterogeneous sources:** EC data from multiple measurement methods (e.g. ALUMET vs. KEPLER) can be evaluated based on a configurable error threshold. Consensus is only reached when most sources fall within the accepted deviation range.

2) **Verification blocks:** Each EC record is converted into a structured Verification Block:
    - Format: ⟨timestamp, EC#, domain, component_ID, method_ID, value, hash, signature⟩.
    - These are hashed and recorded on-chain via the Ndle interface.

3) **Real-time auditability via Grafana-Prometheus integration:** Energy records are exported in Prometheus-compatible formats. This allows automated querying and visual inspection to support post-facto traceability.

4) **Multi-tool attestation:** For sensitive domains, data is only accepted when validated by at least two distinct toolchains (e.g. Scaphandre + external smart socket like Netio), increasing robustness.

Depending on deployment topology and data aggregation architecture, two distinct post-verification models are defined.

6.1.2 EC data distributed verification

6.1.2.1 Single-domain EC data aggregation

Before EC data is shared across different domains, EC data collected from multiple sources within a domain are first aggregated at a centralized entity (typically the EIF or a designated energy controller), the aggregation procedure is depicted in Figure 6 and elaborated as follows.

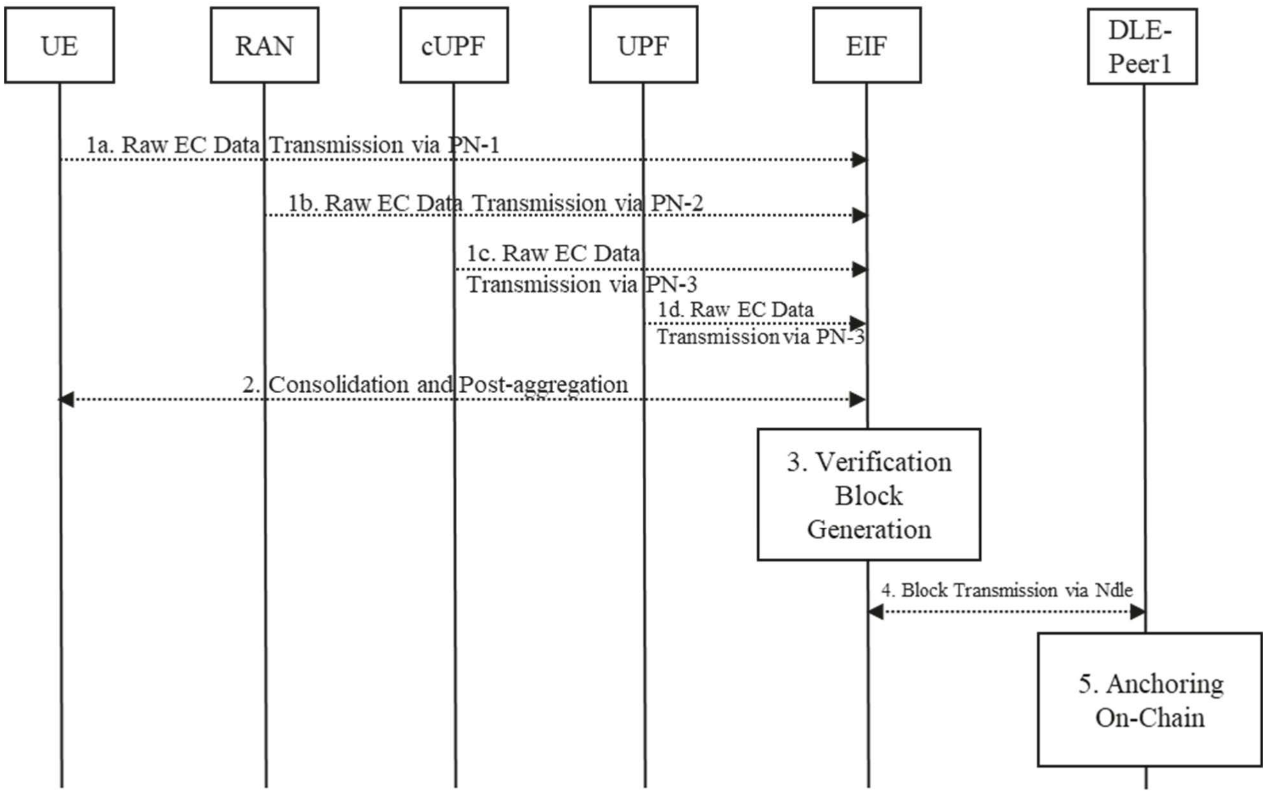

Figure 6: Single-domain EC Data Aggregation Procedure

1) All raw EC data from User Equipment (UE), Radio Access Networks (RAN), User Plane Functions (UPF), and Control Plane Network Functions (CP NFs) are transmitted via the PN-1, PN-2, PN-3, and Neif interfaces into the EIF.

2) The EIF consolidates energy metrics from heterogeneous sources (e.g. telemetry collected via ALUMET agents, KEPLER exporters, or embedded monitoring within UPFs). Post-aggregation, the EIF performs cross-validation by comparing measurement outputs across sources. A configurable deviation threshold (Δ_EC) is defined to assess whether data from different origins converge within acceptable variance.

3) Consensus is locally established if the aggregated measurements meet internal consistency criteria. Otherwise, data records are flagged and quarantined.

4) A structured **Verification Block** is generated for each validated EC record, formatted as:

⟨timestamp, EC#, domain, component_ID, method_ID, value, hash, signature⟩

5) The verified blocks are transmitted via the Ndle interface to the domain-local DLE-Peer for anchoring on-chain.

6.1.2.2 Cross-domain EC data verification

After EC data is aggregated in one domain, it will be shared across different domains and validated independently at multiple points. The distributed consensus procedure is depicted in Figure 7 and explained as follows.

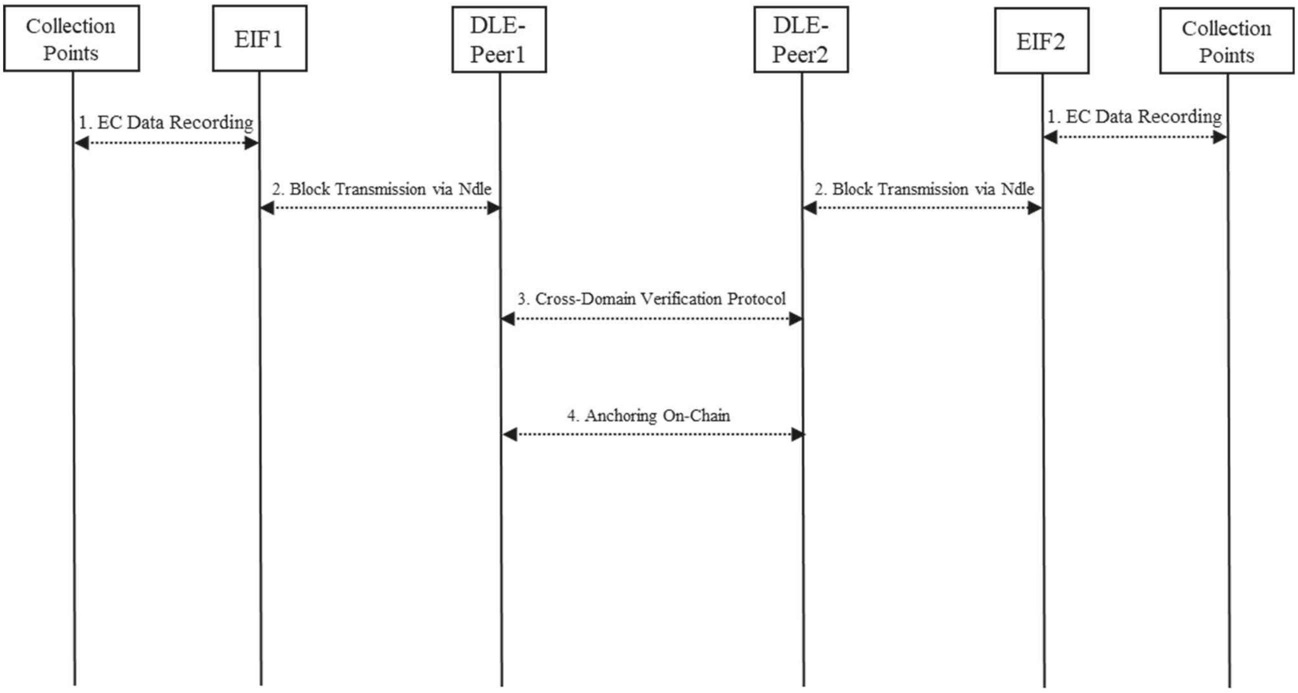

Figure 7: Cross-domain EC Data Verification Procedure

1) EC data records are independently generated at distributed collection points, including cUPFs, DUs, service-specific edge nodes, or even external smart sockets (e.g. Netio power meters).

2) Each validator node submits its measurement via EIF to the DLE-Peer without intermediate consolidation.

3) During consensus execution, DLE-Peer nodes initiate a coordinated cross-domain verification protocol. Each peer exchanges signed, hashed Measurement Tuples representing the same logical resource - such as the same UE session or network slice - but captured by different measurement tools or from different domains. Once exchanged, nodes perform precise input-output consistency checks: for instance, comparing the energy consumed during uplink at the cUPF with the energy recorded during downlink at the DU, or between software-level tools such as Scaphandre and external meters like Netio. Validation occurs only if a configured quorum (e.g. $\geq N/2 + 1$ domains) confirms alignment within a predetermined deviation tolerance (e.g. ±10 %).

4) Once quorum is reached, the Measurement Tuple is approved and committed on-chain as a tamper-proof Verification Block.

6.1.3 Real-time auditability and monitoring

The mechanism defined in this clause shall support **real-time operational monitoring**, **management-level reporting**, and **post-event auditability** of validated EC records. Standardized telemetry output shall be provided for consumption by both internal operations and external audit entities.

Responsibilities and Triggers:

- **Operational Triggers:** The EIF shall automatically export telemetry metrics upon each newly validated EC record or when an operational anomaly (e.g. Δ_EC threshold violation) is detected. These metrics shall be consumed by operations teams for immediate alerting and visualization.

- **Management/Compliance Triggers:** The EIF shall periodically (e.g. every 60 minutes or per configurable interval) export aggregated EC data and its on-chain proof anchors for management review and regulatory audit.

Telemetry Interface:

- The EIF shall support export using one or more of the following standard protocols:
  - OpenTelemetry Protocol (OTLP) v1.0.
  - OpenMetrics/Prometheus text or exposition formats.

This allows flexible integration with diverse observability backends and dashboarding tools. Entities such as operations teams or external auditors can consume these metrics to drive real-time alerts, visualize energy consumption trends, or perform anomaly detection. Furthermore, the use of standardized telemetry ensures vendor-neutral compatibility and future-proofing, enabling seamless extension across deployment environments. Crucially, export formats shall include cryptographic metadata - such as on-chain anchors or hash references - so that any tampering after EC data commitment can be detected by cross-checking telemetry records against ledger traceability.

6.1.4 Robustness through multi-tool attestation

For sensitive or regulatory-critical domains, energy records shall pass attestation from at least two independent toolchains before being considered valid. For example, a single UE's session-level energy usage may need to be simultaneously confirmed by both an internal telemetry agent (e.g. Scaphandre) and an external socket-level monitor (e.g. Netio PowerBox). This redundancy enhances validation robustness and mitigates risks of instrumentation failure or data fabrication.

6.2 Smart contracts for service enforcement

6.2.1 Smart Contract Design

1) Smart contracts serve as automated compliance agents for energy-aware service governance. To maintain architectural clarity and scalability, three dedicated contract types are defined, each aligned with a specific stage in the EC governance workflow:Smart Contract of Measurement Validator (SC-MV):
   - **Inputs:** VerificationBlock {timestamp, EC#, domain, component_ID, method_ID, value, hash, signature}.
   - **Outputs:** ValidationCompleted(event, verdict) - indicating Pass or Fail.
     Deployed on all domain DLE-Peer nodes, this contract validates EC records through final integrity checks - such as matching uplink energy at cUPF with downlink energy at DU. **SC-MV** does not trigger other contracts autonomously; instead, it emits a **ValidationCompleted** event when it receives a Verification Block written on-chain.

2) Smart Contract of Policy Enforcement (SC-PE):
   - **Inputs:**
     1) ValidationCompleted (event, verdict == Pass) from SC-MV.
     2) EC metrics retrieved on-chain.
   - **Outputs:** PolicyOutcome(event, {penalty | compliance})
     Invoked externally - typically at regular intervals - by the **Session Manager** in each domain. The Session Manager periodically retrieves newly validated EC records via emitted **SC-MV** events or on-chain queries. It then calls the **SC-PE** to evaluate SLA compliance (based on EC#4 and EC#5 metrics) or carbon budget adherence. **SC-PE** independently enforces corrective actions or issues compliance outcomes, without requiring synchronous invocation from **SC-MV**.

3) Smart Contract of Incentive & Token Management (SC-ITM):

- **Inputs:**

  1) PolicyOutcome (event, compliance).

  2) Renewable Energy Factor metric (EC#9).

- **Outputs:** IncentiveIssued(event, tokenDetails)
  Called during the **billing or settlement phase**, which may occur daily or weekly. The billing system invokes **SC-ITM** against stored EC data that meets reward criteria (e.g. high Renewable Energy Factor from EC#9). **SC-ITM** processes incentive issuance - minting tokens or updating allocations - based on prevailing governance rules, operating independently of real-time validation flows.

6.2.2 Usage Scenarios and Invocation Mechanisms

- **SC-MV** is triggered each time a verified EC record is appended on-chain by a DLE-Peer node. It ensures final integrity over multi-source measurements and emits corresponding events.
- **SC-PE** is executed by the Session Manager or Orchestration Layer to enforce energy governance policies at the end of each session or defined reporting window. It utilizes SC-MV-provided records to calculate penalties or confirm compliance, logging outcomes on-chain.
- **SC-ITM** is explicitly called by the billing system during settlement, after SC-PE confirms compliance. It mints and allocates incentives according to configured REF thresholds and token policies, maintaining transparent audit trails.

6.2.3 Deployment and Governance

All three contracts are deployed on-chain under the control of a **governance consortium** of domain operators. Contract upgrades or parameter adjustments require collective approval via threshold-signature mechanisms (e.g. $\geq N / 2 + 1$ domain consensus). Each contract incorporates **role-based access controls** that restrict administrative actions - such as upgrades, parameter updates, or fail-safe shutdowns - to authorized governance participants. This ensures secure and auditable lifecycle management aligned with telecommunications-grade standards.

7 Conclusion

The present document specifies potential use cases for multi-domain Energy Consumption (EC) data metering, collection and sharing. Metrics usable to characterize the EC of a service at different granularities are also specified, particularly when the service involves multi-domain telecommunication infrastructures (typically 3GPP networks, 3rd party service providers and possibly end users).

An architecture enhancement is proposed with a focus on the 3GPP networks to enable inter-domain EC data metering and sharing. By integrating PDL service capability, inter-domain EC data metering and sharing can be affected in a decentralized manner with trustworthy data exchange and post verification.

Interested stakeholders are recommended to refer to the present document when designing similar EC data sharing capability within infrastructures, especially in multi-domain scenarios.

History

Version	Date	Status
V1.1.1	November 2025	Publication