

Regulatory Analysis of Machine Learning in Smart Meter Gateway: Legal Metrology Perspective

Mahbuba Moni, Daniel Peters, Florian Thiel

0000-0002-7956-3863

0000-0001-6299-4176

0000-0002-5908-518X

Physikalisch-Technische Bundesanstalt (PTB)

Berlin, Germany

Email: {mahbuba.moni, daniel.peters, florian.thiel}@ptb.de

Axel Sikora

0000-0003-0878-2919

Offenburg University of Applied Sciences

Offenburg, Germany

Email: axel.sikora@hs-offenburg.de

Abstract—The integration of machine learning-based algorithms into regulated measuring instruments raises questions for approval frameworks that govern software-controlled functions and assume stable, fully specified states. In Europe, legally regulated smart metering is shaped by a layered framework combining international legal metrology concepts, EU-level harmonization, and Member-State-specific implementation and certification rules. This paper develops a 13-topic cross-standard assessment for integrating machine-learning-based algorithms into such instruments and applies it to the Smart Meter Gateway, a particularly stringent national case within the European smart-metering landscape. The results show that fixed-state deployment is largely compatible with the current framework, whereas evolving-state deployment produces ten conflicts. The analysis, therefore, clarifies the conditions under which fixed-state deployment can be accommodated within the current architecture and the regulatory extensions required for evolving-state deployment. The resulting structure provides a basis for future comparative analysis across other national smart-metering regimes and comparable regulated measurement domains.

I. INTRODUCTION

MODERN measuring instruments and smart-metering systems increasingly depend on software-based functions whose regulatory assessment must ensure that legally relevant behavior remains identifiable, protected, and verifiable throughout their operation. In the smart-metering domain, this assessment is embedded within a layered regulatory environment that spans international legal metrology, European harmonization and guidance, and national implementation and certification regimes. The Smart Meter Gateway (SMGW) in Germany is a particularly stringent instance of such an environment, in which software affecting tariff application, billing-relevant data handling, protected parameters, or audit logging must satisfy requirements from multiple regulatory frameworks [1], [2]. Any modification to such functions must preserve the integrity of the approved software state. New functionality can therefore be introduced only if the approval, verification, and post-approval control mechanisms that govern the certified device's behavior remain intact.

This requirement becomes more challenging when the new functionality involves machine learning (ML) algorithms. Unlike a conventional software update, which yields a deter-

ministic, fully specified program state before operation, an ML model is defined in part by learned parameters derived from training data, and, in some deployment modes, these parameters may continue to change during operation. The regulatory issue is therefore not the specific ML algorithm, but the stability of the approved model state. Once that state changes after approval, the usual control questions of software identification, integrity verification, logging, update control, and re-examination arise immediately. This paper distinguishes between *fixed-state deployment*, in which the model parameters remain unchanged after approval, and *evolving-state deployment*, in which the model state continues to change during operation.

The International Organization of Legal Metrology OIML D31 [3] provides the conceptual starting point for this analysis by introducing the notions of *dynamic modules*, *snapshot*, and distinct update (*verified vs traced*) pathways for legally relevant software. These concepts enable a structured examination of how adaptive or data-driven functionality should be classified, what constitutes an approved reference state, and how post-approval change should be regulated. However, the SMGW cannot be assessed through OIML D31 [3] alone: its approval environment also includes the requirements of the German National Metrology Institute (PTB) [4] for legally relevant gateway software, the cybersecurity certification framework of the German Federal Office for Information Security (BSI) [5], and adjacent European software regulation and guidance.

Existing studies have investigated the relationship between individual standards, particularly in the context of the EU AI Act and legal metrology [6], and have proposed evidence mechanisms for regulated ML models [7]. Building on these foundations, this paper focuses on ML integration in the SMGW device, which serves as a case study chosen for the strictness of its combined-approval setting. The methodology and resulting assessment structure are intended to be applicable to other software-controlled measuring instruments under comparable multi-layer approval frameworks. The paper makes four contributions:

- 1) A cross-standard regulatory analysis that traces relevant

legal metrology and cybersecurity requirements across the international, European, and national levels and applies them to ML integration in the SMGW.

- 2) A gap analysis across 13 regulatory topics (3 anchor topics and 10 cross-standard control topics), showing where each deployment mode aligns with the current framework and where it remains open or requires further specification.
- 3) A bilingual retrieval workflow that extracts and organizes clause-level evidence from regulatory documents in English and German, providing traceable support for manual assessment.
- 4) A certification-lifecycle perspective identifying the control conditions for fixed-state deployment and the regulatory extensions required for evolving-state deployment.

The remainder of the paper is organized as follows. Section II presents the regulatory frameworks for SMGW. Section III describes the methodology, including deployment modes, illustrative scenarios, corpus selection, assessment structure, and the supporting retrieval workflow. Section IV presents the cross-standard gap analysis. Section V discusses the certification outlook and future work, and finally Section VI concludes the paper.

II. REGULATORY FRAMEWORK

This section outlines the regulatory layers governing ML integration in the SMGW, organized from international through European to national level in legal metrology, and describes how cybersecurity and adjacent horizontal regulation interact when software behavior depends on learned model parameters.

A. Legal Metrology and Software Regulation

At the international level, OIML D31 [3] defines a *dynamic module of legally relevant software* as software whose behavior depends on device-specific parameters that may change during operation, including ML or artificial-intelligence (AI) techniques. The concept of a *snapshot* is a static representation of the module at a specific point in time, including the algorithmic structure, current parameter values, and the history of parameter evolution. Beyond these concepts, OIML D31 [3] establishes general requirements for legally relevant software, including mechanisms for unique software identification, separation between legally relevant and non-relevant software components, protective software interfaces that control data flows into legally relevant modules, and audit trails that document software events and parameter changes. It also distinguishes two software update pathways: *verified update*, which requires subsequent reverification, including on-site inspection and seal renewal, and *traced update* [3], which is logged in an audit trail but does not require reverification, provided the update does not affect legally relevant parameters.

At the European level, the Measuring Instruments Directive (MID) 2014/32/EU [8] defines harmonized essential requirements for measuring instruments. Its software-related provisions in Annex I (Sections 7.6, 8.3, and 8.4) address software identification, software protection, and data integrity.

WELMEC Software Guide 7.2 [9], issued by the European legal metrology cooperation, interprets these MID requirements for software-controlled instruments and distinguishes between Type P (built-for-purpose embedded systems) and Type U (based on universal computer platforms) instruments. It also introduces modular requirement extensions for complex software architectures: O (general-purpose operating systems), L (long-term storage of measurement data), T (transmission of measurement data), D (software download), and S (software separation). ML model updates may resemble downloadable software under D; protected model-state elements may need to be treated as controlled device-specific data; and an ML module that affects legally relevant processing may require clear separation and defined interfaces under extension S. Furthermore, WELMEC Guides 7.3 [10] and 7.4 [11] complement Guide 7.2 [9] by providing reference architectures and exemplary technical solutions for applying the modular software requirements to modern measuring instruments. WELMEC Guide 7.6 [12] further supports this framework with a software risk-assessment methodology for evaluating software and IT-related threats, especially in more complex implementations.

At the national level, PTB-A 50.8 [4] specifies requirements for the gateway, including tariff application, billing-relevant data storage, time synchronization, protection against manipulation, display unit, calibration log, audit log, and others. It also indicates that legally relevant gateway software must remain under strict role and function-based control throughout operation. Tasks such as meter data acquisition, calibration-log maintenance, parameter changes, access control, self-tests, and software update control must remain within the legally relevant software domain [1]. This requirement further requires protection of these functions against interference from other software components and, where non-legally-relevant software is present and separated through a defined protective interface. Accordingly, an ML module that influences tariff logic, billing-relevant functions, or legally relevant parameters cannot be treated as an auxiliary add-on; it falls within the same type-examination burden as other software that affects the gateway's approved behavior.

B. IT Security of the SMGW

The Federal Office for Information Security (BSI) is responsible for defining and certifying the security architecture of the German smart-metering system. This includes developing the technical guidelines (TR-03109 series) [13] for SMGW operation, publishing the Protection Profile (PP) for the Gateway of a Smart Metering System (BSI PP-0073) [5], and certifying SMGW devices through Common Criteria (CC) evaluations. The BSI Security Target defines the Target of Evaluation (TOE) as the part of the SMGW comprising the hardware and software/firmware components that acquire, process, retain, and make available meter data across one or more metering points [5]. The SMGW connects the wide-area network (WAN), the local metrological network (LMN), and the home area network (HAN) and provides controlled communication among meters, gateway administrators, service

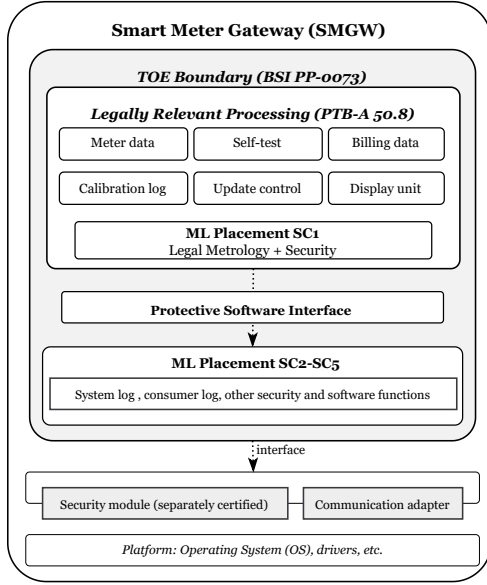


Fig. 1. Reference architecture of the SMGW showing the TOE boundary, legally relevant processing functions, and the placement of ML deployment scenarios SC1–SC5.

technicians, consumers, and external systems. These interfaces are governed by strict access-control and information-flow policies. The TOE can interface with a separately certified security module and with communication adapters that transmit data between the gateway and the external domains.

From a legal metrology perspective, this architecture establishes the trusted technical environment in which legally relevant measurement data are processed, transmitted, and documented. In particular, the SMGW implements a differentiated logging architecture comprising a system log for security and administrative-related events, a consumer log for consumer-related information flows and tariff-related events, and a calibration log for calibration-related events [5], [13]. The calibration log is particularly important because it records legally relevant changes affecting the measuring system. BSI requires calibration-relevant events to remain available for three years from the end of the year in which they were generated [5], in accordance with national regulations. To satisfy this retention requirement, most calibration-log data may be stored outside the TOE, for example, before device reset, de-installation, or when log storage is full. However, certain specified events, such as successful firmware updates, must remain available on the TOE itself [5]. Fig. 1 shows the layered architecture of the SMGW derived from PTB-A 50.8 [4] and BSI PP-0073 [5]. The architecture shown is a reference representation. BSI does not prescribe a specific physical architecture, and actual implementations may vary. The legally relevant functions shown represent a subset of those defined in section II-A.

C. Adjacent Horizontal Regulation

Beyond legal metrology and gateway-specific cybersecurity, ML integration on the SMGW also intersects with broader European regulation. The EU Artificial Intelligence Act (Regulation (EU) 2024/1689) [14] establishes a risk-based regulatory framework for AI systems. According to Article 6(2) in conjunction with Annex III, AI systems intended to be used as safety components in the management and operation of critical infrastructure, including the supply of electricity, gas, heating, and water, may be classified as high-risk AI systems. Such systems must comply with requirements including risk management (Article 9), data governance and training data quality (Article 10), technical documentation (Article 11), logging capabilities (Article 12), transparency obligations (Article 13), and conformity assessment procedures (Article 43) [14].

The Cyber Resilience Act (CRA), Regulation (EU) 2024/2847 [15] introduces lifecycle cybersecurity requirements for products with digital elements. SMGWs are explicitly classified as critical products under Annex IV of the CRA, alongside hardware devices with security boxes and smartcards. Critical products are subject to the most stringent conformity assessment requirements, including the obligation to obtain a European cybersecurity certificate.

The General Data Protection Regulation (GDPR, Regulation (EU) 2016/679) [16] governs the processing of personal data in the European Union. According to Article 4(1), detailed energy consumption data can constitute personal data, as fine-grained load profiles may reveal individuals' behavioral patterns. Processing must therefore comply with the principles defined in Article 5 [16], including purpose limitation and data minimization. Within the German smart metering framework, additional privacy provisions are defined in the MsbG (§ 49–54) [17].

These sources do not replace sector-specific metrology and security rules, but they add interpretive constraints and may shape future guidance for ML-enabled gateway functions.

III. METHODOLOGY

This study combines regulatory analysis with a deterministic retrieval workflow. Topic definitions, clause interpretation, and the compatible/open/conflict judgments are made by manual review. The software workflow extracts text from the corpus, segments it at the clause level, retrieves candidate evidence via bilingual keyword search, and exports reports for review.

A. Deployment Modes and Illustrative Scenarios

ML may support several functions in the SMGW environment, including anomaly detection, load forecasting [20], tariff support, device or consumption-pattern classification, and local decision support for controllable systems. Recent embedded-AI research has shown that compact ML models can run on constrained edge devices for on-device inference [18] and, in some cases, for limited on-device training [19]. In the smart-metering infrastructure, the SMGW is therefore a plausible integration point for such functions.

The two deployment classes introduced in Section I are mapped to two technical operating modes. *Fixed-State Deployment (FSD)* is analyzed through on-device inference, in which the approved model parameters remain unchanged during operation. *Evolving-State Deployment (ESD)* is analyzed through on-device training, in which the parameters continue to change after deployment.

To apply this distinction consistently, the analysis uses five illustrative deployment scenarios (SC1–SC5). These scenarios are methodological reference cases rather than proposed product functions or legal categories. They structure the assessment by varying three parameters: the location of the ML functionality, the legal relevance of its output, and whether the model operates in fixed-state or evolving-state mode (Table I). SC1 is derived from the legally relevant gateway functions defined in PTB-A 50.8 [4] and from the SMGW-specific regulatory framing discussed in [1]. SC2–SC5 represent recurrent ML application types identified in the general and embedded ML research [18], [20], [21], [22] and are mapped against relevant use cases for legal metrology. These scenarios bridge general software control requirements and concrete ML-related questions, providing the reference set used in Section IV to examine how existing requirements apply to different ML placements and operating modes. As shown in Fig. 1, SC1 is placed inside the legally relevant processing domain, where ML output directly affects metrological or billing functions. SC2–SC5 are placed outside the protective software interface and may be classified as legally non-relevant from a metrology perspective. However, because these functions remain within the TOE boundary, they remain subject to the security requirements of BSI [5].

B. Regulatory Corpus

The corpus comprises 21 official regulatory documents (Table II), covering OIML D31 [3], the WELMEC 7.x software guide family (Guides 7.2–7.4, 7.6) [9], [10], [11], [12], PTB-A 50.8 [4], the BSI gateway Protection Profile (PP-0073) with its annex, and selected BSI TR-03109 series documents [5], [13], [23], [24], [25], [26]. Only officially or institutionally published public sources were included. The selection criterion was direct relevance to software control, conformity assessment, or security assurance in the SMGW context. The corpus is used in two roles. The *retrieval corpus* comprises all 21 documents and supports bilingual clause-level evidence search, cross-checks between German (DE) and English (EN) sources, and document tracing. The *analysis corpus* comprises the 11 documents used to operationalize the topic assessment in Section IV and referenced in Tables III and IV. The remaining documents (mainly cryptography-related annexes and specific BSI TR-03109 series parts) were reviewed for context but not used as primary regulatory anchors for the topic verdicts.

C. Assessment Structure

The assessment structure was derived in two stages from the reviewed legal metrology and SMGW security framework, yielding 13 topics in total. In Stage 1, recurring ML-relevant

compliance themes were extracted from OIML D31 [3] to identify three anchor topics: dynamic module classification, snapshot as the reference state, and update-pathway classification. In Stage 2, these anchors were extended into 10 cross-standard control topics by mapping OIML D31 [3], WELMEC 7.x [9], [10], [11], [12], PTB-A 50.8 [4], and the BSI SMGW protection profile (PP) and technical series (TR) [5], [13], [23], [24], [25], [26] against the illustrative deployment scenarios. The resulting topics are arranged into three groups: separation and execution control (T01–T03); state identity, update, and traceability (T04–T08); and storage and risk assessment (T09–T10). Together, these topics capture how the current framework governs the definition, protection, and post-approval treatment of ML functionality in the SMGW.

The analysis is use-case-driven rather than keyword-driven. Searching only for explicit terms such as “artificial intelligence”, “machine learning” or “neural network” would miss much of the regulatory context, as most relevant standards express their constraints in general software language - software protection, parameter protection, software identification, integrity, separation, audit trail, update/download control, etc. Starting from concrete deployment scenarios enables tracing which requirements apply, even when no ML-specific wording appears in the source documents.

Three assessment categories are used: *Compatible*, *Open*, and *Conflict*. Each topic is assessed separately for the two deployment modes (FSD and ESD). Across the corpus, the approved software is treated as a discrete, versioned, and reviewable artifact identified by a fixed hash or version number, examined during type approval, and can be changed only through a controlled update. A topic is classified as *Compatible* when this treatment remains applicable, where the model is frozen after approval and behaves like a conventional certified software item. *Open* denotes that the clause applies but does not yet specify the ML case, leaving a gap that interpretation or a supplementary provision can close. *Conflict* denotes that the deployment mode removes the fixed artifact, in which the hash, version, examined state, and logged version change continuously, so that the clause cannot be met without a new or extended regulatory specification. Regulatory questions are formulated in broad ML terms where possible, though some wording follows neural-network-specific terminology.

D. Tool Architecture

A lightweight, modular workflow supports evidence retrieval and organization (Fig. 2), making evidence collection systematic and preserving provenance across the multilingual corpus. The overall workflow consists of six stages: (1) hybrid PDF text extraction, (2) Markdown loading, (3) clause-boundary detection, (4) bilingual keyword retrieval, (5) report generation, and (6) manual clause selection and reproducibility packaging. In Stage 1, PDF text is extracted using PyMuPDF [27] as the primary extraction method. For documents in which clause numbers appear in the left margin and are not reliably recovered from the text layer, a supplement-

TABLE I
ILLUSTRATIVE ML DEPLOYMENT SCENARIOS USED AS ANALYTICAL REFERENCE CASES

ID	Illustrative scenario	Example	Likely regulatory role	Mode
SC1	Legally relevant gateway software baseline	Tariff application, billing-relevant data storage, time synchronization, protection against manipulation, display unit, calibration and audit logs, meter-data acquisition, parameter changes, access control, self-tests, and software-update control, etc.	Legally relevant by definition	Inf./train
SC2	Load and generation forecasting	Forecasts values later used in tariff application or billing-relevant transformations	Usually advisory; may become legally relevant	Inf./train
SC3	Anomaly and tampering detection	Detects suspected manipulation in meter-data acquisition or gateway operation	Usually advisory; may become legally relevant	Inference
SC4	Predictive maintenance	Predicts faults in gateway components, meters, or communication links for service planning	Usually advisory	Inference
SC5	Documentation support	Supports calibration-log review or software documentation as a secondary aid	Usually advisory; may become legally relevant if it writes or controls protected logs	Inference

TABLE II
REGULATORY CORPUS PROCESSED BY THE RETRIEVAL TOOL (21 DOCUMENTS)

Document Source	Document family	Lang.	Docs
OIML	D31 (software requirements)	EN	1
WELMEC	Guides 7.2–7.4,7.6 (software-related guidance)	EN	4
PTB	PTB-A 50.8 (SMGW requirements)	DE	1
BSI	PP-0073, Annex PP, PKI	EN	3
BSI	TR-03109 series and related annexes	DE	12
<i>Total</i>			21

tary Optical Character Recognition (OCR) pass on rendered page images is used to recover these labels. The recovered labels are merged positionally into the extracted text, and the resulting content is stored in per-document Markdown files with hash-based caching to avoid re-extraction on subsequent runs. In Stages 2–4, the generated Markdown files are loaded, clause boundaries are identified using document-family-specific regular-expression patterns (e.g. 6.3.2.2.5, O3, L2, FAU_GEN.1, 4.3.4.6.3, etc.), and each clause is searched against a bilingual keyword list with text normalization to reduce variation from OCR artifacts and encoding differences. In Stage 5, the retrieved evidence is assembled into review outputs for manual inspection. The automated workflow exports a CSV listing clause-level keyword hits with merged keywords, a review-oriented CSV grouped by regulatory body and topic, and a self-contained interactive HTML report for manual inspection. In Stage 6, relevant clauses are manually selected for inclusion in the paper and consolidated into curated CSV and JSON files for reproducibility. Currently, no automated legal reasoning, semantic reasoning, or sentence-level classification is used.

IV. CROSS-STANDARD ANALYSIS

The analysis follows the 3 anchor and 10 control topics introduced in Section III and represented in Tables III and IV.

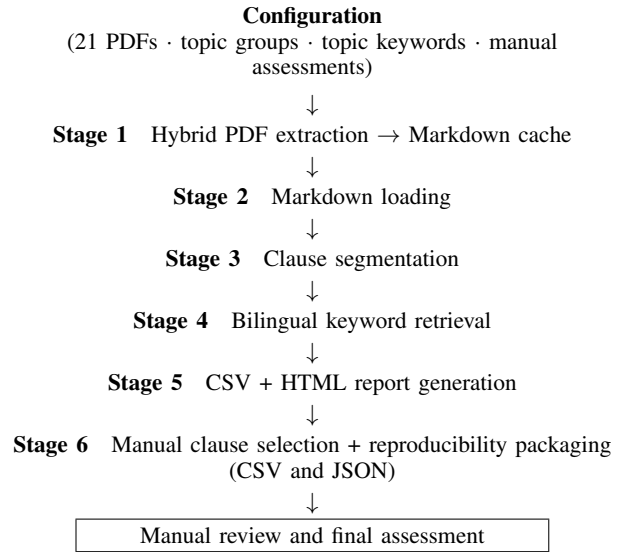


Fig. 2. Retrieval workflow from multilingual PDF sources to clause-level evidence, followed by manual clause selection and final assessment.

A. Anchor Topic

The anchor topics derive OIML D31 [3] into the following three threshold questions for ML-enabled SMGW software. *Dynamic Module Classification (A1)*. Under the dynamic-module concept discussed in Section II-A, OIML D31 [3] (Cl. 3.2.16) explicitly covers ML techniques, and recognizes continuous learning (Cl. 6.3.2.2.5) as a relevant case of software control. The standard also distinguishes *type-specific parameters* (common to all instruments of a given type) from *device-specific parameters* (values bound to the individual instrument). Under this distinction, neural-network weights qualify as device-specific. FSD then aligns with conventional, legally relevant software and is compatible. For ESD — for example, under on-device training in SC1 or SC2, the category is recognized, but the conformity treatment of continuing post-approval evolution required further specification; therefore, the topic remains open.

TABLE III
OIML D31 [3] DERIVED ANCHOR TOPICS. ✓ = COMPATIBLE, ○ = OPEN, × = CONFLICT. FIX. = FIXED-STATE; EVOL. = EVOLVING-STATE.

ID	Anchor Topic	OIML D31 Source	Assessment Question	ML-Specific Implication	Fix.	Evol.
A1	Dynamic Module Classification	Cl. 3.2.16 (definition), Cl. 6.3.2.2.5 (continuous learning)	When should an ML component with changing operational parameters be treated as a dynamic module of legally relevant software?	Adaptive ML is expressly recognized through the dynamic module and continuous-learning concepts; the remaining issue is the conformity treatment of ongoing post-approval evolution.	✓	○
A2	Snapshot as Reference State	Cl. 3.2.56 (definition), Cl. 7.3.2.2	To what extent can a static snapshot serve as the certifiable reference for fixed-state ML under the current framework, and what remains unresolved for the verification of evolving-state ML?	A static snapshot can anchor a certifiable reference for a fixed state ML, whereas the equivalent verification concept needs to be defined for evolving-state ML.	✓	×
A3	Update Pathway Classification	Cl. 6.3.8.3 (verified update), Cl. 6.3.8.4.1 (traced update)	How should post-approval ML-related changes be classified under the verified-update and traced-update pathways?	Existing update pathways provide the regulatory basis for classifying post-approval ML-related change within the software-control framework for fixed state.	✓	×

TABLE IV
SUBORDINATE CROSS-STANDARD CONTROL TOPICS ORGANIZED BY TOPIC GROUP AND BASED ON MANUALLY CURATED EVIDENCE FROM THE REGULATORY CORPUS [3], [4], [5], [9], [10], [11], [12], [23], [24], [25], [26]. ✓ = COMPATIBLE, ○ = OPEN, × = CONFLICT. FIX. = FIXED-STATE; EVOL. = EVOLVING-STATE.

ID	Topic	Assessment Question	Relevant Clauses	Fix.	Evol.
Separation and Execution Control					
T01	Architectural Separation of ML Functions	Can ML functionality be integrated while preserving the required architectural separation between legally relevant and non-relevant software modules?	OIML D31 (6.3.2.2.1); WELMEC 7.2 (S1, S3); PTB-A 50.8 (4.6.3); BSI PP-0073 (4.1)	✓	✓
T02	Protective Interface Control	Under what conditions may an ML component receive input from or influence legally relevant functions through a protective interface?	OIML D31 (6.3.2.2.2); WELMEC 7.2 (S3); PTB-A 50.8 (10.2); BSI PP-0073 (1.1, 1.4.6.5)	✓	○
T03	Execution Timing and Resource Control	Can ML functionality meet the stability requirements for execution priority, runtime behavior, and resource allocation during operation?	OIML D31 (6.3.2.2.4, 6.3.2.2.5); WELMEC 7.2 (O3); PTB-A 50.8 (4.3); BSI TR-03109-1 OpProc. (3.7.10)	✓	×
State Identity, Update, and Traceability					
T04	Identification of Approved ML State	How is the approved ML-relevant software state identified before and after deployment?	OIML D31 (6.2.1); WELMEC 7.2 (P2, U2); PTB-A 50.8 (6.3.1, 7); BSI PP-0073 (1.4.6.7)	✓	×
T05	Integrity and Authenticity of ML State	How can the approved ML software state be protected against unauthorized change and verified for integrity and authenticity?	OIML D31 (6.3.2.1.4, 8.2.2); WELMEC 7.4 (D1); PTB-A 50.8 (4.3, 4.4); BSI PP-0073 (1.4.6.1, FPT_TST.1)	✓	×
T06	Deployment and Software Download Control	How are deployment, replacement, or remote delivery of ML model artifacts treated under current software download rules?	OIML D31 (6.3.8.4.1); WELMEC 7.2 (D1); PTB-A 50.8 (4.6.1.4); BSI PP-0073 (4.2, FAU_GEN.1.2/CAL); BSI TR-03109-1 OpProc. (3.10.1–3.10.4)	✓	×
T07	Type Evaluation and Examination	Do current type-evaluation and examination procedures cover both FSD and ESD ML?	OIML D31 (3.2.49, 6.2.6.1); WELMEC 7.3 (4.1.3); PTB-A 50.8 (4.6.1.4); BSI TR-03109-1 OpProc. (7.4); BSI TR-03109-1 Test, Conformity (1.4)	✓	×
T08	Audit Trail	Can ML-related state changes be logged in a form suitable for verification and traceability?	OIML D31 (6.3.8.4.4); WELMEC 7.2 (P7, U7); PTB-A 50.8 (4.1.1, 4.1.2); BSI PP-0073 (1.4.6.2, FAU_GEN.1); BSI TR-03109-1 OpProc. (3.7.3)	✓	×
Storage and Risk Assessment					
T09	Storage of Data	How must measurement data, ML-derived outputs, and ML-related state evidence be stored under the current framework?	OIML D31 (6.3.4.2); WELMEC 7.2 (L1); PTB-A 50.8 (4.1.3); BSI TR-03109-1 OpProc. (3.12.3); BSI TR-03109-1 Interop. (3.16.19.1)	✓	×
T10	Risk Assessment and Change Impact Evaluation	Are the risks and regulatory implications of post-approval ML-related change subject to structured assessment under the current framework?	OIML D31 (5.2); WELMEC 7.6 (3.3.4, 3.4.2, 4.1, 4.2, 4.2.2, 5.1, 5.1.4); PTB-A 50.8 (4.6.1.2); BSI PP-0073 (3.3, 3.4)	○	×

Snapshot as Reference State (A2). Building on the snapshot (Clause 3.2.56) concept discussed in section II-A, OIML D31 [3] treats the snapshot as a verifiable reference in conformity assessment (Cl. 7.3.2.2). For FSD, this mapping is direct. The snapshot can serve as a stable and hashable reference for type examination and subsequent verification (compatible). ESD breaks this mapping because the parameter state changes during operation, so the approved snapshot no longer matches the deployed model. The current framework does not specify how a continuously evolving ML model can be tied to a verifiable reference state, which creates a conflict.

Update Pathway Classification (A3). The verified and traced-update distinction is summarized in Section II-A (Clauses 6.3.8.3 and 6.3.8.4.1) from OIML D31 [3], which defines how post-approval change is handled. Under FSD, an updated ML model is a discrete replacement of one approved state by another and can be assigned to either pathway within the existing framework (compatible). Under ESD, model-state change occurs during operation rather than through an externally initiated replacement. The current logic assumes discrete, identifiable changes and does not classify continuous post-approval adaptation as such (conflict).

B. Separation and Execution Control

This group addresses how ML functionality is separated, interfaced, and operated within software subject to metrological and security certification. It is most sensitive for SC1, where ML operates inside legally relevant functions, and for SC2 when forecasting outputs enter legally relevant processing. SC3–SC5 are more likely to remain outside the core metrological domain unless their outputs influence protected functions.

Architectural Separation of ML Functions (T01): Architectural separation concerns whether an ML algorithm can be integrated into the SMGW while preserving the required boundary between legally relevant and non-legally relevant software. Across all regulatory layers, the core requirement is architectural rather than algorithmic: software that affects legally relevant functions must remain within a controlled domain and be separated from other software through defined boundaries and interfaces. An FSD model can be assigned to such an approved domain and assessed as part of the certified architecture. In ESD, a parameter change does not, by itself, alter the model's architectural placement, so the separation requirement is satisfied in both modes (compatible).

Protective Interface Control (T02): The protective-interface concept is explicit in OIML D31 [3] and WELMEC 7.2 [9], while PTB-A 50.8 [4] requires controlled software interfaces for legally relevant gateway functions. The BSI [5] framework addresses interface separation at the communication level through the security objective O.SeparateIF, but this is not identical to the legal-metrology concept of a protective software interface. For FSD, the relevant inputs, outputs, and interface conditions can be specified and assessed during type examination (compatible). ESD raises a different question: the external interface may remain unchanged while the behavior

behind it evolves with the model state. Whether an interface approved for one model state remains sufficient once the controlled behavior is no longer static remains an open question.

Execution Timing and Resource Control (T03): A shared requirement of legal metrology and the BSI TR-03109-1 Operational Processes [23] documents is that legally relevant software must remain predictable in its use of processor time, memory, and other system resources. For FSD, the runtime behavior of an ML model can be profiled, bounded, and assessed during type examination, for example, via worst-case execution-time (WCET) [28] analysis and priority scheduling (compatible). For ESD, however, on-device training introduces variable compute and memory demand at runtime. In scenarios such as SC1 and SC2, this variability may interfere with the timing and priority of legally relevant gateway functions on constrained SMGW hardware, producing a conflict.

C. State Identity, Update, and Traceability

State identity, update, and traceability become the most critical issues once the ML model is treated as part of the SMGW's approved software state. The relevant cross-standard questions are whether that state can be uniquely identified, protected against unauthorized change, replaced through controlled update procedures, assigned to an existing update pathway, and documented in a form that remains reviewable after approval. The topic-specific implications are examined below.

Identification of Approved ML State (T04): Identification of the approved ML state requires a unique, verifiable software identity across all regulatory layers, each of which assumes a stable, version-controlled software state. For FSD, an ML model can be versioned and hashed like any other software component, so the existing identification logic remains applicable (compatible). For ESD, the identifier bound to the approved model changes with every parameter update, and no current mechanism maintains a unique, reviewable identity for a continuously adapting model (conflict).

Integrity and Authenticity of ML State (T05): Integrity and authenticity controls across the regulatory layers are anchored to a static approved state, typically via checksums, hash values, digital signatures, or self-test mechanisms. A fixed-state model remains unchanged after approval and can therefore be verified against its certified reference (compatible). Under ESD, the reference hash or digital signature no longer remains stable. Even minor weight changes produce entirely different cryptographic digests [7]. Therefore, current integrity mechanisms require further specification (conflict).

Deployment and Software Download Control (T06): All regulatory frameworks treat software delivery as an authenticated, integrity-checked process with defined update procedures, particularly via an authorized gateway administrator (GWA) for SMGW. In FSD, an ML model is a discrete software artifact that can be packaged and handled within this update process (compatible). Under ESD, parameter changes in SC1 and SC2 occur during operation rather than through an externally initiated replacement. Therefore, the current download and

deployment procedures would need to be extended to cover autonomous model adaptation (conflict).

Type Evaluation and Examination (T07): Type evaluation and examination are based on the assumption that legally relevant software exists as a discrete, reviewable, approved state. That assumption holds for a retrained fixed-state ML model, which can be examined as any other modified firmware component (compatible). It breaks for ESD, the approved state no longer remains fixed between examinations, and current examination procedures are not designed to accommodate continuous post-approval parameter changes, leading to a conflict.

Audit Trail (T08): The reviewed provisions require a reviewable record of software-related and calibration-relevant changes, encompassing audit-trail entries, calibration-log records, logbook functions, and retained event-counter information. For a fixed-state model, a single audit-trail entry at deployment is sufficient once the approved state remains unchanged thereafter (compatible). Under ESD, repeated parameter adaptation generates a potentially continuous stream of new model states; recording each change individually is currently impractical on embedded SMGW hardware (conflict).

D. Storage and Risk Assessment

Two questions are addressed here: how ML-related information must be stored after deployment, and how post-approval changes should be evaluated from a risk perspective. Both arise across all scenarios, but are most demanding in evolving-state cases such as SC1 and SC2, where repeated parameter changes impose additional storage and reassessment burdens. They also matter for SC5 if ML output enters protected logs or compliance-relevant documentation.

Storage of Data (T09): Storage of legally relevant data is covered across all regulatory layers and requires calibration-log entries to be exported and archived by the gateway administrator before firmware updates, with defined retention periods [4], [23], [24]. OIML D31 [3] Clause 6.3.4.2 is particularly relevant, which requires that measurement data produced by a dynamic module be marked as such, and that marking must form part of the legally relevant record. Under FSD, the approved model identifier (T04) remains stable, so a single marking on stored data is sufficient, and existing retention and archiving practices can apply (compatible). Under ESD, the model identifier changes continuously with parameter adaptation, leaving the marking convention, archiving trigger, and retention obligation for ML-derived data unspecified (conflict).

Risk Assessment and Change Impact Evaluation (T10): OIML D31 [3] links examination depth to software risk level, WELMEC Guide 7.6 [12] provides a structured method for risk identification, attack-vector analysis, and risk evaluation. PTB-A 50.8 [4] requires safeguards against known attack scenarios, and the BSI protection profile contributes threat-model and attack-potential concepts for the SMGW security environment [5]. By analyzing these sources, an FSD component can be assessed at design time and upon submission of a discrete update. ML-specific vulnerabilities, however, require

additional criteria, for example, data poisoning through manipulated training datasets, model poisoning through compromised pre-trained components, and adversarial inputs crafted to induce model errors [14]. Therefore, the topic remains open. Under ESD, the problem is more fundamental; it is unclear when a new risk assessment, vulnerability evaluation, or conformity decision should be triggered (conflict).

V. DISCUSSION

This section interprets the cross-standard results as a certification-lifecycle problem, examines the scope and limitations of the retrieval workflow, and identifies the priorities for future work that follow directly from the assessment.

A. Certification Outlook and Lifecycle

Fixed-state deployment yields 12 compatible topics and 1 open topic. By contrast, evolving-state deployment results in 10 conflicts, 2 open topics, and only 1 compatible topic (Tables III and IV). The central result is a structural mismatch between continuous model-state evolution and a certification framework predicated on identifiable, integrity-protected, and reviewable software states.

These results can be illustrated as three lifecycle conditions for fixed-state deployment: (i) *Model-state identification* (A2, T04): the fixed-state model is versioned and hashed as a firmware component, covering the full parameter state at certification. (ii) *Controlled update channel* (A3, T06): a retrained model replacing the deployed one passes through either the verified or the traced update pathway, depending on whether legally relevant functions change. (iii) *Traceability* (T08): audit log entry at model deployment suffices, as parameters do not change during operation; the OIML D31 [3] snapshot concept serves as the certifiable reference state.

For evolving-state deployment, the 10 conflicts reflect a fundamental design mismatch between the static-image certification model and a continuous-change operating mode. Resolving this mismatch requires, at a minimum, four extensions. The EU AI Act [14], as a horizontal regulation applicable to high-risk AI systems that continue to learn after deployment, provides independent regulatory precedent for each of these extensions. Their application to sector-specific SMGW certification remains to be resolved through harmonized standards (Article 40) [14]. The four extensions can be assumed as follows: (i) *Pre-defined adaptation envelope*: under which post-approval parameter changes are admissible only if specified in advance and included in the approved technical documentation. Changes outside those bounds constitute a substantial modification requiring renewed examination (Article 43(4), Recital 128) [14]; (ii) *Re-examination trigger condition*: classifying parameter changes as either a traced update within pre-approved bounds, requiring no re-examination or a verified update exceeding those bounds, subject to renewed type examination, monitored through the post-market monitoring plan (Article 72) [14]; (iii) *Pre-authorized learning permission*: granted by the Gateway Administrator (GWA) at deployment, permitting bounded autonomous

parameter changes while retaining the authority to monitor and intervene (Article 14) [14]; and (iv) *Checkpoint-based parameter log*: recording a cryptographic hash of the model parameter state at defined intervals, providing verifiable traceability without per-update storage overhead.

B. Retrieval Workflow and Limitations

The retrieval workflow served as a structured evidence-management tool for tracing clause-level provisions across the multilingual corpus. The workflow has several limitations. Scanned pages without embedded text can be processed through OCR, but the quality and the recovery of marginal clause numbers remain imperfect, especially in dense layouts and annex material. Keyword matching remains context-free and may retrieve passages that use relevant terminology in unrelated contexts. Integrity-related topics are particularly prone to such noise because general cryptographic clauses share vocabulary with specific integrity provisions. Clause extraction relies on regulatory document-specific regular-expression patterns, which improve coverage but may still miss irregular numbering schemes or assign matches to an imprecise nearby clause. The described methodology remains manual, relying on compatible/open/conflict categorizations. Furthermore, as technical and empirical validation are left for future work, the present study remains conceptual in scope. Different reviewers can classify borderline cases differently.

C. Future Work

The priorities for future work follow directly from the assessment results. Fixed-state deployment appears closer to the existing SMGW regulatory framework, but still requires technical and procedural validation before it can be considered practically deployable. By contrast, evolving-state deployment raises issues that are not only technical but also regulatory.

Fixed-state feasibility on SMGW hardware: The absence of conflicts for fixed-state deployment identifies a regulatory path, but it does not demonstrate implementation feasibility on the SMGW-class hardware. Future work should therefore examine whether compact, certifiable ML models can operate within the gateway's power, memory, and execution constraints. This includes model quantization, secure model packaging, and experiments on metrological separation between ML functionality and legally relevant software components.

Controlled deployment of fixed-state ML: Beyond hardware feasibility, FSD also requires a more detailed study of the certification and operational controls surrounding approved ML artifacts. In particular, future work should examine secure update delivery, rigorous model-state identification, and audit logging for deployment and replacement events. These issues do not produce conflicts in the present assessment, but they remain important conditions for making fixed-state ML deployable in practice on certified gateways.

Bounded Retraining and Post-Approval Control: Evolving-state deployment would require explicit regulatory extensions rather than technical adaptation alone. One possible direction is a bounded retraining category, with defined thresholds,

hardware-feasible audit mechanisms, and an ML-specific update pathway. Such an approach would address the conflicts concentrated in state identity, update control, and traceability. OIML D31 [3] suggests that certain parameter adaptations may remain admissible if properly constrained and audit-logged. This recommendation can be extended to a certified parameter envelope within which bounded ML adaptation can occur without automatically triggering a full re-examination.

Transferability: Applying the 13-topic assessment structure to other EU member-state smart-metering regimes and, more broadly, to other legally controlled measuring instruments would help determine whether the ESD conflicts identified in this paper are structurally general or specific to the German SMGW framework. This would test the extent to which the same ML-related control questions arise across regulated measurement domains beyond energy metering. Future work could also examine how other regulated domains, such as medical devices or intelligent transport systems, handle ML-related software adaptation, to assess whether their approaches could inform the regulatory treatment of machine learning in legal metrology.

VI. CONCLUSION

This paper examined the integration of machine learning into the SMGW within the applicable international, European, and national regulatory frameworks [3], [9], [4], [5]. Fixed-state deployment can be accommodated within the current approval architecture, provided that model-state identification, a controlled update channel, and deployment-time audit logging are in place. Evolving-state deployment, by contrast, lies outside this architecture and would require at least four standardized elements. This includes clear limits on admissible adaptation, a rule specifying when renewed examination is required, defined human oversight, and feasible logging of model-state changes. OIML D31 [3] offers the closest starting point through its dynamic module, snapshot, and update pathway concepts, while the EU AI Act provides an independent regulatory reference for the same concerns in high-risk AI systems that continue to learn after deployment. The 13-topic assessment structure developed here is therefore best understood as a cross-standard analytical framework derived from recurring control questions across the regulatory layers. Its applicability beyond the SMGW, however, should be validated through comparative assessment in other software-controlled measuring instruments.

APPENDIX

Table V lists the 21 processed documents, with a stable identifier (D01–D21), issuing body, short label, source language, year, and full title. Titles are reproduced with current source PDFs, including their edition and version markers, so that the specific revisions used here remain identifiable.

Table VI presents the 13-topic configuration: three anchor topics (A1–A3) and ten control topics (T01–T10), grouped into Separation and Execution Control, State Identity, Update and Traceability, and Storage and Risk Assessment. Each row

lists the topic identifier, name, group, and parallel EN/DE keyword sets. Keywords are drawn from recurring clause-level vocabulary with additional related terms included to capture closely associated meanings.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the colleagues from PTB's Working Group 8.51 (Metrological Software) for their support and valuable contributions.

GENERATIVE AI DISCLOSURE STATEMENT

During the preparation of this work, the first author used Claude (Anthropic) and ChatGPT (OpenAI), in order to improve the language, clarity of the manuscript, structure the regulatory analysis, and generate initial code for the clause-level retrieval workflow described in Section III-D. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the content of the publication.

REFERENCES

- [1] U. Grottker, M. Esche, and M. Elfroth, "PTB-Anforderungen 50.8 an BSI-zertifizierte Smart Meter Gateways," *PTB-Mitteilungen*, vol. 125, no. 3, pp. 23–30, 2015. <https://doi.org/10.7795/310.20150304>
- [2] K. Förderer, M. Lösch, R. Növer, M. Ronczka, and H. Schmeck, "Smart meter gateways: Options for a BSI-compliant integration of energy management systems," *Appl. Sci.*, vol. 9, no. 8, art. 1634, 2019. <https://doi.org/10.3390/app9081634>
- [3] OIML, *D 31: General requirements for software controlled measuring instruments*, International Organization of Legal Metrology, Paris, France, 2023. [Online]. Available: https://www.oiml.org/en/files/pdf_d/d031-e23.pdf
- [4] Physikalisch-Technische Bundesanstalt (PTB), *PTB-Anforderungen PTB-A 50.8 "Smart Meter Gateway"*, Ausgabe Dezember 2014. <https://doi.org/10.7795/510.20150728C>
- [5] Bundesamt für Sicherheit in der Informationstechnik, *Protection Profile for a Smart Meter Gateway (SMGW-PP)*, BSI-CC-PP-0073-V2-2024, Version 2.0, 2024. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03109/PP0073_SMGW.html
- [6] F. Thiel, "The European Artificial Intelligence Act and Legal Metrology," *OIML Bulletin*, vol. LXVI, no. 3, 2025. [Online]. Available: <https://www.oiml.org/en/publications/oiml-bulletin/2025-03/20250306>
- [7] M. Esche, L. Ho, M. Nischwitz, and S. Glesner, "Practical security of evidence for regulated artificial intelligence modules," in *Position Papers of the 20th Conf. on Computer Science and Intelligence Systems*, ser. *Annals of Computer Science and Information Systems*, vol. 44, 2025, pp. 23–30. <http://dx.doi.org/10.15439/2025F4971>
- [8] "Directive 2014/32/EU of 26 February 2014 on the harmonisation of the laws of the Member States relating to the making available on the market of measuring instruments (recast)," *Official Journal of the European Union*, 2014. [Online]. Available: <https://eur-lex.europa.eu/eli/dir/2014/32/oj/eng>
- [9] "WELMEC Guide 7.2: Software Guide (EU Measuring Instruments Directive 2014/32/EU)," Version 2025. [Online]. Available: https://www.welmec.org/welmec/documents/guides/7.2/2025/WELMEC_Guide_7.2_2025.pdf
- [10] "WELMEC 7.3: Reference Architectures Based on WELMEC Guide 7.2," 2020. [Online]. Available: https://www.welmec.org/welmec/documents/guides/7.3/2020/WELMEC_Guide_7.3_v2020.pdf
- [11] "WELMEC Guide 7.4: Exemplary Applications of WELMEC Guide 7.2," 2022. [Online]. Available: https://www.welmec.org/welmec/documents/guides/7.4/2022/WELMEC_Guide_7.4_v2022.pdf
- [12] "WELMEC Guide 7.6: Software Risk Assessment for Measuring Instruments," 2021. [Online]. Available: https://www.welmec.org/welmec/documents/guides/7.6/2021/WELMEC_Guide_7.6_v2021.pdf
- [13] Bundesamt für Sicherheit in der Informationstechnik, *Detailspezifikationen zur TR-03109-1, Version 2.0*, 2024. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publicationen/TechnischeRichtlinien/TR03109/TR-03109-1_Detailspezifikation_v2_0.pdf?__blob=publicationFile&v=2
- [14] "Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act)," *Official Journal of the European Union*, 2024. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>
- [15] "Regulation (EU) 2024/2847 of the European Parliament and of the Council of 23 October 2024 on horizontal cybersecurity requirements for products with digital elements and amending Regulations (EU) No 168/2013 and (EU) 2019/1020 and Directive (EU) 2020/1828 (Cyber Resilience Act)," *Official Journal of the European Union*, 2024. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2024/2847/oj/eng>
- [16] "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," *Official Journal of the European Union*, 2016. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- [17] "Gesetz über den Messstellenbetrieb und die Datenkommunikation in intelligenten Energienetzen (Messstellenbetriebsgesetz – MsbG)," 2016. [Online]. Available: <https://www.gesetze-im-internet.de/messbg/MsbG.pdf>
- [18] J. Huang, Y. Zhu, L. Xu, Z. Zheng, W. Cui, and M. Sun, "On-device training of PV power forecasting models in a smart meter for grid edge intelligence," *arXiv:2507.07016*, 2025. [Online]. Available: <https://arxiv.org/abs/2507.07016>
- [19] M. Rüb and A. Sikora, "A practical view on training neural networks in the edge," *IFAC-PapersOnLine*, vol. 55, no. 4, pp. 272–279, 2022. [Online]. Available: <https://doi.org/10.1016/j.ifacol.2022.06.045>
- [20] X. Wang, H. Wang, B. Bhandari et al., "AI-empowered methods for smart energy consumption: A review of load forecasting, anomaly detection and demand response," *Int. J. Precis. Eng. Manuf.-Green Techn.*, vol. 11, pp. 963–993, 2024. [Online]. Available: <https://doi.org/10.1007/s40684-023-00537-0>
- [21] V. Artiushenko, S. Lang, C. Lerez, T. Reggelen, and M. Hackert-Oschätzchen, "Resource-efficient Edge AI solution for predictive maintenance," *Procedia Computer Science*, vol. 232, pp. 348–357, 2024. [Online]. Available: <https://doi.org/10.1016/j.procs.2024.01.034>
- [22] M. Moni, D. Peters, F. Thiel, and A. Sikora, "Assessing the viability of on-device language models in legally controlled measuring instruments," in *Proc. 2025 IMEKO Joint Conf. TC8–TC11–TC24*, Torino, Italy, 2025. [Online]. Available: <https://doi.org/10.21014/tc11-2025.036>
- [23] Bundesamt für Sicherheit in der Informationstechnik, *Technische Richtlinie BSI TR-03109-1: Anlage VI: Betriebsprozesse, Version 2.0*, 2024. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publicationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Betriebsprozesse_v2_0.pdf?__blob=publicationFile&v=2
- [24] Bundesamt für Sicherheit in der Informationstechnik, *Technische Richtlinie BSI TR-03109-1: Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems, Version 2.0*, 2024. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publicationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anforderungen_Interoperabilitaet_2_0.pdf?__blob=publicationFile&v=2
- [25] Bundesamt für Sicherheit in der Informationstechnik, *Testspezifikation zur Technischen Richtlinie TR-03109-1, Version 1.0*, 2024.
- [26] Bundesamt für Sicherheit in der Informationstechnik, *Testspezifikation zur Technischen Richtlinie TR-03109-1 Version 2.0*, 2025. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publicationen/TechnischeRichtlinien/TR03109/TR-03109-1_Testspezifikation_Konformitaet.pdf?__blob=publicationFile&v=3
- [27] PyMuPDF Documentation, "PyMuPDF," [Online]. Available: <https://pymupdf.readthedocs.io/>
- [28] D. Peters, M. Peter, J.-P. Seifert, and F. Thiel, "A secure system architecture for measuring instruments in legal metrology," *Computers*, vol. 4, no. 2, pp. 61–86, 2015. [Online]. Available: <https://doi.org/10.3390/computers4020061>

TABLE V
FULL DOCUMENT CORPUS USED BY THE RETRIEVAL TOOL. TITLES FOLLOW THE NAMES SHOWN IN THE SOURCE DOCUMENTS.

ID	Issuing body	Short label	Lang.	Year	Document title
D01	OIML	D31	EN	2023	General requirements for software controlled measuring instruments (OIML D 31, Edition 2023 (E))
D02	PTB	PTB-A 50.8	DE	2014	PTB-Anforderungen: Smart Meter Gateway (PTB-A 50.8)
D03	WELMEC	WELMEC 7.2	EN	2025	WELMEC Guide 7.2: Software Guide (EU Measuring Instruments Directive 2014/32/EU)
D04	WELMEC	WELMEC 7.3	EN	2020	WELMEC 7.3: Reference Architectures Based on WELMEC Guide 7.2
D05	WELMEC	WELMEC 7.4	EN	2022	WELMEC Guide 7.4: Exemplary Applications of WELMEC Guide 7.2
D06	WELMEC	WELMEC 7.6	EN	2021	WELMEC Guide 7.6: Software Risk Assessment for Measuring Instruments
D07	BSI	PP-0073	EN	2024	Protection Profile for a Smart Meter Gateway (SMGW-PP), Version 2.0
D08	BSI	Annex PP	EN	2024	Annex to Protection Profile for a Smart Meter Gateway (SMGW-PP) – Functional Package Multiple Grid Connection Points, Version 1.0
D09	BSI	TR-03109-1 OpProc.	DE	2024	Technische Richtlinie BSI TR-03109-1, Anlage VI: Betriebsprozesse mit Beteiligung des SMGW, Version 2.0
D10	BSI	TR-03109-1 Interop.	DE	2024	Technische Richtlinie BSI TR-03109-1: Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems, Version 2.0
D11	BSI	TR-03109-1 Detail	DE	2024	Detaillspezifikationen zur TR-03109-1: Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems, Version 2.0
D12	BSI	TR-03109-1 Webservices	DE	2024	Technische Richtlinie BSI TR-03109-1, Anlage II: Webservice-APIs des SMGW, Version 2.0
D13	BSI	TR-03109-1 Lifecycle	DE	2024	Technische Richtlinie BSI TR-03109-1, Anlage VIII: Lebenszyklus, Version 2.0
D14	BSI	TR-03109-1 Test	DE	2024	TS-03109-1: Testspezifikation zur Technischen Richtlinie TR-03109-1, Version 1.0
D15	BSI	TR-03109-1 Conformity	DE	2025	TS-03109-1: Testspezifikation zur Technischen Richtlinie TR-03109-1 Version 2.0
D16	BSI	TR-03109-3 Crypto	DE	2014	Technische Richtlinie TR-03109-3: Kryptographische Vorgaben für die Infrastruktur von intelligenten Messsystemen, Version 1.1
D17	BSI	TR-03129-4 PKI	EN	2017	Technical Guideline TR-03129-4: Protocols for the Management of Certificates and CRLs in Public-Key-Infrastructures (PKIs), Part 4: Smart Metering Infrastructure, Version 1.3.1
D18	BSI	TR-03109-4 PKI	DE	2017	Technische Richtlinie BSI TR-03109-4: Smart Metering PKI – Public Key Infrastruktur für Smart Meter Gateways, Version 1.2.1
D19	BSI	TR-03109 Control	DE	2025	Implementierungshinweis zur TR-03109-1: Steuerung mit Nachweisführung im SMGW
D20	BSI	TR-03109 TAF	DE	2025	Implementierungshinweis zur TR-03109-1: Ereignisbasierte Tarifanwendungsfälle TAF5 und TAF16
D21	BSI	TR-03109-1 Impl.	DE	2022	Implementierungshinweis zur TR-03109-1: Einheitliche Verwendung der Geräte-ID im LKS1

TABLE VI
FULL TOPIC CONFIGURATION WITH BILINGUAL KEYWORD SETS (13 TOPICS)

ID	Name	Group	EN Keywords	DE Keywords
A1	Dynamic Module Classification	Anchor	dynamic module, continuous learning, artificial intelligence, machine learning, adaptive software	dynamisches Modul, kontinuierliches Lernen, maschinelles Lernen
A2	Snapshot as Reference State	Anchor	snapshot, reference state, neural network, topology and weights, parameter state	Schnappschuss, Referenzzustand, neuronales Netz, Topologie und Gewichte, Parameterzustand
A3	Update Pathway Classification	Anchor	verified update, traced update, software update, reversion, parameter change	verifizierte Aktualisierung, nachverfolgbare Aktualisierung, Softwareaktualisierung, Nacheichung, Parameteränderung
T01	Architectural Separation of ML Functions	Separation and Execution Control	software separation, separation of software, non-relevant software, software domain, extension S, separation kernel, SeparateIF, HAN-Separation, physically separated ports	Softwaretrennung, Trennung der Software, Erweiterung S, Separationskernel, Trennung dieser Netzwerke, Netzwerk-Separation
T02	Protective Interface Control	Separation and Execution Control	protective interface, information flow control, extension S, software interface, data flow policy	Schutzschnittstelle, Informationsflusskontrolle, Erweiterung S, Softwareschnittstelle, Datenflusspolitik
T03	Execution Timing and Resource Control	Separation and Execution Control	execution priority, real-time requirement, extension O, continuous learning, prioritization, resource allocation, system resources, operating system	Ausführungspriorität, Echtzeitanforderung, Erweiterung O, kontinuierliches Lernen, Ressourcenzuweisung, Programmlaufüberwachung, metrologisch relevanter, Betriebssystem, Speicherfullstand
T04	Identification of Approved ML State	State Identity, Update, and Traceability	software identification, software version, version number, hash value, checksum, approved software	Softwareidentifikation, Softwareversion, Versionsnummer, Hashwert, Prüfsumme, zugelassene Software
T05	Integrity and Authenticity of ML State	State Identity, Update, and Traceability	integrity check, integrity verification, software integrity, authenticity verification, tamper detection, self-test, digital signature	Integritätsprüfung, Softwareintegrität, Authentizitätsprüfung, Manipulationserkennung, Selbsttest, Selbsttests, digitale Signatur
T06	Deployment and Software Download Control	State Identity, Update, and Traceability	software download, extension D, software update, firmware update, remote update, software delivery	Software-Download, Erweiterung D, Firmware-Update, Fernaktualisierung, Softwarebereitstellung
T07	Type Evaluation and Examination	State Identity, Update, and Traceability	type evaluation, type examination, reversion, re-examination, conformity assessment, security assurance, security assurance requirements, security target evaluation, evaluation assurance level, assurance components, evaluation and certification, product certification, EAL	Baumusterprüfung, Nacheichung, erneute Prüfung, Konformitätsbewertung
T08	Audit Trail	State Identity, Update, and Traceability	audit trail, audit log, calibration log, event log, event counter, calibration log export, archive, archiving, retention period, retention periods, evidence	Audit-Trail, Eichprotokoll, Ereignisprotokoll, Ereigniszaehler, Kalibrierprotokoll, Eichtechnisches Logbuch, eichtechnisches Log, Logbuch, Eichlog
T09	Storage of Data	Storage and Risk Assessment	long-term storage, data retention, extension L, measurement data storage, billing data, storage of data, stored measurement data, automatic storage	Langzeitspeicherung, Datenaufbewahrung, Erweiterung L, Messdatenspeicherung, Abrechnungsdaten, Speicherung von Daten, gespeicherte Messdaten, Vorhaltefristen, Aufbewahrungsfrist, Archivierung, Datenspeicherung, historischen Messwerte
T10	Risk Assessment and Change Impact Evaluation	Storage and Risk Assessment	risk assessment, risk analysis, change impact, Risk Evaluation, reassessment, threat model, vulnerability assessment, attack potential, threat analysis, impact assessment	Risikobewertung, Risikoanalyse, Änderungsauswirkung, Neubewertung, Angriffsszenarien