An End-to-End Trustworthy Knowledge Graph Engineering Methodology

Emna Amdouni
IRT SystemX,
Palaiseau, France
emna.amdouni@irt-systemx.fr

Lucas Mattioli
IRT SystemX, Onera
Palaiseau, France
lucas.mattioli@irt-systemx.fr

Faouzi Adjed
IRT SystemX,
Palaiseau, France
faouzi.adjed@irt-systemx.fr

Afef Awadid
IRT SystemX,
Palaiseau, France
afef.awadid@irt-systemx.fr

Martin Gonzalez
IRT SystemX,
Palaiseau, France
martin.gonzalez@irt-systemx.fr

Loic Cantat
SafenAI,
Paris, France
loic@safenai.io

Juliette Mattioli
Thales SA, cortAIx,
Palaiseau, France
juliette.mattioli@thalesgroup.com

***Abstract*—Existing knowledge graph engineering methodologies provide limited support for governance, quality assessment, and accountability across the lifecycle, particularly in collaborative and industrial settings. This limits the use of knowledge graphs in safety-critical and high-risk AI systems subject to regulatory and ethical requirements, such as the EU AI Act. We propose an end-to-end Trustworthy Knowledge Graph (TKG) engineering methodology structured into three complementary dimensions: a methodology dimension covering KG construction phases, from knowledge elicitation and modeling to validation and deployment; a lifecycle dimension capturing continuous use and updates; and a transverse trustworthiness dimension integrating governance and quality assessment across all phases.**

Keywords-Trustworthy AI Engineering; Knowledge Graph's Lifecycle; Knowledge Graph Engineering; Trustworthiness Assessment.

I. Introduction

A. Trustworthy AI Engineering for Regulatory Compliance

As Artificial Intelligence (AI) grows in capability and scale, ensuring its reliability, robustness, and alignment with human intent is critical. Trustworthy AI Engineering [1] has emerged as a paradigm for developing and operating AI systems, especially in high-stakes and safety-critical domains [2]. This field integrates software engineering, systems engineering, cybersecurity, ethics, design, and cognitive science to address the complex challenges of modern AI. Its goal is to ensure AI systems are technically sound, accurate, ethical, transparent, compliant with regulations such as the EU AI Act, and resilient in uncertain, dynamic environments.

This emerging discipline tackles the uncertainty, limited reproducibility, and restricted verifiability of AI systems' behavior and decisions. Unlike conventional software, whose behavior can usually be deterministically specified and verified, many AI models function as "black boxes" [3], limiting assurances of consistency, safety, and value alignment. Deploying trustworthy AI in environments with ambiguity, non-stationarity, and adversarial threats demands a lifecycle approach spanning design, data engineering, deployment, monitoring, and maintenance, while embedding fairness, accountability, transparency, and robustness [4]. Trustworthy AI engineering therefore integrates advanced AI with safety, cybersecurity, reliability, ethical, and regulatory requirements to ensure systems are both innovative and compliant [5].

B. Key Concepts of Knowledge Graphs

Over the last decade, **data-driven AI** has become dominant, overshadowing symbolic AI. Connectionist and statistical methods mimic the brain's data-driven learning and excel at image and pattern recognition, but are limited in high-level reasoning, problem-solving and interpretability. **Knowledge-based AI** (symbolic AI) instead uses formal logic, rule-based systems and structured knowledge. In the Cartesian tradition, it defines intelligence via axioms, logical inference and domain expertise. Unlike connectionist systems, which learn implicit statistical correlations, symbolic AI encodes knowledge transparently, supporting precise reasoning, verifiability and adaptation to complex rule-governed settings.

In safety-critical domains such as aerospace, healthcare and industrial automation, marked by uncertainty and complexity, it is crucial to combine heterogeneous techniques. Integrating physics-based approaches (*e.g.*, differential equations and mechanistic models) with data-driven methods (*e.g.*, neural networks) yields **hybrid AI** that enables theoretically grounded, robust and adaptable decision-making.

A **Knowledge Graph** (KG) is a structured representation of entities and relationships, typically modeled as a graph. It enables semantic integration, reasoning, and explainability, making it suitable for industrial AI systems [6]. KGs are commonly formalized using the Resource Description Framework (RDF), where knowledge is expressed as triples (subject, predicate, object). For example, "*Symbolic AI is a subdiscipline of AI*" can be represented as $(\text{Symbolic AI}, \text{subdiscipline}, \text{AI})$. This formalism enables machine-processable and semantically rich knowledge [7].

KGs are large networks of entities and relations that explicitly model connections across domains, unlike traditional databases. For example, a KG might not only store that "*AI engineering is a new discipline*" but also link it to related methodologies and tools. Modern KGs use NLP, machine learning, and data mining to automatically extract, refine,

and update their contents, distinguishing them from static ontologies or taxonomies [6]. Systems like Google's Knowledge Graph and Microsoft's Satori use web-scale extraction to continually expand their repositories. KGs are application-agnostic, supporting use cases from semantic search (*e.g.*, enriching search results with context) to decision-support (*e.g.*, recommending treatments from patient data and medical literature). By integrating heterogeneous data into a unified, queryable layer, KGs have become central to AI-driven analytics and generative AI.

A KG is a dynamic, graph-structured knowledge base that formally represents entities, their attributes, and relationships. Combining semantic richness, scalability, and automated knowledge acquisition, KGs turn raw data into actionable insights, driving advances in AI, data science, and beyond.

C. Outline of the Study

While the Confiance.ai program [8] offers a tool-based methodology for ML-based system engineering, this study targets trustworthy engineering for symbolic AI. We examine the lifecycle of KG-based systems, focusing on development phases and the integration of KG-specific qualification. We then review evaluation measures needed for KG qualification. This work complements the ML engineering body of knowledge of the "European Trustworthy AI Association" [9], which defines an end-to-end methodology for trustworthy ML-based AI engineering.

II. A Trustworthy KG Engineering Methodology

A. Limits of Existing KG Engineering Methodologies

The current state of the art in knowledge engineering lacks accurate methodologies to address knowledge explicability, traceability, auditability, versioning, and governance in KGs within a trustworthy AI industrial context.

Early R&D ontology-centric methodologies such as METHONTOLOGY (METH) [10] and NeOn [11] focused on knowledge elicitation, conceptual modeling, and ontology formalization. NeOn added flexibility by integrating alignment, modularity, refinement, and evaluation to support ontology and graph development for real use cases. These methods emphasize requirement analysis, conceptualization, formal representation, implementation, and evaluation, but offer limited support for operational lifecycle management, advanced semantic quality evaluation, knowledge graph (KG) maintenance, and governance.

The On-To-Knowledge Methodology (OKTM) [12] was among the first to adopt a lifecycle view of ontology engineering for industrial use, defining phases such as feasibility study, baseline ontology development, refinement, evaluation, and maintenance. However, it does not explicitly address traceability, trustworthiness, or governance. More recent industry-oriented approaches, such as LOT4KG [13], summarize the KG lifecycle into four stages: implementation, publication, maintenance, and update. This work focuses on constructing a KG from an ontology and on knowledge updating and change analysis, but does not adequately address validation, evaluation, KG usage and integration within AI systems, or governance. Trust-related aspects are also not explicitly defined as lifecycle objectives, despite their importance for industrial AI applications.

Across these R&D methodologies, core knowledge engineering activities (e.g., elicitation, modeling, implementation, publication, and update) are well addressed. However, three major gaps remain: absence of an explicit trust-by-design lifecycle, limited integration of governance and accountability roles, and lack of continuous monitoring mechanisms for trustworthiness during operational usage.

In industry, Neo4j, a leading graph database provider, offers technical pipelines for automated KG construction from heterogeneous sources such as text, tabular data, relational databases, and ontologies. A typical pipeline includes data ingestion, KG modeling, data mapping, KG construction, enrichment, maintenance, and usage [14]. However, these pipelines do not explicitly address KG validation and quality assessment and do not define a comprehensive end-to-end KG lifecycle, especially regarding monitoring and continuous governance.

We propose a KG-centric Trustworthy Knowledge Graph (TKG) framework that combines construction and lifecycle. Unlike ontology-driven approaches, it enables continuous KG evolution without requiring a predefined ontology, while integrating refinement, validation, usage, update, and governance to ensure trust. Ontology-level integration is left for future work.

B. Trustworthy KG Engineering Dimensions and Phases

We describe the TKG framework through three complementary dimensions: (i) a **methodology dimension** defining the phases required to build a KG version, (ii) a **lifecycle dimension** governing its continuous evolution, and (iii) a **transverse trustworthiness dimension** ensuring governance, traceability, and quality across all phases.

The methodology involves five **main actors (domain expert, KG developer, curator, publisher, and end-user)** who contribute to KG engineering.

We refer to *knowledge artifacts* as all outputs of KG engineering, including specifications, conceptual models, knowledge graphs, and underlying data sources.

The following phases belong to the methodology dimension and describe the construction of a single KG version.

- **Knowledge Elicitation**: *Lead*: Domain expert. *Participant*: KG developer. Identify the scope of the use case, operational requirements, user stories, trustworthiness attributes, and data sources in collaboration with domain experts.
- **KG Modeling**: *Lead*: KG developer. *Participants*: Domain expert(s). Design the semantic model based on elicited requirements, defining domain concepts, relationships, and constraints, as well as trust-related metadata. Such metadata captures the provenance of the information stored in the KG, including its origin, how it was produced, and who it was validated. It also supports

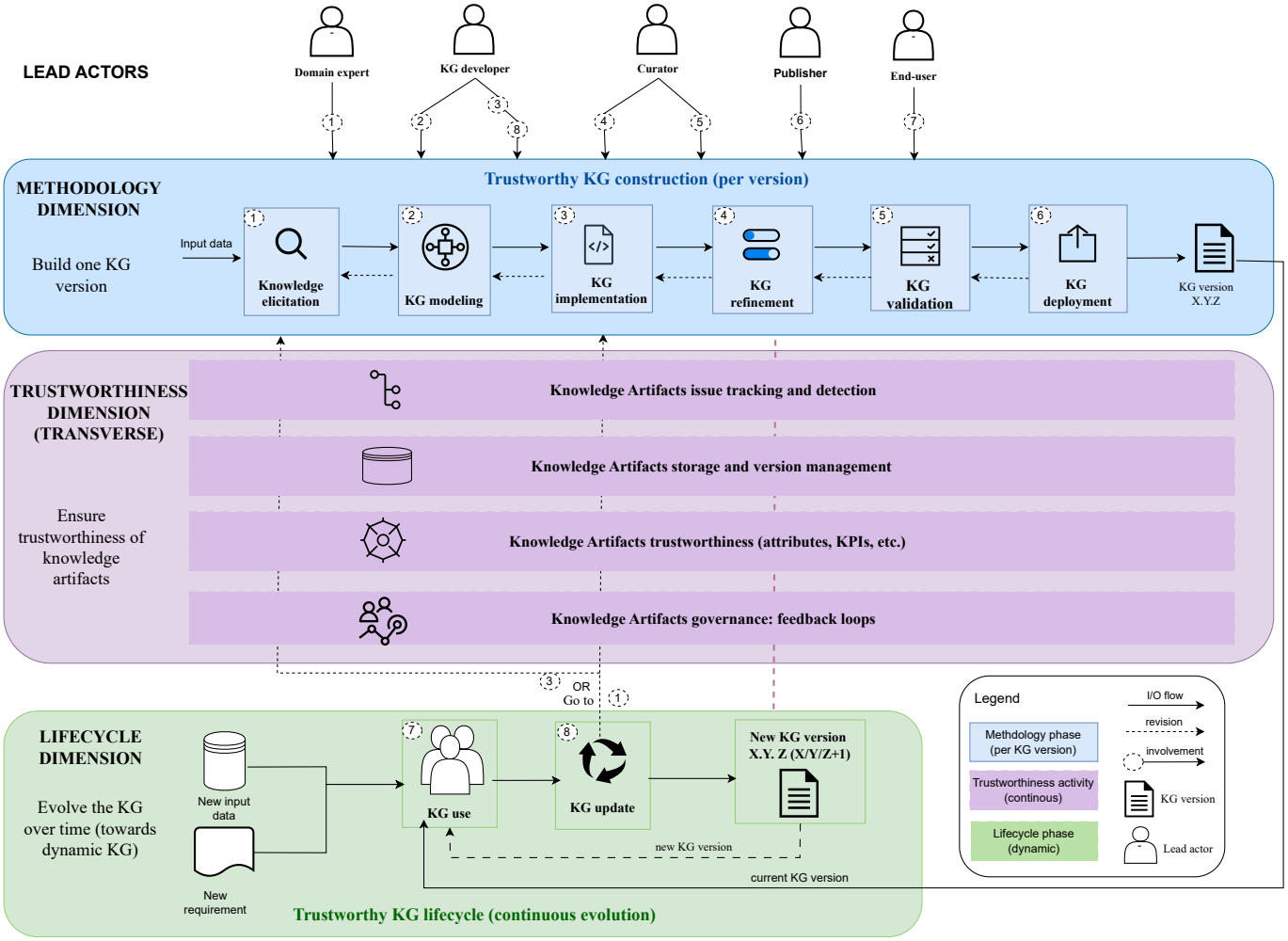

Figure 1. End-to-end TKG engineering with three dimensions: methodology (construction), lifecycle (evolution), and transverse trustworthiness (governance, traceability, and quality) applied to knowledge artifacts.

governance and maintenance processes. These aspects can be represented using established vocabularies such as Prov-O [15], Dublin Core Terms (DCT) [16], and the Data Quality Vocabulary (DQV) [17].

- **KG Implementation**: *Lead*: KG developer. Develop the RDF-based KG according to the conceptual model and agreed trust-related metadata (dct:source, prov:wasDerivedFrom, prov:wasGeneratedBy, etc.), ensuring traceability and version management.
- **KG Refinement**: *Lead*: Curator. *Participant*: KG developer and domain expert. Improve the quality of the resulting KG by adding new facts. This phase focuses on correcting errors, removing inconsistencies, improving the schema, merging duplicate entities and enriching relationships (better tying or clearer hierarchy). The refinement improves the correctness, consistency, precision and structure of what already exists in the resulted KG.
- **KG Validation**: *Lead*: Curator. *Participant*: KG developer and domain expert. Evaluate KG trustworthiness through KPIs and structured validation processes, including semantic, data, and expert validation. Verify compliance with constraints, detect inconsistencies, and document validation.
- **KG Deployment**: *Lead*: Publisher. *Participant*: KG developer. Deploy validated KG version through persistent URIs, machine-readable formats, and human-readable documentation to ensure transparency and reuse. In particular, define licensing, access policies, and usage governance to ensure transparency, reproducibility, and responsible reuse.
- **KG Usage**: *Lead*: End-user. *Participant:* publisher and curator. Knowledge graph usage should be transparent and controlled through secured APIs and catalogues. The objective of this phase is to control KG usage, track access logs, and detect new updates.
- **KG Update**: *Lead*: KG developer. *Participant*: curator. Incorporate new or changed data/requirements into the KG. It focuses on adding entities/relationships, removing out-dated facts, and adopting new changes. Updates are managed through trustworthiness mechanisms.

The trustworthiness dimension operates continuously across both the methodology and lifecycle dimensions, including

version management, issue tracking, KPI monitoring, provenance capture, and feedback loops. These mechanisms ensure auditability, accountability, and controlled evolution of the KG in industrial trustworthy AI systems.

C. Added Value of our Engineering Methodology

The proposed TKG lifecycle does not replace existing KG engineering methodologies, but extends them towards a trust-oriented engineering. As shown in Table I, key phases such as elicitation, modeling, implementation, and publication are already covered. We use the labels **"Explicit"**, **"Implicit"**, **"Limited"**, and **"Absent"** to indicate the level of support.

However, these phases are not integrated into a trust-oriented framework. Ontology approaches focus on defining knowledge schemas, while KG approaches focus on data construction and processing, without fully addressing trust across the KG lifecycle.

Our approach addresses this gap by introducing a KG-centric, lifecycle-oriented methodology structured into three complementary dimensions: a methodology dimension for building KG versions, a lifecycle dimension for their continuous evolution, and a transverse trustworthiness dimension.

III. Trustworthiness Attributes and Assessment

A. Trustworthiness Attributes

Trustworthiness attributes are fine-grained, measurable properties that define specific quality dimensions of an AI system. They clarify what constitutes trust in critical AI systems and fall into three capability areas: technical, usage, and governance. "Technical" covers verification of an AI component's validity and robustness; "Governance" concerns fundamental rights; and "Usage" addresses transparency, explainability, and usability. These attributes also cover relationships with third parties, especially quality assurance, audit, and certification.

Based on the EU AI Act and the European AI HLEG guidelines, trustworthy AI consists of six main requirements: robustness, effectiveness, dependability (including safety and security), usability, human agency (including transparency, interpretability and explainability), and human oversight (including ethical aspects). These characteristics are defined as follows:

- **Robustness** describes the system's ability to maintain its desired performance and functionality even when faced with challenging conditions, such as dealing with adversarial, uncertain, or imprecise inputs;
- **Effectiveness** is a measure of its ability to perform the functions necessary to achieve goals or objectives; it specifies the ability of a system to deliver a service that can be justifiably trusted;
- **Usability** denotes the degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a defined context of use.
- **Human agency** refers to the capacity of individuals to interact with, understand, and control the AI systems, ensuring these technologies are transparent, explainable, and aligned with human intentions;
- **Human oversight** encapsulates the evaluation and guidance of AI systems they operate within legal frameworks, fundamental rights, and general benevolence.

Trustworthiness can only be assessed when the Operational Design Domain (ODD) is clearly defined, specifying the conditions in which the AI system operates. Many AI prototypes fail to do this. Trustworthiness measures can identify issues before failures, support improvements in critical systems, and help designers build reliable, safe, and secure systems. No single assessment covers all trust dimensions, so trade-offs are required. Trustworthiness also spans the broader AI lifecycle, involving actors and processes such as engineers, operators, certification authorities, and insurance companies.

B. KG Effectiveness Assessment

In this section, we reuse the operational definition of correctness from [18] and summarize it for completeness. Among the six higher-level requirements introduced above, we focus on **effectiveness**, as it is most directly tied to KG content quality. Effectiveness measures how justifiably the KG can be trusted. To operationalize this notion, we decompose effectiveness into five complementary sub-dimensions, each with an associated metric.

Notation. Let:

- r denote the set of triples in the *produced* KG (assessed);
- r^* denote the set of triples in the *reference* KG (ground-truth);
- $r_{\text{crt}} = (r \cap r^*)|_{\Pi r}$ denote the *correct* subset of r, defined as the projection of the intersection $r \cap r^*$ into r, i.e., the triples in r that are confirmed as correct by r^*;
- $r_{cpt} = (r \cap r^*)|_{\Pi r^*}$ denote the *covered* subset of r^*, defined as the projection of the intersection $r \cap r^*$ onto r^*, i.e., the triples in r^* that are retrieved by r;
- $r \setminus r_{crt}$ denote the set of triples in r that are *not* confirmed by r^* (spurious or erroneous facts);
- $r^* \setminus r_{cpt}$ denote the set of triples in r^* that are *not* covered by r (missing facts).

All metrics take values in $[0, 1]$, where 1 indicates perfect performance on the corresponding dimension.

Correctness: Ensuring Factual Accuracy and Truthfulness - Correctness [19] concerns the factual accuracy and truthfulness of information encoded in the KG. It requires verifying that entities are correctly identified, relationships accurately reflect real-world connections, and attribute values match ground truth. It is the degree to which a produced answer matches the reference output *without introducing new spurious content*.Noted μ_{correct} [20] it is defined as $\mu_{\text{correct}} = 1 - \frac{|r \setminus r_{\text{crt}}|}{|r|} \in [0, 1]$, with $r_{\text{crt}} = (r \cap r^*)|_{\Pi r}$ being the KG resulting from the projection of the intersection of r and r^* on r and $r \setminus r_{\text{crt}}$ is the complementary of r in r^*

Completeness: Capturing the Domain Knowledge - Complementary to correctness, completeness [21] measures whether the system returns all required elements of the reference output. It reflects how fully the KG captures all relevant

TABLE I. COMPARISON OF ONTOLOGY ENGINEERING AND KNOWLEDGE GRAPH APPROACHES

Engineering Activities	Ontology Engineering			Knowledge Graph Approaches	
	METH	NEON	OTKM	LOT4KG	TKG
Elicitation / Specification	Explicit	Explicit	Limited	Explicit	Explicit
Conceptual modeling / Design	Explicit	Explicit	Implicit	Explicit	Explicit
Implementation	Explicit	Explicit	Explicit	Explicit	Explicit
Refinement / Enrichment	Absence	Limited	Explicit	Explicit	Explicit
Validation / Evaluation	**Implicit**	**Implicit**	**Limited**	**Limited**	**Explicit**
Deployment / Publication	Absence	Absence	Limited	Explicit	Explicit
Usage	**Limited**	Explicit	Explicit	**Absence**	**Explicit**
Update / Maintenance	**Limited**	**Limited**	**Limited**	Explicit	**Explicit**
Governance	**Absence**	**Limited**	**Limited**	**Limited**	**Explicit**

entities, relationships, and attributes within its intended scope. The completeness (denoted as μ_{complete}) measures the quantity of r contained in r^*. It is formerly defined as: $\mu_{\text{complete}} = 1 - \frac{|r^* \setminus r_{\text{cpt}}|}{|r^*|} \in [0, 1]$,

Consistency: Maintaining Internal Logical Coherence - Consistency assesses the internal logical coherence of the KG, ensuring that assertions do not conflict and that representations follow defined constraints and business rules [22]. It includes syntactic consistency, where structures follow established patterns, and semantic consistency, where relationship meanings remain uniform across the graph.

Logical Consistency quantifies the absence of contradictions: $\mu_{\text{LC}} = 1 - (\frac{|contradictions|}{|r|})$ measuring logical coherence by tracking contradictions discovered during reasoning processes relative to the total number of inferences made.

Representativeness: Faithfully Reflecting Domain Distributions - Representativeness concerns whether the KG accurately reflects the real distribution and characteristics of its domain. It focuses on coverage biases, ensuring minority cases are properly represented alongside dominant patterns and that the graph does not systematically favor certain entity or relationship types.

Timeliness: Maintaining Currency and Relevance - Beyond the core dimensions, Timeliness measures whether information remains current and relevant. Thus, a metric associated to timeliness $\mu_{\text{timeliness}}$ can be defined as [23] $\mu_{\text{timeliness}} = \max(0, 1 - \frac{currency}{volatility}) \in [0, 1]$, In the formula, $currency$ correspond to the age of the data when delivered to the user and $volatility$ is the length of time the data remains valid.

IV. ILLUSTRATION ON THE BODY OF KNOWLEDGE OF THE TRUSTWORTHY ML ENGINEERING

A. The Body of Knowledge

A body of knowledge (BoK) is "*structured knowledge employed by members of a discipline to inform their practice or work*" [24]. BoK design is widely used to define and model concepts through knowledge acquisition, fusion, storage, and retrieval. Knowledge is acquired from diverse data types by extracting entities, attributes, and relations, and is typically stored in KG databases. Thus, BoK design

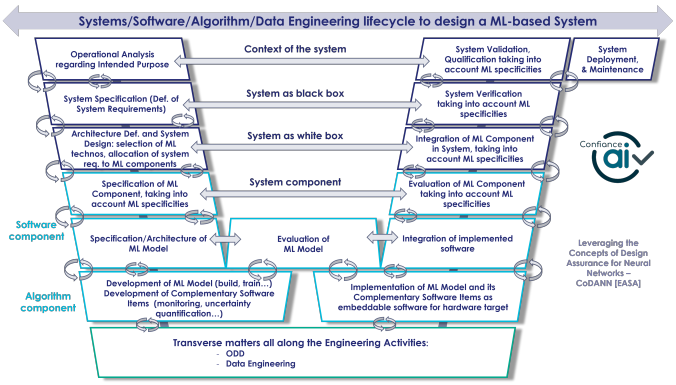

Figure 2. The view of the ML Engineering BoK - https://bok.confiance.ai/

entails developing a KG that provides users with comparable problem-solving capabilities. In our context, the BoK (see figure 2) serves as the "ground truth" for ML Engineering [25]. Its core components are the concepts, knowledge, skills, standards, terminology, guidelines, practices, and activities that define a field or specialization. It includes data repositories, performance indicators, and other tools to ensure reliable, trustworthy ML engineering, and spans multiple engineering domains [26].

B. Main Issues of Trustworthiness

As with any symbolic model, a BoK can only ever be an approximation of reality. New observations based on ML engineering use cases can inform the acquisition of further knowledge. Therefore, evaluating the accuracy of the knowledge represented with respect to reality is essential for creating an adequate model. These limitations are related to the symbol grounding problem [27], and concern the extent to which representational elements are hand-crafted rather than learned from data. Thus, several features must be taken into account when developing a BoK:

- **Redundancy**: Are there any knowledge items that are identical or equivalent to another (subsumed)?
- **Consistency**: Is there inconsistency, ambiguity or indeterminacy? Is it deliberate? Are there multiple outcomes?

- **Minimality**: Can the knowledge set be reduced/ simplified? Is the shortened version logically equivalent to the original?
- **Completeness**: Does the knowledge set include all entries?

A well-designed BoK should have: representational accuracy, to capture all necessary knowledge; inferential adequacy, to manipulate representations and generate new knowledge consistent with existing structures; inferential efficiency, to guide reasoning effectively by storing relevant information; and acquisition efficiency, to easily incorporate new knowledge, preferably through automatic methods.

Peer reviews were carried out with various stakeholders (data scientists, software and systems engineers, and cybersecurity and safety engineers, among others) to assess the appropriateness and quality of the acquired knowledge in relation to the end-to-end ML engineering methodology [28].

V. Conclusion and future works

This paper analyzes the structural and governance dimensions of knowledge graph engineering through a combined methodology and lifecycle perspective, enabling the continuous construction and evolution of KG-centric systems. A formal specification of input/output artifacts, activities, and tasks, as well as deeper investigation of dynamic aspects, remains future work. The CSIA program has adapted this lifecycle to AI engineering via the ML Engineering Body of Knowledge [25][26] and is systematically revising the BoK to rigorously assess and improve its consistency, minimality, and completeness. The proposed lifecycle extends existing KG engineering methodologies by embedding governance, quality assurance, and reliability requirements in all phases, as required for safety-critical AI systems. We also introduce a structured taxonomy of reliability attributes and explore quantitative techniques for evaluating KG effectiveness, enabling systematic, actionable, and measurable assessment.

Acknowledgment

This work has been supported by the French government under the "France 2030" program, as part of the SystemX Technological Research Institute within the CSIA Project.

References

[1] M. Adedjouma et al., *Towards the Engineering of Trustworthy AI Applications for Critical Systems. The Confiance.ai Program*, 2022.

[2] J. Perez-Cerrolaza et al., "Artificial intelligence for safety-critical systems in industrial and transportation domains: A survey", *ACM Computing Surveys*, vol. 56, no. 7, pp. 1–40, 2024.

[3] V. Hassija et al., "Interpreting black-box models: a review on explainable artificial intelligence", *Cognitive Computation*, vol. 16, no. 1, pp. 45–74, 2024.

[4] M. Poretschkin et al., "Guideline for Trustworthy Artificial Intelligence–AI Assessment Catalog", *arXiv preprint arXiv:2307.03681*, 2023.

[5] J. Mattioli et al., "AI Engineering to Deploy Reliable AI in Industry", in *5th Int. Conf. on Transdisciplinary AI (TransAI)*, 2023, pp. 228–231.

[6] D. Fensel et al., "Introduction: What Is a Knowledge Graph?", in *Knowledge graphs: Methodology, tools and selected use cases*, Springer, 2020, pp. 1–10.

[7] Z. Chen et al., "Knowledge Graph Completion: A Review", *IEEE Access*, vol. 8, pp. 192 435–192 456, 2020.

[8] K. Quintero et al., "An End-to-End Method for Operationalizing Trustwothiness in AI-Based Critical Systems", in *15th Int. Conf. on Performance, Safety and Robustness in Complex Systems and Applications (PESARO)*, 2025.

[9] European Trustworthy AI Association, *European trustworthy ai association*, Accessed: 2026-04-20, 2025. [Online]. Available: https://www.trustworthy-ai-association.eu/.

[10] M. López, "METHONTOLOGY: from ontological art towards ontological engineering", in *Proceedings of the…*, 1997.

[11] S. Singhania et al., "NeOn: News Entity-Interaction Extraction for Enhanced Question Answering", *arXiv preprint arXiv:2411.12449*, 2024.

[12] Y. Sure et al., "On-to-knowledge: Semantic web-enabled knowledge management", in *Web Intelligence*, Springer, 2003, pp. 277–300.

[13] R. Pernisch et al., "When ontologies met knowledge graphs: Tale of a methodology", in *European Semantic Web Conf.*, Springer, 2024, pp. 286–290.

[14] Neo4j, Inc., *Building knowledge graphs: A practical guide*, https://neo4j.com/developer/knowledge-graph/, Accessed: 2026-04-17, 2023.

[15] W3C Provenance Working Group, *Prov-o: The prov ontology*, https://www.w3.org/TR/prov-o/, W3C Recommendation, 2013.

[16] Dublin Core Metadata Initiative, *Dublin core metadata element set, version 1.1*, https://www.dublincore.org/specifications/dublin-core/dces/, DCMI Recommendation, 2012.

[17] W3C Data on the Web Best Practices Working Group, *Data quality vocabulary (dqv)*, https://www.w3.org/TR/vocab-dqv/, W3C Recommendation, 2016.

[18] J. Mattioli et al., "A Brief Overview of Key Quality Metrics for Knowledge Graph Solution Illustration on Digital NOTAMs", in *AAAI Symposium Series*, vol. 7, 2025, pp. 206–213.

[19] C. Laudy et al., "HLIF2024: a Competition for High-Level Information Fusion", in *2024 27th International Conf. on Information Fusion (FUSION)*, IEEE, 2024, pp. 1–8.

[20] C. Laudy and N. Museux, "How to evaluate high level fusion algorithms?", in *2019 22th International Conf. on Information Fusion (FUSION)*, IEEE, 2019, pp. 1–8.

[21] S. Issa et al., "Knowledge graph completeness: A systematic literature review", *IEEE Access*, vol. 9, 2021.

[22] J. Lehmann et al., "Quality assessment for linked data: A survey.", *Semantic Web (1570-0844)*, vol. 7, no. 1, 2016.

[23] O. Hartig and J. Zhao, "Using web data provenance for quality assessment.", *SWPM*, vol. 526, 2009.

[24] T. Ören, "Toward the body of knowledge of modeling and simulation", in *Interservice/industry training, simulation, and education Conf. (I/ITSEC)*, vol. 2005, 2005.

[25] J. Mattioli et al., "Leveraging Knowledge Graph to design the Machine-Learning Engineering Body-of-Knowledge", in *2024 Conf. on AI, Science, Engineering, and Technology (AIxSET)*, IEEE, 2024, pp. 258–265.

[26] J. Mattioli et al., "ML System Engineering Supported by a Body of Knowledge", in *Proceedings of the 16th International Joint Conf. on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, vol. 3, 2024, pp. 331–338.

[27] S. Harnad, "The symbol grounding problem", *Physica D: Nonlinear Phenomena*, vol. 42, no. 1-3, pp. 335–346, 1990.

[28] M. Adedjouma et al., "Engineering Dependable AI systems", in *2022 17th Annual System of Systems Engineering Conf. (SOSE)*, IEEE, 2022, pp. 458–463.