Leveraging Semantic Knowledge Graphs for Smart Manufacturing Integration via Asset Administration Shells

Mohamad Wael Kheshfeh, Sebastian Linke, Markus Müller, Tobias Tute and Rüdiger Mecke
Department of Engineering and Industrial Design, Magdeburg-Stendal University of Applied Sciences, Breitscheid Str. 2, 39114 Magdeburg, Germany
mohamad.kheshfeh@h2.de

Keywords: Asset Administration Shell, Knowledge Graph, Manufacturing Technologies, IIoT Platform, Semantic Integration, Digital Twin, SPARQL, Automated Reasoning.

Abstract: Industrial IoT platforms face semantic interoperability challenges when integrating heterogeneous manufacturing equipment. While Asset Administration Shell (AAS), standardized by IEC 63278-1:2023, provides vendor-neutral digital representations, existing implementations lack automated reasoning capabilities. This work presents a semantic integration framework enhancing AAS with Knowledge Graph technologies through bidirectional transformation mechanisms. Developed within the PlatinA Industrial Internet of Things platform research (infrastructure designed for 50 concurrent devices, currently integrating 3 manufacturing systems), our hybrid architecture combines containerized AAS implementations with Apache Jena Fuseki RDF triple stores. We validate technical feasibility through proof-of-concept implementation using a precision milling system with real-time force monitoring, demonstrating successful AAS-to-RDF transformation preserving IEC 63278 compliance, sophisticated SPARQL-based queries for force deviation detection and equipment capability discovery, and practical integration with existing infrastructure. A three-layer manufacturing ontology enables automated reasoning through SWRL rules for equipment selection, maintenance prediction, and quality assessment. This proof-of-concept establishes the foundation for addressing critical manufacturing challenges including measurement validation (targeting reduction from 80% redundancy to 5-6%) and process anomaly detection (projected 89% automation rate), enabling intelligent resource allocation and cross-system interoperability through knowledge-driven decision making.

1 INTRODUCTION

Modern manufacturing environments struggle to integrate diverse production equipment from multiple vendors into intelligent systems [1]. Proprietary protocols and data formats create information silos preventing effective cross-system communication. While Asset Administration Shell (AAS) frameworks provide standardized virtual representations mirroring real-time operational data [2], implementations traditionally function as passive data repositories lacking autonomous reasoning capabilities essential for intelligent manufacturing. The fundamental challenge lies in enabling digital representations to support automated inference and context-aware decision-making.

Domain experts identified c ritical improvement opportunities in surface finish quality control and process anomaly detection. Current manual measurement validation processes exhibit approximately 80% redundancy due to double-checking require-ments, with experts estimating that automated vali-dation could reduce this to target levels around 5-6%(implying potential 94% measurement efficiency). Similarly, process anomaly detection currently relies entirely on manual expert review, with domain spe-cialists projecting that automated semantic reasoning could detect approximately 89% of anomalies without human intervention[1].

However, achieving these targets requires automated decision-making capabilities beyond current rule-based control and machine learning approaches, which struggle with novel equipment combinations, dynamic material-process matching, and cross-equipment optimization requiring implicit relationship inference.

[1]The estimate of roughly 89 % automatable anomaly detection is based on historical process data from which typical patterns, recurrence rates, and semantically distinguishable deviations can be derived.

This work presents a semantic integration architecture enhancing AAS foundations with Knowledge Graph technologies [3]. Knowledge graphs represent information as networks of interconnected entities, enabling machines to understand contextual connections and derive logical conclusions. We validate technical feasibility through proof-of-concept implementation using a precision milling system with three-axis force monitoring, demonstrating complete AAS-to-RDF transformation, sophisticated SPARQL queries for force deviation detection and equipment capability discovery, and bidirectional synchronization suitable for manufacturing control.

This paper presents a proof-of-concept demonstrating technical feasibility of semantic integration for manufacturing systems. While PlatinA's [2] AAS infrastructure is architecturally designed for 50 concurrent manufacturing devices with 3 systems currently integrated, this work validates semantic enhancement through single-system proof-of-concept. Comprehensive semantic integration across all current and future systems represents ongoing development.

2 RELATED WORK

Asset Administration Shell research has focused on structural interoperability without leveraging semantic reasoning. Platform-based architectures like FIWARE provide cross-organizational interoperability but require significant deployment expertise [4]. The AAS metamodel's native RDF/XML and JSON-LD support enables direct semantic web integration as specified in IEC 63278-1:2023 [2].

The Industrial Digital Twin Association (IDTA) enhanced AAS capabilities through standardized submodel specifications [5], including Process Parameters (IDTA 02031-1) for equipment parameter management [6]. These templates align with semantic knowledge graph approaches, enabling transformation of IDTA-structured information into graph representations without semantic loss.

Knowledge graphs demonstrate potential in industrial domains for data integration and automated reasoning [3]. Schleipen et al. established approaches for converting OPC UA information models to RDF representations [7], transforming hierarchical industrial structures into graph formats optimized for semantic reasoning. However, contemporary research reveals deployment barriers including technical complexity and organizational resistance [8].

[2]https://www.h2.de/forschung/wissens-und-technologietransfer/efre-und-esf-projekte.html

Table 1 positions our work relative to existing approaches. Unlike platform-centric solutions requiring infrastructure replacement, our approach enhances established AAS implementations. While Schleipen et al. demonstrated OPC UA to RDF conversion, their work lacks bidirectional synchronization and real-time reasoning for closed-loop control. Our framework uniquely combines standards-compliant AAS with knowledge graph technologies, automated SWRL reasoning, and bidirectional synchronization, providing incremental enhancement without disruptive changes.

Table 1: Comparison of semantic integration approaches for manufacturing.

Approach	Standards Compliance	Semantic Reasoning	Key Characteristic
FIWARE [4]	Partial (NGSI-LD)	Limited	Requires full infrastructure replacement
Schleipen et al. [7]	OPC UA focus	Yes (RDF)	Uni-directional transformation only
Standard AAS [2]	Full (IEC 63278)	No	Passive data repository
Our Work	Full (IEC 63278)	Yes (SWRL)	Incremental enhancement with bidirectional sync

3 SYSTEM ARCHITECTURE

Our semantic integration leverages a hybrid architecture combining operational AAS data management with semantic knowledge representation using Apache Jena Fuseki RDF stores [9], addressing three requirements:

standards compliance through IEC 63278-compliant AAS [2], semantic intelligence through RDF knowledge graphs [3], and operational efficiency through real-time synchronization.

Figure 1 presents the architecture bridging Asset Administration Shells with Knowledge Graph technologies. Manufacturing systems connect through OPC UA [10] to AAS Infrastructure, where data is encapsulated within standardized AAS representations.

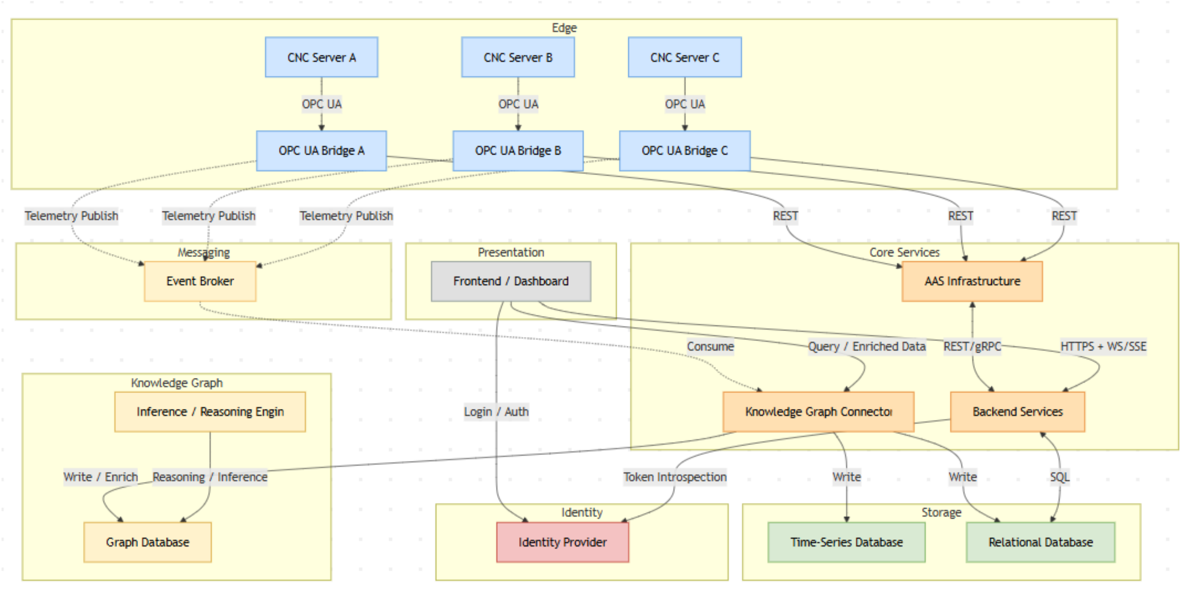

Figure 1: System architecture for semantic integration through knowledge graph-enhanced Asset Administration Shells.

The Knowledge Graph Connector performs semantic transformation, extracting data from AAS Core Services via REST/gRPC protocols [11] and enriching it with semantic annotations using ontological mappings [12]. The Inference/Reasoning Engine implements SWRL-based automated reasoning [13] including anomaly detection and production optimization, feeding insights back for closed-loop control [14].

Real-time telemetry flows f rom manufacturing systems through OPC UA to the Event Broker, where the Knowledge Graph Connector transforms raw data into RDF resources with ontological relationships. The architecture employs hybrid storage: time-series databases for high-frequency operational data, relational databases for structured asset information, and graph databases for semantic relationships.

3.1 Reasoning Rules

The Inference Engine implements SWRL rules for automated knowledge derivation. Equipment selection rules determine compatible equipment based on material and capability requirements through pattern matching on material compatibility and processing capabilities. Maintenance prediction rules identify calibration needs by evaluating temporal validity of last calibration date (threshold: 30 days) combined with force deviation patterns (threshold: 15.0 N deviation). Quality assessment rules detect process anomalies by comparing measured force values against target parameters across three axes simultaneously. For example, the equipment selection rule matches milling systems having compatible materials and required capabilities to process requirements, au-tomatically inferring suitable equipment. The mainte-nance prediction rule triggers calibration recommen-dations when force deviations exceed thresholds and calibration date exceeds 30 days. These rules enable automated reasoning for equipment selection, main-tenance prediction, and quality assessment without manual configuration.

3.2 Ontology Design

Our manufacturing ontology consists of three layers (Fig. 2). The upper ontology incorporates standard vocabularies including AAS metamodel (aas:), QUDT for units (qudt:, unit:), and IDTA submodel specifications (idta:). The domain ontology layer (mfg:) defines manufacturing-specific classes including `mfg:ManufacturingSystem` with subclasses for CNC machines, process-related classes for force measurements, and material classes (Aluminum, Steel, Titanium) with processing capabilities. Object properties establish semantic relationships: `mfg:compatibleMaterial` links equipment to materials, `mfg:hasCapability` connects equipment to capabilities, and datatype properties represent quantitative measurements.

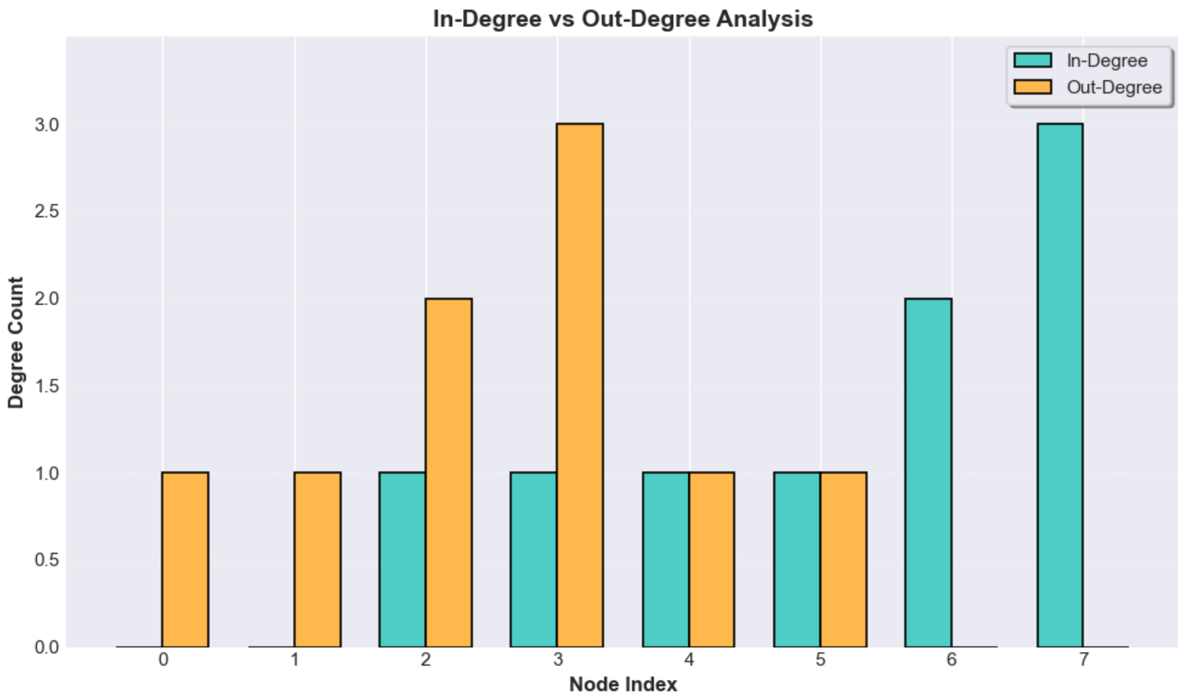

Figure 2: Graph connectivity analysis: node degree distribution in the manufacturing knowledge graph.

4 PROOF-OF-CONCEPT IMPLEMENTATION

We implemented a complete transformation pipeline using a CNC precision milling system with three-axis force monitoring [15], demonstrating systematic conversion of standardized AAS structures into semantically-enriched knowledge graphs while maintaining IEC 63278-1:2023 compliance [2].

The implementation builds upon PlatinA's existing AAS infrastructure integrating heterogeneous equipment through AAS-Tool [16] with OPC UA connectivity [10]. While PlatinA achieves strong performance through rule-based control and machine learning, it lacks automated semantic reasoning for cross-equipment decision-making. Our enhancement introduces a knowledge graph layer transforming operational AAS data into machine-understandable semantic representations supporting automated inference [3]. The milling system generates real-time telemetry including cutting forces along X, Y, Z axes (Newtons), spindle speed and feed rate, and calibration status. It connects via OPC UA [10] to AAS Infrastructure where its digital representation follows IEC 63278 [2] with IDTA Process Parameters submodel (IDTA 02031-1) [6].

```
{"assetAdministrationShells": [{
    "idShort": "MILLING_SYSTEM_001_AAS",
    "id": "urn:manufacturing:milling:001",
    "assetInformation": {
      "assetKind": "Instance",
      "globalAssetId": "urn:manufacturing:milling:001:asset",
      "specificAssetIds": [{
        "name": "ManufacturerID", "value": "DMG_MORI_DMU_50"
      }]},
    "submodels": [{"type": "ModelReference",
        "keys": [{"type": "Submodel",
          "value": "urn:manufacturing:submodel:force-monitoring"}]}]
  }],
  "submodels": [{
    "idShort": "ForceMonitoring",
    "id": "urn:manufacturing:submodel:force-monitoring",
    "submodelElements": [{
      "idShort": "cutting_forces",
      "modelType": "SubmodelElementCollection",
      "value": [
        {"idShort": "force_x", "value": "245.2", "unit": "N"},
        {"idShort": "force_y", "value": "189.7", "unit": "N"},
        {"idShort": "force_z", "value": "312.1", "unit": "N"}
      ]}]}]}
```

Listing 1: Production AAS structure.

This equipment exemplifies typical semantic integration challenges: heterogeneous data formats from industrial sensors [17], real-time processing requirements for production control, context-aware decision support needs, and integration with standardized AAS implementations. The transformation pipeline implements bidirectional synchronization between AAS Infrastructure and Apache Jena Fuseki [9] through a Knowledge Graph Connector. Real-time sensor data flows through OPC UA to the Event Broker, where the Connector extracts AAS submodel data, enriches it with semantic annotations using ontological mappings [12], and transforms it into RDF triples [18]. The Inference Engine processes semantic knowledge using SWRL rules [13] to detect patterns and generate actionable insights flowing back to AAS Infrastructure for closed-loop control [14]. The bidirectional synchronization employs event-driven update propagation with timestamp-based conflict resolution. When the Inference Engine generates new knowledge, changes propagate back to AAS Infrastructure, updating relevant submodel elements while preserving IEC 63278 compliance through last-write-wins semantics. Listing 1 presents the complete AAS structure with hierarchical asset information, submodel references, structured process parameters with real-time measurements, and temporal metadata. This organization follows IDTA specifications [6] while providing semantic-ready data structures suitable for RDF transformation.

```
@prefix aas: <http://www.admin-shell.io/aas/3/0/>
@prefix mfg: <http://example.org/ontology/manufacturing/>
@prefix qudt: <http://qudt.org/schema/qudt/> .
@prefix unit: <http://qudt.org/vocab/unit/> .
<urn:manufacturing:milling:001>
    rdf:type aas:AssetAdministrationShell ;
    aas:idShort "MILLING_SYSTEM_001_AAS" ;
    aas:hasSubmodel <urn:manufacturing:submodel:force-monitoring> .
<urn:manufacturing:milling:001:asset>
    rdf:type mfg:MillingSystem , mfg:CNCMachine ;
    mfg:manufacturer "DMG MORI" ;
    mfg:compatibleMaterial mfg:Aluminum , mfg:Steel .
<element:cutting_forces>
    rdf:type mfg:ForceVector ;
    mfg:forceX [ qudt:numericValue "245.2"^^xsd:double ;
                 qudt:unit unit:N ] ;
    mfg:forceY [ qudt:numericValue "189.7"^^xsd:double ;
                 qudt:unit unit:N ] .
```

Listing 2: Transformed RDF knowledge graph.

The force values shown (245.2, 189.7, 312.1 N) represent typical operational measurements for aluminum milling operations under normal cutting conditions. The Knowledge Graph Connector transforms this AAS structure into RDF through systematic mapping rules. Listing 2 shows the resulting knowledge graph in Turtle notation, preserving structural information while enriching with explicit semantic relationships including type hierarchies enabling inheritance-based reasoning and physical quantities modeled using QUDT ontology supporting automated unit conversion. The semantic knowledge graph enables sophisticated queries impossible with traditional databases. Listing 3 demonstrates real-time force deviation monitoring identifying equipment operating outside normal parameters by comparing measurements against targets across three axes simultaneously.

```
PREFIX mfg: <http://example.org/ontology/manufacturing/>
PREFIX qudt: <http://qudt.org/schema/qudt/>

SELECT ?system ?axis ?current ?deviation
WHERE {
    ?system rdf:type mfg:MillingSystem ;
            aas:hasSubmodel ?forceSubmodel .
    ?forceSubmodel aas:hasSubmodelElement ?cuttingForces .
    ?cuttingForces mfg:forceX ?currentX .
    ?currentX qudt:numericValue ?current .
    BIND("X" AS ?axis)
    BIND(ABS(?current - 150.0) AS ?deviation)
    FILTER(?deviation > 10.0)
}
```

Listing 3: Force deviation detection.

Listing 4 demonstrates cross-equipment capability discovery based on material requirements and technical specifications, combining multiple semantic dimensions including material compatibility and quantitative constraints.

```
PREFIX mfg: <http://example.org/ontology/manufacturing/>

SELECT ?equipment ?manufacturer ?capability
WHERE {
    ?equipment rdf:type mfg:MillingSystem ;
               mfg:compatibleMaterial mfg:Titanium ;
               mfg:manufacturer ?manufacturer ;
               mfg:processingCapability ?capability ;
               aas:hasSubmodel/mfg:positioningAccuracy ?accuracy .
    FILTER(?accuracy <= 0.010)
}
```

Listing 4: Capability discovery.

4.1 Performance Evaluation

Table 2 presents performance measurements for critical operations in the semantic integration pipeline, averaged over 10 executions to account for JVM warm-up and system load variations.

Table 2: Performance metrics for semantic integration pipeline.

Operation	Time	Execution Context
AAS-to-RDF Transform	118ms	State change trigger, asynchronous
Force Deviation Query	43ms	On-demand SPARQL
Capability Discovery	76ms	On-demand SPARQL
SWRL Inference	227ms	Periodic batch (5-10s), Jena Rules
Write-back to AAS	95ms	Event-driven insights
Graph Storage	2.1MB	10,000 RDF triples (Turtle)
Sensor data sync	8 Hz	Direct AAS (no semantic overhead)

4.2 Real-Time Architecture

The semantic integration employs event-driven architecture optimized for real-time operation. AAS Infrastructure maintains 8 Hz synchronization with physical assets (125ms cycle) using conventional time-series storage. Semantic transformation operates asynchronously: continuous monitoring flows real-time sensor data to AAS at 8 Hz without semantic overhead, while selective transformation (118ms) triggers only on significant state changes rather than every sensor reading. Periodic reasoning (227ms) executes in background threads at 5-10 second intervals, with on-demand queries (43-76ms) executing when intelligence is required. Event-driven synchronization (95ms) occurs only when reasoning generates actionable insights. This architecture ensures the semantic layer enhances rather than impedes real-time control, with high-frequency operational data in time-series storage for immediate access while the knowledge graph provides reasoning for decision support tasks where millisecond response is not critical.

Comparison with traditional databases reveals trade-offs. While simple SELECT queries execute in 5-15ms, they cannot express semantic relationships in Listings 3-4 without complex multi-table JOINs and application logic. SPARQL queries combine material compatibility, equipment capabilities, technical specifications, and process parameters in single declarative statements impossible to formulate efficiently in SQL, justifying additional latency for reasoning-intensive tasks.

4.3 Deployment Experience

The proof-of-concept implementation required approximately 6 person-weeks for initial development, including ontology design (2 weeks with 3 domain expert consultations), transformation pipeline implementation (2 weeks), SWRL rule development (1 week), and integration testing (1 week). Integration with existing PlatinA infrastructure was achieved through standardized REST APIs without modifications to core AAS components, demonstrating the incremental enhancement approach. The containerized architecture enabled seamless deployment alongside existing AAS services with minimal infrastructure changes.

5 DISCUSSION

The semantic enhancement introduces computational overhead (118ms transformation, 227ms reasoning) and implementation complexity (ontology engineering, maintenance) that must be justified by specific requirements. Our analysis suggests knowledge graphs provide maximum value in scenarios characterized by: (1) cross-equipment reasoning where decisions span multiple assets, materials, and processes; (2) dynamic material-process matching where novel combinations arise frequently; (3) implicit relationship inference where capabilities, properties, and requirements must be reasoned over without explicit programming; and (4) knowledge preservation where expert manufacturing knowledge should be captured formally.

Conversely, semantic overhead may not be justified for: (1) simple monitoring where time-series queries suffice; (2) well-defined repetitive processes where rule-based control captures decision logic; (3) single-equipment operations without cross-system dependencies; and (4) environments lacking domain expertise for ontology development. The PlatinA platform demonstrates hybrid deployment where traditional databases handle high-frequency sensor data while the knowledge graph addresses reasoning-intensive tasks. This architectural pattern provides practical balance between performance and intelligence.

The incremental enhancement approach mitigates deployment barriers by maintaining existing AAS infrastructure while introducing semantic capabilities selectively where they provide clear value. Manufacturing facilities can begin with limited integration for specific use cases (equipment selection, maintenance scheduling) and expand coverage as expertise develops, avoiding all-or-nothing adoption challenges hindering semantic technology deployment in conservative industrial environments.
This proof-of-concept demonstrates technical feasibility but has limitations requiring future work.

First, validation with a single milling system limits generalizability claims across diverse manufacturing equipment types. While PlatinA's AAS infrastructure is designed for 50 concurrent devices and currently integrates 3 systems, semantic integration has only been validated with one proof-of-concept system. Empirical validation across multiple heterogeneous systems under production loads with hundreds of thousands of RDF triples remains necessary to demonstrate scalability from single-system proof-of-concept to infrastructure-wide deployment. Second, ontology engineering and maintenance require specialized expertise in both semantic technologies and manufacturing domain knowledge, potentially limiting adoption in resource-constrained environments. Third, SWRL reasoning performance (227ms per inference cycle) may require optimization strategies such as pre-computation or incremental reasoning for production deployments with complex rule sets exceeding our current 8 rules. Fourth, the proof-of-concept lacks comprehensive evaluation of failure scenarios, error recovery mechanisms, and long-term operational stability in 24/7 manufacturing environments. Finally, while performance measurements demonstrate acceptable latency for decision support tasks, integration with time-critical control loops requiring sub-100ms response may necessitate architectural refinements or hybrid reasoning approaches combining semantic inference with traditional rule engines.

6 CONCLUSIONS

This proof-of-concept validates technical feasibility of semantic AAS integration through hybrid AAS-Apache Jena Fuseki architecture, systematic AAS-to-RDF transformation maintaining IEC 63278 compliance, and SWRL-based automated reasoning for equipment selection, maintenance prediction, and quality assessment.

Practical applications include automated resource allocation through semantic queries over capability ontologies, cross-equipment process optimization identifying compatible tool-material-machine combinations, and predictive maintenance scheduling combining calibration validity, force deviation patterns, and sensor health indicators.

The primary objective of this work is establishing the semantic foundation necessary for cognitive assistance systems supporting manufacturing operators and engineers. This research contributes foundational infrastructure enabling the PlatinA platform to integrate AI-driven capabilities including natural language interfaces for intuitive equipment interaction, intelligent decision support combining machine learning with semantic reasoning, and context-aware assistance adapting to operator expertise levels and current manufacturing scenarios.

The knowledge graph architecture enables AI systems to understand manufacturing relationships, equipment capabilities, and process constraints, facilitating human-AI collaboration where cognitive assistants augment rather than replace human expertise. This work establishes semantic infrastructure enabling the PlatinA platform to integrate AI-driven capabilities including natural language interfaces, intelligent decision support, and context-aware assistance for manufacturing operators through federated knowl-edge sharing across regional networks.

7 FUTURE WORK

Future work will address current limitations through multi-system validation across diverse equipment types to demonstrate generalizability, empirical scalability testing expanding from single-system proof-of-concept to PlatinA's current 3-device deployment and ultimately to the infrastructure's 50-device design capacity, enhanced reasoning capabilities through temporal and probabilistic reasoning, collaborative ontology engineering methodologies, and comprehensive industrial validation assessing long-term stability and ROI.

ACKNOWLEDGEMENTS

This work is carried out as part of the PlatinA project (ZS/2023/12/182752), supported by funding from the European Union and state of Sachsen-Anhalt.

REFERENCES

[1] M. Resman, M. Pipan, M. Šimic, and N. Herakovič, "A new architecture model for smart manufacturing: A performance analysis and comparison with the rami 4.0 reference model." Advances in Production Engineering & Management, 2019, pp. 153–165.

[2] International Electrotechnical Commission, "Iec 63278-1:2023 asset administration shell for industrial applications - part 1: Asset administration shell structure," IEC, Geneva, Tech. Rep., 2023.

[3] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs." ACM Computing Surveys, 2021, pp. 1–37.

[4] M. Platenius-Mohr, S. Malakuti, S. Grüner, J. Schmitt, and T. Goldschmidt, "File- and api-based interoperability with the asset administration shell." Automation, 2020, pp. 586–597.

[5] Industrial Digital Twin Association (IDTA), "Specification of the asset administration shell part 4: Security," IDTA, Berlin, Tech. Rep. IDTA Specification 01004-3-0, Version 3.0, 2025.

[6] I. D. T. A. (IDTA), "Submodel template "process parameters part 1: Type"," IDTA, Berlin, Tech. Rep. IDTA Specification 02031-1-0, 2025.

[7] M. Schleipen, S. S. Gilani, T. Bischoff, and R. Pichler, "Opc ua & industrie 4.0 - enabling technology with high diversity and variability." Procedia CIRP, 2016, pp. 315–320.

[8] L. D. Xu, E. L. Xu, and L. Li, "Industry 4.0: State of the art and future trends." International Journal of Production Research, 2018, pp. 2941–2962.

[9] Apache Software Foundation, "Apache jena fuseki: A sparql server," 2023, accessed: 2024-10-10. [Online]. Available: https://jena.apache.org/documentation/fuseki2/

[10] OPC Foundation, "Opc unified architecture specification, part 1: Overview and concepts," OPC Foundation, Technical Report IEC 62541-1, 2017.

[11] T. Ahmad, Z. Al-Ars, and H. P. Hofstee, "Benchmarking apache arrow flight – a wire-speed protocol for data transfer, querying and microservices." in Proceedings of the 4th Benchmarking in the Datacenter: Expanding to the Cloud (BID 2022), 2022, pp. 1–10.

[12] M. Grüninger and M. S. Fox, "Methodology for the design and evaluation of ontologies." in Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI, 1995, pp. 1–10.

[13] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "Swrl: A semantic web rule language combining owl and ruleml." W3C Member submission, 2004, pp. 1–31.

[14] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems." Manufacturing letters, 2015, pp. 18–23.

[15] Y. Altintas, "Manufacturing automation: metal cutting mechanics, machine tool vibrations, and cnc design." Cambridge University Press, 2012.

[16] T. Kuhn, P. O. Antonio, and F. Schnicke, "Eclipse basyx – industrie 4.0 einfach machen! das open-source-betriebssystem für die industrie 4.0." atp magazin, 2021.

[17] N. Jazdi, "Cyber physical systems in the context of industry 4.0." 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, 2014, pp. 1–4.

[18] R. Cyganiak, D. Wood, and M. Lanthaler, "Rdf 1.1 concepts and abstract syntax," W3C, W3C Recommendation, 2014.