

A Semantic Backend Architecture for Digital Twin Spacecraft Lifecycle Integration

Rojina Shrestha^{1,*†}, Björn Beyreuther^{2,†}, Harald Eisenmann^{3,†}, Ken Wenzel^{1,†} and Armin Müller^{2,†}

¹Fraunhofer-Institut für Werkzeugmaschinen und Umformtechnik IWU, Digitalisation in Production, Chemnitz, Germany

²ScopeSET Technology Deutschland GmbH, Fischbachau, Germany

³AIRBUS Defence & Space, Friedrichshafen, Germany

Abstract

Spacecraft development involves many tools and stakeholders across different lifecycle phases such as design, manufacturing, testing, operation, and maintenance. These tools generate heterogeneous datasets with different formats and vocabularies, making interoperability and integrated data access difficult. This paper presents a semantic data aggregation framework designed to support lifecycle data integration using Semantic Web technologies. The framework uses RDF, OWL, SHACL, and SPARQL to integrate distributed lifecycle data into a centralized semantic platform. The architecture supports semantic validation, RDF data comparison, and linked data querying while allowing independent evolution of datasets and vocabularies. A prototype implementation demonstrates how semantic technologies can support interoperability and lifecycle data management across heterogeneous spacecraft systems.

Keywords

Semantic interoperability, Semantic Web technologies, RDF, OWL, SPARQL, SHACL,

1. Introduction

A Digital Twin (DT) is a digital representation of a product, resource, or process, including its structure, behaviour, and functions. In the spacecraft domain, DTs support use cases such as simulation, event analysis, predictive maintenance, and digital assistance. Airbus is currently leading a collaborative project with industrial and research partners to realize a Digital Twin Spacecraft (DTSC) ecosystem supporting lifecycle-wide reuse of engineering and operational data. [1, 2, 3]

The project spans system definition, design, manufacturing, assembly and integration, testing, operation, and maintenance. Data are generated by heterogeneous PLM, MBSE, simulation, manufacturing, and telemetry systems that use different schemas, identifiers, formats, and vocabularies. Consequently, lifecycle information is fragmented across isolated systems, limiting interoperability, traceability, and integrated access.

Information about the same technical object also evolves throughout the lifecycle. A subsystem represented in an engineering model may later correspond to a manufactured instance, a tested configuration, and an operational asset producing time-series measurements. Maintaining these links requires persistent semantic identifiers and explicit relationships between independently evolving datasets. At the same time, structural knowledge and high-volume operational measurements require different storage mechanisms but must remain semantically connected.

To address these challenges, ScopeSET and Fraunhofer IWU developed a semantic backend architecture based on Linked Data principles. RDF provides identifiable resources and links between

Posters, Demos, Blue Sky, and Tutorials at SEMANTiCS 2026, Sep 2026, Ghent, Belgium

*Corresponding author.

†These authors contributed equally.

✉ rojina.shrestha@iwu.fraunhofer.de (R. Shrestha); beyreuther@scopeset.de (B. Beyreuther); harald.eisenmann@airbus.com (H. Eisenmann); ken.wenzel@iwu.fraunhofer.de (K. Wenzel); amueller@scopeset.de (A. Müller)

🌐 www.iwu.fraunhofer.de (R. Shrestha); http://www.scopeset.de (B. Beyreuther); https://www.airbus.com/ (H. Eisenmann); www.iwu.fraunhofer.de (K. Wenzel); http://www.scopeset.de (A. Müller)

🆔 0000-0002-6047-6153 (K. Wenzel)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

heterogeneous datasets, OWL describes domain concepts and relationships, SPARQL supports integrated querying, and SHACL enables project-specific validation [4, 5, 6, 7]. RDF Diff further supports configuration evolution by identifying additions, deletions, and modifications between dataset versions.

The paper presents a LinkedFactory-based prototype combining RDF knowledge graphs and time-series data management with services for project, dataset, vocabulary, and configuration management. Because confidential spacecraft data cannot be disclosed, the prototype is demonstrated using a representative manufacturing dataset containing RDF descriptions and associated time-series measurements.

2. System Architecture

The prototype developed provides a semantic backend architecture for integrating heterogeneous spacecraft lifecycle data generated by different engineering and operational systems. Figure 1 presents the overall architecture of the proposed backend. Engineering artifacts originating from different lifecycle phases are transformed into RDF and integrated into a common semantic environment, while operational measurements remain stored in dedicated time-series storage. The backend combines these two data representations through semantic links, enabling unified lifecycle queries, semantic validation, and data management without requiring changes to the original engineering systems.

The architecture is designed to support semantic representation and aggregation of lifecycle knowledge using RDF/OWL while integrating semantic metadata with operational time-series data. In addition, it provides configuration control for datasets and vocabularies, allowing independently evolving engineering information to be managed consistently across different projects and lifecycle phases.

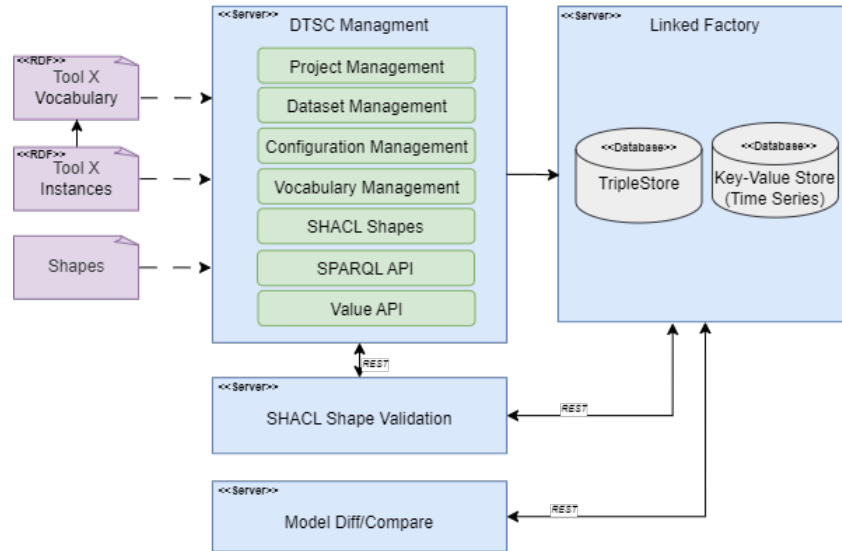


Figure 1: System architecture for data management.

The prototype consists of three main components: LinkedFactory, the Management Server, and Semantic Services.

2.1. LinkedFactory

Conventional RDF triple stores work well to represent semantic relationships, but are less effective in managing large volumes of time-series data. In contrast, time-series databases efficiently manage time-series data but provide limited support for semantic data. Consequently, neither technology alone satisfies the requirements of a digital twin that spans the lifecycle. The proposed architecture therefore combines semantic knowledge graphs with dedicated time-series storage, allowing structural engineering knowledge and operational measurements to be managed using technologies optimized for their respective data characteristics. [8]

Linkedfactory (LF) provides the core storage infrastructure of the proposed backend by combining RDF-based semantic descriptions with time-series management within a unified platform. Engineering components, subsystem relationships, lifecycle metadata, and provenance information are represented as RDF resources, while operational measurements remain stored as efficiently indexed time-series. Semantic identifiers maintain the relationship between both representations, enabling integrated SPARQL-based access to engineering metadata and operational information.[9] [10]

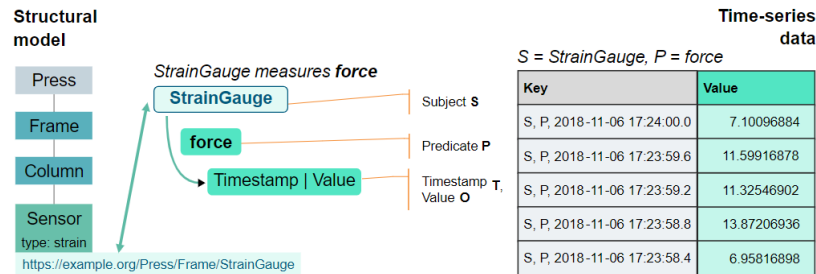


Figure 2: Representation of structural models and measured values in the LinkedFactory data architecture using a force sensor of a press. [11].

Figure 2 illustrates how LF combines structured data with associated time-series data. Subject and predicate are expressed as URIs and reference elements from RDF. They make up the key for time-series entries along with timestamp [8]. This hybrid representation allows semantic context to be associated with operational measurements without materializing all time-series information inside the RDF graph, thereby preserving efficient storage and query performance.

2.2. Management server

The Management Server provides the governance and orchestration layer of the DTSC backend. While LF stores semantic and time-series data, the Management Server manages the lifecycle of semantic assets and provides a higher-level abstraction for Digital Twin applications. It supports project organization, dataset and vocabulary management, configuration control, and secure access to the underlying knowledge graph.

Project Management establishes the organizational boundary of a Digital Twin use case by grouping the associated datasets, vocabularies, and configurations. This facilitates multiple projects to coexist while maintaining independent semantic models, configurations, and access rights.

Engineering and operational tools continuously generate semantic datasets that describe different aspects of a spacecraft. The Dataset Management service registers these datasets together with metadata such as provenance, version, ownership, and originating tool. The semantic content is stored as RDF named graphs in LF, while the Management Server maintains the associated metadata required for governance and traceability.

The Vocabulary Management service maintains the semantic vocabularies associated with each engineering tool or modelling context. Each project maintains one active vocabulary per tool, ensuring consistent semantic interpretation while allowing semantic models to evolve independently across engineering domains.

Configuration Management defines coherent views of the Digital Twin by grouping related dataset versions into engineering baselines. Instead of duplicating semantic data, configurations reference the corresponding RDF named graphs, thus allowing configuration-specific queries while preserving version history and supporting comparison between engineering baselines.

Finally, the Management Server exposes standardized SPARQL and Value APIs that abstract the underlying LF infrastructure. Client applications specify a project and configuration context, while the Management Server resolves the corresponding named graphs and delegates the requests to LF.

2.3. Semantic services

Semantic Services extend the core backend with functionality supporting data quality assurance and lifecycle evolution. They provide semantic validation of incoming datasets as well as comparison of successive dataset versions, ensuring that engineering information remains consistent as Digital Twin models evolve.

SHACL is a constraint-based validation language for RDF graphs. Although Shape Expressions (ShEx) provides similar validation capabilities, SHACL was selected due to its easier integration with RDF stores as well as its faster processing performance [12]. Also, LF internally utilizes the RDF4J SAIL stack, therefore, SHACL Sail could be integrated directly into the backend architecture. Validation is provided on demand against project-specific SHACL constraints, allowing incomplete datasets to be stored during development and validated when required.

RDF Diff service was implemented to detect changes between two datasets. As the focus was on structure-level graph comparison, RDF4J was selected due to its compatibility with the LF architecture and its support for repository-level RDF management through the SAIL stack [13]. The RDF Diff service identifies semantic additions, deletions, and modifications between two RDF datasets. This functionality supports configuration management by highlighting structural changes introduced during model evolution and assists engineers in understanding how Digital Twin representations evolve over time.

3. Prototype Demonstration

Due to confidentiality restrictions on spacecraft engineering data, the proposed backend is demonstrated using a representative manufacturing dataset containing semantic and time-series data. Although it does not represent a complete spacecraft lifecycle, the dataset demonstrates the backend capabilities for integrated querying, on-demand semantic validation, and comparison of evolving datasets.

Listing 1 illustrates integrated querying of semantic and time-series data. Semantic relationships identify machines participating in a printing process and their associated time-series resources. Based on this context, temperature measurements of an extruder are retrieved from LinkedFactory for a particular workpiece and visualized over time, as shown in Figure 3. This demonstrates how semantic relationships provide contextual access to operational measurements.

Figure 4a demonstrates on-demand SHACL validation of a dataset or configuration against project-specific constraints managed by the Management Server. Validation is intentionally decoupled from ingestion, to allow incomplete datasets to be stored and incrementally developed.

Figure 4b demonstrates RDF Diff between two dataset versions belonging to different configurations. The service identifies additions, deletions, and modifications at the subject level, which supports traceability of structural changes as Digital Twin representations evolve.

Listing 1: Example SPARQL query retrieving extruder temperature measurements

```
SELECT ?time ?Temperature WHERE {
  BIND( IRI("tag:cps.iwu.fraunhofer.de,2024:parts/24002998") AS ?identification )
  {
    SELECT DISTINCT ?item WHERE {
      ?machine rdf:type vocab:Machine .
      ?machine vocab:participatesIn inst:PrintingProcess .
      ?machine vocab:hasTimeSeriesResource ?item .
    }
  }
  SERVICE <kvin:> {
    ?item p:start [ <kvin:time> ?start ; <kvin:value> [ p:workpiece ?identification ] ] ;
    p:stop [ <kvin:time> ?stop ; <kvin:value> [ p:workpiece ?identification ] ] .
  }
  ...
}
```

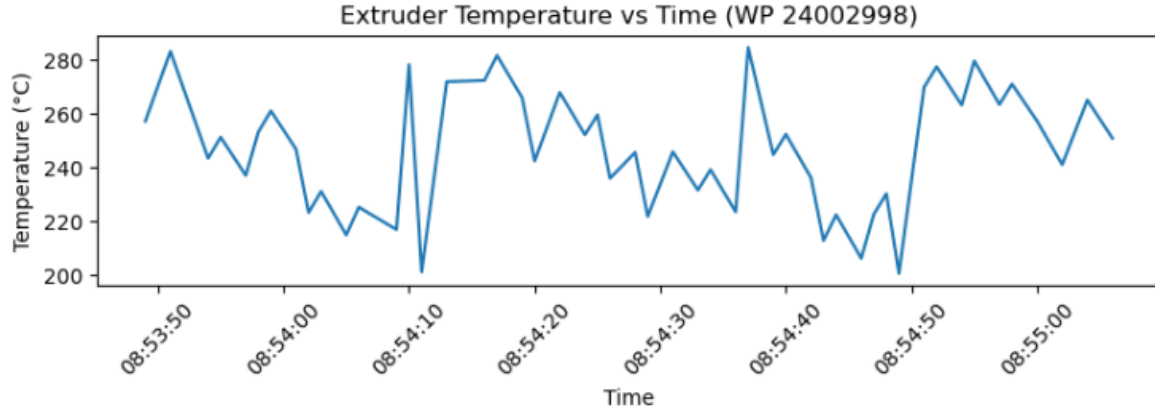
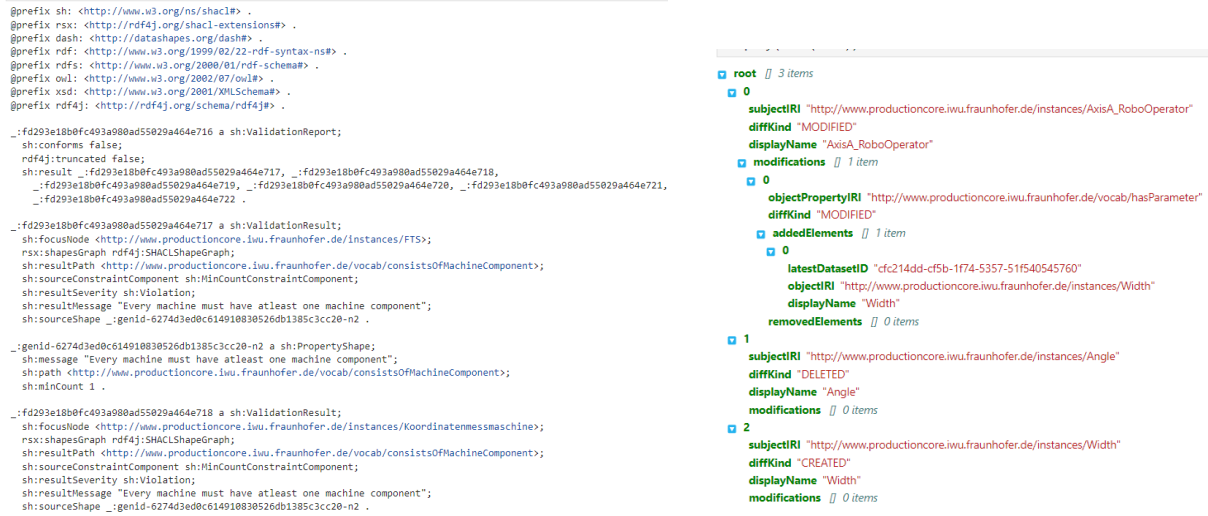


Figure 3: Semantic query retrieving contextualized time-series measurements through the proposed backend.



(a) On-demand SHACL validation of a semantic dataset against project-specific constraints. (b) Comparison of two semantic dataset versions using the RDF Diff service.

Figure 4: Prototype semantic services supporting validation and lifecycle evolution.

4. Conclusion

This paper presented a semantic backend architecture for integrating heterogeneous data across the spacecraft lifecycle within the Digital Twin Spacecraft (DTSC) project. The proposed architecture combines RDF-based knowledge graphs with integrated time-series data management through Linked-Factory, providing a unified semantic foundation for managing engineering and operational information. In addition, the backend introduces management services for projects, datasets, vocabularies, and configurations together with semantic validation using SHACL and graph comparison using RDF Diff to support lifecycle data governance and evolution.

A prototype implementation was demonstrated using a representative manufacturing dataset, illustrating the integration of semantic metadata with associated time-series measurements, semantic validation, and comparison of evolving dataset versions. Although spacecraft datasets could not be used due to confidentiality restrictions, the demonstration validates the feasibility of the proposed architecture and its ability to support semantic integration, querying, and lifecycle-aware data management.

Future work will focus on validating the architecture using additional engineering use cases from the DTSC project, extending semantic services for larger and more complex datasets, and evaluating the scalability and performance of the proposed backend in realistic Digital Twin environments.

Declaration on Generative AI

During the preparation of this work, the authors used GPT-4.5 ChatGPT in order to: Paraphrase. After using these tool, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

- [1] M. Gill, M. Weigand, T. Jeleniewski, J. Reif, A. Fay, Application potentials of Semantic Technologies for Digital Twins in Aircraft Design, Manufacturing and Maintenance, 2022, pp. 266–271. doi:10.24405/14562.
- [2] B. Vogel-Heuser, F. Ocker, I. Wei, R. Mieth, F. Mann, Potential for combining semantics and data analysis in the context of digital twins, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 379 (2021) 20200368. doi:10.1098/rsta.2020.0368.
- [3] W. Kritzinger, M. Karner, G. Traar, J. Henjes, W. Sihn, Digital twin in manufacturing: A categorical literature review and classification, in: IFAC-PapersOnLine, volume 51, 2018. doi:10.1016/j.ifacol.2018.08.474.
- [4] R. Cyganiak, D. Wood, M. Lanthaler, RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation, World Wide Web Consortium (W3C), 2014. URL: <https://www.w3.org/TR/rdf11-concepts/>.
- [5] W3C OWL Working Group, OWL 2 Web Ontology Language: Document Overview, W3C Recommendation, World Wide Web Consortium (W3C), 2012. URL: <https://www.w3.org/TR/owl2-overview/>.
- [6] S. Harris, A. Seaborne, SPARQL 1.1 Query Language, W3C Recommendation, World Wide Web Consortium (W3C), 2013. URL: <https://www.w3.org/TR/sparql11-query/>.
- [7] H. Knublauch, D. Kontokostas, Shapes Constraint Language (SHACL), W3C Recommendation, World Wide Web Consortium (W3C), 2017. URL: <https://www.w3.org/TR/shacl/>.
- [8] M. Friedemann, K. Wenzel, A. Singer, Linked data architecture for assistance and traceability in smart manufacturing, MATEC Web of Conferences 304 (2019). doi:10.1051/mateconf/201930404006.
- [9] F. Rudolph, K. Wenzel, Von datenflut zur datenklarheit: Wie die linkedfactory-datenarchitektur die nutzung von produktionsdaten für digitale zwillinge ermöglicht, 2025. URL: <https://www.kognitive-produktion.de/von-datenflut-zur-datenklarheit-wie-die-linkedfactory-datenarchitektur-die-nutzung-von-produktionsdaten-fuer-digitale-zwillinge-ermoeslicht/>.
- [10] J. M. Grützmann, F. Rudolph, M. Boesler, K. Wenzel, Flexible data architecture for enabling ai applications in production environments, HTWK Leipzig, 2024. doi:10.33968/2024.77.
- [11] LinkedFactory, Linkedfactory pod repository, <https://github.com/linkedfactory/linkedfactory-pod>, 2026. GitHub repository, accessed 13 June 2026.
- [12] D. Tomaszuk, Rdf validation: A brief survey, in: Communications in Computer and Information Science, volume 716, 2017. doi:10.1007/978-3-319-58274-0_28.
- [13] Eclipse Foundation, Eclipse rdf4j, 2026. URL: <https://rdf4j.org/>.