

Knowledge-Based Aerospace Engineering - A Systematic Literature Review

Tim Wittenborg* and Ildar Baimuratov†

L3S Research Center, Hannover, Lower Saxony, 30167, Germany

Cluster of Excellence SE²A – Sustainable and Energy-Efficient Aviation, Technische Universität Braunschweig, Germany

Ludvig Knöös Franzén‡

Linköping University, Linköping, Östergötland, 58183, Sweden

Ingo Staack§ and Ulrich Römer¶

Technische Universität Braunschweig, Braunschweig, Lower Saxony, 38106, Germany

Cluster of Excellence SE²A – Sustainable and Energy-Efficient Aviation, Technische Universität Braunschweig, Germany

Sören Auer||

TIB – Leibniz Information Centre for Science and Technology and L3S Research Center, Hannover, Lower Saxony, 30167, Germany

Cluster of Excellence SE²A – Sustainable and Energy-Efficient Aviation, Technische Universität Braunschweig, Germany

Acronyms

3D three-dimensional.

CSV Comma-separated values.

AI Artificial Intelligence.

DL Description Logic.

AIAA American Institute of Aeronautics and Astronautics.

DLR Deutsches Zentrum für Luft- und Raumfahrt.

ANFIS Adaptive Neuro-Fuzzy Inference System.

DTAD Digital Thread for Airplane Design.

ATMonto Air Traffic Management Ontology.

FAIR Findable, Accessible, Interoperable, Reusable.

BFO Basic Formal Ontology.

JSON JavaScript Object Notation.

CAD Computer-Aided Design.

JSON-LD JSON for Linked Data.

CATIA Computer Aided Three Dimensional Interactive Applica-

KADMOS Knowledge- and graph-based Agile Design with

tion.

multi-disciplinary optimization System.

CLIF Common Logic Interchange Format.

KBE Knowledge-Based Engineering.

CMDOWS Common multi-disciplinary optimization workflow

KIF Knowledge Interchange Format.

schema.

KM knowledge management.

CPACS Common Parametric Aircraft Configuration Schema.

MBSE model-based systems engineering.

*PhD Student, L3S Research Center, Appelstr. 9a, 30167 Hannover, tim.wittenborg@l3s.de

†Postdoc, L3S Research Center, Appelstr. 9a, 30167 Hannover, ildar.baimuratov@l3s.de

‡Assistant Professor, Division of Fluid- and Mechatronic Systems, SE-581 83 Linköping, Sweden, ludvig.knoos.franzen@liu.se

§Professor, Overall Aircraft Design, Hermann-Blenk-Str. 35, ingo.staack@tu-braunschweig.de

¶Professor, Modeling of Complex Systems under Uncertainty, Langer Kamp 19, u.roemer@tu-braunschweig.de

||Professor and Director, Data Science and Digital Libraries, Welfengarten 1B, 30167 Hannover, auer@tib.eu

MDAO multi-disciplinary analysis and optimization.	SLR Systematic Literature Review.
MDAx MDAO Workflow Design Accelerator.	SPARQL SPARQL Protocol And Resource Description Framework Query Language.
MDO multi-disciplinary optimization.	STEP Standard for the Exchange of Product model data.
NASA National Aeronautics and Space Administration.	SWARM-SLR Streamlined Workflow Automation for Machine-actionable Systematic Literature Reviews.
OMG Object Management Group.	SWRL Semantic Web Rule Language.
ORKG Open Research Knowledge Graph.	SysML Systems Modeling Language.
OWL Web Ontology Language.	TU Technische Universität.
PIDO Process Integration and Design Optimization.	UML Unified Modeling Language.
PLM Product Lifecycle Management.	URI Uniform Resource Identifier.
RaDEX Rationale-based Design Explanation.	W3C World Wide Web Consortium.
RCE remote component environment.	XML Extensible Markup Language.
RDF Resource Description Framework.	
RDFa RDF in Attributes.	
SE²A Sustainable and Energy-Efficient Aviation.	

I. Introduction

The aerospace industry operates at the frontier of technological innovation, combining complex systems and cutting-edge research to achieve advancements in design, manufacturing, and operational efficiency. With the current trend towards new system architectures, including novel technologies, the complexity of aircraft design and its verification and validation is significantly growing and can no longer be handled with traditional engineering approaches. At the same time, the aerospace industry is known for high standards regarding safety and reliability, leading to rather incremental design evolution, where radical new configurations and products are rare. In this environment, with an enormous potential for re-use and adaptation of existing solutions and methods, it is not surprising that Knowledge-Based Engineering (KBE) [1] has been applied for decades.

KBE is an engineering methodology that focuses on capturing, structuring, and automating engineering knowledge to support decision-making and improve efficiency. Nowadays widely used within Computer-Aided Design (CAD) and design automation, traditional rule-based KBE can be applied to any engineering domain in which data or information may be reused in a narrow or broader context. Examples range from structural design information reuse [2], requirement engineering [3], and system component and architecture design [4, 5]. An overview of KBE methods and tools is given in [6, 7]. [Figure 1](#) highlights the increasing spread throughout the aerospace domain in recent years with a trend to extend KBE towards knowledge capture and management not solely focusing on explicitly stated information but making use of automatically extracted metadata from various sources, see e.g., [8]. While conventional KBE is well-suited for traditional evolutionary design engineering, the conventional KBE methods have shortcomings with radically new design challenges, that are imposed by the need towards sustainable aviation [6, 9]. Here, the potential of KBE with

regards to flexible data reuse and information needs to be unlocked. These needs imply findability, accessibility, interoperability and reusability (FAIR) [10] of aerospace engineering knowledge. This knowledge management (KM) process, focusing on extraction, creation and utilization of machine-readable knowledge representation, is referred to as knowledge engineering.

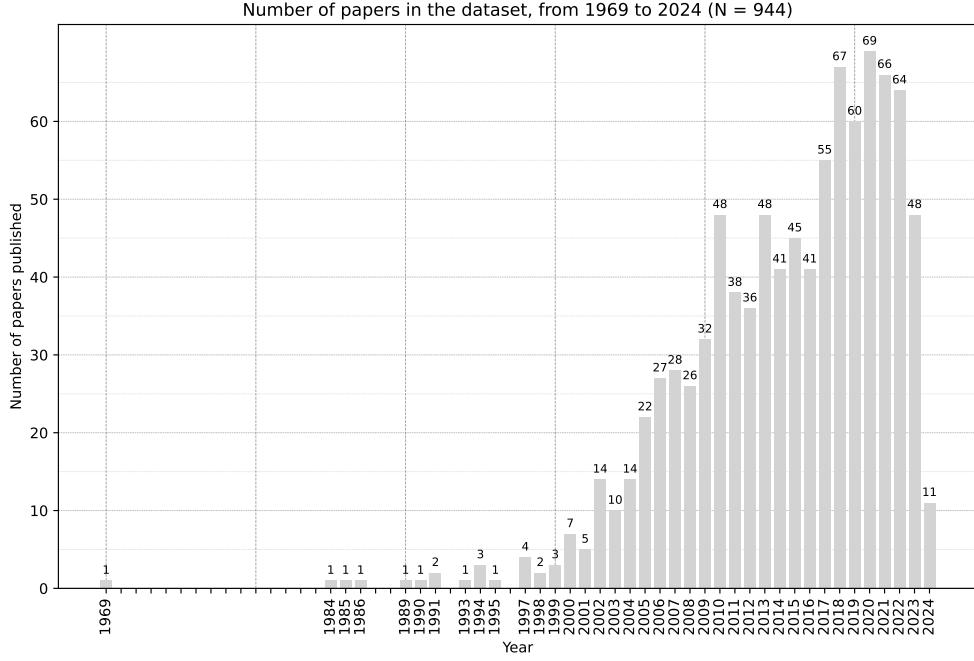


Fig. 1 The 944 unique available papers to our knowledge-based aerospace engineering literature review, plotted over their publication year. A rise in relevant papers in recent years is noticeable.

The objective of this study is to identify and examine state-of-the-art knowledge management practices in the field of aerospace engineering. As outlined in section IV, we performed a preliminary literature review in advance to refine the objective and methodology, leading to the formulation of three research questions:

- **RQ1:** By which means, such as knowledge representations and schemas, do aerospace engineers externalize, such as formalize and visualize, knowledge?
- **RQ2:** By which means, such as knowledge bases and services, do aerospace engineers utilize, such as organize and interface with, explicit knowledge?
- **RQ3:** By which means, such as tools and methods, do aerospace engineers exchange, such as transfer and distribute explicit knowledge?

Answering these questions, we aim to provide a recommendation framework for the further adoption and improvement of knowledge management practices.

In order to achieve this objective, the systematic literature review methodology Streamlined Workflow Automation for Machine-actionable Systematic Literature Reviews (SWARM-SLR) [11] was used. This approach offers guidance and state-of-the-art tool support for conducting systematic literature reviews, while simultaneously producing FAIR artifacts. With the help of this streamlined workflow, we proceeded to survey over 1,000 articles aiming to answer our research questions. A thorough manual review of 245 computationally recommended articles resulted in 164 articles included for dedicated analysis. With this approach, we collected over 700 knowledge-based aerospace-engineering-related items and classified them into several categories, such as processes, tools, and data, which are discussed in the sections **Process Findings**, **Software Findings** and **Data Findings** respectively. Our findings are discussed in **section IX**, and the work is concluded in **section X** with a final summary.

Our contributions include:

- 1) A SWARM-SLR of over 1,000 articles with qualitative analysis of 164 selected articles, supported by two aerospace engineering domain expert surveys.
- 2) A knowledge graph of over 700 knowledge-based aerospace engineering processes, software, and data, formalized in the interoperable Web Ontology Language (OWL) and mapped to Wikidata* entries where possible. The knowledge graph is represented on the Open Research Knowledge Graph (ORKG)[†], and an aerospace Wikibase[‡], for reuse and continuation of structuring aerospace engineering knowledge exchange.
- 3) Our resulting intermediate and final artifacts of the knowledge synthesis, available as a Zenodo dataset[§].

The findings of this review not only provide a detailed understanding of current practices but also pave the way for developing interoperable knowledge-sharing infrastructures. These contributions are anticipated to foster collaborative innovation, enhance operational efficiency, and uphold the aerospace industry's commitment to excellence and sustainability.

The description of this survey and development process is structured as follows: Initially, the **section II** and **section III** will provide the required context for this work. Following this, **section IV** will cover all information related to the survey design. Consecutively, **section V** covers the general results, while three dedicated sections discuss the findings for each partition of the survey: **section VI** describes the processes relevant to either aerospace or knowledge engineers, **section VII** describes the tools used to solve these tasks, and **section VIII** concludes on which data items, formats, and models are used and produced by these activities. How these findings could be further developed will be discussed in **section IX**. Finally, the **section X** concludes this article.

*<https://www.wikidata.org/>

[†]https://incubating.orkg.org/observatory/Knowledge_Based_Aerospace_Engineering

[‡]<https://aerospace.wikibase.cloud>

[§]https://zenodo.org/records/14790367?preview=1&token=eyJhbGciOiJIUzUxMiJ9.eyJpZCI6ImQzMzTczMwY1LWI2NmMtNDhiNS04Y2NjLTMwOGI1Mzdmg5InBiNzr8pSu-YT0FzcGuMc76_F-PHW2jlFn-niYjbFRxTZBwUfkLEnAYyRWze0SNy98cfA8yYRZk5cA (Note to the reviewer: this is a preview accessible via token, for camera-ready we will provide a public url)

II. Related Work

Systematic literature reviews, which offer an unbiased and comprehensive overview of existing research, are rare in the field of knowledge-based aerospace engineering. Tallying the literature reviews and literature surveys present in the American Institute of Aeronautics and Astronautics (AIAA) search index, a new literature review is published only once every five years, starting from 1952, as illustrated in [Figure 2](#). Using Google Scholar, OpenAlex, and AIAA, we identified a total of 34 relevant literature reviews, which are curated in the Zotero library[¶]. However, most of these reviews focus on peripheral topics, lacking either an aerospace domain focus, a knowledge focus, or both, as depicted in [Table 1](#).

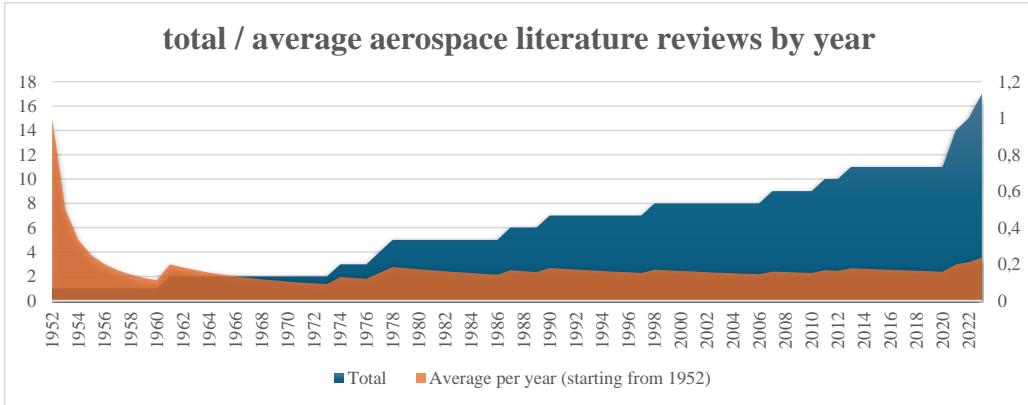


Fig. 2 Development of AIAA literature reviews over time. An increase since 2021 is noticeable.

Table 1 Related literature reviews by topics, in ascending similarity to our scope

Source	Topic
[12]	model management
[13] [14]	multi-disciplinary optimization (MDO)
[15] [16] [17] [18] [19] [20]	digital twins
[21]	application of Artificial Intelligence (AI) in aerospace engineering
[22]	knowledge maps
[23]	sensor ontologies in manufacturing
[24] [25]	Product Lifecycle Management (PLM), [25] particularly on ontologies in PLM
[26]	model-based systems engineering (MBSE)
[6]	Knowledge-Based Engineering (KBE)
[27]	platform health management in aerospace
[28]	knowledge representations in concept development
[29] [30] [31]	knowledge management, [30] specifically using linked data in the aerospace industry
[32]	knowledge acquisition, exchange and collaboration in aerospace software development

[¶]<https://www.zotero.org/groups/5420201/sea-b-aerospace-knowledge-swarm-slr/collections/3J2G3DAG>

Only three of these reviews focus on knowledge-based engineering. Hassan et al. [21] examine the application of AI in aerospace engineering. Oprea et al. [32] discuss knowledge acquisition, exchange standards, and multidisciplinary collaboration in software development. The most relevant work to our focus on knowledge-based engineering is Procko and Ochoa [30]: “*Leveraging Linked Data for Knowledge Management: A Proposal for the Aerospace Industry*”. To our knowledge, none of these reviews apply FAIR principles for their resulting artifacts, which are primarily single PDFs, nor do they incorporate a software-assistant methodology such as SWARM-SLR. In contrast, our systematic literature review provides digital artifacts, including knowledge graphs and software, that allow researchers to access, reuse, and extend our findings.

Preliminary Work To complement the related work, it is worth mentioning that this Systematic Literature Review (SLR) succeeds preliminary work. Within the German Cluster of Excellence “Sustainable and Energy-Efficient Aviation (SE²A)” the project “B4.2 - Consistent Multilevel Model Coupling and Knowledge Representation in Multidisciplinary Analysis and Design”^{||} conducted a preliminary literature review. The aim was to answer questions such as “How do aerospace engineers formalize and exchange domain knowledge?”, with a strong focus on knowledge representations in multidisciplinary design with high-fidelity or multi-fidelity modeling. This review was conducted throughout 2023 and informed the early stages of the B4.2 project while simultaneously laying the groundwork for the SWARM-SLR [11]. Equipped with this enhanced workflow, a better understanding of the domain specific literature, a set of 1,102 documents, and early results, the preliminary literature review was concluded in late 2023 to make way for a fresh, systematically structured start in early 2024. This article represents this succeeding Systematic Literature Review.

III. Background

This work discusses aerospace KBE with a focus on knowledge engineering data and knowledge representation. To introduce these topics briefly, we provide context on [Knowledge-Based Aerospace Engineering](#) and [Knowledge Engineering](#) to finally provide a clearer grasp on [Semantic Web in Aerospace Engineering](#).

A. Knowledge-Based Aerospace Engineering

Knowledge-based engineering leverages expert knowledge, rules, and computational tools to enhance engineering processes and improve their outcomes. In the aerospace industry, the development and management of complex systems require structured methodologies and robust data frameworks to ensure efficiency, traceability, and compliance. Central to these methodologies is the concept of a process, which defines the sequence of activities required to design, develop, and maintain such systems.

^{||}<https://www.tu-braunschweig.de/se2a/research/projects/area-b/b42-consistent-multilevel-model-coupling-and-knowledge-representation-in-multidisciplinary-analysis-and-design>

1. The Digital Thread

Product Lifecycle Management (PLM) is a structured approach for managing the entire lifecycle of a product from its initial concept through design, production, operation, and ultimately its retirement phase [33]. It emphasizes the integration of data, processes, and stakeholders across all lifecycle stages to improve efficiency and informed decision-making. A key enabler of this connectivity is the digital thread. The digital thread represents a continuous flow of data and information across a system's lifecycle [34]. It thereby serves as a digital representation of the system, thus enhancing aspects such as traceability, risk management, and decision-making support. These improvements are enabled through the comprehensive overview that the digital thread offers from the first design stage to the system's eventual retirement and all stages there in between. Together, a digital thread and knowledge-based engineering enable data-driven decision-making capabilities that facilitate information exchange, efficiency, and potential for innovation. This also complements the procedural model known as the “engineering V-model” by strengthening its emphasis on continuous verification and validation throughout a system's lifecycle and providing a digital representation of it.

2. The V-Model

The V-model, or engineering V-model, is a widely accepted systems engineering framework used to guide the development of complex systems, for example within aerospace applications [35, 36]. The V-model emphasizes rigorous validation and verification processes throughout the product's lifecycle. It is named for its characteristic “V” shape, where the left-hand side represents system definition and design phases, and the right-hand side represents integration and testing phases, ultimately ending in system deployment and maintenance. Figure 3 shows an illustration of the V-model structure with its corresponding phases.

The emphasis on structured validation and verification processes from the V-model aligns well with the principles of knowledge-based engineering. However, the increasing complexity and scale of aerospace systems require data management-, integration-, and analysis tools that go beyond traditional frameworks, and this is where Semantic Web technologies come into the picture.

B. Knowledge Engineering

With the exponential growth of data, the need for computational tools to manage, interpret, and utilize such information has become a topic of increasing importance. In engineering domains, especially aerospace, the complexity of systems and processes necessitates structured methodologies that enable efficient data handling and informed decision-making capabilities. The FAIR [10] principles emphasize the capacity of computational systems to **Find**, **Access**, **Interoperate**, and **Reuse** data with minimal human intervention. Another initiative is Linked Data **, which aims to interlink data in the Web so it becomes possible to find them with semantic queries. To do that, Uniform Resource

**<https://www.w3.org/DesignIssues/LinkedData.html>

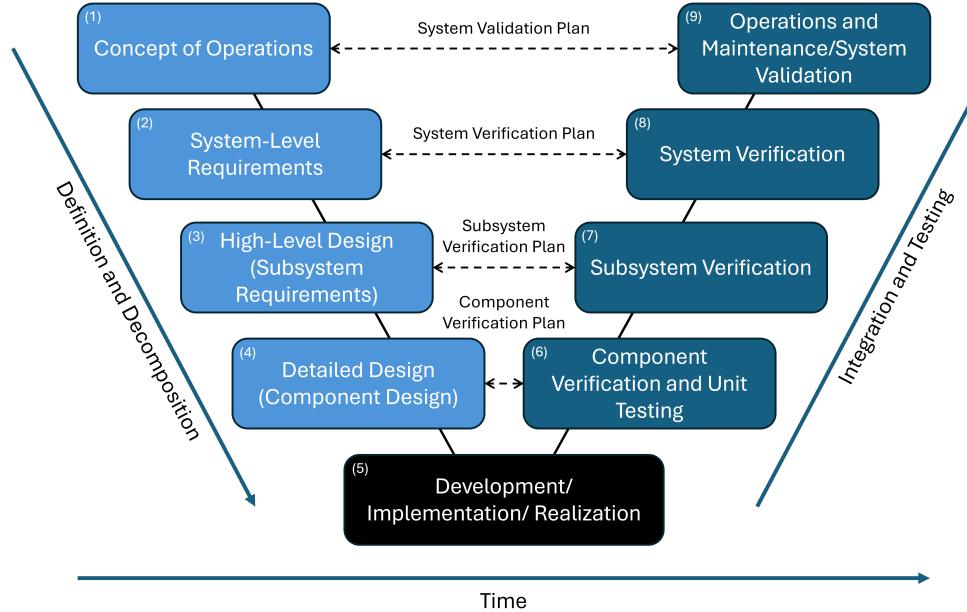


Fig. 3 An illustration of the engineering V-model and its corresponding phases. Adapted from [36].

Identifiers (URIs) are used to uniquely identify resources in the Web. An example of the Linked Data implementation are DBpedia [37] – a project aiming to extract structured content from the information created in the Wikipedia project, and Wikidata [38] – a collaboratively edited multilingual knowledge graph hosted by the Wikimedia Foundation. Finally, Google’s *Knowledge Graph* ^{††} became a common term for graph knowledge bases.

One of the technologies supporting FAIR principles and Linked Data is the Semantic Web [39], which embeds machine-readable semantics into data through standards such as the Resource Description Framework (RDF) and the Web Ontology Language (OWL).

RDF is a framework for expressing information about resources. Resources can be anything, including documents, people, physical objects, and abstract concepts. RDF is intended for situations in which information on the Web needs to be processed by applications, rather than being only displayed to people. It also provides a common framework for exchanging information between applications without loss of meaning. RDF allows us to make statements about resources. A statement always has the following structure: $< \text{subject} > < \text{predicate} > < \text{object} >$. The subject and the object represent the two resources being related and the predicate represents their relationship. As RDF statements consist of three elements, they are called triples. The ability to have the same resource in the subject position of one triple and in the object position of another one makes it possible to find connections between triples and build connected graphs. These graphs can be queried with the SPARQL Protocol And RDF Query Language (SPARQL). Figure 4 shows an example of triples that together form a small knowledge graph.

^{††}<https://blog.google/products/search/introducing-knowledge-graph-things-not/>

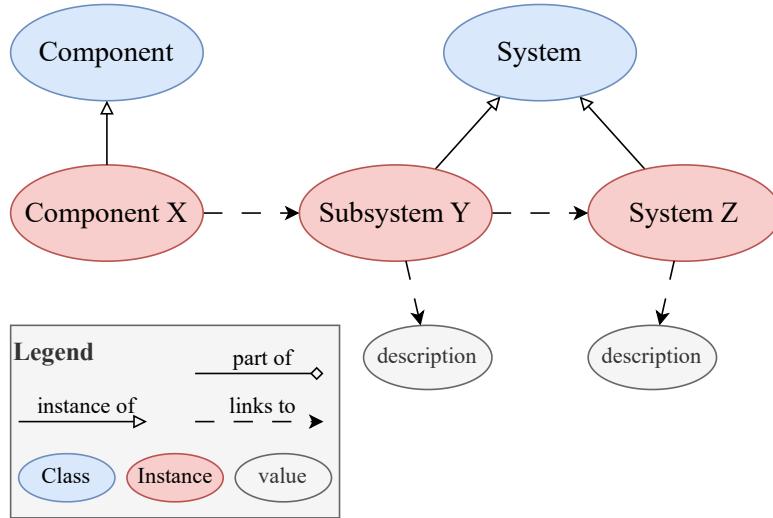


Fig. 4 Simple knowledge graph example, derived from [40]

OWL is a vocabulary extension of RDF that facilitates greater machine interpretability of web content. OWL can be used to explicitly describe semantics of data, resulting in an ontology. Ontologies define *Classes*, which unify certain *Instances* under a common definition of shared attributes. Beside class membership relations, instances can be inter-connected via *ObjectProperties*. Additionally, instances can have connections with non-object data such as integers, strings or similar via *DataProperties*. To increase the human-readability, additional *AnnotationProperties* can further specify information that is not structurally significant, such as a label or description. Given such an ontology, reasoning engines can derive logical consequences from data, i.e. facts not literally present in the knowledge base, but entailed by the semantics. Figure 5 shows an illustration of a small ontology within the aerospace domain.

Semantic Web technologies inherently promote findability, accessibility, interoperability, and reusability (FAIR). They ensure that data is machine-readable, consistently structured, and easily discoverable by researchers and engineers. These technologies facilitate the automation of repetitive design and analysis tasks through rule-based systems and AI-driven tools, streamlining workflows and fostering innovation. In addition, as the Semantic Web is inherently distributed, ontologies must allow for information to be gathered from distributed sources. This is partly done by making the open-world assumption. This assumption essentially means that just because something is not in the knowledge base, it does not mean that it is false; new information can always be added. A number of different serialization formats exist for writing down RDF graphs and ontologies: N-Triples, Turtle, JSON for Linked Data (JSON-LD), RDF in Attributes (RDFa), RDF/XML, etc. For a more detailed introduction of RDF and OWL languages and their serialization we invite readers to read the RDF Primer^{††} and the OWL Language Overview from World Wide Web Consortium (W3C)^{§§}.

^{††}<https://www.w3.org/TR/rdf11-primer/>
^{§§}<https://www.w3.org/TR/owl2-overview/>

C. Semantic Web in Aerospace Engineering

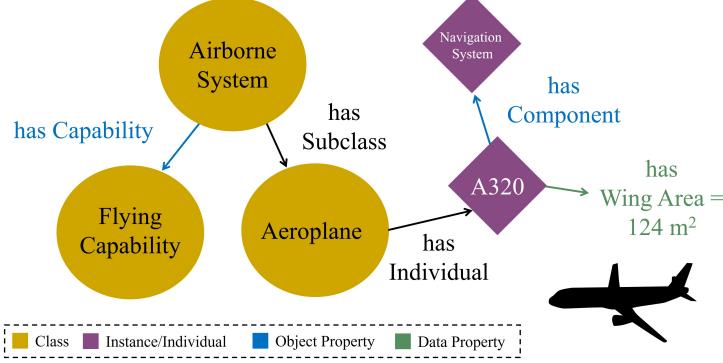


Fig. 5 An aerospace example that illustrates some of the key aspects of an ontology.

Semantic Web technologies provide structured frameworks and tools that significantly enhance KBE in the aerospace domain by improving the organization, accessibility, and automation of knowledge:

- **Data Interoperability:** RDF and OWL enable the creation of ontologies that standardize how aerospace engineering knowledge is represented, thus ensuring that diverse systems can seamlessly exchange and interpret data.
- **Knowledge Representation:** Ontologies define relationships between concepts in aerospace engineering, such as processes, tools, and data formats. This structured representation allows for detailed modeling of complex engineering workflows, improving the organization and retrieval of relevant knowledge.
- **Enhanced Querying and Reasoning:** Using query languages like SPARQL, Semantic Web technologies allow engineers to retrieve specific knowledge from large datasets efficiently. Reasoning engines can infer new knowledge from existing data, aiding decision-making and problem-solving.
- **Linked Data:** By linking aerospace engineering data with external resources (e.g., Wikidata or DBpedia), Semantic Web approaches provide richer context and enable cross-domain knowledge integration. This interconnected data can support multidisciplinary collaborations crucial in aerospace projects.

This capability is particularly relevant in knowledge-based engineering, where domain-specific knowledge, automation, and optimization drive the design and lifecycle management of complex systems. Both RDF and OWL have been used in different works related to aerospace. An example of this is [41] where ontologies with Description Logic (DL) reasoning capabilities have been used to represent and process design spaces for an aerospace system of systems at an early design stage. Other examples reach from Dadzie et al. [42] investigating the potential of using RDF and OWL for knowledge sharing in aerospace engineering to Rovetto [43] to reliably using these for space data KM while extending towards Knowledge Interchange Format (KIF) and Common Logic Interchange Format (CLIF). Our systematic literature review primarily focuses on such studies that apply Semantic Web technologies in knowledge-based aerospace engineering.

IV. Review Methodology: A SWARM-SLR Approach

Conducting a Systematic Literature Review (SLR) for a domain as complex and multidisciplinary as aerospace engineering requires a robust and efficient methodology. This review was conducted according to the Streamlined Workflow Automation for Machine-actionable SLRs (SWARM-SLR) [11]. This methodology is particularly well-suited for knowledge-based aerospace engineering, as it addresses the challenges of managing diverse sources, ensuring comprehensive coverage, and maintaining consistency in data extraction and synthesis. The SWARM-SLR approach is designed to streamline the review process through its four distinct stages: Planning, Searching and Selecting, Analyzing and Synthesizing, and Reporting, as depicted in [Table 2](#). These stages incorporate iterative refinement, enabling researchers to adapt the methodology to the evolving objectives and findings of the review.

Table 2 Eight tasks conducted over the four stages of this SWARM-SLR.

Stage I	Task 1: Planning a Review Task 2: Defining a Scope
Stage II	Task 3: Search Task 4: Select Task 5: Evaluate
Stage III	Task 6: Analysis Task 7: Synthesis
Stage IV	Task 8: Reporting

According to the SWARM-SLR, each task has certain steps (*and results*). This section will go over each task and step to understand the approach, while the achieved results are generally discussed in the consecutive section.

A. Task 1: Planning a Review

Initiating a literature review generally begins with an imprecise inquiry:

- 1) Formulate research interest (*explicit informal research interest*)

As already described, the preliminary SLR significantly informed this first task and enabled a very clear outline in advance. The abstract interests, such as “Which services, software or algorithms that interface, utilize or visualize knowledge bases do aerospace engineers use/suggest?”, were already very close to the final research questions.

B. Task 2: Defining a Scope

As the final task of Stage I, this task concludes the definition of what this particular systematic literature aims to achieve:

- 1) Check for related research questions (*narrowed informal research interest*)
- 2) Refine scientific interest (*specific research question*)
- 3) Formulate Search Query (*preliminary weighted keywords and queries*)

- 4) Refine with related literature (*weighted keywords and refined queries*)
- 5) Reevaluate with domain experts (*validated weighted keywords and queries*)

Related research questions were continuously checked for during the preliminary work. With that information, the scientific interest was eventually refined to three final research questions:

- **RQ1:** By which means, such as *knowledge representations* and *schemas*, do aerospace engineers *externalize*, such as *formalize* and *visualize*, knowledge?
- **RQ2:** By which means, such as *knowledge bases* and *services*, do aerospace engineers *utilize*, such as *organize* and *interface with*, explicit knowledge?
- **RQ3:** By which means, such as *tools* and *methods*, do aerospace engineers *exchange*, such as *transfer* and *distribute*, explicit knowledge?

These three research questions provide structured access to the knowledge-based engineering means and activities, selected as representatives of a larger relevant set that remains relevant as additional keywords. Each of these questions was accompanied by an extensive list of weighted keywords and a search query. For example, research question one was associated with the keywords depicted in [Listing 1](#) and the search query shown in [Listing 2](#).

Listing 1 SWARM-SLR readable representation of RQ1: By which means, such as knowledge representations and schemas, do aerospace engineers externalize, such as formalize and visualize, knowledge?

```
### aerospace engineering           Knowledge graph:: 10          graph:: 2
aerospace engineer:: 10            Ontology:: 10             plot:: 2
aerospace engineering:: 10         information representation:: 8 mapping:: 2
airspace engineer:: 10            Taxonomy:: 8              wiki:: 2
airspace engineering:: 10          Semantic Web:: 8          technique:: 2
airplane engineer:: 10            knowledge engineering:: 7 classification:: 2
airplane engineering:: 10          schema:: 7              categorization:: 2
airnautic engineer:: 10           expert system:: 6          data:: 1
airnautic engineering:: 10         knowledge:: 5             map:: 1
aircraft engineer:: 10            data representation:: 4 computer-aided design:: 1
aircraft engineering:: 10          semantic:: 4             CAD:: 1
spacecraft engineer:: 10          information:: 3           computer-aided design:: 1
spacecraft engineering:: 10        repository:: 3           virtual reality:: 1
avionics engineer:: 10            documentation:: 3          VR:: 1
avionics engineering:: 10          graphic:: 3              simulation:: 1
aviation engineer:: 10            index:: 3               platform:: 1
aviation engineering:: 10          formal:: 2              report:: 1
                                  standard:: 2
### knowledge representation       format:: 2               ### externalize
knowledge representation:: 10      structure:: 2           externalize:: 10
```

represent:: 10	model:: 2	XML:: 4
formalize:: 8	document:: 1	CPACS:: 4
visualize:: 8	codify:: 1	SysML:: 4
store:: 4	transfer:: 2	UML:: 4
structured:: 4	explicit knowledge:: 8	JSON-LD:: 3
articulate:: 3		JSON:: 2
express:: 3	### Formats	YAML:: 2
manifest:: 3	RDF:: 4	GraphML:: 1
embody:: 2	OWL:: 4	XSD:: 4

Listing 2 RQ1 Query with 929 Google Scholar results

```
"aerospace engineering" OR "aviation engineering" "Knowledge Representation" OR schema OR  
"Knowledge graph" externalize OR represent OR formalize OR visualize RDF OR OWL OR XML
```

Each research question was associated with one such search query, based on its weighted keyword list. The keyword lists, without the `### aerospace engineering` block which are equal among all research questions, are displayed in the appendix in [Listing 3](#) and [Listing 5](#), alongside their respective queries in [Listing 4](#) and [Listing 6](#).

Concluding this step, the weighted keywords and queries were evaluated by domain experts and deemed effective. The search queries remained unchanged, but the keyword list and weights were continuously refined throughout the literature review to improve the recommendation further.

C. Task 3: Search

Searching is a two-part, straight-forward task:

- 1) Find resources (*preliminary document set*)
- 2) Remove duplicates and find missing documents (*curated document set*)

Following the SWARM-SLR workflow, we used the three designed search queries on Google Scholar with the Zotero Plugin, resulting in 916, 509 and 30 results, providing gradually less results as Google Scholar limited the access over Zotero. Exporting the list from Google Scholar directly somewhat compensated this limitation, but the provided overviews were not qualitatively or even quantitatively identical to the list available through the frontend, resulting in 892, 732 and 210 entries, respectively. Combined with the 1102 entries from the preliminary work, this formed our document set of 4073. After removing duplicates and finding available PDFs, 1029 documents remained for further processing.

D. Task 4: Select

The selection task aims to identify relevant literature among the search results, requiring several steps to get there:

- 1) Extract structured data from documents (*metadata; content; bag of words; contribution statements*)
- 2) Calculate relational measurements within the document set (*document representation; similarity of documents within the document set*)
- 3) Identify documents relevant for the research questions (*relevance of documents for research question*)

The SWARM-SLR algorithm processed the documents and scored them according to the weighted keywords from [Task 2: Defining a Scope](#).

While each RQ had its keyword set weighting in the aerospace aspect, it was made apparent that documents that sufficiently scored in other keywords could be classified as relevant without being related to the aerospace domain. To address this, the aerospace aspect itself became an artificial fourth “research question” consisting of only the `### aerospace engineering` block as shown in [Listing 1](#). This score was used as a visual and algorithmic indicator to discourage investing time in non-aerospace literature. Since this task is where Systematic Review Automation (SRA) is currently at its strongest, little further intervention was required to progress to the next task.

E. Task 5: Evaluate

Evaluation is the intermediate task between select and analysis, ensuring no categorically excludable artifact progresses to analysis.

- 1) Remove out-of-scope document subset (*in-scope document set*)
- 2) Evaluate relevancy measurement (*selection approval*)
- 3) Classify document subsets (*literature subsets*)

[Figure 6](#) depicts the Obsidian based visual interface used in *Task 5: Evaluate*.

A 5-score system was established, manually scoring each artifact already graded by the algorithm with a score from 5 (highly recommend) to 1 (reject), with an additional category 0 for categorically excluded artifacts such as duplicates, lists of abbreviations, university course structures, etc. The evaluation surfaced the many different perspectives and domains that can mention aerospace terminology. We assigned different weights to each domain to facilitate the classification of in-scope and out-of-scope documents:

- **5 strong match:** e.g. Aircraft design
- **4 weak match:** e.g. aerospace software development, aviation regulation and safety, aeronautics
- **3 borderline:** e.g. aircraft health, lifecycle management, mission planning, pilot training, naval, air traffic management, air logistics
- **2 weak mismatch:** e.g. e-commerce
- **1 strong mismatch:** e.g. military, air force, warfare, command and control

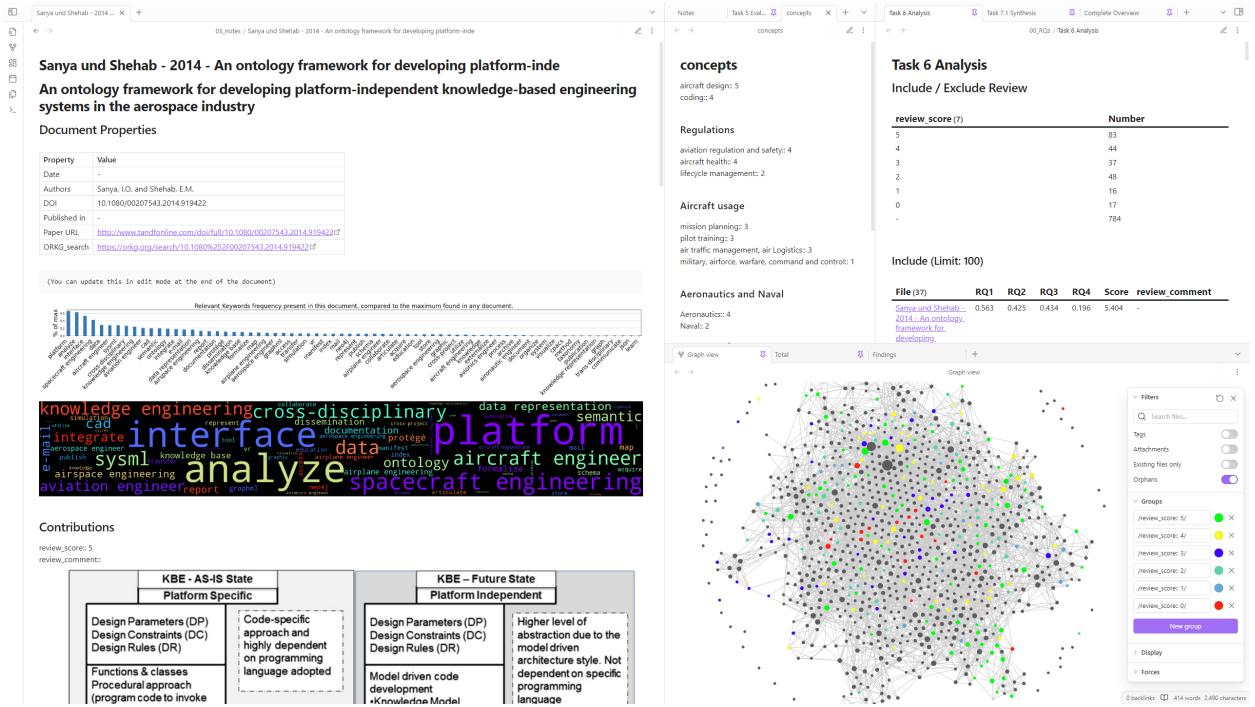


Fig. 6 Obsidian[¶] interface for Task 5: Evaluate

F. Task 6: Analysis

The analysis is a time-intensive manual process with a single step:

- 1) Annotate resources (*contribution representation*)

The qualitative analysis of the included and borderline papers was conducted by the first two authors of this work. Starting from the highest rated documents, each reviewer annotated with an individual focus on certain research questions. One reviewer focussed on how explicit knowledge is utilized and exchanged (RQ1 and RQ3), while the other reviewer paid particular attention on how explicit knowledge is formalized (RQ1). Since both reviewers read the same papers, both could verify and complement each other's findings. This synergetic effect became particularly prevalent during **Task 7: Synthesis**.

G. Task 7: Synthesis

Given annotated resources, the knowledge can now be synthesized into a systematic overview:

- 1) Compare contributions (*comparison*)
- 2) Craft arguments based on findings (*document claims*)
- 3) Critique the literature (*thesis claims*)

After an initial pooling of the analysis results, we inquired with the help of **Expert Interviews** to facilitate a research-question- and domain-tailored **Knowledge Graph Creation** process. We also utilized **Expert Surveys** to further strengthen

[¶]<https://obsidian.md/>

this data acquisition and interconnection. Finally, we conducted extensive knowledge graph curation towards, among others, semantic disambiguation, decomposition, and ontology mapping.

1. Expert Interviews

To streamline the annotation, a data model was designed to structure further synthesis. The first entries were sorted into a framework composed of categories such as processes, agents as well as data items, models, and formats.

Conclusively, a simple sentence covers all relevant classes:

Which Aerospace Engineering **Process** is completed by which **Software** using which **Data** in which **Format** and **Schema**?

All instances and their relations relevant to answering our research questions can be sorted into one of these classes. For example, RQ1 “*By which means, such as knowledge representations and schemas, do aerospace engineers externalize, such as formalize and visualize, knowledge?*” asks which data items and data models are used for the respective processes.

Below we provide examples for each class:

- Process: problems, tasks or activities.
- Software: tools, applications or agents.
- Data item: image, document, or three-dimensional (3D) model.
- Data schema: Common Parametric Aircraft Configuration Schema (CPACS), OWL, or Python.
- Data format: Extensible Markup Language (XML), JavaScript Object Notation (JSON), or Comma-separated values (CSV).

These interviews grounded the theoretical synthesis in the domain.

2. Expert Surveys

Just as Cook et al. [44] complemented their well-known literature review with a climate expert survey, we reached out to aerospace engineers. The initial idea was to mimic the [Expert Interviews](#), present [Figure 8](#), and (a subset of) the 500+ instances to complete and validate the findings. Surveys were distributed to provide a secondary source of information for validating the findings. With the help of the consulted engineers from said interviews, this scope was reduced, lowering the response threshold as far as possible: *I am working on <Task> using <Software> and <Data Format>*. Data items and models could then be derived from the used software and formats.

The initial survey was extended and distributed again with an additional entry for the position of tasks, tools, and file formats along the engineering V-model phases to gain insight into where knowledge-based practices typically occur in aerospace applications. Additionally, the survey served as an initial indication of how the classification of processes in the generated ontology could be supported by survey responses for future iterations of this study.

3. Knowledge Graph Creation

To properly synthesize a knowledge graph, many additional processing steps are required: First, many instances might be different representations of the same process, e.g. “implementation” and “realization”. Others are very specific, e.g. “collection and sharing of design information through a single source of truth (ssot)”, which is constructed from *collection of information of design by single source of truth*, including the same composition for sharing such information. For comparison, the Wikidata entry “[data curation \(Q15088675\)](#)” is “[subclass of \(P279\)](#)” “[content curation \(Q3007618\)](#)” “[of \(P642\)](#)” “[data \(Q42848\)](#)”. Curating the available data requires several knowledge engineering activities, such as merging duplicates, untangling composite instances, entity linking and human validation.

Statistical Analysis Co-occurrence and proximity of instances within our document set could be identified through statistical data analysis and used for identifying relationships between items of different classes, such as what software is used for what process, or what data formats are processed by what software. The software we developed extends SWARM-SLR to calculate these statistics and is available on GitHub***.

H. Task 8: Reporting

The final task of the literature review is to report on it:

- 1) Write the literature review (*review article*)

Naturally, the article itself is the final, central artifact of a literature review. But throughout a SWARM-SLR, several intermediate artifacts are created. Making these available to future research is also part of reporting. To further describe these artifacts, their content and the knowledge within, this description of the methodology is followed by an overview of the general [Results](#).

V. Results

This section illustrates the multitude of results of this SWARM-systematic literature review. [Figure 7](#) presents an overview of how these particular results converge into this article while being independently accessible. These artifacts include:

- a Zotero library††† of all 4,073 retrieved and processed references, alongside the literature used for this article, i.e. related surveys,
- a Zenodo dataset containing
 - our ontology with 782 individuals and relevant mappings to 154 contributions.
 - a selection of intermediate data, such as occurrences and association rules.
- several ORKG representations of the contributions, including

***<https://github.com/borgnetzwerk/tools/tree/main/scripts/SLR>

†††<https://www.zotero.org/groups/5420201/se2a-b42-aerospace-knowledge-swarm-slr/library>

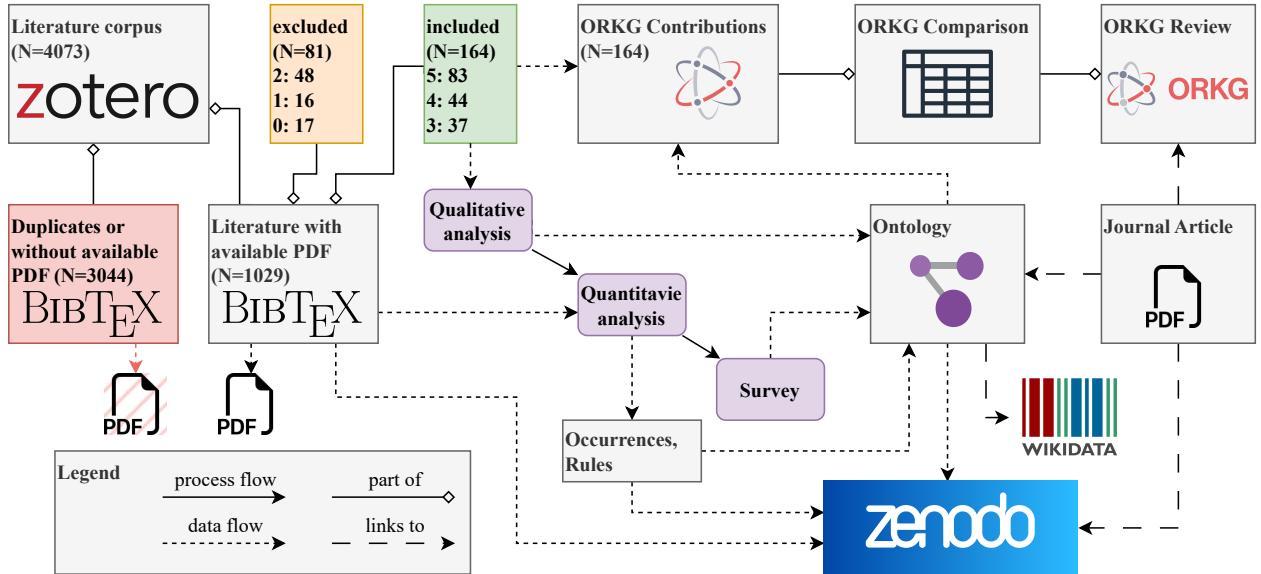


Fig. 7 Artifact overview from a Zotero library to an ontology, linked to Wikidata and represented in the ORKG.

- 944 ORKG papers^{###},
- an ORKG comparison^{\$\$\$} of the 164 included papers and their covered resources, namely processes, software as well as data items, schema and formats,
- a review^{¶¶¶} compositing these ORKG artifacts alongside more details regarding search queries, keywords, etc.

Several subsections and dedicated finding sections will structure these results:

- subsection V.A Qualitative Analysis gives insight into several most relevant of the 164 analyzed articles.
- subsection V.B Synthesis presents the ontology derived from this analysis, including its further detailing through expert surveys and interviews.
- subsection V.C Statistical Analysis discusses what is achieved with this ontology, ranging from a lightweight quantitative analysis over the entire retrieved literature to an ORKG comparison and review.
- section VI Process Findings, section VII Software Findings and section VIII Data Findings detail class-specific findings for the three major categories.

A. Qualitative Analysis

After the documents were computationally scored in Task 4, the highest scored articles were manually evaluated as described in Task 5. In total, we classified the 245 documents as shown in Table 3:

^{###}https://sandbox.orkg.org/observatory/KnowledgeBased_Aerospace_Engineering?contentType=Paper (Note to the reviewer: for the reviewing phase, we keep them on the sandbox or incubating servers, for camera-ready they will be published on production)

^{\$\$\$}<https://incubating.orkg.org/comparison/R819284>

^{¶¶¶}<https://incubating.orkg.org/review/R819286>

Table 3 Scores from the manual review of the 245 highest-rated documents

5 strong include	4 weak include	3 borderline	2 weak exclude	1 strong exclude	0 categorical exclude
83	44	37	48	16	17

As described earlier, reviewers annotated the 164 included and borderline papers concerning their contributions to the three research questions. A selection of particularly noteworthy literature includes the following:

Sanya et al. is likely one of the prime authors to highlight in this SLR, discussing semantic knowledge management challenges in aerospace [45] and the potential of using ontologies there [46] a decade ago. A set of noteworthy sentences warrant citation:

“It was established that improving the structure and representation of knowledge would benefit and increase the competency of design engineering activities. [...] The authors suggest that something more clear and concise is needed beforehand, namely that knowledge is structured and organised in an appropriate manner with appropriate vocabulary. The authors argue that this structure is needed in order to bring forth the next generation of knowledge base building, and ontological engineering is a prerequisite for this vision.”

–Sanya et al. [45]

They specifically mention that “engineers often spend large amounts of time searching for a solution to problems that may have been solved.”, partially because “engineers are sometimes reluctant to share documents” or “do not want to be burdened with the task of knowledge classification or annotating the semantic contents of a document”. The authors recommend “acquiring a dedicated knowledge/ontological engineer that will be responsible for such an activity”. For details, reading the entire paper and ideally some consecutive work like [46] is highly recommended.

The dissertation by Xie [47] provides a good introductory work as to what KBE, particularly knowledge itself, may mean in the context of aerospace. The work also contains a review of systems classified by knowledge capture automation, data mining techniques deployed, origin and applications. Exactly such overviews were already required back then and are now facilitated through interoperable technologies, such as the ORKG, as also employed in this work later. Using this technology, their identified knowledge is reused and enhanced for future dynamic extension. They describe a gap for exactly that as their final key finding, “to apply semantic representation to capture aerospace context in an integrated and open manner.”

Gonzalez [48] provides an overview on complex multidisciplinary system composition for aerospace vehicle conceptual design, where particularly the list of 126 *Aircraft Synthesis Systems* (based on [49–51]) is of interest to this work. It is apparent that semantic ambiguity even occurs within collections, within the same departments, listing the abbreviation to both mean “Multi-Disciplinary Integrated Design Analysis & Sizing” for DaimlerChrysler Military, as well as “Multi-Disciplinary Integration of Deutsche Airbus Specialists” for DaimlerChrysler Aerospace Airbus.

Examples like these highlight the complexity (tool-assisted) entity recognition has to face when digesting such knowledge sources, where arguably not even humans could differentiate one MIDAS mention from the other, given that both existed as separate instances in the first place.

Haney [52] speaks of “aerospace data-engineering”, continuing the aforementioned issue of polysemes and synonyms. Likely, the tasks of an aerospace data engineer are virtually identical to a aerospace knowledge engineer. Formalizing these “said to be the same as” relationships is equally important for maintaining interoperability in systems that do not use perfectly aligned vocabularies, as is also the case in the scope covered here. Haney is also a strong source for making the flow of knowledge and the benefit of capture and reuse tangible, spanning several illustrations of how accessible, actively used digital libraries enrich the engineering process.

Aigner et al. [53] discuss Knowledge- and graph-based Agile Design with System (KADMOS), Common workflow schema (CMDOWS) and remote component environment (RCE) in the context of graph-based formulation and visualization of MDO systems. As the nested abbreviations already suggest, these highly intertwined systems are very complex and already benefitting from a KBE approach. It is a very good example of knowledge-based, interoperability-focussed system design, widely used throughout the AGILE project it was part of. Many related works of the project, such as [54] of the AGILE 4.0 followup project, further build on these interoperable frameworks.

Van Gent et al. [55] discusses similar themes, tools and models, with focus on CPACS as a central data model. It is positioned as a central data model to reduce the complexity of $O(N^N)$, every system X_i developing its own adapter for every other system $X_{j \neq i}$ and back, to 2 adapters converting data from $X \rightarrow$ CPACS and CPACS $\rightarrow X$, $O(2N)$. Recommendations of such universal aerospace interchange formats go back to 2004 [56], potentially beyond.

Prock and Ochoa [30], as teased by the related work section, is actually a literature review of knowledge management and knowledge collaboration systems in the aerospace and defense industries. They build upon everything mentioned up until now and propose an OWL based knowledge management system for the aerospace domain. Their work follows a very similar research question, but using a less systematic literature review approach on a smaller subset. They do, however, go much more into detail regarding the specific needs of the aerospace domain. They, too, identify a quote-worthy sentiment:

“In addition to these challenges, the aerospace and defense industries are perhaps the most governed of all sectors, being regulated by a plethora of agencies, both national and regional [22]. KM is, put simply, about providing knowledge to people quickly; yet the tight regulations of these industries may hamper effective knowledge dissemination.”

–Prock and Ochoa [30] (and their source “[22]”, Harvey and Holdsworth [57])

They follow this up with by discussing knowledge protection:

“As aerospace and defense sector project work is often performed in a collaborative effort between firms, there is concern that a firm’s specialized knowledge may be appropriated by the cooperating firm during collaboration, in a move of opportunistic advancing. “... firms experience a fundamental paradox: to gain the greatest benefits they must exchange information and knowledge with external parties yet, at the same time, they must protect themselves against knowledge appropriation. This dilemma is particularly acute in the aerospace sector where political imperatives strongly influence partner choice and collaborators are often strong rivals in other contexts” [68].”

—Procko and Ochoa [30] (and their source “[68]”, Jordan and Lowe [58])

Conclusively, they advise for knowledge sharing despite the challenges, recommending accessible knowledge graphs for collaboration. Since their recommendation is again exactly aligned with what our work has done, we can confirm that independently our conclusions overlap to recommend the aerospace domain to adopt FAIR knowledge management, as exemplified by our work’s immediate knowledge-graph-based results.

Every one of the 164 documents, whether mentioned here or not, contributed to the creation of the knowledge graph, whose synthesis is discussed in the following subsection.

B. Synthesis

While we could already identify 533 total aerospace KBE items via literature review, we validated our scope through expert interviews and surveys, as described in the previous chapter. The results are displayed in the following subsections:

1. Wikidata Linking

Eventually, instances were manually mapped to Wikidata items for disambiguation, following the Linked Data approach. For example, *Cypher* might mean one of 607 different things, but in our knowledge graph is clearly defined by being mapped to <https://www.wikidata.org/wiki/Q16834355>. As another example, Wikidata contains 69745 file formats, many of which are likely versions of the same file format families. The lexeme *trace* alone has 41930 entries in which we need to identify *TRACE*, a software used for *computational fluid dynamics (CFD)*.

Our literature review revealed that 40 % of the 215 processes, 37 % of the 135 software, 52 % of the 65 data items, 54 % of the 68 data models, 90 % of the 50 formats could be mapped to Wikidata entries.

The more universal the relevance of an instance, like *XML* or *optimization*, the easier their identification. Remarkably, even largely discussed domain-specific instances like *Common Parametric Aircraft Configuration Schema (CPACS)* or *multi-disciplinary optimization (MDO)* did not have Wikidata entries. Others, like the *National Airspace System (NAS)*, were identifiable via Wikidata, but could not be categorized into our scope of software, processes, and data, since they are instances of concepts outside these categories, and thus excluded from our ontology.

2. Expert Interviews

We invited three aerospace engineers over three months for individual, intermediate evaluations. These interviews resulted in a refined concept depicted in [Figure 8](#). For example, *agent* was reduced to a subcategory of agents,

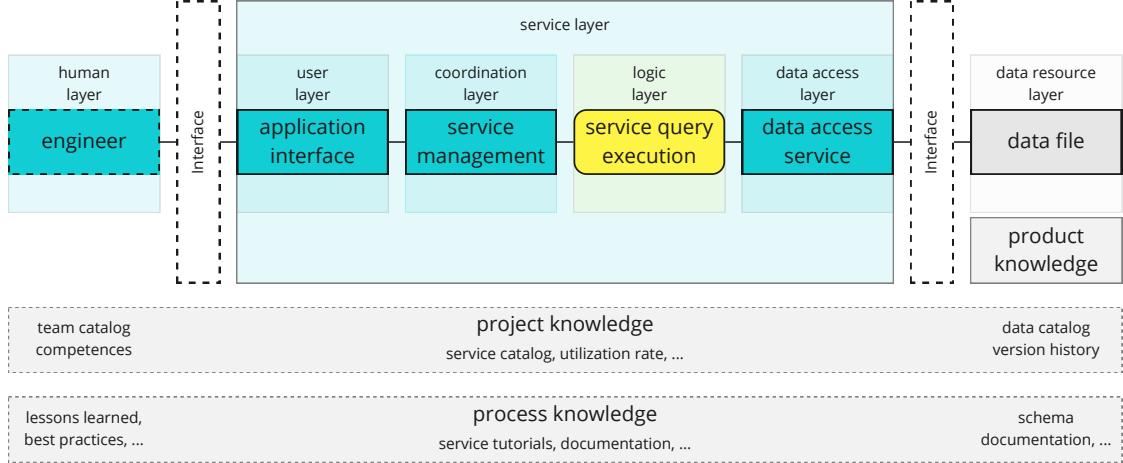


Fig. 8 Knowledge architecture concept, based on surveyed literature (especially Herrero [59]). An engineer interfaces with a service, which in turn interfaces with explicit product knowledge in the form of data. Beyond product knowledge, knowledge-based engineering also covers project knowledge (to manage the product creation process) and process knowledge (to solve tasks efficiently).

namely *software*, which is technically more restrictive and potentially not a perfect fit, for example when considering non-software tools, services, or the involvement of a human actor in a process. These compromises were taken to reduce an initial precise, but abstract ontology to a simple, yet sufficient model.

3. Expert Surveys

A dedicated workshop for this systematic literature review was held within the SE²A cluster in the summer of 2024. 16 participants were introduced to the survey, approach, and results, and were offered to contribute to the survey. 14 responses were collected that day. The survey was then distributed to the entire cluster of over 200 engineers working in the aerospace domain. Over the following three months, three additional responses were collected. The second survey included a brief description of the V-model phases for the participants' reference and was distributed to additional universities, such as Delft University of Technology (Technische Universität (TU) Delft) and Linköping University (LiU), industry partners, and again in the SE²A cluster. Ultimately, this survey received answers from a total of 14 engineers. These answers contribute an extension to the answers for the first survey while providing some additional insight into where knowledge-based practices typically are used along the engineering V-model phases. The results from this analysis, along with a brief description of the engineering V-model, can be found in [section VI](#).

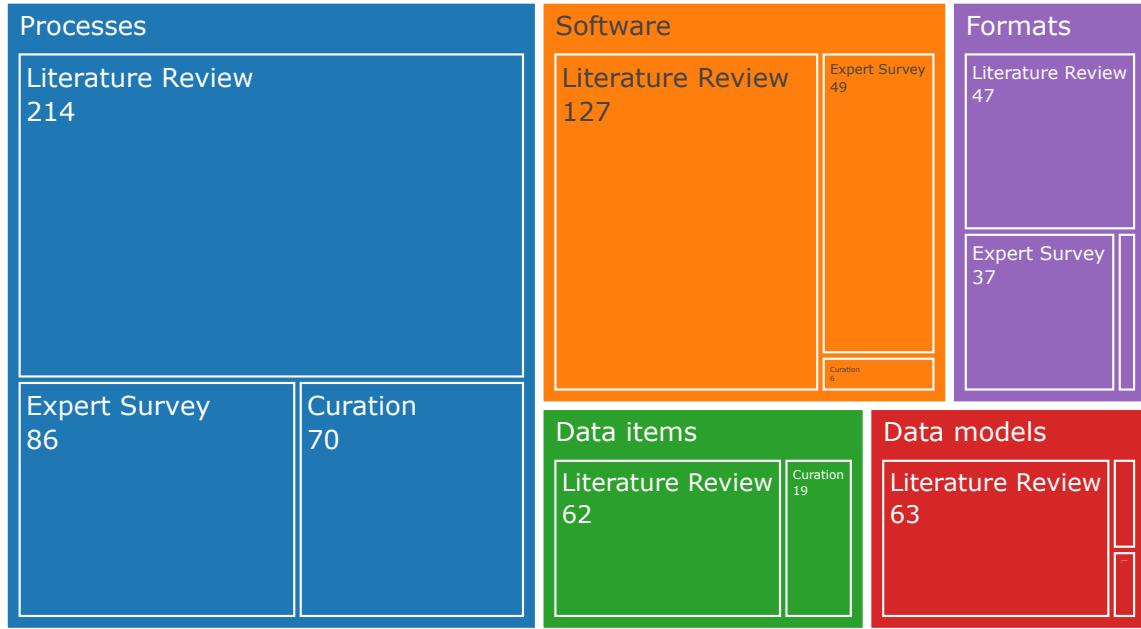


Fig. 9 Treemap of the Source distribution of the knowledge graph items

4. Knowledge Graph Curation

As mentioned, 513 aerospace KBE items were identified throughout the literature review. With the expert surveys, this total grew to 695 instances. The final, extensive curation raised the total to 792 instances as illustrated in [Table 4](#) and [Figure 9](#). This table also specifies the percentages of items gained from each source. The literature review contributed the most items (65%), proving most effective for data models (90%) and least effective for formats (53%). The expert survey added 23% of the items, with the highest increase in formats (42%) and no additional data items. Finally, curation accounted for 12% of new items, with the largest gain in data items (23%) and no significant increase in software, data models or formats.

Table 4 Sources of the aerospace engineering knowledge graph items

Source	Total	Processes	Software	Data items	Data models	Formats
Literature review	513 (65%)	214 (58%)	127 (70%)	62 (77%)	63 (90%)	47 (53%)
Expert survey	182 (23%)	86 (23%)	49 (27%)	0 (0%)	4 (6%)	37 (42%)
Curation	97 (12%)	70 (19%)	6 (3%)	19 (23%)	3 (4%)	5 (6%)
Total	792	370	182	81	70	89

Though the performed curation is extensive, it still leaves room for further improvements, including inference, connection to existing ontologies, further interrelationship mapping, etc. This curation is final for the scope of the present SLR, but just a first step for a future interoperable knowledge base for the aerospace engineering community.

C. Statistical Analysis

Given a large, natural-language-processed document set of 1029 documents and a knowledge graph, we applied quantitative analysis techniques to derive insights. These methods produced several visual artifacts based on statistical findings to further support curation. This subsection provides an overview of the quantitative analysis techniques that have been used. For access to the very large visualization, please see the Zenodo dataset¹⁷.

1. Paper-Instance-Occurrence

Using the extracted text and the exported ontology, literal matches in the text were identified. Immediately, this enabled a binary occurrence matrix as a quantitative complement to the conducted qualitative analysis. While the occurrence matrix, having dimensions of 793×1029 (or 793×164 if just for the reviewed papers) is too large to be displayed here, it is available in the appendix as [Figure 15](#), and in full as part of the Zenodo dataset and on the ORKG, as further described in [subsubsection V.D.1 Open Research Knowledge Graph](#).

Having this occurrence data alongside the rich semantic data also allowed for dedicated, class- and year-specific overviews, as depicted in [Figure 10](#). This matching is by its very nature inaccurate. False positives occur, not only when one MIDAS tool is misinterpreted for the other [48], but also if an occurrence of the word “give” is misinterpreted for the Geographic Independent Virtual Environment (GIVE) [56]. This effect is especially strong the more common a word is (like “word” itself), presenting more potential for false positives, and the less used an instance itself is, resulting in fewer true positives. Initially, these false positives vastly outmatched the true, which is why several safeguards were implemented. Matches inside other words, like *engineer* in *engineering*, were no longer counted, with the resulting false negatives being somewhat compensated by introduced aliases such as *engineers*. To not only match *knowledge extraction*, but also *extraction of knowledge* or similar, a checkbox was established to divide an instance into its containing words and check if all co-occur in a 100-character window.

None of these checks could prevent all false positives. To the authors’ knowledge, automated semantic disambiguation is currently not capable of preventing them, and potentially never will be for some cases, taking earlier mentioned MIDAS for example. While none of these checks could remove them entirely, they cleaned the data to a degree usable as a demonstrator of how semantically described knowledge could look like. Every occurrence should be taken with appropriate consideration for false positives, and the presence of false positives should further support the recommendation of dedicated human oversight in the loop, preferably connected as a knowledge community. With all that in mind, these checks, such as the co-occurrence window, allowed us to strengthen a different set of statistical rules, as described in the following subsection.

¹⁷https://zenodo.org/records/14790367?preview=1&token=eyJhbGciOiJIUzUxMiJ9.eyJpZCI6ImQzMzczMjY1LWI2NmMtNDhiNS04Y2NjLTMwOGRIgzdmgs-5KInBiNzr8pSu-YT0FzcGuMc76_F-PHW2jlFn-niYjbFBRxTZHBwUfkLEnAYyRW2e0SNy98cfA8yYRZk5cA

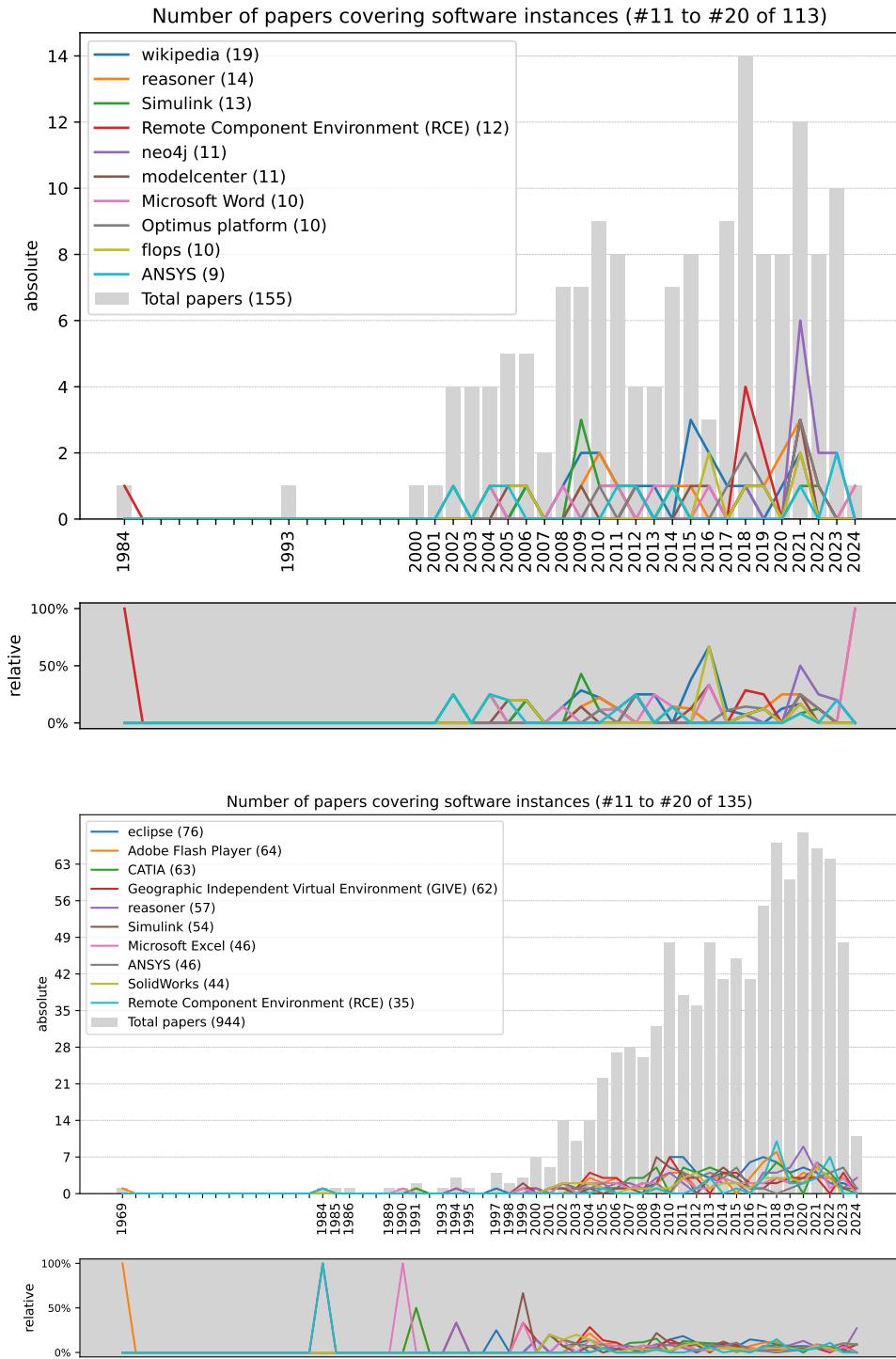


Fig. 10 Instance occurrences in reviewed literature (above) compared to occurrences in entire corpus (below).

2. Instance-Instance-Co-occurrence

The previous paragraph introduced checks for detecting the fragment words of “knowledge engineering” being present within a 100 character window. This also allows to check if any other instance co-exists in the same space, hinting at a connection between the two. We applied the association rule mining algorithm to the identified instances across the papers and filtered the mined rules using a support value of 0.4 and lift metric [60] with a threshold of 1. An example of the identified rule is the connection from the process “knowledge management” to the data item “document” with 94 % confidence, support of 0.45, and a lift of 1.18. Such association rules were employed to visually represent a graph, which is again too large and too dense for display here. A preview is provided in the appendix as [Figure 16](#), the entire version is in the Zenodo dataset. Equally, false positives from the aforementioned issues, alongside co-occurrence not being a perfect proxy for relationships, resulted in only strong association rules being used for further curation. They were added to our knowledge graph as relations between the instances, using the object properties already established for paper-instance occurrences. All of these results culminate into the knowledge graph of

- 8 classes, 9 annotation properties, 13 object properties and 957 individuals;
- 14556 axioms, 957 of which are class assertions, 4258 are annotation assertions and 8334 are object property assertions.

How to best access this data is described in the next subsection.

D. Open Accessible Knowledge Graph

Our knowledge graph was curated using WebProtégé¹⁸ [61], which is perfect for curating with a set team but not designed as an open-access knowledge infrastructure like Wikidata. Similarly, Wikidata is designed to be a general-purpose knowledge base, not a domain-specific knowledge graph, neither for science nor aerospace engineering. As experts like Sanya et al., Xie, Procko and Ochoa have pointed out, however, exactly this collaborative knowledge-sharing is what the domain needs. As such, we present our results as foundations of a future knowledge-based aerospace engineering knowledge community, on the [Open Research Knowledge Graph](#) and a dedicated [Wikibase](#) instance.

1. Open Research Knowledge Graph

The ORKG contains the entire knowledge graph created by this literature review:

- Every single instance is an ORKG resource, providing unique identifiers for future reuse. If they are more richly described, such as Computer Aided Three Dimensional Interactive Application (CATIA) ([R1355942](#)) in [Figure 11](#), their relations to other resources are easily accessible.
- All papers are added to the ORKG, listing their contributed processes, software and data. This mapping allows identifying papers that mention a resource of interest, see [Figure 12](#) as an example.

¹⁸<https://webprotege.stanford.edu/>

- All reviewed papers are added to an ORKG comparison, which makes it possible to filter papers as depicted in [Figure 13](#). For example, one paper listing both the KADMOS data model and the CATIA software is the paper ([R1356673](#)), representing Van Gent et al. [55]. The comparison is often the most desired result of a systematic literature review. The comparison presented here is an automatic result of properly represented paper contributions and resources in the ORKG. As such, any further contributions added to the knowledge graph can simply be added to the comparison, extending and updating it.
- The comparison is added to an ORKG review¹⁹. Alongside additional relevant context and navigational guidance over the ORKG artifacts, it serves as a living version of the present SLR that can be updated with new papers.
- Every ORKG artifact described by now is added to the ORKG observatory dedicated to Knowledge-Based Aerospace Engineering²⁰. This observatory is specifically setup to provide a structure for a future knowledge-based aerospace engineering community. It provides a common root to every relevant node in the graph, regardless of the specific problem or context at hand. Every paper or comparison relevant to the domain can be connected to this central node, alongside visualizations, reviews and other artifacts, making it the domains' main entry point for relevant, curated knowledge.

2. Wikibase

In several instances, we have highlighted the importance of Wikidata for open linked data. A significant amount of the domain-specific software and processes did not have Wikidata items. Changing this by steadily increasing the representation on this central knowledge graph is desirable, but not uncomplicated. To allow as many aerospace engineers to test and experiment with semantic representations, we have setup several testing grounds: Where the ORKG has the sandbox²¹ to collect experience, we have setup a dedicated wikibase instance for the aerospace community: aerospace.wikibase.cloud.

This Wikibase provides an example of how a sovereignly manageable, but fundamentally interconnected knowledge graph could be a central knowledge sharing infrastructure, fitted to the domain specific requirements. Ideally, such a Wikibase is picked up by central institutions representing aerospace, such as National Aeronautics and Space Administration (NASA), where MediaWiki is already used for internal knowledge management, AIAA, Deutsches Zentrum für Luft- und Raumfahrt (DLR), TU Delft, or similar. Independently, even a single engineer could populate this or their own Wikibase, or start contributing on wikidata and the ORKG right away. The knowledge infrastructure, required by the previous authors, is available today.

Following the overview on how our identified knowledge may generally be accessed, the sections [Process Findings](#), [Software Findings](#) and [Data Findings](#) are dedicated to focus on class specific findings.

¹⁹<https://incubating.orkg.org/review/R819286>

²⁰https://incubating.orkg.org/observatory/Knowledge_Based_Aerospace_Engineering

²¹<https://sandbox.orkg.org/>

CATIA

28 January 2025 - 16:02 → Referred 28 times Created by Ildar Baimuratov Extraction: Unknown

Organization Sustainable and Energy-Efficient Aviation Observatory Knowledge-Based Aerospace Engineering

Resource information In papers In statements

Data browser

CATIA

Instance of: Software Applied template: Software template → Referred: 28 times

alias	Computer Aided Three Dimensional Interactive Application Text
alias	CATIA V5 Text
data format specification	catdrawing C21024
Source	expert survey Text
alias	.catdrawing Text
data format specification	catproduct C21024
Source	expert survey Text
alias	.catproduct Text
data format specification	catpart C21024

Fig. 11 CATIA resource information and paper occurrence view on the ORKG²²

Resource information In papers In statements

- Status, Comparison, and Issues of Computer-Aided Design Model Data Exchange Methods Based on Standardized Neutral Files and Web Ontology Language File
- Yuchu Qin, Lu, Wenlong, Qunfen Qi, Liu, Xiaojun, Yanru Zhong et al. 2017
 - Path: Paper » KBAE SLR
- Knowledge based engineering: notion, approaches and future trends
- Reddy, E. Jayakiran, Sridhar, C. N. V., Rangadu, V. Pandu 2015
 - Path: Paper » KBAE SLR
- Knowledge-Based Management System and Dearth of Flexible Framework for Software Development
- Wemembu, U. R., Okonta, O. E., Imiere, E. E., Ajani, D., Owolabi, A. A. 2015
 - Path: Paper » KBAE SLR

Fig. 12 Occurrence of the CATIA resource in papers on the ORKG

²²<https://orkg.org/resource/R1355942>



The image shows a screenshot of a comparison interface between CATIA and KADMOS. The interface has a sidebar on the left with categories: Properties, data format specification, data item, data model, process, and software. The main area compares terms from both systems.

Properties	Agile MDAO systems: a graph-based methodology to enhance collaborative multidisciplinary design_v4 <i>Contribution 1 - 2019</i>	Cloud-Based Environment for Aircraft Design Collaboration_v4 <i>Contribution 1 - 2021</i>
<u>data format specification</u>	<u>.inp</u> <u>.pdf Portable Document Format</u> <u>.pdf Portable Document Format (PDF)</u> <u>Access Add-in file format</u>	<u>comma-separated values (csv)</u> <u>eXtensible Markup Language (XML)</u> <u>MATLAB M-File</u> <u>Simple Object Access Protocol/v (SOAP)</u>
		<u>dat</u> <u>eXtensible Markup Language (XML)</u> Show 4 more
<u>data item</u>	<u>CAD model</u> <u>database</u> <u>description</u> <u>document</u>	<u>CAD model</u> <u>database</u> <u>description</u> <u>document</u>
	<u>eXtended Design Structure Matrix (XDSM)</u> Show 9 more	<u>form</u> Show 4 more
<u>data model</u>	<u>C_programming language</u> <u>common mDAO workflow schema (cmdows)</u> <u>common parametric aircraft schema (cpacs)</u>	<u>C_programming language</u> <u>common mdao workflow schema (cmdows)</u> <u>common parametric aircraft schema (cpacs)</u>
	<u>EXPRESS</u> <u>Javascript Object Notation (JSON)</u> Show 8 more	<u>EXPRESS</u> <u>Javascript Object Notation (JSON)</u> Show 7 more
<u>process</u>	<u>Knowledge- and graph-based Agile Design with Multidisciplinary Optimization System (KADMOS)</u> Show 26 more	<u>Knowledge- and graph-based Agile Design with Multidisciplinary Optimization System (KADMOS)</u> Show 16 more
<u>software</u>	<u>analysis</u> <u>assembly</u> <u>calculating</u> <u>certification</u> <u>computational fluid dynamics (CFD)</u> <u>computer-aided design (CAD)</u> Show 21 more	<u>analysis</u> <u>calculating</u> <u>computational fluid dynamics (CFD)</u> <u>computer-aided design (CAD)</u> <u>creation</u> <u>design</u> Show 11 more

Fig. 13 Comparison with filters for CATIA and KADMOS applied²³

²³<https://incubating.org/comparison/R819284>

VI. Process Findings

A challenge in the created ontology was to classify the obtained processes in a structured way. As seen before, the question from subsection IV.G of “Which Aerospace Engineering **Process** is completed by which **Software** using which **Data** in which **Format** and **Schema**? ” was used to initially classify the knowledge in a selection of distinct classes. However, to further improve the ontology structure, the processes themselves also had to be classified. Taking the research questions as an example, we asked “by which means, such as knowledge representations and schemas, do aerospace engineers externalize, such as formalize and visualize, knowledge? ”. This externalization already categorically includes formalization and visualization, but likely other processes. Similarly, the other questions contain “utilize, such as organize and interface with,” and “exchange, such as transfer and distribute, explicit knowledge”. These knowledge engineering processes are intertwined with engineering processes prevalent throughout all product development phases. As such, we provide some context on the structuring of processes along the development in subsection VI.A and conclude this section with the findings of the Qualitative Study and Quantitative Study.

A. Knowledge-Based Practices along the V-Model

The V-model was chosen as a suitable structure for classifying the various processes found in the ontology for this work. This involved creating processes for each of the V-model phases from Fig. 3 and thereafter classifying each process to its corresponding phase or phases. Naturally, this classification is prone to mistakes and biases without proper support and guidance from experts for each process instance. However, certain processes such as “aircraft concept analysis” can with a relatively high certainty be classified as belonging to the “System-Level Requirements” phase from Fig. 3. Processes such as “document classification” are on the other hand harder to classify due to their knowledge engineering nature allowing them to be valuable indifferent to the current V-model phase.

To support a proper classification of the processes, the V-model positioning part in the second survey was added as a means for receiving expert inputs on where they typically use their tasks, software, and data formats.

Even though the second survey from subsubsection V.B.3 saw a low response rate, some initial insights on where knowledge-based practices typically are used along the different phases of the V-model could be gained. Figure 14 illustrates the responses from the survey for where tasks related to knowledge-based practices typically are used in aerospace engineering,

The size of each circle in Figure 14 corresponds to the number of entries from the positioning in each phase of the V-model according to the second survey results. Also, the number in each circle represents the corresponding V-model phase from Figure 3. As seen, the left-hand side of the V-model is where most of the knowledge-based practices take place. This is particularly true for the “High-Level Design” and “Component Design” phases that received the largest number of entries. Consequently, knowledge-based practices are predominantly used in the phases of the “Definition and Decomposition” part of a development process and thereby indicate that many of the processes in the ontology could be

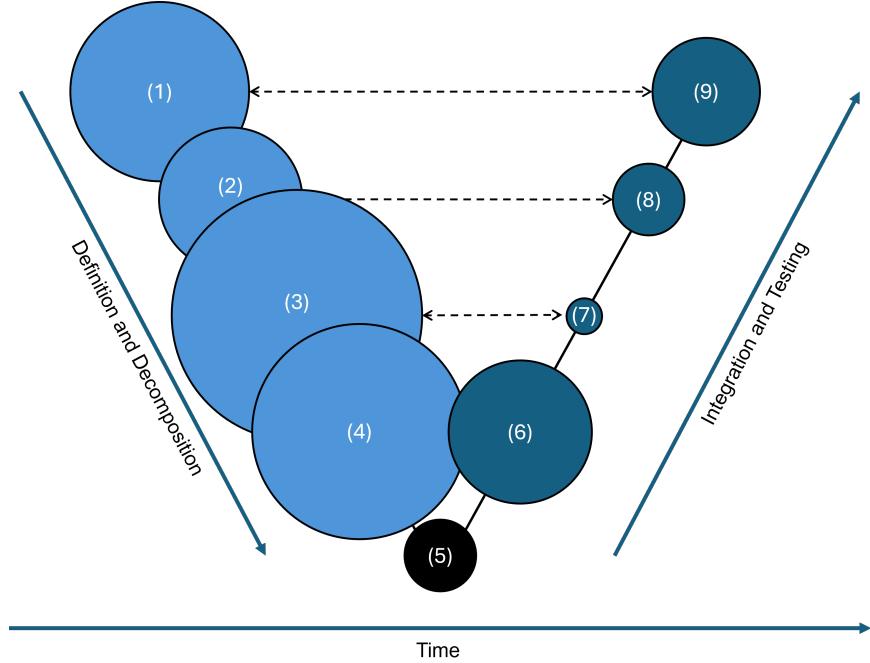


Fig. 14 The answers from the second survey positioned along the V-model.

classified as belonging to this side and its corresponding phases. As a result, this reduces some of the classification uncertainty for the processes that could not be classified directly from the survey answers. However, the positioning of survey answers along all phases of the V-model can be observed in [Figure 14](#). One of the explanations for this is that many engineers also deal with the corresponding phases on the “Integration and Testing” side. Interestingly, there is a large difference between the subsystem requirements and the subsystem verification phases. This may consequently indicate that engineers working on high-level design are typically less involved in the verification processes. However, more survey responses are needed to draw a proper conclusion about this. Overall, the results from the second survey and [Figure 14](#) indicate that these responses help with classifying the processes into the generated ontology along the V-model phases. However, more survey participants are needed to substantiate a comprehensive and correct classification of all identified processes in the ontology.

B. Quantitative Study

In the knowledge graph, 370 instances were classified as processes. All 164 papers mention some process, where Li et al. [62] mentions the most number of processes, 26, and Adala et al. [63] mentions the least with just three. The most frequently mentioned process is design, 163 papers mention it, while “flight test reporting”, “information push”, “semantic information transferring”, “semantic lexicon”, “share documents”, “design concept formalization”, “engineering java programming” and “tooling development” were mentioned only once. We have identified associations for 58 processes with software, where the “function plotting” process is associated with the most number of software,

three. Eleven processes are associated with two software instances and the remaining 46 processes are associated only with one instance. There are 29 processes associated with a data item. Manufacturing and reasoning are associated with five data items, while 15 processes are associated with four data items, two processes are associated with three data items, the only inference is associated with two data items, and nine processes are associated with only one data item. Finally, 110 processes were identified with Wikidata entries, which makes almost 30% of all processes. 214 processes come from literature review, 86 from expert survey, and 58 from curation, where 6 processes have more than one source.

C. Qualitative Study

The main process types of interest were the aforementioned externalization, utilization and exchange of knowledge. As such, any processes such as formalization, visualization, interfacing and distribution were sought after. Evidently, various methods for formalization were found, as will be further described in [section VIII](#). Different pathways of interfacing were also described, ranging from user-centered interfaces like Excel or a Systems Modeling Language (SysML) diagram described in Herrero [59] to highly decentralized and automatized tool-chains, as discussed by Flink et al. [64] and many others. These tools attempt to address the issues highlighted for over a decade, e.g. by Jackson and Hildreth [65] when stating that “The current state-of-the-art is such that several staff-months if not staff-years are required to ‘rehost’ each release of a flight dynamics model from one simulation environment to another one”. These tools will be further discussed in [section VII](#).

Especially troubling, however, were the findings towards distribution and visualization. The general findings already described the issue of distributing knowledge in aerospace. Royal [66] specifically states that aircraft development in a “central hub environment is generally not a practical option”, since partners “wish – or are even forced” to keep their data and tools on their own premises. Yet, Vucinic [67] emphasizes the importance of expert synergy, particularly between computer scientists and engineers, focussing on the necessity of interpersonal knowledge exchange through means of visualization. “Visualization”, however, “is only an issue for human interpretation”, as [68] state. In a field where sharing knowledge with others is already a low, potentially negative priority, establishing a common language, particularly a visual one, is tricky. As such, we have identified a knowledge visualization gap, to be discussed in the following subsection.

D. Gap Analysis

As established, aerospace engineering knowledge is complex, highly heterogeneously modeled, scarcely exchanged between humans and even less between organizations. As such, visualization is of very little priority, barely not existent beyond several classes of diagrams. Of particular interest is that the forefront of knowledge engineering, OWL, lack of comprehensive visualization means at scale. WebVOWL²⁴ for example seems not capable of displaying instance nodes.

²⁴<https://github.com/VisualDataWeb/WebVOWL>

This functionality is only available in tools like OWLviz²⁵, an extension for the desktop Protégé, and WebProtégé²⁶, the more modern, lightweight web version, which was primarily used for in this work. To the authors' knowledge, both are state-of-the-art and, while they are to be commended for their capabilities, are still beneath today's requirements. WebProtégé has the cleanest display and untangling of resources, but little customization beyond that, becoming impractical beyond a few dozen connections. OWLviz provides exactly this customization, but its default sorting are practically unusable at scale, leading to hours of manual time investment required for displaying a single graph, only for this time investment to become largely irrelevant as soon as the graph is updated and loading an old save is no longer meaningful. Since not only aerospace, but many other domains need to describe ever more complex systems at ever more degree of detail, it is an alarming signal that today's state of the art appears to be a decade old. An updated, more flexible and better way of visually representing knowledge in ontologies is therefore a desirable feature and gap yet to be filled.

VII. Software Findings

By which means, such as knowledge bases and services, do aerospace engineers utilize, such as organize and interface with, explicit knowledge? By which means, such as tools and methods, do aerospace engineers exchange, such as transfer and distribute, explicit knowledge? These questions particularly lead the inquiry in our survey. In short, which tools do aerospace engineers use to externalize, utilize, and exchange knowledge? As with the previous and following section, the findings regarding this question are separated into [Quantitative Study](#) and [Qualitative Study](#), concluded by the [Gap Analysis](#).

A. Quantitative Study

The 182 tools were identified, where 127 come from the literature review, 49 from expert surveys and only 1 from curation. The most number of tools mentioned, 16, is in Herbst [69]. “Access” was mentioned in 149 papers, followed by “Word” mentioned with 102 mentions and API with 74, while 14 tools were mentioned only once. The first two are pristine examples of false positives since both “access” and “word” are common lexemes. There are 29 associations between software tools and processes. Software associated with the most number of processes is logically Access, with 31 connections, followed by MATLAB with ten, and both Simulink and Wolfram Mathematica with four processes, while 17 software instances are associated only with one process. Four tools are associated with a data item, with “Word” having the most number of data items, eight. Eight tools are associated with a data format, where TAU has the most number of connections, eight, and three tools are associated with only one format. Finally, five tools are associated with a data model. When trying to disambiguate tools using Wikidata, only 37% of the initial 135 software instances were linked.

²⁵<https://protegewiki.stanford.edu/wiki/OWLViz>

²⁶<https://webprotege.stanford.edu/>

B. Qualitative Study

Tools generally can be classified into one of the following two groups: Highly popular tools, such as CATIA, MATLAB or Excel, are universally usable and accessible, and niche specialized tools are mostly used by their developers, like cpacsy or DAVEtools. Especially the latter group is prone to be entirely unavailable, discontinued, or rebranded by the time we identified them, such as Vivisimo, now called IBM Watson Explorer for example. Generally, the popular group was highly condensed, and the lower popularity was widely distributed.

In trying to disambiguate these resources using Wikidata and web search, we coincidentally came across a r/AerospaceEngineering Reddit post²⁷, largely confirming this finding of a very small set of software being used for a majority of all tasks. As for more scientific sources, some publications described workflows throughout interoperable software, called software workflow management frameworks, such as the remote component environment (RCE), particularly for coupling multi-disciplinary analysis and optimization (MDAO) processes. In these large conceptual mappings, several tools were just abbreviations, which sometimes simply referred to steps, not tools, e.g. “TR” for “Term Recognition”. Some of these abbreviations could be disambiguated, like CoOM referring to a tool developed at TU Darmstadt, or were difficult to find in the literature, such as COCO, a boundary layer solver developed in 1998 at TU Braunschweig [70]. Especially unhelpful for identification is the prevalence of naming software after existing concepts, such as Fiji or SALOME, which could only be mapped if the tool in question was described well enough. While SALOME even had a Wikidata entry, the majority were barely described at all, with some publications not even mentioning the tools they used, or just mentioning their concepts like “Python application”. For example, a process called Process Integration and Design Optimization (PIDO) ([Q7119347](#)) will have PIDO platforms and applications associated with it, but these might not be named.

Concluding this overview of what software is used, we can say with relative certainty that the answer is generally “Microsoft Office, a small set of widely applicable scripting tools, one or two use-case specific software to solve exactly one given problem, and a self-developed software with short development lifespan that has little usage outside the current project”.

C. Gap Analysis

As for our leading questions, which tools do aerospace engineers use to externalize, utilize, and exchange explicit knowledge, the general answer appears to be “barely any, if at all”. Knowledge is usually not formalized, but distributed via resource management on the human resources level, in reports, meetings, and direct communication. Explicit knowledge bases are widely mentioned, recommended, established, then rarely used, abandoned, and eventually cease to exist. If they continue to exist, they are generally closed to the outside, such as at NASA or Siemens.

²⁷https://www.reddit.com/r/AerospaceEngineering/comments/mr9unc/what_software_do_you_use_as_an_aerospace_engineer/

To our knowledge, there are no open-access aerospace knowledge graphs, wikibase instances, ORKG observatories, or similar. The closest thing could be the Wikipedia Aerospace Engineering glossary²⁸, all other open and collaborative knowledge bases seem to have been discontinued. As such, it is vital that collaborative infrastructure is reused, such as a wikibase.cloud instance, to prevent future knowledge graphs from fading like every previous iteration apparently has.

VIII. Data Findings

In this section, we present our findings related to RQ1: By which means, such as knowledge representations and schemas, do aerospace engineers externalize, such as formalize and visualize, knowledge? Here, we discuss the items classified in our knowledge graph as data models, data items, and data formats.

A. Quantitative Study

Totally, 89 data formats were mentioned in 46 papers, 47 coming from the literature review and 37 from the surveys. The most commonly mentioned data formats are PDF (32 mentions), VRML (10) and RDS (6). We identified Wikidata URIs for 61 (69%) of the formats. Most knowledge not exchanged in PDF or Office format is based on one of these two.

Regarding data items, 81 instances were identified over all papers and 44 of them (54%) have Wikidata URIs. In a single paper, the number of data item mentions varies from one (two papers) to 14 [47]. The most common data items are description (144 mentions), database (138), and material (108). Description, form, document, and database have 21, 20, 19, and 17 associated processes respectively, while eight data items are associated only with one process each. Regarding sources, 62 data items come from the literature review and 14 from curation.

Finally, we identified 70 data models mentioned over 142 papers, 63 of them come from the literature review, four from the expert surveys, and one from curation. The number of data model mentions in a paper ranges from one (64 papers) to seven [30]. The three most common data models are EXPRESS (117 mentions), Unified Modeling Language (UML) (52) and SPARQL (26) with seven data models mentioned only once. Three data models, EXPRESS, XML, and IDEF1X are associated with a software instance, where EXPRESS has the most number of connections, three. Only 39 data models were identified with Wikidata URIs, which makes 56%.

B. Qualitative Study

Knowledge-based aerospace engineering employs various data models, languages, and ontologies. Qualitative study showed, XML is the most commonly used modeling language, as described in [Table 5](#). Other widely adopted options include UML or SysML, RDF and OWL sometimes enhanced with Semantic Web Rule Language (SWRL). Studies utilizing semantic modeling with OWL predominantly rely on the Protégé tool. Additionally, a few works make use of MS Access [52] and MATLAB [69].

²⁸https://en.wikipedia.org/wiki/Glossary_of_aerospace_engineering

Table 5 Languages used for data modeling

Language	Papers
XML	[71] [65] [56] [72] [73] [74] [75]
UML	[76] [77] [78] [4] [79]
SysML	[80] [68]
RDF	[42] [81] [59] [82] [83]
OWL	[84] [85] [86] [87]
SWRL	[88] [89] [90]

Moreover, researchers develop specific data models implemented with the languages mentioned above. These include bond graphs [71], MDO Framework for Analysis and Design (MDO FAD) [56], CPACS [72] and CMDOWS [73] represented with XML; Digital Thread for Airplane Design (DTAD) [59] and Codex [83] represented with RDF; Standard for the Exchange of Product model data (STEP)²⁹ formulated with the EXPRESS language; KADMOS [73] and MDAO Workflow Design Accelerator (MDAx) [91] implemented in Python; Rationale-based Design Explanation (RaDEX) [87] modeled with OWL; Ezhilarasu and Jennions [92] develop Adaptive Neuro-Fuzzy Inference System (ANFIS). Additionally, well-established data models are used. Thus, Roelofs and Vos [89] utilize Basic Formal Ontology (BFO) and WordNet is utilized in [93] and [94].

Ontologies related to the aerospace domain include the NASA Taxonomy³⁰, XCALIBR [95], the Aerospace Ontology [96], Air Traffic Management Ontology (ATMONTO) [97], Airspace System Ontology [98], Avionics Analytics Ontology [99], Orbital Space Environment Domain Ontology [43], Orbital Debris Ontology [100], and Space Situational Awareness Ontology [101]. Additionally, ontological modeling without specific naming is described in [102], [28], [103], [104], [105], [106], [107], and [108].

C. Gap Analysis

Standards For interoperability, knowledge-based aerospace engineering requires a common language. While we were able to identify 70 data models in the literature for our knowledge graph, we could not identify a single one that is widely adapted throughout the domain. Even capable models rarely find applications outside their originator's publications, like DLR for CPACS or TU Delft for CMDOWS. This is further substantiated by recent works such as Tikayat Ray et al. [109], creating a glossary from the scarce aerospace resources publicly available. While this work aims to unify aerospace terminology openly available and share its software just as such, the glossary itself is once again unavailable. All previous attempts are either no longer available or never were due to organizational boundaries. While tailored knowledge is highly proprietary, a general ontology does not conflict with system boundaries. Even proprietary knowledge can still be available “open by default” within these boundaries, as is already the default in organizations

²⁹<https://www.iso.org/obp/ui/#iso:std:iso:10303:-1:ed-3:v1:en>

³⁰<https://github.com/nasa/dictionaries>

such as NASA and Siemens. If standards are established by these organizations, they will facilitate interoperability and reusability, especially if knowledge facilitators such as the AIAA or the Object Management Group (OMG) support them.

Semantics While having a shared glossary is commendable, truly interoperable knowledge engineering requires crafting a terminology by hierarchical structuring these terms, an ontology by separating entities into concepts and individuals, and a connected knowledge graph using defined properties. A higher level of interpretability is supported by systems utilizing RDF, OWL, and especially SWRL language. However, XML-based models, such as CPACS, are prevalent in the aerospace domain. Though most of them ensure interoperability, they provide primarily human-, not machine-interpretability. This is a common issue throughout aerospace software and data, being mainly used to exchange knowledge between individuals, not larger, machine-supported systems.

IX. Discussion and Future Work

We will briefly discuss general aspects regarding [Knowledge-Based Aerospace Engineering](#), rounded up with [Perspectives on Literature Reviews](#) in the domain.

A. Knowledge-Based Aerospace Engineering

1. Aerospace Knowledge Engineers

We identified a divide between Aerospace Engineers and Knowledge Engineers. While both collaborate towards Knowledge-Based Aerospace Engineering, their goals are not necessarily aligned: One primarily solves a current issue, the other is focused on aiding the solution of future instances of this current issue. The first prefers a practical solution, and the latter a theoretically interoperable one. While many works have highlighted the difficulty of approaching the aerospace domain from a knowledge management standpoint, a better approach seems to be to fundamentally think of both of them in a coupled system. If aerospace engineers work in a team with dedicated knowledge engineers, the domain engineers can focus on the current task, while the knowledge engineers interfaces to previous and future use cases. There are several works that already suggested or implemented this, indicating this to be a very promising composition for future Knowledge-Based Aerospace Engineers.

2. Knowledge-Based Practices and the V-Model

The study of knowledge-based practices positioned along the V-model model phases from [section VI](#) gave some initial insight into where tasks, software, and tools are used. However, more responses to the survey are really needed to draw proper conclusions on this particular part of the presented study. Nevertheless, the survey shows that the classification of processes in the created ontology from the literature survey may be supported by such survey responses.

Also, the survey was distributed to a variety of aerospace engineers but not all might have an understanding of what knowledge-based practices imply. Consequently, a future iteration of the survey would include more descriptions and examples of this. This would then add to more accurate responses to the survey and corresponding entries in terms of positioning along the V-model phases. The next survey would also be targeted at a wider group of aerospace engineers in order to gain more diverse input on knowledge-based practices.

The classification of the processes in the ontology was done manually for this work. However, this can be a challenging task due to the inherent complexity and ambiguity of knowledge-based practices, especially when expert input is not available. Without domain expertise, there is a risk of misclassification or oversimplification, which could lead to inaccuracies in the ontology representation. This issue becomes even more pronounced as additional future survey responses may be integrated, increasing the diversity and volume of entries that must be accurately categorized. A possible future venture would be to investigate the use of, for example, a Large Language Model (LLM) as an effort to automate the classification process and thereby add to the overall scalability of the study. This could also enable quick additional classifications according to other frameworks than the V-model in the same ontology. Such a study could potentially add further insights into knowledge-based aerospace engineering practices in general and thereby contribute supplementary perspectives to the answers to our research questions.

B. Perspectives on Literature Reviews

1. Lessons Learned

This work is based on the SWARM-SLR and tested the applicability of several approaches to facilitate a knowledge-based aerospace engineering literature review. We sourced 245 relevant documents out of a set of over 4000 using the established workflow, read over 150 full papers, dissertations, and articles to craft a preliminary ontology, and statistically evaluated all 1000 available PDFs. Aligning data models and terminology even among the authors of this work was a significant endeavor, but eventually resulted in several lessons learned: The SWARM-SLR workflow [11] is sufficiently capable until including Task 5. Starting with Task 6, specifically with requirement 51. *annotate collaboratively*, several lessons were learned:

- 1) It is highly advised to **setup a WebProtégé instance** and spend a few minutes familiarizing all involved researchers with the software and establishing simple conventions.

Initially appealing approaches, such as annotating inside a PDF reader, in Obsidian, Zotero, a CSV, or even Google Sheets, are no replacement for the semantic disambiguation required later on, including entity linking, synonym, and polyonym handling, etc. The benefits of proper up-front knowledge management are obvious even to this work, where having started with this work could have easily saved dozens of hours of untangling 700+ instances.

2) **Integrating Wikidata as early as possible** is strongly recommended.

Wikidata is an invaluable source of structured information and helps to disambiguate concepts and individuals.

Even initiating an instance such as [aerospace.wikibase.cloud](#) could be beneficial, setting a starting point for future collaboration while alleviating the pressure of interacting with an established knowledge community.

3) **Expert surveys are not efficient semantic data sources**, but they are helpful to differentiate between the sanitized environment of a publication and everyday problems.

Responses require large amounts of manual post-processing and provide the participants little in return, leading to low response rate and immediate drop-off after initial distribution. A dynamic approach like collaborative editing in a Wikibase or the ORKG might be much more rewarding.

2. Reuse and Extend This Work

While this work provides substantial work towards semantifying an interoperable aerospace engineering knowledge graph, it is still just a first step. This initial momentum should lower the barrier to entry for the following endeavors:

- 1) **Establish a common taxonomy in your organization.** Taking this work and examples like the NASA taxonomy as a starting point, this step should be straightforward to implement.
- 2) **Annotate your contributions.** It is very likely that valuable knowledge was missed in this literature review, ranging from unavailable documents to simply unidentified articles outside the search scope. To ensure your contributions partake in future overviews, constructing their FAIR representations, ideally already at inception, is of high importance.
- 3) **Invest in manual knowledge management**, since as highlighted by many authors, an equal, if not larger amount of work is done down the line either way when future work attempts to pick up where the current left off. Even individual engineers may save time by iteratively building upon and reusing their own knowledge base. This will also address the issue of accuracy highlighted in this work, not only clearing up false positives but also providing previously missed true positives.

We also encourage systemic change:

- 1) **Establish a common standard for aerospace engineering data models.** Several candidates are available, none are currently perfect, but likely, some are sufficient.
- 2) **Take up centralized knowledge curation** in the aerospace Wikibase or the ORKG observatory. Since this literature review has surfaced the phrasing “XY was a database for aerospace engineering knowledge” or similar several times, it is highly recommended to steer close to established giants like Wikidata before creating another knowledge base and waiting for the corresponding project funds to dry out.

Despite the wealth of results, findings, and recommendations, this work has only scratched the surface of what is still required in knowledge-based aerospace engineering. Our ontology is hundreds of entries long, but facing the

scale required to adequately represent the domain, it is merely a first step, requiring expansion in width, depth, and interconnectivity. Our paper representations in the ORKG are even greater in number, but they lack the hours of curation invested in the items they display and require individuals to collaboratively scale the literature flood. This review concludes with an extensive, but inherently limited glance on which knowledge aerospace engineering can rely, but it is up to the domain to build upon the provided foundation.

X. Conclusion

In this Systematic Literature Review (SLR), we focused ourselves on Knowledge-Based Engineering (KBE) practices in the aerospace domain. Compared to related reviews, ours features a software-assisted methodology and follows FAIR principles, providing reusable and machine-readable artifacts. Namely, our results include a Zotero library, a Zenodo repository, and a knowledge graph with identified papers, terms, and relations between them. The latter is also represented in the Open Research Knowledge Graph (ORKG) for open public access. Contentwise, our main contribution is identifying through literature review, expert surveys, and manual curation more than 700 specific terms, or items, related to aerospace knowledge-based engineering. We classified them into three main categories: processes, software, and data. For papers and items of each category, we conducted both qualitative and quantitative studies, discovering gaps in stat-of-the-art.

This systematic literature review on knowledge-based aerospace engineering highlights the evolving landscape of how explicit knowledge is formalized, utilized, and exchanged in this highly complex domain. Despite the aerospace industry's longstanding tradition of applying KBE methodologies, significant gaps in interoperability, standardization, and adoption still persist. These facts, in connection with deficiencies in compliance with the FAIR principles and in addition to data protection topics and the necessary efforts for creating and maintaining KBE systems hamper a wide and standardized application of this approach. While the present systematic literature review uncovered numerous suitable tools and frameworks, such as CMDOWS and CPACS, their widespread adoption remains limited. Challenges in cross-disciplinary collaboration, proprietary knowledge sharing, and a lack of unified data standards hinder the full realization of KBE's potential. Moreover, the divide between domain-specific engineers and knowledge engineers further stresses the need for collaborative frameworks that bridge practical and theoretical knowledge applications. Our findings demonstrate the urgent need for standardized, interoperable ontologies tailored to aerospace engineering. Establishing such standards can drive innovation, streamline workflows, and facilitate sustainability goals. Future efforts should focus on fostering open-access knowledge repositories, enhancing semantic data models, and integrating knowledge engineering practices more effectively into engineering workflows.

In conclusion, this review sets a precedent for structured, semantic-based approaches to managing aerospace engineering knowledge. By advancing these principles, research, and industry can achieve more efficient design processes, enhanced collaboration, and a stronger commitment to sustainable aviation.

Appendix

Listing 3 RQ2: By which means, such as knowledge bases and services, do aerospace engineers utilize, such as organize and interface with, explicit knowledge?

### knowledge base	standard :: 3	leverage :: 6
knowledge base :: 10	information :: 3	analyze :: 6
knowledge representation :: 8	network :: 3	process :: 6
repository :: 8	format :: 2	query :: 4
structure :: 6	wiki :: 2	
expert system :: 6	data :: 1	### Formats
software :: 5	platform :: 1	API :: 5
service :: 5		protege :: 3
system :: 5	### utilize	Neo4J :: 3
method :: 5	utilize :: 10	XML :: 2
archive :: 5	interface :: 10	JSON :: 2
catalog :: 5	manage :: 8	RDF :: 2
knowledge :: 5	organize :: 8	OWL :: 2
knowledge management :: 8	access :: 7	
tool :: 3	integrate :: 7	

Listing 4 RQ2: Query with 806 Google Scholar results

```
"aerospace engineering" OR "aviation engineering" "knowledge base" OR  
"knowledge representation" OR "knowledge repository" OR "knowledge structure" utilize OR  
interface OR manage OR organize OR access OR integrate API OR prot\'eg\'e OR Neo4J OR JSON
```

Listing 5 RQ3: By which means, such as tools and methods, do aerospace engineers exchange, such as transfer and distribute, explicit knowledge?

### knowledge representation	semantic :: 4	mapping :: 2
knowledge representation :: 10	information :: 3	wiki :: 2
Knowledge graph :: 10	repository :: 3	technique :: 2
Ontology :: 10	documentation :: 3	classification :: 2
information representation :: 8	graphic :: 3	categorization :: 2
Taxonomy :: 8	index :: 3	data :: 1
Semantic Web :: 8	formal :: 2	map :: 1
knowledge engineering :: 7	standard :: 2	computer-aided design :: 1
schema :: 7	format :: 2	CAD :: 1
expert system :: 6	structure :: 2	computer-aided design :: 1
knowledge :: 5	graph :: 2	virtual reality :: 1
data representation :: 4	plot :: 2	VR :: 1

simulation :: 1	articulate :: 3	RDF :: 4
platform :: 1	express :: 3	OWL :: 4
report :: 1	manifest :: 3	XML :: 4
	embody :: 2	CPACS :: 4
### externalize	model :: 2	SysML :: 4
externalize :: 10	document :: 1	UML :: 4
represent :: 10	codify :: 1	JSON-LD :: 3
formalize :: 8	transfer :: 2	JSON :: 2
visualize :: 8	explicit knowledge :: 8	YAML :: 2
store :: 4		GraphML :: 1
structured :: 4	### Formats	XSD :: 4

Listing 6 RQ3: Query with 817 Google Scholar results

```
"aerospace engineering" OR "aviation engineering" "explicit knowledge" OR  
"knowledge dissemination" transfer OR share OR exchange discipline OR domain
```

Use of AI tools declaration

During the preparation of this work the author(s) used Grammarly, LanguageTool, DeepL, and ChatGPT in order to check grammar, refine language, summarize and improve clarity. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Funding Sources

The present study was co-funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2163/1 Sustainable and Energy Efficient Aviation Project ID 390881007, as well as the NFDI4Ing project funded by the German Research Foundation (project number 442146713) and NFDI4DataScience (project number 460234259).

Acknowledgments

We would like to acknowledge the valuable discussions and feedback from the SE²A B4.2. group, particularly Susanna Baars and Ihar Antonau. This work was conducted using Protégé.

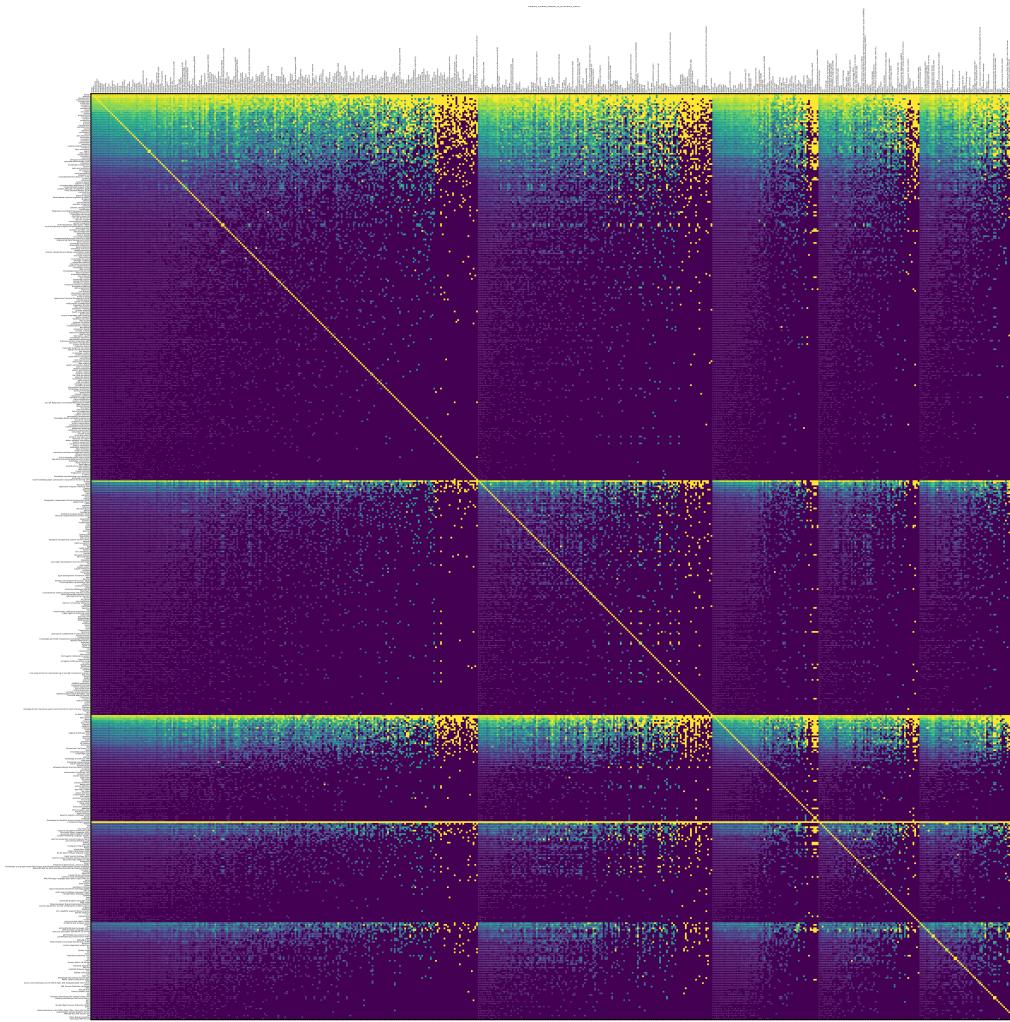


Fig. 15 Instance-instance relative-co-occurrence Matrix. From purple (0) to yellow (1), each cell represents the likelihood of a given item (column) occurring within 100 characters in the same document as a given other item (row). Instances are Sorted as processes, software, data items, data models, data formats, and within that by total occurrence. The diagonal is 1, since the rows and columns are identical, and every word co-occurs with itself 100 % of the time. The horizontal yellow stripes indicate that an item is so common in every document that it practically co-occurs with everything. The further down a yellow cell is (within its category block), the more meaningful it is, since it means a rare instance co-appears with another rare instance. Equally, if Both i, j and j, i are yellow, both items i and j occur when the other does, indicating a bi-directional, stronger relationship.

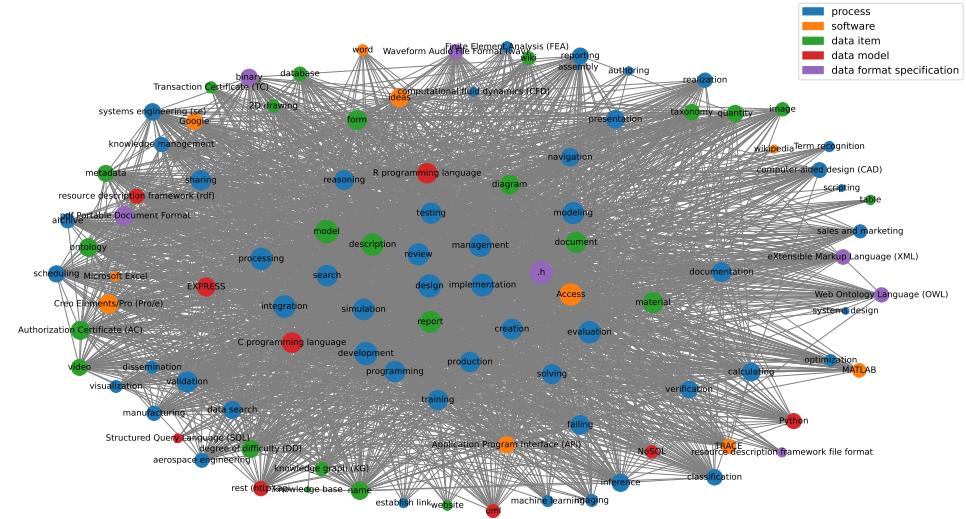


Fig. 16 Instance-instance co-occurrence Graph. Taking after Figure 15, it represents the strongest relationships between instances, color-coded by their class and sized according to their occurrence. It was primarily an indicator of false positives, where it is visible that software like “UG”, an acronym identified in the literature, but still unresolved to its meaning, is far too prevalent. Several other issues could be identified and resolved using this intermediate validation method, but as evident, some others remain.

Biographies



Fig. 17 Tim Wittenborg

Tim Wittenborg is a PhD Student at the L3S Research Center, working in the SE²A - Sustainable and Energy-Efficient Aviation Cluster of Excellence and on the Open Research Knowledge Graph. He has a background in Mechanical Engineering and Computer Science, completing his Master degree at the University of Paderborn in 2022 and spending 10 months as a mechanical engineering PhD student in the information management group of the product creation department. Since progressing to Leibniz University Hannover in January 2023, he studies under professor Sören Auer, focussed on developing knowledge infrastructure and communities. He has published a paper on

such infrastructure supporting the systematic literature process and founded a non-profit organization to support the establishment of a knowledge community for distributed scientific communication.

Ildar Baimuratov earned a Master's degree in Logic from St. Petersburg University in 2014, followed by a PhD in Computer Science from ITMO University in 2020. His doctoral thesis was focused on clustering hyperparameter tuning, while simultaneously working on projects involving machine learning and knowledge engineering. Since 2022, Ildar has been a postdoctoral researcher at the joint lab of the L3S Research Center and TIB - Leibniz Information Centre for Science and Technology, as well as the Sustainable and Energy-Efficient Aviation (SE²A) Cluster of Excellence. His research emphasizes advancing knowledge extraction, representation, and utilization in scholarly contexts — such as automating peer review — and engineering domains, including building construction and aerospace engineering.



Fig. 18 Ildar Baimuratov



Fig. 19 Ludvig Knöös Franzén

Ludvig Knöös Franzén is an assistant professor in aircraft design at the division of Fluid- and Mechatronic Systems in the department of management and engineering at Linköping University (LiU) in Sweden. He holds a Bachelor's degree in Mechanical Engineering, as well as a Master's degree in Aeronautical Engineering from LiU. In 2023, Ludvig also defended his Ph.D. thesis at LiU with the title "A System of Systems View in Early Product Development – An Ontology-based Approach". Ludvig's research is primarily focused on aircraft design and design space explorations from a System-of-Systems perspective. He is currently a work package leader in the COLOSSUS (Collaborative System of Systems Exploration of Aviation Products, Services & Business Models) project funded by the European Union Horizon Europe program under grant agreement

No. 101097120. Furthermore, Ludvig received a best Ph.D. project award in 2020 on the Doctoral Consortium on Knowledge Discovery, Knowledge Engineering and Knowledge Management (DC3K/IC3K).

Ingo Staack is professor for aircraft design at the Technische Universität Braunschweig. With a background in Mechanical Engineering at the TU Munich. He obtained his PhD in Aerospace Engineering from Linköping University, Sweden in 2016. His research encompasses a wide range of engineering design topics, ranging from early conceptual design to prototyping of flying demonstrators, extending aircraft design through systems engineering (SE) and system-of-systems engineering (SoSE) concepts for innovative and complex product development. His institute is a member of the Aeronautics Research Centre Niedersachsen (NFL) and the Sustainable and Energy-Efficient Aviation (SE²A) Cluster of Excellence.



Fig. 20 Ingo Staack



Fig. 21 Ulrich Römer

Ulrich Römer is associate professor for Modeling of Complex Systems under Uncertainty in Mobility at TU Braunschweig, Germany. He holds a double degree in Electrical Engineering and Information Technology from TU Darmstadt, Germany, and General Engineering from École Centrale de Lyon, France. His PhD, completed in 2015 at TU Darmstadt, focused on uncertainty quantification in computational magnetics. He is a member of the Aeronautics Research Centre Niedersachsen and founding PI as well as board member of the Sustainable and Energy-Efficient Aviation (SE²A) Cluster of Excellence. His research focuses on computational and data-driven modeling and

design, with an emphasis on uncertainty quantification.

Sören Auer is Director of TIB - Leibniz Information Centre for Science and Technology and professor for Data Science and Digital Libraries at Leibniz University of Hannover. Prof. Auer has made important contributions to semantic technologies, knowledge engineering, and information systems. He is the author (resp. co-author) of over 200 peer-reviewed scientific publications. He has received several awards, including an ERC Consolidator Grant from the European Research Council, a SWSA ten-year award, and the ESWC 7-year Best Paper Award. He is a co-founder of high-potential research and community projects such as the Wikipedia semantification project DBpedia, the Open Research Knowledge Graph ORKG.org, and the innovative technology start-up eccenca.com.



Fig. 22 Sören Auer

References

- [1] Stjepandić, J., Verhagen, W. J. C., Liese, H., and Bermell-Garcia, P., *Knowledge-Based Engineering*, Springer International Publishing, Cham, 2015, pp. 255–286. https://doi.org/10.1007/978-3-319-13776-6_10, URL https://doi.org/10.1007/978-3-319-13776-6_10.
- [2] La Rocca, G., and van Tooren, M. J., “A Knowledge Based Engineering Approach to Support Automatic Generation of FE Models in Aircraft Design,” *45th AIAA Aerospace Sciences Meeting and Exhibit, Aerospace Sciences Meetings*, Reno, Nevada, 2007.
- [3] Boden, B., Padilha, M. H., Burschyk, T., Cabaleiro de la Hoz, C., Moerland, E., and Fioriti, M., “Automating the Verification of Geometric Requirements for Aircraft Fuel Systems using Knowledge-Based Engineering,” *34th Congress of the International Council of the Aeronautical Sciences Florence, Italy*, 2024.
- [4] Van der Laan, A. H., “Knowledge based engineering support for aircraft component design,” Ph.D. thesis, Faculty of Aerospace Engineering, Delft University of Technology, Netherland, 2008.
- [5] Staack, I., and Krus, P., “Integration of On-Board Power Systems Simulation in Conceptual Aircraft Design,” *Proceedings of the 4th CEAS European Air & Space Conference*, Council of the European Aerospace Societies (CEAS), Linköping University Electronic Press, 2013, 2013, p. 709. URL <https://login.e.bibl.liu.se/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edsswe&AN=edsswe.oai.DiVA.org.liu.103197&site=eds-live&scope=site>.
- [6] Reddy, E. J., Sridhar, C., and Rangadu, V., “Knowledge Based Engineering: Notion, Approaches and Future Trends,” *American Journal of Intelligent Systems*, Vol. 5, 2015, pp. 1–17. URL <https://www.semanticscholar.org/paper/01a20390b48152ec02dc0bf29b109aded7e0d9a5>.
- [7] Quintana-Amate, S., Bermell-Garcia, P., and Tiwari, A., “Transforming expertise into Knowledge-Based Engineering tools: A survey of knowledge sourcing in the context of engineering design,” *Knowledge-Based Systems*, Vol. 84, 2015, pp. 89–97. <https://doi.org/https://doi.org/10.1016/j.knosys.2015.04.002>, URL <https://www.sciencedirect.com/science/article/pii/S0950705115001409>.
- [8] Schlichting, G. S., Boswell, W., Shaver, A. T., Evans, J. P., Pinon-Fischer, O. J., and Mavris, D. N., “Knowledge Capture and Management within an Academic Research Laboratory: a Case Study,” *AIAA SCITECH 2024 Forum*, 2024, p. 1134.
- [9] Matt, D. T., Pedrini, G., Bonfanti, A., and Orzes, G., “Industrial digitalization. A systematic literature review and research agenda,” *European Management Journal*, Vol. 41, No. 1, 2023, pp. 47–78. <https://doi.org/10.1016/j.emj.2022.01.001>, URL <https://www.sciencedirect.com/science/article/pii/S0263237322000019>.
- [10] Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., et al., “The FAIR Guiding Principles for scientific data management and stewardship,” *Scientific data*, Vol. 3, No. 1, 2016, pp. 1–9.

- [11] Wittenborg, T., Karras, O., and Auer, S., “SWARM-SLR – Streamlined Workflow Automation for Machine-actionable Systematic Literature Reviews,” , 2024. URL <http://arxiv.org/abs/2407.18657>.
- [12] Torres, W., Van Den Brand, M. G. J., and Serebrenik, A., “A systematic literature review of cross-domain model consistency checking by model management tools,” *Software and Systems Modeling*, Vol. 20, No. 3, 2021, pp. 897–916. <https://doi.org/10.1007/s10270-020-00834-1>, URL <https://link.springer.com/10.1007/s10270-020-00834-1>.
- [13] Viana, F., Simpson, T., Balabanov, V., and Toropov, V., “Metamodeling in Multidisciplinary Design Optimization: How Far Have We Really Come?” 2014. <https://doi.org/10.2514/1.J052375>, URL <https://www.semanticscholar.org/paper/2fda5d8df4c4cba99b2580c075acaf33a2bbbbd8b>.
- [14] Umland, N., Winkler, K., and Inkermann, D., “Multidisciplinary Design Automation of Electric Motors—Systematic Literature Review and Methodological Framework,” *Energies*, Vol. 16, No. 20, 2023, p. 7070. Publisher: MDPI.
- [15] Kritzinger, W., Karner, M., Traar, G., Henjes, J., and Sihn, W., “Digital Twin in manufacturing: A categorical literature review and classification,” *IFAC-PapersOnLine*, Vol. 51, No. 11, 2018, pp. 1016–1022. <https://doi.org/10.1016/j.ifacol.2018.08.474>, URL <https://www.sciencedirect.com/science/article/pii/S2405896318316021>.
- [16] Enders, M. R., and Hoßbach, N., “Dimensions of digital twin applications - A literature review,” 2019. URL <https://publica.fraunhofer.de/handle/publica/410694>.
- [17] Lim, K. Y. H., Zheng, P., and Chen, C.-H., “A state-of-the-art survey of Digital Twin: techniques, engineering product lifecycle management and business innovation perspectives,” *Journal of Intelligent Manufacturing*, Vol. 31, 2020, pp. 1313–1337. URL https://idp.springer.com/authorize/casa?redirect_uri=https://link.springer.com/article/10.1007/s10845-019-01512-w&casa_token=cmE7J2P9D3wAAAAA:_AP0kziL3dltWHdEy1swNU_T4l9_mvai9T9bj4ezOfIs7g_C-Zrh8j-HlhKVBGOTs6rOrfsRJ8qOjI, publisher: Springer.
- [18] Rathore, M. M., Shah, S. A., Shukla, D., Bentafat, E., and Bakiras, S., “The Role of AI, Machine Learning, and Big Data in Digital Twinning: A Systematic Literature Review, Challenges, and Opportunities,” *IEEE Access*, Vol. 9, 2021, pp. 32030–32052. <https://doi.org/10.1109/ACCESS.2021.3060863>, URL <https://ieeexplore.ieee.org/document/9359733>, conference Name: IEEE Access.
- [19] Kosacka-Olejnik, M., Kostrzewski, M., Marczevska, M., Mrówczyńska, B., and Pawlewski, P., “How digital twin concept supports internal transport systems?—Literature review,” *Energies*, Vol. 14, No. 16, 2021, p. 4919. URL <https://www.mdpi.com/1996-1073/14/16/4919>, publisher: MDPI.
- [20] Xames, M. D., and Topcu, T. G., “A Systematic Literature Review of Digital Twin Research for Healthcare Systems: Research Trends, Gaps, and Realization Challenges,” *IEEE Access*, 2024. URL <https://ieeexplore.ieee.org/abstract/document/10379674/>, publisher: IEEE.
- [21] Hassan, K., Thakur, A. K., Singh, G., Singh, J., Gupta, L. R., and Singh, R., “Application of Artificial Intelligence in Aerospace Engineering and Its Future Directions: A Systematic Quantitative Literature Review,” *Archives of Computational*

- Methods in Engineering*, Vol. 31, No. 7, 2024, pp. 4031–4086. <https://doi.org/10.1007/s11831-024-10105-7>, URL <https://doi.org/10.1007/s11831-024-10105-7>.
- [22] Balaid, A., Abd Rozan, M. Z., Hikmi, S. N., and Memon, J., “Knowledge maps: A systematic literature review and directions for future research,” *International Journal of Information Management*, Vol. 36, No. 3, 2016, pp. 451–475. <https://doi.org/10.1016/j.ijinfomgt.2016.02.005>, URL <https://www.sciencedirect.com/science/article/pii/S026840121600098>.
- [23] Schlenoff, C., Hong, T., Liu, C., Eastman, R., and Foufou, S., “A literature review of sensor ontologies for manufacturing applications: 2013 11th IEEE International Symposium on Robotic and Sensors Environments, ROSE 2013,” *ROSE 2013 - 2013 IEEE International Symposium on Robotic and Sensors Environments, Proceedings*, 2013, pp. 96–101. <https://doi.org/10.1109/ROSE.2013.6698425>, URL <http://www.scopus.com/inward/record.url?scp=84893266184&partnerID=8YFLogxK>.
- [24] Cao, H., and Folan, P., “Product life cycle: the evolution of a paradigm and literature review from 1950–2009,” *Production Planning & Control*, Vol. 23, No. 8, 2012, pp. 641–662. <https://doi.org/10.1080/09537287.2011.577460>, URL <https://www.tandfonline.com/doi/full/10.1080/09537287.2011.577460>.
- [25] Kadiri, S. E., and Kiritsis, D., “Ontologies in the context of product lifecycle management: state of the art literature review,” *International Journal of Production Research*, Vol. 53, 2015, pp. 5657 – 5668. <https://doi.org/10.1080/00207543.2015.1052155>, URL <https://www.semanticscholar.org/paper/186f8b0040ee75ed0b75e84fd16206a31181ef37>.
- [26] Rashid, M., Anwar, M. W., and Khan, A. M., “Toward the tools selection in model based system engineering for embedded systems—A systematic literature review,” *Journal of Systems and Software*, Vol. 106, 2015, pp. 150–163. <https://doi.org/10.1016/j.jss.2015.04.089>, URL <https://www.sciencedirect.com/science/article/pii/S016412121500103X>.
- [27] Kwakye, A. D., Jennions, I. K., and Ezhilarasu, C. M., “Platform health management for aircraft maintenance – a review,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 238, No. 3, 2024, pp. 267–283. <https://doi.org/10.1177/09544100231219736>, URL <http://journals.sagepub.com/doi/10.1177/09544100231219736>.
- [28] Wang, Z., Zheng, P., Lee, C.-H., and Chen, C.-H., “Knowledge representation and reasoning methods in the concept development of product design: a state-of-the-art review,” *Transdisciplinary Engineering Methods for Social Innovation of Industry 4.0*, 2018, pp. 478–487. URL <https://ebooks.iospress.nl/volumearticle/49830>, publisher: IOS Press.
- [29] Cerchione, R., and Esposito, E., “Using knowledge management systems: A taxonomy of SME strategies,” *International Journal of Information Management*, Vol. 37, No. 1, Part B, 2017, pp. 1551–1562. <https://doi.org/10.1016/j.ijinfomgt.2016.10.007>, URL <https://www.sciencedirect.com/science/article/pii/S0268401216300597>.
- [30] Procko, T., and Ochoa, O., “Leveraging Linked Data for Knowledge Management: A Proposal for the Aerospace Industry,” Available at SSRN 4703237, 2022. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4703237.
- [31] Yao, Y., Patterson, E. A., and Taylor, R. J., “The Influence of Digital Technologies on Knowledge Management in Engineering: A Systematic Literature Review,” *IEEE Transactions on Knowledge and Data Engineering*, 2023. Publisher: IEEE.

- [32] Oprea, A., Eek, M., Staack, I., Ölvander, J., and Gavel, H., “Aspects of flight simulation development in relation to scientific software: a literature review,” *AIAA AVIATION 2021 FORUM*, AIAA AVIATION Forum, American Institute of Aeronautics and Astronautics, 2021. <https://doi.org/10.2514/6.2021-2998>, URL <https://arc.aiaa.org/doi/10.2514/6.2021-2998>.
- [33] Stark, J., *Product Lifecycle Management (Volume 6) - Increasing the Value of PLM with Innovative New Technologies*, Decision Engineering, Springer Nature, Geneva, Switzerland, 2024.
- [34] IBM, Chrystal R. China, “Digital Twin vs. Digital Thread: Two Complementary Ways to Digitally Replicate Assets,” , 2023. URL <https://www.ibm.com/think/topics/digital-thread-vs-digital-twin>, Accessed on: 2024-12-13.
- [35] *Systems Engineering Handbook*, 4th ed., INCOSE, Wiley, San Diego, CA, USA, 2015, pp. 33–36.
- [36] *Systems Engineering Guide - Collected Wisdom From MITRE's Systems Engineering Experts*, The MITRE Corporation, Burlington Road, Bedford, 2014.
- [37] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z., “Dbpedia: A nucleus for a web of open data,” *international semantic web conference*, Springer, 2007, pp. 722–735.
- [38] Vrandečić, D., and Krötzsch, M., “Wikidata: A Free Collaborative Knowledgebase,” *Commun. ACM*, Vol. 57, No. 10, 2014, p. 78–85. <https://doi.org/10.1145/2629489>, URL <https://doi.org/10.1145/2629489>.
- [39] Berners-Lee, T., Hendler, J., and Lassila, O., “The Semantic Web,” *ScientificAmerican.com*, 2001.
- [40] Meng, X., Jing, B., Wang, S., Pan, J., Huang, Y., and Jiao, X., “Fault Knowledge Graph Construction and Platform Development for Aircraft PHM,” *Sensors*, Vol. 24, No. 1, 2023, p. 231. URL <https://www.mdpi.com/1424-8220/24/1/231>, publisher: MDPI.
- [41] Franzén, L. K., *A System of Systems View in Early Product Development - An Ontology-Based Approach*, Linköping Studies in Science and Technology, Dissertation No. 2317, LiU-Tryck, Linköping University, 2023.
- [42] Dadzie, A.-S., Bhagdev, R., Chakravarthy, A., Chapman, S., Iria, J., Lanfranchi, V., Magalhães, J., Petrelli, D., and Ciravegna, F., “Applying semantic web technologies to knowledge sharing in aerospace engineering,” *Journal of Intelligent Manufacturing*, Vol. 20, No. 5, 2009, pp. 611–623. <https://doi.org/10.1007/s10845-008-0141-1>, URL <http://link.springer.com/10.1007/s10845-008-0141-1>.
- [43] Rovetto, R. J., “An ontology for satellite databases,” *Earth Science Informatics*, Vol. 10, 2017, pp. 417–427.
- [44] Cook, J., Nuccitelli, D., Green, S. A., Richardson, M., Winkler, B., Painting, R., Way, R., Jacobs, P., and Skuce, A., “Quantifying the consensus on anthropogenic global warming in the scientific literature,” *Environmental Research Letters*, Vol. 8, No. 2, 2013, p. 024024. <https://doi.org/10.1088/1748-9326/8/2/024024>, URL <https://dx.doi.org/10.1088/1748-9326/8/2/024024>, publisher: IOP Publishing.
- [45] Sanya, I., Shehab, E., and Lowe, D., “Challenges in semantic knowledge management for aerospace design engineering,” *DS 68-6: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through*

- Engineering Design, Vol. 6: Design Information and Knowledge, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011*, 2011, pp. 241–248. URL https://www.designsociety.org/publication/30631/challenges_in_semantic_knowledge_management_for_aerospace_design_engineering.
- [46] Sanya, I., and Shehab, E., “An ontology framework for developing platform-independent knowledge-based engineering systems in the aerospace industry,” *International Journal of Production Research*, Vol. 52, No. 20, 2014, pp. 6192–6215. <https://doi.org/10.1080/00207543.2014.919422>, URL <http://www.tandfonline.com/doi/full/10.1080/00207543.2014.919422>.
- [47] Xie, Y., “Application of context aware systems to support knowledge work in the aerospace,” PhD Thesis, University of Bath, 2013. URL https://purehost.bath.ac.uk/ws/portalfiles/portal/187906137/Yifan_EngD_Thesis_FinalVersion.pdf.
- [48] Gonzalez, L., *Complex multidisciplinary system composition for aerospace vehicle conceptual design*, The University of Texas at Arlington, 2016. URL <https://search.proquest.com/openview/a29f17f9cdfb64c7fb124d618c3e01b0/1?pq-origsite=gscholar&cbl=18750>.
- [49] Chudoba, B., *Stability and control of conventional and unconventional aircraft configurations: a generic approach*, BoD–Books on Demand, 2001.
- [50] Huang, X., “A Prototype Computerized Synthesis Methodology for Generic Space Access Vehicle (SAV) Conceptual Designs,” *1st Space Exploration Conference: Continuing the Voyage of Discovery*, Space Exploration Conferences, American Institute of Aeronautics and Astronautics, 2005. <https://doi.org/10.2514/6.2005-2539>, URL <https://arc.aiaa.org/doi/10.2514/6.2005-2539>.
- [51] Coleman, G., “Aircraft Conceptual Design - An Adaptable Parametric Sizing Methodology,” *Mechanical and Aerospace Engineering Dissertations*, 2010. URL https://mavmatrix.uta.edu/mechaerospace_dissertations/15.
- [52] Haney, E., “Data Engineering in Aerospace Systems Design & Forecasting,” PhD Thesis, 2016. URL <https://rc.library.uta.edu/uta-ir/handle/10106/25877>.
- [53] Aigner, B., Van Gent, I., La Rocca, G., Stumpf, E., and Veldhuis, L. L. M., “Graph-based algorithms and data-driven documents for formulation and visualization of large MDO systems,” *CEAS Aeronautical Journal*, Vol. 9, No. 4, 2018, pp. 695–709. <https://doi.org/10.1007/s13272-018-0312-5>, URL <http://link.springer.com/10.1007/s13272-018-0312-5>.
- [54] Baalbergen, E., Vankan, J., Boggero, L., Bussemaker, J., Lefèuvre, T., Beijer, B., Bruggeman, A.-L., and Mandorino, M., “Advancing Cross-Organizational Collaboration in Aircraft Development.”, Jun. 2022. <https://doi.org/10.5281/zenodo.6801277>, URL <https://zenodo.org/record/6801277>, publisher: Zenodo.
- [55] Van Gent, I., La Rocca, G., and Veldhuis, L. L., “Composing MDAO symphonies: graph-based generation and manipulation of large multidisciplinary systems,” *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, American Institute of Aeronautics and Astronautics, Denver, Colorado, 2017. <https://doi.org/10.2514/6.2017-3663>, URL <https://arc.aiaa.org/doi/10.2514/6.2017-3663>.

- [56] Agrawal, G., Parashar, S., English, K. W., and Bloebaum, C. L., “Web-based visualization framework for decision-making in multidisciplinary design optimization,” *45th AIAA/ASME Conference*, Citeseer, 2004, pp. 19–22. URL <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=e94eda1e9ac59fb4202d213685594c903e7910d7>.
- [57] Harvey, D., and Holdsworth, R., “Knowledge management in the aerospace industry,” *IPCC 2005. Proceedings. International Professional Communication Conference, 2005.*, 2005, pp. 237–243. <https://doi.org/10.1109/IPCC.2005.1494182>, URL <https://ieeexplore.ieee.org/document/1494182>, iSSN: 2158-1002.
- [58] Jordan, J., and Lowe, J., “Protecting Strategic Knowledge: Insights from Collaborative Agreements in the Aerospace Sector,” *Technology Analysis & Strategic Management*, Vol. 16, No. 2, 2004, pp. 241–259. <https://doi.org/10.1080/09537320410001682900>, URL <https://www.tandfonline.com/doi/abs/10.1080/09537320410001682900>, publisher: Routledge.
- [59] Herrero, J., “A System Architecture for the Digital Thread in Commercial Airplane Design,” PhD Thesis, Massachusetts Institute of Technology, 2022. URL <https://dspace.mit.edu/handle/1721.1/143229>.
- [60] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S., “Dynamic itemset counting and implication rules for market basket data,” *Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, 1997, pp. 255–264.
- [61] Musen, M. A., and Protégé Team, “The Protégé Project: A Look Back and a Look Forward,” *AI matters*, Vol. 1, No. 4, 2015, pp. 4–12. <https://doi.org/10.1145/2757001.2757003>.
- [62] Li, J., Xiao, M., and Tang, X., “Research on Intelligent Fault Diagnosis of Portable ATE Based on Ontology,” *IOP Conference Series: Materials Science and Engineering*, Vol. 394, IOP Publishing, 2018, p. 032055. URL <https://iopscience.iop.org/article/10.1088/1757-899X/394/3/032055/meta>, issue: 3.
- [63] Adala, A., Tabbane, N., and Tabbane, S., “A framework for automatic web service discovery based on semantics and NLP techniques,” *Advances in Multimedia*, Vol. 2011, 2011, pp. 1–1. URL <https://www.hindawi.com/journals/am/2011/238683/abs/>, publisher: Hindawi Limited London, UK, United Kingdom.
- [64] Flink, J., Mischke, R., Schaffert, K., Schneider, D., and Weinert, A., “Orchestrating tool chains for model-based systems engineering with RCE,” *2022 IEEE Aerospace Conference (AERO)*, IEEE, 2022, pp. 1–9. URL <https://ieeexplore.ieee.org/abstract/document/9843838/>.
- [65] Jackson, E. B., and Hildreth, B. L., “Progress Toward a Format Standard for Flight Dynamics Models,” 2006. URL <https://ntrs.nasa.gov/citations/20060048508>.
- [66] Royal, N. L. R., “Advancing Cross-Organizational Collaboration in Aircraft Development,” 2022. URL <https://reports.nlr.nl/server/api/core/bitstreams/3655ed15-bd79-403a-93d9-9899bfa5bedf/content>.
- [67] Vucinic, D., “Multidisciplinary visualization aspects in European R&D projects,” *ASME International Mechanical Engineering Congress and Exposition*, Vol. 44489, 2010, pp. 143–159. URL <https://asmedigitalcollection.asme.org/IMECE/proceedings-abstract/IMECE2010/143/340064>.

- [68] Böhnke, D., “Data integration in preliminary airplane design,” PhD Thesis, 2009. URL <https://elib.dlr.de/59735/>.
- [69] Herbst, S., “Development of an aircraft design environment using an object-oriented data model in MATLAB,” PhD Thesis, Technische Universität München, 2018. URL <https://mediatum.ub.tum.de/1431402>.
- [70] Krimmelbein, N., and Radespiel, R., “Transition Prediction for Three-Dimensional Configurations,” *MEGADESIGN and MegaOpt - German Initiatives for Aerodynamic Simulation and Optimization in Aircraft Design*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 93–106.
- [71] Vargas-Hernandez, N., Shah, J., and Lacroix, Z., “Development of a computer aided conceptual design tool for complex electromechanical systems,” *Computational Synthesis: From Basic Building Blocks to High Level Functionality, Papers from the 2003 AAAI Symposium Technical Report SS-03-02*, 2003, pp. 255–261. URL <https://cdn.aaai.org/Symposia/Spring/2003/SS-03-02/SS03-02-035.pdf>.
- [72] Nagel, B., Böhnke, D., Gollnick, V., Schmollgruber, P., Rizzi, A., La Rocca, G., and Alonso, J. J., “Communication in aircraft design: Can we establish a common language,” *28th International Congress of the Aeronautical Sciences*, Vol. 201, 2012. URL https://www.icas.org/ICAS_ARCHIVE/ICAS2012/PAPERS/201.PDF, issue: 2.
- [73] van Gent, I., “Agile MDAO systems: a graph-based methodology to enhance collaborative multidisciplinary design,” 2019. URL <https://research.tudelft.nl/en/publications/agile-mdao-systems-a-graph-based-methodology-to-enhance-collabora>.
- [74] Munjulury, V. R. C., “Knowledge based integrated multidisciplinary aircraft conceptual design,” PhD Thesis, Linköping University Electronic Press, 2014. URL <https://www.diva-portal.org/smash/record.jsf?pid=diva2:719775>.
- [75] Pulvermacher, M. K., Brandsma, D. L., and Wilson, J., “A Space Surveillance Ontology Captured in an XML Schema,” *MITRE, Defense Technical Information Center. Online at: https://www.mitre.org/sites/default/files/pdf/space_pulvermacher.pdf (as of 9 April 2017)*, 2000.
- [76] Lu, Z., “Data management in an object-oriented distributed aircraft conceptual design environment,” PhD Thesis, Citeseer, 2005. URL <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=dca1203c612b3e35449e180cb605431b8b627511>.
- [77] Sung, C. S., and Park, S. J., “A component-based product data management system,” *The International Journal of Advanced Manufacturing Technology*, Vol. 33, 2007, pp. 614–626.
- [78] Berquand, A., “Text mining and natural language processing for the early stages of space mission design,” 2021. URL <https://stax.strath.ac.uk/concern/theses/xw42n836q>.
- [79] Oh, Y., Han, S.-h., and Suh, H., “Mapping product structures between CAD and PDM systems using UML,” *Computer-aided design*, Vol. 33, No. 7, 2001, pp. 521–529.
- [80] De Leon, S. S., “Intelligent Data Understanding for Entry, Descent, and Landing, Architecture Analysis,” PhD Thesis, Texas A&M University, 2021. URL <https://search.proquest.com/openview/c19723dd75db58eaa895a2026b5d308b/1?pq-origsite=gscholar&cbl=18750&diss=>.

- [81] Khilwani, N., and Harding, J. A., “Managing corporate memory on the semantic web,” *Journal of Intelligent Manufacturing*, Vol. 27, No. 1, 2016, pp. 101–118. <https://doi.org/10.1007/s10845-013-0865-4>, URL <http://link.springer.com/10.1007/s10845-013-0865-4>, publisher: Springer.
- [82] Bang, H., Virós Martin, A., Prat, A., and Selva, D., “Daphne: An Intelligent Assistant for Architecting Earth Observing Satellite Systems,” *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, American Institute of Aeronautics and Astronautics, Kissimmee, Florida, 2018. <https://doi.org/10.2514/6.2018-1366>, URL <https://arc.aiaa.org/doi/10.2514/6.2018-1366>.
- [83] Padilha, M. H., “Enabling fuel system component placement on the basis of geometric and certification requirements within a semantic knowledge-based engineering framework,” PhD Thesis, Politecnico di Torino, 2023. URL <https://webthesis.biblio.polito.it/28848/>.
- [84] Zhao, H., Sun, Z., and Zhang, H., “Application of knowledge engineering in spacecraft overall design,” *Journal of Physics: Conference Series*, Vol. 1510, IOP Publishing, 2020, p. 012015. URL <https://iopscience.iop.org/article/10.1088/1742-6596/1510/1/012015/meta>, issue: 1.
- [85] Curran, R., Verhagen, W., and Van Tooren, M., “The KNOMAD Methodology for Integration of Multidisciplinary Engineering Knowledge Within Aerospace Production,” *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, American Institute of Aeronautics and Astronautics, Orlando, Florida, 2010, p. 1315. <https://doi.org/10.2514/6.2010-1315>, URL <https://arc.aiaa.org/doi/10.2514/6.2010-1315>.
- [86] Franzén, L. K., Munjulury, R. C., and Krus, P., “ONTOLOGY-ASSISTED AIRCRAFT CONCEPT GENERATION,” ????. URL https://www.researchgate.net/profile/Ludvig-Knoeoes-Franzen/publication/364359536_Ontology-Assisted_Aircraft_Concept_Generation/links/634fbfd812cbac6a3ed84475/Ontology-Assisted-Aircraft-Concept-Generation.pdf.
- [87] Kuofie, E. J., “Radex: A rationale-based ontology for aerospace design explanation,” Master’s thesis, University of Twente, 2010. URL <http://essay.utwente.nl/59926/>.
- [88] Hoogreef, M. F., d’Ippolito, R., Augustinus, R., and La Rocca, G., “A multidisciplinary design optimization advisory system for aircraft design,” *5th CEAS Air and Space Conference Challenges in European Aerospace*, 2015. URL <https://repository.tudelft.nl/islandora/object/uuid:8beb86d6-40a7-4c03-bc18-8512bb5bb548/datastream/OBJ/download;A>.
- [89] Roelofs, M. N., and Vos, R., “Automatically inferring technology compatibility with an ontology and graph rewriting rules,” *Journal of Engineering Design*, Vol. 32, No. 2, 2021, pp. 90–114. <https://doi.org/10.1080/09544828.2020.1860202>, URL <https://www.tandfonline.com/doi/full/10.1080/09544828.2020.1860202>, publisher: Taylor & Francis.
- [90] Markusheska, N., Srinivasan, V., Walther, J.-N., Gindorf, A., Biedermann, J., Meller, F., and Nagel, B., “Implementing a system architecture model for automated aircraft cabin assembly processes,” *CEAS Aeronautical Journal*, Vol. 13, No. 3, 2022, pp. 689–703. <https://doi.org/10.1007/s13272-022-00582-6>, URL <https://link.springer.com/10.1007/s13272-022-00582-6>, publisher: Springer.

- [91] A.P, R., J.H, B., P.D, C., and B, N., “MDAx: Agile Generation of Collaborative MDAO Workflows for Complex Systems,” , June 2020. <https://doi.org/10.2514/6.2020-3133>, URL <https://zenodo.org/record/4672051>.
- [92] Ezhilarasu, C. M., and Jennions, I. K., “Development and implementation of a framework for aerospace vehicle reasoning (FAVER),” *IEEE Access*, Vol. 9, 2021, pp. 108028–108048.
- [93] Faris, H., Totaro, S., and Corallo, A., “A SEMANTIC KNOWLEDGE SEARCH PLATFORM FOR AEROSPACE COLLABORATIVE WORKING ENVIRONMENTS,” 2011. URL https://www.researchgate.net/profile/Salvatore-Totaro/publication/255784863_A_semantic_knowledge_search_platform_for_aerospace_collaborative_working_environments/links/0f317530466ec2de91000000/A-semantic-knowledge-search-platform-for-aerospace-collaborative-working-environments.pdf.
- [94] Gargiulo, F., Zazzaro, G., Romano, G., Gigante, G., Raggioli, A., and Fusco, R., “Aerospace Information System based on Semantic Technologies and Ontology Management,” *Proceedings of 3rd International Conference on Data Management Technologies and Applications*, 2014, pp. 341–348.
- [95] Putten, B.-J. v., Wolfe, S., and Dignum, M., “An ontology for traffic flow management,” 2008.
- [96] Malin, J. T., and Throop, D. R., “Basic concepts and distinctions for an Aerospace ontology of functions, entities and problems,” *2007 IEEE Aerospace Conference*, IEEE, 2007, pp. 1–18.
- [97] Keller, R. M., “The NASA Air Traffic Management Ontology: Technical Documentation,” Tech. rep., 2017.
- [98] Miller, D., “An ontology for future airspace system architectures,” *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, IEEE, 2017, pp. 1–8.
- [99] INSAURRALDE, C. C., and BLASCH, E., “Uncertainty in avionics analytics ontology for decision-making support.” *Journal of Advances in Information Fusion*, Vol. 13, No. 2, 2018.
- [100] Rovetto, R. J., Kelso, T. S., and O’Neil, D. A., “Orbital debris ontology, terminology, and knowledge modeling,” *Journal of space safety engineering*, Vol. 7, No. 3, 2020, pp. 451–458. URL <https://www.sciencedirect.com/science/article/pii/S2468896720300720>, publisher: Elsevier.
- [101] Rovetto, R. J., and Kelso, T. S., “Preliminaries of a Space Situational Awareness Ontology,” , June 2016. URL <http://arxiv.org/abs/1606.01924>, arXiv:1606.01924 [cs].
- [102] Halvorson, M. C., Moyers, N., and Thomas, I. D., “An Ontology for Prognostic Health Management in Spacecraft Avionics,” *Annual Conference of the PHM Society*, Vol. 14, 2022. URL https://scholar.archive.org/work/526ewz2qkza5hcb2reytgbajyi/access/wayback/https://papers.phmsociety.org/index.php/phmconf/article/download/3195/phmc_22_3195, issue: 1.
- [103] Verhagen, W. J., and Curran, R., “Ontological modelling of the aerospace composite manufacturing domain,” *Improving Complex Systems Today: Proceedings of the 18th ISPE International Conference on Concurrent Engineering*, Springer, 2011, pp. 215–222.

- [104] Neumayr, B., Gringinger, E., Schuetz, C. G., Schrefl, M., Wilson, S., and Vennesland, A., “Semantic data containers for realizing the full potential of system wide information management,” *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, IEEE, 2017, pp. 1–10.
- [105] Okoh, C., Roy, R., Mehnen, J., Redding, L., and Harrison, A., “Development of an Ontology for Aerospace Engine Components Degradation in Service,” *International Conference on Knowledge Engineering and Ontology Development*, Vol. 2, ScitePress, 2014, pp. 108–119.
- [106] Wright, R., Sánchez-Gestido, M., Gómez-Pérez, A., Pérez-Hernández, M. S., González-Cabero, R., and Corcho, O., “A semantic data grid for satellite mission quality analysis,” *International Semantic Web Conference*, Springer, 2008, pp. 818–832.
- [107] Wicks, E., “Ontological Modeling for Integrated Spacecraft Analysis,” 2011.
- [108] Cambrésy, L., Derriere, S., Padovani, P., Preite Martinez, A., and Richard, A., “Ontology of astronomical object types,” *International Virtual Observatory Alliance*. Available at: <http://www.ivoa.net/Documents/Notes/AstrObjectOntology/20100117/NOTE-AstrObjectOntology-1.3-20100117.html>. Accessed on, Vol. 11, No. 06, 2010, p. 2011.
- [109] Tikayat Ray, A., Pinon Fischer, O. J., White, R. T., Cole, B. F., and Mavris, D. N., “Development of a Language Model for Named-Entity-Recognition in Aerospace Requirements,” *Journal of Aerospace Information Systems*, Vol. 21, No. 6, 2024, pp. 489–499. <https://doi.org/10.2514/1.I011251>, URL <https://doi.org/10.2514/1.I011251>, publisher: American Institute of Aeronautics and Astronautics _eprint: <https://doi.org/10.2514/1.I011251>.