Semantic Representation of Processes with Ontology Design Patterns

Ebrahim Norouzi[1,2,*], Sven Hertling[1,3], Jörg Waitelonis[1] and Harald Sack[1,2]

[1]FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

[2]Karlsruhe Institute of Technology, Institute of Applied Informatics and Formal Description Methods, Kaiserstr. 89, 76133 Karlsruhe

[3]University of Mannheim, Data and Web Science Group, School of Business Informatics and Mathematics B6 26, 68159 Mannheim

Abstract

The representation of workflows and processes is essential in materials science engineering, where experimental and computational reproducibility depend on structured and semantically coherent process models. Although numerous ontologies have been developed for process modeling, they are often complex and challenging to reuse. Ontology Design Patterns (ODPs) offer modular and reusable modeling solutions to recurring problems; however, these patterns are frequently neither explicitly published nor documented in a manner accessible to domain experts. This study surveys ontologies relevant to scientific workflows and engineering process modeling and identifies implicit design patterns embedded within their structures. We evaluate the capacity of these ontologies to fulfill key requirements for process representation in materials science. Furthermore, we propose a baseline method for the automatic extraction of design patterns from existing ontologies and assess the approach against curated ground truth patterns. All resources associated with this work—including the extracted patterns and the extraction workflow—are made openly available in a public GitHub repository.[1].

1. Introduction

Process modeling is a broad field that plays a critical role across numerous scientific and engineering domains, particularly in Materials Science and Engineering (MSE). For decades, the development of new materials has relied on understanding the intricate correlation between processing methods and resulting material properties [1]. The well-established paradigm of processing–structure–properties–performance encapsulates the fundamental principles of MSE: material performance is governed by its properties, which are determined by its structure, and the structure is ultimately shaped by the applied processing route [2, 3].

Accurately modeling processing steps is therefore essential for representing the dynamic transformations that occur during the material's life cycle. Several large-scale initiatives[1] [2] [3] [4] [4] aim to formalize materials data and workflows through taxonomies, ontologies, and knowledge graphs, encompassing the entire life cycle from synthesis to performance evaluation. A key challenge, however, lies in capturing the dynamic events during which materials transition between states due to processing interventions[5].

Multiple ontologies have been developed for process modeling in MSE, yet issues with interoperability and reuse persist. A major gap is the limited adoption of Ontology Design Patterns (ODPs) [6], which provide reusable template solutions for recurring modeling problems [7]. In many cases, ontology developers or domain experts embed these patterns unintentionally, failing to explicitly publish them as reusable modules [8]. Identifying and extracting these embedded patterns could significantly enhance ontology reuse and provide domain experts with modular building blocks for representing workflows and processes without requiring them to navigate entire ontologies.

In this work, we focus on scientific workflow modeling and the extraction of ontology design patterns relevant to process representation in MSE. We achieve this by surveying existing ontologies, identifying patterns that align with domain-specific requirements, and proposing a baseline method for automatically extracting ODPs using semantic similarity techniques.

The remainder of this paper is structured as follows. We begin in Section 2 with a review of related ontologies and existing ontology design patterns for process modeling. Section 3 details our methodology for automatically extracting and evaluating these patterns. In Section 4, we discuss the patterns we extracted based on our defined requirements and present an evaluation of the results. Finally, Section 5 provides our conclusion and suggests directions for future research.

2. Related Work

Numerous ontologies have been developed to model processes, workflows, and experimental activities. However, their relevance to Materials Science and Engineering (MSE) depends on their domain focus and the granularity of their process semantics.

Several general-purpose ontologies have been created to represent scientific experimentation. For instance, the EXPO ontology[5] was designed to formalize experimental design, methodology, and result representation across various disciplines [9]. Although comprehensive, it lacks the specificity required to model the technical details of MSE processes. In the life sciences, ontologies like EXACT2[6] [10] and SMART Protocols[7] [11] were developed to capture biomedical and biological experimental protocols. These ontologies are domain-specific and not directly applicable to engineering workflows considered in this study.

In contrast, process ontologies in the engineering domain often focus on industrial or chemical workflows. OntoCAPE[8] [12], for example, is a large-scale ontology for computer-aided process engineering. It models chemical plant design and operation, making it more suitable for the chemical industry than for experimental workflows common in materials science. The Procedural Knowledge Ontology (PKO)[9] [13] was developed to manage and reuse procedural knowledge, capturing procedures as sequences of executable steps. However, its focus lies on execution tracking and procedural documentation, not on modeling the structure of processes. Similarly, the BPMN-Based Ontology (BBO)[10] [14] represents business-oriented process models and organizational workflows using the BPMN standard, which differs in scope from the scientific workflows relevant to this study.

Recently, a number of ontologies have been developed specifically to support workflow and process modeling in MSE [6]. PMDcore[11] [4] describes material processing activities and ensures semantic interoperability in research data. The General Process Ontology[12] [15] generalizes engineering process structures, enabling the composition of complex processes. The Workflows in Linked Data (WILD)[13] [16] ontology supports the modeling of executable workflows in RDF, while P-PLAN[14] [17] enables the

---

[5]http://expo.sourceforge.net/

[6]https://bmcbiomed.biomedcentral.com/articles/10.1186/1471-2105-15-S14-S5

[7]https://smartprotocols.org/

[8]https://www.avt.rwth-aachen.de/Ontocape

[9]https://w3id.org/pko

[10]https://www.irit.fr/recherches/MELODI/ontologies/BBO

[11]https://w3id.org/pmd/co/2.0.8

[12]https://gpo.ontology.link#

[13]http://purl.org/wild/vocab#

[14]http://purl.org/net/p-plan#

representation of scientific planning activities. Similarly, Metadata4Ing[15] [18] supports the documentation of data-generating processes in scientific experiments. The BWMD ontology [19] was developed to support the semantic representation of material-intensive process chains. While the ontology offers a rich process modeling structure, we do not include it in our analysis due to its limited use of annotation properties, which makes it challenging to extract patterns and understand the modeling intent.

Foundational ontologies have also been used to construct reusable ontology design patterns (ODPs) for process modeling. A prominent example is the Basic Formal Ontology (BFO)[16], which serves as the top-level ontology for many process-centric models in the Open Biological and Biomedical Ontology (OBO) Foundry[20]. A comprehensive review by Dooley et al. [21] systematically examined BFO-based process models and their associated patterns across the OBO ecosystem. Since that study provides an in-depth analysis of BFO-derived patterns, these are not re-evaluated in this work. Another example is NFDICore[17], which builds on BFO and defines patterns for data handling, contribution, and publishing processes [22], with an emphasis on agent–information interactions[18]. Although NFDIcore is well-suited for representing metadata about research resources within the National Research Data Infrastructure (NFDI) programme[19], its focus does not fully align with the process modeling needs of MSE, which emphasize material transformations and experimental workflows. Alternatively, the PlansLite ontology[20], based on the DOLCE foundational ontology [23], captures planning constructs through a task-based structure. While suitable for modeling general workflow logic, its notion of "process" is defined as a placeholder for evolving events not strictly tied to agents, tasks, or plans. As a result, it lacks the formalization needed to represent material-centered processes and transformations.

Beyond foundational ontologies, various domain-independent ontology design patterns (ODPs) have been proposed to address recurring challenges in process modeling. These include patterns such as Algorithm Implementation Execution[21] [24], Material Transformation[22] [25], and Reactive Processes[23] [26]. Additional patterns like Sequence[24], Transition[25] [27], Task Execution[26], Activity Reasoning[27], and Activity Specification[28] [28] are perticularly useful for capturing the semantics of tasks, actions, and events. These patterns provide modular constructs for representing execution order and dependencies between tasks. In manufacturing domains, the VDI 3682 ODP[29] [29] and standards such as ISO 22400-2[30], ISO 10303-44 (STEP)[31], and DIN EN 62264-2[32] address the need to unify and structure heterogeneous data environments across production systems.

Despite their general applicability, these patterns have seen limited reuse in ontology development within the MSE community. One contributing factor is that many of these patterns are indeed too generic to fully capture the process modeling requirements of MSE contexts. However, it is essential to recognize that the adoption of semantic web technologies in MSE, including domain-specific ontologies and ontology design patterns, is still in its early stages. Consequently, even resources specifically designed for MSE have not yet achieved widespread uptake, often due to constraints such as a lack

---

[15] http://w3id.org/nfdi4ing/metadata4ing#

[16] https://basic-formal-ontology.org/

[17] https://ise-fizkarlsruhe.github.io/nfdicore/

[18] https://ise-fizkarlsruhe.github.io/nfdicore/docs/patterns/#processes

[19] https://www.nfdi.de

[20] http://www.ontologydesignpatterns.org/ont/dul/PlansLite.owl

[21] https://odpa.github.io/patterns-repository/AlgorithmImplementationExecution/AlgorithmImplementationExecution

[22] https://odpa.github.io/patterns-repository/Material_Transformation/Material_Transformation

[23] https://odpa.github.io/patterns-repository/Reactor_pattern/Reactor_pattern

[24] https://odpa.github.io/patterns-repository/Sequence/Sequence

[25] https://odpa.github.io/patterns-repository/Transition/Transition

[26] https://odpa.github.io/patterns-repository/TaskExecution/TaskExecution

[27] https://odpa.github.io/patterns-repository/An_Ontology_Design_Pattern_for_Activity_Reasoning/An_Ontology_Design_Pattern_for_Activity_Reasoning

[28] https://odpa.github.io/patterns-repository/ActivitySpecification/ActivitySpecification

[29] https://github.com/hsu-aut/IndustrialStandard-ODP-VDI3682

[30] https://github.com/hsu-aut/IndustrialStandard-ODP-ISO22400-2

[31] https://www.hsu-hh.de/aut/forschung/forschungsthemen/ontology-engineering-for-collaborative-embedded-systems/ontology-design-patterns-for-the-manufacturing-domain/iso-10303-44-product-structures

[32] https://github.com/hsu-aut/IndustrialStandard-ODP-DINEN62264-2

of tooling support and the steep learning curve for non-semantic web experts. In this sense, making ontology design patterns more accessible—both conceptually and technically—can help improve the adaptability and reuse of existing semantic assets in MSE.

In this work, we focus on the subset of ontologies and patterns most relevant to scientific workflow and process modeling in MSE. Specifically, we analyze PMDcore, the EMMO General Process Ontology, WILD, P-PLAN, and Metadata4Ing to identify reusable ontology design patterns that align with the defined process modeling requirements in materials science and engineering.

3. Methodology

The methodology for this study was designed to identify and evaluate ontology design patterns (ODPs) for process modeling in Materials Science and Engineering (MSE). Guided by a review from domain experts, our approach focuses on capturing reusable patterns that address key requirements for representing scientific workflows and engineering processes. We began by selecting candidate ontologies based on their relevance to scientific workflows and material processing [6]. From these, we extracted ODPs by isolating coherent modules that align with our defined requirements.

3.1. Requirements Definition

The first step in our methodology involved defining the core requirements for ontology-based process modeling in MSE. These requirements were derived from an analysis of existing process ontologies and adapted to the specific needs of the MSE domain. The following three requirements guided the subsequent pattern extraction.

- **Requirement 1: Process Structure**
  The ontology must describe the organization of processes, including their constituent steps, sub-processes, and execution order. In MSE, this corresponds to modeling transformations such as heat treatments, alloy mixing, thin-film deposition, and sequential multi-step processing stages that directly influence material microstructures.
- **Requirement 2: Data and Resources**
  The ontology must represent the flow of inputs and outputs, as well as the parameters that define each process step. For MSE, this specifically refers to capturing experimental parameters such as temperature, pressure, atmosphere conditions, measurement devices, and calibration data that are essential for reproducibility.
- **Requirement 3: Project Goals and Participant Roles**
  The ontology must capture project goals, experimental stages, and their organizational context, including the agents involved and their specific roles within the project or process. In MSE, this relates to modeling multi-lab collaborations, research campaigns, project milestones, and the allocation of responsibilities (e.g., sample synthesis, microscopy, or simulation).

The extracted requirements and patterns were evaluated by nine domain experts within the Linked Open Data Working Group (LOD-WG)[33]. This group, a part of the *National Research Data Infrastructure for Materials Science and Engineering (NFDI-MatWerk)*[30][34], focuses on creating and reusing Linked Open Data across the MSE domain. Feedback from the LOD-WG was crucial for refining the extracted ODPs and validating their alignment with the domain-specific requirements of process modeling in MSE.

Ontology design patterns were first extracted and evaluated against our defined requirements. Following this, a baseline approach was developed to automatically identify these ODPs. The details of this method are presented in the subsequent section.

---

[33]https://iuc12-nfdi-matwerk-ta-oms-7fd4826d9051b0dd93b21aa77d06d1d8c71c4.pages.rwth-aachen.de/docs/meetings/2025/

[34]https://nfdi-matwerk.de/

3.2. Baseline Method for Automated ODP Extraction

Our methodology applies semantic similarity techniques to align ontology concepts with textual requirements, as illustrated in Figure 1. Each requirement is reformulated into natural language sentences to clearly express the intended representation. For each ontology, we extract textual metadata from its classes and properties, specifically the values of `rdfs:label`, `skos:definition`, `rdfs:comment`, and related annotation properties. These are concatenated into unified textual descriptions for each IRI (Internationalized Resource Identifier). We then embed both the requirements and ontology IRIs into a shared vector representation using sentence transformer models[31]. A similarity matrix $S(i, j)$ is computed between every requirement r_i and ontology concept u_j, using cosine similarity. We apply a similarity threshold θ to identify relevant ontology terms for each requirement.
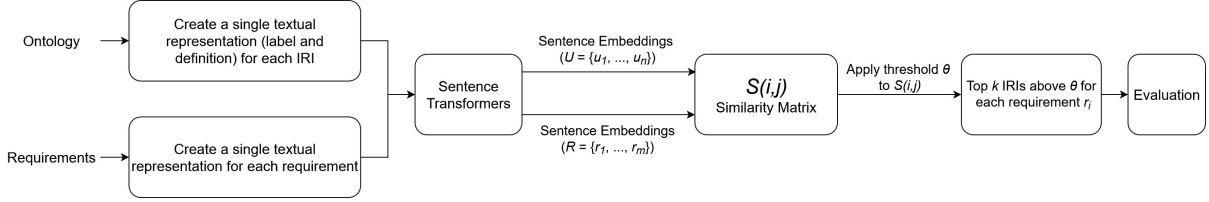
Figure 1: Baseline workflow for automatic extraction of ontology design patterns (ODPs).

The selected IRIs are evaluated against manually curated ground truth mappings using standard information retrieval metrics: precision, recall, and the F_1-score. These metrics are defined as follows:

Let C_i denote the set of ontology concepts retrieved for requirement r_i, and C_i' the set of ground truth concepts for r_i. Precision is the proportion of correctly retrieved ontology IRIs out of all retrieved IRIs:

$$\text{Precision} = \frac{|C_i \cap C_i'|}{|C_i|}$$

Recall is the proportion of correctly retrieved ontology IRIs relative to the total number of relevant IRIs:

$$\text{Recall} = \frac{|C_i \cap C_i'|}{|C_i'|}$$

F_1-score is the harmonic mean of precision and recall:

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

To generate reusable ontology design patterns (ODPs) aligned with domain-specific requirements in materials science, a modular extraction strategy was applied using the ROBOT command-line framework.[35] This process transformed the identified IRIs into ontology modules, with ROBOT preserving the relevant structural and logical relations around the selected terms.

We evaluated four ROBOT-supported extraction methods: STAR, BOT, TOP, and subset.

- **STAR** extracts a minimal module containing only the specified terms and those directly involved in logical entailments. It avoids a hierarchical structure, resulting in a compact module.
- **BOT** builds a module that includes seed terms and all their superclasses, preserving hierarchical context in a bottom-up manner.
- **TOP** includes the seed terms along with their subclasses, typically yielding larger modules.
- **Subset** materializes existential relations (e.g., `someValuesFrom`) and includes inferred axioms between seed terms. This method is ideal for extracting patterns that involve complex class expressions.

We excluded the Minimum Information to Reference an External Ontology Term (*MIREOT*) [32] method from our evaluation. Although useful for referencing external terms while minimizing dependencies, its lack of entailment preservation makes it unsuitable for logic-based pattern extraction. Each of the four extraction methods was also tested with varying levels of intermediate class inclusion using the `all`, `minimal`, and none options.

- `all` includes all intermediate classes between seed terms and ontology roots.
- `minimal` retains only those intermediates with multiple siblings, preserving a minimal hierarchy.
- `none` removes all intermediate classes unless they are explicitly referenced in logical axioms.

We selected the STAR method with `–intermediates none` option as our default configuration after an empirical evaluation. This setup generated semantically coherent and compact modules, preserving just enough logic to enable accurate pattern identification without extraneous context. For each requirement, we curated a list of seed terms and then extracted a separate ontology module accordingly.

4. Results and Discussion

4.1. Process Modeling Patterns

This section presents the manually derived design patterns, which were reviewed and evaluated by domain experts, and mapped to our defined requirements: Process ODP (Req. 1), Resource ODP (Req. 2), and Project ODP (Req. 3). In the accompanying figures, the ODPs are visually grouped using dashed red boundaries with corresponding labels. While the above descriptions emphasize the manually derived structures, it is important to highlight how these patterns are subsequently integrated into the automated workflow (Section 3.2. In practice, the extracted modules were directly reused for semantic similarity evaluation and ODP benchmarking (see Section 4.2). This ensures that the manually curated insights are not standalone but feed into the workflow's automated pattern extraction and validation steps.

P-PLAN Ontology P-PLAN supports both Process and Resource ODPs. Process structures are represented using `p-plan:Step`, `p-plan:Activity`, with their sequencing handled by `p-plan:isPrecededBy`. Inputs and outputs are modeled with `p-plan:Variable` and the properties `p-plan:hasInputVar/hasOutputVar`. This configuration provides a compact but expressive model for basic workflow representations. In this work, we focus on P-PLAN as it provides explicit constructs for modeling scientific planning activities and thus directly supports our defined requirements for process and resource ODPs. We did not include the core PROV ontology or other extensions such as ProvONE and D-PROV in our analysis, since our emphasis was on ontologies that are already specialized toward workflow representation.

Metadata4Ing (M4I) M4I provides strong support for both the Resource and Project ODPs. It satisfies the Resource ODP requirements by linking experimental tools, methods, and variables via `m4i:hasEmployedTool`, `m4i:hasParameter`, and related properties. For Project ODPs it models project roles and participants using `schema:Project`, `prov:Association`, and `prov:Role`.

OPMW Ontology OPMW addresses both the Process and Resource ODPs. It models the execution and templating of workflows using `opmw:WorkflowTemplateProcess`, `opmw:WorkflowExecutionProcess`, and their associated input/output artifacts. However, while agents are present, the model lacks detailed project-level constructs and necessary to fulfill Project ODP requirements.
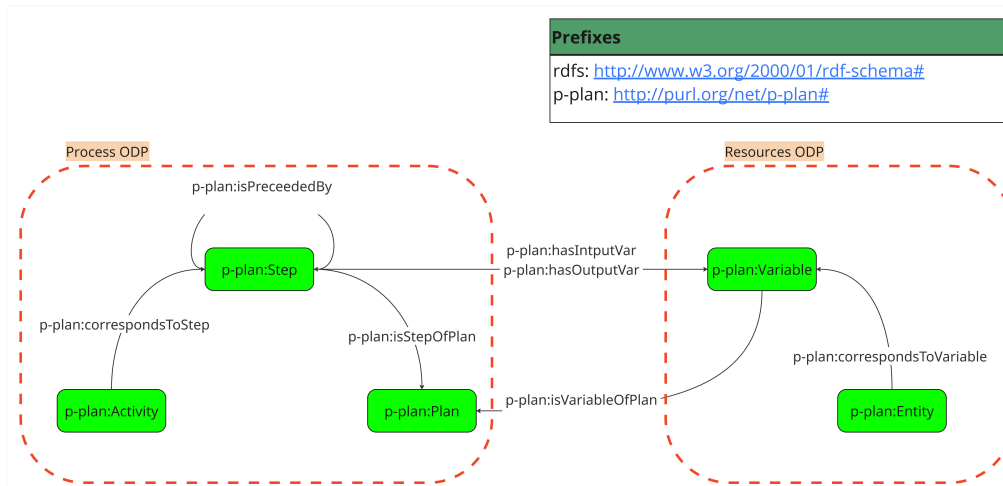
Figure 2: Extracted Process and Resource ODPs from P-PLAN based on defined requirements.
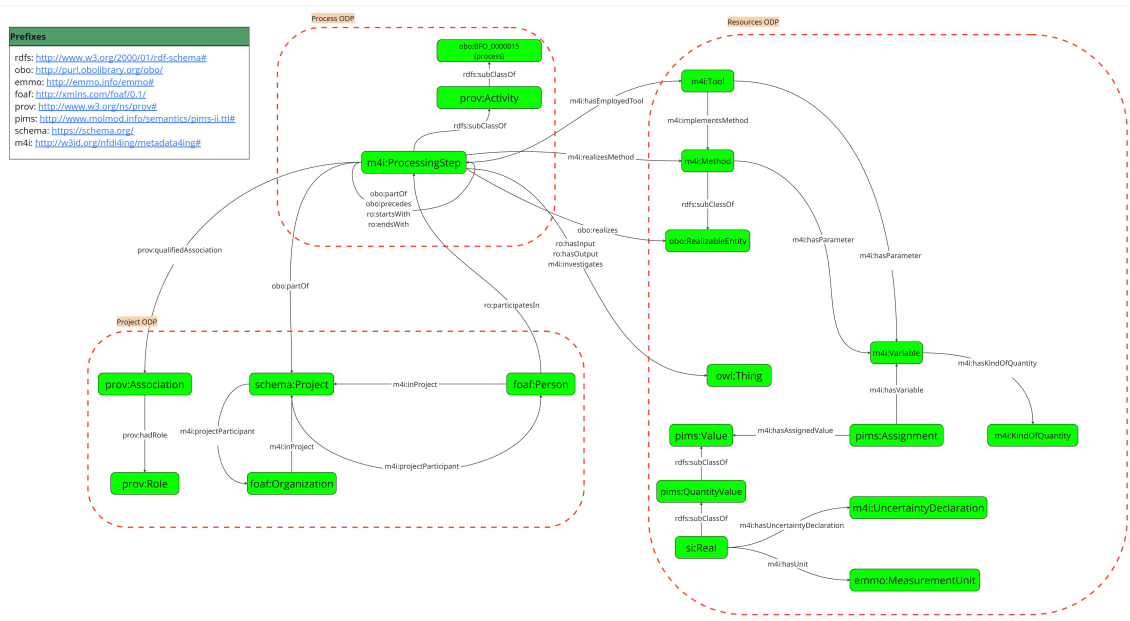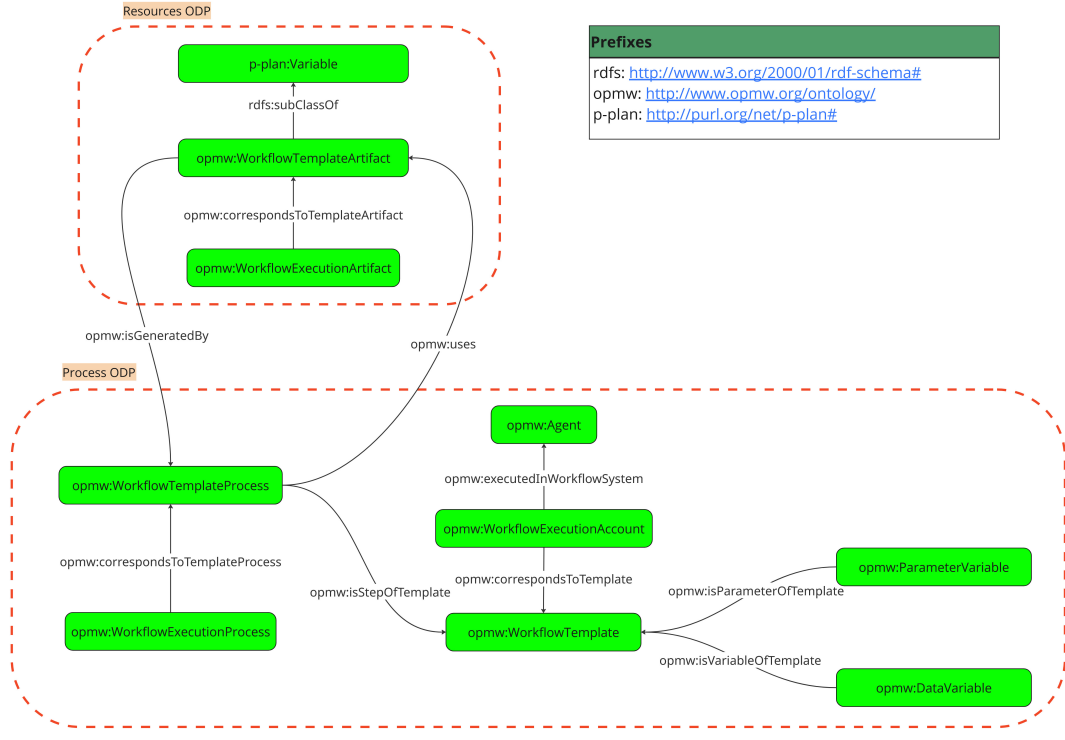

Figure 3: Extracted Resource and Project ODPs from Metadata4Ing.

Workflows in Linked Data (WILD) WILD models workflow activities using classes like `wild:Activity`, `wild:WorkflowModel`, and their subclasses (e.g., `wild:CompositeActivity`, `wild:SequentialActivity`). However, despite this structural expressiveness, the ontology lacks alignment with concrete input/output or project-level constructs. Consequently, while it supports Process ODP, it does not sufficiently fulfill Resource and Project ODPs.

General Process Ontology (GPO) GPO supports both the Process and Resource ODPs. The `gpo:Process` class and its related elements (`gpo:SubProcess`, `gpo:hasPredecessor`, `gpo:hasProcessInput`) provide a robust structure for process execution. While inputs, outputs, and tools are also modeled, the ontology lacks constructs for agent participation or organizational context and therefore does not fulfill the Project ODP requirements.

PMDcore Ontology PMDcore supports all three ODP categories. It models process breakdown using `pmdco:Process` and `pmdco:ProcessingNode`, while `pmdco:ValueObject`, `pmdco:Object`, and

Figure 4: Extracted Process and Resource ODPs from OPMW.

their associated roles fulfill Resource ODP requirements. To support the Project ODP requirements, `pmdco:Project` and `pmdco:ProjectIdentifier` represent project membership and identifiers, enabling linkage of processes to organizational contexts.

4.2. Evaluation Results

To assess how well each ontology supports the defined process modeling requirements in MSE, we applied a semantic similarity-based matching approach. For each requirement, we curated a set of ground truth IRIs (GT) and then retrieved a ranked list of relevant IRIs (Ext) using the `hkunlp/instructor-large`[33] embedding model, as introduced in Section 3.2. It is important to note that our evaluation considers all IRIs, including both classes and object properties.

We evaluated the performance of this extraction using precision (P), recall (R), and F_1-score (F_1) by comparing the extracted IRIs against the ground truth. A dash (−) indicates that an ontology lacks adequate coverage for the given requirement, meaning no ODP could be extracted in that category. Table 1 summarizes these results for all evaluated ontologies and requirement groups.

Our baseline results confirm that semantic similarity techniques can be applied to support the extraction of ontology design patterns aligned with defined requirements. In terms of F_1-score, P-PLAN achieved the top scores for the Process and Resource ODPs, demonstrating high precision. PMDcore supported all three ODPs (Process 0.36; Resource 0.21; Project 0.25) and showed strong recall for the Project ODP. M4I performed moderately for the Resource and Project ODPs but showed limited performance on the Process ODP.

However, the method has several limitations. First, ontologies with more descriptive labels and definitions tend to yield higher scores, introducing a bias based on annotation quality. Second, the formulation of requirements can influence the results by favoring certain ontologies that use specific terminology. To address these sources of bias in future work, several strategies can be pursued. First, the reliance on descriptive labels and definitions could be mitigated by incorporating structural features of the ontology (e.g., graph topology, axiomatic richness) alongside textual embeddings. Second,
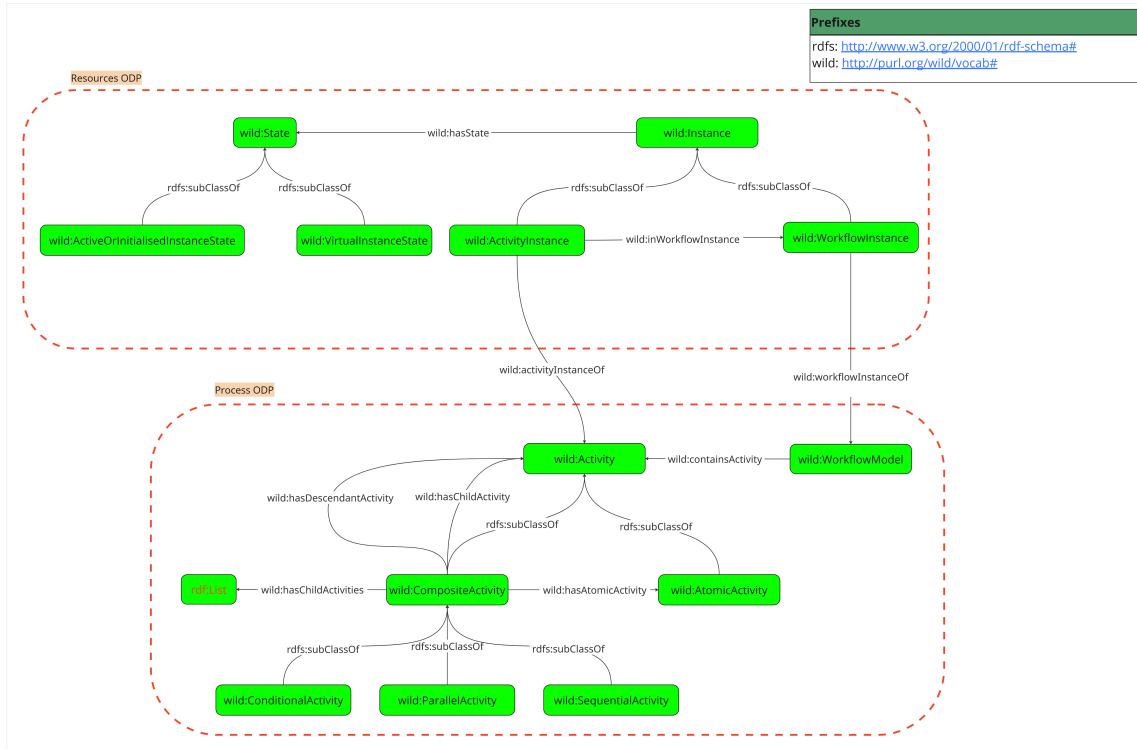
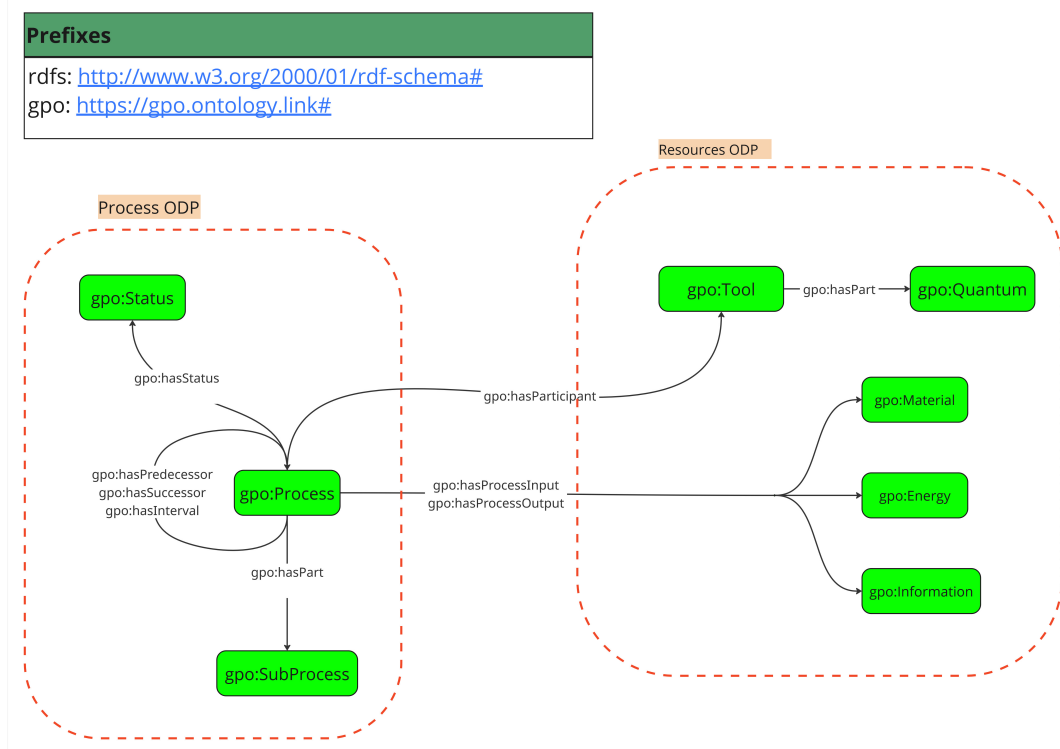**Figure 5:** Extracted Process ODPs from WILD ontology.

Table 1

Evaluation of ODP extraction across requirement categories using `hkunlp/instructor-large`. P = Precision, R = Recall, F_1 = F_1-score, GT = Number of ground truth IRIs, Ext = Number of extracted IRIs. Many Ext values equal 20, which corresponds to the fixed retrieval size in our semantic similarity pipeline. This cutoff was chosen to balance coverage and noise, ensuring each requirement is matched against a manageable subset of ontology terms.

| Ontology	Process ODP (Req. 1)					Resource ODP (Req. 2)					Project ODP (Req. 3)				
	P	R	F_1	GT	Ext	P	R	F_1	GT	Ext	P	R	F_1	GT	Ext
GPO	0.10	0.17	0.12	12.0	20.0	0.05	0.11	0.07	9.0	20.0	–	–	–	–	–
P-Plan	0.56	0.62	0.59	8.0	9.0	0.44	0.67	0.53	6.0	9.0	–	–	–	–	–
WILD	0.00	0.00	0.00	11.0	0.0	0.00	0.00	0.00	9.0	0.0	–	–	–	–	–
M4I	0.05	0.12	0.07	8.0	20.0	0.25	0.28	0.26	18.0	20.0	0.15	0.25	0.19	12.0	20.0
OPMW	0.45	0.69	0.55	13.0	20.0	0.15	0.60	0.24	5.0	20.0	–	–	–	–	–
PMDcore	0.25	0.62	0.36	8.0	20.0	0.15	0.33	0.21	9.0	20.0	0.15	0.75	0.25	4.0	20.0

diversifying the set of requirement formulations by paraphrasing or using controlled vocabularies can reduce the sensitivity of results to specific wording choices. Finally, cross-ontology validation, where extraction results are compared against multiple ground truths curated by different expert groups, would help balance annotation subjectivity and improve the robustness of the evaluation.

A further limitation is that our current workflow processes each ontology independently, which makes it difficult to identify patterns that are shared or distributed across multiple sources. In addition, our focus on ontology modularization as the primary means of extracting reusable ODPs narrows the scope of reusability strategies. While modularization facilitates compact module creation and automated processing, it is not the only pathway to reuse. Alternative approaches include the direct publication of curated pattern catalogs, the use of competency questions to guide pattern identification, and the incorporation of ontology mappings to capture cross-ontology patterns. Despite these limitations, the proposed method shows promise as a first step toward automating the identification of reusable semantic structures for process modeling in MSE.

Figure 6: Extracted Process and Resource ODPs from GPO.
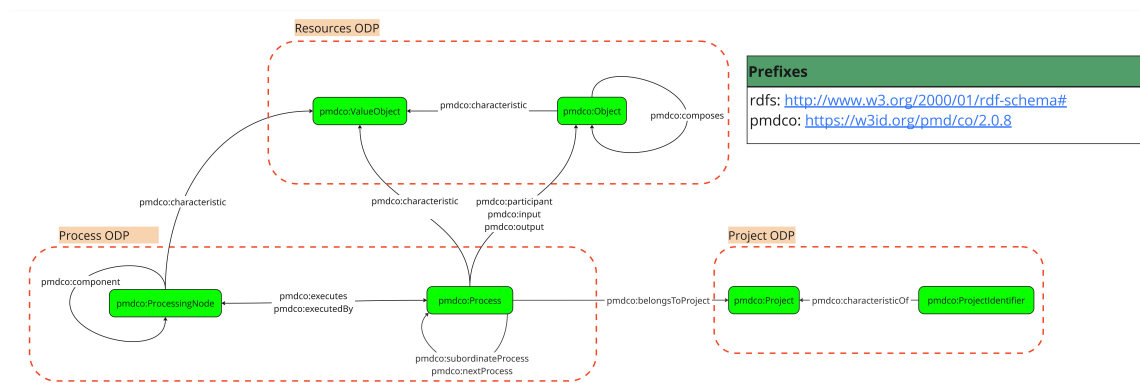
Figure 7: Extracted Process, Resource, and Project ODPs from PMDcore.

5. Conclusion and Future Work

This work surveyed and evaluated ontologies and ontology design patterns (ODPs) relevant to process modeling in Materials Science and Engineering (MSE). We defined a set of domain-specific requirements to guide the assessment of existing ontologies and to support the identification of reusable semantic patterns. Based on this framework, process-related ODPs—covering aspects such as process structure, resource involvement, and project context—were extracted from relevant ontologies and consolidated into ground truth datasets.

Building on this manual analysis, we introduced and evaluated a baseline method for the automatic extraction of ODPs. This approach leverages semantic similarity techniques and modular ontology extraction to support scalable pattern identification aligned with domain requirements.

Future work will expand this analysis to additional MSE subdomains and modeling needs, such as value specification and property mapping. Beyond expanding the corpus, we plan to improve the methodology itself by integrating ontology mappings into the extraction workflow, allowing shared

patterns to be identified across multiple ontologies rather than in isolation. We also intend to enhance the evaluation procedure by addressing the influence of annotation bias and requirement formulation, for example, through controlled vocabularies and multi-expert validation. In parallel, a platform will be developed to publish the extracted patterns, provide usage scenarios, and enable community feedback, thereby improving both methodological rigor and practical accessibility for the broader MSE community.

Acknowledgments

Declaration on Generative AI

During the preparation of this work, the author(s) used GPT-4 and Claude Sonnet 4 in order to: Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] G. Olson, Computational design of hierarchically structured materials, Science 277 (1997) 1237–1242. doi:`10.1126/science.277.5330.1237`.

[2] M. F. Ashby, D. Cebon, Materials selection in mechanical design, Le Journal de Physique IV 3 (1993) C7–1.

[3] W. D. Callister Jr, D. G. Rethwisch, Callister's materials science and engineering, John Wiley & Sons, 2020.

[4] B. Bayerlein, M. Schilling, H. Birkholz, M. Jung, J. Waitelonis, L. Mädler, H. Sack, Pmd core ontology: Achieving semantic interoperability in materials science, Materials & Design 237 (2024) 112603.

[5] M. Alam, H. Birkholz, D. Dessì, C. Eberl, H. Fliegl, G. Peter, L. Madler, M. Niebel, H. Sack, A. Thomas, et al., Ontology modelling for materials science experiments, in: CEUR WORKSHOP PROCEEDINGS, volume 2941, CEUR-WS, 2021.

[6] E. Norouzi, J. Waitelonis, H. Sack, The landscape of ontologies in materials science and engineering: A survey and evaluation, arXiv preprint arXiv:2408.06034 (2024).

[7] A. Gangemi, V. Presutti, Ontology design patterns, in: Handbook on ontologies, Springer, 2009, pp. 221–243.

[8] E. Blomqvist, P. Hitzler, K. Janowicz, A. Krisnadhi, T. Narock, M. Solanki, Considerations regarding ontology design patterns, 2015.

[9] L. N. Soldatova, R. D. King, An ontology of scientific experiments, Journal of the royal society interface 3 (2006) 795–803.

[10] L. N. Soldatova, D. Nadis, R. D. King, P. S. Basu, E. Haddi, V. Baumlé, N. J. Saunders, W. Marwan, B. B. Rudkin, Exact2: the semantics of biomedical protocols, BMC bioinformatics 15 (2014) S5.

[11] O. L. e. a. Giraldo, Using semantics for representing experimental protocols, Journal of Biomedical Semantics 8 (2017).

[12] J. Morbach, A. Yang, W. Marquardt, Ontocape—a large-scale ontology for chemical process engineering, Engineering applications of artificial intelligence 20 (2007) 147–161.

[13] V. A. Carriero, M. Scrocca, I. Baroni, A. Azzini, I. Celino, Procedural knowledge ontology (pko), in: European Semantic Web Conference, Springer, 2025, pp. 334–350.

[14] A. Annane, N. Aussenac-Gilles, M. Kamel, Bbo: Bpmn 2.0 based ontology for business process representation, in: 20th European Conference on Knowledge Management (ECKM 2019), volume 1, HAL open science, 2019, pp. 49–59.

[15] Emmo general process ontology, 2022. URL: https://github.com/General-Process-Ontology/ontology.

[16] T. Käfer, S. Lauber, A. Harth, Using workflows to build compositions of read-write linked data apis on the web of things., 2018.

[17] D. Garijo, O. Corcho, P-plan: A vocabulary for describing the plans of scientific processes, 2013. URL: http://vocab.linkeddata.es/p-plan/index.html.

[18] S. Arndt, B. Farnbacher, M. Fuhrmans, S. Hachinger, J. Hickmann, N. Hoppe, M. T. Horsch, D. Iglezakis, A. Karmacharya, G. Lanza, S. Leimer, J. Munke, D. Terzijska, J. Theissen-Lipp, C. Wiljes, J. Windeck, Metadata4Ing: An ontology for describing the generation of research data within a scientific activity., 2022. URL: https://doi.org/10.5281/zenodo.5957104. doi:10.5281/zenodo.5957104.

[19] E. Garcia Trelles, C. Schweizer, A. Thomas, P. von Hartrott, M. Janka-Ramm, Digitalizing material knowledge: A practical framework for ontology-driven knowledge graphs in process chains, Applied Sciences 14 (2024). URL: https://www.mdpi.com/2076-3417/14/24/11683. doi:10.3390/app142411683.

[20] R. Arp, B. Smith, A. D. Spear, Building ontologies with Basic Formal Ontology, MIT Press, 2015.

[21] D. Dooley, M. Weber, L. Ibanescu, M. Lange, L. Chan, L. Soldatova, C. Yang, R. Warren, C. Shimizu, H. K. McGinty, et al., Food process ontology requirements, Semantic Web 15 (2024) 1133–1164.

[22] O. Bruns, T. Tietz, J. Waitelonis, E. Posthumus, H. Sack, Nfdicore 2.0: A bfo-compliant ontology for multi-domain research infrastructures, arXiv preprint arXiv:2410.01821 (2024).

[23] S. Borgo, R. Ferrario, A. Gangemi, N. Guarino, C. Masolo, D. Porello, E. M. Sanfilippo, L. Vieu, Dolce: A descriptive ontology for linguistic and cognitive engineering, Applied ontology 17 (2022) 45–69.

[24] A. Ławrynowicz, D. Esteves, P. Panov, T. Soru, S. Džeroski, J. Vanschoren, An algorithm, implementation and execution ontology design pattern, in: Advances in Ontology Design and Patterns, IOS Press, 2017, pp. 55–68.

[25] C. F. Vardeman, A. A. Krisnadhi, M. Cheatham, K. Janowicz, H. Ferguson, P. Hitzler, A. P. Buccellato, An ontology design pattern and its use case for modeling material transformation, Semantic Web 8 (2017) 719–731.

[26] M. Solanki, C. Chapman, Reactive processes., in: WOP, 2012.

[27] V. A. Carriero, A. Gangemi, A. G. Nuzzolese, V. Presutti, Chapter 10. An Ontology Design Pattern for Representing Recurrent Situations, IOS Press, 2021. URL: http://dx.doi.org/10.3233/SSW210013. doi:10.3233/ssw210013.

[28] M. Katsumi, M. S. Fox, Defining activity specifications in owl., in: WOP@ ISWC, 2017.

[29] C. Hildebrandt, A. Köcher, C. Küstner, C.-M. López-Enríquez, A. W. Müller, B. Caesar, C. S. Gundlach, A. Fay, Ontology building for cyber–physical systems: Application in the manufacturing domain, IEEE Transactions on Automation Science and Engineering 17 (2020) 1266–1282. doi:10.1109/TASE.2020.2991777.

[30] C. Eberl, M. Niebel, E. Bitzek, T. Dahmen, F. Fritzen, P. Gumbsch, T. Hickel, S. Klein, F. Mücklich, M. S. Müller, J. Neugebauer, C. Pauly, M. Politze, H. Sack, S. Sandfeld, R. Schwaiger, P. Slusallek, R. Stotzka, A. Streit, M. Zimmermann, Consortium proposal nfdi-matwerk, 2021. URL: https://doi.org/10.5281/zenodo.5082837. doi:10.5281/zenodo.5082837.

[31] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: http://arxiv.org/abs/1908.10084.

[32] M. Courtot, F. Gibson, A. L. Lister, J. Malone, D. Schober, R. R. Brinkman, A. Ruttenberg, Mireot: The minimum information to reference an external ontology term, Applied Ontology 6 (2011) 23–33.

URL: https://journals.sagepub.com/doi/abs/10.3233/AO-2011-0087. doi:10.3233/AO-2011-0087. arXiv:https://journals.sagepub.com/doi/pdf/10.3233/AO-2011-0087.

[33] H. Su, W. Shi, J. Kasai, Y. Wang, Y. Hu, M. Ostendorf, W.-t. Yih, N. A. Smith, L. Zettlemoyer, T. Yu, One embedder, any task: Instruction-finetuned text embeddings, arXiv preprint arXiv:2212.09741 (2022).