INTERNATIONAL DESIGN CONFERENCE – DESIGN 2024
https://doi.org/10.1017/pds.2024.44

Introducing a multipliable BOM-based automatic definition of information retrieval in plant engineering

Max Layer [1,✉], Sebastian Neubert [1] and Ralph Stelzer [2]

[1] Siemens Energy Global GmbH & Co.KG, Germany, [2] Technische Universität Dresden, Germany

✉ max.layer@siemens-energy.com

Abstract

The complexity of process plants and the growing demand for digitalization require efficient and accurate information retrieval throughout the lifecycle phases of a process plant. This paper discusses the concept of instantiation and introduces a method for identifying and multiplying required information in plant engineering using scalable so-called Instantiation Blocks linked to the Bill of Material. Core functionality, an ontology graph and a user interface based on Python and React are developed to demonstrate the implementation of the framework and validate its effectiveness in practice.

Keywords: information retrieval, knowledge management, plant engineering, natural language processing (NLP), semantic modelling

1. Introduction

The increasing complexity of process plants and the growing demand for digitalization have created a need for efficient and accurate information retrieval from design phase (Sierla *et al.*, 2021), but also during the construction, commissioning and operation of process plants. However, current practices in plant engineering often struggle with managing this complexity, leading to inefficiencies and errors. In addition, the process industry is experiencing a transition from a traditional engineering-to-order approach (Ergen *et al.*, 2007) to a more volume-based system. Through modularization, reference plants can be used for similar, following projects (Leidich, 2022). In context of the information accessibility, virtual representations of manufacturing and maintenance require high quality data accuracy representing the real-time physical counterpart (Rojek *et al.*, 2021). Furthermore, current information retrieval approaches based on statistical methods and keyword matching are not directly applicable to the engineering domain (Li *et al.*, 2007b).

The authors suggest that predefining information retrieval improves information backflow by capturing and managing essential data throughout plant lifecycle phases, ensuring accurate representation and traceability of the physical plant and components. By identifying and organizing the required information in advance, plant designers and engineers can streamline the data collection process, reduce the time and effort needed to gather and process information from various sources. Furthermore, it enables plant designers and engineers to modify and optimize information retrieval processes for different projects based on a reference, helping to reduce inefficiencies and surplus efforts.

This paper builds up on a conducted systematic review of data management in process industry (Layer *et al.*, 2023a), a generic approach of pre-defining information retrieval (Layer *et al.*, 2023c) and a first approach to automatically identify relevant information (Layer *et al.*, 2023b). A novel framework is investigated, that focuses on the automatic definition of information retrieval by utilizing so-called Instantiation Blocks (IB), acting as placeholders and as a definition of to-be instantiated information, both connected to the Bill of Material (BoM) of process plants as shown in Figure 1A. An IB is connected to either system or component level in a BoM structure and hold the information of the properties of information to be retrieved in a later stage, such as commissioning.

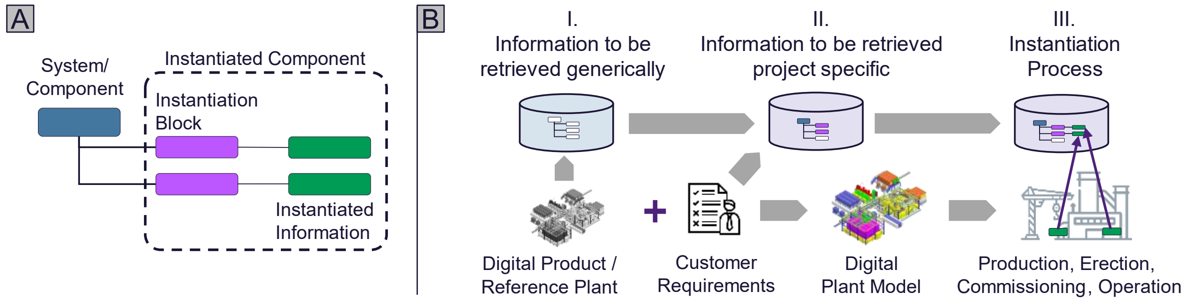

Figure 1. A - Instantiation Block architecture; B - Proposed concept for defining information retrieval introduced in Layer *et al.* (2023c)

The overall concept is based on three steps as shown in Figure 1B, where first, information to be retrieved is identified generically and connected with components of a reference plant, second, project specific adaptations are made and third, the information is retrieved through an instantiation process. During this instantiation process, the actual information is connected to this IB, connecting all instance specific information with their respective article counterpart. The relationship between instance specific information and instantiation is further explained in section 3. The methodology aims to be integrated into an overall plant lifecycle management approach, where all information of different origin, stored in various native data management systems are linked through a middleware in a holistic digital representation. The concept has been introduced by Saske *et al.* (2022) and was further detailed out by Schwoch *et al.* (2023). This paper seeks to answer the research question: How can a framework for automating the definition of information retrieval in plant engineering improve efficiency and accuracy in managing plant complexity?

2. State of the art

2.1. Information retrieval in plant engineering

In plant engineering, standardized frameworks and data exchange formats like ISO 15926, AutomationML (AML), System Modeling Language (SysML), ISO 23247 and Industry Foundation Classes (IFC) address information retrieval challenges. ISO 15926-1 (2004) is an international standard that defines a data model and ontology for the representation and exchange of process plant lifecycle information. It provides a standardized approach to capturing, storing, and sharing information related to process plant design, construction, operation, and decommissioning. However, its focus on plant lifecycle information means that it does not specifically address the aspect of predefining information retrieval. In context of volume driven production, ISO 23247-1 (2021) introduces a framework for developing Digital Twins of observable manufacturing elements to assist with real-time control, predictive maintenance and other function objectives focused on manufacturing. SysML is a generalized modeling language that supports the specification, analysis, design, verification, and validation of a broad range of systems and systems-of-systems (Friedenthal et al., 2015).
The Quality Information Framework (QIF) as part of ISO 23952 (2020) on the other hand, is an ANSI standard that provides a comprehensive and integrated approach to defining, organizing, and exchanging quality information throughout the manufacturing process. It focuses on creating a standardized data model for representing quality-related information, including product and manufacturing information, measurement plans, and quality inspection results. While QIF is primarily targeted at the discrete quality domain within manufacturing industry, its principles and concepts can also be applied to the process plant domain. AML is a data exchange format based on XML that aims to provide a comprehensive and unified approach to representing engineering data across different domains, including mechanical,

electrical, and software engineering and can be adapted to represent a wide range of information, including component specifications, system architectures, and process models. (Drath, 2021)

2.2. Approaches for predefining and automating information retrieval

In literature, approaches and concepts exist for predefining and automating information retrieval, ranging from manual methods to more advanced techniques. Rule-based systems can be employed to extract specific data from documents based on predefined rules or patterns, reducing the manual effort required (Hayes-Roth, 1985). Fully automated approaches rely on advanced algorithms and machine learning techniques to identify, classify, and retrieve relevant information with minimal human interaction. Natural language processing (NLP) and text mining techniques can be used to automatically extract information from unstructured data sources, such as documents (Cascini *et al.*, 2004) or technical reports (Feng *et al.*, 2021). Lee *et al.* (2022) suggested a framework to explain the sources of data, their nature and the information challenge particularly facing the considerations of process safety. Their P3-model focuses on people, plant and procedures within the field of process safety. Safety as key concern of planning processes in process plant industry is also investigated by Penteado and Ciric (1996). Azarmipour *et al.* (2020) proposed a secure gateway for the interaction between information technology and automation systems, focusing on the process control system. Their architecture consists of three main components: the process control system, a gateway, and a manufacturing cloud. The integrity of their Verification of Request (VOR) and checking its plausibility is accomplished by using a Digital Twin (DT) of the process control system.

2.3. Ontologies and semantic modeling

Ontologies and knowledge graphs are often mentioned together due to their similar visual representations as nodes and edges. Both are based on the Resource Description Framework (RDF) triples for representing entities and relationships. However, ontologies specifically serve to create a formal representation of the entities in the graph, often based on a taxonomy. Since ontologies can contain multiple taxonomies, they maintain their separate definition. (Fensel *et al.*, 2020)

In context of engineering and industrial information retrieval Li *et al.* (2007a) proposed a methodology for developing an engineering ontology to structure unstructured engineering documents and improve information retrieval. Yao *et al.* (2009) presented an ontology-based framework for multi-source engineering information retrieval in an integrated enterprise environment. Sriti *et al.* (2015) introduced SPIKE, an ontology-based methodology for exchanging product lifecycle information across various systems. An ontology-based approach for PLM was discussed in the research of Matsokis and Kiritsis (2010). Pierra (2006) explores the role of ontologies in data integration and presents the PLIB ontology model for neutral exchange and automatic integration of industrial component catalogues and technical data. The utilization of ontologies and graph networks has been investigated in maintenance planning by Xia *et al.* (2023).

While the adoption of neural networks on graphs was already introduced in 2005 Gori *et al.* (2005), in recent years there has been extended research in this area especially in the field of recommender systems (Tang *et al.*, 2023; Zhao *et al.*, 2023). Semantic similarity, a measure of the degree to which two entities are related in meaning or concept is also utilized in context of recommenders and ontologies (Lu *et al.*, 2015).

3. Definition of instance specific information and instantiation

Instantiation is a core concept in computer programming, particularly in the object-oriented programming paradigm and refers to the creation of an object from a class (McAllester and Zabih, 1986). Beyond programming, the concept of instantiation has been employed to describe the process of creating specific embodiments of concepts (Gorecky *et al.*, 2016), models (Ballesteros, 2004; Kashmar *et al.*, 2023) or Digital Twins (Moenck *et al.*, 2022; Slot *et al.*, 2020). Instantiation, in this broader sense, involves applying a general framework, template, or concept to a specific situation or context.

In the context of engineering information, the authors differentiate between article specific information and instance specific information related to physical components and systems. This differentiation has already been outlined by Schwoch *et al.* (2023) and is further displayed in Figure 2. Article specific information pertains to general data, properties and specifications, that apply to all instances of a particular type of component or system and is traditionally managed in Product Lifecycle Management (PLM) systems (Eigner and Stelzer, 2009). Instance specific information, conversely, refers to data and properties unique to a particular instance or realization of a physical component or system. While commonly known representatives of this instance specific information are quality information, serial numbers and result data in production environments, there are multiple more of these in plant engineering. These consist of parametrization values of valves for automation control, documentation of conducted tests during commissioning phase or the as-built line-out.

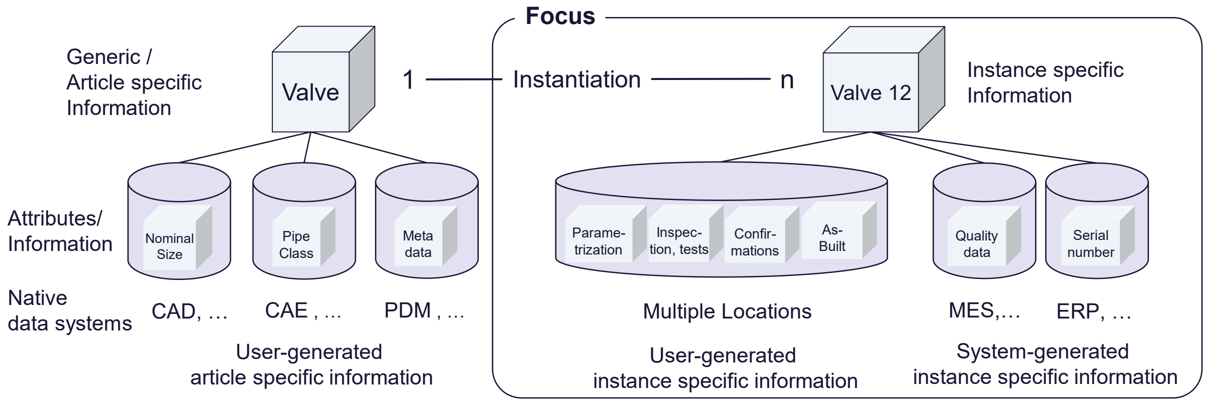

Figure 2. Differentiation of article specific and instance specific information

Instantiation refers to the process of providing and supplying instance-specific information about physical components and systems, which is crucial for their effective management, maintenance, and optimization. The process encompasses several steps, including identifying relevant article-specific information, determining necessary instance properties, collecting and recording instance-specific information, organizing the information for easy retrieval, and ensuring its accuracy and up-to-dateness. Information backflow, in this context, pertains to capturing and managing essential data emanating from production, construction, commissioning phases, and plant operation to guarantee accurate representation of traceability or Digital Twins of the physical plant and its components. The proposed framework foundation identified different categories of information backflow, each necessitating predefined specific information retrieval methods based on a general architecture (Layer *et al.*, 2023c). The overall process of creating and multiplying these modular IBs is outlined in the following.

4. Introducing a method of multiplying required information

4.1. Identification of relevant information

The extraction and identification of relevant information from various sources, such as documents, norms, standards, or expert knowledge, can be conducted manually or through automated means. Existing literature approaches, such as Gräßler *et al.* (2023), Luttmer *et al.* (2023) as well as our own research (Layer *et al.*, 2023b) demonstrate the possibilities of automating and streamlining this process. Regardless of the degree of automation, the identified required information should be linked with their origin while this linkage should be maintained even when the BoM structure is applied. The first part of the overall process is illustrated in Figure 3. For this and the following, an abstracted Business Process Model and Notation (BPMN) notation is chosen and explained in the legend. In the initial stage, information sources defining requirements, such as norms, standards, or expert knowledge as well as customer requirements or existing manuals or documentation need to be analyzed either manually or using automated methods. If existing documents contain the retrieved knowledge, the information may not be in a structured format. In both cases, the information analysis process aims to identify relevant information items containing components, tests, references, and other related entities. These identified items are transformed into a structured, ontology-based semantic network.

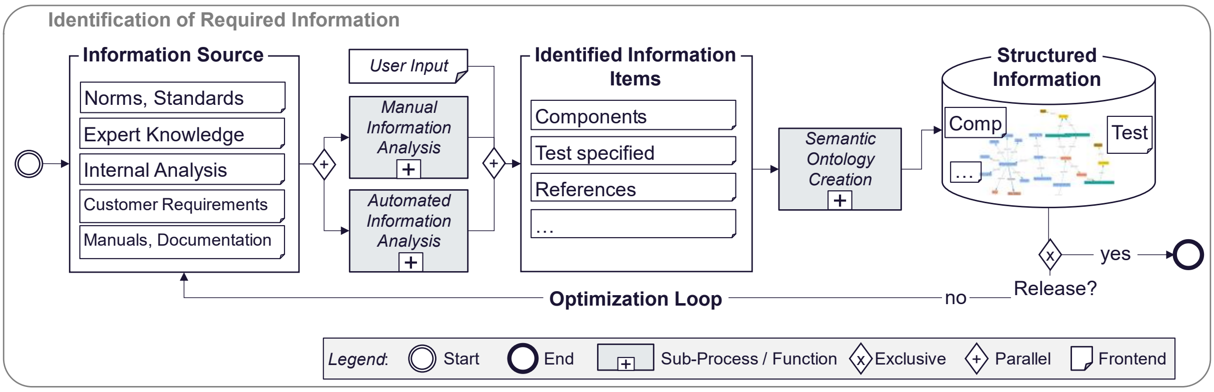

Figure 3. Identification of required information

A self-programmed *InformationAnalysis* module is responsible for the manual or automated analysis of information sources and the extraction of relevant entities. For automated analysis, we use a self-fine-tuned Bidirectional Encoder Representations from Transformers (BERT) classifier (Devlin *et al.*, 2018) to identify requirements in sentences and a fine-tuned Named Entity Recognition (NER) model for identifying components, references, and tests or tasks. This pipeline has been showcased in Layer *et al.* (2023b) and was further developed. The extracted entities and classifications are used to create and populate an ontology-based semantic network, holding the information for the later creation of new IBs. For this, RDF triples are utilized to model the structure and dynamics of the system components, tests and origins, as well as their interdependencies. The python rdflib library was utilized to create and manipulate the ontology, while an *OntologyHandler* class is responsible for ontology creation, population, and querying. The function accepts both the analysis results or user input from a frontend in the form of a list of dictionaries, where each dictionary contains information about components, IBs, and other related entities.

4.2. Setting up the reference IB structure based on structured information

Based on the structured information stored in the graph network, a reference structure of IBs is being created as shown in Figure 4. This involves an information differentiation method to determine precise and vague definitions of IBs, a recommender algorithm to suggest existing or new IBs and a position analysis function to understand the relative position of IBs in the reference plant structure.

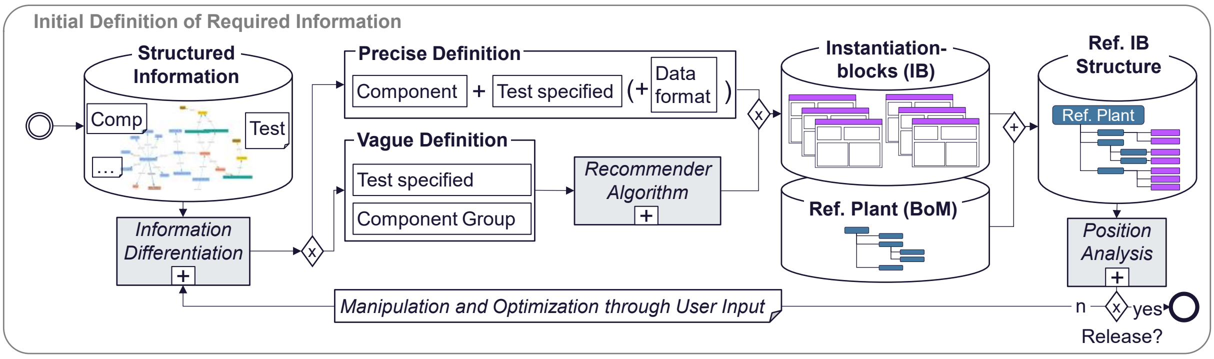

Figure 4. Initial definition of required information

The information differentiation method analyzes the structured IBs in the ontology to identify defined information retrieval and vague definition. Precise definitions have sufficient information to directly create IBs out of them and then be directly connected to the components in the reference plant structure. In contrast, vague definition requires further expert knowledge to determine the most appropriate IBs and connections. For vague definition IBs, we utilize the recommender function, which is a core component of *InformationAnalysis*. This function is responsible for recommending IBs based on specific input criteria. The identified IBs, both precise and recommended, are created and transferred to a relational database. This creation of IBs entails an iterative approach, that use existing IBs as a

foundation for the recommender algorithm. By employing SPARQL queries on the existing RDF triplestore, recommendations for identical or similar IBs can be generated based on extracted entities and classifications. A pre-trained BERT model is used to calculate embeddings for both the new and the available IBs. By measuring the cosine similarity between the new embedding and the IB embeddings, the IBs are ranked based on their relevance to the user's query. Consequently, the function returns a list of top-k most similar IBs with similarity scores to aid users in selecting the most suitable IBs for their needs.
Once the new IBs have been identified and the recommendations have been made, the *OntologyManager* class is also responsible for updating the ontology graph accordingly. This includes adding new instances and relationships based on the extracted entities and classifications, as well as updating existing instances and relationships if necessary. Its responsibilities extend beyond updating the ontology graph to include creating and querying the ontology, ensuring that the graph accurately reflects the most up-to-date information and relationships between the components, IBs, and other entities in the system.
The newly created IBs are combined with the product reference plant BoM structure by either linking them directly or matching by component class. The choice between these methods is determined by several parameters, such as the nature of the information, the type of component, and the specific requirements of the project. Together, they form the reference IB structure which after a position analysis and optimization loop serve as the foundation for the information backflow process as a template for incoming projects based on the reference.
The position analysis function is employed to analyze the relative position of IBs within the reference plant structure. This analysis helps to understand the hierarchies and relationships between the IBs and the components they are connected to. The position analysis function examines the reference IB structure and calculates the relative position of each IB based on their connections to the components, as well as the hierarchical and dependency relationships among the components themselves. This information can be later used to optimize the information backflow process, prioritize the most critical IBs, and identify potential areas for improvement.

4.3. Automatic definition of required Information

Based on the derived reference IB structure, we suggest to automatically define required information for a project specific plant structure represented by a BoM. This involves a similarity analysis function identifying and recommending relevant IBs for adapted or newly added components which have not been in the reference BoM because to customer adaptations. Afterwards, a final manipulation of all project specific IBs is conducted leading to the project IB structure as shown in Figure 5.

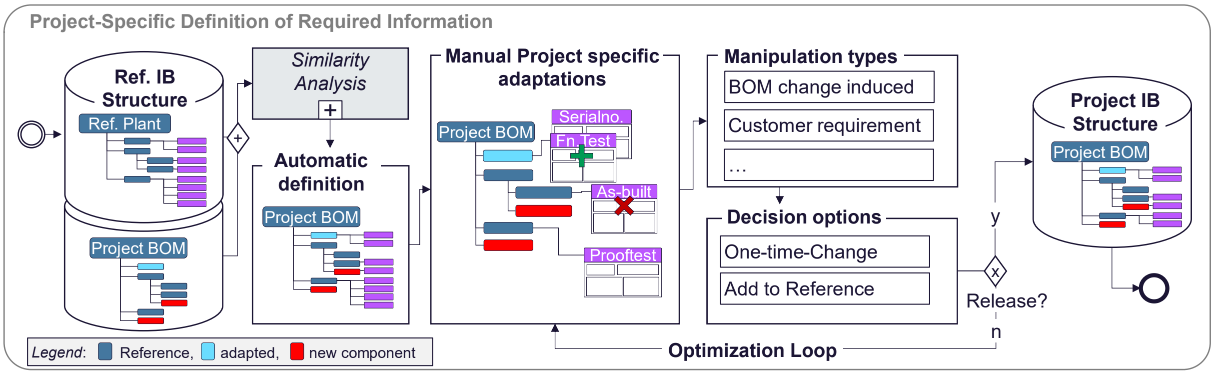

Figure 5. Project specific definition process

The first step in the automatic definition of required information is to analyze the project specific plant structure or BOM to identify newly added components and their attributes. The similarity analysis function compares the newly added components in the project specific BOM to the existing components in the reference ontology. This comparison is based on all component attributes, such as component classes, data formats, information types, or other relevant properties. The function calculates similarity

scores using string matching algorithms, semantic similarity measures and vector-based similarity for numeric values. The function returns a list of recommended IBs for each new component, along with their similarity scores or other relevant information, which can be used to assist users in selecting the most appropriate IBs for their needs.
For manual adaptations of existing IBs stored in a project specific relational DB, users can add, modify, or remove IBs through a drag-and-drop frontend to fulfill project requirements. Upon each adaptation, a manipulation type questionnaire is asked to determine if the change is induced by a BoM-change, customer requirements or similar. The user responses guide the treatment of changes as either one-time alterations or triggers for feedback loops through an update function to propagate relevant changes back into the relational DB and reference ontology, ensuring its up-to-dateness and integration of valuable project-specific insights.

4.4. Information retrieval based on project specific IBs

In this section, we discuss the process of information retrieval based on the finalized project specific IB structure. This process is conducted in parallel to the lifecycle phases such as production, erection, and commissioning phases until no open IBs are left forming the Digital Representation of the plant. We also introduce a "context information retrieval" questionnaire for user feedback on ad-hoc information that was not predefined by IBs.

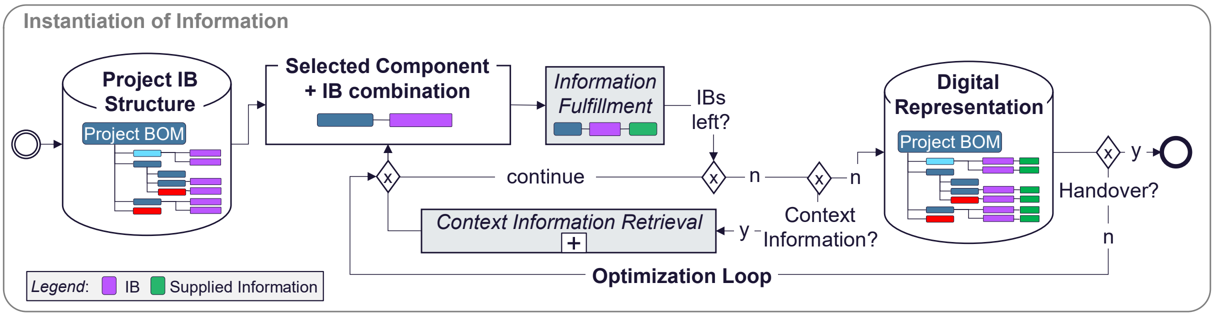

Figure 6. Instantiation of information

The information fulfillment process is conducted iteratively, with the project IB structure serving as the foundation. Both the ontology and the frontend interface are used to assist users in gathering and connecting instance specific information to the respective components and IBs. This process takes place in parallel to the overall project, ensuring that information backflow and validation are continuously updated until handover to the customer.
The finalized project specific IB structure also serves as a source of maintenance information, enabling plant operators and maintenance personnel to access relevant details about the components and their relationships. This information can be used for preventive and corrective maintenance tasks, ensuring the plant's optimal performance and reliability.

4.5. Utilizing the internal element structure of AML for data exchange

AML is employed to represent components and systems in an automation project. The AML model can be extended to incorporate instance-specific information by introducing elements like InstanceAttributes and InstanceHistory under the InternalElement structure. This allows storing unique properties and historical records for each instance, such as identifiers, installation dates, or maintenance records, and managing this information effectively. The proposed instantiation framework can be exchanged and stored using AML by defining article-specific information using RoleClassLib instances, extending the InternalElement structure, and developing a consistent methodology for data management within the AML model. This leverages AML's extensibility and integration capabilities to enable seamless data exchange and interoperability between various engineering tools and systems.
A sample AML file as an illustrative example is provided in Figure 7, representing a **ControlValveInstance1** with a **Parametrization** IB connected through the RoleClassLib **ControlValve** item.

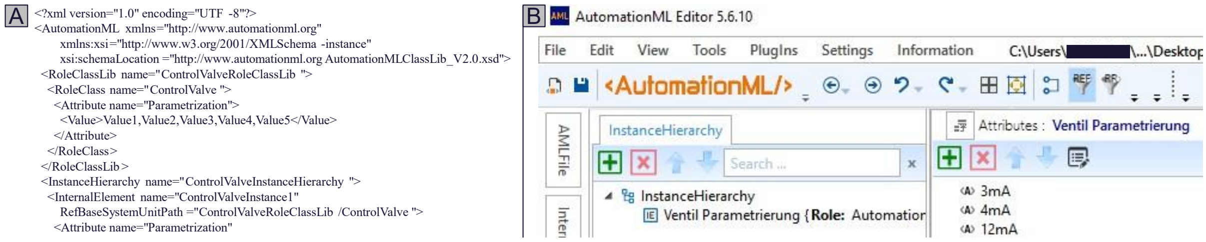

Figure 7. AutomationML file as raw (left) and in viewer (right)

5. Results and discussion

To demonstrate the proposed framework's practical implementation, we developed a React-based user interface, allowing users to access and execute the backend functionalities described in section 4. The frontend, as shown in Figure 8, includes functionalities for running the analyzer, displaying the semantic network, and manipulating the reference IB structure. Lastly, through an instantiation frontend, various information (e.g. conducted tests) were captured exemplary and stored in a database.

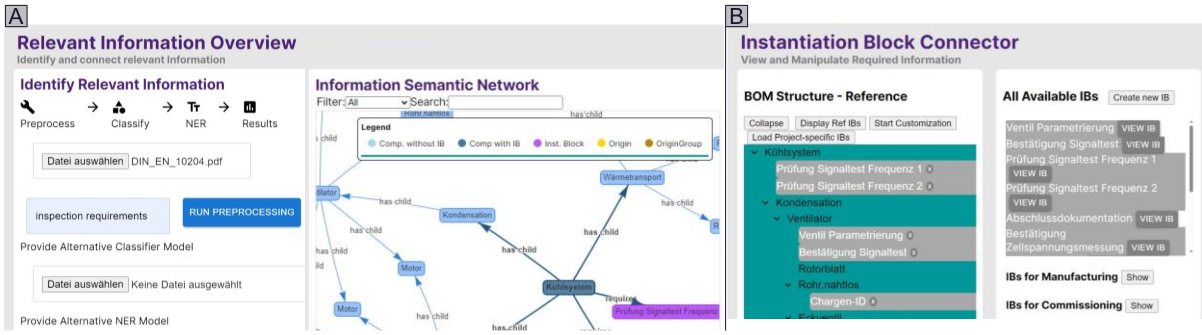

Figure 8. Developed frontend snippets for A: Executing the analyzer functionality and displaying the semantic network; B: manipulating the Ref. IB structure by drag-and-drop

By leveraging the BoM and multipliable IB, the framework enabled efficient and accurate information retrieval and instantiation processes, ultimately leading to improved plant performance and sustainability. This integration ensures that the framework can adapt to the evolving needs of the plant engineering industry and drive innovation in the field of predefining information backflow.

6. Conclusion and outlook

This paper addresses the increasing complexity of process plants and the growing demand for digitalization and traceability, both requiring efficient and accurate information retrieval throughout the lifecycle of process plants. It introduces a novel framework for automating the definition of information retrieval in plant engineering by leveraging the Bill of Material and multipliable so-called Instantiation Blocks. A method of multiplying required information through several steps is discussed, starting with the identification of relevant information, creating an ontology, setting up the reference IB structure, automatically defining required information project specific and ultimately, retrieving the instance specific information. The implementation of the framework relies on libraries and tools for information analysis, ontology handling, similarity analysis, and manipulation of AML files as an exchange format. The successful implementation and adoption of this framework was analyzed on exemplary data. Furthermore, the development of a React-based user interface demonstrator facilitated practical implementation and validation of the framework in real-world scenarios.

Future research directions include refining the methodologies proposed in this paper and expanding the framework's capabilities to handle more complex information. Additionally, investigating the integration of graph neural networks and other machine learning techniques could further enhance the automation and accuracy of the information retrieval process. Ultimately, the continuous improvement and adaptation of the framework will be essential for meeting the evolving needs of the plant engineering industry and driving innovation in the field of predefining information backflow.

References

Azarmipour, M., Trotha, C. von, Gries, C., Kleinert, T. and Epple, U. (2020), "A Secure Gateway for the Cooperation of Information Technologies and Industrial Automation Systems", in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, 18.10.2020 - 21.10.2020, Singapore, Singapore*, IEEE, pp. 53–58, https://dx.doi.org/10.1109/IECON43393.2020.9254634

Ballesteros, J.R. (Ed.) (2004), *Meta-model instantiation for geoscientific data collection*, ITC.

Cascini, G., Fantechi, A. and Spinicci, E. (2004), "Natural Language Processing of Patents and Technical Documentation", in Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., Sudan, M., Terzopoulos, D., Tygar, D., Vardi, M.Y., Weikum, G., Marinai, S. and Dengel, A.R. (Eds.), *Document Analysis Systems VI*, *Lecture Notes in Computer Science*, Vol. 3163, Springer Berlin Heidelberg, pp. 508–520, https://dx.doi.org/10.1007/978-3-540-28640-0_48

Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018), *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, https://dx.doi.org/10.48550/arXiv.1810.04805

Drath, R. (2021), *Automationml: A practical guide*, *De gruyter textbook*, 1st ed., De Gruyter Oldenbourg, Boston.

Eigner, M. and Stelzer, R. (2009), *Product lifecycle management: Ein Leitfaden für product development und life cycle management*, Springer Science & Business Media.

Ergen, E., Akinci, B. and Sacks, R. (2007), "Life-cycle data management of engineered-to-order components using radio frequency identification", *Advanced Engineering Informatics*, Vol. 21 No. 4, pp. 356–366, https://dx.doi.org/10.1016/j.aei.2006.09.004.

Feng, X., Dai, Y., Ji, X., Zhou, L. and Dang, Y. (2021), "Application of natural language processing in HAZOP reports", *Process Safety and Environmental Protection*, Vol. 155, pp. 41–48, https://dx.doi.org/10.1016/j.psep.2021.09.001.

Fensel, D., Şimşek, U., Angele, K., Huaman, E., Kärle, E., Panasiuk, O., Toma, I., Umbrich, J. and Wahler, A. (2020), *Knowledge Graphs*, Springer International Publishing, Cham, https://dx.doi.org/10.1016/j.psep.2021.09.001

Friedenthal, S., Moore, A. and Steiner, R. (2015), A practical guide to SysML: The systems modeling language, Third edition, Elsevier MK Morgan Kaufmann is an imprint of Elsevier, Amsterdam, Boston.

Gorecky, D., Weyer, S., Hennecke, A. and Zühlke, D. (2016), "Design and Instantiation of a Modular System Architecture for Smart Factories", *IFAC-PapersOnLine*, Vol. 49 No. 31, pp. 79–84, https://dx.doi.org/10.1016/j.ifacol.2016.12.165.

Gori, M., Monfardini, G. and Scarselli, F. (2005), "A new model for learning in graph domains", in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005, 31 July-4 Aug. 2005, Montreal, Que., Canada*, IEEE, pp. 729–734, https://dx.doi.org/10.1109/IJCNN.2005.1555942

Gräßler, I., Ozcan, D. and Preuß, D. (2023), "AI-based extraction of requirements from regulations for automotive engineering", https://dx.doi.org/10.35199/dfx2023.17

Hayes-Roth, F. (1985), "Rule-based systems", *Communications of the ACM*, Vol. 28 No. 9, pp. 921–932.

ISO 15926-1 (2004), *Industrial automation systems and integration: Integration of life-cycle data for process plants including oil and gas production facilities - Part 1: Overview and fundamental principles*, Vol. 25.040.40 No. ISO 15926:2004, ISO copyright office, Switzerland (accessed 16 November 2022).

ISO 23247-1 (2021), *Automation systems and integration: Digital twin framework for manufacturing - Part 1: Overview and general principles*, Vol. 25.040.40 No. ISO 23247:2021, ISO copyright office, Switzerland

ISO 23952 (2020), *Automation systems and integration — Quality information framework (QIF): An integrated model for manufacturing quality information*, Vol. 25.040.40 No. ISO 23952:2020, ISO copyright office, Switzerland (accessed 31 October 2022).

Kashmar, N., Adda, M., Ibrahim, H., Morin, J.-F. and Ducheman, T. (2023), "Instantiation and Implementation of HEAD Metamodel in an Industrial Environment: Non-IoT and IoT Case Studies", *Electronics*, Vol. 12 No. 15, p. 3216, https://dx.doi.org/10.3390/electronics12153216.

Layer, M., Leidich, J., Schwoch, S., Saske, B., Neubert, S., Robl, P. and Paetzold-Byhain, K. (2023a), "Data management of process plants as complex systems: systematic literature review and identification of challenges and opportunities", *Reviews in Chemical Engineering*, No. aop, https://dx.doi.org/10.1515/revce-2022-0077.

Layer, M., Neubert, S., Boda, B. and Stelzer, R. (2023b), "Towards a Framework for Identifying Relevant Information in regard to Specific Context on the Use Case of Standards and Directives".

Layer, M., Neubert, S., Tiemann, L. and Stelzer, R. (2023c), "Identification and Retrieval of Relevant Information for Instantiating Digital Twins during the Construction of Process Plants", *Proceedings of the Design Society*, Vol. 3, pp. 2175–2184, https://dx.doi.org/10.1017/pds.2023.218.

Lee, J., Cameron, I. and Hassall, M. (2022), "Information needs and challenges in future process safety", *Digital Chemical Engineering*, Vol. 3, p. 100017, https://dx.doi.org/10.1016/j.dche.2022.100017.

Leidich, J. (2022), “Optimized planning of the integration of a Reference Plant into existing brownfield environments based on an entity model”, in *DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022), 22 and 23 September* 2022, The Design Society, p. 10, https://dx.doi.org/10.35199/dfx2022.14

Li, Z., Raskin, V. and Ramani, K. (2007a), “A methodology of engineering ontology development for information retrieval”, in *DS 42: Proceedings of ICED 2007, the 16th International Conference on Engineering Design, Paris, France, 28.-31.07.* 2007, The Design Society, pp. 429–430.

Li, Z., Raskin, V. and Ramani, K. (2007b), “Developing Ontologies for Engineering Information Retrieval”, in *Volume 2: 27th Computers and Information in Engineering Conference, Parts A and B, 04.09.2007-07.09.2007, Las Vegas, Nevada, USA*, ASMEDC, pp. 737–745, https://dx.doi.org/10.1115/DETC2007-34530

Lu, J., Wu, D., Mao, M., Wang, W. and Zhang, G. (2015), “Recommender system application developments: A survey”, *Decision Support Systems*, Vol. 74, pp. 12–32, https://dx.doi.org/10.1016/j.dss.2015.03.008.

Luttmer, J., Prihodko, V., Ehring, D. and Nagarajah, A. (2023), “Requirements extraction from engineering standards – systematic evaluation of extraction techniques”, *Procedia CIRP*, Vol. 119, pp. 794–799, https://dx.doi.org/10.1016/j.procir.2023.03.125.

Matsokis, A. and Kiritsis, D. (2010), “An ontology-based approach for Product Lifecycle Management”, *Computers in industry*, Vol. 61 No. 8, pp. 787–797, https://dx.doi.org/10.1016/j.compind.2010.05.007.

McAllester, D. and Zabih, R. (1986), “Boolean classes”

Moenck, K., Laukotka, F., Krause, D. and Schüppstuhl, T. (2022), “Digital Twins of existing long-living assets: reverse instantiation of the mid-life twin”, in *DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022), 22 and 23 September* 2022, The Design Society, p. 10, https://dx.doi.org/10.35199/dfx2022.20

Penteado, F.D. and Ciric, A.R. (1996), “An MINLP Approach for Safe Process Plant Layout”, *Industrial & Engineering Chemistry Research*, Vol. 35 No. 4, pp. 1354–1361, https://dx.doi.org/10.1021/ie9502547.

Pierra, G. (2006), “Context-explication in conceptual ontologies: Plib ontologies and their use for industrial data”, *Journal of Advanced Manufacturing Systems*, Vol. 5, pp. 243–254.

Rojek, I., Mikołajewski, D. and Dostatni, E. (2021), “Digital Twins in Product Lifecycle for Sustainability in Manufacturing and Maintenance”, *Applied Sciences*, Vol. 11 No. 1, p. 31, https://dx.doi.org/10.3390/app11010031.

Saske, B., Schwoch, S., Paetzold, K., Layer, M., Neubert, S., Leidich, J. and Robl, P. (2022), “Digitale Abbilder als Basis Digitaler Zwillinge im Anlagenbau: Besonderheiten, Herausforderungen und Lösungsansätze”, *Industrie 4.0 Management*, Vol. 2022 No. 5, pp. 21–24, https://dx.doi.org/10.30844/IM_22-5_21-24.

Schwoch, S., Leidich, J., Layer, M., Saske, B., Paetzold-Byhain, K., Robl, P. and Neubert, S. (2023), “A conceptual framework for information linkage and exchange throughout the lifecycle of process plants”, https://dx.doi.org/10.35199/dfx2023.25.

Sierla, S., Azangoo, M., Rainio, K., Papakonstantinou, N., Fay, A., Honkamaa, P. and Vyatkin, V. (2021), “Roadmap to semi-automatic generation of digital twins for brownfield process plants”, *Journal of Industrial Information Integration*, p. 100282, https://dx.doi.org/10.1016/j.jii.2021.100282.

Slot, M., Huisman, P. and Lutters, E. (2020), “A structured approach for the instantiation of digital twins”, *Procedia CIRP*, Vol. 91, pp. 540–545, https://dx.doi.org/10.1016/j.procir.2020.02.211.

Sriti, M.F., Assouroko, I., Ducellier, G., Boutinaud, P. and Eynard, B. (2015), “Ontology-based approach for product information exchange”, *International Journal of Product Lifecycle Management*, Vol. 8 No. 1, p. 1, https://dx.doi.org/10.1504/IJPLM.2015.068011.

Tang, H., Wu, S., Xu, G. and Li, Q. (2023), “Dynamic Graph Evolution Learning for Recommendation”, in Chen, H.-H., Duh, W.-J., Huang, H.-H., Kato, M.P., Mothe, J. and Poblete, B. (Eds.), *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 23 07 2023 27 07 2023, Taipei Taiwan*, ACM, New York, NY, USA, pp. 1589–1598, https://dx.doi.org/10.1145/3539618.3591674

Xia, L., Liang, Y., Leng, J. and Zheng, P. (2023), “Maintenance planning recommendation of complex industrial equipment based on knowledge graph and graph neural network”, *Reliability Engineering & System Safety*, Vol. 232, p. 109068, https://dx.doi.org/10.1016/j.ress.2022.109068.

Yao, Y., Lin, L. and Dong, J. (2009), “Research on Ontology-Based Multi-source Engineering Information Retrieval in Integrated Environment of Enterprise”, in 2009 *International Conference on Interoperability for Enterprise Software and Applications China, 21.04.2009 - 22.04.2009, Beijing, China*, IEEE, pp. 277–282, https://dx.doi.org/10.1109/I-ESA.2009.25

Zhao, Z., Zhu, X., Xu, T., Lizhiyu, A., Yu, Y., Li, X., Yin, Z. and Chen, E. (2023), “Time-interval Aware Share Recommendation via Bi-directional Continuous Time Dynamic Graphs”, in Chen, H.-H., Duh, W.-J., Huang, H.-H., Kato, M.P., Mothe, J. and Poblete, B. (Eds.), *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 23.07*.2023-*27.07*.2023*, Taipei Taiwan*, ACM, New York, NY, USA, pp. 822–831, https://dx.doi.org/10.1145/3539618.3591775