A conceptual framework for information linkage and exchange throughout the lifecycle of process plants

Sebastian Schwoch[1], Jonathan Leidich[2], Max Layer[3], Bernhard Saske[1], Kristin Paetzold-Byhain[1], Peter Robl[2], Sebastian Neubert[3]

[1] TUD Dresden University of Technology, Dresden, Germany
[2] Siemens AG, Munich, Germany
[3] Siemens Energy GmbH & Co.KG, Erlangen, Germany

* Corresponding Author:
 Dipl.-Ing. Sebastian Schwoch
 TUD Dresden University of Technology
 Faculty of Mechanical Science and Engineering
 Institute of Machine Elements and Machine Design
 Chair of Virtual Product Development
 01062 Dresden
 ☎ *+49 351 - 463 36721*
 ✉ *sebastian.schwoch@tu-dresden.de*

Abstract

Based on the need of unified access to process plant related information across disciplines, companies and lifecycle phases as well as efforts to establish methods of modularization and standardization, this paper introduces a conceptual framework to address these challenges. Based on a systematic literature review and an analysis of data and information flows of two companies, a comprehensive list of requirements is derived. A conceptual framework is proposed based on these requirements and consisting of four linked structures with distinct types of elements. Also, the concept of metadata management within the framework is illustrated which utilizes interfaces to the native Data Management Systems. These interfaces are used to retrieve source data of asset attributes managed within the framework.

1. Introduction and Motivation

Digitalization and the advances of Industry 4.0 have manifested a data-based industry in which process plants generate a wealth of heterogeneous but content-networked data throughout their lifecycle. This spectrum of data includes, among other things, development, production, commissioning, and operational data. Despite this wealth of information, current projects are often individually tailored to the customer, which results in a high number of variants, low standardization, and little reuse of engineering results. This individual approach is reinforced by a variety of specific authoring and data management systems that produce different partial models and structures of process plants within the development phase [1, 2].

In many cases, significant parts of the plants are developed and produced by suppliers according to specified requirements. However, the final plant is assembled from these components in factories and on site, commissioned and handed over to the customer. This method poses challenges, especially in terms of capturing and merging documentation of the as-built condition of the assets in the heterogeneous production environments of factory and site. Inconsistent documentation can affect subsequent service processes such as maintenance and servicing.

Furthermore, there is a need to track asset condition during the operational phase. It may happen that changes are made to the plant structure to optimize operations. These changes need to be synchronized with the existing data of the plant, such as the development data. However, it is apparent that a break in data continuity occurs during the phases of production, construction and operation, and the physical instances of the plant components are not currently comprehensively recorded in a structure. As a result, the potential for using this data is lost. Instance-specific data, i.e. data related to physical components or implemented software, could serve as a reference for digital twins and provide important interface information for data acquisition. They could also be used to optimize plant design or requirements based on real data, but this requires linking instance-specific data with development data, which has been lacking so far [3].

Due to these challenges, there is no complete, consolidated representation of the state of the process plant in its current form along the entire lifecycle. The form of this state varies depending on the lifecycle phase. To manage this complexity and provide a comprehensive representation of the state, a concept for a framework is required. This framework should be able to consolidate and make available the data of the assets across disciplines, companies, and lifecycles. In this context, the idea of a Digital Representation as a unified source of information for all assets is introduced in [3] and will be elaborated further in the following chapters.

2. State of the art and research

2.1. Existing concepts and solutions for Product Lifecycle Management

The structured management of product data arose from the requirements of the digitalization of product development in the 1980s and 1990s. With the introduction of CAD systems in product development, deeply branched hierarchical file structures emerged with a high number of individual parts and assemblies for describing the products. The introduction of product data management systems (PDM systems) was necessary for the consistent and audit-proof storage of data. These requirements applied to companies in the aircraft and automotive industries [4]. Initially, PDM systems were primarily aimed at the management of mechanical products, but quickly reached their limits when it came to mapping branched corporate and communication structures to quickly place products on the market or manage them over their entire lifecycle [5].

With the strategy of Product Lifecycle Management (PLM), these requirements were considered and the management of an integrated product model over the entire lifecycle was

aimed at through the integration of various software systems. In this strategy, PDM systems continue to function as the company's data backbone. Due to the character of the strategic approach of PLM, there are a multitude of versions of PLM that integrate different systems depending on the understanding and purpose of use and are tailored to the requirements of a company. The tasks of the integrated systems often overlap, so that even an identically structured system landscape can require different processes and system interfaces. Especially in the process industry, in addition to the distribution of data via different native Data Management Systems (nDMS), there is also the distribution of data across company boundaries. It is currently not possible to realize a holistic model of a process plant for digitized data exchange across system boundaries. However, for audit-proof operation and efficient scaling during the development and construction of process plants, it is necessary to link the data across the entire lifecycle. An integrated product model of the process plant is required that can be used universally, from article-based development parts lists to instance-based as-built and as-maintained parts lists.

Among other research directions in the context of PLM, the Systematic Review "Data Management of Process Plants as Complex Systems: Systematic Literature Review and Identification of Challenges and Opportunities" highlights new approaches such as the Asset Administration Shell [6]. None of these concepts, however, assumes an approach that only includes and relates information to available metadata and its source system, leaving the data itself in the nDMS. What is needed is an approach that manages as little as possible and rather "observes" and maps.

2.2. Diversity of authoring and data management systems

Within the process plant lifecycle, there is a wide variety of authoring and data management systems in use [7]. In relation to the different lifecycle phases, there is an increase in specialized authoring and data management systems, which creates a conflict of objectives between specialization in the disciplines, and companies and the general need for comprehensive information management. Table 1 looks at the application of different data management systems in the respective subject disciplines and the corresponding lifecycle phases.

Table 1: Overview of data management systems assigned disciplines and lifecycle phases

System Type	Discipline	Lifecycle Phase	Examples
PLM System	Mechanics, Automation Technology	Engineering, Production, Maintenance	Siemens Teamcenter, PTC Windchill, ARAS Innovator, SAP PLM
PDM System	Mechanics	Engineering	Siemens Teamcenter, ISD HELiOS, Dassault Systemes ENOVIA
CAD/CAE System	Process Engineering, Electrical Engineering, Mechanics	Engineering, Production, Maintenance	Siemens COMOS, AVEVA Plant, Siemens Simcenter, ANSYS Workbench, Autodesk Inventor, ISD HiCAD, E-Plan
SCADA System (Supervisory Control and Data Acquisition)	Automation Technology	Production, Monitoring, Maintenance	Siemens WinCC, Rockwell FactoryTalk
DCS System (Distributed Control System)	Automation Technology	Production, Maintenance	Siemens PCS7, ABB 800xA, Honeywell Experion PKS

The increasing diversity of authoring and data management systems poses a challenge as it increases the complexity of data integration and makes data consolidation more difficult across the lifecycle. Despite these challenges, it also opens new opportunities for specific data analysis and use in individual disciplines and lifecycle phases.

3. Research problem and objective

No solution exists to combine the cross-discipline, cross-company, and cross-lifecycle data silos into a common database. The development of an all-encompassing data model is not practical and hard to maintain. The underlying conflict of goals between the unification of data management and the progressive specialization of authoring and data management systems requires the linkage of the distributed data sets via their metadata. A flexible, minimal management concept is needed that is tailored to circumstances in the lifecycle of process plants. The user data of the individual disciplines and companies will continue to be managed in the nDMS and must be made available.

In view of this problem, the need for an intelligent solution that enables the linking of the distributed data stocks becomes clear. Metadata seems to be the promising key to this. It is important to note here that the source data of the individual disciplines and companies will continue to be managed in the nDMS, but that this data must nevertheless be made accessible for a comprehensive overview.

The objective of this research can be divided into three main points. Firstly, the aim is to merge and link the individual sub-models and sub-structures of the process plant over the entire lifecycle. Secondly, a reduction of engineering efforts through the implementation of modularization and standardization is an important concern. Thirdly, the feedback and consolidation of information across all lifecycle phases is of great importance. These aspects lead to the conclusion that a framework is needed that meets certain criteria. It should provide a consistent and complete representation of the process plant within all phases of the plant lifecycle that can be used individually by each discipline. Furthermore, this framework should enable the application of modularization and standardization [8].

The conceptualization and implementation of such a framework raises further research questions:

1) Which boundary conditions and requirements apply for a framework which unifies, joins and makes information of process plants available over their entire lifecycle?
2) Which functions, methods and structures does this framework have to include to fulfil the given requirements?

These research questions are the focus of the following investigations, with the aim of developing an efficient and effective framework. It should meet the specific needs of the various disciplines and companies and enable holistic and efficient data management over the entire lifecycle of process plants.

4. Research methods

A systematic literature review is conducted to summarize the current state of research on data management and data exchange concepts in process plant engineering [6]. Additionally, the processes and information flows of two companies from the process industry are analyzed using extended Event-driven Process Chains.

Using the derived extended Event-driven Process Chains, a generalized reference process of the plant lifecycle is derived in which use cases are defined for every activity with regard to the framework. Furthermore, the state of research is analyzed to define the specific boundary conditions and requirements the framework has to fulfill.

The development of the framework follows optimization cycles in which the data structures and data transformations and related data models, methods, and functions are defined and

then verified against the derived boundary conditions, and requirements. Validation is conducted on consistency, plausibility and accuracy through triangulation with the literature, on feasibility via the implementation of a prototype of the framework and on suitability through experiments with participants from the engineering, production and service phase.

5. Results and discussion

5.1. Identified requirements for the framework

The requirements listed in Table 2 result from the research needs identified in the state of the art (see chapter 2) and the analysis of the current state of cross-discipline and cross-company information and data flows in process plant engineering.

Table 2: Overview of the requirements for the framework

Description
Data consistency
No redundant storage of source data
No redundant management of structures within the framework
Consideration of different structures throughout the lifecycle (As-designed, As-manufactured, As-built und As-maintaned)
Complete representation of all planned and build components of the process plants
Synchronization of As-maintained and As-designed structure (e.g., with structural changes of the built plants)
Traceability of data through the lifecycle of process plants
Provide access to all data throughout the lifecycle of components or assemblies or plants
Linking planned components with their produced and implemented instances
Support of processes and methods
Support of modularization and standardization methods (e.g., VDI standard 2776 [9])
Support of variant management
Support of change management
Support of root cause analysis
Support of building and usage of Digital Twins
Support of the instantiation process while production, installation, and commisioning of plant components
Support of information and data exchange
Linkage of data source distributed over different lifecycle phases, disciplines and companies
Linking of article-based engineering data and instance-based data of the production phase and following lifecycle phases
Maintaining the independency of the nDMS
Read-only access to the nDMS
Support of data export to standardized formats (e.g, AutomationML and AssetAdministrationShell)

The requirements overview in Table 2 shows the diversity and complexity of the requirements which the framework must fullfill. In the following, the key concepts of the framework will be presented.

5.2. Key concepts of the proposed framework

The framework consists of four data structures and their respective elements which are build and transformed into each other after specific rules throughout the lifecycle of the process plants. An example for these structures is shown in Figure 1.
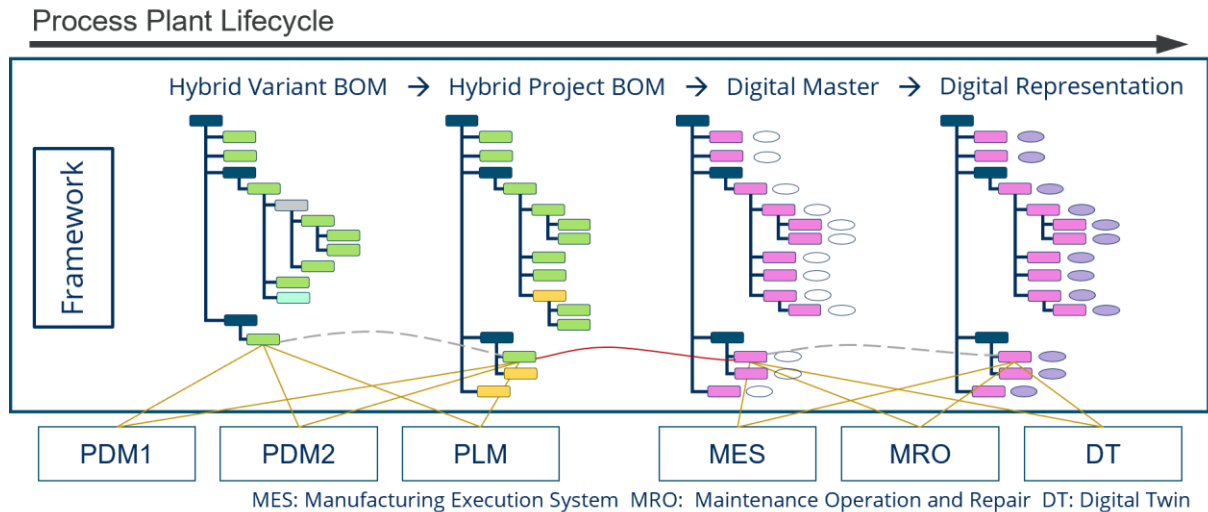

Figure 1: The four datastructures of the framework with their resprective elements and linkage to nDMS

The four structures are the Hybrid Variant Bill of Material (Hybrid Variant BOM), the Hybrid Project BOM, the Digital Master, and the Digital Representation.

The Hybrid Variant BOM and Hybrid Project BOM are used to manage engineering related data as hybrid structures of functions (blue boxes in Figure 1) and articles (green, turquoise and yellow boxes in Figure 1). The Digital Master and the Digital Representation consist of structure of instances (pink boxes in Figure 1). The four structures represent the entire lifecycle of the process plants and are linked to each other (grey and red lines in Figure 1). Hence, a backward and forward data continuity is guaranteed.

Due to the requirement that source data remain in and are managed by nDMS, the framework focuses on the management of metadata. Within this metadata, only the information is stored which type of source data is available in the nDMS. For example, the information that there is a diameter of a pipe available in an nDMS is stored in the corresponding element of the framework. The specific value of the pipe diameter is still stored exclusively in e.g., a PDM system and will be retrieved from the framework via references to this source data to the PDM system.

The references are links from the framework to the specific source data in the nDMS (yellow lines in Figure 1). These references facilitate the web interfaces and Application Programming Interfaces (APIs) of the nDMS to read the source data on demand (see chapter 5.2.2).

5.2.1. The assignment and management of metadata of articles and instances

Figure 2 shows an example of the framework structures with an Article X and two instances Y and Z. Article X is an arbitrary article element within the Hybrid Variant BOM or the Hybrid Project BOM and is embedded in article structures ("Used-In" and "Consists-Of" articles).

Instance Y and Z, on the other hand, are embedded in the instance structures of the Digital Master and in the Digital Representation later in the lifecycle. They each have a parent instance ("Integrated-In"-instance) and may consist of child instances ("Build-Of" instances).

An article may have multiple built or implemented instances as real-world implementations of itself. Within the framework the aricles are linked to their respective instances (red lines and

Figure 1 and 2). This allows the instances of the Digital Master and Digital Representation to access the information associated with their linked article (backward relation). Vice versa, the article knows its real instances (forward relation).
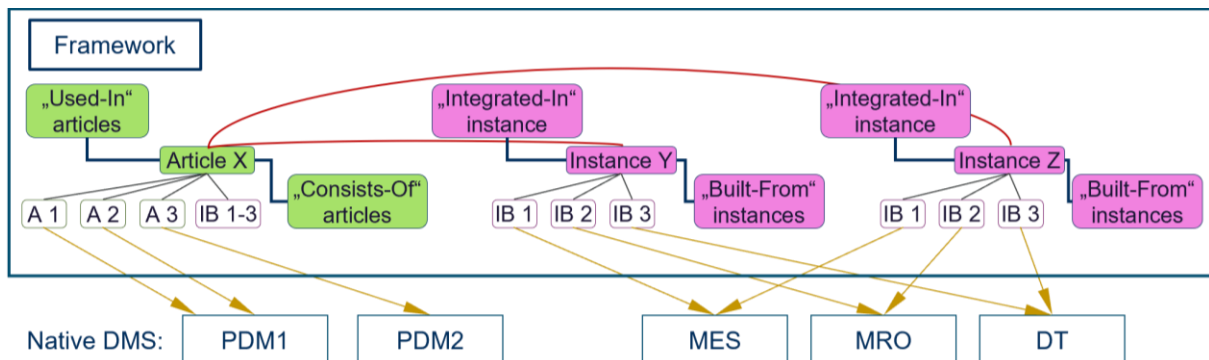

Figure 2: An examplary article (X) with linked instances (Y and Z) and associated attributes, and Instantiation Blocks and the references to nDMS within the framework

Article X has multiple associated attributes and Instantiation Blocks (A1 – A3, IB1 – 3 in Figure 2). The attributes already have references to nDMS (yellow lines in Figure 2), while the Instantiation Blocks serve as placeholders for required information of the instances in following lifecycle phases as introduced in [10]. The instances inherit the Instantiation Blocks from the article (see IB1 – 3 of the instances Y and Z, Figure 2). Then, the Instantiation Blocks are given references to the specific source data in the nDMS (e.g., MES- or MRO-systems or Digital Twins). Examples of attributes are weight, cost or standardized identifiers. Examples of Instantiation Blocks are serial numbers, batch numbers, quality protocols or test certificates.

For each attribute of an article and Instantiation Block of an instance, there is a unique reference to the respective source data within the nDMS (see next chapter).

With this concept, the framework consolidates information distributed across multiple nDMS and thus serves as a consistent, cross-lifecycle source of information on articles and instances for all project participants.

5.2.2. References to the native Data Management Systems

The framework uses specific interfaces to nDMS to retrieve detailed and relevant data on the articles or instances concerned. These interfaces allow access to the required source data and make it available for further use. The principle of data referencing allows the various management systems (as listed in Table 1) to be connected and integrated with little effort.

To meet the requirements of a constantly growing IT infrastructure, the framework should use an open architecture. This openness makes it possible to connect further systems at any time without major effort and thus ensures scalability and flexibility of the entire system.

Data consolidated in the framework is brought into a uniform, industry-standard format. This consolidation and normalization of data ensures a standardized retrieval of data from different systems and parties. The format chosen for this data integration is AutomationML, a widely used specification in automation technology [11]. AutomationML has become an industry standard due to its compatibility and flexibility and can be represented by the Asset Administration Shell [12]. This ensures that data integration across different systems is smooth and without data loss, which greatly increases the efficiency and effectiveness of the overall data management system.

5.2.3. The data structures of the framework

Figure 3 and Figure 4 on page 10 each show a section of the four data structures of the framework. The element types and their relations within the individual data structures are depicted, as are the transformation rules and links between the data structures (each represented by an ID).

All four structures have in common that they serve as connecting points for the discipline-specific and company-specific structures in the nDMS. The data distributed among the nDMS is linked to articles and instances via these connection points. The framework thus forms a central point for information retrieval. The structures and transformations are explained below.
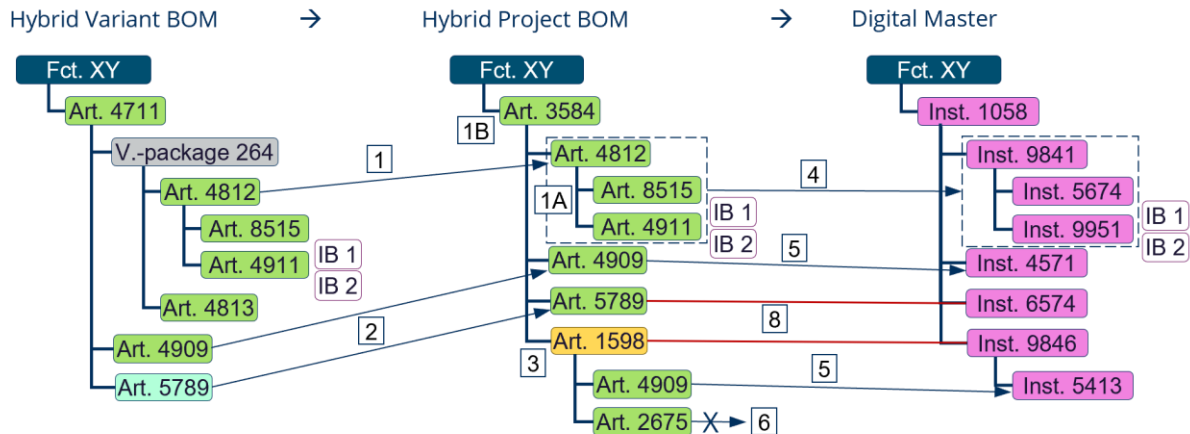

Figure 3: Structures and transformations of the Hybrid Variant BOM, the Hybrid Project BOM and the Digital Master

The **Hybrid Variant BOM** is used to summarize the structures and different variants of the components and assemblies of a specific type of process plant. It is the first structure in the plant lifecycle and the starting point for the requirements-based derivation of multiple Hybrid Project BOMs.

The framework does not specify the concrete structuring approach for the Hybrid Variant BOM. However, a function-based structure in combination with a modularization strategy such as VDI 2776 [9] in combination with VDI/VDE/NAMUR 2658 [13] is recommended.

The Hybrid Variant BOM consists of different interlinked element types. *Functions* (dark blue boxes in Figure 3) represent a hierarchical function structure of the plant in which the articles of the (sub)systems and components can be arranged. *Articles* (green boxes in Figure 3) represent the components, assemblies or subplants of the process plant. They each have a unique ID within the framework that is independent of their IDs within the nDMS. The attributes associated to the article apply to all usages of that article within the structures of the framework. The articles as parent elements of variant packages are to be understood as placeholders containing general information about the concrete assemblies being derived from the Hybrid Variant BOM (see Figure 3 - Art. 4711). *Variant packages* (grey boxes in Figure 3) contain articles on the next lower hierarchy level, which can be selected when deriving Hybrid Project BOMs from the Hybrid Variant BOM. *Options* (turquoise box in Figure 3) are articles that are not mandatory to be selected.

The Hybrid Variant BOM must be initially created and actively maintained. However, it can be used to quickly create any number of Hybrid Project BOMs. To do this, articles have to be chosen from the variant packages based on customer requirements. If there are no articles matching the given requirements an article has to be created as placeholder for Costumer-Order-Engineering (COE) (yellow box in Figure 3).

The transition between Hybrid Variant BOM and Hybrid Project BOM follows specific transformation rules which are described in Table 3. The IDs in Table 3 serve as orientation in Figure 3 and 4 for the visualization of the transformation rules.

Table 3: Transformation rules between the four data structures of the framework

ID	Description
	Hybrid Variant BOM → Hybrid Project BOM (see Figure 3)
1	Articles, which are chosen from a variant package, are linked to a parent article: Art. 4812 is linked to Art. 3584.
1A	Underlying article structures are transferred (Art. 8515 & 4911).
1B	The parent article of the Hybrid Project BOM gets a new ID (Art. 3584 vs. Art. 4711).
2	Mandatory articles and selected options are transferred to the Hybrid Project BOM.
3	Definition of placeholders in the BOM for COE.
	Hybrid Project BOM → Digital Master (see Figure 3)
4	The structure of the Digital Master is based on the structure of the Hybrid Project BOM.
5	For each usage of an article, a distinct instance is created: Art. 4909 is used twice in the Hybrid Project BOM. Two instances (Inst. 4571 und 5413) are created based on Art. 4909.
6	Precondition for the instantiation: The instantiated article or an article in its underlying structure has to have an associated Instantiation Block. Art. 2675 has no Instantiation Block. No instance is created.
7	Instances inherit Instantiation Blocks from articles.
8	Instances and their respective articles are linked: See red line between Art 5789 and Inst. 6575, as well as Art. 1598 and Inst. 9846.
	Digital Master → Digital Representation (see Figure 4)
9	Information of an Instantiation Block is directly referenced with the specific instance of the Digital Master.
10	An existing structure of instances with linked information is inserted into the Digital Master.

The **Hybrid Project BOM** (see Figure 1 and 3) represents the project-specific structure of the costumer-specific plants. It is created through requirements-based variant selection from the Hybrid Variant BOM. By deriving from the Hybrid Variant BOM, the greatest possible reuse of articles is achieved. The articles and structure of the Hybrid Project BOM serve as the basis for engineering to develop discipline-specific development documents. For this purpose, the Hybrid Project BOM can be filtered by discipline and made available to the development teams.

The Hybrid Project BOM is not complete directly after its derivation from the Hybrid Variant BOM. Components for which no suitable variant was found in the Hybrid Variant BOM must be flagged as COE (Costumer Order Engineering) articles (yellow box in Figure 3). The COE components are now to be specified via OEM-internal engineering or external engineering via suppliers. The substructures of the COE components that may arise in engineering are attached to the COE elements in the Hybrid Project BOM after they have been specified and released (see COE article (Art. 1598) in Figure 3).

During development, the Hybrid Project BOM serves as an information source that can be used to retrieve information on the components of the process plant across disciplines and companies. This is possible via the article attributes stored in the framework and their concrete value retrieval through the references (see chapter 5.2.2). At the end of the development phase, a 100% specified Hybrid Project BOM without open COE components is available. The complete Hybrid Project BOM now serves as the basis for the Digital Master and the Digital Representation.

The **Digital Master** is the preliminary stage for the Digital Representation. The Digital Master defines the structure of the Digital Representation with *Instance* elements (pink boxes in Figure 3 and 4) and the information to be provided for the instances via associated instantion blocks. The Digital Master is derived from the Hybrid Project BOM. A characteristic of the derivation is that a separate instance is created for each article usage. The transformation rules from Hybrid Project BOM to Digital Master apply (see Figure 3 and Table 3, ID 4 - 8).

The instances of the Digital Master are placeholders for which no concrete information has yet been linked to the Instantiation Blocks. By capturing and linking the information of the Instantiation Blocks, the transition from Digital Master to Digital Representation takes place (see Figure 4). This process is called the instantiation process. The overall concept of defining an instantiation process was proposed by [3] and is further elaborated in [10].
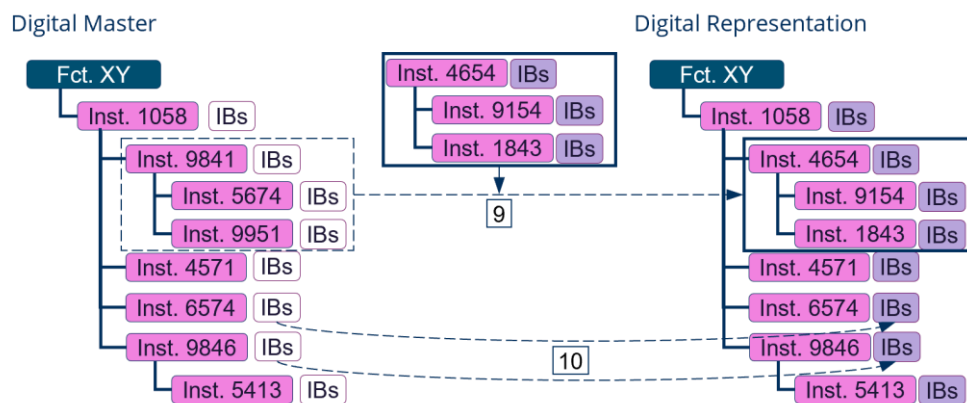
Figure 4: Transformation rules from the Digital Master to the Digital Representation

Two basic use cases can be distinguished as to how to collect instance-specific information and link this information to Instantiation Blocks during the production, construction, and commissioning phase. As a first case, information can be directly linked to a specific instance within the Digital Master (see transformation rule 9 in Figure 4 and Table 3). Secondly, already existing instance structures can be inserted into the Digital Master. This is shown in Figure 4 using the example of instance 4654 (e.g., a pump in stock). The instance 4654 is already assigned with concrete instance-related information (such as a serial number or a test certificate). If this instance is built into instance 1058 (e.g., a cooling system), instance 4654 and its underlying structure replaces the placeholder instance 9841 in the Digital Master (see transformation rule 10 in Figure 4 and Table 3). The detailed description and categorization of these use cases within the instantiation process is subject to further research by the authors.

The completion of the instantiation process, mainly during the production, installation and commissioning of a plant, results in the final version of the **Digital Representation**, which represents all the instances and the information that represents them. During the operational phase of the plant, the Digital Representation should respond to changes of the instances and their structures and ensure consistency between the Hybrid Project BOM as "As-designed" structure and the instance structure of the real plant. Methods are needed to deal with changes in instance-specific information (e.g., when components are replaced during maintenance) and changes to the instance structure during the operational phase of the plants (e.g., when structural changes are made to the plant in the case of process optimization). These methods will be further investigated by the authors.

By linking the instances of the Digital Representation and the articles of the Hybrid Project BOM, the framework is a source of information for:

- instance-specific information (serial numbers, interfaces, test results, service info, ...),
- article-specific information (dimensions, weight, flows, manufacturer, ...).

5.3. Advantages and Challenges

As briefly introduced in sections 1 and 3, a central access point through an instance-based BOM offeres several advantages by linking article-specific design data and instance-specific information. With multiple projects based on the same Hybrid Variant BOM, engineers are able to fully understand the realitionships between project specific requirements and utilized articles within different projects, leading to less COE in the future. Linking all article- and instance-specific data provides intuitive root cause data analysis forward and backward over the plant lifecycle. Furthermore, maintenance, repair and operations are enabled with an overview of the as-maintained structure through the Digital Representation of the plant and can plan and act more efficiently.

However, to exploit the full potential of the proposed framework, some organizational requirements have to be met by the OEM and the suppliers:

- References must be implemented in terms of information technology. This requires changes in the IT structures of the companies involved,
- OEMs and suppliers must agree on processes to organize information and data flows.

In the current version of the framework, direct access via the references is only possible to database stored information in the nDMS. Information stored in files (e.g., CAD or PDF files) is only accessible indirect via the reference to the location where the file is stored. A direct retrieval of file-specific information requires the programming of appropriate extraction tools, which can become very complex.

Data availability and data access between different actors along the lifecycle of process plants is another challenge. Currently, several initiatives and projects are addressing this challenge, like the CATENA-X or DIAMOND projects [14, 15]. Implementations of the framework can rely on the knowledge and methods gained with these projects.

6. Summary and Outlook

In order to optimize the exchange of information, and processes across disciplines and companies along the lifecycle of process plants, the challenges and requirements from industry are first determined in an analysis phase. For this purpose, both a systematic literature review and a qualitative investigation of two industrial companies are conducted. The result is a comprehensive list of requirements that serves as the basis for developing a conceptual framework. The framework focuses on the management of metadata and consists of four structures, each with clearly defined properties. Transformation rules for the transition between the structures are described, and references for querying source data from nDMS are presented. The conclusion is a discussion of the advantages and challenges of the framework and its implementation.

The next steps are the implementation of the presented elements of the framework in form of a prototype. Hence, following publications will focus on the detailed investigation of the challenges that arise with the implementation of the framework. This includes mainly a well-defined data model, which represents and stores the different structures, their elements and the transformations of the framework. Furthermore, the instantiation process is to be examined

in detail and, finally, a comprehensive validation is to be carried out. This ensures the general validity and feasibility of the presented framework as a unified source of information for all stakeholders of process plants.

Acknowledgement

References

[1] RAHM, Julian; HENSELMANN, Daniel; URBAS, Leon: Synchronization network of data models in the process industry. In: *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)* : IEEE, 2021, S. 1–8

[2] BOROWSKI, Piotr: *Digitization, Digital Twins, Blockchain, and Industry 4.0 as Elements of Management Process in Enterprises in the Energy Sector*. In: *Energies* 14 (2021), Nr. 7, S. 1–20

[3] SASKE, Bernhard; SCHWOCH, Sebastian; PAETZOLD, Kristin; LAYER, Max; NEUBERT, Sebastian; LEIDICH, Jonathan; ROBL, Peter: *Digitale Abbilder als Basis Digitaler Zwillinge im Anlagenbau: Besonderheiten, Herausforderungen und Lösungsansätze*. In: *Industrie 4.0 Management* 2022 (2022), Nr. 5, S. 21–24

[4] EIGNER, Martin; STELZER, Ralph: *Product Lifecycle Management*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2009

[5] EIGNER, Martin: *System Lifecycle Management*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2021

[6] LAYER, Max; LEIDICH, Jonathan; SCHWOCH, Sebastian; SASKE, Bernhard; NEUBERT, Sebastian; ROBL, Peter; PAETZOLD-BYHAIN, Kristin: *Data management of process plants as complex systems: systematic literature review and identification of challenges and opportunities*. In: *Reviews in Chemical Engineering* 0 (2023), Nr. 0

[7] PFENNING, Michael: *Durchgängiges Engineering durch die Integration von PLM und MBSE*. Technische Universität Kaiserslautern. Dissertation

[8] KRAUSE, Dieter; GEBHARDT, Nicolas: *Methodische Entwicklung modularer Produktfamilien*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2018

[9] VDI-Richtlinie 2776. Nov 2020. *Process engineering plants - Modular plants - Fundamentals and planning modular plants*

[10] LAYER, Max; NEUBERT, Sebastian; TIEMANN, Lea; STELZER, Ralph: *Identification and retrieval of relevant information for instantiating digital twins during the construction of process plants*. In: *Proceedings of the Design Society* 3 (2023), S. 2175–2184

[11] LÜDER, Arndt; SCHMIDT, Nicole; DRATH, Rainer: Standardized Information Exchange Within Production System Engineering. In: BIFFL, Stefan; LÜDER, Arndt; GERHARD, Detlef (Hrsg.): *Multi-Disciplinary Engineering for Cyber-Physical Production Systems*. Cham : Springer International Publishing, 2017, S. 235–257

[12] DEUTER, Andreas; IMORT, Sebastian: *PLM/ALM Integration With The Asset Administration Shell*. In: *Procedia Manufacturing* 52 (2020), S. 234–240

[13] VDI/VDE/NAMUR-RICHTLINIE 2658-1. Oct 2019. *Automatisierungstechnisches Engineering modularer Anlagen in der Prozessindustrie*

[14] MÜGGE, Janine; GROSSE ERDMANN, Julian; RIEDELSHEIMER, Theresa; MANOURY, Marvin Michael; SMOLKA, Sophie-Odette; WICHMANN, Sabine; LINDOW, Kai: *Empowering End-of-Life Vehicle Decision Making with Cross-Company Data Exchange and Data Sovereignty via Catena-X*. In: *Sustainability* 15 (2023), Nr. 9, S.1–24

[15] *DIAMOND - Digitale Anlagenmodellierung mit neutralen Datenformaten*. URL https://diamond-project.de/. – Last updated: 2023 – Last checked: 2023-07-28