Journal of Industrial Information Integration 28 (2022) 100369

Contents lists available at ScienceDirect

Journal of Industrial Information Integration

journal homepage: www.sciencedirect.com/journal/journal-of-industrial-information-integration

Full length article

Managing virtual factory artifacts in the extended PLM context

Iman Morshedzadeh [a,*], Amos H.C. Ng [a], Manfred Jeusfeld [b], Jan Oscarsson [a,1]

[a] School of Engineering Science University of Skövde, Högskolevägen, Skövde 54128, Sweden
[b] School of Informatics University of Skövde, Högskolevägen, Skövde 54128, Sweden

ARTICLE INFO

Keywords:
Virtual model
Product lifecycle management
Information model
Knowledge management
Ontology

ABSTRACT

Virtual engineering increases the rate of and diversity of models being created; hence requires maintenance in a product lifecycle management (PLM) system. This also induces the need to understand their creation contexts, known as historical or provenance information, to reuse the models in other engineering projects. PLM systems are specifically designed to manage product- and production-related data. However, they are less capable of handling the knowledge about the contexts of the models without an appropriate extension. Therefore, this research proposes an extension to PLM systems by designing a new information model to contain virtual models, their related data and knowledge generated from them through various engineering activities so that they can be effectively used to manage historical information related to all these virtual factory artifacts. Such an information model is designed to support a new Virtual Engineering ontology for capturing and representing virtual models and engineering activities, tightly integrated with an extended provenance model based on the W7 model. In addition, this paper presents how an application prototype, called Manage-Links, has been implemented with these extended PLM concepts and then used in several virtual manufacturing activities in an automotive company.

1. Introduction

Industrial Information Integration Engineering (IIIE) comprises methods for solving complex problems when developing information technology infrastructure for industrial sectors, especially in information integration, as defined in an extensive review of the field [1]. It is an emerging discipline, but multiple stakeholders have increased their demand for research of IIIE topics because of the ever-increasing impact of Information and Communication Technology (ICT) on the industry [2]. In [1], manufacturing, among other engineering disciplines, has been identified as the second biggest research category in the literature related to IIIE.

IIIE research for manufacturing applications spreads a huge variety of topics, but many of them regard the information modeling and system development for Product Lifecycle Management (PLM) processes in general and digital factory management in specific. Regarding digital factory management, as a part of the digitalization trend in modern societies, the utilization of virtual models and simulations is gradually increasing in various industrial sectors. A model is a representation of a system or an object of interest, and a virtual model is a model that has been created by computer systems. Simulation is the process of using a model to study the characteristics and behavior of a physical or conceptual system. Frequently, simulation models are operated by changing one or more variables of the model and analyzing changes in the simulation results [3,4]. The use of different modeling and simulation tools for analyzing products and their productions can help industries to develop their products and processes faster (shorter lead-time) and cheaper (more efficient). By using virtual models and simulators, experiments on different designs or changes can be performed without using costly real equipment. The advantage of cutting lead-time is apparent when designing a new product or a process that does not yet exist. Even for an existing process, it is valuable to be able to carry out experiments on its virtual, digital counterpart to evaluate possible changes to improve efficiency, without the need to stop the physical process and the risk of possible damage [5]. Given the constant demands to further reduce product development lead times, there is a great potential for applying virtual models and simulation technologies to effectively support the design and operation of factories. With the

* Corresponding author.
E-mail addresses: iman.morshedzadeh@his.se, iman.morshedzadeh@gmail.com (I. Morshedzadeh), amos.ng@his.se (A.H.C. Ng), manfred.jeusfeld@his.se (M. Jeusfeld), jan.oscarsson@his.se (J. Oscarsson).
[1] Present address: Innovatum Science Park, 461 53, Trollhättan, Sweden.

https://doi.org/10.1016/j.jii.2022.100369
Received 7 April 2021; Received in revised form 16 October 2021; Accepted 29 May 2022
Available online 2 June 2022

initiation of the fourth industrial revolution (Industry 4.0) and the convergence of the real and virtual worlds as one of the needs for smart manufacturing in Industry 4.0, the role of virtual models has become more prominent.

Despite the apparent potential of virtual models and simulations in manufacturing, there are some major challenges to their use. First, despite the popularity of modeling and simulation software packages with intuitive graphical user interfaces and easy-to-use features, virtual modeling and development of simulation models are still considered time-consuming and costly. Apart from the acquisition of domain knowledge and the required modeling skills, most of the time in a simulation project is spent on data gathering [6]. Since the validity of any model is entirely dependent on the accuracy of the input data, the amount of time required to ensure the accuracy of input data can be overwhelming. Because of the cost and effort of developing valid models, some industrial companies have proposed a concept called virtual confidence that can be used to classify different levels of trust and to promote the use and reuse of virtual models [7]. This concept is based on the idea that the more virtual models are used effectively in an enterprise, the higher the level of confidence the enterprise has in these virtual models. In this manner, higher reuse of them can save engineering time. Nevertheless, in order to reuse a virtual model, it is not enough to have the model itself stored in a central location accessible to others in the organization. Other potential users must also have sufficient information about the development context and usage history of that model, especially if they were not involved in its original development.

While it seems obvious to associate virtual models with optimization and decision-making, the second challenge is that simulation models are actually evaluative by nature. Rather than generating optimal solutions by themselves, they can only provide performance evaluations of alternative design/operational settings. It is when the virtual models are connected to efficient optimization algorithms that optimal solutions can be sought automatically. Simulation-Based Optimization (SBO) is the technology that allows an optimization algorithm (heuristic or metaheuristic-based) to connect to a virtual model to search the vast number of possible solutions and to identify the ones that optimize a given objective. In the presence of more than one objective, multi-objective optimization (MOO) algorithms are used to generate optimal trade-off solutions for the decision-maker to choose which solution to implement [8]. Since most real-world problems, such as manufacturing systems development and improvement, are inherently multi-criteria by nature and hence inevitably involve multiple optimization objectives, these MOO algorithms have been increasingly used in practice during the last two decades [9]. By applying data mining methods on the optimization outcomes, SBO and MOO are able to support the extraction of knowledge impossible to be found in a historical database [10]. While this new way of extracting knowledge from virtual models and simulation offers the value-adding functions to virtual engineering activities, this large set of virtual artifacts, i.e., simulation outputs, optimization results in the form of trade-off solutions, and knowledge represented by rules and patterns, will drastically further increase the challenges in managing the virtual models effectively, for re-use and training purposes, as discussed above.

There is yet a third challenge, namely, to manage the additional type of knowledge that can be gained from the process of generating and utilizing virtual models through different engineering activities that involve virtual factory models. This knowledge can be managed and presented through the classification of different types of virtual models, their related engineering activity types and other types of provenance information and later, relate them to each other to specify workflows of engineering activities for generating and utilizing different virtual model types. Groth and Moreau defined "provenance" information as "information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness." The concept of provenance information has been used in various application domains [11,12]. In this research, some provenance information about the engineering activities, entities, and people that led to the generation of a model has to be stored. Based on the Zachman framework [13] and the W7 model, presented by Ram and Liu [14], provenance information can be identified by answering seven questions: Who, Where, Why, What, When, Which, and hoW [15].

Overall speaking, this paper argues that there is a critical need for an information system that can address these challenges and be capable of managing a wide variety of information in order to effectively manage virtual factory artifacts. In principle, existing PLM systems commonly used in the industry today are capable of managing and controlling how product and production data are transferred between the PLM system and other engineering software programs. It is also possible to integrate various engineering tools with the system. However, conventional PLM systems are inadequate for managing the provenance data as, for instance, defined by the W7 model. Hence, an "extension" is needed - the extended system needs to be able to manage virtual models and associated types of artifacts like simulation outputs, optimization results and extracted knowledge as listed above, together with their provenance data.

Information models can represent these objects as concepts and structure them by specifying their relationships, constraints, and rules. Using such information models in an extended PLM will render the structured data, information and knowledge more manageable. As will be reviewed extensively in Section 2, although some information models exist in the literature for managing virtual models and their related data, an information model for storing, linking, and managing virtual factory artifacts including extracted knowledge to overcome the above-mentioned challenges is missing. Particularly important is the linking aspect for the structured data, information and knowledge to be searchable, retrievable and reusable in an effective manner. When the variety of virtual models is wide, and the volume of data, information and knowledge is vast, we believe that such a new information model has to be developed with a new ontology that adequately captures and formalizes the linkages of virtual models, engineering activities and domain knowledge. An ontology captures the concepts, their attributes, and their relationships to other concepts; it is used for specifying standard conceptual vocabularies, in which they are provided services for answering queries and publishing reusable knowledge bases [16]. For managing the above-said two types of knowledge generated from SBO and the engineering activities using the virtual models, an ontology-based knowledge management method has been developed.

As a short summary, this paper aims at the following contributions to knowledge:

- The need and requirements of a new information model to contain virtual models and knowledge from the manufacturing industry have been identified and analyzed.
- We propose a new domain ontology, called Virtual Engineering (VE-) ontology, for capturing and representing the concepts and their relationships for virtual models and knowledge generated from SBO and/or engineering activities to facilitate the inferencing and inquiry operations against any knowledge bases that adopt it.
- An extended provenance model, based on the above-said W7 model, but tightly integrated with the VE-ontology that can be incorporated into an existing PLM system, is introduced. This is for recording the historical information about the creation of the artifacts generated in engineering activities. This is purposed for searching and re-using virtual models, their data and information as well as knowledge generated with them.

The above three innovations are integrated into a prototype software system called Manage-Link, wherein the ontology and the provenance model are implemented in a database system and connected to Siemens' PLM system, Teamcenter[2] (denoted as Tc hereafter). This software development aims to realize how an existing PLM system can be extended to incorporate the innovations created in the current paper. Its applicability for industrial-scale knowledge management is demonstrated with a series of application examples in a multi-level industrial case found at an automotive manufacturer. However, the information model, the ontology and the provenance model can work with any PLM system; they have been realized through such a prototype developed to operate as a stand-alone application linked to Tc.

Fig. 1 gives an overview of how the key concepts proposed in this paper are interlinked to each other, all the way to the computer-aided engineering (CAx) models. The extended information model improves the capabilities of a PLM system to manage virtual models. The ontology allows to tag virtual engineering models (including particular knowledge artifacts) by the concepts of the ontology to support their re-use. The *concepts* of the provenance data model represent the top-level concepts in the ontology, gluing these two parts together. While modern PLM systems are continuously improved for better integration with CAx systems and improved workflow support, they still fall short in the knowledge dimension, specifically in supporting the reuse of existing virtual models for new engineering projects. We argue that our combination of an extended information model with a provenance data model and a domain ontology for virtual engineering is filling this gap.

The rest of the paper is organized as follows: Section 2 covers an extensive literature review to identify the research gap to justify the need for a VE-ontology based knowledge management method. The requirements and design of the information model to support the reuse of virtual engineering artifacts are presented in Section 3, followed by the provenance model and VE-ontology development, including the introduction Manage-Link in Section 4. The multi-level industrial application case is covered in Section 5, followed by the conclusions and future work in Section 6.

2. Literature review

This section provides an extensive literature review to cover two wide topics to identify the research gaps and justify the need for the current study, namely, (1) information modeling in manufacturing and (2) knowledge management and ontologies in PLM.

2.1. Information modeling in manufacturing

A product is a physical thing that is produced and offered to the market. Each product has a lifecycle. Various researchers have divided this lifecycle into different phases. Crnkovic, for example, divides a product lifecycle into six phases: business idea, requirements management, development, production, operation, maintenance, and disposal [17]. There are also many different definitions of product lifecycle management [18] affected by points of view and utilization. Saaksvuori and Immonen [19] describe PLM as a holistic business concept. It is sometimes defined as approaches and activities such as an information-driven approach, a strategic business approach, or the business activity of managing [20–24]. From the point of view of information system research, PLM has been considered an information management system, which is a blanket term for a group of software applications [25,26]. Building on these definitions, PLM has been defined as, "a business approach for the management and use of product-, process-, and resource-related data, information, and knowledge as corporate intellectual capital over the extended enterprise", for the purpose of this paper.

PLM systems are designed for managing product, process and resource (PPR) data. Hence, Bill of Material (BoM), Bill of Process (BoP) and Bill of Resources (BoR) are considered as the three main structures of a PLM system [27,28]. These three structures can be considered as the core of any PLM system. BoM is about the structure of the product and it consists of its different parts and assemblies. BoP shows different processes and their relationships for manufacturing a product. BoR represents the structure of a factory. It contains the resources which are needed for producing a product. Based on the definitions of PLM and the general capabilities of current PLM systems for transferring data between different CAx models, they are suitable platforms for the management of virtual models and their related data, information and knowledge, if properly extended to incorporate the management of provenance data and knowledge as proposed in the current paper.

As mentioned in the introduction, many information models exist in the literature in the domain of manufacturing, which are used as a basis for PLM systems. Rachuri et al. [29] provided some examples of standards for these purposes: (1) standards for implementation languages; (2) information modeling standard; (3) content standards – domains of discourse; and (4) architectural frameworks standards. They identified that for product information modeling in the PLM systems, ISO 10303 that is known as STandard for Exchange of Product model data (STEP) with its extension called PLCS (Product Life Cycle Support) is well used [29,30]. ISO standard 10303 consists of different application protocols that each protocol represents information models of one application area. For example, AP214 (ISO 10303-214) defines the "Core Data for Automotive Mechanical Design Processes" and AP212 (ISO 10303-212) is labeled "Electrotechnical design and installation". The ASD Strategic Standardisation Group (SSG) provides a map showing how different STEP application protocols cover different phases of product lifecycle management [31].

Researchers have tried to connect different application protocols to models of product lifecycle information [32,33]. For example, Euler-Chelpin presents an information model for managing product, process, and resource information by combining AP214 (Core data for automotive mechanical design processes) and AP239 (Application protocol: Product life cycle support) [34]. PLM systems use different APs of ISO 10303 to structure information in different applications of their system according to areas and domains. For example, AP 242 is used to manage product models and their revisions, and AP 240 is used for process plans for machined products. Because of the variety of application areas and different points of view and levels of detail, they cannot easily be connected and used to manage information throughout the whole product lifecycle [34].

The core product model (CPM) and its extension CPM2 are developed at National Institute of Standards and Technology (NIST) to support the management of product models in three levels of conceptual, intermediate, and implementation [33]. As Fenves et al. [33] emphasized in their article, CPM2 is a generic, abstract model and it mainly covers product and product design-related information. ANSI/ISA–95 is another standard that represents information models in manufacturing [35]. This standard presents three main types of models for the definition of functions and integration associated with control and enterprise systems:

- Hierarchy models that define different levels of functions and domains.
- A data flow model that defines data flow in an organization.
- An object model that defines the information that may cross the enterprise and control system.

In this standard, most of the information types described fall into three areas [35]:

1 Production capability information area, including capability scheduling and maintenance information.

[2] All trademarks used in this paper are property of their respective owners.

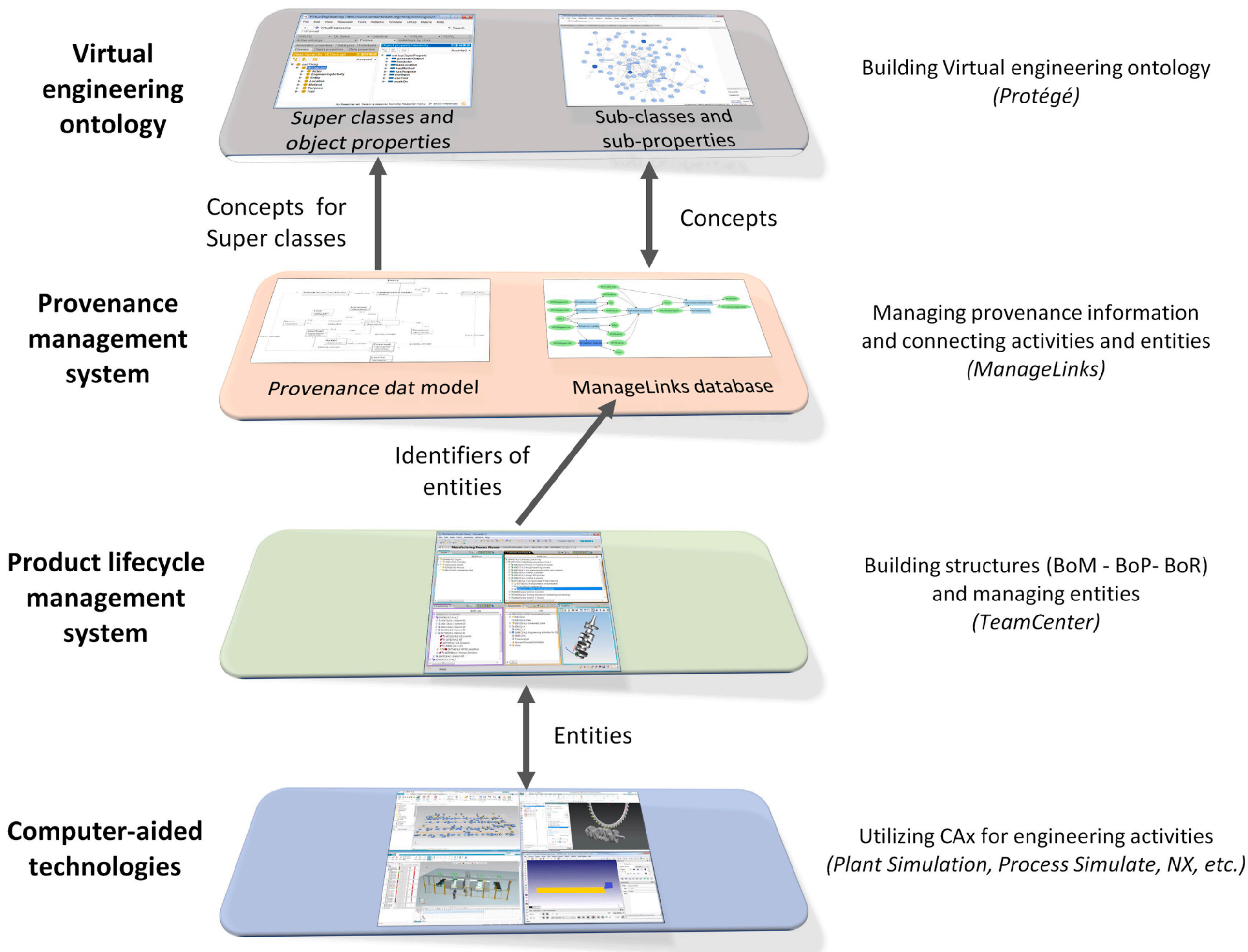


Figure 1. Different engineering and management systems and their interactions

2 Product definition information area, including product production rules, bill of material and bill of resources.
3 Production information area: including production history, inventory and scheduling information,

Some of these information areas are generally managed by PLM systems and others can be managed by other information systems such as enterprise resource planning (ERP) systems. In summary, although the ANSI/ISA–95 standard covers manufacturing-related information and data, the focus of this standard is not on the PLM core data. Like the other standards reviewed, it is too generic to be used to fill the research gap identified in this paper.

Some existing information models are more specifically related to PLM core data. Some researchers present structures for products, processes, plants, and resources (P3R) [36] or products, processes, and resources (PPR) [34,37–39]. In some information models, humans are not considered a resource, and the information model is based on PPR+H [40–42]. Most of these information models were developed for very specific application areas. None of them consider the management of virtual models and their related data and information.

There are some information models related to virtual models and specifically simulations. Core manufacturing simulation data (CMSD) is the most well-known standard for manufacturing simulation data. It provides information models that represent essential manufacturing entities needed for simulation as well as their relationships for the efficient exchange of manufacturing data in a simulation environment [43]. Two different languages are used in this standard for modeling information: (1) Unified Modeling Language (UML) and (2) XML Schema definition language. CMSD divides manufacturing information into six main packages: layout; (2) part information; (3) support; (4) resource information; (5) production operations; and (6) production planning. This standard is specifically used to structure and manage discrete event simulation (DES) data, especially when the automation of input data is desired [44,45]. The CMSD has more attributes of classes that cover manufacturing-related data compared to other standards such as ISA-95, but its structure is not suitable for PLM systems, as noted by Lee et al. [46].

Lee et al. [36] present an object-oriented model of P3R information that uses the output data from PLM systems and prepares the data to generate DES models. Their paper shows the role of PLM and P3R in providing simulation data, but it does not deal with the structuring of information in the PLM system. Damarapurapu and Gargatte [47] offer two information models for structuring output data from the generic data management (GDM) tool. GDM is a system for automating input data for DES. In their research, two structured systems based on CDMS and ISA-95 were presented for managing DES input data. They emphasize the necessity of future research for structuring output data from the GDM tool in the PLM system. Later this output data can be transferred from the PLM system to DES applications for simulation and analysis [47,48].

Since a PLM system is used as a platform for product and production-related data transformation, it can be readily used as a corporate business system to be integrated with simulation tools for transferring data automatically to simulation tools. For the implementation of the fourth method, the necessity of an information model in PLM systems for structuring simulation models and data becomes evident.

Table 1 relates the reviewed papers on information models for PLM and simulation data and specifies the information areas covered by information models. Related areas can be one or several areas denoted with the notations attached to the bottom of the table. This table shows that some information models cover several areas, while others cover only one area and are explicitly focused on that area, such as product. In this table, checkmarks are used to indicate whether articles present information models related to any engineering activity as a cause or outcome of virtual models or present information models related to PLM systems. Modeling, simulation, optimization, and knowledge extraction were selected as engineering activities because they are within the scope of this research and are related to generating and utilizing virtual models. Information models about PLM were evaluated either as the core of PLM systems specifically related to product, process, and resources (PPR) or as PLM covers more than PPR such as services, access management and document management. The level of detail for the structures is specified next. In some articles, the structures presented are very abstract. However, in others, the information models are presented in more detail and formal languages or semi-formal structured languages are used, such as UML class diagrams. Some articles provide structures at a lower level of abstraction within data models. In the last column, papers are marked as either specific or generic to indicate if the information models are mainly for a specific case study or particular types of data.

As a general conclusion, there exists no information model for managing virtual models in the PLM context, covering the produced and extracted data, information, and knowledge by experimentation on these models. Even those information models related to modeling or simulation merely manage input data for merely simulation-related activities; they do not manage the virtual models themselves and their related knowledge. Therefore, in this research, an information model described in Section 3.2 is designed to manage virtual models and their related data, information, and knowledge, as well as the provenance data in the extended PLM that is tightly connected to the core of PLM or PPR structures.

2.2. Knowledge management and ontologies

Knowledge can be used for creating a new and better solution for decision-making [59]. Data, information, and knowledge are the three interrelated concepts with different definitions. In some articles and books, wisdom also has been defined as the top level in the hierarchy [60,61]. Defining knowledge is much more complicated than data and information. In many books and articles, knowledge has been divided into only two different types: tacit and explicit, but Nickols added implicit knowledge [62]. Explicit knowledge is objective knowledge and can be articulated and codified [59,63]. Tacit knowledge is subjective; it is in human minds rather than codified [63–65]. The kind of knowledge that exists but has not been articulated yet is implicit knowledge [62].

Managing manufacturing knowledge is a subject of interest for many researchers. This knowledge can be about different aspects of manufacturing, such as products, processes, and resources. For example, Baxter et al. [66] have defined a knowledge management framework for design-related knowledge. In this study, we consider two types of knowledge to be represented and managed. *One type* is the knowledge extracted from designing and using virtual models in different engineering activities such as simulation and optimization. Bandaru et al. [10] identified different types of knowledge representation from knowledge discovery in multi-objective optimization such as unseriated and seriated heatmaps, directed graphs, visual clusters from implicit knowledge and regression model, decision rules, association rules from explicit knowledge.

The *second type* is ontology-based knowledge management. The information systems aspects of knowledge identification, acquisition, sharing, and distribution are important. Different publications focus on different parts of these functions. In the last two decades, retrieving knowledge for the purpose of sharing and distribution has led to the use of ontologies in the knowledge modeling community. Compared to earlier knowledge representation notations like semantic networks and semantic frames, ontologies provide the well-defined, precise, and formal semantic specifications required for the shared conceptualization of a particular subject matter of discourse [67]. By formally defining the concepts, relationships, and axioms, ontologies promote mutual understanding between different stakeholders. By formalizing the domain knowledge, different types of applications can be supported using the standardized terms defined in the ontologies [68] as well as enabling cost-effective sharing of design knowledge between knowledge-based engineering software systems [69].

The lack of a systematic and constructive methodology for developing manufacturing ontologies has been seen as an impediment to wide knowledge reuse in distributed manufacturing environments [70]. They propose a two-level knowledge modeling approach to systematically developing manufacturing ontologies using software engineering and semantic web paradigms. In recent years, there has been an increasing number of articles related to ontological frameworks or ontology development methodologies in the manufacturing domain. Yao et al. [71] developed a multi-perspective, multi-layered knowledge association modeling approach that combines the ideas of ontologies and topic maps to represent and organize manufacturing knowledge, mainly for managing heterogeneous documents. Another manufacturing reference ontology was proposed by Usman [72]. It was later extended to support interoperability across multiple assembly systems [73] and global product service production [74]. Sanya and Shehab [75] developed an ontology-based knowledge-based framework for implementing platform-independent knowledge-enabled product design systems to strengthen the modularity and reuse of engineering design ontologies in the aerospace industry. Wu et al. [76] proposed a systematic, multi-stage approach to developing knowledge integration and sharing for product development based on ontology.

A state-of-the-art review of ontologies in the context of PLM can be found in El Kadiri and Kiritsis [77]. Petnga and Austin [78] introduced an ontological framework for knowledge modeling and decision support in cyber-physical systems. As an outcome of the EU-FLEXINET project, Palmer et al. [74] proposed a comprehensive manufacturing reference ontology that can (1) support the clarification of understanding across domains, (2) support the ability to flexibly share information across interacting software systems, and (3) provide the ability to readily configure company knowledge bases to support interoperable manufacturing systems. Apart from FLEXINET, three more important projects are mentioned in the extensive review of ontology-based solutions for interoperability of PLM systems by Fraga, Vegetti and Leone [79]: Interoperable Manufacturing Knowledge Systems (IMKS) [80], MSEE [81] and Manufacturing Information ontological model introduced by Hastilow [82]. An ontological model for the cloud-manufacturing domain to support information exchange between cloud-manufacturing resources for PLM can be found in Talhi et al. [83].]. Information specifically related to context-oriented KM for production networks (globalized supply chains) can be found in [84].

Negri et al. [85] provide a systematic and extensive review of available semantic languages and features to justify the selection of the Web Ontology Language (OWL) as the semantic language for developing manufacturing domain ontologies. In fact, Lemaignan et al. [86] had previously proposed a Manufacturing Semantics Ontology (MASON) to formally capture the semantics of concepts related to manufacturing industries using OWL. An initial effort to develop the model into an ontology using Web Ontology Language-Description Logic (OWL-DL) was proposed by (Matsokis and Kiritsis [87]. However, to the best of our knowledge, no ontology-based publications except Groth and Moreau [11], have mentioned the use of engineering activities and virtual models in production system development. They reported that the PROV

Table 1
Related articles and standards about different information models for PLM, manufacturing, and simulation data.

ID	Title	Reference	Area	Related to engineering activities				Related to PLM		Level of detail			Specific/ Generic
				Modeling	Simulation	Optimization	knowledge extraction	PLM	PPR	Info Concept	Info Model	Data Model	
1	A core product model for representing design information (Fenves, 2002)	[49]	Pr					√				√	G
2	CPM2: A Core Model for Product Data (Fenves et al., 2008)	[50]	Pr					√				√	G
3	A product information modeling framework for product lifecycle management (Sudarsan et al., 2005)	[33]	Pr					√			√		G
4	Total object unified model driven architecture of product lifecycle management (Sumei, 2011)	[51]	La, Pr, Po, Re, Pp					√				√	G
5	Energy Simulation Framework Integrated with Green Manufacturing-Enabled PLM Information Model (Zhao et al., 2015)	[41]	Pr, Po, Pl, Or					√			√		S
6	Integration Framework and PPR+H Hub for DiFac (Lee, Kim, et al., 2011)	[40]	Pr, Po, Re, Hu						√		√		S
7	XML-based concurrent and integrated ergonomic analysis in PLM (Kim et al., 2008)	[52]	Pr, Po, Re, Hu						√		√		S
8	Development of a conceptual reference framework to manage manufacturing knowledge related to products, processes and production systems (Colledani et al., 2008)	[38]	Pr, Po, Re, Or, Tr						√			√	S
9	Product-Process-System Information Formalization (Colledani et al., 2009)	[37]	Pr, Po, Re						√			√	S
10	"Product-Process-Machine" System Modeling: Approach and Industrial Case Studies (Smirnov et al., 2013)	[39]	Pr, Po, Re, Or						√	√			S
11	Information modelling for the manufacturing system life cycle(Euler-Chelpin, 2008)	[34]	Pr, Po, Re						√		√		G
12	A manufacturing process information model for design and process planning integration (Feng and Song, 2003)	[53]	Po, Re					√			√		S
13	ISO 10303 (STEP) AP 239 edition 3 Application Protocol For Product Life Cycle Support (PLCS) (The ASD Strategic Standardisation Group (SSG), 2015)	[31]	La, Pr, Po, Re, Pp, Su	√				√			√		G
14	Design and Implementation of a PLM System for Sustainable Manufacturing (Zhao et al., 2012)	[42]	Pr, Po, Re, Or	√				√			√		S
15	ANSI/ISA-95.00.01 Enterprise-Control System Integration - Part 1: Models and Terminology (Instrument Society of America, 2000)	[54]	Pr, Po, Re, Pp, Mt, Pe	√					√		√		G
16	Manufacturing information integration in product lifecycle management (PLM) (Qiao and McLean, 2004)	[55]	La, Pr, Po, Re, Pp	√	√						√		G
17	Automation of Input Data Management for Discrete Event Simulation (Damarapurapu and Gargatte, 2016)	[47]	La, Pr, Po, Re, Pp	√	√						√		S
18	Standard for Core Manufacturing Simulation Data SISO-STD-008-01-2012 Simulation Interoperability Standards Organization, 2012, p. 008-01-2012)	[43]	La, Pr, Po, Re, Pp, Su	√	√						√		G
19	KE tool: An open source software for automated input data in Discrete Event Simulation projects (Barlas and Heavey, 2016b)	[45]	Pr, Po, Re, Pp, Mt, Pe	√	√						√		G
20	Generation of STEP AP214 Models From Discrete Event Systems for Process Planning and Control (Falkman et al., 2008)	[56]	Pr, Po, Re, Pp	√	√							√	G
21	Concurrent material flow analysis by P3R-driven modeling and simulation in PLM (Lee et al., 2012)	[36]	Pr, Po, Re, Pp, Su, Pj	√	√							√	G
22	Core Manufacturing Simulation Data – a manufacturing simulation integration standard: overview and case studies (Lee, Riddick, et al., 2011)	[46]	La, Pr, Po, Re, Pp, Su	√	√							√	S
23	ISO 10303-1:1994 - Industrial automation systems and integration – Product data representation and exchange (International Organization for Standardization, 1994)	[57]	Pr, Po, Re, Ma, Tr	√								√	G
24	Shop Data Model and Interface Specification (McLean et al., 2005)	[58]	La, Pr, Po, Re, Pp	√								√	G

Pr: Product – Po: Process – Re: Resource – Pl: Plant – Hu: Human – Or: Organization – Pp: Production plan – La: Layout – Su: Support – Mt: Material – Pe: Personal – Tr: Transformation – Pj: Project – Ma: Management Info Concept: Information Concept – Info Model: Information Model ontology was developed by the World Wide Web Consortium (W3C) about activities, entities and people involved in producing a piece of data [11]. Although the activity class in this ontology is somewhat compatible with engineering activities, PROV-ONTO only covers three types of provenance data (What, Who and When) out of the previously mentioned W7 questions. It is not related to production system development nor to the manufacturing domain. A more striking observation is that ontologies appear to have had little impact on Big Data applications, despite the clear need for a better understanding of the meaning of the data and the results of data mining [88].

OntoCommons (ontocommons.eu) is an EU project started in 2020 to support interoperability between engineering tools, in particular for facilitating data exchange. The project develops top-level, mid-level and domain ontologies as a collaboration between academia and industry. In our approach, the VE-ontology has a different purpose: the ontology concepts act as meta-data tags for the provenance data created during the virtual engineering activities. Hence, the purpose is to support the discovery of reusable artifacts rather than facilitating data exchange between tools. Still, the proposed approach shall be able to benefit from the ontologies developed in such projects as OntoCommons. The provenance data model presented in section 4 allows linking artifacts with any number of concepts imported from various ontologies. To summarize, a new ontology is needed to cover different management aspects of virtual models, virtual engineering, and provenance, as will be elaborated in the subsequent sections.

3. Analysis of virtual models and development of a solution

As a prerequisite to designing a new information model for managing virtual models, their roles in the manufacturing system lifecycle have to be studied and analyzed. The analysis presented in Section 3.1 helps to identify the different management aspects for specifying data and information that need to be structured in the information model. Through this analysis, the relationship of virtual models to physical objects that are modeled as well as their corresponding engineering activities has been identified. Section 3.2 then presents the designed information model as a solution for structuring and managing virtual models and other artifacts such as entities, projects, studies, engineering activities and provenance data, followed by the overall information flow between these artifacts presented in Section 3.3.

3.1. The role of virtual models and their management aspects

Virtual models are used for modeling different entities such as products, resources, or processes, based on their related data. Since PLM systems manage these entities and their related data, information structuring in the PLM needs to be considered for managing virtual models. As mentioned, BoM, BoP and BoR are three hierarchical structures in PLM systems. The hierarchy in these structures is based on the levels of aggregation, and virtual models are modeling the corresponding entities in different levels of hierarchy. Fig. 2 (a) shows how manufacturing systems are structured in this research, based on available structures in ISO standards [54,89]. Manufacturing systems have been divided into plant/site, area, line, cell/station, machine/equipment, and tool, as shown in the middle of Fig. 2 (a). At each level, several virtual models have been produced and used for different engineering activities.

Each manufacturing system on any of the levels of the manufacturing system has a lifecycle, which can be divided into elements before the start of operation and after the start of operation (SOP). Fig. 2 (b) shows the lifecycle of manufacturing systems and corresponding virtual models, adapted from an illustration in an annex to ISO 15704 for parallel processes in the entity's life history [90]. The time axis of the graph in Fig. 2 (b) shows the lifecycle of a manufacturing system. In this figure, blue rectangles with diamond hatching indicate different virtual models in different levels of study, such as conceptual, detail, design, and as-built, before the start of operation of a manufacturing system. Virtual models in the conceptual and detail studies are used for doing different experiments and defining detailed specifications of the final design of that manufacturing system. The virtual model of the design study is the one that is finally accepted. Since, in most cases, there are some differences between the designed manufacturing system and the system actually built, after building the manufacturing system, there will be an as-built virtual model corresponding to the manufacturing system.

Manufacturing systems are dynamic and constantly changed during their lifecycle. Sometimes those changes are minor and are made without any previous plans. After those changes, a new as-built virtual model has to be generated to show the current status of the manufacturing system. In some cases, changes are planned with studies before implementation. Those changes can arise from conceptual or detailed studies or can be just a new design for changes in the manufacturing system. There is always one virtual model generated from an as-built study that shows the current status of the manufacturing system and is the digital twin of that manufacturing system. After any change, the former virtual model is no longer the digital twin; the new as-built virtual model becomes the digital twin of the changed manufacturing system. In Fig. 2 (b), the digital twin virtual model is shown as a solid green rectangle. If there is an ongoing project to change or improve a manufacturing system, then there is a possibility of using virtual models for different studies and experiments. These virtual models, shown as virtual models of future status in Fig. 2 (b), have to be managed as well.

Planned changes can be parts of different projects and studies. The desired information model is thus required to manage projects that include several studies and experiments. Fig. 2 (c) shows part of a project consisting of different studies to develop a manufacturing system or make a change that requires conceptual studies (CS) and/or detailed studies (DS). Some studies and experiments will probably be rejected, while others are accepted. Data, information, and knowledge will have been generated in the rejected studies as well as in the accepted ones. By managing both sets, these can be reused later in other studies.

Engineering activities are defined as activities for realizing a product by using different CAx technologies. In the context of this article, they include modeling, simulation, optimization, and knowledge extraction, which differ from manufacturing-related activities, such as machining a part, performing some tests on products, or maintaining a manufacturing resource, etc. For each engineering activity, three groups of data, information, and knowledge have to be identified and managed. Fig. 2 (d) shows these three groups as input, output, and provenance. Inputs and outputs for an engineering activity can be virtual models, data about the properties of the desired manufacturing system, knowledge, and other data that are not related to properties. An example of an engineering activity is the simulation of a cutting process. The virtual models and properties of the tool and part are input data for that simulation, and the result is a simulation model of the cutting process. Besides inputs and outputs, provenance data also needs to be managed to provide more information about the engineering activity. Such data may include the W7 related data like the method of the simulation and the software used, etc.

Based on the above analysis, four interrelated aspects of management for virtual models have been explored:

1 Hierarchical structures in PLM systems and manufacturing data.
2 The lifecycle of manufacturing systems.
3 Projects, studies and experiments.
4 Engineering activities, provenance data and knowledge management.

3.2. Information model Ddesign

The information model can be divided into different packages related

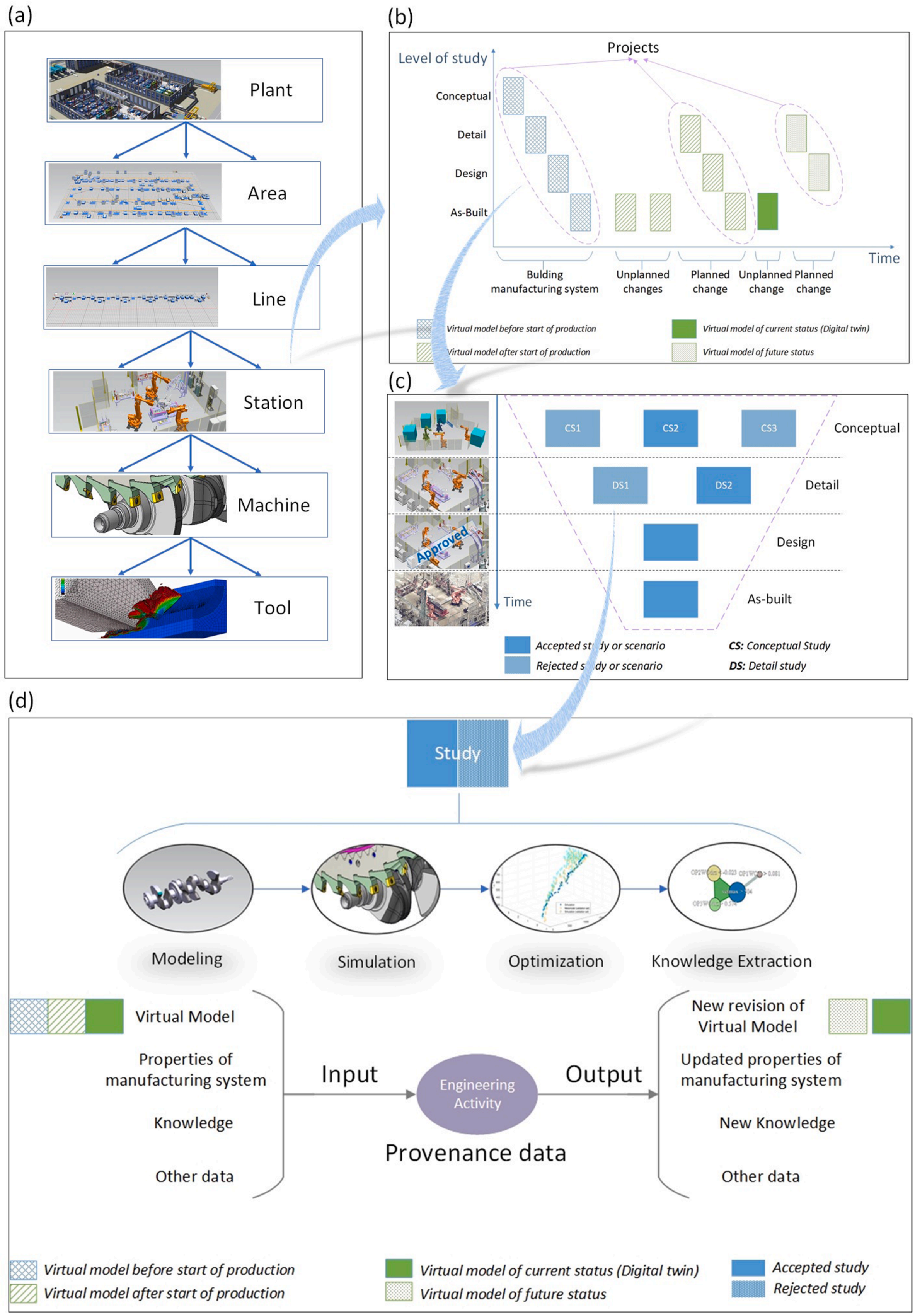


Figure 2. (a) Different levels of manufacturing systems, (b) The lifecycle of manufacturing systems, (c) Development project for a manufacturing system, (d) Studies and engineering activities

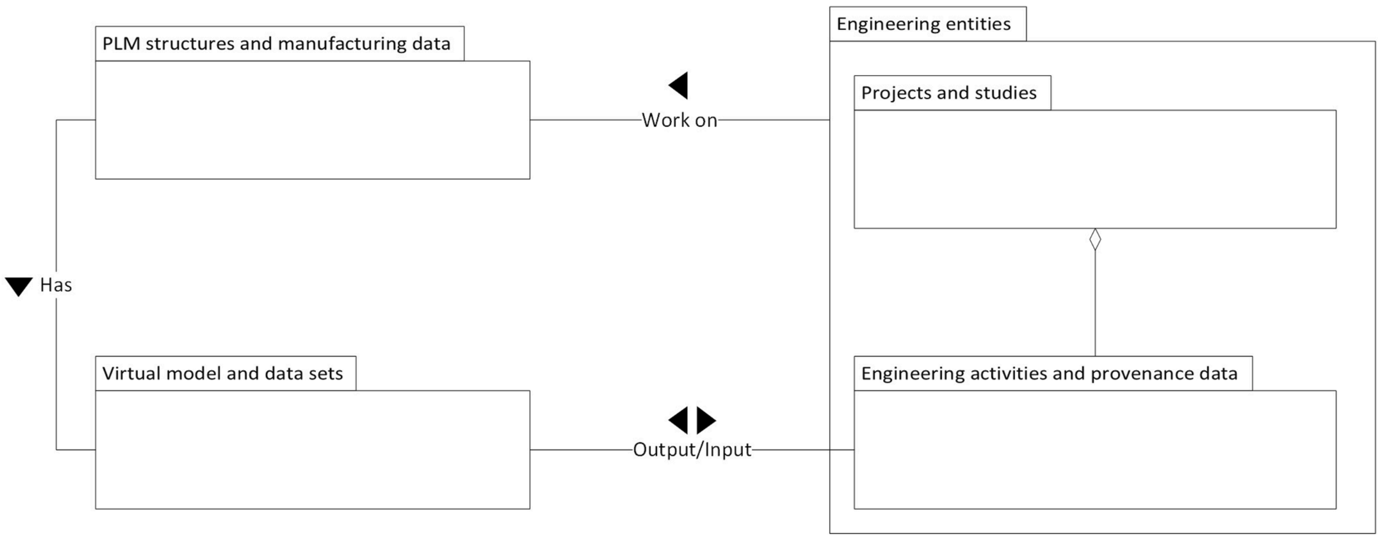


Figure 3. The UML Packages of the information model.

to the four management aspects explained above, as shown in Fig. 3. The first package consists of information models for the main structures supported in many PLM systems, namely, BoM, BoP, and BoR. This package also covers manufacturing data as properties and changes in the manufacturing data. The second package consists of virtual models and data sets that belong to entities in the PLM structure, such as 3D models of a tool or a simulation model of a production line. Classes in the engineering package work on PLM entities. This package consists of projects, studies, and engineering activities. Virtual models and datasets can be either the input to or the results of engineering activities, as indicated in Fig. 3.

Fig. 4 shows an overall view of the information model with all its packages. The first package (P1) consists of the three main hierarchical core structures in PLM systems: BoM, BoP, and BoR. This package also includes a class for manufacturing data ("Property") and a class for saving information about changes in the manufacturing data ("Change"). In the second package (P2), a parent class has been defined and named "Supplementary Entity," with two subclasses of "Virtual Models" and "Data". Virtual models can have a status of either current or future. The "Virtual Model" class uses an enumeration literal to represent its status.

The "Data" class is used to manage any kind of data set used in engineering activities. Note that this is not referring to manufacturing-related data but a simulation specification or an optimization algorithm.

Three classes ("Project," "Study/Experiments," and "Engineering activity") make up the hierarchical structure of the engineering entity package (P3). Each project can contain several studies and experiments, and each study can contain several engineering activities. Studies can have the status of "Accepted" or "Rejected", and maybe on the level of "Conceptual", "Detail", "Design", or "As-built", following the analysis outcome presented in Section 3.1. Fig. 4 also shows that two enumeration literals were defined to identify these choices in the information models. The last part of the information model is engineering activities and their provenance data (P4). For engineering activities, an "Activity" class was defined. It is associated with seven other classes for managing provenance data, as shown in the figure. These seven classes are "Actor," "Time," "Location," "Tool," "Method," Concept," and "Purpose." Engineering activities work on entities in the PLM structure and have an "Input/Output" association with "Supplementary Entity." This allows supplementary entities such as virtual models to be inputs for an activity or outputs of an activity.

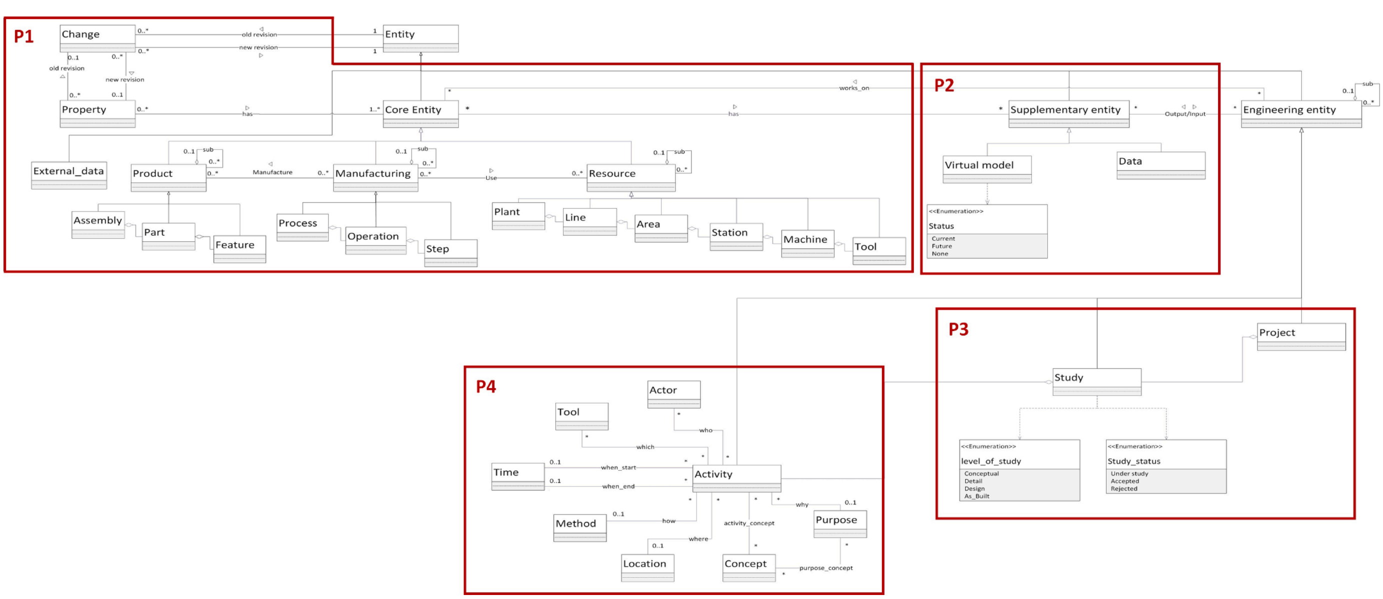


Figure 4. The information model of the extended PLM.

Multiple rules have to be considered when this information model structure is implemented in an application. The full details of these rules are over the scope of this paper, but here two examples can be given. First, subclasses of product, manufacturing activity, and resource classes can be their own superclass, to just one level. Second, virtual models can attain a current status when they are the output of activities that belong to an "As-built" study but later attain a future status when they are the output of activities that belong to an accepted study.

3.3. Information flow diagram

An information flow diagram schematically illustrates parts of the hierarchical structures of manufacturing systems and engineering entities, with their connections to virtual models and their related data, information, and knowledge is shown in Fig. 5. In the figure, station 1 is a manufacturing system in the manufacturing systems hierarchy and the current revision of the virtual model, which is the digital twin of that station assigned to the latest revision of that station. There are also some properties that belong to station 1, and the latest revision of those properties should show the latest manufacturing data about the station.

The information flow starts when an activity uses property data and the virtual model as the inputs. Sometimes other types of data that are not directly related to the station and different knowledge can be used as inputs for an activity. Engineering activity has results, which can be a new virtual model, new knowledge, changes in the manufacturing system properties, or some extra data. Provenance data is also generated during engineering activities. If the study is an as-built one, then the status of the generated virtual model will be the current and it will be attached to the new revision of the station with new properties. If the study is not for an as-built case, then it can be attached to the current revision of the manufacturing system with future status, and the manufacturing system will be changed physically based on this virtual model. Based on the information model and information flow, the latest revision of a manufacturing system entity represents the current status of that manufacturing system. Properties of the manufacturing system are updated, and a virtual model that is the output of an as-built study is attached to the manufacturing system entity. Provenance data management and knowledge management as main extensions to PLM systems are explained in more detail in the next section.

4. Provenance data management and ontology development

Engineering activities such as virtual modeling and simulation with their corresponding data are crucial to building a digital factory in Industry 4.0. Because of the essential role of manufacturing data in Industry 4.0, PLM became the backbone of virtual development [91–94]. As emphasized in this paper, historical data and knowledge can help tremendously for reusing virtual models, and this information and knowledge can be acquired from provenance information and knowledge of engineering activities that led to generating that model. In addition to the development of an extended PLM information model, a provenance management system (PMS) prototype, called Manage-Links, which connects input and output entities to activities, is also developed. This PMS prototype, together with the original PLM system functions and the integrated CAx tools, provides a new platform for managing virtual models with their related engineering activities and provenance data.

As above-mentioned, knowledge management refers to the broad field that concerns the ability to identify, store and retrieve the knowledge to aid decision-making. One of the knowledge management methods that is provided by developed PMS is ontology-based information sharing. Another method for explicit knowledge management within the PMS is managing outputs of knowledge extraction activities. Engineering activities such as modeling, simulation and optimization lead to knowledge extraction by designing and using virtual models. This extracted knowledge belongs to the types of explicit knowledge, such as decision rules, regression models and association rules, as explained. These knowledge types are saved as an output of knowledge extraction activities in the PMS system so that they can be searched and retrieved later.

The developed PMS can be used for different kinds of engineering activities, but here, the focus is on activities such as modeling,

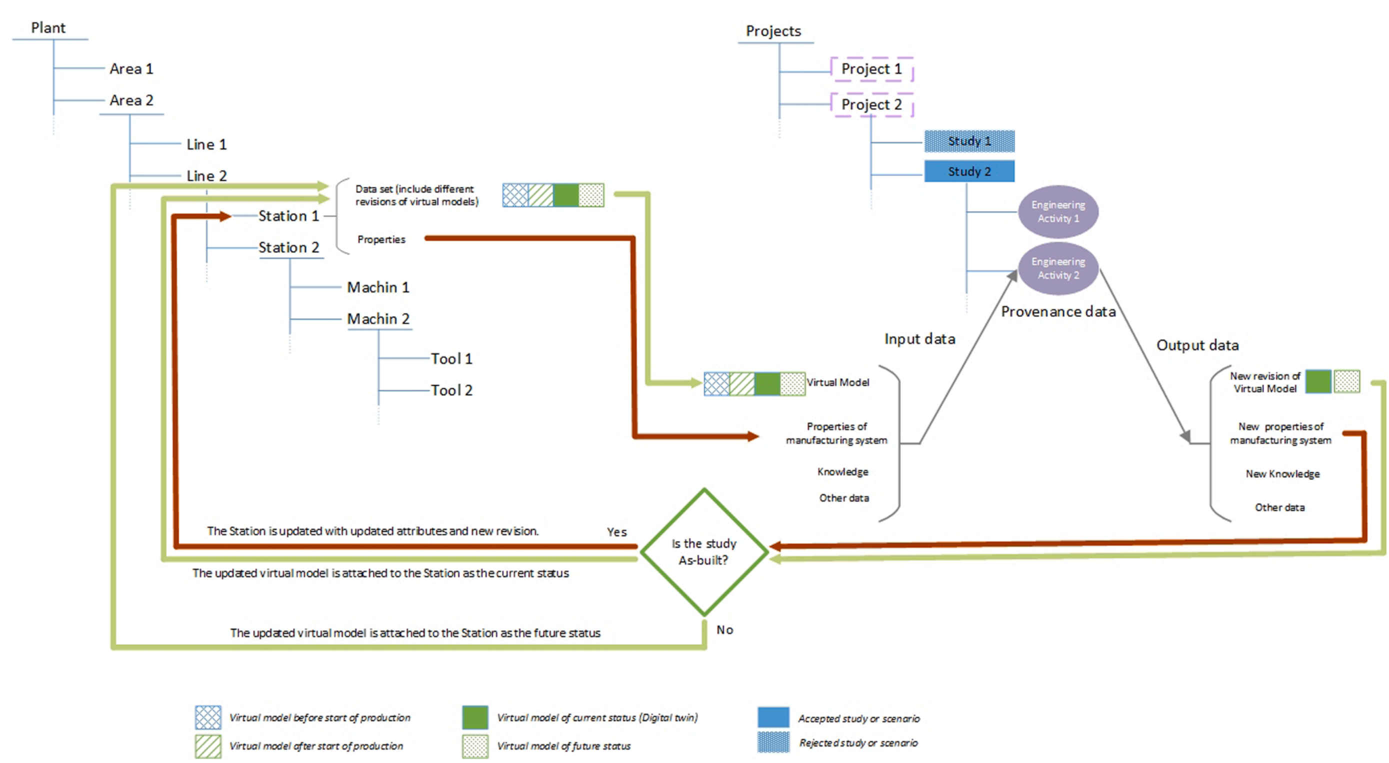


Figure 5. The information flow diagram of virtual models.

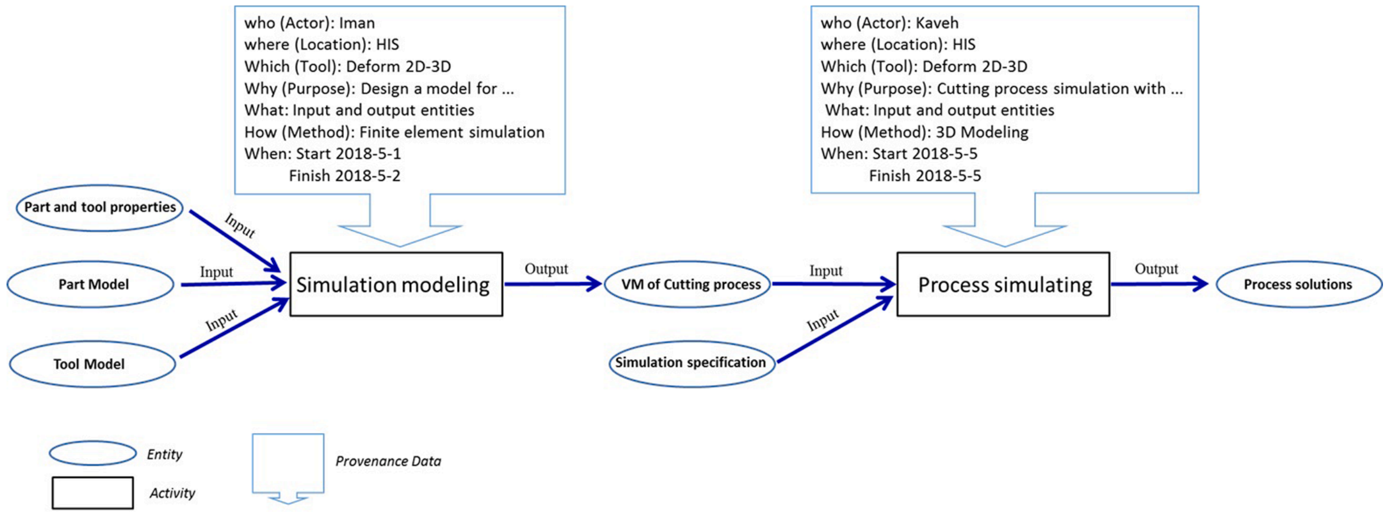


Figure 6. An entity activity sequence with two activities of designing a cutting model and running simulation (adapted from [7]).

simulation, optimization, and knowledge extraction, which are all related to virtual models. Virtual models are used during engineering activities and are also produced from engineering activities. Therefore, engineering activities have a strong relation to virtual models. Each engineering activity can have data, information and knowledge as inputs or outputs. The output of one activity can be used as an input for another activity and these links make a sequence of entity activity. Here, entities can be core entities or supplementary entities from the presented information model and activity means engineering activities.

Fig. 6 shows an example of an entity activity sequence with two activities of designing a cutting model and running a simulation with that model. In addition to the inputs and outputs of engineering activities, there is also provenance data for each activity. Provenance data provides information about the situation in which that activity happened and answers the W7 questions.

4.1. The provenance data model

Fig. 7 shows the fourth package of the developed UML class diagram in more detail, which is designed and implemented based on a concept by Oscarsson et al. [7]. In this model, the "Activity" class was defined to add engineering activities and is associated with the "Entity" class. Entities are objects that are used in activities as inputs or outputs. Entities have been retained in the integrated PLM system, and their identifier originated from there. An entity can be an object from PLM structures such as a machine or a part, or it can be a virtual model, data, or extracted knowledge. "Location," "Purpose," "Method," "Actor," "Tool," and "Time" are six different classes in the data model, which have been defined to cover questions similar to W7. The "Location" class stands for "Where" questions, and it specifies the location of the associated activity. It can be a physical location, the department, or the discipline in the enterprise. The "Purpose" of the activity class provides the reasons for executing the activity and gives important clues as to which other purposes the generated entities (such as virtual models) can be used for. The "Method" identifies how the activity is executed. For example, Finite element simulation is a method used for simulation modeling. In the data model, the "Actor" class is used to determine the person (or automated agent) that performed the activity. The "Tool" class specified the engineering software program used for the activity, and finally, the "Time" class records the start time and end time of the activity. There is also a class in the data model for associating concepts to other objects such as entities and activities. These concepts can be linked to concepts in manufacturing domain ontologies. The relations between different objects in the PMS with their links to the subclass hierarchies of concepts and standardized names can be used to uniformly express and index knowledge. This knowledge can be searched and reused later. The identifiers for the objects stored in the PLM system are translated into Resource Description Framework (RDF) notation and can then be used to form knowledge statements in terms of the domain and provenance ontologies.

Note that Fig. 7 is an integral part of the overall information model (Fig. 4). The link to the information model is via the class “engineering entity”, which is the subject of the change management part in Fig. 6 as well. The link of the provenance data model to the ontology is established via the class “Concept”. It allows the classes of the provenance data model to be associated with ontology concepts. At the data level, this is instantiated by tagging activities, actors, methods, and entities by the labels of ontology concepts. Our provenance data model extends the original W7 provenance data model by this ability and by the central role of the class “Activity”. Moreover, note that the classes Activity, Entity, Actor, Method and Tool are also top-level concepts of our ontology, i.e., they are essentially subclasses of the classes in the provenance data model.

4.2. Manage-link and its integration with a PLM system

The described data model for provenance management can be considered as an extension proposed to existing PLM systems. In this work, Teamcenter (Tc) from Siemens has been selected as an exemplified PLM platform for implementing Manage-Links because of its comprehensiveness and commonality in the industry. It also provides multiple integrated applications and solutions with its open framework. The provenance data model has been implemented to the Manage-Links outside of the Tc database due to two reasons. First, the accessibility to the Tc database and the limitation for defining dependencies in the Tc data model. The second reason is to ensure the generalizability of the Manage-Links. When the Manage-links has a separate database, then it is independent of PLM system and it can be integrated and used with other available PLM systems. Tc offers an opportunity to customize and integrate user-defined applications with the database. These customizations can be done in different ways, such as server customization, portal customization, web client customization, service-oriented architecture (SOA) customization, and Business Modeler IDE (integrated development environment) customization. As is shown in Fig. 8, the provenance data model has been implemented on a high-performance database (Caché from InterSystems) and it is integrated with Tc through its application programming interface (API). Objects from PLM structures get their identifier from the Tc database and are used as inputs or

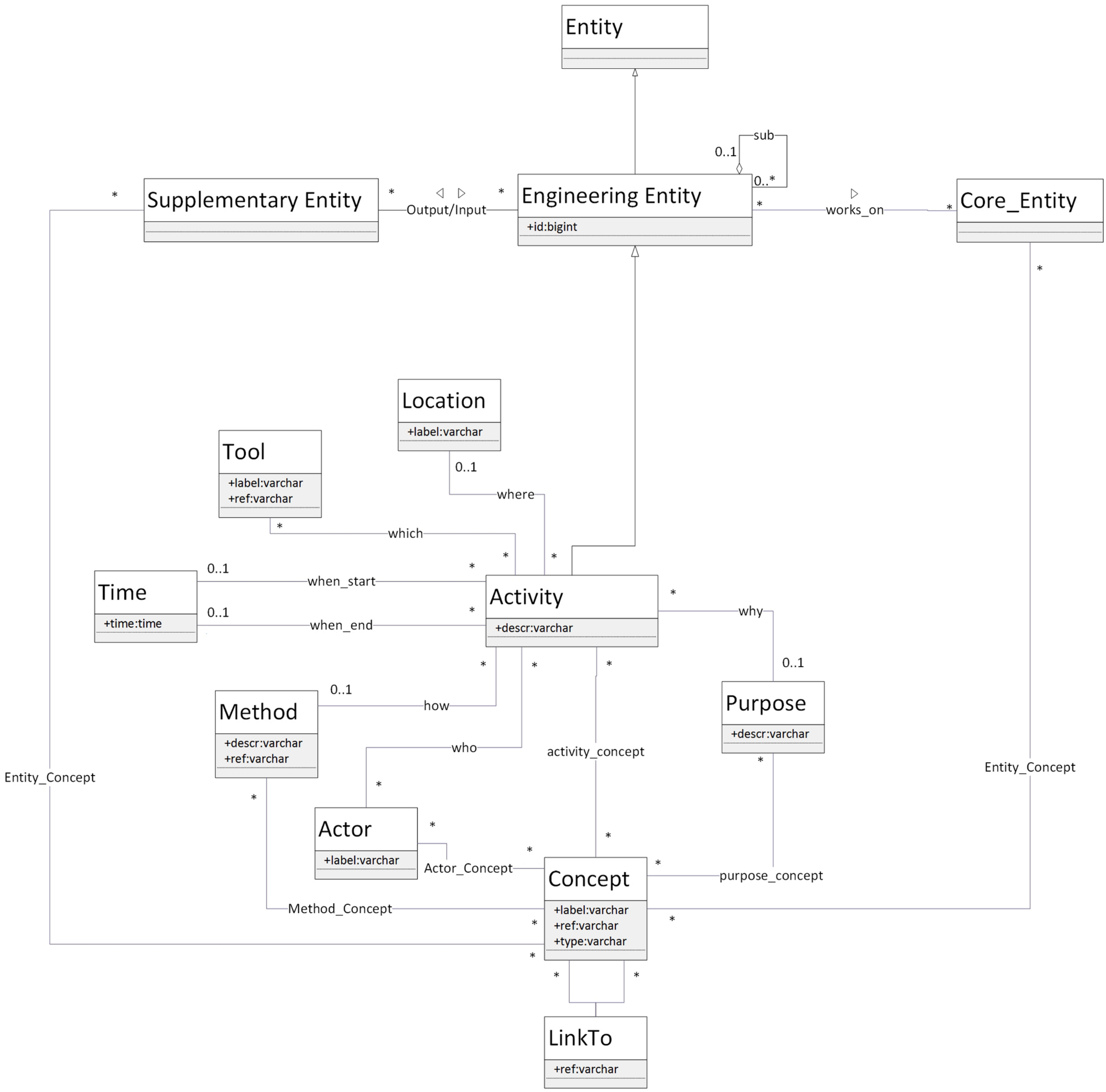


Figure 7. Provenance data model.

outputs of engineering activities in Manage-Links.

The database is also connected to an open-source ontology editor (Protégé) for the purpose of defining ontologies. Concepts in Manage-Links can be linked to concepts in manufacturing domain ontologies. Fig. 9 shows the Manage-Links user interface after login into the application. In the current implementation, whenever a user login to the Manage-Links through the PMS user interface, Manage-Links connects to the Tc server and retrieves the data from the Tc database. Manage-Links consists of different tabs for data entering. The "Activities" tab is a place for browsing the list of all activities in the PMS. Users can also add a new activity in this tab. Items (business objects) can be searched and added from Tc as entities to the list of entities in PMS, through the "Entities" tab by the users. Users can store all ontological terms that are candidates to describe the classes (Entity, Purpose, Actor, Method, and Activity) in the "Concepts" tab. The ontological terms can be referred to where they are defined, such as a link to a web page or a link to Protégé. "Location", "Tools", "Actors", "Projects", and "Methods" are other tabs to enter data about activity and cover all W7s questions.

The search tab in Manage-Links allows users to search for previously added data such as activities, entities, tools, and methods, etc. For example, users can find all related activities to a specific virtual model or find all methods that are used to extract knowledge from specific optimization results. The Manage-Links has a graphical component that visualizes the dependencies stored in the Caché database. The graph can visualize the activities either with a full, ancestors or descendent activities and entities (see also Fig. 11 upper-right). This graph helps when tracing activities and entities to see their full history. The Manage-Links application is integrated with an ontology editor tool. Since the ultimate

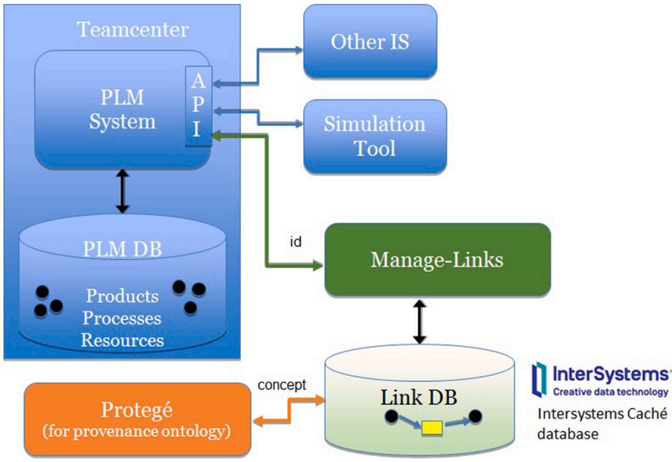


Figure 8. Structure of PMS and its integrations [95].

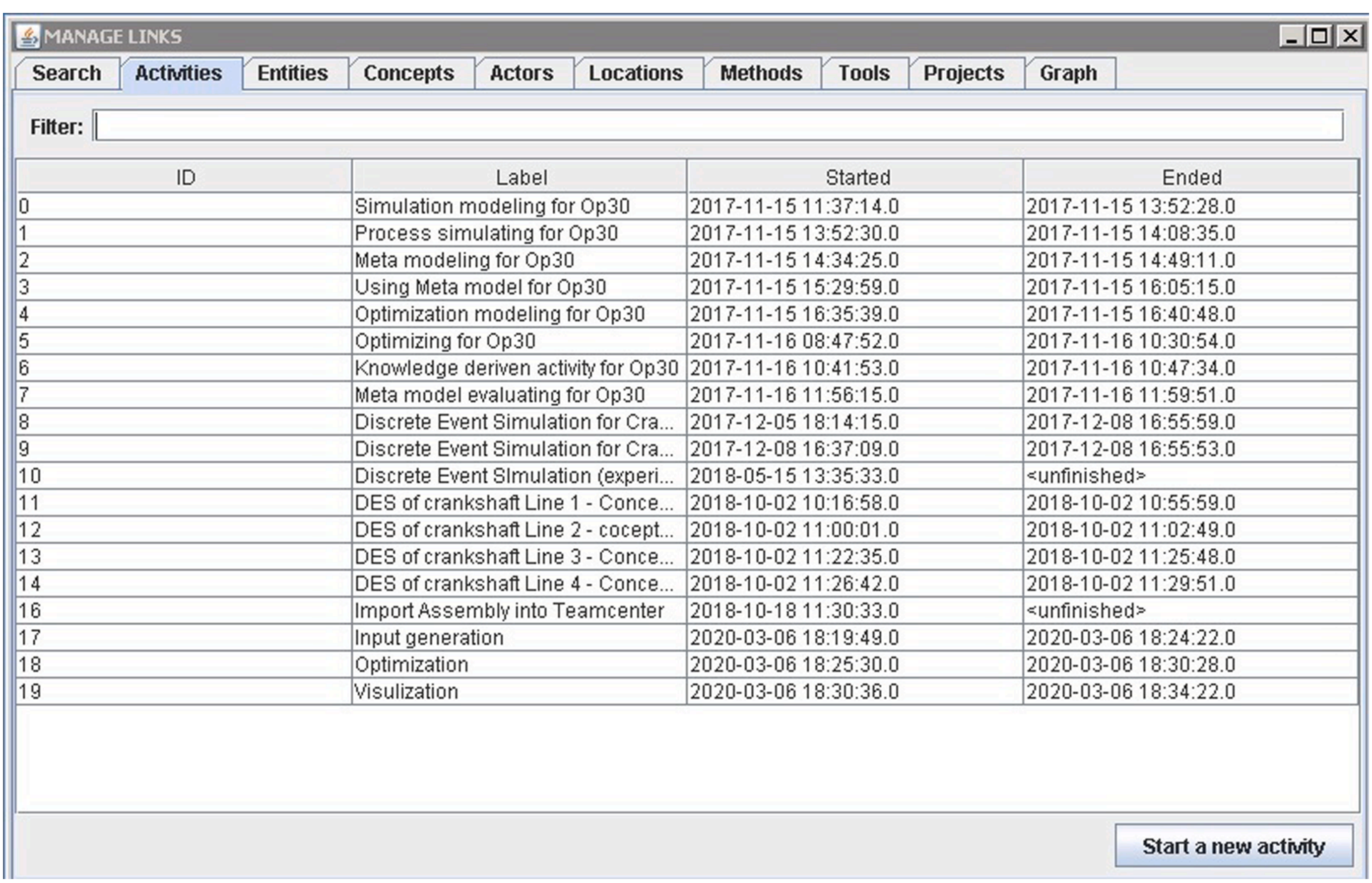
use of PMS is managing knowledge, the following sub-section will explain this integration in more detail.

4.3. Ontologies in PMS

As previously mentioned, the "Concept" class was defined in the provenance data model, and there is a link in the Manage-Links user interface for adding concepts and referring those concepts to any URL on the web. Concepts are referred to as classes of ontology in the virtual engineering domain within the Protégé ontology editor. Fig. 10 shows the concept tab and the concept editor window. Concepts can have different types, and they can be selected for different items in Manage-Links based on their types.

As shown in Fig. 11 (lower-left), several super-classes have been defined for engineering activities, entities, methods, purposes, tools and actors in the VE-ontology. Each of these super-classes consists of sub-classes that are defined based on what has been entered in Manage-Links. Input and collaboration with domain experts are required for creating an ontology, but domain experts are rarely experienced in ontology development [96,97]. Even though the role of domain experts in building ontologies is unavoidable, there are doubts about the quality of their performance. One study showed that eight experts had an average of 2.65 different opinions about the correct classification of a domain entity in an upper ontology [98]. The virtual engineering ontology (VE-ontology) has been developed based on the data added to the Manage-Links application by experts instead of the experts' direct contribution. This means that the concepts and their relationships were defined based on engineering activities that happened in the past. This process prevents misinterpretation and differences of opinion about classes and relationships because the provenance information is provided. Also, the ontology will be enriched using the data entered into Manage-Links over time, and this will prevent missing concepts.

Fig. 11 (lower-left) also shows the first level of the object property hierarchy of the ontology. Properties have been defined according to associations that exist in the provenance data model. The table in this figure shows different properties and the kind of classes that can be selected as domain and range for each property. The "useInput" properties correspond to the input association in the provenance data model. They have domains from subclasses of the "EngineeringActivity" class and range from sub-classes of the "Entity" class. For clarification, an example of two engineering activities that have been done by engineers for a cutting process of automotive crankshaft production in OP30, the industrial application case which will be presented in the next section, is



ID	Label	Started	Ended
0	Simulation modeling for Op30	2017-11-15 11:37:14.0	2017-11-15 13:52:28.0
1	Process simulating for Op30	2017-11-15 13:52:30.0	2017-11-15 14:08:35.0
2	Meta modeling for Op30	2017-11-15 14:34:25.0	2017-11-15 14:49:11.0
3	Using Meta model for Op30	2017-11-15 15:29:59.0	2017-11-15 16:05:15.0
4	Optimization modeling for Op30	2017-11-15 16:35:39.0	2017-11-15 16:40:48.0
5	Optimizing for Op30	2017-11-16 08:47:52.0	2017-11-16 10:30:54.0
6	Knowledge deriven activity for Op30	2017-11-16 10:41:53.0	2017-11-16 10:47:34.0
7	Meta model evaluating for Op30	2017-11-16 11:56:15.0	2017-11-16 11:59:51.0
8	Discrete Event Simulation for Cra...	2017-12-05 18:14:15.0	2017-12-08 16:55:59.0
9	Discrete Event Simulation for Cra...	2017-12-08 16:37:09.0	2017-12-08 16:55:53.0
10	Discrete Event SImulation (experi...	2018-05-15 13:35:33.0	<unfinished>
11	DES of crankshaft Line 1 - Conce...	2018-10-02 10:16:58.0	2018-10-02 10:55:59.0
12	DES of crankshaft Line 2 - cocept...	2018-10-02 11:00:01.0	2018-10-02 11:02:49.0
13	DES of crankshaft Line 3 - Conce...	2018-10-02 11:22:35.0	2018-10-02 11:25:48.0
14	DES of crankshaft Line 4 - Conce...	2018-10-02 11:26:42.0	2018-10-02 11:29:51.0
16	Import Assembly into Teamcenter	2018-10-18 11:30:33.0	<unfinished>
17	Input generation	2020-03-06 18:19:49.0	2020-03-06 18:24:22.0
18	Optimization	2020-03-06 18:25:30.0	2020-03-06 18:30:28.0
19	Visulization	2020-03-06 18:30:36.0	2020-03-06 18:34:22.0

Figure 9. The Manage-Links user interface.

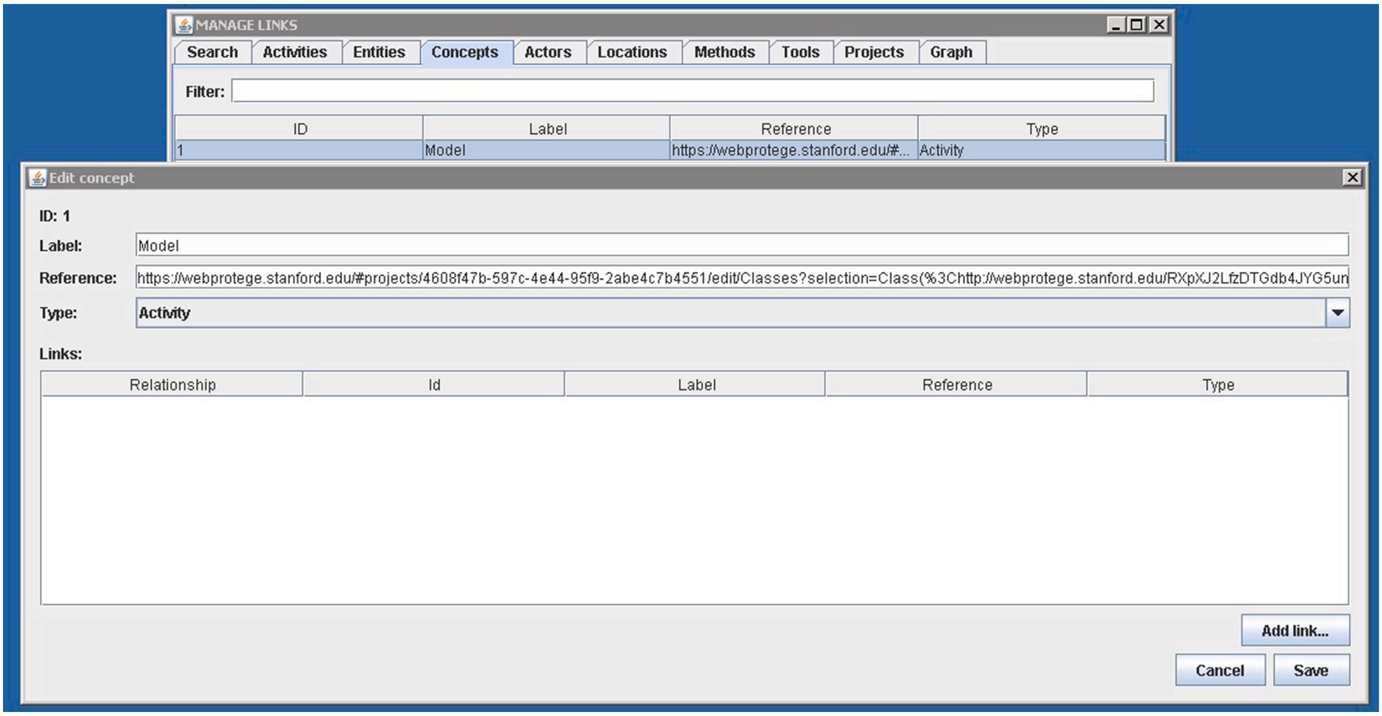


Figure 10. The concept tab and the concept editor window of Manage-Links.

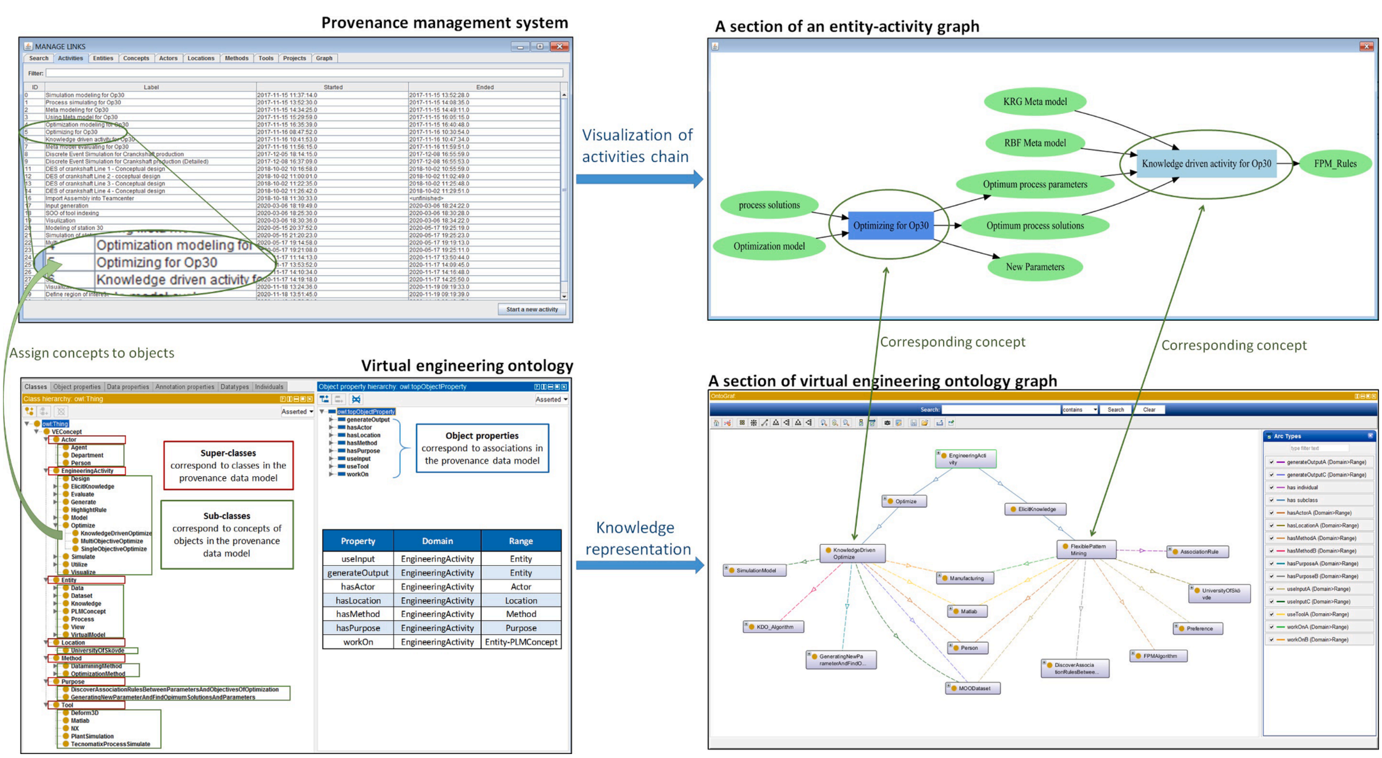


Figure 11. Interaction of Mange-Links and Virtual engineering ontology.

presented in the figure. These activities are "optimizing for OP30" and "knowledge-driven activity for OP30." They were entered into Manage-Links with their provenance data. Fig. 11 (upper-right) shows these two activities and their inputs and outputs in Manage-Links. Based on these activities, two classes of "KnowledgeDrivenOptimize" and "FlexiblePatternMining" were defined as sub-classes of "EngineeringActivity." If their types of provenance data had not been defined before in the ontology, new classes would have had to be defined. The provenance data are connected to activities with corresponding property types, as shown in Fig. 11 (lower-right).

This example shows how concepts of instances in Manage-Links were defined in the VE-ontology. Together with specifying their relationships, knowledge about these engineering activities has been built and presented. Next time an engineer intends to conduct a similar activity, comprehensive knowledge about the needs and processes of that activity will exist. By using Manage-Links, engineers can enter their engineering activities and entities into the database and add concepts to the virtual engineering ontology. In the long run, using Manage-Links and merging the virtual engineering ontology with the available manufacturing ontologies will improve and extend the ontology to uniformly expressed and indexed knowledge.

5. A Multi-level industrial application

The above-mentioned OP30 represents a complete application example of Manage-Link. The automotive factory has four production lines consisting of multiple workstations. In each workstation, one or several machines and equipment are working, and some of those machines have tools attached to those machines. For reducing the production time, the bottleneck in the crankshaft production area and four production lines had been identified by DES using the method described in [99]. The bottleneck was the station called operation 30 (hence the name OP30), which consists of two gantry robots for transporting crankshafts and two turning machines. Through the simulation of OP30, the cutting process in turning machines had been selected to be optimized. Each turning machine consists of different types or a set of tools in a specific order, based on the process plan. Tool turret (magazine or automatic tool changer) can hold multiple tools in the tool indexes. For optimizing the process in OP30, the tool sequencing and metal cutting steps had been optimized and the optimization results had been used for knowledge extraction. Fig. 12 shows the four applications in this multi-level example, which are:

1 Area and line levels: Modeling and DES simulation on the crankshaft production area and its four production lines.
2 Station level: Modeling and simulation of two gantry robots in station 30 in the first production line.
3 Machine level: Optimization of tool-indexing in the turret magazine of one of the turning machines in OP30.
4 Tool level: Modeling, simulation, optimization and knowledge extraction of metal cutting at the tool level in a turning machine.

Full details of how extended PLM has been used for the two lowest levels (machine and tool) are included in the following sub-sections.

5.1. Machine-level application: tool indexing optimization

This application study is at the machine level of the manufacturing system. There are two turning machines at OP30 and each machine has two tool turrets. One of the gantry robots transports a crankshaft to the machine and places it between the chucks. In the operation, the tool cuts

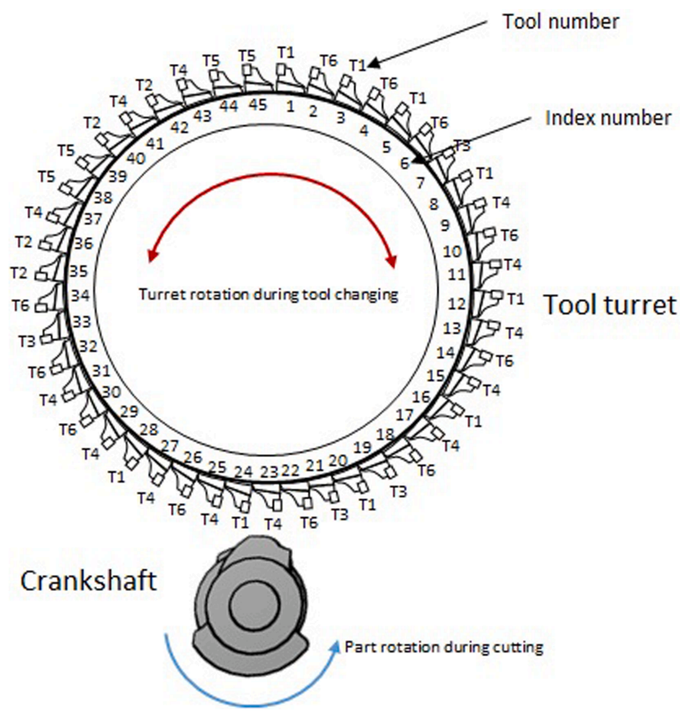


Figure 13. Tool indexing on turret magazine.

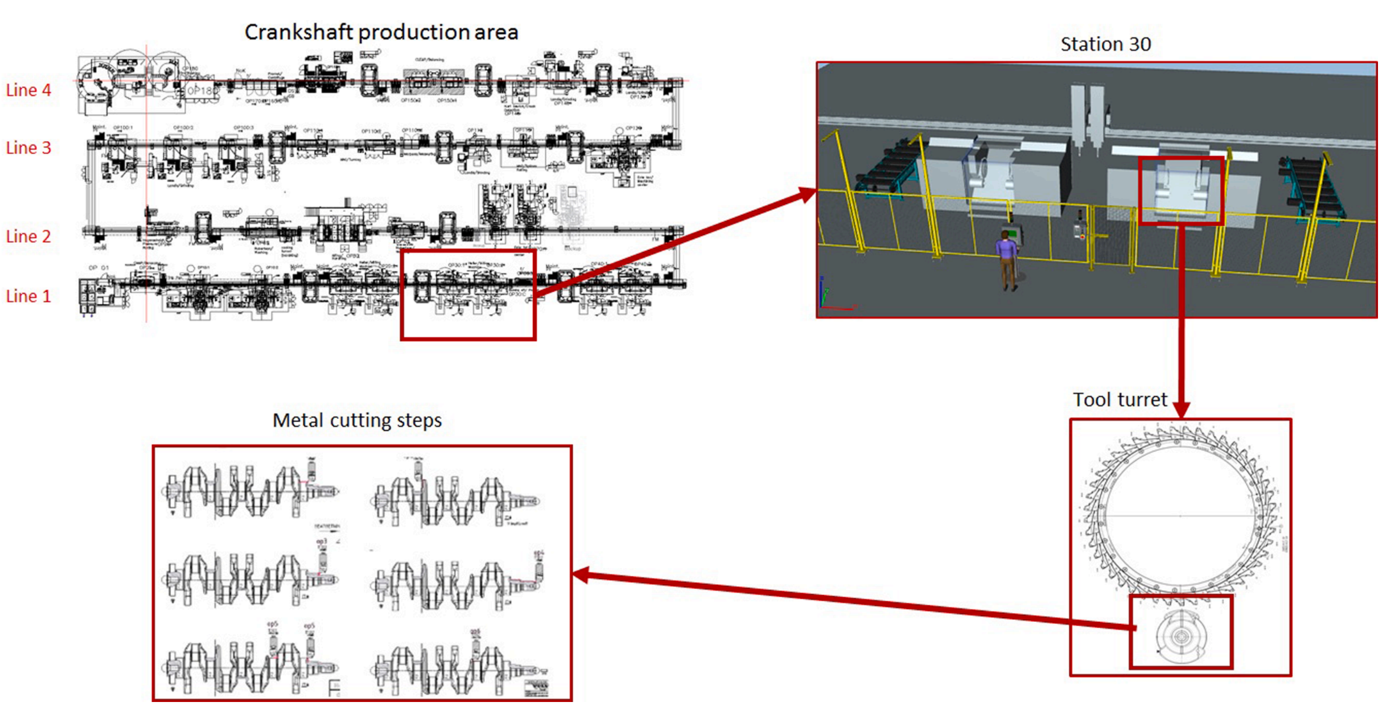


Figure 12. Multi-level application studies of crankshaft manufacturing.

metal by using different tool inserts in a number of steps. Each turret has several tool inserts located in different insert holders (Fig. 13). The turret rotates around the vertical axis after each cutting step and adjusts another insert in line with the crankshaft for the next step. The time to rotate from one insert to another is the tool indexing time.

For the tool indexing improvement, both single-objective and multi-objective optimizations have been used to minimize the total tool indexing time, taking into account the entire life-span of all the tools in the turret magazine. Each insert has a life-span and can be used for a fixed number of cutting steps. For this reason, the same insert types are located in several index locations on the turret so that that all inserts are worn out and need to be changed at the same time. In this study, six types of inserts were distributed in 45 insert holders in the turret.

The product is the crankshaft from the BoM and the manufacturing system is the turning machine at OP30. The selected operation for this study is "OP-30-Turning Machining" from the turning process for the main bearing. MATLAB is the main software program for engineering activities such as optimization. Even though MATLAB is not known as a CAx tool, it is commonly used by engineers for running algorithms and simulations. It offers the same functionality as other CAx, and can be considered as a CAx. In addition to MATLAB files, 2D drawings of the turret were used as input data for input generation. The MATLAB dataset had previously been defined in Tc, and by uploading .mat or .m files to Tc, they can be opened in MATLAB directly.

An optimum solution for locating inserts in different insert holders was found by running a single-objective optimization. Since changing the location of inserts is costly and time-consuming, the other objective was to minimize the number of changes in the location of inserts. The result of the MOO activity is a Pareto-front formed by non-dominated solutions with the minimum number of changes and minimum total indexing-time as objectives. Some patterns were identified in the repeated sequences of the index number of different inserts in different non-dominated solutions as a form of knowledge extraction. For example, in half of the solutions, the first eight index numbers belong to tool numbers of 6-2-1-2-1-2-1-1.

The optimization results were also saved as MATLAB files that were stored into Tc as MATLAB datasets. The extracted knowledge is a JSON file that was uploaded into the PLM system and can be searched and retrieved there. Fig. 14 (top) shows how the optimization activities are divided into two optimization runs, a single-objective optimization and a MOO. Those two substudies are children of the "Tool sequence optimization" study which belongs to the project. These activities are also attached to the "OP30-turning machining" operation at the machine level Fig. 14 (bottom).

The provenance data were entered into Manage-Links for each activity in the same way as described previously. The last step is connecting entities from Tc to engineering activities in Manage-Links as illustrated by the Entity-Activity graph in Fig. 15. The graph shows that

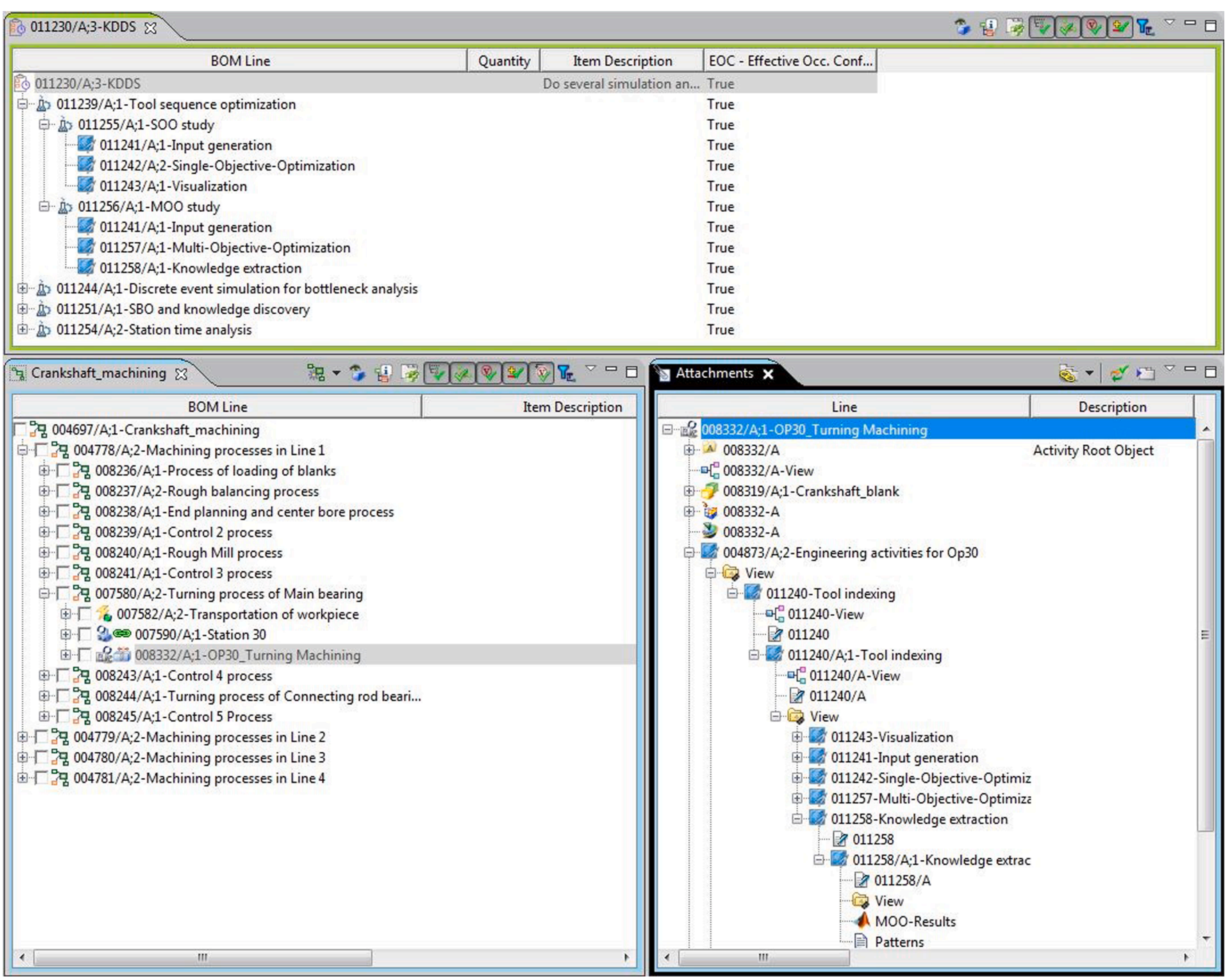


Figure 14. Engineering entity structure (top), bill of process, and attached activities (bottom) for the tool indexing optimization.

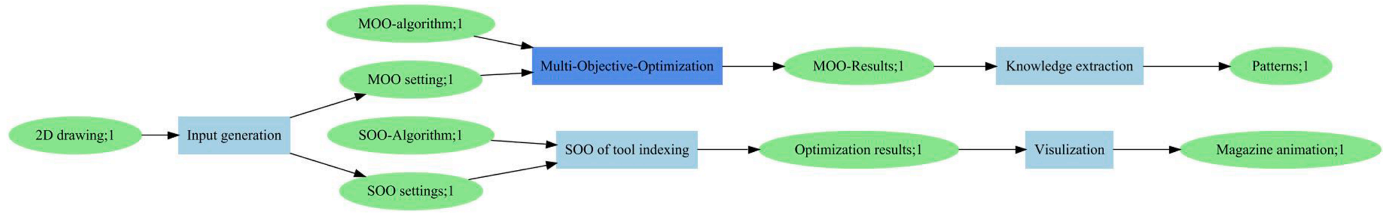


Figure 15. The Entity-Activity graph for tool indexing optimization.

the "Input generation" activity has an output for both the single-objective optimization and MOO activities.

The results of this application study showed that 36 changes of insert locations reduce the tool indexing time for each part by 70% [100]. By managing these engineering activities and their related data and knowledge, a user can trace back to find how these results were achieved and evaluate the methods used and consider reusing the algorithms in another case.

5.2. *Tool-level application: metal cutting SBO and knowledge discovery*

One of the cutting steps in OP30 (machining of the main bearing in a turning operation) was modeled, simulated, and optimized by optimization methods. Again, knowledge was extracted from the optimization results but, in this case, in the form of metamodels. It would then be possible to replace the simulations by the trained metamodels that approximate the actual simulations to reduce the running time [100]. The aim of this optimization was to find optimal cutting data and develop a metamodel approach to speed up the optimization of the cutting step. This study is one of the complete application studies in this research in that it covers several types of activities such as simulation modeling, process simulation, constructing metamodels, using metamodels, metamodel evaluation, optimization, and knowledge extraction.

From the above activities, the simulation modeling activity is explained below as the first activity in this application study. For this activity, the CAx tool is the Deform 2D/3D software program for finite element analysis. The same three main structures (BoM, BoP, BoR) prepared for the previous application studies were used. However, in this study, the focus has shifted to the tool level and cutting step.

The integration of Deform 2D/3D with Tc allows files generated by Deform 2D/3D to be saved in Tc and be opened from Tc. A spreadsheet

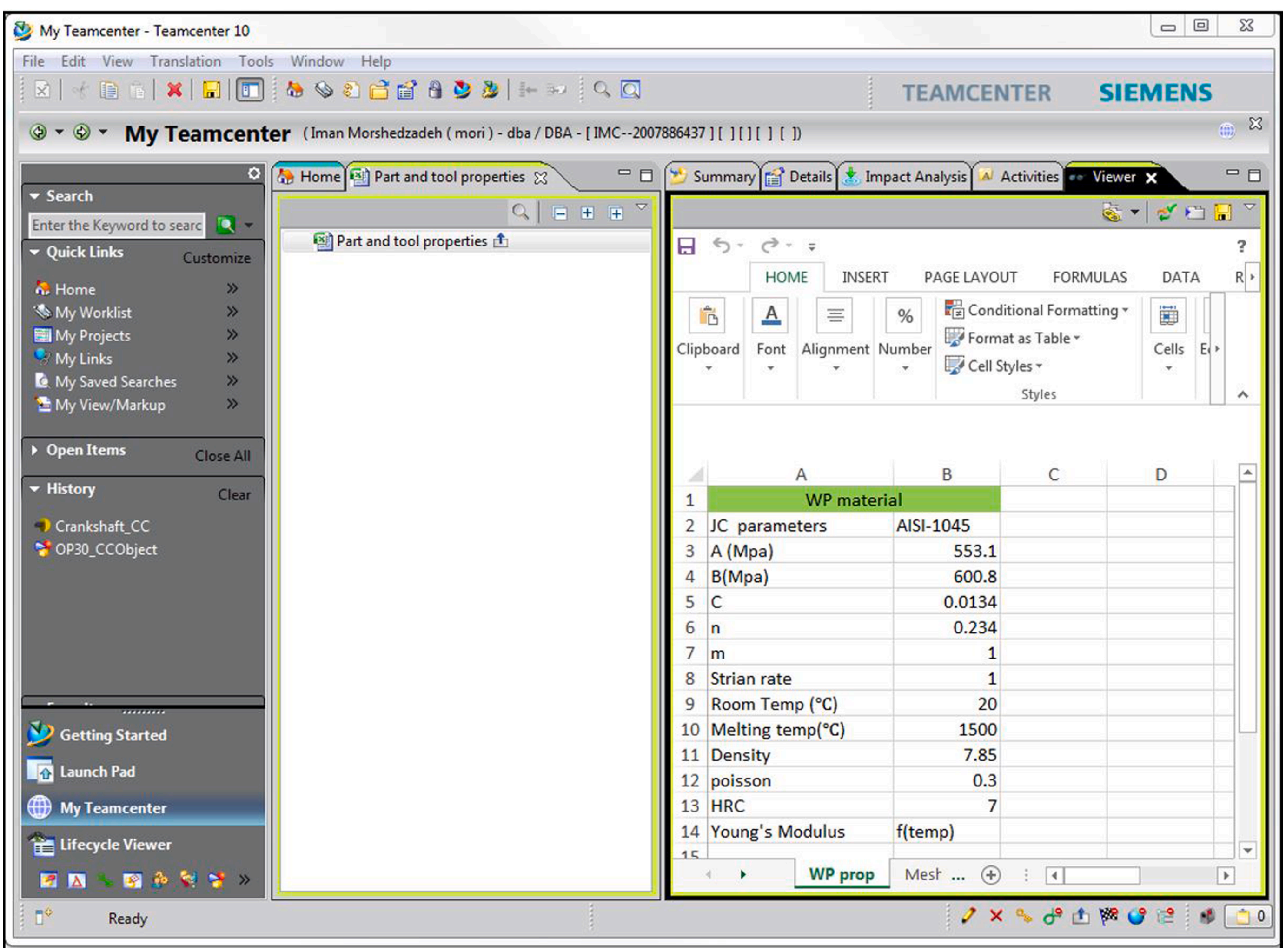


Figure 16. View of spreadsheet file in the Teamcenter environment.

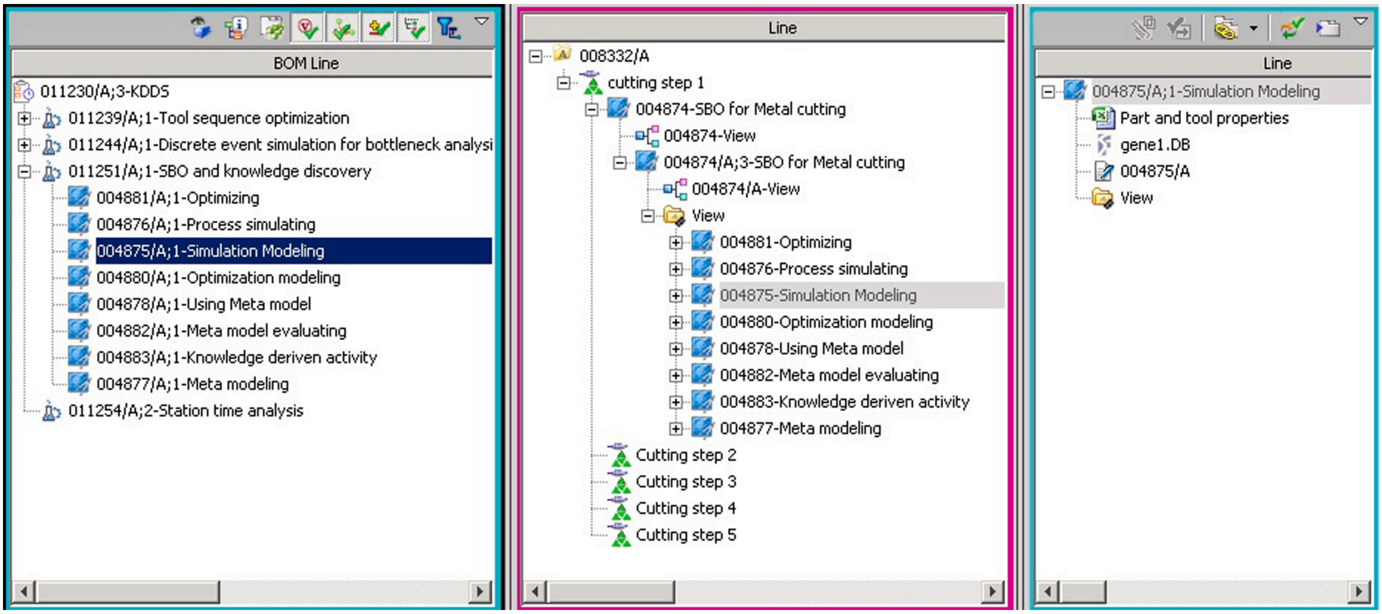


Figure 17. Engineering entity structure (left), cutting step (middle), data attached to the activity (right).

file with all the required data to generate the cutting simulation model was created and added to Tc.

Fig. 16 shows the uploaded spreadsheet file in Tc. It contains the data needed for simulation modeling, such as workpiece properties, mesh data, tool properties and process specifications.

The spreadsheet file with the input data can be exported from Tc and saved on the user's computer when designing the cutting process. These data were used in the Deform 2D/3D application to model the cutting process. For the designing and running experiments step, the workpiece and the tool are modeled based on the data in the spreadsheet file. The model is meshed and the simulation file is generated in a database format (.DB).

The output of this activity is a Deform 2D/3D database (.DB) file. This file is created by Deform 2D/3D after the simulation is completed. It contains the information for all the finite element simulation steps from the cutting operation of the main bearing in the crankshaft production. The simulations can be regenerated and the results can be analyzed by using this database file. Like the previous application study, an engineering activity item was created and attached to the BoP, but at the step level (Fig. 17 middle). The engineering activity created for simulation modeling is also attached to the "SBO and knowledge discovery" study in the project hierarchy, as shown in Fig. 17 (left).

On the right-hand side of Fig. 17, an spreadsheet file that contains the properties of the workpiece and the cutting tool was attached to the simulation modeling activity as the input data and the "gene1.DB" output file was attached as the Deform 2D/3D output model. After structuring the data in the PLM system, the data need to be connected to the activity with provenance data in the next steps.

In the step of providing the provenance data, an activity of "Simulation modeling for OP30" is created in Manage-Links and all provenance data including time, location, method, purpose, actors, and tools are entered into the PMS. The last step for this activity was attaching related data from the PLM system to the activity as inputs and outputs. The spreadsheet file of "part and tool properties," the "OP30 main bearing turning" operation, the "Spindle_Late" turning machine and the "crank-shaft" part are items from Tc. They were attached to the simulation modeling activity in the PMS as input entities. The "gene1.DB" item from Tc was the only output of this activity. The "Simulation Modeling" activity was the first of the activities of the SBO and knowledge extraction process to be implemented for the turning operation of the main bearing in the crankshaft production. Fig. 18 shows the entire Entity-Activity graph for those activities, which are shown as rectangles.

In this SBO study, the objectives of optimization were maximizing material removal rate (MMR), minimizing wear depth (ω), and minimizing the interface temperature (Tint) by changing different tool variables such as clearance angle, rake angle, cutting angle radius, cutting speed, and feed rate. The results of the MOO are shown as a Pareto front in Fig. 19 [101].

Knowledge generated in the form of rules from the MOO are represented in a JSON file in the following format:

```
{
  "RuleID": 1,
  "SelSignificance": 61.76,
  "UnselSignificance": 23.03,
  "Rule": "Feed > 0.37894"
  },
```

This rule means 23.03% of all solutions and 61.76% of Pareto solutions have a feed rate of more than 0.37894 (mm/rev). The last activity shown in Fig. 18 is about knowledge discovery, and the output is the

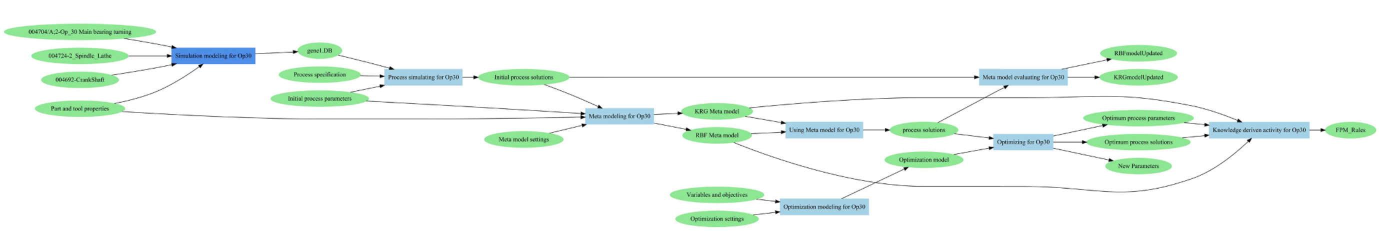


Figure 18. Entity-Activity graph of simulation-based optimization and knowledge extraction for a cutting step in OP30.

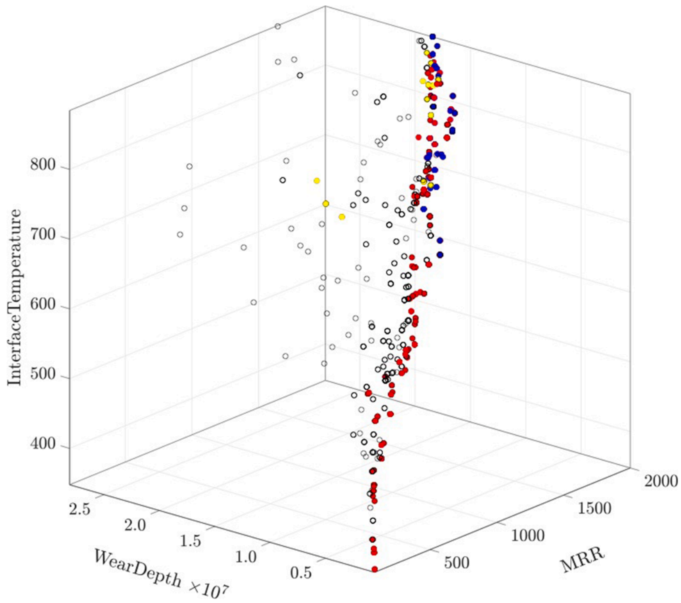


Figure 19. Pareto-front from simulation-MOO [101]

JSON file of explained rules. Engineering activities and entities of SBO and knowledge extraction were also added as Protégé classes to express uniformly and index knowledge. By defining all activities and connecting inputs and outputs to them, in the SBO and knowledge extraction study, each entity, such as a virtual model, can be analyzed according to its previous connected activities and provenance data.

6. Conclusions

This research had emerged from a need in the industry for reusing virtual models and the generated knowledge from them. The extensive usage of virtual models in different application areas produces a vast amount of models that demand long development times and comprehensive efforts to develop. Most of the virtual models and the gained knowledge from designing and using them can be employed again. Nevertheless, because of the lack of proper management, they are not easy to be found at best or even lost at worst. While virtual models are supposed to represent real objects, they cannot imitate all behaviors of their real-life counterparts entirely. Because of that, just having access to a model is not enough for reusing it for another case - it needs information about the origin and causation of that model. There is a tremendous advantage if the gained knowledge from producing and using that model is also available.

In principle, PLM systems can provide the proper platform for managing virtual models because of their integration with various CAx and capability for structuring and managing data. By studying the related research works, several standards and information models have been found for managing product and production-related data in PLM systems. There are also several information models available relating to virtual models, such as information models for managing input data of simulations. But there is a lack of any information model for managing virtual models themselves and also their related data, information and knowledge, through the PLM structures. In this paper, we have proposed four extensions or "packages" to PLM systems to cope with these new requirements: (1) extending the main hierarchical structure in PLM systems; (2) adding two subclasses of "Virtual models" and "Data" defined to cover the management of virtual models with their status; (3) introducing an information model for managing studies and projects, and (4) adding the last package of the developed information model is engineering activities with their provenance information.

A new information model has been designed to be aligned with the existing core structure in PLM systems. The first three packages of the information model had been deployed on a PLM system and for managing provenance data, a provenance management system had been developed and integrated with the PLM system. In addition to that, an ontology editor tool has been used for building a VE-ontology for engineering activities and their inputs and outputs to provide the well-defined, precise and formal semantic specifications in the area of manufacturing generally and engineering activities and virtual models particularly. Application examples in tool indexing and metal cutting optimization in an automotive manufacturer have demonstrated the outcomes of this research, including the information model, provenance data model and the VE-ontology. These are done through the implemented Manage-Links prototype system which uses Siemens' Tc as an example of an extended PLM system to demonstrate the applicability of the concepts introduced in this paper for storing, searching and retrieving the virtual models and the knowledge extracted from optimization runs. In future work, we will investigate how the Entity-Activity graph representation can be replaced by by a knowledge graph, which can provide more advanced features of linking the model objects and knowledge.

Funding

This work was supported by the Knowledge Foundation (KKS) through the IPSI Research School at the University of Skövde [grant number 20140330], and VF-KDO Profile research project [grant number 20180011].

CRediT authorship contribution statement

Iman Morshedzadeh: Conceptualization, Writing – original draft, Methodology, Visualization. **Amos H.C. Ng:** Project administration, Supervision, Writing – review & editing. **Manfred Jeusfeld:** Conceptualization, Methodology, Writing – review & editing. **Jan Oscarsson:** Conceptualization, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research is partially financed by Knowledge Foundation (KKS), Sweden, through the IPSI Research School and later the VF-KDO Profile. The authors would also like to acknowledge the technical supports from Volvo Car Engine, Sweden.

References

[1] Y. Chen, Industrial information integration—a literature review 2006–2015, Journal of Industrial Information Integration 2 (2016) 30–64, https://doi.org/10.1016/j.jii.2016.04.004. Jun.
[2] L.D. Xu, Industrial information integration – an emerging subject in industrialization and informatization process, Journal of Industrial Information Integration 17 (2020), 100128, https://doi.org/10.1016/j.jii.2020.100128. Mar.
[3] A. Maria, "Introduction to Modeling and Simulation," in Proceedings of the 29th Conference on Winter Simulation, Washington, DC, USA, 1997, pp. 7–13. doi: 10.1145/268437.268440.
[4] M. Schumann, M. Schenk, E. Bluemel, Numerically controlled virtual models for commissioning, testing and training. Virtual Reality & Augmented Reality in Industry, Springer, Berlin, Heidelberg, 2011, pp. 163–170, https://doi.org/10.1007/978-3-642-17376-9_10.
[5] E. Miller, "Marrying product and process design," American Machinist, Oct. 10, 2005. https://www.americanmachinist.com/machining-cutting/article/21894874/marrying-product-and-process-design (accessed Oct. 28, 2020).
[6] C. Robinson, Real World Research. Real World Research, 3 ed., Wiley, 2011.
[7] J. Oscarsson, M.A. Jeusfeld, A. Jenefeldt. Towards virtual confidence - extended product lifecycle management," in Product Lifecycle Management in the Era of

Internet of T, Springer International Publishing, 2015, pp. 708–717, https://doi.org/10.1007/978-3-319-33111-9_64.
[8] K. Deb, K. Sindhya, J. Hakanen, K. Sindhya, J. Hakanen, Multi-objective optimization. Decision Sciences: Theory and Practice, CRC Press, 2016, pp. 145–184, https://doi.org/10.1201/9781315183176-12.
[9] A.H.C. Ng, S. Bandaru, M. Frantzén, Innovative design and analysis of production systems by multi-objective optimization and data mining, Procedia CIRP 50 (2016) 665–671, https://doi.org/10.1016/j.procir.2016.04.159. Jan.
[10] S. Bandaru, A.H.C. Ng, K. Deb, Data mining methods for knowledge discovery in multi-objective optimization: part A - survey, Expert Systems with Applications 70 (2017) 139–159, https://doi.org/10.1016/j.eswa.2016.10.015. Mar.
[11] P. Groth and L. Moreau, "PROV-Overview. An Overview of the PROV Family of Documents," World Wide Web Consortium, Monograph, Apr. 2013. Accessed: Apr. 14, 2020. [Online]. Available: http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/.
[12] Y.L. Simmhan, B. Plale, D. Gannon, A survey of data provenance in e-science, SIGMOD Rec 34 (3) (2005) 31–36, https://doi.org/10.1145/1084805.1084812. Sep.
[13] J.A. Zachman, A framework for information systems architecture, IBM Systems Journal 26 (3) (1987) 276–292, https://doi.org/10.1147/sj.263.0276.
[14] S. Ram, J. Liu, Understanding the semantics of data provenance to support active conceptual modeling, in: P.P. Chen, L.Y. Wong (Eds.), Active Conceptual Modeling of Learning, Springer, Berlin Heidelberg, 2006, pp. 17–29, https://doi.org/10.1007/978-3-540-77503-4_3.
[15] A. Iriondo, J. Oscarsson, and M. A. Jeusfeld, "Simulation Data Management in a Product Lifecycle Management Context," 2017, pp. 476–481. Accessed: Nov. 24, 2017. [Online]. Available: http://his.diva-portal.org/smash/record.jsf?pid=diva2:1158759.
[16] T. Gruber, Ontology, in: L. LIU, M.T. ÖZSU (Eds.), Encyclopedia of Database Systems, Springer US, Boston, MA, 2009, pp. 1963–1965, https://doi.org/10.1007/978-0-387-39940-9_1318.
[17] I. Crnkovic, U. Asklund, A.P. Dahlqvist, Implementing and Integrating Product Data Management and Software Configuration Management, Artech Print on Demand, Boston, 2003.
[18] W. Liu, Y. Zeng, M. Maletz, and D. Brisson, "Product Lifecycle Management: A Survey," in Volume 2: 29th Computers and Information in Engineering Conference, Parts A and B, San Diego, California, USA, 2009, pp. 1213–1225. doi: 10.1115/DETC2009-86983.
[19] A. Saaksvuori, A. Immonen, Product Lifecycle Management, Springer Science & Business Media, 2005.
[20] CIMdata, "Product Lifecycle Management 'Empowering the Future of Business,'" CIMdata, Inc, Michigan, Oct. 2002.
[21] M. Grieves, Product Lifecycle Management: Driving the Next Generation of Lean Thinking. Product Lifecycle Management: Driving the Next Generation of Lean Thinking, 1st ed., McGraw-Hill Education, New York, 2005.
[22] John Stark, Product Lifecycle Management: 21st Century Paradigm for Product Realisation, Springer-Verlag, London, 2005. Accessed: May 10, 2016. [Online] Available: http://link.springer.com/10.1007/b138157.
[23] Stackpole, Beth, "CIO magazine," IDG, p. 94, May 15, 2003.
[24] E. Subrahmanian, S. Rachuri, A. Bouras, S. J. Fenves, S. Foufou, and R. D. Sriram, "The role of standards in product lifecycle management support," National Institute of Standards and Technology, Gaithersburg, MD, NIST IR 7289, 2006. doi: 10.6028/NIST.IR.7289.
[25] M. Burkett, K. O'Marah, and L. Carrillo, "CAD Versus ERP Versus PDM: How Best To Anchor a PLM Strategy?," 2003. Accessed: Feb. 14, 2019. [Online]. Available: https://www.gartner.com/doc/1343330/cad-versus-erp-versus-pdm.
[26] Siemens, "PLM Software /Product Lifecycle Management Software," Siemens PLM Software, 2016. https://www.plm.automation.siemens.com/global/en/our-story/glossary/product-lifecycle-management-plm-software/12506 (accessed Feb. 14, 2019).
[27] and Y. Tae-hyuck, K. Gun-yeon, N. Sang-do, PPR INFORMATION MANAGEMENTS FOR AUTOMOTIVE DIE SHOP, Korean Journal of Computational Design and Engineering 12 (6) (2007) 14.
[28] P. Martin, A. D'Acunto, Design of a production system: an application of integration product-process, International Journal of Computer Integrated Manufacturing 16 (7–8) (2003) 509–516, https://doi.org/10.1080/0951192031000115831. Jan.
[29] S. Rachuri, E. Subrahmanian, A. Bouras, S.J. Fenves, S. Foufou, R.D. Sriram, Information sharing and exchange in the context of product lifecycle management: role of standards, Computer-Aided Design 40 (7) (2008) 789–800, https://doi.org/10.1016/j.cad.2007.06.012. Jul.
[30] T. Paviot, V. Cheutet, S. Lamouri, A PLCS framework for PDM/ERP interoperability, International Journal of Product Lifecycle Management 5 (2/3/4) (2011) 295, https://doi.org/10.1504/IJPLM.2011.043182.
[31] The ASD Strategic Standardisation Group (SSG), "White Paper - ISO 10303 (STEP) AP 239 edition 3 Application Protocol For Product Life Cycle Support (PLCS)." Oct. 23, 2015. [Online]. Available: http://www.asd-ssg.org/c/document_library/get_file?uuid=3e01b539-7d77-48dc-a7a5-076511664054&groupId=11317.
[32] S. Rachuri et al., "Information Models for Product Representation: Core and Assembly Models," NIST Interagency/Internal Report (NISTIR) - 7173, Dec. 2004, doi: 10.6028/NIST.IR.7173.
[33] R. Sudarsan, S.J. Fenves, R.D. Sriram, F. Wang, A product information modeling framework for product lifecycle management, Computer-Aided Design 37 (13) (2005) 1399–1411, https://doi.org/10.1016/j.cad.2005.02.010. Nov.
[34] A. von Euler-Chelpin, Information Modelling for the Manufacturing System Life Cycle. Industriell produktion, Production Engineering, Kungliga Tekniska högskolan, 2008.
[35] Instrument Society of America, "ANSI/ISA-95.00.01 Enterprise-Control System Integration - Part 1: Models and Terminology." 2000. Accessed: Mar. 26, 2018. [Online]. Available: https://www.isa.org/store/products/product-detail/?productId=116636.
[36] J.Y. Lee, H.S. Kang, G.Y. Kim, S.D. Noh, Concurrent material flow analysis by P3R-driven modeling and simulation in PLM, Computers in Industry 63 (5) (2012) 513–527, https://doi.org/10.1016/j.compind.2012.02.004. Jun.
[37] M. Colledani, W. Terkaj, T. Tolio, Product-process-system information formalization. Design of Flexible Production Systems, Springer, Berlin, Heidelberg, 2009, pp. 63–86, https://doi.org/10.1007/978-3-540-85414-2_4.
[38] M. Colledani, W. Terkaj, T. Tolio, M. Tomasella, Development of a conceptual reference framework to manage manufacturing knowledge related to products, processes and production systems. Methods and Tools for Effective Knowledge Life-Cycle-Management, Springer, Berlin, Heidelberg, 2008, pp. 259–284, https://doi.org/10.1007/978-3-540-78431-9_15.
[39] vol. 165 A. Smirnov, K. Sandkuhl, N. Shilov, A. Kashevnik, 'Product-process-machine' system modeling: approach and industrial case studies, in: J. Grabis, M. Kirikova, J. Zdravkovic, J. Stirna (Eds.), The Practice of Enterprise Modeling, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 251–265, https://doi.org/10.1007/978-3-642-41641-5_18. vol. 165.
[40] J.Y. Lee, G.Y. Kim, S.D. Noh, Integration framework and PPR+H Hub for DiFac, in: L. Canetta, C. Redaelli, M. Flores (Eds.), Digital Factory for Human-oriented Production Systems, Springer, London, 2011, pp. 59–72, https://doi.org/10.1007/978-1-84996-172-1_4.
[41] W.-B. Zhao, J.-W. Jeong, S.D. Noh, J.T. Yee, Energy simulation framework integrated with green manufacturing-enabled PLM information model, International Journal of Precision Engineering and Manufacturing-Green Technology 2 (3) (2015) 217–224, https://doi.org/10.1007/s40684-015-0025-8. Jul.
[42] vol. 388 W.-B. Zhao, Y.H. Park, H.Y. Lee, C.M. Jun, S.Do Noh, Design and implementation of a PLM system for sustainable manufacturing, in: L. Rivest, A. Bouras, B. Louhichi (Eds.), Product Lifecycle Management. Towards Knowledge-Rich Enterprises, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 202–212, https://doi.org/10.1007/978-3-642-35758-9_18. vol. 388.
[43] Simulation Interoperability Standards Organization, "Standard for Core Manufacturing Simulation Data SISO-STD-008-01-2012," 2012. https://www.sisostds.org/ProductsPublications/Standards/SISOStandards.aspx (accessed Mar. 27, 2018).
[44] P. Barlas, C. Heavey, Automation of input data to discrete event simulation for manufacturing: a review, Int. J. Model. Simul. Sci. Comput. 07 (01) (2016), 1630001, https://doi.org/10.1142/S1793962316300016. Mar.
[45] P. Barlas and C. Heavey, "KE tool: an open source software for automated input data in Discrete Event Simulation projects," in Proceedings of the Winter Simulation Conference (WSC), Dec. 2016, pp. 472–483. doi: 10.1109/WSC.2016.7822113.
[46] Y.-T.T. Lee, F.H. Riddick, B.J.I. Johansson, Core manufacturing simulation data – a manufacturing simulation integration standard: overview and case studies, International Journal of Computer Integrated Manufacturing 24 (8) (2011) 689–709, https://doi.org/10.1080/0951192X.2011.574154. Aug.
[47] P.K. Damarapurapu, S. Gargatte, Automation of Input Data Management for Discrete Event Simulation. CHALMERS UNIVERSITY OF TECHNOLOGY, Department of Product and Production Development, 2016.
[48] A. Skoogh, B. Johansson, J. Stahre, Automated input data management: evaluation of a concept for reduced time consumption in discrete event simulation, SIMULATION 88 (11) (2012) 1279–1293, https://doi.org/10.1177/0037549712443404. Nov.
[49] S. J. Fenves, "A Core Product Model for Representing Design Information," NIST Interagency/Internal Report (NISTIR) - 6736, Oct. 2002, doi: 10.6028/NIST.IR.6736.
[50] S.J. Fenves, S. Foufou, C. Bock, R.D. Sriram, CPM2: a core model for product data, Journal of Computing and Information Science in Engineering 8 (1) (2008), https://doi.org/10.1115/1.2830842. Feb.
[51] Z. Sumei, Total object unified model driven architecture of product lifecycle management, International Journal of Product Lifecycle Management 5 (2/3/4) (2011) 242, https://doi.org/10.1504/IJPLM.2011.043190.
[52] G.Y. Kim, S.D. Noh, Y.H. Rim, J.H. Mun, XML-based concurrent and integrated ergonomic analysis in PLM, Int. J. Adv. Manuf. Technol. 39 (9) (2008) 1045–1060, https://doi.org/10.1007/s00170-008-1400-1. Nov.
[53] S.C. Feng, E.Y. Song, A manufacturing process information model for design and process planning integration, Journal of Manufacturing Systems 22 (1) (2003) 1–15, https://doi.org/10.1016/S0278-6125(03)90001-X.
[54] 62264-1, Enterprise-control system integration, part 1: models and terminology." Industrial Process Measurement and Control Sectional, Committee (2003).
[55] G. Qiao and C. McLean, "Manufacturing information integration in product lifecycle management (PLM)," pp. 935–941, Jan. 2004, doi: 10.1115/DETC2004-57745.
[56] P. Falkman, J. Nielsen, B. Lennartson, A. von Euler-Chelpin, Generation of STEP AP214 models from discrete event systems for process planning and control, IEEE Transactions on Automation Science and Engineering 5 (1) (2008) 113–126, https://doi.org/10.1109/TASE.2007.909632. Jan.
[57] International Organization for Standardization, "ISO 10303-1:1994 - Industrial automation systems and integration – product data representation and exchange –

part 1: overview and fundamental principles," 1994. https://www.iso.org/standard/20579.html (accessed Mar. 26, 2018).
[58] C.R. McLean, Y.-T.T. Lee, G. Shao, F.H. Riddick, Shop Data Model and Interface Specification, NIST Interagency/Internal Report (NISTIR) - 7198 (2005), https://doi.org/10.6028/NIST.IR.7198. Jan.
[59] K.M. Wiig, Knowledge Management Foundations: Thinking About Thinking - How People and Organizations Represent, Create, and Use Knowledge, Schema Press, Limited, 1994.
[60] A. Jashapara, Knowledge Management: An Integrated Approach, Financial Times/Prentice Hall, 2011.
[61] L. Jessup, J. Valacich, Information Systems Today: Managing in the Digital World, 3 edition, Prentice Hall, Upper Saddle River, N.J, 2007.
[62] F.W. Nickols, The Knowledge in Knowledge Management (KM), in: J.W. Cortada, J.A. Woods (Eds.), The knowledge management yearbook 2000-2001, Butterworth-Heinemann, Boston, MA, 2000, pp. 12–21. Accessed: Apr. 04, 2016. [Online]Available: http://www.nickols.us/Knowledge_in_KM.htm.
[63] I. Nonaka, "The Knowledge-Creating Company," Harvard Business Review, Jul. 01, 2007. https://hbr.org/2007/07/the-knowledge-creating-company (accessed Mar. 24, 2016).
[64] K. Grant, Tacit Knowledge Revisited - We Can Still Learn from Polanyi, The Electronic Journal of Knowledge Management 5 (2) (2007).
[65] M. Polanyi, Personal Knowledge, Routledge, 2012.
[66] D. Baxter, R. Roy, A. Doultsinou, J. Gao, M. Kalta, A knowledge management framework to support product-service systems design, International Journal of Computer Integrated Manufacturing 22 (12) (2009) 1073–1088, https://doi.org/10.1080/09511920903207464. Dec.
[67] T.R. Gruber, Toward principles for the design of ontologies used for knowledge sharing? International Journal of Human-Computer Studies 43 (5) (1995) 907–928, https://doi.org/10.1006/ijhc.1995.1081. Nov.
[68] H. Zhang, U. Roy, A semantics-based dispatching rule selection approach for job shop scheduling, Journal of Intelligent Manufacturing 30 (7) (2019) 2759–2779.
[69] M. Uschold, R. Jasper, and P. Clark, "Three Approaches for Knowledge Sharing: A Comparative Analysis," 1999.
[70] L.F. Lin, W.Y. Zhang, Y.C. Lou, C.Y. Chu, M. Cai, Developing manufacturing ontologies for knowledge reuse in distributed manufacturing environment, International Journal of Production Research 49 (2) (2011) 343–359, https://doi.org/10.1080/00207540903349021. Jan.
[71] Y.G. Yao, L.F. Lin, F. Wang, W.Y. Zhang, Multi-perspective modeling: managing heterogeneous manufacturing knowledge based on ontologies and topic maps, International Journal of Production Research 51 (11) (2013) 3252–3269, https://doi.org/10.1080/00207543.2012.756152. Jun.
[72] Z. Usman, R.I.M. Young, N. Chungoora, C. Palmer, K. Case, J.A. Harding, Towards a formal manufacturing reference ontology, International Journal of Production Research 51 (22) (2013) 6553–6572, https://doi.org/10.1080/00207543.2013.801570. Nov.
[73] M. Imran, R.I.M. Young, Reference ontologies for interoperability across multiple assembly systems, International Journal of Production Research 54 (18) (2016) 5381–5403, https://doi.org/10.1080/00207543.2015.1087654. Sep.
[74] C. Palmer, E. N. Urwin, R. I. M. Young, and E. Marilungo, "A reference ontology approach to support global product-service production," Jan. 2017, Accessed: Aug. 20, 2020. [Online]. Available: https://repository.lboro.ac.uk/articles/journal_contribution/A_reference_ontology_approach_to_support_global_product-service_production/9574130.
[75] I.O. Sanya, E.M. Shehab, An ontology framework for developing platform-independent knowledge-based engineering systems in the aerospace industry, International Journal of Production Research 52 (20) (2014) 6192–6215, https://doi.org/10.1080/00207543.2014.919422. Oct.
[76] Z.Y. Wu, X.G. Ming, L.N. He, M. Li, X.Z. Li, Knowledge integration and sharing for complex product development, International Journal of Production Research 52 (21) (2014) 6296–6313, https://doi.org/10.1080/00207543.2014.923121. Nov.
[77] S.E. Kadiri, D. Kiritsis, Ontologies in the context of product lifecycle management: state of the art literature review, International Journal of Production Research 53 (18) (2015) 5657–5668, https://doi.org/10.1080/00207543.2015.1052155. Sep.
[78] L. Petnga, M. Austin, An ontological framework for knowledge modeling and decision support in cyber-physical systems, Advanced Engineering Informatics 30 (1) (2016) 77–94, https://doi.org/10.1016/j.aei.2015.12.003. Jan.
[79] A.L. Fraga, M. Vegetti, H.P. Leone, Ontology-based solutions for interoperability among product lifecycle management systems: a systematic literature review, Journal of Industrial Information Integration 20 (2020), 100176, https://doi.org/10.1016/j.jii.2020.100176. Dec.
[80] N. Chungoora, et al., Extending product lifecycle management for manufacturing knowledge sharing, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 226 (12) (2012) 2047–2063, https://doi.org/10.1177/0954405412461741. Dec.
[81] EFFRA Innovation Portal, "MSEE Manufacturing SErvice Ecosystem," 2014. https://portal.effra.eu/project/982 (accessed Oct. 12, 2021).
[82] N. Hastilow, "Manufacturing systems interoperability in dynamic change environments," thesis, Loughborough University, 2013. Accessed: Oct. 12, 2021. [Online]. Available: https://repository.lboro.ac.uk/articles/thesis/Manufacturing_systems_interoperability_in_dynamic_change_environments/9517406/1.
[83] A. Talhi, V. Fortineau, J.-C. Huet, S. Lamouri, Ontology for cloud manufacturing based product lifecycle management, J. Intell. Manuf. 30 (5) (2017) 2171–2192, https://doi.org/10.1007/s10845-017-1376-5.
[84] K. Sandkuhl, A. Smirnov, Context-oriented knowledge management in production networks, Applied Computer Systems 23 (2) (2018) 81–89, https://doi.org/10.2478/acss-2018-0011. Dec.
[85] E. Negri, L. Fumagalli, M. Garetti, L. Tanca, Requirements and languages for the semantic representation of manufacturing systems, Comput. Ind. 81 (C) (2016) 55–66, https://doi.org/10.1016/j.compind.2015.10.009. Sep.
[86] S. Lemaignan, A. Siadat, J.-Y. Dantan, A. Semenenko, MASON: a proposal for an ontology of manufacturing domain, Intelligent Systems: Collective Intelligence and Its Applications (DIS'06) (Jun. 2006) 195–200, https://doi.org/10.1109/DIS.2006.48.
[87] A. Matsokis, D. Kiritsis, An ontology-based approach for product lifecycle management, Computers in Industry 61 (8) (2010) 787–797, https://doi.org/10.1016/j.compind.2010.05.007. Oct.
[88] M. Bennett, K. Baclawski, The role of ontologies in linked data, big data and semantic web applications, Applied Ontology 12 (3–4) (2017) 189–194, https://doi.org/10.3233/AO-170185. Jan.
[89] A.W. Jones, C.R. McLean, A proposed hierarchical control model for automated manufacturing systems, Journal of Manufacturing Systems 5 (1) (1986) 15–25. Jan.
[90] MITRE Corporation, "Generalised Enterprise Reference Architecture and Methodology," p. 31, 1999.
[91] M. Eigner, R. Stelzer, PLM als Backbone der virtuellen Produktentstehung, in: M. Eigner, R. Stelzer (Eds.), Product Lifecycle Management: Ein Leitfaden für Product Development und Life Cycle Management, Springer, Berlin, Heidelberg, 2009, pp. 47–63, https://doi.org/10.1007/b93672_4.
[92] S. Jeschke, C. Brecher, T. Meisen, D. Özdemir, T. Eschert, Industrial Internet of Things and cyber manufacturing systems, in: S. Jeschke, C. Brecher, H. Song, D. B. Rawat (Eds.), Industrial Internet of Things: Cybermanufacturing Systems, Springer International Publishing, Cham, 2017, pp. 3–19, https://doi.org/10.1007/978-3-319-42559-7_1.
[93] H. Kropsu-Vehkapera, H. Haapasalo, J. Harkonen, R. Silvola, Product data management practices in high-tech companies, Industrial Management & Data Systems 109 (6) (2009) 758–774, https://doi.org/10.1108/02635570910968027. Jan.
[94] J. Madrid, J. Vallhagen, R. Söderberg, K. Wärmefjord, Enabling reuse of inspection data to support robust design: a case in the aerospace industry, Procedia CIRP 43 (2016) 41–46, https://doi.org/10.1016/j.procir.2016.02.137. Jan.
[95] I. Morshedzadeh, J. Oscarsson, A. Ng, M. Jeusfeld, J. Sillanpaa, Product lifecycle management with provenance management and virtual models: an industrial use-case study, Procedia CIRP 72 (2018) 1190–1195, https://doi.org/10.1016/j.procir.2018.03.157. Jan.
[96] V. Dimitrova, R. Denaux, G. Hart, C. Dolbear, I. Holt, and A. G. Cohn, "Involving domain experts in authoring OWL ontologies," in The Semantic Web - ISWC 2008, Berlin, Heidelberg, 2008, pp. 1–16. doi: 10.1007/978-3-540-88564-1_1.
[97] A. Westerinen, R. Tauber, Ontology development by domain experts (without using the 'O' word), Applied Ontology 12 (3–4) (2017) 299–311, https://doi.org/10.3233/AO-170183. Jan.
[98] R. Stevens, P. Lord, J. Malone, N. Matentzoglu, Measuring expert performance at manually classifying domain entities under upper ontology classes, J. Web Semant. (2019), https://doi.org/10.1016/J.WEBSEM.2018.08.004.
[99] L. Pehrsson, A.H.C. Ng, J. Bernedixen, Automatic identification of constraints and improvement actions in production systems using multi-objective optimization and post-optimality analysis, Journal of Manufacturing Systems 39 (2016) 24–37, https://doi.org/10.1016/j.jmsy.2016.02.001. Apr.
[100] K. Amouzgar, S. Bandaru, T. Andersson, and A. H. C. Ng, "Metamodel-based multi-objective optimization of a turning process by using finite element simulation," Engineering Optimization, vol. 0, no. 0, pp. 1–18, Jul. 2019, doi: 10.1080/0305215X.2019.1639050.
[101] K. Amouzgar, "Metamodel Based Multi-Objective Optimization with Finite-Element Applications," 2018, Accessed: Oct. 18, 2018. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-15145.