Journal of Manufacturing Systems 47 (2018) 128–140

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Resource virtualization: A core technology for developing cyber-physical production systems

Yuqian Lu*, Xun Xu

Department of Mechanical Engineering, The University of Auckland, 1010, New Zealand

ARTICLE INFO

Keywords:
Smart factory
Cyber-physical production system
Resource virtualization
Digital twin
Ontology
Semantic web

ABSTRACT

Smart factory in the context of Industry 4.0 is the next wave of smart manufacturing solution to empower companies to rapidly configure manufacturing facilities and processes to enable the fast production of individualized products at change scales. A key enabling technology for developing a smart factory is resource virtualization or creation of digital twins. The presented research fills the gap that the industry needs a practical methodology to enable themselves to easily virtualize their manufacturing assets for developing a smart factory solution. A test-driven resource virtualization framework is proposed as the recommendation for the industry to adopt to create digital twins for a smart factory. The proposed framework draws inspiration from past resource virtualization outcomes with special attention paid to the usability of the proposed framework in a business environment. It provides a straightforward process for companies to create digital twins by specifying the digital twin hierarchy, the information to be modeled, and the modeling method. To validate the proposed framework, a case study was undertaken at an international company, to create digital twins for all their manufacturing resources. The testing result showed that the proposed resource virtualization framework and developed tools are easy to use in a practical business environment to virtualize complex factory setups in the cyberspace.

1. Introduction

Today's ever-connected and decentralized business environment requires companies to be capable of quickly responding to evolving market demands to stay competitive in the market. The ability to rapidly configure manufacturing facilities and processes to enable the fast production of individualized products at change scales has become a significant focus for many manufacturing companies. This trend has been recognized at a global scale with the launch of several government initiatives aiming at delivering smart manufacturing reference models for companies to adopt to build up this capability. Industry 4.0 proposed by Germany, which is probably the most influential initiative, aims at creating smart factories where smart manufacturing systems communicate with each other and rapidly configure themselves for on-demand production [1]. A smart factory needs to:

- **Make real-time engineering decisions:** Smart factories allow in-house production processes to be radically optimized to meet personalized production needs in almost real-time conditions. This goal has been recognized by many manufacturing companies with a desire to shift to an agile production environment to support the fast production of products of diverse variants. The configuration of legacy engineering systems to support agile production is a challenge. The quality of the integration between the dominant PDM (Product Data Management), PLM (Product Lifecycle Management) and ERP (Enterprise Resource Planning) systems with a goal to support personalized production is a key component of facilitating fast and accurate engineering decision-making.
- **Monitor all manufacturing resources via industry internet:** Monitoring is an important aspect of smart production. Thanks to the ubiquity of sensors, wireless network, and cloud storage implementing sensor networks to collect key engineering information related to people, machine and processes in a factory is achievable. Analytics on manufacturing data can provide factory management team with insights on the snapshot of each machine, and the capacity of a factory so that data-driven production planning and forecast can be achieved.
- **Understand its own capabilities and self-organize production activities:** A smart factory can understand its capabilities based on gained engineering knowledge via self-learning and therefore organize production activities autonomously. Engineering knowledge learned from manufacturing data requires context and meaning. The generated knowledge graph based on collected manufacturing data will be used for decision-making at a higher level.

* Corresponding author.
E-mail address: ylu633@aucklanduni.ac.nz (Y. Lu).

https://doi.org/10.1016/j.jmsy.2018.05.003
Received 1 February 2018; Received in revised form 15 April 2018; Accepted 3 May 2018
Available online 15 May 2018

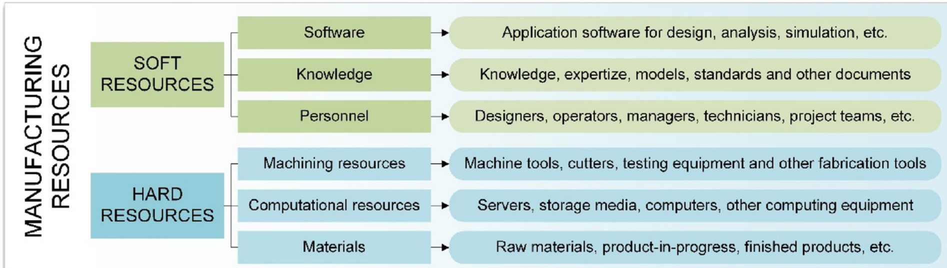

Fig. 1. Classification of manufacturing assets in cyber-physical systems.

To achieve the above vision, there has been a good amount of scientific efforts devoted to developing smart factory models and case studies. Jay Lee et al. proposed a generic high-level cyber-physical system (CPS) architecture towards digital factory [2]. The primary use of CPS in factory floors is to lay the foundation for smart factory management by virtualizing physical assets into the cyberspace and creating resilient, intelligent, and self-aware machines. Five layers are specified in the proposed CPS architecture: 1. smart connection to physical assets via various sensors, 2. conversion of data to information for each connected asset, 3. cyber level as the central information hub where machines are interconnected into a virtual network, 4. knowledge generation of the acquired information to present to end users for making business decisions, and 5. machine self-configuration based on human decisions. The second and third layer in this architecture requires a semantic model that allows a physical asset to be abstracted to a digital twin in the cyberspace. Constructing mirrors of physical manufacturing resources in the cyberspace is a significant underlying technology for developing a cyber-physical system [3]. The research work in this paper focuses on developing a methodology to enable the development of a feasible semantic model that supports easy creation of digital twins for physical assets in a factory so that high-level factory monitoring and planning systems can rely on the generated digital twins.

Digital twin reflects two-way dynamic mapping of physical objects and virtual models [4]. The essence of a digital twin presents a middleware architecture that abstracts the shop-floor hardware for usage at high-level engineering management systems to make real-time decisions. At its technical core, the concept of a virtualized version of the physical manufacturing asset signifies a data model that encapsulates its technical specifications and information relationship with its external environment [3]. Specifically, it is the virtualization of physical entities [5]. A common practice is to develop a semantic model for encapsulating machine specifications and capabilities and relationships between the resources. There has been some research work on developing semantic models for predictive maintenance [6], machine fault diagnosis [7], digital factory [8], prognostics [9] etc. To take these research outcomes to real industry application, there is a need to develop guidelines and frameworks that empower companies to systematically develop semantic models for smart factories in Industry 4.0 environment based on their own factory setups and business needs. The research work presented in this paper proposes a systematic framework for developing smart factory semantic models and virtualizing factory assets using the development model. The framework for developing a feasible virtual smart factory using a proper modeling language with the ability to enable on-demand knowledge-based business decisions is considered as the key contribution of the presented research work. The authors understand that another key aspect of a CPS is that a digital twin is required to always stay in synchronization with the physical entity using advanced sensor technologies. This is not the focus of this research. A sample semantic model for a real factory environment is presented as a case study to show the process of virtualizing a factory using the proposed framework. The rest of the paper is organized as follows. Section 2 reviews the literature in related research areas and highlights the research gap that motivated the presented research work. The framework for virtualizing resources of a factory is presented in Section 3 with detailed discussions on the logic of the resource virtualization process and essential data to be abstracted in the digital counterpart. A case study with a global manufacturing company is presented in Section 4 to validate the proposed framework. Discussions on the testing results and industry feedback are presented in Section 5. Section 6 concludes the research work and highlights future research directions.

2. Literature review and research gaps

The role of a cyber-physical production system in a factory environment is to allow companies to quickly adapt to market changes via flexible configuration of manufacturing resources for the rapid production of one-off personalized products while maintaining required margins. It requires cyber-physical systems to be able to self-organize production activities at the configuration level using machine dynamics information from cyber level [2]. Resource virtualization is a key enabling technology for developing cyber-physical production systems [10]. This section reviews the reported research related to resource virtualization and highlights the research gap.

2.1. Literature review

Cyberspace in cyber-physical systems is an information hub that stores all the digital twins of all physical manufacturing assets. Manufacturing assets consist of diversified and distributed manufacturing resources (equipment, computational resources, materials, software, knowledge, and skills) (see Fig. 1 for further details) [11]. These resources in a factory are the key manufacturing assets to be virtualized in the cyberspace. With regard to manufacturing resource modeling, standards such as STEP-NC were recognized as playing an important role [12]. A novel STEP-NC compliant machine tool data model was developed to enable modeling of machine tools in the cyberspace for process planning and manufacturing [13]. The developed machine tool data model provides adequate information, such as machine geometry, cutter information, process information to support process planning in a smart factory. Similarly, a STEP-NC based data model was proposed to model CNC machine tools and their auxiliary devices. The focus of this developed model is to define the physical components and the kinematic chains of a machine tool in the cyberspace. Wang and Xun also investigated abstracting capabilities details from distributed manufacturing resource and utilizing these information in the cyberspace to drive decision-making in matching manufacturing facilities with production jobs [14]. More importantly, the functionality of a resource at different granularity levels needs to be modeled [15]. Before the virtualization of resources, the granularity levels, the resource categories in each granularity level and the virtual models of each kind of resource need to be defined. Another multi-granularity manufacturing resource model was also proposed to model manufacturing capabilities at different granularity levels [16].

To enhance the semantic interpretation of virtualized

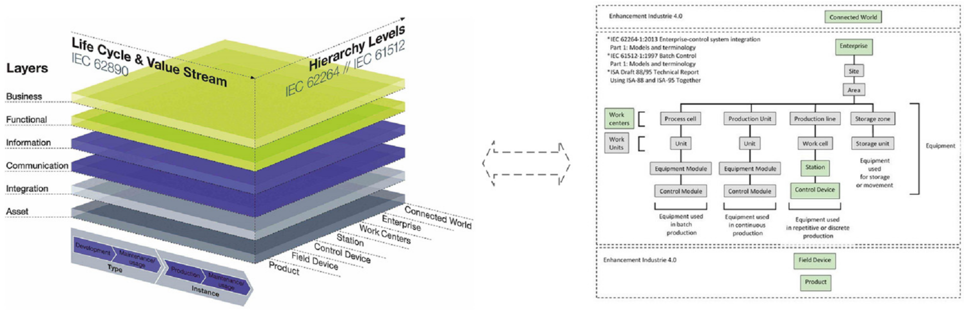

Fig. 2. Hierarchical system architecture in Industry 4.0 (adapted from [22] and [23]).

manufacturing resources in the cyberspace, semantic web languages such as OWL (Ontology Web Language) were used and the capability of querying the semantic model of virtualized manufacturing resources was demonstrated [17]. The possibility of modeling machine tool concepts defined in established industry standards using OWL language was explored in [18]. In this research, the mapping mechanism between ontology model and concepts in existing industry standards was discussed in detail. A more systematic model for describing manufacturing equipment resources was presented in [19]. They described manufacturing resource from two aspects: static functional capability and dynamic production capability, using the ontological method. Functional capabilities are stationary and describe what kind of work a machine can perform, whereas production capability reflects the performance of a machine during a given time. In a more recent study, an ontology was developed to define the semantics of machine components and of all the data that will be communicated from and to physical machines [20]. The developed ontology provides mapping attributes for machine status and operation at device, component, and subcomponent levels.

2.2. Research gaps

The literature has well recognized that physical manufacturing assets need to be virtualized in the cyberspace to enable high-level decision-making in a cyber-physical system. The physical attributes and functional attributes of a manufacturing resource should be specified explicitly using a machine-readable language. Early prototypes demonstrated various technologies to virtualize physical objects including XML-based description and parsing, SQL database storage, Non-SQL methods and semantic web technologies. However, these findings are not exhaustive on the methodologies for resource virtualization in cyber-physical systems. How to implement these methods in a manufacturing company to enable virtualizing the factory environment and building a CPS-powered smart factory is critical to the success of resource virtualization in a real industry environment. Hence, there is a need to create a framework and a set of tools that enable a manufacturing company to easily build the digital twins for a factory. The developed framework needs to explicitly represent all the essential information of all the manufacturing resources in a factory and the developed digital twins can be easily inferred by high-level programming languages systems and will support self-organizing decision making at a later stage. The presented research was motivated to develop such a framework that can meet the above requirements.

3. The proposed resource virtualization framework

This section introduces a resource virtualization framework that is envisioned to enable a company to successfully virtualize all the required manufacturing resources for developing a smart factory solution. Creating the digital counterpart of a factory involves resolving three main problems: (1) Selecting a factory reference model, (2) selecting a resource virtualization process that ensures high-quality digital twins are created and verified, (3) developing a semantic model that contains all the required concepts for explicitly representing a physical resource and creating the digital twins. The rest of this section discusses the details of the proposed framework in these three aspects.

3.1. Hierarchical system architecture of a factory

Creating the digital twin of a factory requires a good understanding of the structure, activities, and processes of the target factory or even the business. This information can be retrieved from enterprise modeling of the structure of the factory and the links between the underlying units [21]. A typical reference model for Industry 4.0 is RAMI 4.0 model that breaks down all elements and IT components in a layer and life cycle model [22]. In this model, the architecture of factories in Industry 4.0 is changed to a multi-level connected network of participants that interact across hierarchy levels [23]. As shown in Fig. 2, Industry 4.0 is a connected world that links groups of factories, external engineering firms, and suppliers. Within a factory, field devices, work units, and work centers are organized systematically for carrying out different types of production.

The hierarchical architecture of a factory can be specified at three abstract level, i.e., enterprise level, factory level, and equipment level. The digital twin at the enterprise level specifies its history and product offerings. The digital twin at the factory level details the specific production capabilities that the factory has, which allows a cyber-physical system to quickly locate the correct factory for a production order without drilling down to the equipment level. Digital twins at the equipment level in the factory are created to abstract its detailed capabilities and physical state.

Digital twins at these three levels form a multi-granularity virtual representation of a factory in the cyberspace. As is seen in many cyber-physical production system applications, resource virtualization focus is placed at the equipment level which makes perfect sense when the goal is to develop smart manufacturing equipment. The business interactions at the factory level and enterprise level are equally important to be virtualized when developing a smart factory solution. Accurate modeling of the business activities at these two levels provides the required data for possible streamlined business execution via integration with business management applications such as PDM and ERP system. The modeling at the enterprise level and factory level is also important for intelligent configuration of production activities and horizontal integration. This is because the efficient production of personalized products relies on close collaboration between companies in the whole product development process. The effective configuration of a suitable production network needs accurate capability description in the cyberspace at the enterprise level and the factory level and of course at the equipment level.

3.2. Components of a digital twin

In the cyberspace, the digital abstraction of a manufacturing asset includes three integral parts: 1. technical specifications, 2. functional

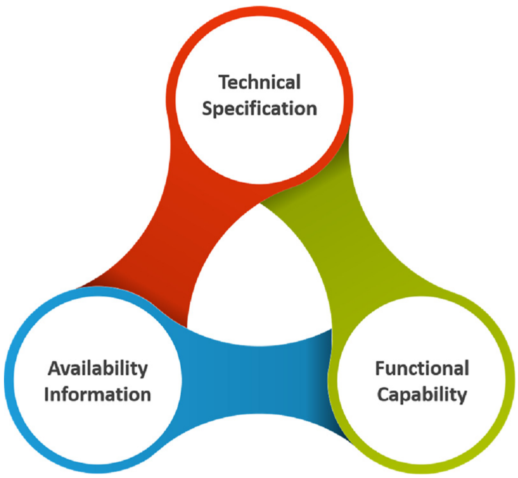

Fig. 3. Components of a virtual manufacturing resource.

capabilities, and 3. availability information (Fig. 3). Technical specifications are a list of attributes that define the technical characteristics of a manufacturing asset. It is often published by the manufacturer of a manufacturing resource to help users understand the general functionalities of the manufacturing resource. Take CNC milling machine as an example. Its technical specifications include manufacturer, max machining envelope, control system, maximum spindle speed, mass, spindle motor power, etc. Technical properties are normally static information which is not likely to change over time after a resource is produced and put into production. These specifications give a CPS a basic profile of a virtualized manufacturing resource in the cyberspace and this information can be used as the criteria for a quick screening when a smart factory configures production activities.

Functional capabilities specify what a manufacturing resource can be used for under what condition and as a result generates what output. Functional capabilities of a manufacturing resource are often shop floor-dependent, though there are general understandings of what a manufacturing resource can do. These different definitions of a manufacturing resource's capability result from differences in business strategies, operator expertise, etc. Functional capabilities can change over time.

Machine availability information reflects the status of a monitored machine, telling its capacity and availability information. Machine availability monitoring is not a problem anymore due to the advances in IoT and network communication technologies. Real-time machine availability and execution status during metal-cutting operations are used as part of a cloud-based process planning system [24]. Recently, Cai et al. developed a method to integrate manufacturing information such as spindle speed, feed rate, and cutter location and sensory data, such as vibration and current to form digital twins of machine tools [25]. These data are analyzed and integrated to enable remote users to monitor machine tool production status via a remote computer.

These three components form the complete digital counterparts of any manufacturing resource. A cyber-physical system depending on its application may need more detailed data abstraction on any of these three components. For instance, cyber-physical machine tool needs comprehensive technical specification and availability status in the cyberspace to capture intuitive and high-fidelity machine snapshot [26], whereas a CPS-based predictive maintenance solution requires detailed functional capability description based on a domain semantic model [6].

3.3. Virtualization methodology

Each cyber-physical system within a factory is designed for a unique business application, in which digital twins are unique per the unique factory environment and business requirements. Therefore, the semantic model that provides the data schema for virtualizing manufacturing resources in a factory environment is unique to the scope of the application to which it is devoted. A bottom-up approach to virtualizing manufacturing resources is required to enable a factory setup can be easily virtualized. The diagram below presents the proposed resource virtualization process that a company can follow to create its digital twins. This resource virtualization process improved existing generic ontology development processes [27,28] by integrating test-driven product development [29] and agile practices [30], both of which are well-accepted software programming practices. The diagram below (Fig. 4) shows the overall process.

Requirement analysis is a process of discovering manufacturing resources to be virtualized and their essential attributes to be abstracted in the cyberspace. This process needs a good understanding of how the target cyber-physical system will be used in the real world and what data will need to be made available to fulfill the application requirements. Once all the application scenarios are identified, all the desired real-world application scenarios using the target cyber-physical system should be converted into testable use cases and these use cases will be used to validate the performance of the virtualized manufacturing

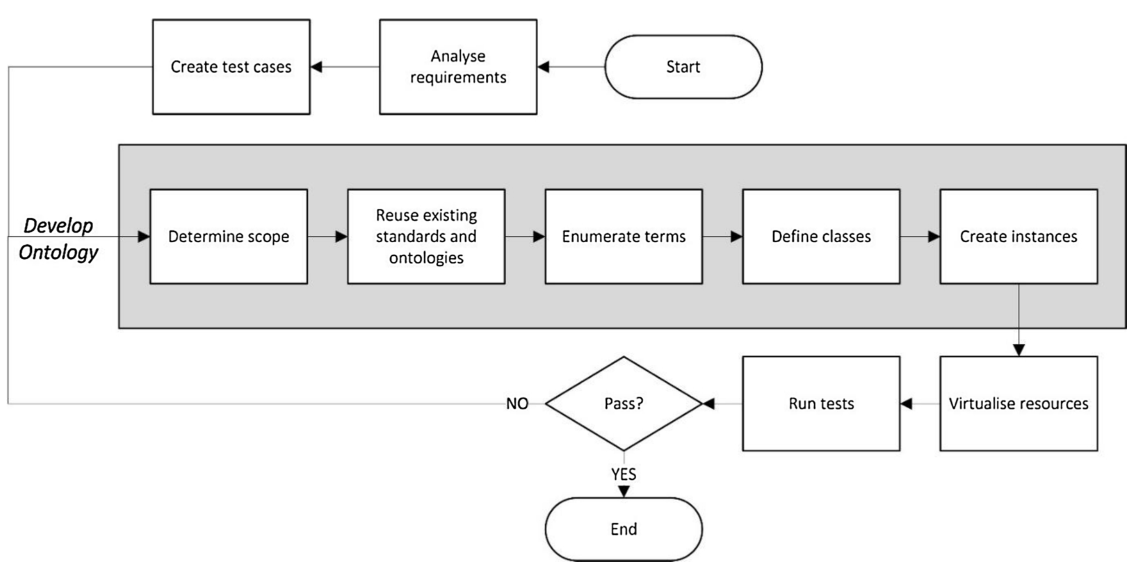

Fig. 4. A test-driven resource virtualization process.

Date: 21/09/2017
Ontology engineer: Michael Lu

Goal of the ontology:
Provide the data model for virtualizing manufacturing resources in the target factory
Scope:
Provide domain concepts for describing the technical properties, functional capabilities and availability status of all the manufacturing resource in the target factory
Knowledge sources:
Manufacturing companies and mechanical engineers (domain experts)
Existing ontologies
Related industrial standards
Knowledge base from machine tool vendors
CAD/CAM/PLM software
Sample test cases:
Test case 1: Able to fetch machine tool status by machine Id
Use Case 2: Able to retrieve the maximum workpiece size by machine Id

Fig. 5. Ontology requirements specification document.

Table 1
STEP-NC parts.

Part Number	Title	Publication date
ISO 14649: 1	Overview and fundamental principles	2003
ISO 14649: 10	General process data	2003
ISO 14649: 11	Process data for milling	2003
ISO 14649: 12	Process data for turning	2005
ISO 14649: 111	Tools for milling	2004
ISO 14649: 121	Tools for turning	2005
ISO 14649: 201	Machine tool data for cutting process	2011

resources later.

Next is to develop the ontology that contains all the required concepts for virtualizing the manufacturing resources. In computer and information science, ontology represents knowledge as a set of concepts within a domain, using a shared vocabulary to denote the types, properties, and interrelationships of those concepts [31]. Ontology model can be described as a set $O = \{C, RS, I\}$, where C is a collection of concepts in the domain also called classes, I is a set of individual instances of classes, RS represents the relations between two concepts or particulars. The key in the ontology development process is to determine the scope and reuse existing international standards [32]. After identifying the important terms, the next step was to ascertain the relationships among these terms. This process involves the creation of classes, object properties and data properties, and organize them in a meaningful hierarchy. The last step is creating individual instances of classes in the hierarchy. Defining an individual instance of a class requires (1) choosing a class, (2) creating an individual instance of that class, and (3) filling in the slot values. This step is optional and will only be required if there is a need to have some constant instances to be referenced or linked by digital twins. Such examples include a specific date time, list of countries, etc.

Up to this point, a trial ontology has been created based on the initial requirements. The next step is to virtualize manufacturing resources using the developed ontology. Based on the discussion in Section 3.1, a digital twin of a physical manufacturing resource consists of three components: technical specifications, functional capabilities, and availability status. In the proposed framework, each manufacturing resource is abstracted as an instance of a defined resource type and its technical specifications are represented as the data properties of the resource instance. Availability status which is the dynamic properties of a resource is also represented as the data properties of a resource instance. These properties can be updated when the status changes. For instance, machine spindle speed can be a data property of a CNC machine instance and its value can be changed as the updated value comes from the connected sensors. Shop floor-dependent functional capabilities can be represented using semantic web rule languages to explicitly model the complex engineering knowledge. Regarding semantic web languages, OWL is a well-recognized common ontology formalization language, which has extensive support for expressing meaning and semantics and has great ability to represent machine interpretable content on the Web. Moreover, semantic web rule languages provide the required expressiveness, enabling machine interpretation, automated processing, and translation into other such semantic web languages, some of which are also the execution syntax of rule engines. They are usually used as additional means to build knowledge-based systems on top of ontology because of its rich expressiveness and good integration with ontology and rule engine. Some of the important semantic web rule languages include RuleML (Rule Markup Language) [33], SWRL [34], and RIF (Rule Interchange Format) [35], as well as platform-specific rule engine languages such as Jena [36]. In the proposed framework, OWL is selected as the ontology language as it provides maximal expressiveness while retaining decision-making. Jena is selected as the rule language for modeling resource capabilities because it has good reasoning infrastructure to interface with RDF and OWL.

After all the manufacturing resources are abstracted as digital twins using the developed ontology, next step is to run the created unit tests and check whether all the required data has been abstracted in the cyberspace. The entire process ends if all the test cases pass unless ontology or digital twin refactoring needs to be carried out to improve the ontology or the related digital twin until all the test cases pass. This test-driven iterative resource virtualization process ensures the final digital twins reflect the nature of the corresponding physical manufacturing resources.

4. Case study

This section presents a case study on using the proposed framework to virtualize manufacturing resources in a cyber-physical system. The case study is part of a knowledge transfer collaboration with a world-leading supplier of sealing solutions. The knowledge transfer collaboration aims at implementing and trailing the R&D outcomes on smart factory in this multi-national enterprise by developing a smart factory solution to automate inter-factory production management between all the subsidiaries via real-time production status-based production configuration. The remainder of this section details the process of virtualizing manufacturing resources for a typical factory in this enterprise with an overview of the created ontology and digital twins and how they are used in the overall smart factory solution.

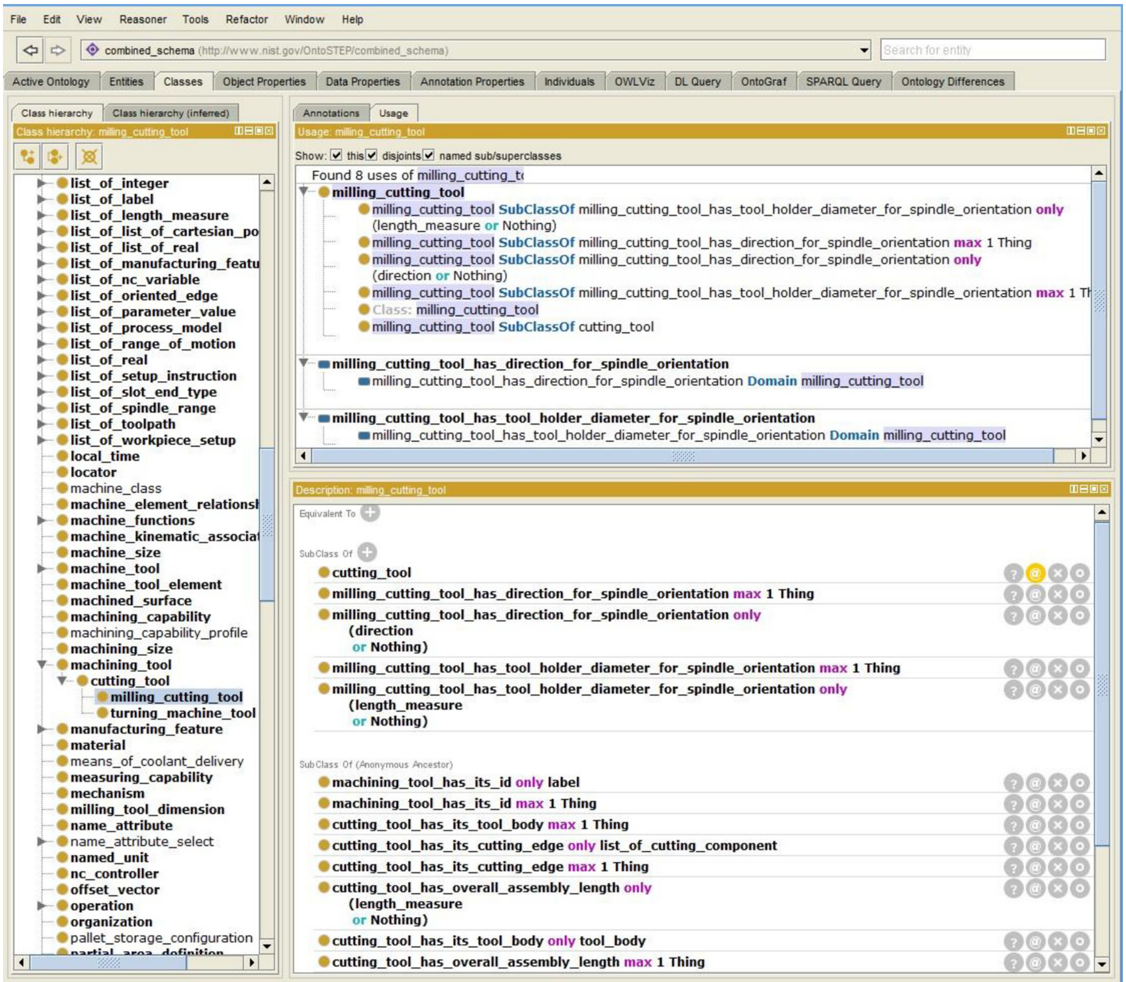

Fig. 6. Screenshot of converted concepts from STEP-NC standards.

4.1. Project background

The case company provides world-leading sealing solutions to almost all industries with factories in German, America, Japan, Singapore, New Zealand etc. They used to supply a wide range of standard mechanical seal systems, but the market has now required them to provide highly custom sealing solutions. To adapt to this market change, their product delivery model has been changed to engineer-to-order; every sealing solution is custom designed and produced. This change has received huge resistance from the shop-floor environment because the production setup of most of their factories is designed for mass production of standard product variants. Currently, production activities are manually planned by a master production planner by checking inwards production orders and factory capacity, which is error-prone and fails to guarantee optimal cost control. At the company level, coordination between factories relies on sales engineers to track the production capability of each workshop via email communication, which was identified as one of the most tedious business processes throughout the project lifecycle. Therefore, the project was originated from the need of developing an inter-factory smart factory system that allows the business to easily check the capabilities of each workshop and their capabilities and ultimately enables the business to deliver optimal on-demand production for product development at changing scale.

In this project, a CPS is proposed as the recommendation for transforming the factory environment into a monitored and connected smart factory, in which manufacturing assets talk to each other in the cyberspace and the system itself constantly optimizes production activities using streamed machine status and capability information. A significant piece of the complete smart factory solution is to create digital twins of all the manufacturing resources in the factory environment.

4.2. Developed ontology

The case study selected a typical factory in this company to validate the effectiveness of the proposed resource virtualization framework. As recommended by the proposed resource virtualization process in Section 3, the first step is requirement analysis. In this project, the goal is to develop a smart factory system that allows the business to understand each factory's capability and real-time project production progress and use this information to automatically schedule new production orders (Fig. 5). Therefore, the main requirements include: being able to (1) represent and store the real-time status of machine tools and cutters, and (2) represent the functional capabilities of each manufacturing resource. The required ontology is mainly for virtualizing manufacturing assets and describing manufacturing capabilities. It mainly deals with information about ownership, technical properties, and functional capabilities. The domain knowledge is sourced from domain experts, reusable ontologies, related industry standards, machine tool vendors, and potential use cases.

After the scope is identified, existing ontologies and industry standards are considered. At the time of this research, there is no officially recommended ontology in the manufacturing domain on the W3C website. In the manufacturing domain, the most notable industry standard is STEP/STEP-NC. The standard consists of several parts, each

Fig. 7. Further integration of machine tools and cutters.

focused on an application domain. Some of the parts are listed in Table 1 and these parts should be incorporated.

In this case, we used a Protégé plugin called OntoSTEP to translate the above standards to ontologies represented in OWL [37]. Hence, we created the proposed ontology by first merging all relevant ISO 14,649 parts, and then converting them to ontology using OntoSTEP. Following this, we examined the integrity of the ontology and manually added additional compulsory classes and individuals in Protégé. To be specific, a combined schema of ISO 14,649—Part 10, Part 11, Part 12, Part 111, Part 121 and Part 201 was created. The verified schema was fed into Protégé and converted into an ontology using OntoSTEP. As can be seen from Fig. 6, EXPRESS entities and instances were mapped respectively to OWL classes and individuals. Attributes correspond to OWL properties ObjectProperties link classes together, while DataProperties link classes to data types. A complete summary of the proposed OWL mapping of the basic concepts of EXPRESS is available at [37].

Next, it is necessary to verify the created ontology against default resource specifications from machine tool vendors and potential application scenarios. The main task is to examine whether the created ontology can readily and truly represent manufacturing resources. New classes and properties are considered where essential information about a resource cannot be handled by the ontology. For instance, the generated ontology lacks a geometrical description of face milling cutters. Therefore, definitions from ISO 13,399 are added to the ontology, for a comprehensive representation of face milling cutters. Moreover, descriptions for cutters from SANDVIK and descriptions for machining centers from OKUMA were integrated into the ontology because cutters used in this company are all sourced from these two suppliers. As can be seen from Fig. 7, definitions for cutters, such as *depth_of_cut_maximum*, *body_diameter,* and *overall_length* from ISO 13,399, were imported into the ontology.

Fig. 8 presents a small subset of the resulting ontology with a capability description for machine tools. In this ontology, engineering practice in classifying a CNC machine tool was incorporated into the ontology. Based on the investigation with major machine tool vendors, such as Mazak, OKUMA, and DMG, it was found that machine tools are mainly classified as milling machines, turning machines, machining centers and multitasking machines. Another attribute is the number of axes that a CNC machine tool can move on simultaneously. At the present stage, there are machine tools that can support 5-axis machining. In the ontology, only 2-axis to 5-axis are defined as machine types. However, to reflect the multi-task capability of a machine tool, a list of machine functions was created, including turning, milling, drilling, grinding, etc. A machine tool can have multiple machine functions associated with it.

A custom virtualization template was created for a unique type of machine tool in the factory. This is to enable the case company to easily create the digital twins for each machine tools in the factory. Take ‘Quick Turn Nexus 100-II MS’ [38] as an example; the final description model is as follows (Fig. 9). This CNC turning center with multitasking capability features milling capability and a second turning spindle to process parts in single operation. For other types of machine tools, a similar description template is created too.

Virtualizing cutters is as important as virtualizing machine tools. The decision-making process for selecting a correct cutter highly depends on the specifications of the target component and available machines. A cutter can only achieve its best performance under

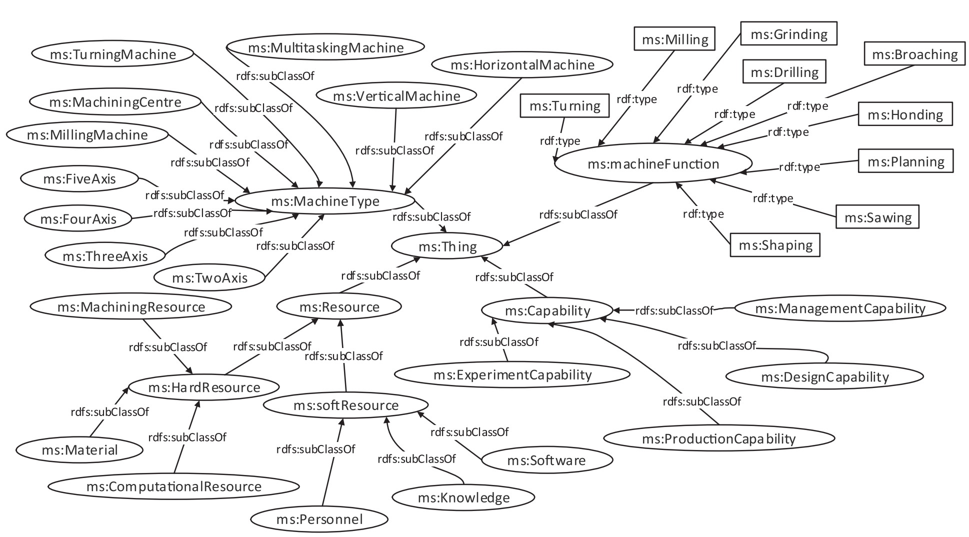

Fig. 8. Top-level capability description for machine tools.

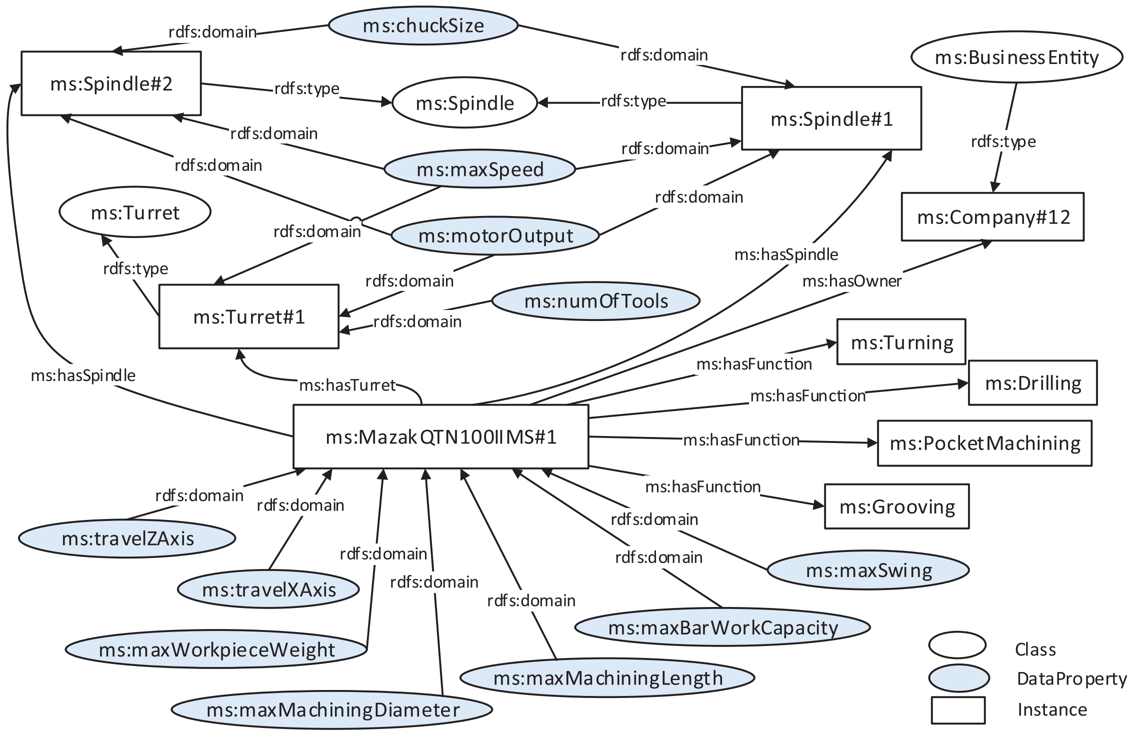

Fig. 9. Ontology model for Mazak Quick Turn Nexus 100-II MS.

Table 2
Grouping of workpiece material.

Material Groups	ISO P		ISO M	ISO K	ISO N	ISO S	ISO H
	Steel	Stainless steel Ferritic & Martensitic	Stainless steel	Cast iron	Nonferrous materials	Heat resistant alloys	Hardened materials

particular conditions. For example, sometimes a cutter can only work with a coolant supply and can only be used for a specific kind of workpiece material. These constraints are important aspects to be considered in the process of constructing the ontology. In this project, the main concern related to cutters is to determine if a cutter can be used to produce a workpiece. Therefore, only component-related constraints need to be included in the description model. For this reason, the main capability description for cutters should focus on the material a cutter can process and the machining process(es) a cutter can be used for.

The metal cutting industry produces a wide variety of components, machined from many different materials. The component material strongly influences the choice of cutting tool geometry, grade, and machining parameters. Therefore, workpiece materials have been divided into six major groups, in accordance with ISO standard [39], and each group has unique properties regarding machinability (Table 2).

Machining is any of various processes in which a piece of raw material is cut into a desired final shape and size by a controlled material-removal process. The differences between the various types arise from the relative motion between a cutting tool and the workpiece and type of cutting tool used. Thus, machining processes are classified as turning, milling, and hole-making processes (Fig. 10). For turning operations, there are dedicated cutters for general turning, parting off, and grooving. General turning can be further classified as external turning and internal turning, based on the spatial pattern between the component and cutter. Turning can be broken down as longitudinal turning, facing or profiling. Grooving requires a different kind of cutter from that used for general turning, although both processes employ a lathe. Based on the characteristics of the target machining feature, a grooving process can be further differentiated as external grooving, internal grooving, and face grooving. Milling includes several highly versatile machining operations taking place in a variety of configurations. It mainly includes plain milling, face milling, end milling, form milling, profile milling, slot milling, chamfering, gear milling, and turn milling. The hole-making process can be classified as general drilling, chamfer drilling, and step drilling, according to the characteristics of the hole. Chamfer drilling produces holes with a chamfer or some deburring; some typical examples are screw and rivet holes. Step drilling is used for producing a stepped or stepped and chamfered hole. Typical applications are components with screws or bolts where the head needs to be hidden. When enlarging or improving the quality of an existing hole, boring is used. Reaming is usually a finishing operation to achieve high-precision holes. Three different threading methods (thread turning, thread milling, and tapping) are used, depending on the component, the machine, and batch size.

A class called *MaterialGroup* was created in the proposed ontology, and six instances of this class were created as built-in individuals in the ontology, namely, ISO_P, ISO_M, ISO_K, ISO_N, ISO_S, and ISO_H (Fig. 11). In the proposed ontology, these machining methods were added as instances of the *MachiningFunction* class. A cutter can be linked to these capability tags when being inserted into the knowledge base. It should be noted that a cutter can be tagged with multiple tags. For example, a cutter may be able to process all the materials under six groups. In addition, a machining process was divided into roughing, semi-finishing, and finishing, according to differences in machining goals and strategies at different machining stages.

4.3. Digital twins of the factory

The next step is to create digital twins of the case factory after the ontology is developed. In this project, we created digital twins for the factory at three hierarchical level: enterprise level, factory level, and

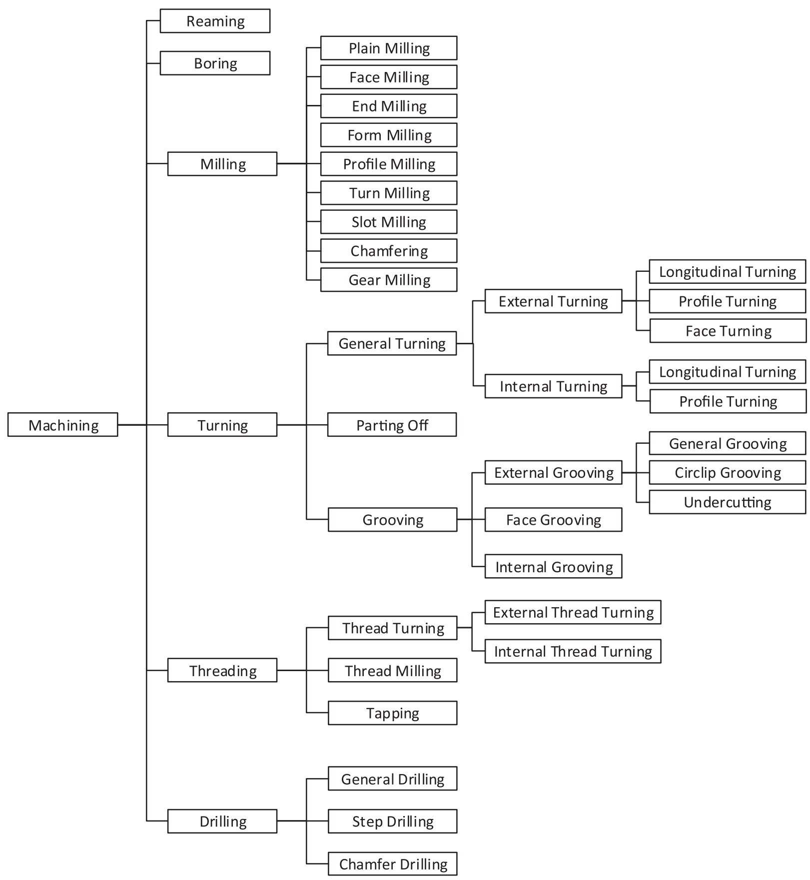

Fig. 10. Classification of machining method for cutters.

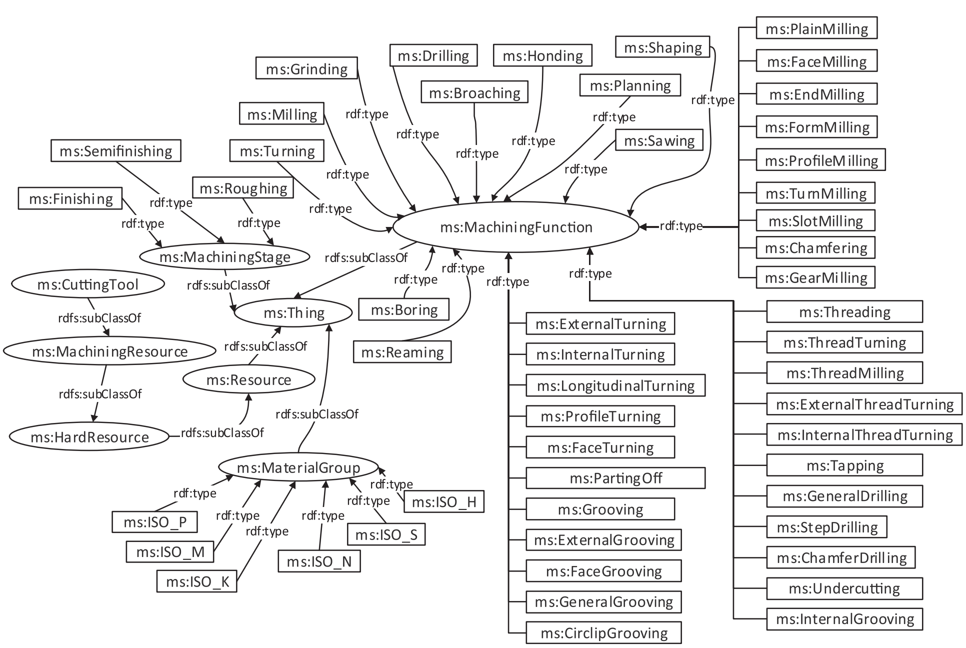

Fig. 11. Top-level capability description for cutters.

equipment level. In the case study, interviews were carried out with senior managers from the company to summarize its vision statement and product offerings. It was concluded with the following company profile:

> *"Company X is one of the leading international suppliers of industrial sealing solutions. Its products are installed wherever safety and reliability are major design considerations, for example in the oil & gas, refinery, chemical, energy, food processing, paper, water, marine, aerospace, and mining industries. A workforce of more than* 5200 *creative and highly motivated employees develops tailored solutions for its customers."*

The translated semantic description is as follows (Fig. 12): this description was input into the database as the digital twin for the

```
<owl:NamedIndividual rdf:about="&ms;CompanyX">
  <rdf:type rdf:resource="&ms;BusinessEntity"/>
  <hasProductionModel rdf:resource="&ms;Engineer_to_order"/>
  <hasProduct rdf:resource="&ms;Sealing_solution"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Engineer_to_order">
  <rdf:type rdf:resource="&ms;ProductionModel"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Food_processing">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Marine">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Mining">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Oil_gas">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Paper">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Refinery">
  <rdf:type rdf:resource="&ms;Industry"/>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;Sealing_solution">
  <rdf:type rdf:resource="&ms;Product"/>
  <productAppliedIn rdf:resource="&ms;Food_processing"/>
  <productAppliedIn rdf:resource="&ms;Marine"/>
  <productAppliedIn rdf:resource="&ms;Mining"/>
  <productAppliedIn rdf:resource="&ms;Oil_gas"/>
  <productAppliedIn rdf:resource="&ms;Paper"/>
  <productAppliedIn rdf:resource="&ms;Refinery"/>
</owl:NamedIndividual>
```

Fig. 12. Digital twin for the case company.

company in the cyberspace.

The factory involved in this case study is the company's development center in New Zealand. This workshop is capable of fabricating and assembling most sealing parts, except that the grinding process must be subcontracted to another factory in Australia because of resource limitations. The resultant digital twin is as follows (Fig. 13):

At the equipment level, the case factory owns several multitasking CNC machines and cutters. The engineers in the factory have years of CNC machining experience, which gives the factory the ability to process some very complex machining requirements with unique process know-how. Thus, additional capability description was added to the equipment to model the functional capabilities.

To enable the case factory to easily create digital twins for their equipment, a 'template-per-resource' approach is adopted; each predominant kind of manufacturing resource is given a custom template with all the required data fields and common attributes prepopulated. A web-based system (Fig. 14) is developed to allow the case company to select the correct template and type in the unique data attributes for a resource to create its digital twin. In addition, functional capabilities are inserted via the same web interface. On this web page, the case company fills in a very simple web form which requires the following fields:

- Name – the name of the new resource. This name should be unique within the company and the resultant resource is in the form of '*Resource Name_Company Name*'. This is to ensure the name of a resource is unique in the database, as it is very important for production configuration at a later stage.
- Description – detailed description of the inserted resource. This can include its location, characteristics, etc.
- Item Code – this provides a way of identifying specific items and categorizing related inventory resources. It allows fast resource identification and tracking.
- Category – this dropdown list includes many predominant resource types in the case company. When a category is selected, the related digital twin template is loaded at runtime. For example, if 'MazakQTN100IIMS' is selected, the template in Fig. 15 will be retrieved from the database and pre-processed.
- Status – this predefined dropdown list contains frequently-used keywords for representing a resource's status. A resource's status is used to determine its capacity.
- Optional Capability Description – this text area is for inputting functional capabilities in Jena rule language.

Taking 'Mazak Quick Turn Nexus 100-II MS' as an example, the semantic template for this kind of machine tool is as follows, with all the basic capability information specified (Fig. 15). When a new instance of this type of machine is virtualized to the cyberspace, only the name of the machine, spindles, turret, and machine owner will be updated. Additional functional capabilities in Jena rules are directly stored in the backend database and associated with the resource instance.

The screenshot below (Fig. 16) shows the web interface for creating the digital twin of a Mazak Quick Turn Nexus 100-II MS machine tool. The case factory uses this machine tool as per standard operations recommended by Mazak product operation manual with only one exception that their engineers sometimes use it for end milling operation

```
<owl:NamedIndividual rdf:about="&ms;CompanyX_WS_NZ_Add">
  <rdf:type rdf:resource="&ms;Address"/>
  <postalCode rdf:datatype="&rdfs;Literal">0631</postalCode>
  <streetAddress rdf:datatype="&rdfs;Literal">47 William Pickering Drive</streetAddress>
  <addressLocality rdf:datatype="&rdfs;Literal">Auckland</addressLocality>
  <addressCountry rdf:datatype="&rdfs;Literal">New Zealand</addressCountry>
  <addressRegion rdf:datatype="&rdfs;Literal">Rosedale</addressRegion>
</owl:NamedIndividual>
<owl:NamedIndividual rdf:about="&ms;CompanyA_WS_NZ">
  <rdf:type rdf:resource="&ms;BusinessEntity"/>
  <hasOwner rdf:resource="&ms;CompanyX"/>
  <hasCapability rdf:resource="&ms;Drilling"/>
  <hasCapability rdf:resource="&ms;Finishing"/>
  <hasCapability rdf:resource="&ms;Milling"/>
  <hasCapability rdf:resource="&ms;Roughing"/>
  <hasCapability rdf:resource="&ms;Semifinishing"/>
  <hasCapability rdf:resource="&ms;Turning"/>
</owl:NamedIndividual>
```

Fig. 13. Digital twin for the case factory.

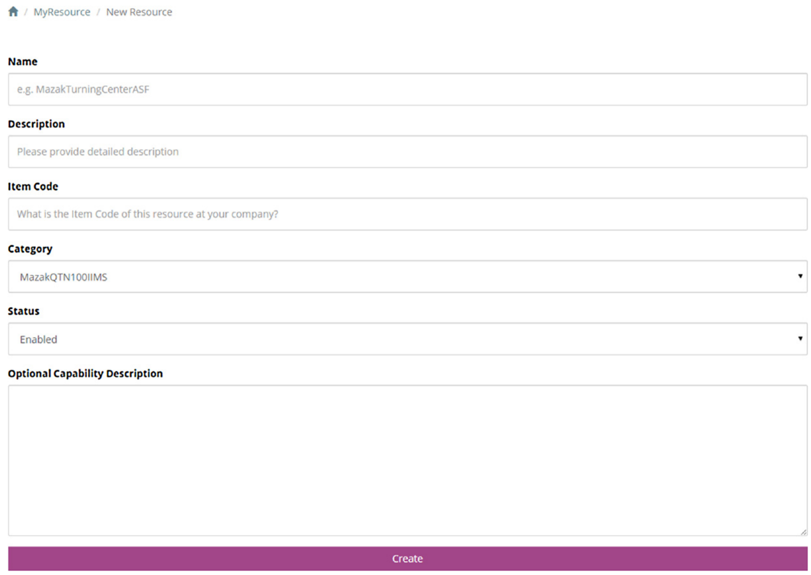

Fig. 14. Web interface for creating a digital twin of a manufacturing resource.


```
<owl:NamedIndividual rdf:about="&ms;%%SPINDLE1%%">
  <rdf:type rdf:resource="&ms;Spindle"/>
  <motorOutput rdf:datatype="&xsd;decimal">11</motorOutput>
  <chunkSize rdf:datatype="&xsd;decimal">6</chunkSize>
  <maxSpeed rdf:datatype="&xsd;decimal">6000</maxSpeed>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&ms;%%SPINDLE2%%">
  <rdf:type rdf:resource="&ms;Spindle"/>
  <motorOutput rdf:datatype="&xsd;decimal">11</motorOutput>
  <chunkSize rdf:datatype="&xsd;decimal">5</chunkSize>
  <maxSpeed rdf:datatype="&xsd;decimal">6000</maxSpeed>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&ms;%%TURRET%%">          To be replaced by
  <rdf:type rdf:resource="&ms;Turret"/>                  real name
  <numOfTools rdf:datatype="&xsd;int">12</numOfTools>
  <maxSpeed rdf:datatype="&xsd;double">4500.0</maxSpeed>
  <motorOutput rdf:datatype="&xsd;decimal">6</motorOutput>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&ms;%%QTNNAME%%">
  <rdf:type rdf:resource="&ms;MazakQTN100IIMS"/>
  <hasOwner rdf:resource="&ms;%%COMPANYNAME%%"/>
  <travelXAxis rdf:datatype="&xsd;decimal">185</travelXAxis>
  <maxMachiningDiameter rdf:datatype="&xsd;decimal">280</maxMachiningDiameter>
  <travelZAxis rdf:datatype="&xsd;decimal">455</travelZAxis>
  <maxMachiningLength rdf:datatype="&xsd;decimal">455</maxMachiningLength>
  <maxBarWorkCapacity rdf:datatype="&xsd;decimal">51</maxBarWorkCapacity>
  <maxSwing rdf:datatype="&xsd;decimal">550</maxSwing>
  <hasFunction rdf:resource="&ms;Drilling"/>
  <hasFunction rdf:resource="&ms;Grooving"/>
  <hasFunction rdf:resource="&ms;Turning"/>
  <hasFunction rdf:resource="&ms;PocketMachining"/>
  <hasSpindle rdf:resource="&ms;%%SPINDLE1%%"/>
  <hasSpindle rdf:resource="&ms;%%SPINDLE2%%"/>
  <hasTurret rdf:resource="&ms;%%TURRET%%"/>
</owl:NamedIndividual>
```


Fig. 15. Digital twin template for Mazak Quick Turn Nexus 100-II MS machine tool.

with a turning set-up. This unique capability is described in Jena rules and stored in the database.

The screenshot below (Fig. 17) shows a complete list of digital twins for the case factory recorded in the web system. A user can also click the icons to view detailed information about a resource, edit a resource or delete a resource.

5. Discussions

The case study demonstrated how the proposed resource virtualization framework enabled the case company to create digital twins for their factories from enterprise level down to equipment level in the context of developing a cyber-physical system enabled smart factory solution. Further research with the case company developing the smart factory solution was found as a straightforward process because all the manufacturing resources have been virtualized with explicit machine-readable capability description. The successful action of creating digital twins and later developing the smart factory solution was attributable to two key factors: an easy-to-follow resource virtualization methodology and an easy-to-use web interface for creating digital twins. It was found that the proposed resource virtualization methodology gives them a standardized way to tackle the problem of creating digital twins in the cyberspace. The test-driven process of analyzing requirements, developing semantic model and virtualizing resources is like common product development processes, which makes it easy to be understood and implemented by in-house developers from the company. Feedback from the case company also highlighted that the clear guidance on the data to be abstracted (i.e., technical properties, functional capabilities and availability status) made the development process much easier. In addition, the resource virtualization templates and developed web interface empowered the case company to quickly create digital twins with simple clicks.

Feedback from the case company suggested that more human-friendly tools should be provided for specifying the functional capabilities of a manufacturing resource though Jena rule syntax is straightforward to understand. Natural language processing tools are encouraged to be developed to automatically mining human language and convert to semantic web rule languages. It is also recommended that commercialization effort should happen to disseminate the research outcomes to a wider group, especially to machine tool vendors, to standardize digital twin templates for recognized manufacturing resources. Then, a network of digital twins can be formed with connections to distributed physical manufacturing resources so that cyber-physical systems could be developed in a drag-and-drop fashion.

6. Conclusion

Companies nowadays are required to transform the current practice of product development management into a smart factory solution that enables rapid production configuration to achieve the fast production of individualized products at changing scales. A key enabling technology in the generic smart factory architecture is resource virtualization or creation of digital twins. The presented research attempted to fill the

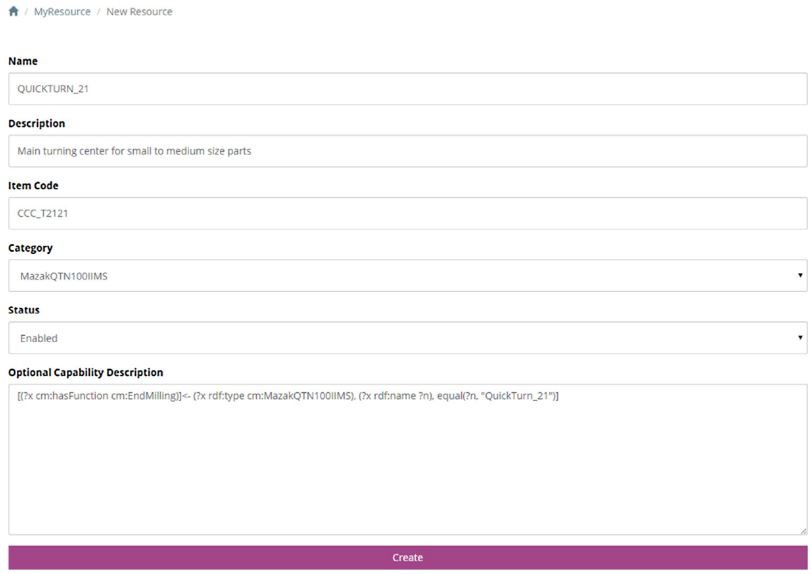

Fig. 16. Digital twin for a Mazak Quick Turn Nexus 100-II MS machine tool.

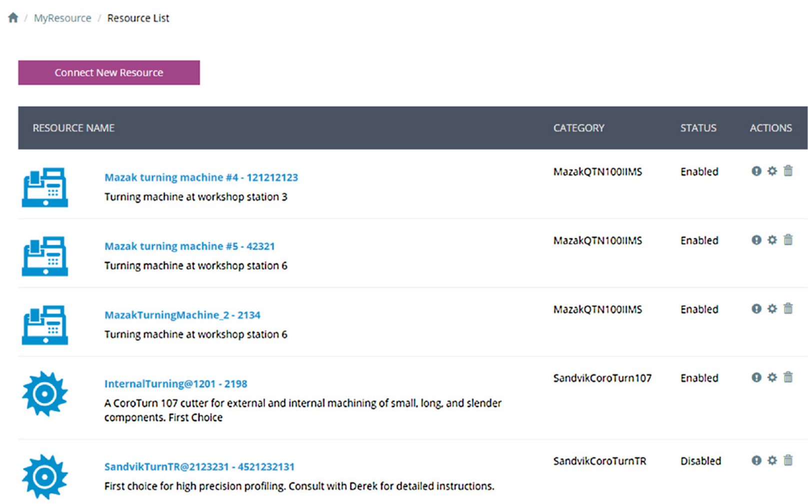

Fig. 17. Web interface for viewing virtualized resources.

gap that the industry needs a practical framework to enable companies to virtualize their manufacturing resources by specifying the data to be abstracted and the processes to virtualize manufacturing resources. A test-driven resource virtualization process is proposed as the recommendation for the industry to adopt to create digital twins for their smart factory solutions. The proposed process draws inspiration from past resource virtualization outcomes with special attention paid to the usability of the proposed solution. It provides a straightforward process for companies to create digital twins especially with the assist of a developed web interface, making the entire process of virtualization just filling a web form. Equally important, the research highlighted that the data to be virtualized in the cyberspace consists of three integral components: technical properties, functional capabilities, and real-time status. Semantic web languages, OWL and Jena are recommended as the modeling languages. To validate the proposed approach, a practical case study was undertaken at an international company, to create digital twins for all their manufacturing resources. This case study showcased how a multi-granularity digital twin structure is facilitated by the proposed virtualization approach. The enterprise-level digital twin was modeled by converting its vision statement and profile into machine-readable semantic data, using concepts from the developed semantic model. At the factory level, more detailed capability was specified, and digital twins were constructed at the equipment level.

The most notable contribution of the resource virtualization methodology is its flexibility in the virtualization process, making it easy for industry to adopt. The framework can be extended to any type of resource virtualization in developing a cyber-physical system. Future work includes the upgrade of the description language for functional capabilities using natural language processing technologies to allow capability description without the Jena syntax.

References

[1] Kagermann H, Helbig J, Hellinger A, Wahlster W. Recommendations for implementing the strategic initiative industrie 4.0: securing the future of German manufacturing industry; final report of the industrie 4.0 working group. Forschungsunion; 2013.
[2] Lee J, Bagheri B, Kao H-A. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. Manuf Lett 2015;3:18–23. http://dx.doi.org/10.1016/J.MFGLET.2014.12.001.
[3] Angrish A, Starly B, Lee Y-S, Cohen PH. A flexible data schema and system architecture for the virtualization of manufacturing machines (VMM). J Manuf Syst 2017;45:236–47. http://dx.doi.org/10.1016/J.JMSY.2017.10.003.
[4] Tao F, Zhang M. Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing. IEEE Access 2017;5:20418–27. http://dx.doi.org/10.1109/ACCESS.2017.2756069.
[5] Qi Q, Tao F. Digital twin and Big data towards smart manufacturing and industry

4.0: 360 degree comparison. IEEE Access 2018;6:3585–93. http://dx.doi.org/10.1109/ACCESS.2018.2793265.
[6] Schmidt B, Wang L, Galar D. Semantic framework for predictive maintenance in a cloud environment. Procedia CIRP 2017;62:583–8. http://dx.doi.org/10.1016/J.PROCIR.2016.06.047.
[7] Zhou Q, Yan P, Xin Y. Research on a knowledge modelling methodology for fault diagnosis of machine tools based on formal semantics. Adv Eng Inf 2017;32:92–112. http://dx.doi.org/10.1016/J.AEI.2017.01.002.
[8] Ameri F, Sabbagh R. Digital factories for capability modeling and visualization. Cham: Springer; 2016. p. 69–78. http://dx.doi.org/10.1007/978-3-319-51133-7_9.
[9] Nuñez DL, Borsato M. An ontology-based model for prognostics and health management of machines. J Ind Inf Integr 2017;6:33–46. http://dx.doi.org/10.1016/J.JII.2017.02.006.
[10] Babiceanu RF, Seker R. Big data and virtualization for manufacturing cyber-physical systems: a survey of the current status and future outlook. Comput Ind 2016;81:128–37. http://dx.doi.org/10.1016/J.COMPIND.2016.02.004.
[11] Lu Y, Xu X, Xu J. Development of a hybrid manufacturing cloud. J Manuf Syst 2014;33:551–66. http://dx.doi.org/10.1016/J.JMSY.2014.05.003.
[12] Laguionie R, Rauch M, Hascoët J-Y, Suh S-H. An eXtended manufacturing integrated system for feature-based manufacturing with STEP-NC. Int J Comput Integr Manuf 2011;24:785–99.
[13] Yang W, Xu X. Modelling machine tool data in support of STEP-NC based manufacturing. Int J Comput Integr Manuf 2008;21:745–63. http://dx.doi.org/10.1080/09511920701810691.
[14] Wang X, Xu X. Virtualise manufacturing capabilities in the cloud: requirements, architecture and implementation. Int J Manuf Res 2014;9:348–68.
[15] Hu C, Xu C, Cao X, Zhang P. Study on the multi-granularity virtualization of manufacturing resources. ASME 2013 int. manuf. sci. eng. conf. collocated with 41st North Am. Manuf. Res. Conf. Am Soc Mech Eng 2013. http://dx.doi.org/10.1115/MSEC2013-1174. V002T02A008-V002T02A008.
[16] Liu N, Li X, Shen W. Multi-granularity resource virtualization and sharing strategies in cloud manufacturing. J Netw Comput Appl 2014;46:72–82.
[17] Jang J, Jeong B, Kulvatunyou B, Chang J, Cho H. Discovering and integrating distributed manufacturing services with semantic manufacturing capability profiles. Int J Comput Integr Manuf 2008;21:631–46.
[18] Kjellberg T, von Euler-Chelpin A, Hedlind M, Lundgren M, Sivard G, Chen D. The machine tool model—a core part of the digital factory. CIRP Ann 2009;58:425–8. http://dx.doi.org/10.1016/J.CIRP.2009.03.035.
[19] Zhao YY, Liu Q, Xu WJ, Gao L. Modeling of resources capability for manufacturing equipments in cloud manufacturing. Appl Mech Mater 2013;271:447–51. Trans Tech Publ.
[20] Al Sunny SMN, Liu XF, Shahriar MR. MTComm: a semantic ontology based internet scale communication method of manufacturing services in a cyber-physical manufacturing cloud. 2017 IEEE Int. Congr. Internet Things 2017. p. 121–8. http://dx.doi.org/10.1109/IEEE.ICIOT.2017.22.
[21] MS Fox, M Barbuceanu, M Gruninger. An organisation ontology for enterprise modelling: preliminary concepts for linking structure and behaviour. Proc. 4th IEEE Work. Enabling Technol. Infrastruct. Collab. Enterp. (WET ICE' 95), IEEE Comput. Soc. Press; n.d., p. 71–81. doi:10.1109/ENABL.1995.484550.
[22] Hankel M, Rexroth B. Industrie 4.0: the reference architectural model industrie 4.0 (RAMI 4.0). Frankfurt Am Main, Ger ZVEI-German Electr Electron Manuf Assoc; 2015.
[23] Zezulka F, Marcon P, Vesely I, Sajdl O. Industry 4.0 – an introduction in the phenomenon. IFAC-PapersOnLine 2016;49:8–12. http://dx.doi.org/10.1016/J.IFACOL.2016.12.002.
[24] Wang L. Machine availability monitoring and machining process planning towards Cloud manufacturing. CIRP J Manuf Sci Technol 2013;6:263–73. http://dx.doi.org/10.1016/J.CIRPJ.2013.07.001.
[25] Cai Y, Starly B, Cohen P, Lee Y-S. Sensor data and information fusion to construct digital-twins virtual machine tools for cyber-physical manufacturing. Procedia Manuf 2017;10:1031–42. http://dx.doi.org/10.1016/j.promfg.2017.07.094.
[26] Liu C, Cao S, Tse W, Xu X. Augmented reality-assisted intelligent window for cyber-physical machine tools. J Manuf Syst 2017;44:280–6. http://dx.doi.org/10.1016/J.JMSY.2017.04.008.
[27] Pinto HS, Staab S, Tempich C. DILIGENT: towards a fine-grained methodology for distributed, loosely-controlled and evolvInG. Proc. 16th Eur. Conf. Artif. Intell. (ECAI 2004) 2004;110:393.
[28] Suárez-Figueroa MC, Gómez-Pérez A, Muñoz O, gOntt Vigo M. A tool for scheduling and executing ontology development projects. SEKE 2010 - Proc. 22nd Int. Conf. Softw. Eng. Knowl. Eng.. 2010. p. 614–9.
[29] Janzen D, Saiedian H. Test-driven development concepts, taxonomy, and future direction. Computer (Long Beach Calif) 2005;38:43–50. http://dx.doi.org/10.1109/MC.2005.314.
[30] Schwaber K, Beedle M. Agile software development with scrum vol. 1. Prentice Hall Upper Saddle River; 2002.
[31] Uschold M, Gruninger M. Ontologies: principles, methods and applications. Knowl Eng Rev 1996;11:93–136.
[32] Lu Y, Wang H, Xu X. ManuService ontology: a product data model for service-oriented business interactions in a cloud manufacturing environment. J Intell Manuf 2016:1–18. http://dx.doi.org/10.1007/s10845-016-1250-x.
[33] Boley H, Paschke A, Shafiq O. RuleML 1.0: the overarching specification of web rules. Lect Notes Comput Sci 2010;6403:162–78.
[34] Horrocks I, Patel-Schneider PF, Boley H, Tabet S, Grosof B, Dean M. SWRL: a semantic web rule language combining OWL and RuleML. W3C Memb Submiss 2004;21:79.
[35] Kifer M. Rule interchange format: the framework. Web reason. Rule syst. Springer; 2008. p. 1–11.
[36] Jena A. Reasoners and rule engines: Jena inference support. Apache Softw Found; 2013.
[37] Barbau R, Krima S, Rachuri S, Narayanan A, Fiorentini X, Foufou S, et al. OntoSTEP: enriching product model data using ontologies. CAD Comput Aided Des 2012;44:575–90. http://dx.doi.org/10.1016/j.cad.2012.01.008.
[38] M. Corporation QUICK TURN NEXUS 100-II MS n.d.; 2015. https://www.mazakusa.com/machines/quick-turn-nexus-100-ii-ms/.
[39] ISO.ISO/TR 15608: 2013 welding - guidelines for a metallic materials grouping system. 2013.