



Online capability-based resource allocation for on-site construction operations utilizing digital twin models

Sören Münker¹ · Peter R. Wildemann^{2,3} · Amon Göppert¹ · Sigrid Brell-Cokcan² · Robert H. Schmitt¹

Received: 22 September 2021 / Accepted: 9 January 2022 / Published online: 25 January 2022
© The Author(s) 2022

Abstract

Planning and controlling on-site construction operations are complex and dynamic procedures, mainly manually executed without algorithmic decision support. An initial challenge is to allocate available resources to construction processes based on required and available capabilities. Due to the dynamic nature of construction projects (e.g., redesigns, resource failure, unpredictable restrictions), there is a demand for frequent reallocation of resources. In recent years, researchers studied capability-based resource allocation approaches by defining ontologies to describe the capabilities of resources. However, since most of the existing approaches focus on ontologies for resources in production environments (e.g., industrial robots), the modeling and application of the models for online allocation in dynamic construction environments remain unsolved. In this study, an ontology-based Digital Twin model, adopted from a production engineering background, is used to enable online capability-based resource allocations for construction-specific approaches. The Digital Twin model can be updated by a lightweight, publish-subscribe network, triggering an update of capability-based feasibility checks for resource allocations. The resulting framework is tested on a demo construction project from the research project “Internet of Construction (IoC)”. The results contribute to the automation of planning and controlling resource allocations for dynamic on-site construction operations. Using machine-readable ontologies, the transition from manually performed activities to robotically supported tasks is enabled.

Keywords Digital twin · Ontology · Planning · Control · Capability · Knowledge-transfer · On-site operations

1 Introduction and motivation

Automatically managing the complexity of construction operations and logistics to reduce resource utilization without compromises in time and quality is an ongoing research topic. A key necessity for automating construction management is the assessment, whether a resource can execute specific activities or not (Weser et al. 2020). For solving these resource allocation problems, machine-readable modeling of required and provided capabilities is a crucial enabler. A capability is defined as the effect that the resource can

cause in a specific context (Weser et al. 2020). For example, a robot has the capability of “pick-and-place” which can be further decomposed in subcapabilities “grasp”, “release” and “move”. Based on this concept, automated matchmaking between resources and processes is defined as capability-based resource allocation. For modeling capabilities in the right detail, data of products, processes and resources is required.

In the construction domain, Building Information Modeling (BIM) is currently considered the most promising methodology to provide a single source of truth with real-time access to all necessary data sources across the whole lifecycle of buildings (Bradley et al. 2016). Since BIM is driven by geometrical datasets capturing as-designed building objects, properties and spatial organization, various authors state that integrating other frameworks is needed to achieve the goal envisioned above (Zhong et al. 2019; Tang et al. 2019). Current BIM models can be considered digital models without communication connections to real assets (Al-Sehrawy and Kumar 2021). Recent developments enhance BIM models by integrating Internet

✉ Sören Münker
s.muenker@wzl.rwth-aachen.de

¹ Chair for Production Metrology and Quality Management, WZL | RWTH Aachen University, Aachen, Germany

² Chair of Individualized Production in Architecture, RWTH Aachen University, Aachen, Germany

³ LEONHARD WEISS GmbH & Co. KG, Göppingen, Germany

of Things (IoT) devices to enrich them with real-time data. However, these applications have mainly monitoring purposes and focus on applying one-directional communication (Tang et al. 2019). The enhancement of digital models by one-directional communication can be considered a Digital Shadow (DS) (Kritzinger et al. 2018; Al-Sehrawy and Kumar 2021). Nevertheless, most BIM and IoT integration concepts consider information-rich monitoring purposes for smart home applications, while applications in resource allocation for construction operations and logistic management are a rarely covered field (Tang et al. 2019).

Digital Twin (DT) models enhance DS models, incorporating automatic bidirectional communication between physical assets and their digital representation (Kritzinger et al. 2018). Most research is done in developing frameworks for standardized modeling of DTs using semantic representations of relevant aspects by defining ontologies (Hildebrandt et al. 2020; Bao et al. 2020; Liu et al. 2020; Boje et al. 2020).

Using ontologies is critical for standardization, but most of these frameworks lack goal-driven data flow for explicit purposes (Tang et al. 2019). Existing ontologies for two-way information flow are currently not implemented in the construction context because the cost-benefit ratio of DT technologies is currently unclear (Scherer and Schapke 2014). A use case specific implementation is needed to apply ontology-driven DT technologies for capability-based resource allocation in construction operations efficiently. Therefore, building-centric information needs to be combined with rich process and resource information to enable the scheduling of applications in construction operation environments.

This work addresses this gap by designing, implementing and deploying an ontology-based DT model for capability-based resource allocation for on-site construction operations. This work is structured as follows: A comprehensive background on DTs and ontology modeling is presented (Sect. 2). Afterward, the methodological approach of defining the ontology-driven DT model is illustrated (Sect. 3). A proposal for a custom ontology for capability-based resource allocation problems is defined by accumulating relevant ontologies (Sect. 4). Standardized DT modeling of the input data for capability-based feasibility checks is presented subsequently (Sect. 5). The automated deployment is illustrated for a sample resource allocation use case (Sect. 6). In the end, a conclusion and outlook on further research are given (Sect. 7).

2 Background

2.1 Modeling digital twins

The concept of a DT consists of three elements. The idea is that a (1) real physical object has an (2) informational virtual duplicate, a twin. Both are connected through a (3)

bidirectional data and information flow and exist during the entire lifecycle of the object. Grieves and Vickers (2017)

Several institutions work on the implementation of the DT concept by developing methods of implementation and contributing to a common understanding of DT. The US-based “Industrial Internet Consortium” (IIC) provides the Industrial Internet Reference Architecture (IIRA), focusing on the Industrial Internet of Things (IIoT) terminology (Industrial Internet Consortium 2019). Similarly, the German initiative “Plattform Industrie 4.0” developed the Reference Architecture Model Industrie 4.0 (RAMI4.0) in 2015. RAMI4.0 introduces the Asset Administration Shell (AAS) as a modeling method to implement the concept of the DT. Boss et al. (2020)

An AAS depicts the digital representation for real and virtual assets, communicating through a standardized communication interface. Assets can be physical (e.g., singular construction resource) or virtual (e.g., digital construction diary). Tantik and Anderl (2017)

The AAS schemes can be connected hierarchically to implement different levels of detail for the DT models. Figure 1 shows the basic structure of an AAS. Every AAS consists of a header and a body. The header contains essential information to identify assets, and the body contains the corresponding properties encapsulated in submodel definitions (Boss et al. 2020). Asset types are defined by their combination of submodels, while instances of the types are generated during the value chain of a specific use case (Bader et al. 2020). In a production environment, the header is usually standardized according to (IEC 62832-2) (IEC 2020). Many submodels are predefined for manufacturing purposes (e.g., based on ecl@ass Hepp 2005). The IIoT Connectivity Stack Model is proposed for connecting DTs with real-world assets (Joshi et al. 2018). By developing AAS for all assets, transparency and shorter response times can be realized along the complete supply chain, such as those required by Ala-Risku and Kärkkäinen (2006), Ala-Risku and Kärkkäinen (2006), which is especially helpful during the construction operation

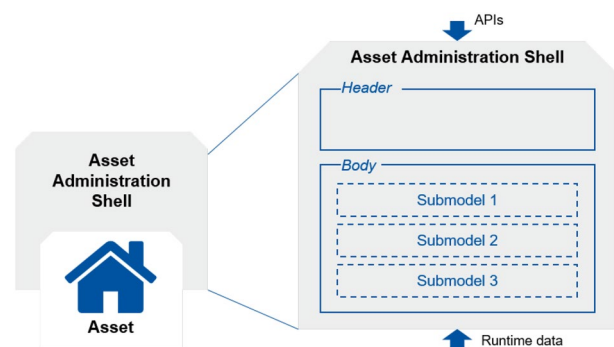


Fig. 1 Asset Administration Shell (AAS) concept, based on Boss et al. (2020); ZVEI (2016)

phase. Through the use of APIs, the asset's data and functions can be accessed, while actual asset data is constantly integrated into the AAS via runtime data (Boss et al. 2020).

Different methodologies for development and deployment of DT models can be found in the literature. Most of these have a particular focus on the ontology-based definition and the standardized modeling for the DT (Hildebrandt et al. 2020; Bao et al. 2020; Wan et al. 2018; Liu et al. 2020; Steinmetz et al. 2017; Park et al. 2020). Automatic deployment of DT is a less frequent topic in research. Steinmetz et al. (2017) developed an add-on for the Protegé ontology editor to deploy IoT-entities from a lightweight ontology (Steinmetz et al. 2017). Kousi et al. (2019) automatically deployed a DT in the Robot Operating System (ROS) framework with a focus on 3D environment representation (Kousi et al. 2019). Based on these approaches, Göppert et al. (2021) proposed a Digital Twin Pipeline for automated deployment of a DT (Göppert et al. 2021).

2.2 Ontology-based modeling for construction operations

For the realization of the DT concept, all assets must be able to communicate. The formal representation of knowledge is achieved by using conceptual ontologies and their instances, providing a common understanding of the environment (Gruber 1993; Noy and McGuinness 2001; Getuli and Capone 2019). An ontology consists of three parts: axioms (rules and constraints), relations and a taxonomy (El-Diraby and Kashif 2005; El-Diraby 2013). These parts ensure that the knowledge is well defined and unequivocal due to a given semantics and syntax. The taxonomy is hierarchically represented by different classes and their subclasses (Noy and McGuinness 2001). Besides ontologies, classification systems are used to represent knowledge (El-Diraby 2013). Organizations like the ISO or the national standards institutes provide domain-related classification systems (taxonomies), giving a standardized vocabulary and structure with a particular commitment in use. It is worth mentioning that there are multiple solutions in the field of ontologies since there are many similar approaches with a different focus in detail. It is much more important that the ontology is accepted and used by all participants and thus only fulfills its function (Noy and McGuinness 2001). Semantic Web is an approach to achieve knowledge representation and communication in a machine-readable and understandable way (Berners-Lee et al. 2001). It has to be possible to integrate ontologies flexibly and dynamically to handle knowledge in a changing environment. Concepts such as Linked Data in the Information Container for Document Delivery (ICDD) meet this requirement, linking information to their source by always containing the defining ontology (ISO 2020).

Ontologies can be described by several descriptive languages (Benevolenskiz 2016). Nowadays, the World Wide Web Consortium (W3C) standards are often used because of their semantic web-friendly structure (Pauwels et al. 2018). Rasmussen et al. (2020) point out that modularity and extensibility are highly needed within an ontology (Rasmussen et al. 2020). Semantic Web Technologies allow a modular and expandable information structure for cross-domain use cases (Pauwels et al. 2018). The Resource Description Framework (RDF) was developed for the technical realization of this idea, enabling structured, machine-readable expressions due to a given syntax. For the machine-readable implementation of taxonomies and relations, Web Ontology Language (OWL) is often used, based on RDF (Fumagalli et al. 2014; Negri et al. 2017). The Semantic Web Rule Language (SWRL), based on RDF, can extend OWL with rules for the instances, not the classes (Bunte et al. 2018). With the help of the query language SPARQL, the data within the ontology can be specifically queried, so that via the links through RDF, information can be specifically extracted from the data models (Pérez et al. 2009).

In order to create a functional DT for construction operation, in which all actors and machines can communicate properly with each other, existing ontologies must be adopted or further developed. Different ontologies, which are linked with each other and extended as required, thus form the semantic basis for the creation of DT. Although different ontologies in heterogeneous systems can be interconnected, this requires a very high manual effort to bring the different ontologies into agreement. The choice of an existing ontology or the creation of a new one, which is available to all participants, minimizes the risk of error and promotes its use. Ontologies thus serve as an important enabler to realize the use of robotics in the dynamic environment of the construction site in the future.

Operating a construction site consists of two main topics: (1) construction processes and (2) intra-logistical processes (Zeng et al. 2017). An ontology that covers both aspects is required. It has to provide the needed knowledge for construction operation, connecting detailed aspects of building products, resources (labor, machines and robots, spaces) and processes (time and procedures). Several knowledge-representation approaches are available, covering these two main aspects individually or at a high level of abstraction. Not all of them follow the strict definition of an ontology. Nevertheless, the approaches shown in Figure 2 have a strong focus on semantic modeling to enable standardized, formalized or machine-readable representation of knowledge. The figure provides an overview of the chronological development of selected ontologies with a generic construction and manufacturing focus.

Mascardi et al. (2007) provide a comparison of upper ontologies, which are often used as foundations (Mascardi

et al. 2007). The studies of Pauwels et al. (2017), Zhong et al. (2019) and Getuli und Capone (2019) provide extensive reviews on existing ontologies for the construction domain (Pauwels et al. 2017; Zhong et al. 2019; Getuli and Capone 2019). Negri et al. (2017) compile recent papers related to cross-industry logistics ontologies (Negri et al. 2017). The most relevant ontologies for this research are described in Table 1.

Most construction-related approaches are very product-centric and do not focus on the construction operation. There is a lack in covering the required aspects for capability-based resource allocation for planning and controlling the construction operation since the “Building Product”-information is an important element, but not the only one in the construction operation process. A construction-operation-centered viewpoint leads to the need for an integrated approach with products, processes and resources, as well-established manufacturing ontologies do. Consequently, the transfer of knowledge and approaches from manufacturing to the construction industry should be considered. IFC contains the class “Resource” (BuildingSmart: Ifcresource 2021), but it is rarely used in practical applications and also not specified for the detailed description of an object. Using IFC for construction operation lacks more differentiated descriptions within the “Resource” and “Process” classes and their overall relationships regarding the construction operation. Due to the building element-orientated approach of IFC the process-orientated challenges of the construction operation along the value chain are not well covered. Although the MEFISTO approach provides the direct link between product and processes, due to its static nature, it is not ideally suited to allow flexible and modular configurations and to propose execution variants (Benevolenskiy 2016).

3 Methodology

The goal of this research is to deploy a proposal for a reusable ontology-based DT application for capability-based resource allocation problems for on-site construction environments by applying an existing method to DT modeling. Based on the approaches from Sect. 2.1, the approach from Göppert et al. (2021), who proposed a Digital Twin Pipeline for automated deployment of a DT, will be applied in the following. Their methodology is used because they provide a holistic approach to DT development (from ontology to deployment) that the other authors cited in Sect. 2.1 do not. The Digital Twin Pipeline proposes a procedure in three phases (see Fig. 3). Göppert et al. (2021)

1. Ontology-based Definition (Sect. 4): Definition of required data and information by domain experts leads to the required knowledge of the DT. Existing and established meta models and domain ontologies need to be combined to create an extended custom ontology. The definition of this custom ontology is the foundation of the DT modeling.
2. Standardized Modeling (Sect. 5): Add use case specific data as instances of the ontology’s classes. The conceptual DT gets instantiated through this individual data. The structure of the individual instances follows the superior classes.
3. Automated Deployment (Sect. 6): Deployment of the DT due to the given structured instances as input data. Dynamic values for measures can be added to the DT. Automatic connection to the Communication Processor for enabling bidirectional communication.

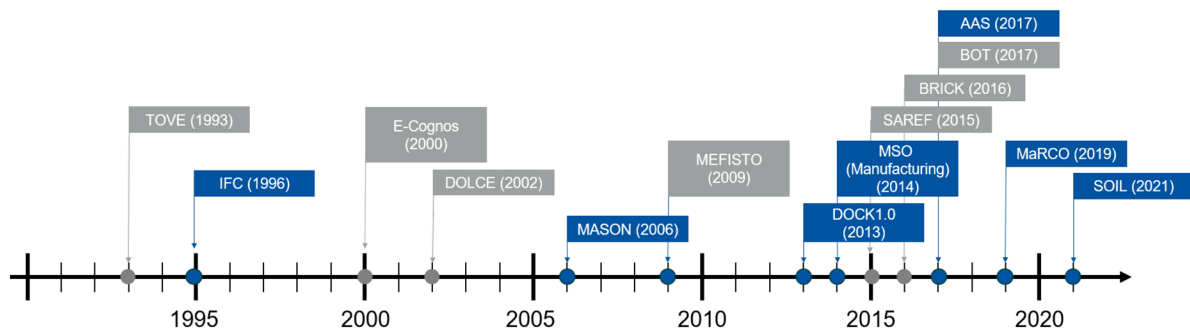


Fig. 2 Relevant ontologies, sorted by year of publication. References for grey approaches: TOVE (Fadel et al. 1993), E-Cognos (Lima et al. 2003; El-Diraby and Kashif 2005), DOLCE (Masolo et al. 2003), MEFISTO (Baling et al. 2010; Benevolenskiy et al. 2012), SAREF

(Daniele et al. 2015; Rasmussen et al. 2017), BRICK (Balaji et al. 2016), BOT (Rasmussen et al. 2017, 2020); blue approaches are referenced in Table 1

Table 1 Selected established ontologies from the categories “Construction” (C), “Manufacturing” (M) and “General” (G)

Name	Cat.	Content	Strengths	References
IFC (1996)	C	Industry Foundation Classes (IFC); Strong focus on 3D geometry representation; Classes e.g.: objects (to highlight in particular: building elements), products, resources and processes	Most commonly used (product) data model in the construction domain at present	(Delgado et al. 2013; Pauwels and Terkaj 2016; Rasmussen et al. 2020)
MASON (2006)	M	Manufacturing’s Semantics Ontology (MASON); Manufacturing-domain, upper level; Main-classes: entities, operations and resources	Focus on resources with differentiation into several resource types, but on an upper level	(Lemaignan et al. 2006)
DOCK1.0 (2013)	C	Domain Ontology for Construction Knowledge (DOCK1.0); Enhances the initial E-Cognos approach; Classes like project, time, process, resource, product and actor	Based on IFC and additionally contains TOVE in many aspects. Provides base platform for additional subdomain and application level ontologies. Lack of detailed process and resource description	(El-Diraby 2013; El-Gohary and El-Diraby 2010; El-Diraby and Osman 2011)
MSO (2014)	M	Manufacturing Systems Ontology (MSO); Domain of production systems; Covers the physical, the process and the control aspect of logistic systems; Logistics resource classes, e.g. Transporter, Storage	Detailed focus on (logistical) resources. Negri et al. (2017) extend the MSO at the domain of internal logistics	(Fumagalli et al. 2014; Negri et al. 2017)
AAS (2017)	G	AAS concept is a meta-model ontology; Divides an object into header and body	Standardization of the implementation of DT models and thus enabling DT applications	(Boss et al. 2020; ZVEI 2016)
MaRCO (2019)	M	Manufacturing Resource Capability Ontology (MaRCO); Relations between product requirements, resources and capabilities	Main objective: enable a comparable description of the functionality of resources to be able to match product requirements with resource capabilities	(Järvenpää et al. 2019)
SOIL (2021)	G	SensOr Interfacing Language (SOIL); Object body consists of four core elements: parameters, variables, functions and components; Each component is an object recursively	Composition of the body allows the flexible adaptation of the asset to changing circumstances due to its modular structure.	(Bodenbenner et al. 2021)

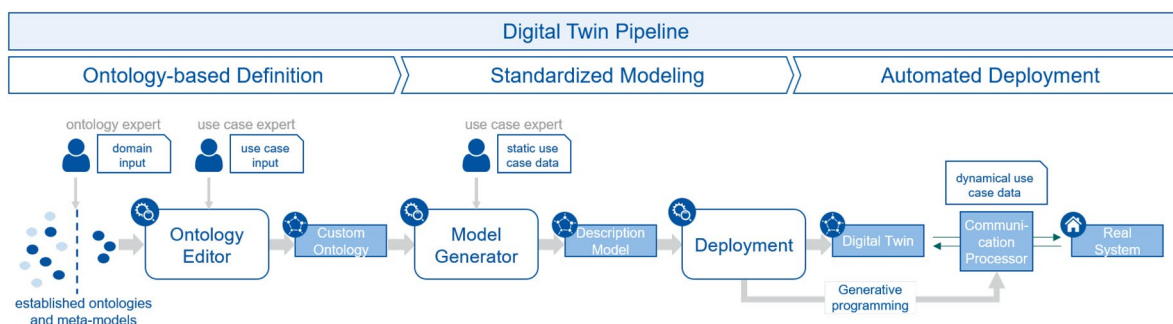


Fig. 3 Digital Twin Pipeline, based on Göppert et al. (2021)

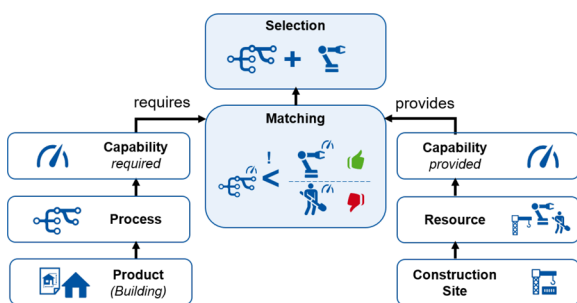


Fig. 4 Capability-based modeling - exemplary for construction operations (based on Kluge 2011)

4 Ontology-based definition for capability-based resource allocation

Facing the planning and controlling of construction operation means the complex handling of different participants and objects in the context of time, cost and availability of resources. This research focuses on basic issues on site: the matching of a given construction task to certain available resources, handling the task under some constraints. The optimization of the processes themselves through the application of specific methods, e.g. Last Planner System®, as considered by Ala-Risku and Kärkkäinen (2006), is not part of this research (Ala-Risku and Kärkkäinen 2006).

This section addresses the required knowledge about the acting objects, their properties and interrelationships, so that the subsequent software implementation, presented in Sect. 5 and 6, is comprehensible and adaptable. It explains the elaboration of the new custom ontology for construction operation purposes, which was created by combining and extending well-established ontologies and meta-models. The main framework is demonstrated in Fig. 4, in accordance with the capability-based modeling approach of Kluge (2011). In order to sharpen the

scope of the new ontology, competency questions (CQ) are formulated (Noy and McGuinness 2001). The CQ are derived from the requirements of the framework, described in more detail below.

The product, process and resource approach forms the basis for the new construction operation ontology. The elements of the concept are divided as follows: The planned building (Product), covered in a building model, is divided into its elements (e.g., walls, columns, slabs). They are arranged in a feasible sequence that can be executed in the construction phase. Every building element is associated with a construction reference process that gets further defined based on the elements' parameters, characteristics and further restrictions. Each process can be described by its required capabilities for its execution. The construction site provides different types of resources, which are needed for process execution. The available resources provide different capabilities depending on their current setup. The required capabilities of the processes are directly matched to the capabilities provided by the available resources. The best-fitting resource-capability match gets selected and is connected to the corresponding process. The level of detail in which the elements for matching are described is closely related to the intended use case and must be chosen accordingly. In the case presented here, the ontology is conceived from the point of view of a general contractor who has to coordinate operations at a comparatively coarse level and, accordingly, does not have to consider all the details of each process.

This results in the following four CQs, which are now used to select the appropriate ontologies and, if necessary, to fill in any remaining gaps in the new custom ontology with elements that are still missing.

- **CQ 1:** What processes are necessary to manufacture a building element on the construction site?
- **CQ 2:** What resources are available on construction site?
- **CQ 3:** What capabilities are required to perform a particular process?

- **CQ 4:** What capabilities does a particular resource provide?

The concept shown in Fig. 4 provides the structural base, which is now filled by the ontologies presented in Sect. 2, to represent this specific domain knowledge. The DOCK1.0 builds the foundation of the custom ontology because it contains the main classes resource, process and product. It also considers the aspects of time and organization of work in projects. Additional ontologies, presented in Table 1 in Sect. 2, get aligned to the DOCK.

DOCK includes IFC, which is taken into account for the building elements (products) due to it being the most common data exchange format in the construction industry. In DOCK, the individual IFC building elements can be assigned to individual processes that are necessary to manufacture the components, answering CQ 1. The “Process” class of DOCK gets extended by object properties as “is Subprocess of” or “is Predecessor” to describe the relations between different processes and be able to design more details in processes. To be able to map differences in the processes, the DOCK “Process” is also extended by the subclasses “Transport process”, “Construction process” and “Production process”, as done by Zeng et al. (2017).

In order to be able to take an even more differentiated view on the resources and their capabilities, the “Resource” class of DOCK is replaced by MASON. In MASON, resources are divided into “Geographical Resources”, “Human Resources” and “Material Resources”, with the latter referring in particular to equipment and machinery (Lemaignan et al. 2006). This allows MASON to answer CQ 2 much more precisely than DOCK would do. The extension of MASON with MSO also contributes to this, allowing logistical resources to be described in detail in addition to the manufacturing and construction site resources.

MaRCO provides dynamic functionalities by connecting “Process” class of DOCK and “Resource” classes of MASON by means of capability classes and appropriate relations. By linking the “Process” class of DOCK with the relation “requires Capability” to the “Capability” class of MaRCO, CQ 3 gets answered. In addition, CQ 4 is answered by connecting the MASON “Resource” classes to the MaRCO “Capabilities” via the “provides Capability” relationship. It has shown great applicability for robotic resources already and thus is a foundation for integrating robots in on-site construction processes (Järvenpää et al. 2019).

The generic meta-model-concepts AAS and SOIL provide the framework for the objects concerning the modular

implementation of the DT-objects. Every class is associated with an AAS that contains the meta-structure of SOIL (including id, type, name, variables, parameters, components etc.). Detailed information of the selected ontologies can be taken from the literature referred to in Sect. 2.2.

Figure 5 shows an extract of the new custom ontology based on selected parts of existing ontologies, called “Internet of Construction On-Site Ontology” (IoC-OSO), composed of the partial meta-models and aligned ontologies. The IoC-OSO contains the required knowledge, which is needed to deploy the on-site DT. The taxonomy of relations and axioms from DOCK1.0 will be applied, although these relations and axioms are not the focus of this research.

5 Standardized modeling of the construction resource allocation problem

Based on the introduced IoC-OSO, a standardized description model needs to be generated, taking into account static use case data. The description model is a static machine-readable description of all relevant data for the capability-based resource allocation problem (e.g., product, process and resource data). Thus, the description model represents all relevant assets of a use case at a given time.

Modeling tools for standardized model generation are provided for the entity-level modeling to convert static use case data in standardized data exchange formats (Fig. 6). JavaScript Object Notation (JSON) format is used for data exchange, where schemes ensure formal correctness according to the IoC-OSO taxonomy. The schemes can be used to interchange entity modeling tools flexibly. After manual modeling of the entities by the domain expert, the entity representations are instanced with the object-oriented model derived from the IoC-OSO. In the following, the entity-level modeling and the object-oriented model are described in more detail.

5.1 Entity-level model generation

The static use case data for capability-based resource allocation in on-site construction operations mainly consist of the (1) precedence graph, (2) building element list, (3) reference processes and (4) resource profiles. This data corresponds to specific use case entities that need to be modeled in a standardized manner. The entity-level representations of an exemplary use case are demonstrated below to illustrate the standardized models. The use case is a sample model

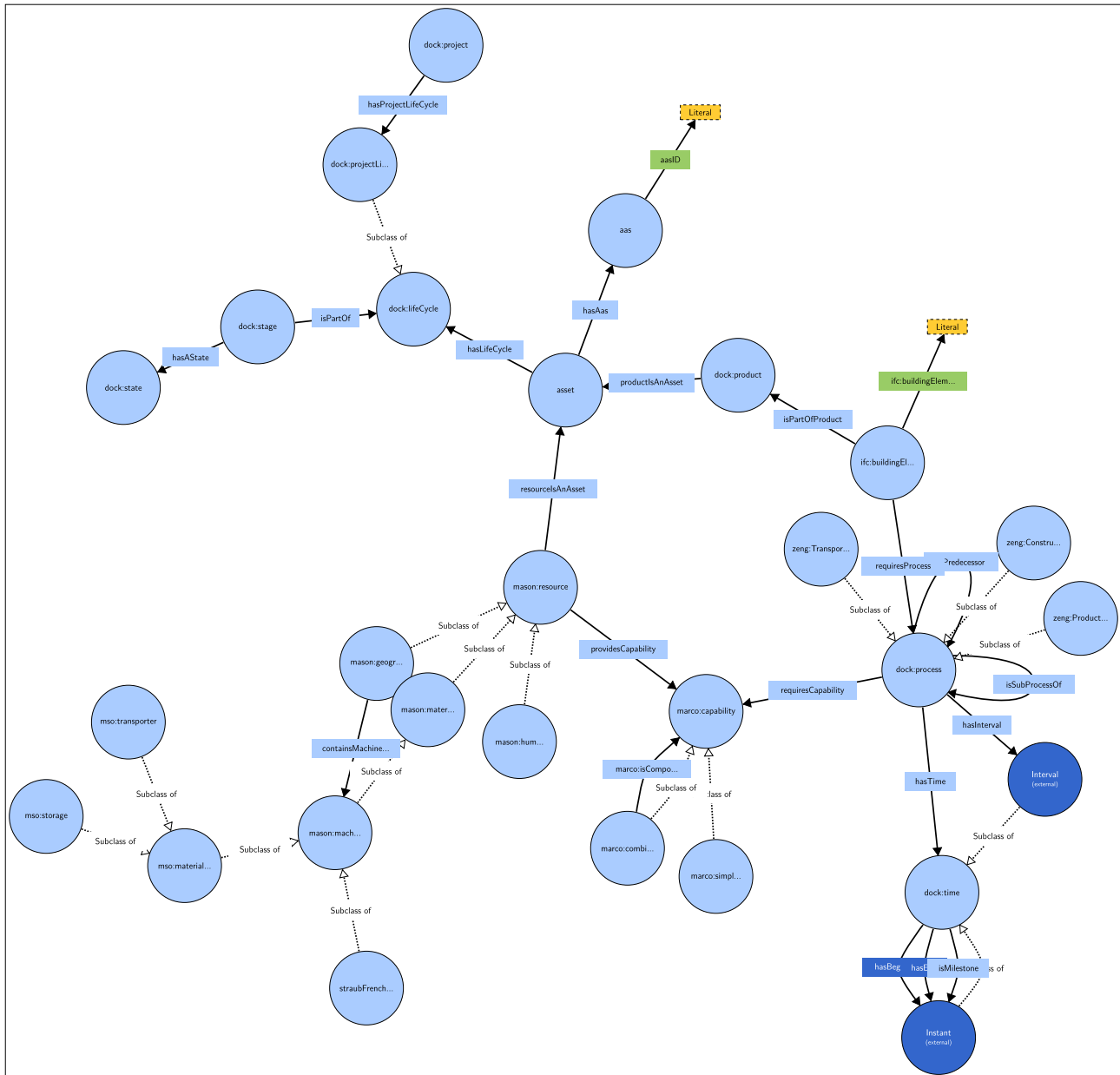


Fig. 5 Extract of the Internet of Construction On-Site Ontology (IoC-OSO)

depicting a bus stop provided for the research project “Internet of Construction (IoC)”.

(1) The precedence graph contains information about geometrically feasible assembly sequences of building elements, represented in the same way as known for precedence diagram methods. The precedence information can be generated by graph modeling tools (e.g., Gephi) and transmitted as a serialized adjacency JSON file. The nodes of the graph

represent the building elements, while the arcs represent the mandatory sequence of assembling the elements. Thus, the precedence graph links the building elements of an IFC model considering feasible assembly sequences (e.g., based on geometrical constraints). An excerpt of the precedence graph of the bus station use case is included in Fig. 7. The full graph can be found in the GitHub-Project.

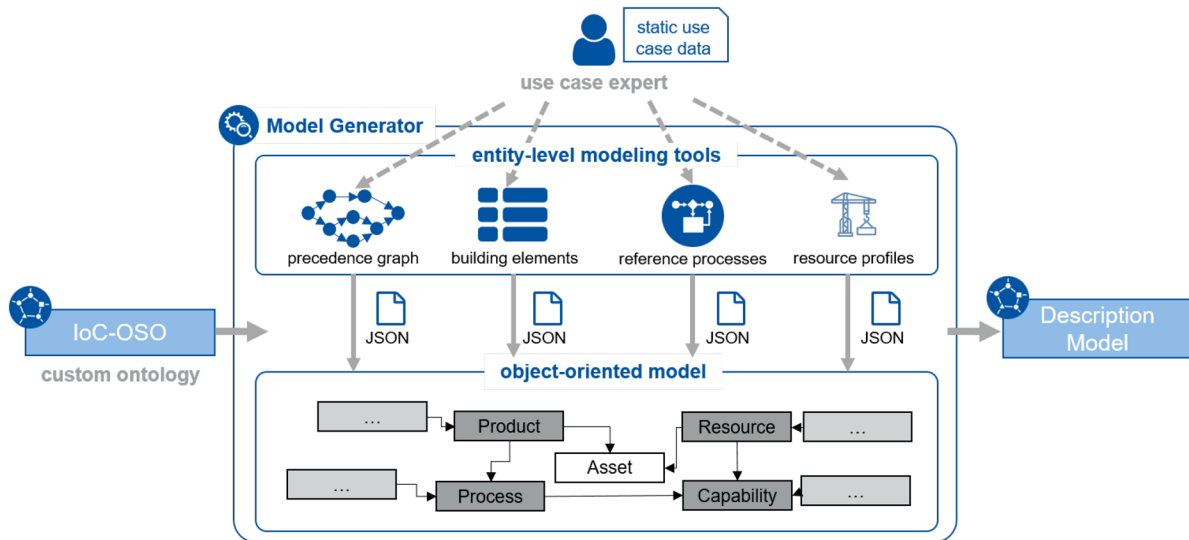


Fig. 6 Model Generator framework to convert static use case data into a standardized description model according to the custom ontology

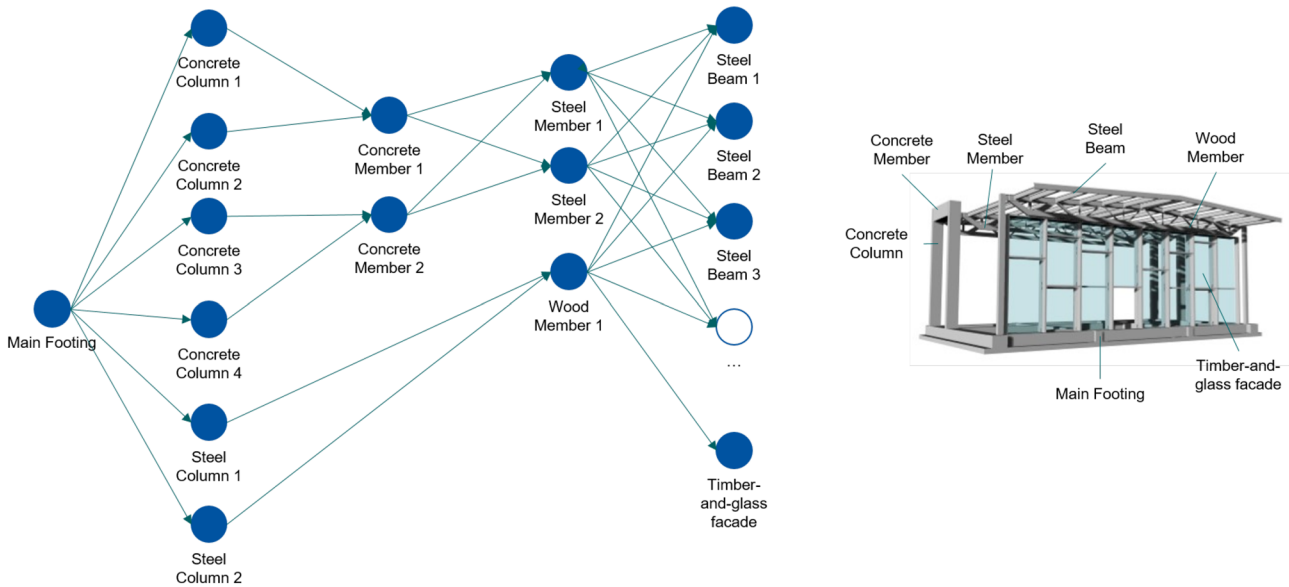


Fig. 7 Excerpt of the precedence graph of the bus station use case

(2) The building element list contains semantic information of the building element instances. The semantic information is needed to link the building element characteristics (e.g., weight and volume) to process data. These values can be extracted directly from IFC files utilizing the IfcOpenShell library and stored in JSON files subsequently. An excerpt for element “Concrete Column 1” is illustrated in

Fig. 8. The entries of the table conform with the IoC-OSO ontology.

(3) Predefined reference processes define the tasks to be executed for each building element. Linking the building element with the reference process results in parametric process descriptions, considering the semantic information of each individual building element. Reference processes can be modeled as Business Process Modeling Notation (BPMN)

Header				
id	bfba99ac57904a3cb8716b21e5506f06			
type	IfcColumn			
name	Concrete Column 1			
Body				
parameters				
id	type	name	value	dimension
PAR_weight	Float	weight	1400	kg
PAR_volume	Float	volume	0.525	m ³
variables				
id	type	name	value	dimension
VAR_state	String	state	IDLE	-

Fig. 8 Exemplary representation of a building element, illustrated as table

Header				
id	78dd8b9e5d5b4f71ae82980c9123c8f7			
type	TR_Crane			
name	Towercrane L1-24			
Body				
parameters				
id	type	name	valueRange	dimension
PAR_weight	Float	lift capacity at max. R	[0, 950]	kg
PAR_max_height	Float	hook height	[0, 19.1]	m
PAR_max_radius	Float	max. radius	[0, 25]	m
variables				
id	type	name	value	dimension
VAR_state	String	state	IDLE	-

Fig. 9 Exemplary representation of a resource profile for a tower crane, illustrated as table

Header				
id	04229d8896464847a7c2c4e11b45ad7f			
type	TR_Robot			
name	KR 120 R3900			
Body				
parameters				
id	type	name	valueRange	dimension
PAR_weight	Float	rated payload	[0, 120]	kg
PAR_max_radius	Float	max. radius	[0, 3901]	mm
PAR_num_axes	Int	number of axes	[1, 6]	-
variables				
id	type	name	value	dimension
VAR_state	String	state	IDLE	-

Fig. 10 Exemplary representation of a resource profile for an industrial robot, illustrated as table

diagrams. The XML-based BPMN files can be translated into JSON files, securing the correct format according to the IoC-OSO. Reference processes are currently mapped to building elements by IfcTypeObject (e.g., IfcMember

or IfcWall), material (e.g., concrete or steel) and the corresponding construction method (e.g., in-situ or pre-fabrication). Figure 11 illustrates an exemplary reference process for the prefabricated IfcMember “Steel Member 1” from the given use case.

(4) Resource profiles contain all relevant information for describing the resource for capability-based resource allocation problems. Relevant parameters could be, e.g., payload, capacity, availability or boundary conditions. This information is stored using the JSON format. A web-based JSON editor form can be used to compile machine-readable resource models. Figure 9 illustrates an excerpt of a tower crane and Figure 10 of an industrial robot resource representation. For both resources, lift capacity as a function of the radius is not considered yet.

5.2 Object oriented model

According to the IoC-OSO, a connected description model is automatically derived using the entity-level machine-readable models of the static use case data. The generation of the connected AAS description model is implemented utilizing object-oriented programming (OOP). The IoC-OSO is predefining the AAS classes as well as the links between corresponding classes. The static use case data defines the parameters and preferences of the objects, instantiating the AAS classes.

The OOP approach allows predefining relevant associations and methods for the entities and subordinate concepts. These associations and methods are crucial to enable the algorithmic planning and control applications built upon the description model. Figure 12 shows a simplified UML representation of the implemented class structure with the most relevant predefined associations and methods. One of the most important aspects is the mapping between building elements and resources via required and provided capabilities (as shown in Figure 4). The association of the action class and the resource class to the capability class realizes this mapping. In both cases, the capability class has a method to get the defining parameters from its associated classes. Thus, the comparison of required parameters (defined by the building element) and the provided parameters (defined by the resources) can be accessed and compared.

6 Exemplary deployment of the DT for resource allocation

With the given description model, the DT can be deployed automatically. This DT can be applied for capability-based online resource allocation problems. In this section, an

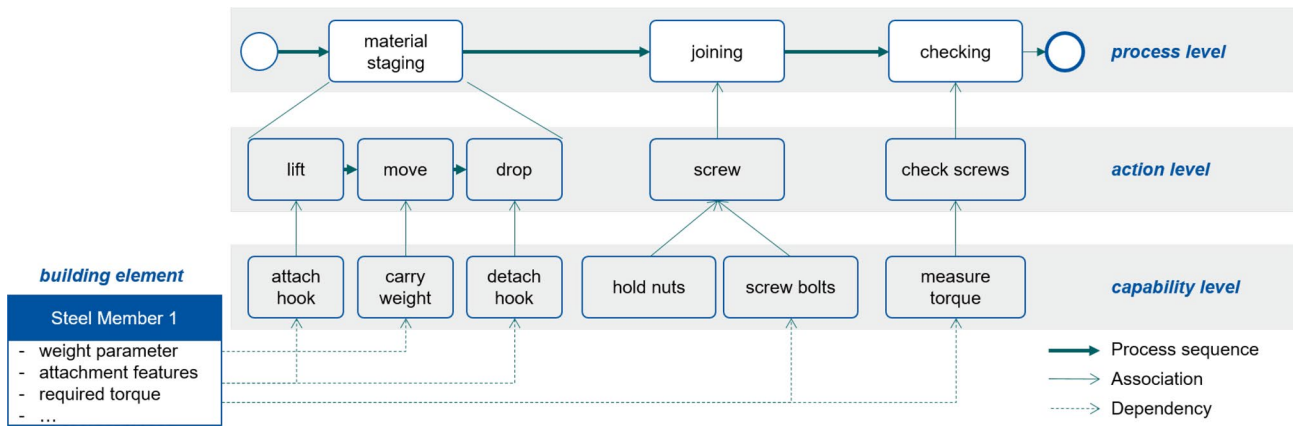


Fig. 11 Exemplary reference process for prefabricated steel I-beam member including capability definition

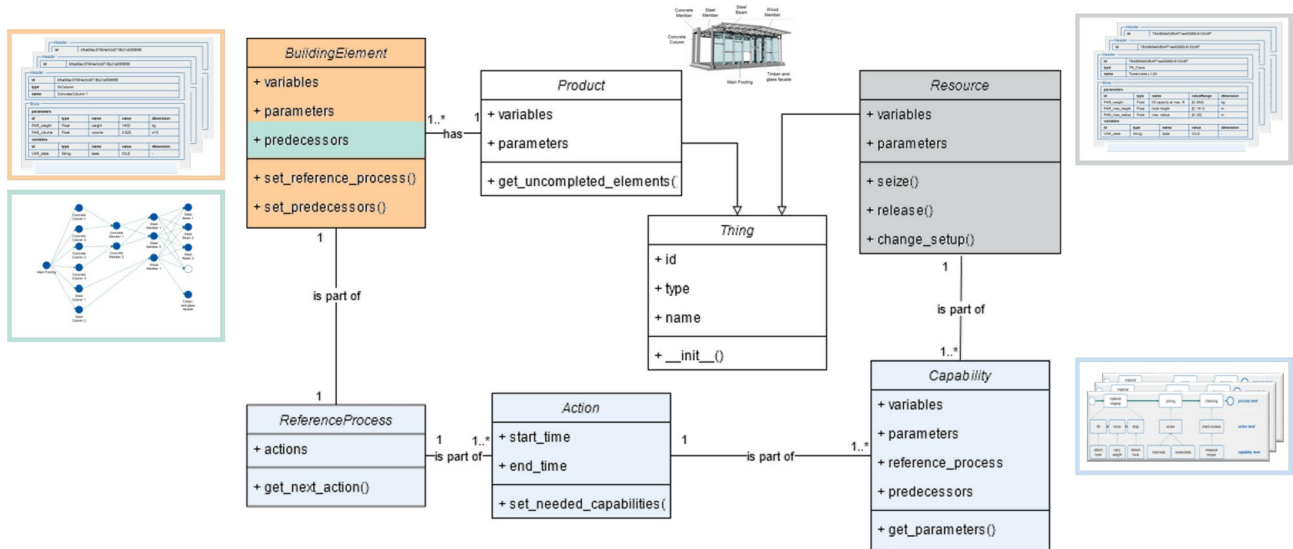


Fig. 12 Simplified UML representation of the IoC-OSO implementation

automated feasibility check is demonstrated based on the example introduced in Sect. 5. For that, a JSON parser is implemented to initiate Python objects from the entity-level JSON files. The JSON-based exchange formats on entity-level support rapid integration with preferred communication protocols (e.g., HTTP, MQTT or OPC UA). Topic and message structures, as well as authentication methods, can be predefined for each object. With predefined protocol components, setting up the bidirectional communication can be automated.

For capability-based feasibility check algorithm follows the logic of the pseudo-code in Algorithm 1. Firstly, all

idle building elements according to the precedence graph are identified. For each building element, the next action of the reference process is identified. The needed capabilities of the next action are compared with the offered capabilities of all resources. On the parameter level, it is checked whether the needed parameter values of each capability are inside the offered parameter value ranges. Only the same capability types are comparable and build a match if all parameters are fulfilled by one resource. The feasibility check algorithm returns a list of all feasible resources that can be used to execute the next action of the requested building element.

Algorithm 1: FeasibilityCheck

```

Input : precedence graph  $PG$ , building
          element list  $BE$ , resource profiles  $R$ ,
          reference processes  $RP$ 
Output: feasible resource allocations
 $be_i \leftarrow getBuildingElements(PG, BE)$ ;
Initialise feasibleResourceList for  $be_i$ ;
 $a_{ij} \leftarrow getCurrentActions(be_i)$ ;
 $c_{ijk} \leftarrow getRequiredCapabilities(a_{ij})$ ;
 $r_m \leftarrow getAvailableResources(R)$ ;
 $c_{mn} \leftarrow getOfferedCapabilities(r_m)$ ;
 $p_{mno} \leftarrow getOfferedParameterRanges(c_{mn})$ ;
foreach capabilities  $c_{ijk}$  &  $c_{mn}$  do
  if  $c_{ijk} == c_{mn}$  then
    foreach parameter  $p_{ijkl}$  of  $c_{ijk}$  do
      if  $p_{ijkl}$  in range of  $p_{mno}$  then
        add  $r_m$  to feasibleResourceList
          of  $be_i$ 
      end
    end
  end
end

```

The feasibility check for the building element is illustrated in Fig. 13 for the previously shown “Concrete Column 1”. The building element is connected to the reference process /RP_IFC_COLUMN, which consists of the four actions Transportation, Form, Reinforce and Concrete. These actions need one or more capabilities, e.g., the /

PositioningCapability and the /AssemblyCapability for the action /RP_IFC_COLUMN/FORM. Each needed capability has one or more quantified parameters, which need to be coaligned with the parameter range of the offered capabilities. For example, all three parameters of the needed /PositionCapability can be fulfilled by the offered Crane_L1-24/s001/PositioningCapability. Thus, the resource Crane_L1-24 is feasible to be allocated to the action bfba.../RP_IFC_COLUMN/FORM/.

For the deployment of the DT to realize the online capability-based feasibility check, the overall network framework is illustrated in Fig. 14. The feasibility check algorithm is embedded in a software application that is connected to an MQTT Broker and subscribes to the AAS of resources and building elements. With this architecture relevant data (e.g., current setup of the robot) can be accessed online. The AAS can be directly connected to its corresponding physical asset. For this AAS-asset connection, any protocol could be used (e.g., TCP/IP via Kuka Ethernet RSI). Implementing communication interfaces to real resources requires intense engineering. Enabling building elements to communicate is an open issue to clarify. Therefore, simulated resources and building elements are used for demonstration purposes. The simulation mainly executes status updates on the resources and building elements AAS. Status updates of the building element AAS or resource AAS (e.g., breakdown of one machine) triggers a new feasibility check through the Publish-Subscribe-Pattern of MQTT. Bidirectional communication between the application and the AAS instances is also realized via the Publish-Subscribe-Pattern of MQTT.

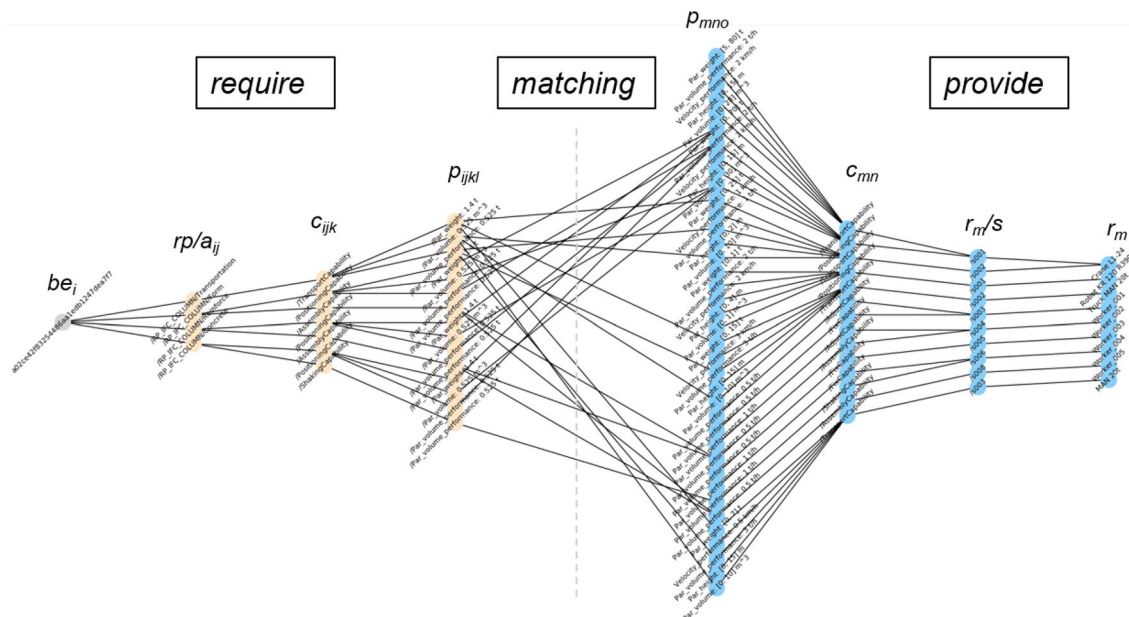


Fig. 13 Exemplary feasibility check result from Algorithm 1 for one building element and multiple resources, with: be = building element, rp = reference process, a = activity, c = capability, p = parameter, s = setup, r = resource

For example, the robot can subscribe to the topic “allocated_resources” of all processes. So the robot gets notified as soon as it is allocated to a specific task. Internal routines of the robot can then schedule the corresponding tasks on its local machine.

7 Conclusion and outlook

The resulting ontology-based DT model has shown high potential for automating capability-based resource allocations. This work contributes to the lack of capability-based modeling of resources and processes in the construction domain by developing an ontology-based DT. The Internet of Construction On-Site Ontology (IoC-OSO) was derived by combining and enhancing well-established domain ontologies (e.g., DOCK1.0, MSO and IFC), enhanced with meta-model approaches from AAS. Machine-readable AAS instances are generated using the IoC-OSO and standardized modeling tools. A prototypical deployment was shown in a Python environment with simulated assets. The feasibility check algorithm has been demonstrated with typical construction resources (e.g., crane) and robotic resources.

The algorithm utilizes the data from the DT through the MQTT network for an online capability-based resource allocation. It is demonstrated that the algorithm can unveil potential resource allocations for robotic applications. Thus the results contribute to identifying new robot applications for on-site construction. The developed modular structure of the information model enables a flexible adoption to other specific use cases with different requirements regarding process sequences and information details. It also allows the combination of manual and autonomous resources on-site, enabling the coexistence of both types in the real and digital world by providing a generic structure.

The effort to build connections of the AAS to its real physical assets was not justified for this proof of concept. Not all considered assets are natively able to automatically provide the needed digital data (e.g., building elements). Machinery can transmit telemetric data, but conversion to the interfaces defined within this work still requires highly individual implementation effort. Further research needs to be conducted on efficiently connecting real building elements and resource assets to their corresponding AAS to fully integrate them into the digital world. The shown feasibility check algorithm serves as a preliminary step for

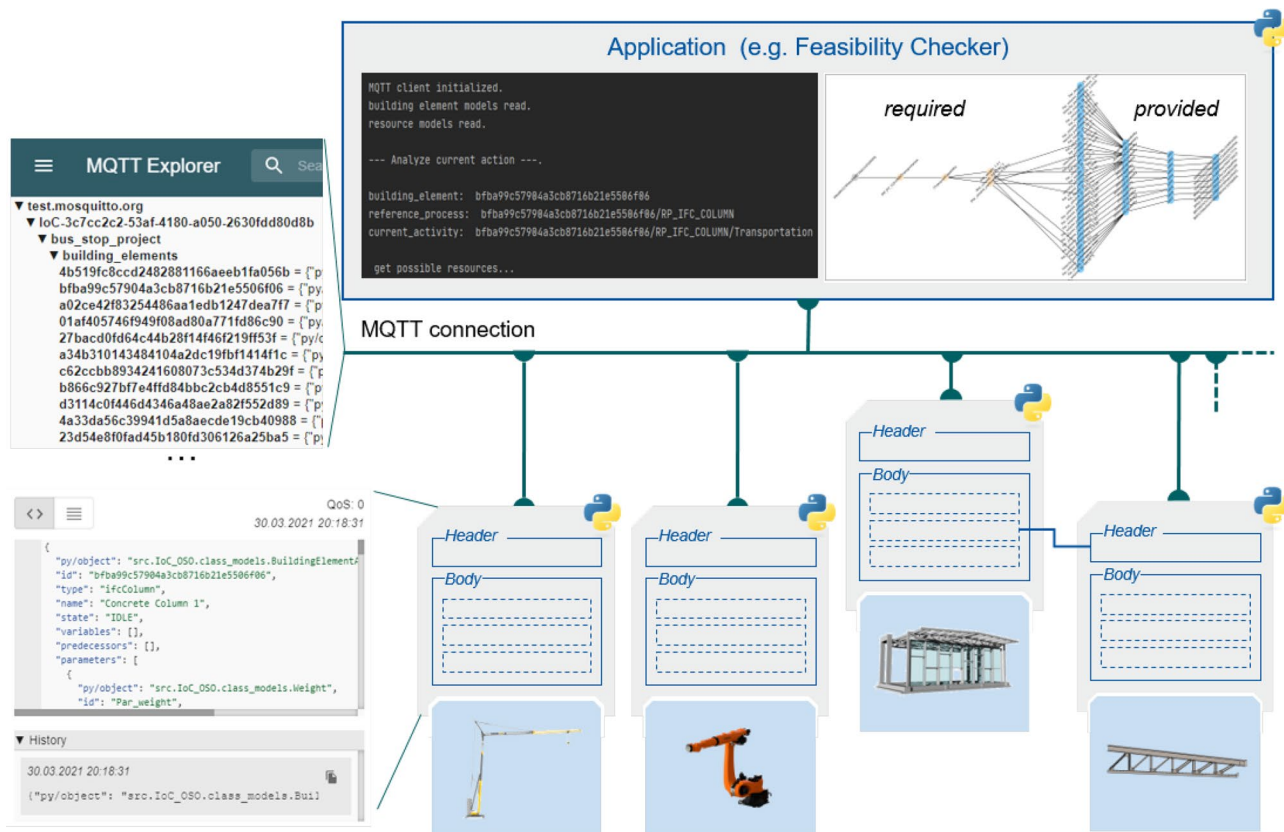


Fig. 14 Framework for the deployed DT model for capability-based online feasibility checks; The feasibility check application is connected via MQTT communication to the building element AAS and resource AAS (illustration based on Boss et al. 2020)

capability-based resource allocation. Further applications need to be developed to fully enhance the value of the DT model for on-site construction scheduling applications. For planning purposes, an optimal schedule can be derived by orchestrating building elements, corresponding construction processes and resources. The needed information for control applications can be accessed by the AAS instances and resulting tasks by the optimal schedule can be pushed back. Furthermore, the DT model could utilize construction progress data to dynamically adjust the schedule according to deviations from the initial schedule. The addition of these applications might result in an extension of the IoC-OSO by other ontologies due to slight use case changes and changes of structure in static use case data. Further research will be conducted in developing the above-mentioned planning and controlling applications based on the IoC-OSO. Furthermore, enhanced tools to enable use case experts to model the static use case data are planned. Automatic generation of precedence graphs based on IFC models is still an open research topic. Finally, to enhance the technology readiness level of the DT-based technologies for construction, stable and multi-purposed, on-site network technologies need to be studied.

Funding Open Access funding enabled and organized by Projekt DEAL. This work is part of the research project “Internet of Construction” that is funded by the Federal Ministry of Education and Research of Germany within the indirective on a joint funding initiative in the field of innovation for production, services and labor. The authors are responsible for the content. The following Git repository provides exemplary schemata and data models for reference and understanding: <https://github.com/Internet-of-Construction/IoC-OSO>

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Al-Sehrawy R, Kumar B (2021) Digital twins in architecture, engineering, construction and operations. a brief review and analysis. In: E. Toledo Santos, S. Scheer (eds.) Proceedings of the 18th International Conference on Computing in Civil and Building Engineering, *Lecture Notes in Civil Engineering*, vol. 98, pp. 924–939. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-51295-8_64
- Ala-Risku T, Kärkkäinen M (2006) Material delivery problems in construction projects: a possible solution. *Int J Prod Econ* 104(1):19–29
- Bader S, Barnstedt E, Bedenbender H, et al. (2020) Details of the asset administration shell: Part 1 - the exchange of information between partners in the value chain of industrie 4.0. https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/Details_of_the_Asset_Administration_Shell_Part1_V3.html
- Balaji B, Bhattacharya A, Fierro G, Gao J, Gluck J, Hong D, Johansen A, Koh J, Ploennigs J, Agarwal Y, Berges M, Culler D, Gupta R, Kjærgaard MB, Srivastava M, Whitehouse K (2016) Brick. In: Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments, pp. 41–50. ACM, New York, NY, USA. <https://doi.org/10.1145/2993422.2993577>
- Baling M, Katranuschkov P, Schapke SE, Scherer RJ (2010) Semi-automatic process generation with reusable process and knowledge modules. In: 2010 IEEE international technology management conference (ICE), pp. 1–8. IEEE
- Bao Q, Zhao G, Yu Y, Dai S, Wang W (2020) Ontology-based modeling of part digital twin oriented to assembly. In: Proceedings of the institution of mechanical engineers, part B: Journal of Engineering Manufacture. <https://doi.org/10.1177/0954405420941160>
- Benevolenskiz A (2016) Ontology-based modeling and configuration of construction processes using process patterns
- Benevolenskiy A, Roos K, Katranuschkov P, Scherer RJ (2012) Construction processes configuration using process patterns. *Adv Eng Inform* 26(4):727–736. <https://doi.org/10.1016/j.aei.2012.04.003>
- Berners-Lee T, Hendler J, Lassila O (2001) The semantic web. *Scientific american* 284(5):34–43
- Bodenbenner M, Sanders MP, Montavon B, Schmitt RH (2021) Domain-specific language for sensors in the internet of production. In: Behrens BA, Brosius A, Hintze W, Ihlenfeldt S, Wulfsberg JP (eds) Production at the leading edge of technology. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 448–456
- Boje C, Guerriero A, Kubicki S, Rezgui Y (2020) Towards a semantic construction digital twin: Directions for future research. *Automation in Construction* 114. <https://doi.org/10.1016/j.autcon.2020.103179>
- Boss B, Malakuti S, Lin SW, Usländer T, Clauer E, Hoffmeister M, Stojanovic L (2020) Digital twin and asset administration shell concepts and application in the industrial internet and industrie 4.0: An industrial internet consortium and platform industrie 4.0 joint whitepaper. <https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/Digital-Twin-and-Asset-Administration-Shell-Concepts.html>
- Bradley A, Li H, Lark R, Dunn S (2016) Bim for infrastructure: an overall review and constructor perspective. *Auto Construction* 71:139–152. <https://doi.org/10.1016/j.autcon.2016.08.019>
- BuildingSmart: Ifcresource (2021). <https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD1/HTML/schema/ifckernel/lexical/ifcresource.htm>
- Bunte A, Niagemann O, Stein B (2018) Integrating owl ontologies for smart services into automationml and opc ua. In: 2018 IEEE 23rd international conference on emerging technologies and factory automation (ETFA), pp. 1383–1390. IEEE. <https://doi.org/10.1109/ETFA.2018.8502593>
- Daniele L, den Hartog F, Roes J (2015) Created in close interaction with the industry: The smart appliances reference (saref) ontology. In: R. Cuel, R. Young (eds.) Formal Ontologies Meet Industry, *Lecture Notes in Business Information Processing*, vol. 225, pp. 100–112. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-21545-7_9
- Delgado F, Martínez-González MM, Finat J (2013) An evaluation of ontology matching techniques on geospatial ontologies. *Int J Geograph Inform Sci* 27(12):2279–2301. <https://doi.org/10.1080/13658816.2013.812215>

- El-Diraby TE (2013) Domain ontology for construction knowledge. *J Construction Eng Manag* 139(7):768–784
- El-Diraby TE, Kashif KF (2005) Distributed ontology architecture for knowledge management in highway construction. *J Construction Eng Manag* 131(5):591–603. [https://doi.org/10.1061/\(ASCE\)0733-9364\(2005\)131:5\(591\)](https://doi.org/10.1061/(ASCE)0733-9364(2005)131:5(591))
- El-Diraby TE, Osman H (2011) A domain ontology for construction concepts in urban infrastructure products. *Auto Construction* 20(8):1120–1132
- El-Gohary NM, El-Diraby TE (2010) Domain ontology for processes in infrastructure and construction. *J Construction Eng Manag* 136(7):730–744
- Fadel FG, Fox MS, Gruninger M (1993) A generic enterprise resource ontology. In: Proceedings of 3rd IEEE workshop on enabling technologies: infrastructure for collaborative enterprises, pp. 117–128. IEEE
- Fumagalli L, Pala S, Garetti M, Negri E (2014) Ontology-based modeling of manufacturing and logistics systems for a new mes architecture. In: E. Bayro-Corrochano, E. Hancock (eds.) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, *Lecture Notes in Computer Science*, vol. 8827, pp. 192–200. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-662-44739-0_24
- Getuli V, Capone P (2019) Ontology-based modeling for construction site planning: towards an ifcowl semantic enrichment
- Göppert A, Grahn L, Rachner J, Grunert D, Hort S, Schmitt RH (2021) Pipeline for ontology-based modeling and automated deployment of digital twins for planning and control of manufacturing systems. *J Intell Manuf*. <https://doi.org/10.1007/s10845-021-01860-6>
- Grieves M, Vickers J (2017) Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: Transdisciplinary perspectives on complex systems, pp. 85–113. Springer
- Gruber TR (1993) A translation approach to portable ontology specifications. *Knowl Acquisition* 5(2):199–220
- Hepp M (2005) eclassowl: A fully-fledged products and services ontology in owl. Poster Proceedings of ISWC2005. Galway
- Hildebrandt C, Köcher A, Küstner C, López-Enríquez CM, Müller AW, Caesar B, Gundlach CS, Fay A (2020) Ontology building for cyber-physical systems: application in the manufacturing domain. *IEEE Trans Auto Sci Eng* 17(3):1266–1282. <https://doi.org/10.1109/TASE.2020.2991777>
- IEC 62832-2: Industrial-process measurement, control and automation—digital factory framework - part 2: Model elements (2020-10)
- Industrial Internet Consortium: The industrial internet of things: Volume g1: Reference architecture (2019). <https://www.iiconsortium.org/IIRA.htm>
- ISO: Information container for linked document delivery—exchange specification—part 1: Container (2020-04)
- Wan J, Yin B, Li D, Celesti A, Tao F, Hua Q (2018) An ontology-based resource reconfiguration method for manufacturing cyber-physical systems. *IEEE/ASME Trans Mech* 23(6):2537–2546. <https://doi.org/10.1109/TMECH.2018.2814784>
- Järvenpää E, Siltala N, Hylli O, Lanz M (2019) The development of an ontology for describing the capabilities of manufacturing resources. *J Intell Manuf* 30(2):959–978. <https://doi.org/10.1007/s10845-018-1427-6>
- Joshi R, Mellor S, Didier P (2018) The industrial internet of things: Volume g5: Connectivity frameworks. https://www.iiconsortium.org/pdf/IIC_PUB_G5_V1.01_PB_20180228.pdf
- Kluge S (2011) Methodology for capability-based planning of modular assembly systems. OPUS, Stuttgart. <https://doi.org/10.18419/opus-4444>
- Kousi N, Gkournelos C, Aivaliotis S, Giannoulis C, Michalos G, Makris S (2019) Digital twin for adaptation of robots' behavior in flexible robotic assembly lines. *Proc Manuf* 28:121–126
- Kritzinger W, Karner M, Traar G, Henjes J, Sihn W (2018) Digital twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine* 51(11):1016–1022. <https://doi.org/10.1016/j.ifacol.2018.08.474>
- Lemaignan S, Siadat A, Dantan JY, Semenenko A (15-16 June 2006) Mason: A proposal for an ontology of manufacturing domain. In: IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06), pp. 195–200. IEEE. <https://doi.org/10.1109/DIS.2006.48>
- Lima C, Diraby TE, Fies B, Zarli A, Ferneley E (2003) The e-cognos project: Current status and future directions of an ontology-enabled it solution infrastructure supporting knowledge management in construction. In: K.R. Molenaar, P.S. Chinowsky (eds.) Construction Research Congress, pp. 1–8. American Society of Civil Engineers, Reston, VA. [https://doi.org/10.1061/40671\(2003\)103](https://doi.org/10.1061/40671(2003)103)
- Liu C, Jiang P, Jiang W (2020) Web-based digital twin modeling and remote control of cyber-physical production systems. *Robot Comput Integrated Manuf* 64:101956. <https://doi.org/10.1016/j.rcim.2020.101956>
- Mascardi V, Cordi V, Rosso P (2007) A comparison of upper ontologies. In: Woa, pp. 55–64. Citeseer
- Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A (2003) Wonderweb deliverable d18-ontology library (final report). National Research Council-Institute of Cognitive Science and Technology
- Negri E, Perotti S, Fumagalli L, Marchet G, Garetti M (2017) Modeling internal logistics systems through ontologies. *Comput Ind* 88:19–34
- Noy NF, McGuinness DL (2001) Ontology development 101: a guide to creating your first ontology
- Park KT, Yang J, Noh SD (2020) Vredi: virtual representation for a digital twin application in a work-center-level asset administration shell. *J Intell Manuf* pp. 1–44
- Pauwels P, Poveda-Villalón M, Sicilia Á, Euzenat J (2018) Semantic technologies and interoperability in the built environment. *Semantic Web* 9(6):731–734. <https://doi.org/10.3233/SW-180321>
- Pauwels P, Terkaj W (2016) Express to owl for construction industry: towards a recommendable and usable ifcowl ontology. *Auto Construction* 63:100–133. <https://doi.org/10.1016/j.autcon.2015.12.003>
- Pauwels P, Zhang S, Lee YC (2017) Semantic web technologies in aec industry: a literature overview. *Auto Construction* 73:145–165. <https://doi.org/10.1016/j.autcon.2016.10.003>
- Pérez J, Arenas M, Gutierrez C (2009) Semantics and complexity of sparql. *ACM Trans. Database Syst.* 34(3). <https://doi.org/10.1145/1567274.1567278>
- Rasmussen MH, Lefrançois M, Schneider GF, Pauwels P (2020) Bot: The building topology ontology of the w3c linked building data group. *Semantic Web* 12(1):143–161. <https://doi.org/10.3233/SW-200385>
- Rasmussen MH, Pauwels P, Hviid CA, Karlshøj J (2017) Proposing a central aec ontology that allows for domain specific extensions. In: Lean and Computing in Construction Congress - Volume 1: Proceedings of the joint conference on computing in construction, pp. 237–244. Heriot-Watt University, Edinburgh. <https://doi.org/10.24928/JC3-2017/0153>
- Scherer RJ, Schapke SE (2014) Informationssysteme im Bauwesen 2: Anwendungen. Springer-Verlag
- Steinmetz C, Schroeder G, dos Santos Roque A, Pereira CE, Wagner C, Saalman P, Hellingrath B (2017) Ontology-driven iot code generation for fiware. In: 2017 IEEE 15th International Conference on Industrial Informatics (INDIN), pp. 38–43. IEEE
- Tang S, Shelden DR, Eastman CM, Pishdad-Bozorgi P, Gao X (2019) A review of building information modeling (bim) and the internet of things (iot) devices integration: present status and future trends. *Auto Construction* 101:127–139. <https://doi.org/10.1016/j.autcon.2019.01.020>

- Tantik E, Anderl R (2017) Integrated data model and structure for the asset administration shell in industrie 4.0. *Procedia CIRP* **60**. <https://doi.org/10.1016/j.procir.2017.01.048>
- Weser M, Bock J, Schmitt S, Perzylo A, Evers K (2020) An ontology-based metamodel for capability descriptions. In: 2020 25th IEEE international conference on emerging technologies and factory automation (ETFA), vol. 1, pp. 1679–1686. IEEE
- Zeng N, Dichtl M, König M (2017) A scenario-based simulation framework of on- and off-site construction logistics. In: 2017 Winter Simulation Conference (WSC), pp. 2348–2359. IEEE. <https://doi.org/10.1109/WSC.2017.8247965>
- Zhong B, Wu H, Li H, Sepasgozar S, Luo H, He L (2019) A scientometric analysis and critical review of construction related ontology research. *Auto Construction* 101:17–31. <https://doi.org/10.1016/j.autcon.2018.12.013>
- ZVEI: Beispiel zur Verwaltungsschale der industrie 4.0-komponente: Basisteil (2016). https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2016/November/Beispiele_zur_Verwaltungsschale_der_Industrie_4.0-Komponente_-_Basis teil/Beispiele-Verwaltungsschale-Industrie-40-Komponente-White-Paper-Final.pdf

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.