Advanced Process Representation for Semi-Automated Linking between Construction Schedules and IFC files

Jonas Schlenger[1],*, André Borrmann[1]

[1]Chair of Computational Modeling and Simulation, Technical University of Munich, Arcisstraße 21, Munich, 80333, Germany

Abstract

This research paper addresses the challenges in exchanging project management information within the construction industry, particularly focusing on construction schedules. The current reliance on XML or spreadsheet-based files for schedule exchange leads to information loss and misinterpretation in commonly used project management software. Diverse scheduling practices across construction companies, software preferences and organizational structures contribute to a lack of standardization. Existing data schemata for schedule description lack detailed representation of the hierarchical structure of construction processes and their dependencies, limiting automated interpretation. To fill this gap, this paper introduces a minimal ontology designed for representing construction schedules' processes. A case study demonstrates the ontology's application in automated schedule interpretation, emphasizing its potential for automated linking between schedules and corresponding BIM models, reducing manual efforts, and enhancing overall efficiency. It concludes by discussing its broader implications in the Architecture, Engineering, Construction, and Operation (AECO) industry.

Keywords

Construction process, Work Breakdown Structure (WBS), Construction management, Ontology

1. Introduction

Navigating the complexities of exchanging project management information of construction projects presents several challenges that impede seamless collaboration and hinder the full potential of automation. Presently, schedule exchange is mainly based on XML or spreadsheet-based files. These files result from export functionalities of project management software tools. However, this translation process from proprietary data formats to open formats often leads to loss and misinterpretation of the exchanged information [1, 2]. One significant challenge arises from the diverse scheduling practices adopted by different construction companies. Each company applies its unique scheduling methodologies, leading to a lack of standardization across the industry. This diversity is fueled by a multitude of factors, including software preferences, organizational structures, project-specific considerations, varying levels of detail, used language, and the use of company-specific naming conventions [3, 4]. Consequently, the absence of an established standard for schedule exchange limits the potential for automation in the Architecture, Engineering, Construction, and Operation (AECO) industry.

While there are existing data schemata designed for schedule description, such as Industry Foundation Classes (IFC), Digital Construction Ontologies (DiCon), Digital Twin Construction Ontology (DTC), Construction Tasks Ontology (CTO), and Internet of Construction Ontology (IOC), they exhibit limitations that hinder their effectiveness [5, 6, 7, 8, 9]. Notably, these schemata lack the representation of detailed process dependencies and struggle with hierarchical structures suitable for automated interpretation. To address these shortcomings, there is a pressing need for explicitly representing semantic information about process decomposition criteria. This additional layer of information is crucial for comprehending the intricate hierarchical structures inherent in construction processes.

To fill this gap, the present paper will introduce a minimal ontology for representing processes of construction schedules. A case study is presented that uses the new process representation for automated schedule interpretation and the linking with a corresponding IFC model to demonstrate the

LDAC 2024: 12th Linked Data in Architecture and Construction Workshop, June 13–14, 2024, Bochum, Germany

*Corresponding author.

jonas.schlenger@tum.de (J. Schlenger); andre.borrmann@tum.de (A. Borrmann)
0000-0002-4875-5117 (J. Schlenger); 0000-0003-2088-7254 (A. Borrmann)

ontology's suitability and usefulness. This is only one of many possible applications for the ontology. Another advantage of using the ontology is that it facilitates coordination between contractors and subcontractors through structured schedule exchange. Due to the enhanced semantic information, the schedule can be interpreted automatically, reducing the need for extensive manual efforts, streamlining digital processes and improving the overall efficiency. Also, the semantic information about the processes can be used to check the schedule for consistency and completeness.

This research is structured as follows. Section 2 provides background information and elaborates on the identified research gap by introducing existing process-related schemata. Subsequently, in Section 3, the developed process ontology covering process dependencies and decomposition criteria is introduced. One possible application of the ontology is demonstrated in the case study in Section 4. The paper closes with a discussion and conclusion in Sections 5 and 6.

2. Background and state-of-the-art

2.1. Work breakdown structures

Planning complex construction projects involves addressing challenges such as long project duration, a multitude of stakeholders, and the unique nature of each project [10]. To navigate these complexities, professionals commonly employ Work Breakdown Structures (WBSs) to systematically deconstruct a construction project into smaller, more manageable pieces. The foundational high-level work packages, which form the core of the WBS, encompass all project-related tasks and are progressively dissected into finer-grained packages. Each decomposition must ensure comprehensive coverage of the parent task to maintain a holistic project perspective. WBSs typically adopt a product-oriented approach, allowing for distinct decomposition levels tailored to different project components. The optimal level of detail is achieved when further granularity cannot improve aspects like cost estimation accuracy or projected execution timelines. This occurs when the level of detail surpasses the scope of possible uncertainties [11, 12].

The WBS often reflects the organizational structure of the construction company. It should always be ensured that every work package can be assigned to a single responsible person. An essential principle here is to avoid consolidating tasks executed by multiple parties into a single work package, fostering unambiguous responsibility allocation. In this way, the WBS is influenced by the part of the subcontracted work and the company's hierarchical structure. Additionally, the WBS forms the basis for other breakdown structures, including resource, cost, and product breakdown structures, providing a comprehensive foundation for project management [13, 14, 15].

The WBS is an essential starting point for creating the construction schedule [11]. Here, the decomposition criteria are relevant to understanding the meaning of the resulting subtasks. Figure 1 provides an example of a WBS with explicit definition of the applied decomposition criteria. Being able to interpret the semantic meaning of the WBS will also significantly help interpret the derived construction schedule since it follows the same hierarchical structure.

2.2. Classification systems in construction

To gain a comprehensive understanding of potential decomposition criteria, existing classification systems within the construction field offer an extensive overview of construction-related concepts. They divide construction concepts into different categories to create a common structure and comparability across projects. There are many nationally and internationally applied classification systems which are similar but yet different from each other. Some national classifications are, e.g., MasterFormat, Uniformat, OmniClass, and CoClass, while ISO 12006-2 and ISO 81346-12 are international examples [12]. Looking only at the topmost classification level, ISO 12006-2 for example is divided into twelve categories: construction information, construction products, construction agents, construction aids, management, construction process, construction complexes, construction entities, built spaces, construction elements, work results, and construction properties [16]. While these classification systems are commonly used

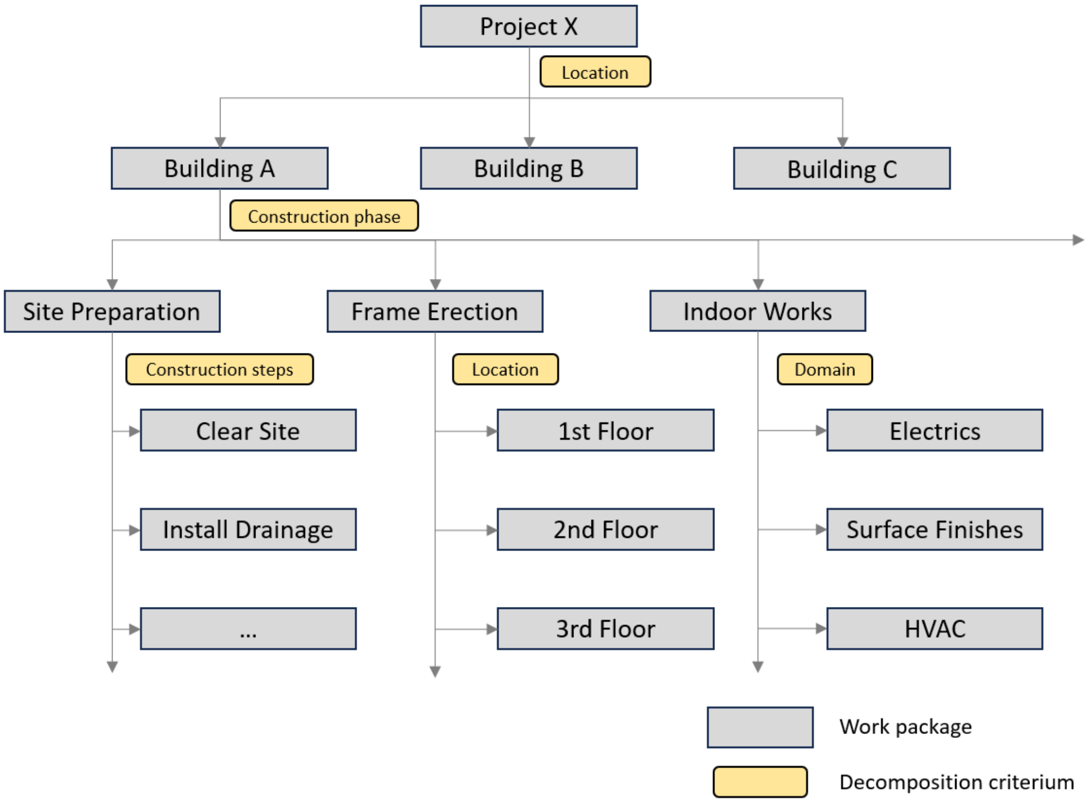


Figure 1: WBS of an example project with the applied decomposition criteria (inspired by [13]).

for creating Cost Breakdown Structures (CBSs), they can also be applied for WBSs since, in many cases, it does not make a conceptual difference if a construction project is decomposed into cost or process-related items. As an example, a work package could be divided into subtasks by differentiating between different construction elements according to ISO 12006-2. While the range of subtasks that can be created with a single decomposition may seem limited, a sequence of divisions as present in a WBS allows representing a wide range of processes, chaining the breakdown into gradually increasing detail. An overview of several classification systems compiled by Cerezo-Narváez et al. [12] is given Table 1. It takes ISO 12006-2 as the starting point and compares its categories with categories from other classification systems.

2.3. State-of-the-art data schemata

Several existing data schemata allow for representing construction processes. Some of them are introduced and discussed here to present their suitability for schedule exchange and automated schedule interpretation.

The Industry Foundation Classes (IFC), maintained by buildingSMART International [5], are an internationally recognised standard for the storage and exchange of construction-related data. Encompassing diverse concepts across all stages of a building's life cycle, it is renowned for its extensive list of geometric representations, such as boundary representations and procedural descriptions. Despite its emphasis on spatial and structural elements in building design, IFC also addresses semantic aspects and incorporates processes-related classes.

Construction processes can be represented with the class `IfcTask`. To form hierarchical structures of processes which are relevant for describing a complete construction schedule, `IfcTasks` can be nested infinitely. Moreover, process dependencies can be described through `IfcRelSequence`, which defines several types of sequence types like start-to-start and end-to-start. An example of using these entities in combination is shown in Figure 2. Finally, the internal sequence of a task can be described

Table 1
Comparison between several construction classification system [12].

ISO 12006-2	ISO 81346-12	OmniClass	CoClass	CCS	UniClass
Information		Information		Documents	Forms
Products	Components	Products Materials	Components	Components	Products
Agents		Disciplines Roles		Documents	Agents
Aids		Tools		Equipment	Tools Equipment
Management		Services		Documents	Project Management
Processes		Phases		Documents	Phases
Complexes			Complexes		Complexes
Entities		By Functions By Forms	Entities	Entities	Entities Activities
Built Spaces	Spaces	By Functions By Forms	Spaces	Built Spaces User Spaces	Spaces Locations
Elements	By Function By Technics	Elements	By Functions By Technics	By Functions By Technics	Functions Systems
Work Results		Work Results	Production		
Properties		Properties	Properties Landscape	Classes	Properties CAD

by using `IfcProcedure` and `IfcEvent` [5]. Since they do not allow for specifying concrete time intervals, their usefulness in the context of construction schedules is very limited.

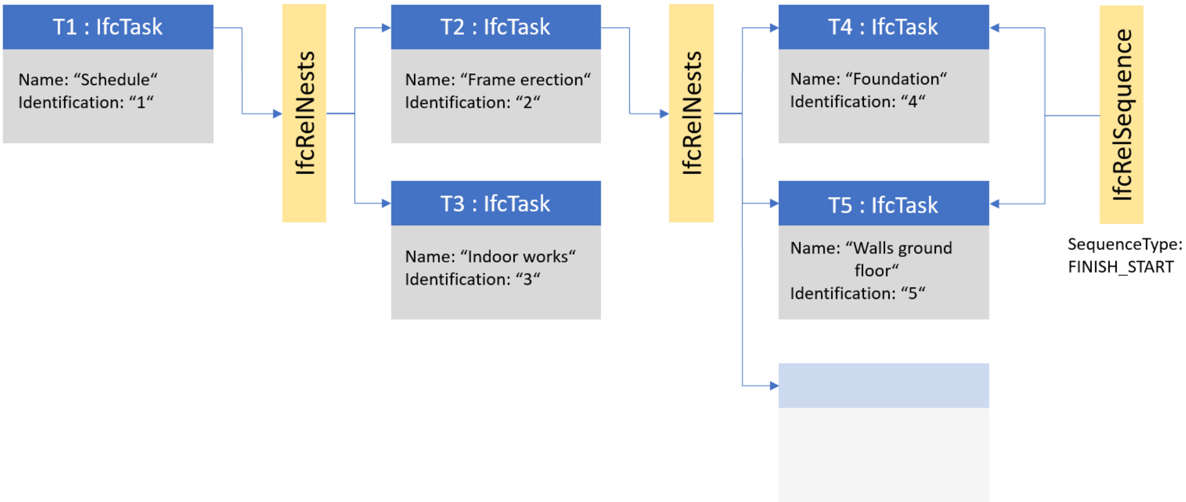


Figure 2: Representation of construction processes according to IFC [5].

Several aspects hinder the use of IFC as a means to schedule exchange and automated interpretation. The Model View Definition (MVD) concept is supposed to reduce the number of entities to be supported by the implementing software vendor according to a well-defined use case. However, MVDs do not yet exist for pure process models or combined BIM-process models (4D models). This creates a large overhead for an implementer who aims only at the process-related entities [17]. Furthermore, the process part of IFC is barely supported by existing software tools. While BIM authoring tools that support IFC are fully focused on the building structure, dedicated project management software is used for scheduling-related tasks, which do not support IFC. Lastly, the use of `IfcTask` makes an automated interpretation of processes very difficult since high-level processes like the complete frame erection phase and a very detailed process like pouring concrete for a specific column are all represented in the same way [5]. Additional information is required to convey the meaning of a particular process, which cannot be adequately represented in IFC.

The Digital Construction Ontologies (DiCon) are a set of interrelated ontologies developed by Zheng et al. [6]. They are covering the management and execution aspects of construction projects. DiCon describes construction processes from a flow perspective. All processes can be assigned with several input or output flows. This covers, e.g. the required input material or equipment and the results like building elements that emerge from the conversion process. Similar to IFC, DiCon also defines only a single type of process, which is called activity. However, the relationship dice:*hasSubActivity* can be used to define process hierarchies (see Figure 3). This relationship is the parent relationship of three additional relationships: *hasLocationPhase*, *hasObjectPhase*, and *hasProcessPhase*. It allows differentiating between location, object, or process step-specific sub-activities [6]. While this allows for the definition of additional semantics related to the work breakdown structure, it does not provide a complete list of possible decomposition criteria. For example, the decomposition of indoor works by trade, like plumbing and painting, cannot be represented with DiCon. Moreover, DiCon is missing different types of process dependencies, like start-to-start and end-to-start. The activity class inherits from the occurrent class of the Basic Formal Ontology (BFO), which only allows defining processes predecessors without specifying additional information. These dependencies are an essential aspect of construction schedules; e.g., they are clearly defined in conventional Gantt charts.

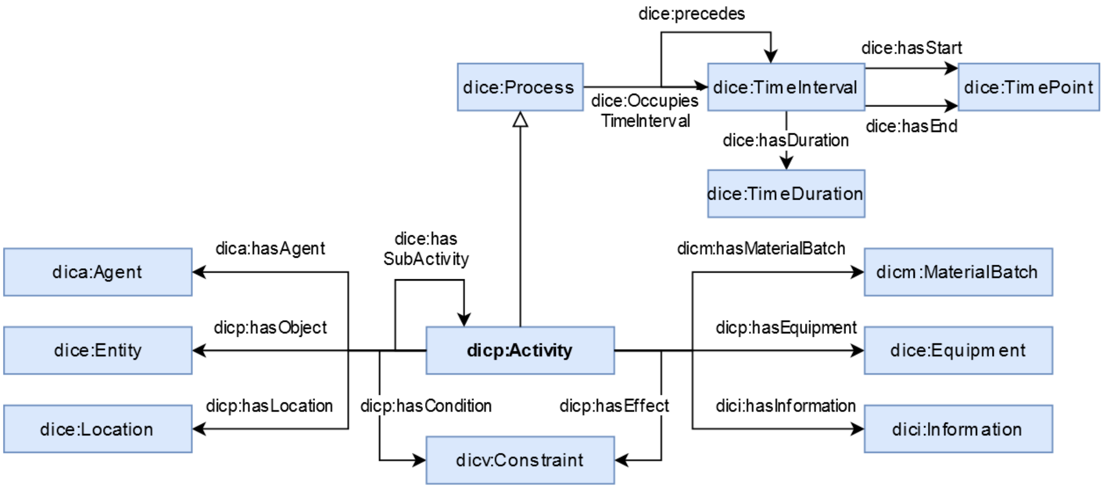


Figure 3: Representation of processes according to DiCon [6].

The Construction Tasks Ontology (CTO) has similar limitations. CTO is a minimal ontology that describes various construction tasks, including installation, modification, removal, inspection, and reparation. These tasks are linked to their target entity through the relationship *cto:isSubjectOfTask*. While the ontology was mainly designed for the maintenance of historical buildings, it is still applicable to other construction-related scenarios. Like DiCon, however, CTO can only define task successors with the relationship *cto:afterFinishedTask*, which does not consider different types of process sequences [9].

The Digital Twin Construction Ontology (DTC) is an ontology developed in the frame of the EU Horizon 2020 project BIM2TWIN[1]. It focuses on describing a digital twin of the construction phase. The representation of process dependencies is handled similarly to IFC with the so-called `ProcessPrecondition`. All four types of sequences are defined as individuals of this class. Regarding the representation of decomposition criteria, three different types of processes are defined. These are `WorkPackage`, `Activity`, and `Task` (see Figure 4). They resemble the decomposition by production method, construction step and building element and must be applied in this specific order [8]. While many schedules can be converted to this structure, it limits its flexibility and may struggle to represent

[1] https://bim2twin.eu/

some types of schedules. For example, a schedule can contain more than three hierarchical levels. To represent such a schedule, the most abstract levels would not be represented explicitly anymore. Another example would be a schedule that applies the process decomposition in a different order, which again creates additional effort in adapting the schedule to the structure defined by the ontology.

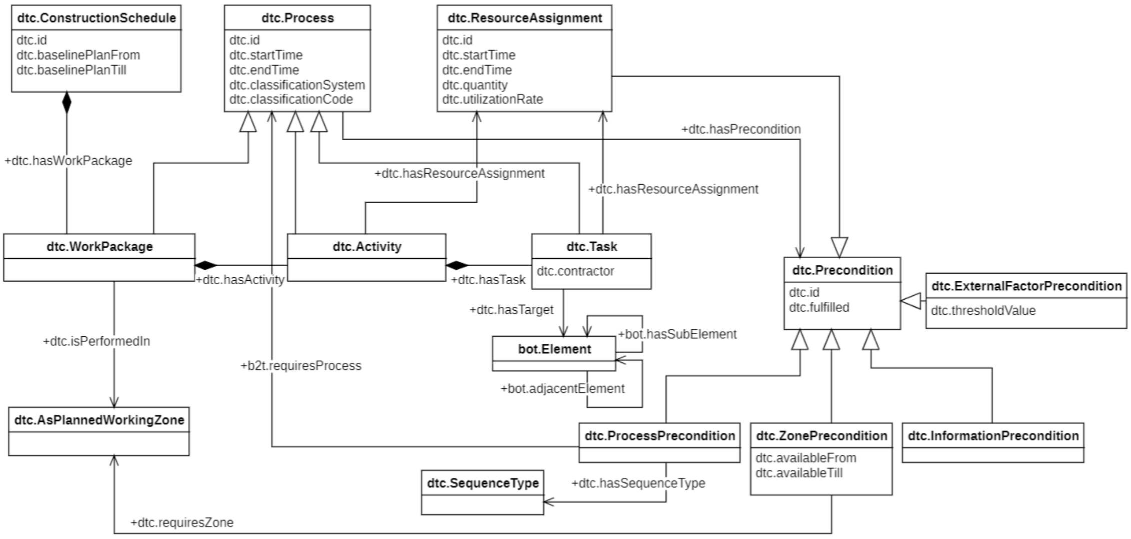


Figure 4: Representation of processes according to DTC [8].

Finally, the Internet of Construction Process Ontology (IOC) is a recent and still ongoing development. It is dedicated to the definition of construction processes with a wide range of involved objects and persons. Furthermore, it defines various types of process states [7]. Like IFC, it also lacks a definition of the decomposition criteria of processes and detailed process dependencies.

All of the introduced schemata for process description are ontologies that can be used in the context of the Semantic Web, initiated by Berners-Lee et al. [18]. While IFC is primarily specified by means of the STEP schema language EXPRESS, Beetz et al. [19] created a Resource Description Framework (RDF)-compliant version called ifcOWL. Ontologies have the advantage that they can be flexibly combined to create a more complex and complete data schema for a given use case. Furthermore, developers can use a wide range of freely available RDF tools for validation, querying, analysis and many other purposes. Moreover, ontologies can be easily combined with other existing ontologies. This allows focusing on small, concise ontologies that can reuse other ontologies for more complex use cases [20]. This is why the Semantic Web approach was chosen to tackle the semantic description and exchange of construction processes.

3. Advanced schedule representation

Based on the research gap presented in Section 2.3, an ontology for the representation of construction schedules was developed. Particular focus is dedicated to different types of process decomposition criteria that are commonly used. For this, the process hierarchies of schedules of various construction projects are analysed. While in this paper, a new ontology for schedule representation is proposed, an alternative option would be to integrate the newly defined classes for process preconditions and decomposition into existing ontologies. Considering the FAIR data-sharing principles, integration into existing ontologies is favoured [21]. However, it is not clear if some required changes to existing ontologies align with the creators' intentions. For this reason, a new ontology is presented here with the intention of future integration.

It is important to note that in the frame of this paper, only processes related to the construction execution are considered. The schedule of the planning phase is out of scope.

3.1. Process decomposition criteria

Process decomposition criteria play a crucial role in understanding the process hierarchy and dependencies between processes. They describe how the parent process is divided into smaller parts and give insights into the relationship between the different child processes. For example, a division of a process by storey creates a spatial dependency of the processes and, therefore, might also imply an order in which the child processes need to be executed. For this reason, explicitly describing these decomposition criteria provides knowledge essential for automated interpretation of the process meaning.

First, construction schedules of eight finished or ongoing construction projects were analyzed regarding the hierarchical structure of the processes and the used decomposition criteria. These sites were pilot sites of the EU Horizon 2020 project BIM2TWIN or active projects of the members of Innovations Management Bau GmbH, an industry partner of the authors. Five different construction companies in four countries (Germany, Spain, Finland, and France) are covered. For all of them, a schedule was available in PDF or Excel format, created by the export functionalities of the scheduling software used by the construction company.

All construction processes in the schedules that were divided into sub-processes were manually analyzed regarding the applied criteria for decomposition. The criteria were sorted according to the classification categories presented in Table 1. On the one hand side, some of these categories, like *CAD* and *Information*, do not make sense as a process decomposition criterion considering only the construction processes and not the planning phase and were therefore never used in any schedule. On the other hand side, the processes category was divided into production methods and construction steps for more detailed differentiation. In total, nine decomposition criteria were used in the analyzed schedules. Table 2 gives a complete list with examples for every criterion and the percentage of schedules in which this criterion was used. While eight construction schedules are insufficient to provide a comprehensive list of all criteria for decomposing processes into sub-processes, the authors are confident that the most common ones are covered since the high-level categories of construction classification systems were taken as guidance. All categories identified as meaningful in the context of process decomposition were encountered in at least one of the available schedules.

Table 2
Process decomposition criteria used in analyzed construction projects.

Criterion	Examples	Percentage used
Phase	earth works, frame errection	1.00
Production method	precast, cast-in-place	0.13
Construction step	place formwork, pour concrete	1.00
Location	building, storey, room	1.00
Element	wall, column, slab	0.75
Discipline / domain	plumber, electrician, painter	0.88
Equipment	crane, excavator, concrete mixer	0.25
Material	wood, tiles, concrete, steel	0.25
Property	diameter, width, height, load-bearing	0.38

While many process decompositions can clearly be assigned to a single criterion, some are fitting more than one criterion. As an example, dividing the process that summarizes the indoor works into separate sub-processes, e.g., plumbing, electrical work, painting, and so on, can be seen as a sequence of construction steps or a division of the indoor works by discipline. This depends on the primary intention of the scheduler. While there is no correct and incorrect option, the focus of the decomposition is slightly shifted, in the given example, to the worker crews belonging to different domains or to the general sequence in which the processes need to be performed.

3.2. Construction Scheduling Ontology

A new ontology for process representation was developed based on the identified shortcomings of existing process schemata. This Construction Scheduling Ontology is a minimal ontology that describes construction processes with their hierarchical structure, process dependencies, and related concepts. First, a UML diagram of the schema was developed, which was manually converted into an OWL ontology using Protégé [22]. The UML diagram of the Construction Scheduling Ontology (CSO) is shown in Figure 5.

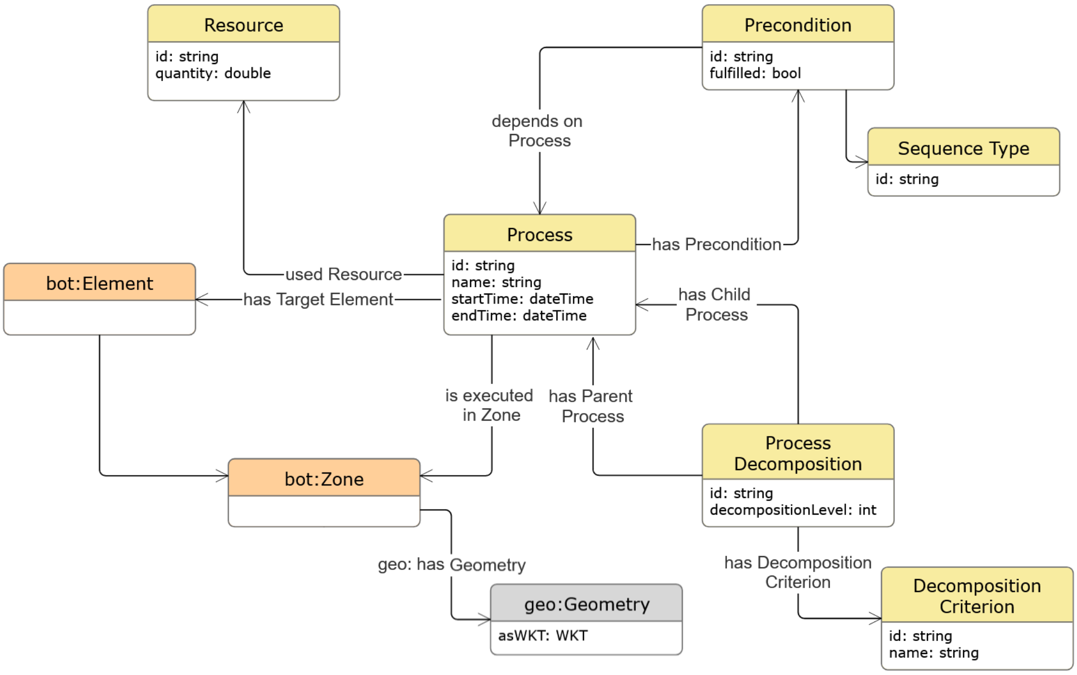


Figure 5: UML diagram of the Construction Scheduling Ontology (yellow: new classes, orange: existing classes, gray: optional classes).

The primary element of the ontology is the `process` class. Every construction schedule is seen as a list of processes containing basic information like start and end time. The class `process decomposition` is used to describe the hierarchical structure of processes. This is modelled as a separate class to be able to attach additional information like the criterion for process decomposition. The relationship `hasParentProcess` and `hasChildProcess` connect the process decomposition with parent processes and its sub-processes. The `decomposition criterion` class is used to define the criterion which is used for the decomposition. The ontology defines nine individuals for this class: phase, production method, construction step, location, element, discipline, equipment, material, and property. In this context, the individuals can be understood as an enumeration for the decomposition criterion class. The nine classes correspond with the criteria presented in Table 2. However, these individuals are not represented in the UML in Figure 5. Since it is not allowed to point an object property to a class or a datatype property (source), the `name` property of the decomposition criterion is used to specify which exact class or property is used to divide a process into several subprocesses. As an example, building elements can be grouped according to their subclass of `bot:Element`, creating separate processes for walls, columns, slabs, and so on. Another example would be grouping elements according to the value of a specific attribute, e.g., separating load-bearing and non load-bearing walls.

With the class `precondition`, the dependencies between processes are represented. Four types of process sequences are defined: start-end, start-start, end-start, and end-end. Moreover, the `resource` class summarizes different types of resources that are required for a particular process. This can include construction workers, equipment, materials, and temporary installations like formwork and guardrails. Since existing ontologies already define these resource sub-classes in close detail, only the resource class without sub-classes is defined in CSO, which shall be used as a connection point to other ontologies.

For the representation of building elements and their spatial distribution within specific buildings, storeys, and rooms, the Building Topology Ontology (BOT) [23] and the GeoSPARQL Ontology [24] are reused. Construction schedules often define work sections that are not represented in the BIM model. Nevertheless, to automatically filter elements by their corresponding zones, the spatial extent of these zones needs to be known. For the case study presented in this paper, GeoSPARQL and the Well-Known-Text (WKT) format are used to assign geometric information to zones [24]. However, this is not a mandatory choice and can be replaced by other geometric representations.

While this paper presents a new ontology for construction processes, the classes about process decomposition and dependencies could alternatively be integrated into already existing ontologies. Integration into ifcOWL does not seem suitable due to the significant overhead and missing integration in project management software. However, DiCon, CTO, and IOC are assessed to be suitable to be extended with the missing concepts with relatively little effort. For example, the classes *cso:Process* and *cto:Task* and *cso:hasTargetElement* and *cto:isSubjectOfTask* can be considered equivalent and form a good starting point for an alignment.

3.3. Workflow of IFC-to-schedule linking

The proposed process model has multiple potential applications. In the context of this study, a methodology for semi-automated linking of building elements defined in an IFC file with individual processes outlined in the construction schedule is introduced. This linkage is facilitated through explicitly defined decomposition criteria, allowing for a predominantly automatic interpretation of the schedule.

Manual linking these elements is highly time-consuming, given that BIM models often comprise several hundreds or even thousands of building elements. Nonetheless, the connection of processes and elements proves valuable, offering benefits such as enhancing the schedule based on precise quantities extracted from the BIM model and facilitating 4D simulation of the construction process.

To implement the proposed workflow, the construction schedule must be available in an open format like CSV, enriched with additional information about the employed process decomposition criteria. A boolean flag indicating whether the building elements corresponding to the processes are modeled in the IFC model is also required, currently added manually to the schedule. Additionally, the BIM model is required to have access to detailed information about the building elements. Depending on the schedule and the exact use case, information regarding resource assignments might also be needed.

Initiating the workflow involves converting both the IFC model and the schedule into RDF. For the IFC model, two commonly used converters exist. The IFCtoRDF converter translates the IFC file into an RDF graph using the ifcOWL ontology. This is suitable when detailed information on the geometry and attributes of building elements is needed [25]. The IFCtoLBD converter is a lightweight alternative. It uses BOT and BEO to represent the spatial structure of the building in the RDF graph without considering geometric information [26, 23, 27]. The construction scheduling ontology is proposed to represent the construction schedule in RDF. Subsequently, these separate RDF graphs must be injected into an RDF database.

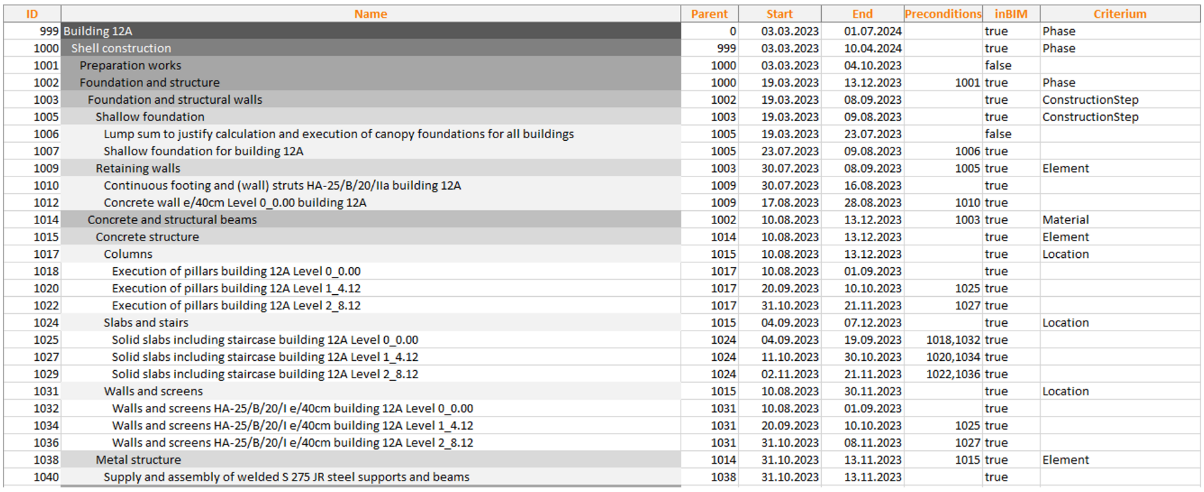
The final step involves linking the processes with their corresponding building elements in the RDF graph through the object property `cso:hasTargetElement`. This is accomplished by an algorithm that traverses each process in the construction schedule according to its hierarchical structure. If the target elements are not represented in the IFC file, the process is skipped, and no element is assigned to it. Conversely, if the target elements are present, the building elements linked to the process's parent serve as a starting point. Depending on the process decomposition criterion, the algorithm filters out all elements that do not fulfill the criterion, and the remaining elements are then connected to the current process in the RDF graph. In order to match a sub-process with the exact filtering condition, the process name is searched for specific keywords. As an example, when a process decomposition by element type is applied it is searched for words like wall, column, and window to know which process should be assigned with which element type. This is, however, a limitation of this implementation. Table 3 provides an overview of how the linking algorithm handles various types of decomposition criteria.

Table 3
Filtering algorithms for all process decomposition criteria.

Criterion	Filtering algorithm
Element	Check for rdf:type of element nodes Filter by bot:element subclasses (e.g. defined in BEO or ifcOWL)
Location	Zone defined in IFC: Check for bot:containsElement object property It assigns elements to either a bot:Building, bot:Storey, or bot:Space Assumption that elements are, .e.g, not divided by space if there are elements that belong to various buildings Largest possible zone has priority
	Zone not defined in IFC: Additional information about the geometry of the zone required Add zone to RDF graph as bot:Zone Check which element bounding boxes lie completly within this zone
Phase Construction step Production method	No immediate filtering Once all subprocesses are handled the parent processes connected with all elements that are connected to any of its subprocesses
Property	Check datatype properties of element node Filter by value of a particular property
Equipment Discipline Material	Check for resources assigned to process with cso:usedResource Filter by class type of equipment, discipline or material

4. Case study

A construction site in Spain was selected as the case study to validate the algorithm's efficacy in linking the schedule with the IFC file. The utilized IFC file was generated by merging the architectural and structural models provided by the construction company. The construction schedule, devised using the Primavera management software, was exported into Microsoft Excel format. Subsequently, the process decomposition criteria and the boolean flag, indicating the general existence of target elements for each process in the IFC file, were manually added based on the knowledge of planners involved in the project. The construction schedule comprises approximately 90 processes, incorporating decomposition criteria such as phase, construction step, material, element, location, and discipline. A visual representation of the IFC file and a segment of the schedule is presented in Figure 6 and 7.



ID	Name	Parent	Start	End	Preconditions	inBIM	Criterium
999	Building 12A	0	03.03.2023	01.07.2024		true	Phase
1000	Shell construction	999	03.03.2023	10.04.2024		true	Phase
1001	Preparation works	1000	03.03.2023	04.10.2023		false	
1002	Foundation and structure	1000	19.03.2023	13.12.2023	1001	true	Phase
1003	Foundation and structural walls	1002	19.03.2023	08.09.2023		true	ConstructionStep
1005	Shallow foundation	1003	19.03.2023	09.08.2023		true	ConstructionStep
1006	Lump sum to justify calculation and execution of canopy foundations for all buildings	1005	19.03.2023	23.07.2023		false	
1007	Shallow foundation for building 12A	1005	23.07.2023	09.08.2023	1006	true	
1009	Retaining walls	1003	30.07.2023	08.09.2023	1005	true	Element
1010	Continuous footing and (wall) struts HA-25/B/20/IIa building 12A	1009	30.07.2023	16.08.2023		true	
1012	Concrete wall e/40cm Level 0_0.00 building 12A	1009	17.08.2023	28.08.2023	1010	true	
1014	Concrete and structural beams	1002	10.08.2023	13.12.2023	1003	true	Material
1015	Concrete structure	1014	10.08.2023	13.12.2023		true	Element
1017	Columns	1015	10.08.2023	13.12.2023		true	Location
1018	Execution of pillars building 12A Level 0_0.00	1017	10.08.2023	01.09.2023		true	
1020	Execution of pillars building 12A Level 1_4.12	1017	20.09.2023	10.10.2023	1025	true	
1022	Execution of pillars building 12A Level 2_8.12	1017	31.10.2023	21.11.2023	1027	true	
1024	Slabs and stairs	1015	04.09.2023	07.12.2023		true	Location
1025	Solid slabs including staircase building 12A Level 0_0.00	1024	04.09.2023	19.09.2023	1018,1032	true	
1027	Solid slabs including staircase building 12A Level 1_4.12	1024	11.10.2023	30.10.2023	1020,1034	true	
1029	Solid slabs including staircase building 12A Level 2_8.12	1024	02.11.2023	21.11.2023	1022,1036	true	
1031	Walls and screens	1015	10.08.2023	30.11.2023		true	Location
1032	Walls and screens HA-25/B/20/I e/40cm building 12A Level 0_0.00	1031	10.08.2023	01.09.2023		true	
1034	Walls and screens HA-25/B/20/I e/40cm building 12A Level 1_4.12	1031	20.09.2023	10.10.2023	1025	true	
1036	Walls and screens HA-25/B/20/I e/40cm building 12A Level 2_8.12	1031	31.10.2023	08.11.2023	1027	true	
1038	Metal structure	1014	31.10.2023	13.11.2023	1015	true	Element
1040	Supply and assembly of welded S 275 JR steel supports and beams	1038	31.10.2023	13.11.2023		true	

Figure 6: Segment of the construction schedule.

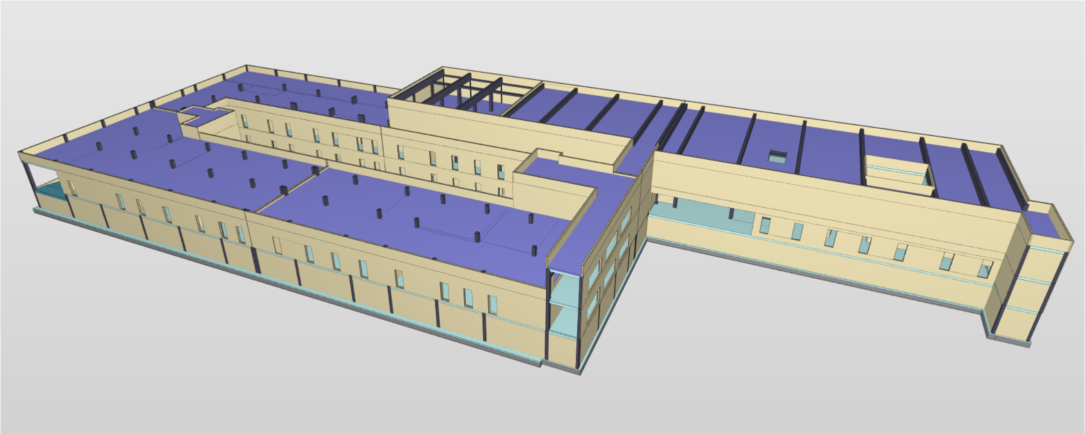


Figure 7: Overview of the Spanish construction project.

To implement the linking algorithm detailed in Section 3.3, the C# programming language was employed. Commencing with the IFC file, the preexisting IFCtoLBD converter was utilized [26]. This converter transforms IFC files into RDF representation, using the ontologies BOT, BEO, and PROPS. The resulting RDF graph was then stored in the GraphDB [28] RDF database. To facilitate interaction with the SPARQL endpoint of the database in the C# programming environment, the dotNetRDF[29] library was incorporated. Given that the IFCtoLBD converter does not include geometric information, furthermore, the xBIM[30] library was employed to extract additional details from the IFC file for the linking process. Regarding the construction schedule, a dedicated converter was developed. This converter reads the process information from the Excel file and transforms it into the corresponding RDF representation. Subsequently, this information is transmitted to the GraphDB SPARQL endpoint to update the database.

The filtering algorithm, as outlined in Section 3.3, has been successfully implemented, featuring distinct functions tailored to handle the different decomposition criteria, as detailed in Table 3. To align a specific sub-process with the precise filtering condition, the process name undergoes analysis for specific keywords. For instance, in the case of process decomposition by element type, the algorithm scans for keywords such as "wall," "column," and "window" to determine the appropriate assignment of processes to element types. However, it is crucial to acknowledge a limitation inherent in this implementation. For the processes related to walls, manual intervention was necessary to point the filtering algorithm to the location where the relevant information could be found within the IFC file. In the given case, load-bearing walls had to be distinguished from walls designed for interior partitions. However, in the IFC file, some load-bearing walls were missing the load-bearing property. For this reason, it could not be used to separate the two different types of walls correctly. Manual analysis identified that load-bearing walls were assigned a different *IfcMaterial* than the interior partitions. Manually defining that the filtering algorithm should use the material as a distinguishing factor between load-bearing walls and interior partitions for this specific process was sufficient to fix the issue. Furthermore, the use of abbreviations in process names poses a challenge for accurate linking. In the frame of the case study, this was resolved by manually replacing abbreviations in the schedule with the complete word.

The outcomes of the linking algorithm are depicted in Figure 8, where building elements linked to the same process share a common color. To enhance clarity, only the leaf processes were considered, preventing elements from being assigned the color of multiple processes. Additionally, the building is separated into the four storeys for visualization purposes.

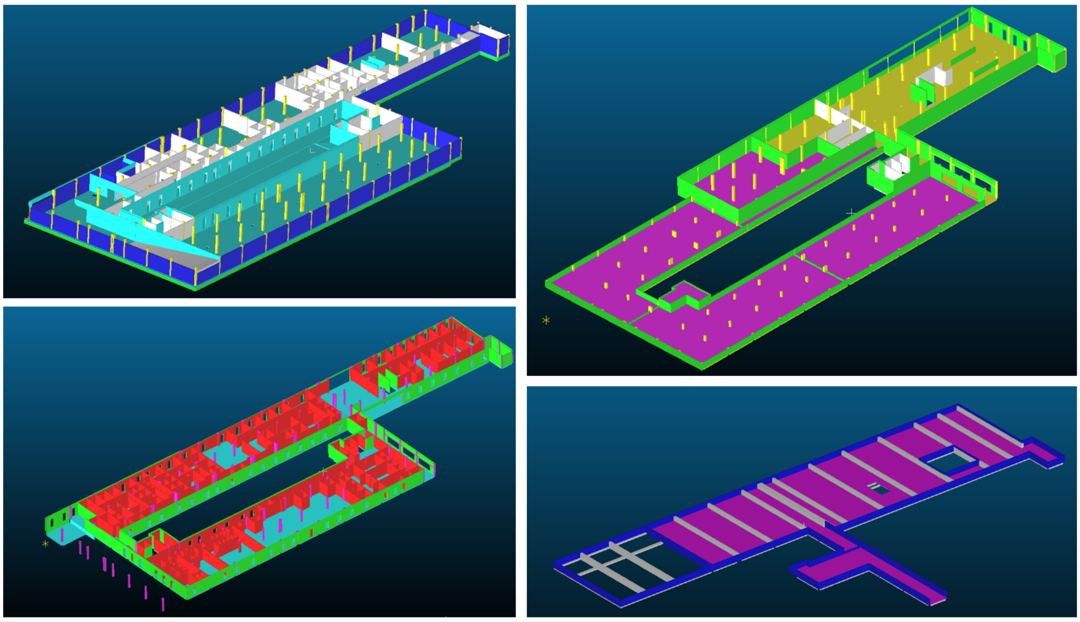


Figure 8: IFC elements colored according to their process correspondence.

5. Discussion

The availability of additional information about the schedule regarding process decomposition proved beneficial in interpreting the process hierarchy. This supplementary information facilitates a largely automated process interpretation, aiding in identifying relevant processes for specific use cases. For instance, the ability to filter out irrelevant parts of the schedule for a particular simulation based on decomposition criteria streamlines the overall analysis and narrows the focus to pertinent subsections. In the present research, the process interpretation was used for semi-automated linking between processes and building elements. Even though minor manual intervention was required because of cases where it is not immediately clear if certain information is specified in the name, a property or the assigned material of an entity in the IFC file, significant time savings could be achieved.

Several limitations deserve acknowledgment in our study. The effectiveness of the linking processes is strongly dependent on the quality of the IFC model. Instances of modeling errors, such as elements erroneously assigned to the wrong storey or lacking material definitions, negatively influence the accuracy of the results. Furthermore, a certain level of correlation between the schedule and the IFC model is required. Shared attribute and class names facilitate the linking process, and without such consistency, full automation becomes challenging. Another limitation involves the granularity of processes and the BIM model, where disparities may arise. For example, an element described in the schedule as having multiple parts might be modeled as a single element in the IFC file. While an algorithm developed by Tulke [31] addresses such cases by dividing elements to match the granularity of the construction schedule, it was not applied in the present paper. Finally, inconsistencies within the schedule also pose challenges. Instances where multiple decomposition criteria are simultaneously applied to a single process decomposition can lead to misinterpretation. It is suggested to divide such decompositions into distinct processes. Although this results in more processes, it enhances comprehension.

6. Conclusion

In addressing the challenges surrounding the exchange of project management information within the construction industry, emphasising construction schedules, this research has introduced a novel approach to enhance automation and efficiency. The prevalent reliance on XML or spreadsheet-based files for schedule exchange has been identified as a source of information loss and misinterpretation within commonly used project management software. Diverse scheduling practices across construction companies intensify the issue by contributing to a lack of standardisation.

To bridge this gap, the paper proposed a minimal ontology specifically designed to represent construction schedules' processes, addressing the limitations of existing data schemata. It relies on analysing process decomposition criteria used in eight construction schedules. A small-scale case study showcased the ontology's application in automated schedule interpretation, highlighting its potential for linking between schedules and corresponding BIM models. This approach reduces manual efforts but also significantly enhances overall efficiency in project management processes. The broader implications of the introduced ontology extend beyond the immediate scope of schedule exchange. The ontology's versatility offers opportunities for consistency checks, completeness assessments, and streamlined coordination between contractors and subcontractors.

Future improvements should revolve around the application of natural language processing techniques to improve the interpretation of process names and identify their filtering criteria fully automated. Furthermore, the developed ontology should be fully integrated into existing ontologies after consultation with the corresponding ontology creators to follow the FAIR data principles and ensure that others can apply the developed concepts more quickly. While this study only considered the topmost level of construction classification systems for process decomposition criteria, it should be investigated how considering the complete classification hierarchy could improve automated schedule interpretation.

Acknowledgments

We thankfully acknowledge Innovation Management Bau for their financial support and for providing us with information about multiple construction projects. Moreover, the research described in this paper received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 958398, "BIM2TWIN: Optimal Construction Management & Production Control". The authors also thankfully acknowledge the support of the European Commission in funding this project.

References

[1] P. Arnold, A. Javernick-Will, Projectwide Access: Key to Effective Implementation of Construction Project Management Software Systems, Journal of Construction Engineering and Management 139 (2013) 510–518. doi:10.1061/(asce)co.1943-7862.0000596.

[2] C. I. D. Gaetani, M. Mert, F. Migliaccio, Interoperability analyses of BIM platforms for construction management, Applied Sciences (Switzerland) 10 (2020). doi:10.3390/app10134437.

[3] D. Cho, M. Lee, J. Shin, Development of cost and schedule data integration algorithm based on big data technology, Applied Sciences (Switzerland) 10 (2020) 1–17. doi:10.3390/app10248917.

[4] T. da C.L. Alves, M. Liu, N. M. Scala, A. Javanmardi, Schedules and Schedulers: A Study in the U.S. Construction Industry, EMJ - Engineering Management Journal 32 (2020) 166–185. doi:10.1080/10429247.2020.1738878.

[5] buildingSMART International, IFC Schema Specifications, 2021. URL: https://technical.buildingsmart.org/standards/ifc/ifc-schema-specifications/.

[6] Y. Zheng, S. Törmä, O. Seppänen, A shared ontology suite for digital construction workflow, Automation in Construction 132 (2021) 103930. doi:10.1016/j.autcon.2021.103930.

[7] L. Kirner, J. Oraskari, P. Wildemann, S. Brell-Cokcan, Internet of Construction Process Ontology (ioc) v 0.5, 2024. doi:10.5281/zenodo.10592282.

[8] J. Schlenger, T. Yeung, S. Vilgertshofer, J. Martinez, R. Sacks, A. Borrmann, A Comprehensive Data Schema for Digital Twin Construction, in: 29th International Workshop on Intelligent Computing in Engineering, 2022, pp. 34–44.

[9] M. Bonduel, CTO: Construction Tasks Ontology, 2020. URL: https://mathib.github.io/cto-ontology/.

[10] V. Franz, Unikatprozesse und ASIM-Aktivitäten - Bericht von der Arbeitsgruppe "Unikatprozesse", in: H.-J. Bargstädt (Ed.), Forschungsworkshop zur Simulation von Bauprozessen, Bauhaus-Universität Weimar Fakultät Bauingenieurwesen, 2010, pp. 5–16.

[11] M. Burghate, Work Breakdown Structure: Simplifying Project Management, in: International Journal of Commerce and Management Studies (IJCAMS), volume 3, 2018, pp. 1–5.

[12] A. Cerezo-Narváez, A. Pastor-Fernández, M. Otero-Mateo, P. Ballesteros-Pérez, Integration of cost and work breakdown structures in the management of construction projects, Applied Sciences (Switzerland) 10 (2020). doi:10.3390/app10041386.

[13] Y. Jung, S. Woo, Flexible Work Breakdown Structure for Integrated Cost and Schedule Control, Journal of Construction Engineering and Management 130 (2004) 616–625. doi:10.1061/(asce)0733-9364(2004)130:5(616).

[14] S. Globerson, Impact of various work-breakdown structures on project conceptualization, International Journal of Project Management 12 (1994) 165–171. doi:10.1016/0263-7863(94)90032-9.

[15] B. Golany, A. Shtub, Work Breakdown Structure, John Willey & Sons, Inc., 2001, pp. 1263–1280. doi:https://doi.org/10.1002/9780470172339.ch47.

[16] International Organization for Standardization, ISO 12006-2: Building Construction. Organization of Information about Construction Works. Part 2: Framework for Classification of Information, 2nd ed., International Organization for Standardization, 2015.

[17] L. van Berlo, T. Krijnen, H. Tauscher, A. van Kranenburg, P. Paasiala, Future of the Industry Foundation Classes: towards IFC 5, in: 38th International Conference of CIB W78, 2021, pp. 123–137.

[18] T. Berners-Lee, J. Hendler, O. Lassila, The semantic web, Scientific american 284 (2001) 34–43.

[19] J. Beetz, J. V. Leeuwen, B. D. Vries, IfcOWL: A case of transforming EXPRESS schemas into ontologies, Artificial Intelligence for Engineering Design, Analysis and Manufacturing 23 (2009) 89–101. doi:10.1017/S0890060409000122.

[20] A. Borrmann, J. Schlenger, N. Bus, R. Sacks, AEC Digital Twin Data - Why structure matters, in: 19th International Conference in Civil & Building Engineering, 2022, pp. 651–669.

[21] M. D. Wilkinson, M. Dumontier, I. Aalbersberg, G. Appleton, M. Axton, et al., The FAIR Guiding Principles for scientific data management and stewardship, Scientific data 3 (2016) 1–9.

[22] N. F. Noy, M. Crubézy, R. W. Fergerson, H. Knublauch, S. W. Tu, J. Vendetti, M. A. Musen, Protégé-2000: An Open-Source Ontology-Development and Knowledge-Acquisition Environment AMIA 2003 Open Source Expo, in: AMIA 2003 Symposium, 2003, pp. 953–953. URL: http://protege.stanford.edu.

[23] M. H. Rasmussen, M. Lefrançois, G. F. Schneider, P. Pauwels, BOT: The building topology ontology of the W3C linked building data group, Semantic Web 12 (2020) 143–161. doi:10.3233/SW-200385.

[24] R. Battle, D. Kolas, GeoSPARQL: Enabling a Geospatial Semantic Web, Semantic Web Journal 3 (2011) 355–370. URL: http://parliament.semwebcentral.org.

[25] P. Pauwels, J. Oraskari, L. J. McGibbney, IFCtoRDF Converter, 2016. URL: https://github.com/pipauwel/IFCtoRDF.

[26] M. Bonduel, J. Oraskari, P. Pauwels, M. Vergauwen, R. Klein, The IFC to linked building data converter - Current status, in: 6th Linked Data in Architecture and Construction Workshop, volume 2159, 2018, pp. 34–43.

[27] P. Pauwels, Building Element Ontology, 2018. URL: https://pi.pauwel.be/voc/buildingelement/.

[28] Ontotext, GraphDB, 2024. URL: https://www.ontotext.com/products/graphdb/.

[29] R. Vesse, R. M. Zettlemoyer, K. Ahmed, G. Moore, T. Pluskiewicz, S. Lang, dotNetRDF, 2023. URL: http://dotnetrdf.org/.

[30] xBIM Team, xBIM Toolkit, 2024. URL: https://docs.xbim.net/.

[31] J. Tulke, Kollaborative Terminplanung auf Basis von Bauwerksinformationsmodellen, Informatik in Architektur und Bauwesen 4 (2010). URL: https://d-nb.info/1116284456/34.