IFC2TSO – Algorithmic processing, complexity reduction, and transfer of information regarding technical systems from IFC to TSO

Nicolas Pauen[1], Jérôme Frisch[1] and Christoph van Treeck[1]
[1] Institute of Energy Efficiency and Sustainable Building - E3D, RWTH Aachen University, Mathieustraße 30, 52074 Aachen, Germany
E-mail(s): pauen@e3d.rwth-aachen.de

Abstract: Technical systems in the Architecture, Engineering, Construction and Operations (AECO) industry are complex interconnected structures that are interdependent with each other and with spatial entities. Building Information Modeling (BIM) makes it possible to represent these systems using structured, digital information in machine-readable representations. However, not all hierarchical, topological, and functional aspects of technical systems can be represented in current data models and linked with information on further domains. Therefore, ontologies such as the TUBES System Ontology (TSO), which are based on Semantic Web Technologies (SWT), offer representations of building service systems and the possibility to link information web-based on data level. To improve the application of TSO in particular and knowledge representations based on SWT in general and, thus, to create a concrete added value in practice, appropriate tools have to be designed or the concepts have to be integrated into existing software applications. This contribution proposes a workflow for algorithmic processing, topological complexity reduction, and transfer of the information regarding technical systems from Industry Foundation Classes (IFC) models to TSO. The process has a modular structure consisting of three individual process steps IFC2GRAPH, GRAPH, and GRAPH2TSO, which can be used independently and, thus, also for review processes and quality assurance of technical systems in BIM models. The proposed workflow is implemented and validated using two application examples.

Keywords: BIM, IFC, Building Service Systems, Linked Data, TUBES System Ontology

1 Introduction

In recent years, the use of Semantic Web Technologies (SWT) in the Architecture, Engineering, Construction and Operations (AECO) industry has increased significantly [1]. For this purpose, different ontologies, such as the Building Topology Ontology (BOT) [2], Semantic Sensor Network

(SSN)/ Sensor, Observation, Sample, and Actuator (SOSA) ontology [3], Smart Applications Reference (SAREF) ontology [4], and the BRICK Schema [5] have been developed, covering various use cases. Another recent development is the TUBES System Ontology (TSO) [6], which aims to explicitly define interconnected technical systems in the AECO industry, their hierarchical structure, structural and functional relationships, and links to spatial entities. As such, TSO supports the effort to represent linkable information in a future semantic web of building data. To improve the application of TSO in particular and knowledge representations based on SWT in general and, thus, to create a concrete added value in practice, appropriate tools have to be designed or the concepts have to be integrated into existing software applications. Efforts such as the transformation of Industry Foundation Classes (IFC) into the Web Ontology Language (OWL) [7] and the Linked Building Data Converter [8] offers the possibility to transfer and represent data from IFC files. But the quality of the data cannot be checked nor can the information be enriched to represent additional concepts of building service systems which are not included in IFC.

Therefore, this work proposes a workflow for algorithmic processing, topological complexity reduction, and transfer of information regarding technical systems from IFC models to TSO which enhances the work presented in [9]. The process has a modular structure consisting of three individual process steps IFC2GRAPH, GRAPH, and GRAPH2TSO, which can be used independently and, thus, also for review processes and quality assurance of technical systems in BIM models. Each process step focuses on a different aspect of technical systems. In IFC2GRAPH, topological information is considered and missing semantic links can be enriched based on the position of the components. GRAPH focuses on hierarchical information and the complexity reduction of topological data. GRAPH2TSO focuses on the functional information and dependencies between technical systems and spatial entities that can be enriched based on spatial representations. The results of each process step can also be exported as a BIM Collaboration Format (BCF) files to allow verification in the authoring software. The workflow is implemented as a Command Line Interface application based on Python 3.7.12 and validated using the fictitious DigitalHub [10] and a real-world project.

2 IFC2TSO

The workflow of the IFC2TSO process and the modular process steps IFC2GRAPH, GRAPH and GRAPH2TSO are visualized in figure 1 and explained in the following sections.

2.1 IFC2GRAPH

In the IFC2GRAPH process step, information regarding building service systems contained in one or multiple IFC4 models is analyzed, enriched and converted into graphs. The focus of the enrichment is on topological information. Therefore, all elements with a subclass of *IfcDistributionElement* and *IfcBuildingElementProxies* are considered. Their classes, predefined types, systems and topological connections through *IfcDistributionPorts* are analysed and parsed into a directed graph. Each node in the graph represents an element and each edge represents a topological connection. To enrich further connections and revise modeling errors, the absolute positions of these *IfcDistributionPorts*

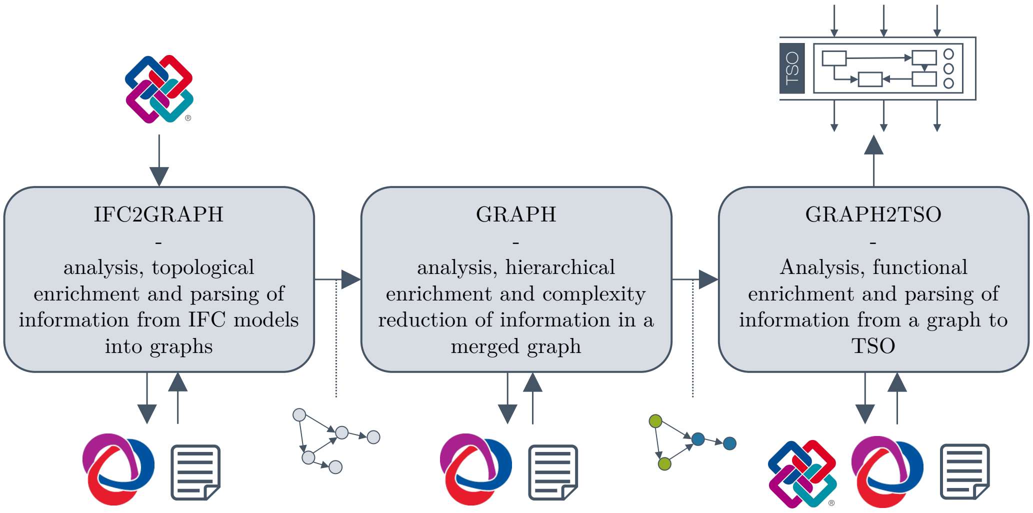

Figure 1: Schematic visualization of the IFC2TSO process

are calculated using a recursive algorithm and organized in a R-tree. R-trees are multidimensional, dynamic index structures, which allow efficient queries based on the location of the objects. To find possible matches for open ports, which have no connection assigned, the constructed R-tree can be queried. Hereby, ports within defined boundaries, which have no topological neighbor and the same *PredefinedType* are considered. The default boundary is 50 mm, but can be customized by the user. If these conditions apply to multiple ports, the closest port in the three-dimensional Euclidean space is selected and the connection is stored in an array. The results of the enrichment can be exported as BIM Collaboration Format (BCF) files, which enable a correction of possible errors in the original authoring software. Additionally, based on the enriched connections, further edges can be added to the graph.

2.2 GRAPH

In the GRAPH process step, the resulting graphs are merged and the contained information is enriched by a hierarchical structure and reduced in its topological complexity. During merging, the user can explicitly define further edges between components of different IFC models. Based on given IFC systems, classification of components and their topological structure, system aggregations on different levels of hierarchy are enriched. For the top level integrated systems, the merged graph is examined for connectivity. For each weakly connected subgraph with a size of at least R, which can be assigned by the user to clear up interfering artifacts, an integrated system is created. Next, the contained functional systems and their classification are determined for each integrated system. Therefore, the IFC systems and component classifications contained in the weakly connected graph are analyzed using regular expressions. If all IFC systems and components can be unambiguously assigned to a classification of functional systems presented in https://w3id.org/tso, the assumption is made that the integrated system consists only of one functional system of the analyzed classification. Thus, the definition as an integrated system is discarded. If more than one differently classified functional system exists in the weakly connected system, the boundaries of the functional systems are defined

based on the given IFC systems and enriched accordingly. This hierarchical structure can be further enriched based on explicit input of the user in the form of a JSON file to take complex structures and detailed subdivisions into consideration.
For the reduction of complexity and memory requirement, the topological complexity of the graph is reduced while maintaining the same topological information content. It is assumed, that active elements, such as sensors, valves, and energy converters, have a higher topological value than passive components, such as pipes, ducts, or corresponding fittings. The topological value of each node is calculated based on its classification and the number of its neighbors. If the number of neighbors is less than three and the classification of the component is a subclass of *IfcFlowSegment* or *IfcFlowFitting*, the node is considered irrelevant and is removed from the graph along with the connected edges. Based on this, a new edge is added that connects the neighbors of the removed node. All edges have the attribute *node_ids* assigned, in which the GUIDs of the reduced nodes are stored. This process of aggregation is performed iteratively until the number of nodes before and after an iteration does not change anymore.

2.3 GRAPH2TSO

In the GRAPH2TSO process step, the functional information contained in the graph is analyzed, enriched and transferred to TSO. The focus of the enrichment is on functional concepts and the relation to spatial entities. To enrich the correct flow direction and the transferred matter, energy or data, additional user input is needed. Based on explicit input, search algorithms, such as depth-first search, are used to evaluate different system topologies in an efficient and scalable way.
Additionally, spatial entities, which are included in a given IFC4 model can be analysed and their relationships to technical system can be enriched. All spatial entities as subclasses of *IfcSpatialElement*, are transferred while taking their hierarchical subdivision into consideration. In addition, their geometric representations are transformed into a triangulated tessellation. Based on the position of the components, which were calculated in the IFC2GRAPH process step and stored as parameters in the graph, the location of the components in these spatial entities can be enriched by calculating the *signed distance* to the triangulated tessellations. Moreover, based on the element classification and the enriched classification of upper hierarchical systems, it is determined whether a spatial entity is served by the component. This can be extended by explicit user specifications. The distinction of IFC models according to spatial concepts and technical systems is necessary, as not in every building service systems model are corresponding spatial entities. Often, these are only represented in the architecture model and, thus, the semantic relations between the spatial entities and the technical systems are not explicitly given.

3 Implementation

An implementation of the IFC2TSO process is published at https://github.com/RWTH-E3D/IFC2TSO and licensed using the MIT License. In addition, an Anaconda environment basend on python 3.7.12 with the libraries used to run the process is given. These are among others ifcopenshell 0.6.0 for parsing the IFC models, lxml 4.6.1 for designing the XML based BCF files, NetworkX 2.5 for handling the graphs, Rtree 0.9.7 for setting up and querying r-trees, trimesh 3.10.2 for spatial dependency analysis, and rdflib 6.1.1 for conception and serialization of the knowledge representation. If the specified libraries are no longer available in the above given versions, it is possible that adjustments have to be made to the program code to ensure functionality. Optional parameters can be used to adjust the functionality of the IFC2TSO process. A subset of possible parameters is given in table 1.

Table 1: Subset of optional parameters to customize the functionality of the IFC2TSO process

Parameter	Description
-bcf_pm	Export of possible topological connections as compressed BCF files at the path of the model.
-ce {L}	Automated extension of the graph with directed edges, based on the possible topological connections with the maximum allowed distance of L in mm.
-cr	Reduction of the topological complexity of the merged graph.
-add_fc {JSON}	Enrichment of functional concepts, such as flow of matter, energy, and data, sources and sinks, contained in the JSON file at the given path.
-add_spatial {IFC}	Transfer of spatial concepts contained in the IFC model at the given path and the enriched dependencies to technical systems.
-ifcowl	Extension of the A-Box with classifications of components based on IFCowl.

4 Validation

The validation of the IFC2TSO process is shown based on the fictitious DigitalHub project [10] and a real-world project. Due to the page limit, the focus is set on the IFC2GRAPH process step and the enrichment of spatial relationships. The figures are created using BIMcollab ZOOM and the resulting BCF files from the IFC2TSO process. The validation was run on a MacBook Pro with 16 GB RAM and an Intel i7 dual core with 3.3 GHz.
Table 2 summarizes the results of the parsing and enrichment of topological information from IFC models to graphs. It is shown, that all elements, which have the correct class and at least one port assigned, can be exported and added as nodes. Moreover, all existing topological connections can be exported as edges. The amount of exported edges is sometimes higher than half of the amount of connected ports, since some ports have an assigned bidirectional flow, which results in two inverse directed edges. This is shown for the enrichment of additional edges based on the spatial position of the ports as well. The results of the enrichment were validated by implementing visual checks of the resulting BCF files. Two examples of enriched connections are presented in figure 2.

Table 2: Results of the parsing and enrichment of topological information in the process step IFC2GRAPH

Model	**MR** [MB]	**CwP**	**Ports**	**open Ports**	**Nodes**	**Edges**	**eE**	**RT** [s]
DH-HC	25.9	1,795	3,760	14	1,795	1,901	0	8.9
DH-V	15.0	1,310	2,656	36	1,310	1,341	0	7.4
DH-S	23.0	911	2,079	33	911	1,284	2	6.3
RWP-HC	682.0	31,607	65,385	7,103	31,607	39,724	4,116	132.1
RWP-V	191.8	12,418	24,892	186	12,418	12,710	50	60.9
RWP-S	575.2	17,024	35,181	2,951	17,024	23,719	1,618	145.1

MR = Memory Requirement, CwP = Components with Ports, eE = enriched Edges, RT = Runtime,
DH = DigitalHub, RWP = Real-World Project, HC = Heating & Cooling, V = Ventilation, S = Sanitary

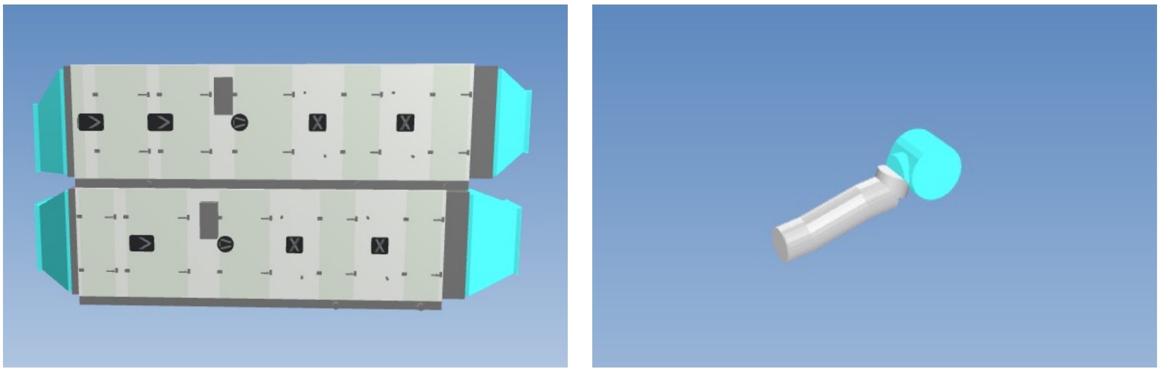

Figure 2: Visualization of components and their enriched topological connections in the ventilation model of the real-world project in BIMcollab ZOOM

The results for the enrichment of spatial concepts are given in table 3. It is shown, that all *IfcSpatialElements*, which are included in the given IFC model, could be parsed into the knowledge representation. Moreover, 2,412 spatial relationships could be enriched for the DigitalHub. This is less than the existing amount of components, since some of those are located outside of the building or in between walls. The components with an enriched relationship to a given spatial entity of the DigitalHub project and the real-world project are visualized in figure 3.

Table 3: Results of the enrichment of spatial concepts in the process step GRAPH2TSO

Model	**IfcSE**	**Spatial Instances**	**Spatial Relationships**	**MR** [MB]	**RT** [s]
DigitalHub-ARC	69	69	2,412	7.8	29.4
RWP-ARC	265	265	59,757	112.6	1412.4

IfcSE = IfcSpatialElements, RT = Runtime, MR = Memory Requirements, RWP = Real-World Project

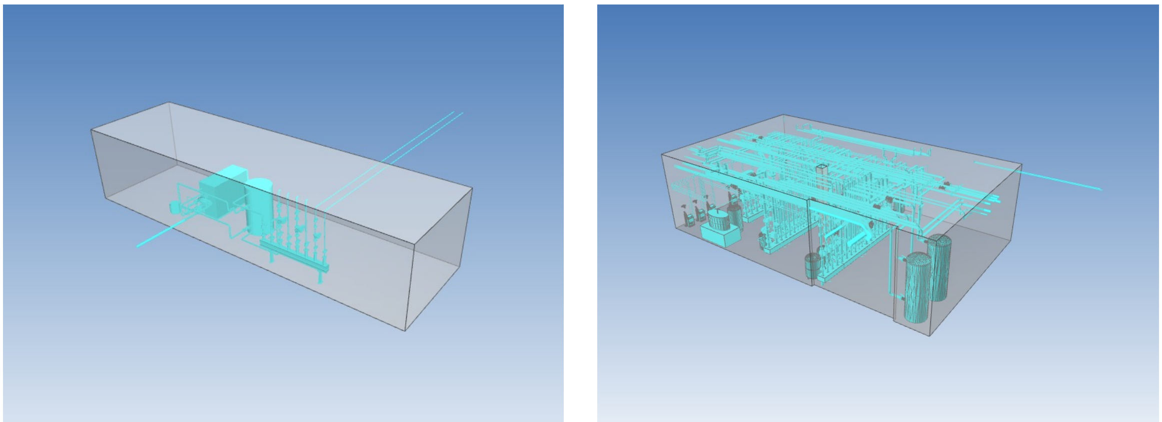

Figure 3: Visualization of components with an enriched relationship to the shown spatial entities of the DigitalHub (left) and the real-world project (right) in BIMcollab ZOOM

5 Conclusion

This contribution presents the modular IFC2TSO process for algorithmic processing, topological complexity reduction, and transfer of information regarding technical systems from IFC to TSO. IFC2TSO consists of three process steps IFC2GRAPH, GRAPH, and GRAPH2TSO, each of which can be used independently for applications such as model checking, quality control of BIM models, or for parsing data to other ontologies. Several optional parameters have been considered, allowing the user to customize the functionality. It has been shown that all relevant topological information inside the IFC models of the application examples could be successfully analyzed and parsed into a graph and subsequently into a knowledge representation. In addition, components that are not correctly connected are exported as BCF files, which enables the analysis and resolving of modelling errors in the original authoring software. Automatic enrichment of the connections is also possible, but not persistent, since the native model is not corrected. The integration of a graphical user interface is a possible field of further research. This could be implemented either by using an open tool, such as IFC.js, or by integrating the process into an existing software application, such as Autodesk REVIT. The user would thus have the opportunity to use algorithms for information enrichment of matter, energy, and data, among others, through direct interaction with the components, as well as receive direct feedback on the potential results of the enrichment. This would make the process more robust and could guarantee continuous use of the enriched data.

Acknowledgements

The research within the project EnergyTWIN leading to these results has received funding from the German Ministry for Economic Affairs and Climate Action (BMWK) under grant agreement no. 03EN1026A.

References

[1] P. Pauwels, S. Zhang, and Y.-C. Lee, “Semantic web technologies in AEC industry: A literature overview”, *Automation in Construction*, vol. 73, pp. 145–165, 2017.

[2] M. H. Rasmussen, M. Lefrancois, G. F. Schneider, and P. Pauwels, “BOT: The Building Topology Ontology of the W3C Linked Building Data Group”, *Semantic Web – Interoperability, Usability, Applicability*, 2020. DOI: https://doi.org/10.3233/SW-200385.

[3] A. Haller, K. Janowicz, S. J. D. Cox, *et al.*, “The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation”, *Semantic Web*, vol. 10, pp. 9–32, 2018. DOI: https://doi.org/10.3233/SW-180320.

[4] L. Daniele, F. den Hartog, and J. Roes, “Created in Close Interaction with the Industry: The Smart Appliances REFerence (SAREF) Ontology”, in *Formal Ontologies Meet Industry*, Springer International Publishing, 2015, pp. 100–112. DOI: https://doi.org/10.1007/978-3-319-21545-7_9.

[5] B. Balaji, A. Bhattacharya, G. Fierro, *et al.*, “Brick: Metadata schema for portable smart building applications”, *Applied Energy*, vol. 226, pp. 1273–1292, 2018. DOI: https://doi.org/10.1016/j.apenergy.2018.02.091.

[6] N. Pauen, D. Schlütter, J. Frisch, and C. van Treeck, “TUBES System Ontology: Digitalization of building service systems”, in *Proceedings of the 9th Linked Data in Architecture and Construction Workshop*, M. Poveda-Villalon, A. Roxin, K. McGlinn, and P. Pauwels, editors, 2021.

[7] J. Beetz, J. van Leeuwen, and B. de Vries, “IfcOWL: A case of transforming EXPRESS schemas into ontologies”, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 23, no. 1, pp. 89–101, Dec. 2008. DOI: 10.1017/s0890060409000122.

[8] M. Bonduel, J. Oraskari, P. Pauwels, M. Vergauwen, and R. Klein, “The IFC to Linked Building Data Converter - Current Status”, in *Proceedings of the 6th Linked Data in Architecture and Construction Workshop*, 2018.

[9] N. Pauen, D. Schlütter, J. Siwiecki, J. Frisch, and C. van Treeck, “Integrated representation of building service systems: Topology extraction and TUBES ontology”, *Bauphysik*, vol. 42, no. 6, pp. 299–305, 2020. DOI: https://doi.org/10.1002/bapi.202000027.

[10] N. Pauen, L. Unruh, D. Schlütter, J. Siwiecki, J. Frisch, and C. van Treeck, “Technical Report: IFC Modell DigitalHub”, Tech. Rep., 2020. DOI: https://doi.org/10.18154/RWTH-2020-11683.