# A roadmap toward a unified ontology for building service systems in the AECO industry: TSO and FSO

Nicolas Pauen[1], Ville Kukkonen[2,3], Ali Kücükavci[4,5], Mads Holten Rasmussen[6], Mikki Seidenschnur[4,7], Dominik Schlütter[1], Christian Anker Hviid[4] and Christoph van Treeck[1]

[1]*RWTH Aachen University, Aachen, Germany*
[2]*Aalto University, Espoo, Finland*
[3]*Granlund, Helsinki, Finland*
[4]*Technical University of Denmark, Copenhagen, Denmark*
[5]*COWI, Lyngby, Denmark*
[6]*NIRAS A/S, Allerød, Denmark*
[7]*Rambøll, Copenhagen, Denmark*

**Abstract**

Building service systems are complex structures consisting of different subsystems and components in varying relationships. Semantic Web Technologies (SWT) can be used to represent these systems in decentralized triplestores using ontologies. To describe interconnected building service systems and the flow of matter, energy and data between them over the whole life-cycle in the context of the Architecture, Engineering, Construction and Operation (AECO) industry, two recent contributions, TUBES System Ontology (TSO) and Flow System Ontology (FSO) have to be considered. This study thus supports the effort towards a future semantic web of building data by validating the given ontologies based on Competency Questions (CQs) and an application example, proposing an alignment of TSO v0.3.0 and FSO v0.1.0 and providing a roadmap to a unified representation of building service systems in the AECO industry.

**Keywords**

Linked Data, TSO, FSO, Building Service Systems, BIM

## 1. Introduction

Building service systems, such as Heating, Ventilation, and Air Conditioning (HVAC) systems, form complex networks of components with varying kinds of relationships. During design, large quantities of information are produced about these systems, their components, and topological as well as functional relationships, which goes underutilized during operations and maintenance. While current data models such as Industry Foundation Classes (IFC) provide a standard format that most design tools can export, this file-based approach has its limitations. Recent research in data exchange for the Architecture, Engineering, Construction, and Operations (AECO) industry has seen an increasing application of knowledge graphs and linked data [1]. To that end, two separate ontologies have been recently developed for describing building service systems and

---

*LDAC 2022: 10th Linked Data in Architecture and Construction Workshop, May 29, 2022, Hersonissos, Greece*

pauen@e3d.rwth-aachen.de (N. Pauen)

CEUR Workshop Proceedings (CEUR-WS.org)

their flow: the TUBES System Ontology [2, 3] and the Flow Systems Ontology [4]. While the ontologies have similar aims, they have their differences in conceptualizations.

In order to align the two ontologies and pave the road towards a future unified representation of building service systems from design to operations, this paper compares the ontologies and describes the similarities and differences. An alignment between the concepts of the latest versions of the ontologies is proposed, and a roadmap for future unification is presented.

The paper is structured as follows. First, Section 2 reviews the state of the art. Following that, in Section 3, the ontologies are briefly introduced and then compared through a set of competency questions aimed at exercising the hierarchical, topological, and functional concepts of the ontologies, ending with the proposed alignments. Section 4 presents the roadmap for a unified representation of building service systems. Finally, Section 5 concludes the paper with closing remarks.

## 2. State of the Art

Several ontologies have been developed to improve interoperability within the AECO industry. A set of ontologies related to building service systems are described in this section.

The ifcOWL ontology is one of the first steps towards connecting BIM and semantic web technologies. It translates the IFC schema directly into the Web Ontology Language (OWL). Since the ifcOWL ontology is directly derived from the IFC schema, the relationships between different building components are intricate [5]. A complex data structure makes it difficult for AECO stakeholders and their tools to easily access building information [6]. Furthermore, ifcOWL includes too many domains, complicating any extension process [7]. To avoid the complexity and monolithic data structure of ifcOWL, a recent trend suggests a more modular and domain-specific approach to model building information with ontologies such as the Building Topology Ontology (BOT) [8, 9].

The Smart Energy Aware System (SEAS) ontology was developed through the EUREKA ITEA 12004 SEAS PROJECT to extend the Semantic Sensor Network (SSN) Ontology [10]. The ontology consists of three core modules: *seas:FeatureOfInterestOntology*, *seas:SystemOntology*, and *seas:EvaluationOntology*. Systems, relationships between systems, and the connection points between them can be expressed using *seas:SystemOntology*. The SEAS ontology focuses on electrical engineering and the supply of electrical energy and describes building service systems on a higher conceptual level. However, the Brick ontology and The Smart Appliances REFerence (SAREF) ontology can be used to represent the relationship of systems and components at a lower-conceptual level and a broader scope.

The Brick ontology defines the relationships among systems, components, sensors, and control parameters [11]. Furthermore, it has a schema definition that categorizes its classes into three types: *brick:Equipment*, *brick:Location*, and *brick:Point*. For example, we can state that an entity belonging to *brick:Air_Handling_Unit* is a subclass of *brick:Equipment*. This equipment can be located in a *brick:Room* which is a subclass of *brick:Location*. An air handling unit can contain a *brick:Supply_Air_Temperature_Sensor* which is a subclass of *brick:Point*.

The SAREF ontology was initially released in 2014, by the SmartM2M ETSI Technical Committee to describe smart devices in smart homes [12]. Currently, it consists of thirteen modules:

SAREF, SAREF4SYST, and eleven extensions. SAREF is the core module and describes smart devices. SAREF4SYST has adopted the concepts of *seas:SystemOntology*, which defines systems, connections, and connections points. SAREF4BLDG is one of the eleven extensions that extend the building domain and describes the IFC taxonomy of building devices in OWL [13].

As part of the building operation phase, SAREF and BRICK focus on active components and their relationships to sensor points. Passive components such as segments and fittings and the flow of matter, energy, and data are not represented in SAREF and BRICK, which is essential information for the Mechanical, Electrical, and Plumbing (MEP) and HVAC engineers over the whole life cycle and especially during the design phase of building service systems. TSO and FSO are introduced [2, 4] to fill this research gap and to describe the connectivity between systems and components, including active and passive components with the focus on the design phase.

## 3. TSO and FSO

TSO aims to explicitly define the hierarchical, structural, and functional aspects of interconnected building service systems in the AECO industry and their relationships to spatial entities throughout their whole life cycle. The version 0.2.0 was published in [3]. At the current time, TSO is available in version 0.3.0. It is documented and available according to best practice via its URI https://w3id.org/tso and the containing concepts are defined in the namespace https://w3id.org/tso#, which will be abbreviated as *tso:* in the following examples. TSO has 40 classes and 101 object properties. The main classes are *tso:Zone*, *tso:System* and *tso:State*.

*tso:Zone* has a strong alignment to *bot:Zone* and is defined as an *owl:equivalentClass* in the given alignment. *tso:System* is defined as a model of a whole which is isolated from the world or a supersystem, which consists of interconnected components or subsystems and has links between attributes such as inputs, outputs, and states. To represent different states of systems and add a level of abstraction, *tso:State* is defined as the internal condition of a planned or abstract system. This includes specific aspects as on, off, open or closed as well as general aspects such as outdoor-air-operation, mixed-air-operation or heating-operation. These main concepts and some of the object properties between them are shown in Figure 1.

FSO aims to describe the matter and energy flow between systems and components, and the composition of such systems [4]. To that end, FSO consists of 14 classes and 23 object properties. The ontology is available and documented at https://w3id.org/fso, and defines concepts in the namespace https://w3id.org/fso#, later abbreviated as *fso:*. Central classes include the disjoint *fso:System* and *fso:Component*, and the rest are subclasses of these two.

An *fso:System* is defined as a collection of components that can have attributes such as design properties attached to it. Instances of *fso:Component*, on the other hand, are the tangible components that participate in the flow of energy or matter. The subclasses of *fso:System* include more specific nomenclature such as *fso:DistributionSystem*, while subclasses of *fso:Component* include IFC-derived abstract component classes such as *fso:EnergyConversionDevice* and *fso:Segment*.

The object properties of FSO enable the hierarchical decomposition of systems into subsystems and their components, expressing the connectivity and direction of fluids and heat, and designating components as "sources" and "sinks" of systems. FSO components are aligned with

SAREF4BDLG. Further, the hierarchical composition of systems and high-level connectivity of components and systems is aligned with SAREF4SYST. While FSO is not explicitly restricted to describing flow systems *in buildings*, the current use cases are limited to those.

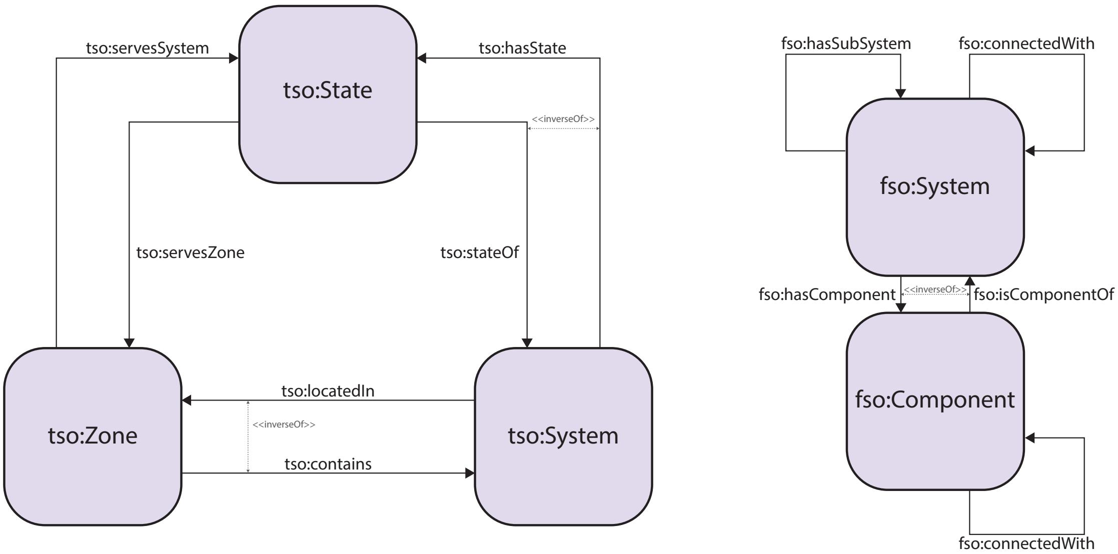

**Figure 1:** Main concepts of TSO and FSO

To compare the given ontologies regarding the representation of building service systems in the AECO industry, the simplified application example involving a six-way valve to simulate a complex real life situation is given. The six-way valve was chosen as it demonstrates complex connectivity of components. The valve is shown with context in Figure 2 and the following competency questions are used. The competency questions were chosen to exercise a set of core requirements as defined by the authors, covering hierarchical, topological, and functional aspects of systems.

- CQ1: Which supersystems does a component have?
- CQ2: What connections to other components does a component have?
- CQ3: What (matter/energy/data) can a component supply to downstream components?

The example consists of two spaces: an office and a conference room. Both are served by an Active Chilled Beam (ACB) as part of the ventilation, heating and cooling systems. Upstream of the ACBs there are two six-way valves, two pumps, two heat exchangers and pipes connecting those components. An IFC-based visualization of the application example, representations using TSO and FSO, as well as SPARQL queries to answer the CQs are given at https://bs-visualizer.web.app

### 3.1. Hierarchical Concepts

To represent the hierarchical aspects of building service systems and answer CQ1, the four subclasses *tso:IntegratedSystem*, *tso:FunctionalSystem*, *tso:TechnicalSystem* and *tso:Component* of

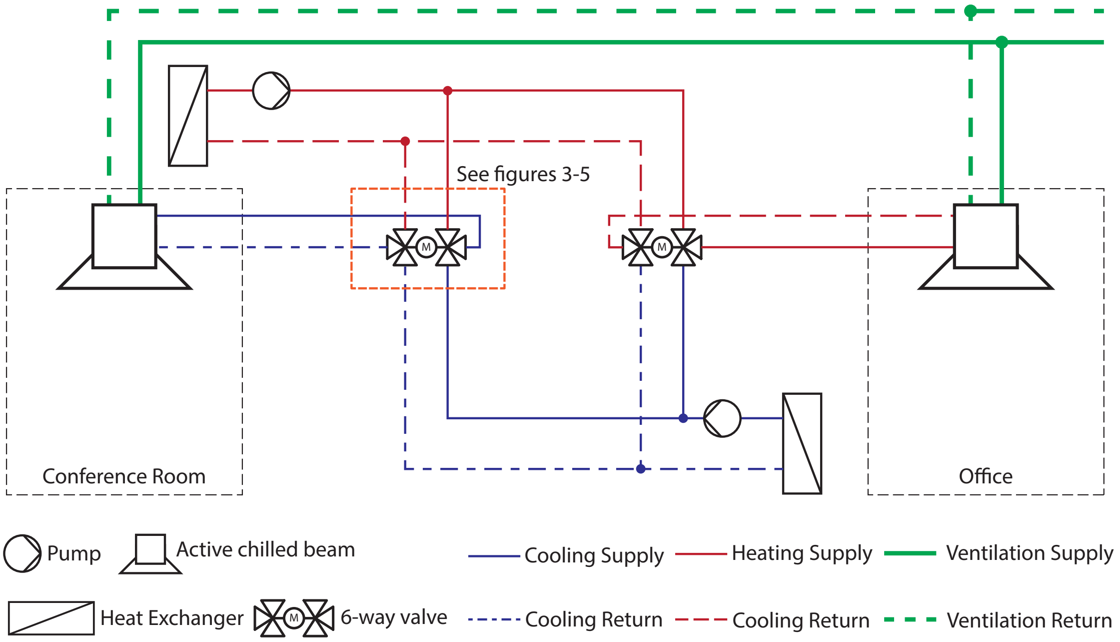

**Figure 2:** Simplified application example

*tso:System* are given in TSO. In the following the term “system" is used to describe all of them. *tso:IntegratedSystem* represents the coupling of different functional systems with independent inherent functions which are interconnected. *tso:FunctionalSystem* denotes a system that is defined by its overall inherent function. Typical examples would be *tso:HeatingSystem*, *tso:CoolingSystem* or *tso:VentilationSystem*. *tso:TechnicalSystem* is defined as a system with a coherent technical solution with which the inherent function (of the upper functional system) is fulfilled. Existing subclasses contain *tso:DistributionSystem* and *tso:ConversionSystem*. *tso:Component* denotes a system, for which the boundary that isolates it from the environment is defined by the manufacturer in terms of the product. Therefore, an air handling unit as well as the included rotary wheel heat exchanger or sensor could be instances of *tso:Component*. To classify components, TSO is aligned with ifcOWL and encourages the use of the international standard DIN EN IEC 81346-2 [14], which is yet to be implemented as an ontology. To link these systems, which represent different levels of hierarchy, the inverse object properties *tso:hasSubSystem* and *tso:subSystemOf*, as well as their eight subproperties (e.g. *tso:hasComponent*), can be used. The domain of this subproperty is defined to *tso:System* and the range to *tso:Component*

Given these concepts, the six-way valve would be an instance of *tso:Component* and be assigned to two instances of *tso:DistributionSystems* by *tso:hasComponent*. These systems are aggregated in two distinct instances of *tso:HeatingSystem* and *tso:CoolingSystem* which are both subsystems of the same instance of *tso:IntegratedSystem*.

In FSO, system hierarchy is modeled through the object properties *fso:hasSubSystem* (inverse *fso:isSubSystemOf*) linking systems to their subsystems, and *fso:hasComponent* (inverse *fso:isComponentOf*) linking systems to individual components. Depending on the use case, a model using FSO would likely have at least two top-level instances of *fso:System*: one for heating and once for cooling. These can be then decomposed to smaller subsystems via *fso:hasSubSystem*, for example, distribution systems as instances of *fso:DistributionSystem*. Finally, the valve could be modeled as an instance of *fso:Component* and assigned as a component of both the distribution systems using *fso:hasComponent*. The hierarchical concepts of TSO and FSO as well as the representation of CQ1 are shown in Figure 3.

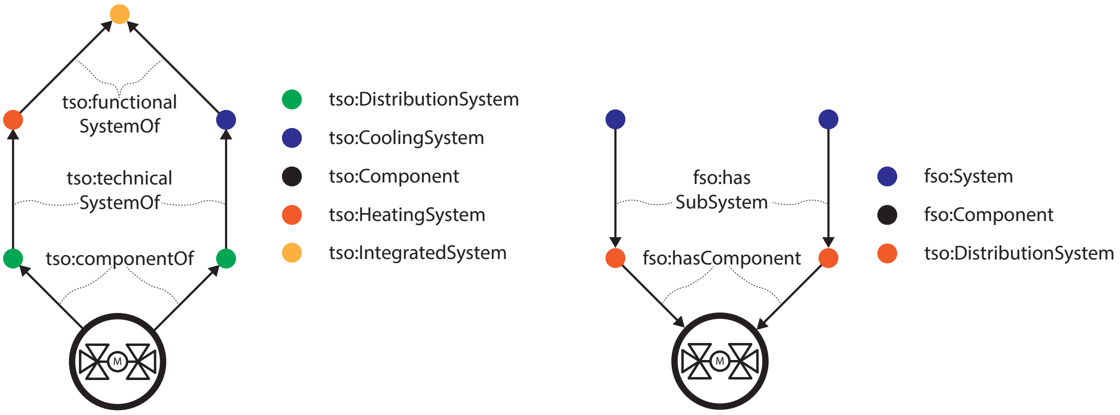

**Figure 3:** Hierarchical concepts and CQ1 in TSO and FSO

In summary, TSO defines four levels of system hierarchy (*tso:IntegratedSystem*, *tso:FunctionalSystem*, *tso:TechnicalSystem* and *tso:Component*) as subclasses of an abstract *tso:System* class and corresponding object properties to describe the composition of building service systems. While in TSO the components are a subclass of system, FSO differentiates between the disjoint *fso:System* and *fso:Component* classes. FSO has no inherent system hierarchy to differentiate between functional or technical systems but uses object properties similar to TSO to decompose systems into subsystems and components.

## 3.2. Topological Concepts

To represent the topological aspects of building service systems and answer CQ2, TSO builds upon the concepts *tso:ConnectionPoint* and *tso:Connection* proposed in [10] and extends these by the subclasses *tso:InnerConnection* and *tso:OuterConnection* to differentiate between the connections inside a system and between different systems. *tso:ConnectionPoint* refers to an inlet or outlet of a system for a connection to other systems or within the same system, where some kind of matter, energy, or data can be transmitted. Using these concepts and the object properties which define the relationships between them, the symmetric object property *tso:connects* can be qualified to describe that two systems are connected. These concepts can be implemented to represent the system topology on every hierarchy level introduced in the last section.

For the example at hand, six instances of *tso:OuterConnections* need to be defined to represent the physical connections of the six-way valve. Each of these instances can be linked via *tso:connectsSystemAt* to a *tso:ConnectionPoint* which is connected via *tso:connectsAt* to the valve. To represent that not all of the six *tso:ConnectionPoints* are interlinked inside the six-way valve, two *tso:InnerConnections* can be defined and linked to the corresponding *tso:ConnectionPoints* using the described object properties.

Unlike TSO, the current version of FSO does not have concepts for connection points or connections. Instead, the top-level symmetric object property *fso:connectedWith* is used to communicate that two components or systems are connected so that they may exchange matter or energy. A central part of FSO is the tree of subproperties under *fso:connectedWith* that enables inferring more generic relationships from more specific ones. As such, the *fso:connectedWith* is not expected to be directly used, but rather inferred from more specific subproperties conveying further functional and logical relationships, such as *fso:suppliesFluidTo* and others, which are discussed in the next subsection with the functional concepts. A visual representation of the CQ2 for both TSO and FSO is given in Figure 4.

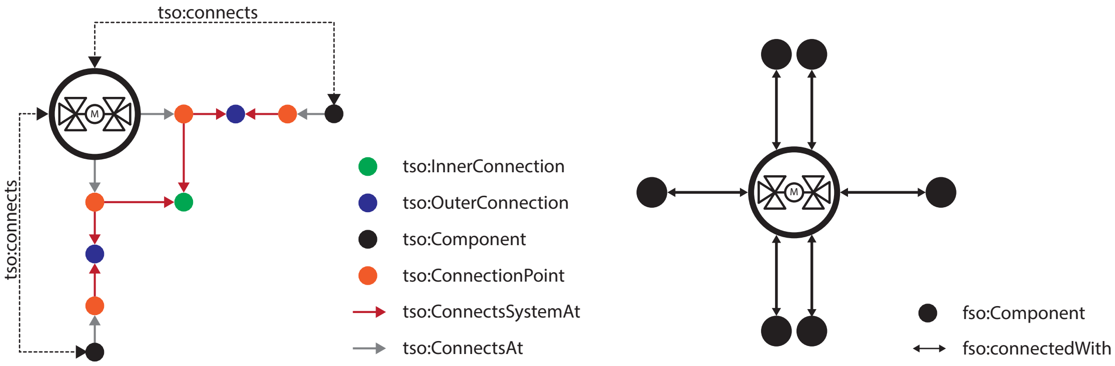

**Figure 4:** Topological concepts and CQ2 in TSO and FSO

In summary, while both ontologies contain a symmetric object property to describe the connection between two systems, TSO further qualifies this connection using the concepts of *tso:ConnectionPoint*, *tso:InnerConnection* and *tso:OuterConnection*.

## 3.3. Functional Concepts

To represent the functional aspects of building service systems and answer CQ3, TSO relies heavily on the concept of states. A *tso:State* is defined as the internal condition of a (planned) system. It can be used to add a level of abstraction to represent systems where the function, and therefore the flow of matter, energy or data, can change due to given states. Hence, a system can have multiple potential states assigned via the *tso:hasState* object property. To represent “what" is exchanged between different systems or inside a certain system, the classes *tso:Matter*, *tso:Energy* and *tso:Data*, as well as multiple subclasses, are defined. These classes can be linked via *tso:hasInput* and *tso:hasOutput* to the states of different systems. Based on these concepts the inverse object properties *tso:supplies* and *tso:suppliedBy*, which link the states directly, can be inferred, as well as their subproperties to denote what is supplied. To represent the flow inside a system, the classes *tso:Matter*, *tso:Energy* and *tso:Data* can be connected to the given state by using the *tso:hasInnerExchange* object property. To qualify the exchange the considered matter, energy or data can be linked to a *tso:Connection* via *tso:transmitsThrough* or to the *tso:ConnectionPoints* via *tso:transmitsFrom* and *tso:transmitsTo*.

To answer CQ3 regarding what the six-way valve can supply to the downstream components and the active cooling beam, two *tso:States* need to be defined and assigned to the six-way valve. Each state is linked to an instance of *tso:Liquid* via *tso:hasInnerExchange* to represent the intended transfer of warm and cold liquid inside the valve through inner connections. As described in the last section these connections range from the connection points linking the pipes of the heating and cooling system to the connection point linking the downstream pipe. The different liquids can be specified using attributes to represent their temperature and further aspects. To represent the transfer to the downstream pipe the instances of *tso:Liquid* are linked to the states of the six-way valve via the *tso:hasOutput* object property and to the state of the downstream pipe via the *tso:inputOf* object property. Therefore, *tso:suppliesLiquid* which links those states directly can be inferred. To answer CQ3 you can follow this property path or, since there is no conversion process between the six-way valve and the active chilled beam, the two liquids can be linked directly to the beam via the *tso:inputOf* object property. Hence, given the state of the six-way valve or of the upper *tso:TechnicalSystem* it can be distinguished between the intended flow of warm or cold liquid which is supplied to the downstream active cooling beam and thus the supply of the conference room.

Using FSO to answer CQ3 and describe what the six-way valve can supply to the downstream ACB requires the use of the more specific subproperties of *fso:connectedWith*, such as *fso:suppliesFluidTo*. The connectivity in itself only describes that there is an *intended* flow of fluid between the components. One way to deduce further what is supplied downstream of the six-way valve would be to look at what systems and components supply the valve. That is, the question could be rephrased as "what is supplied to the valve", which would then be what the valve could supply downstream. Further adding the concept of *source* and *sink* components of systems with *fso:hasSourceComponent* and *fso:hasSinkComponent*, it could be inferred that both the heating and cooling systems have sources that can supply fluid to the ACB. Additionally, the use of component classes such as *fso:FlowController* and *fso:EnergyConversionDevice* can be used to indicate and deduce the intended function of connected components to an extent. An abbreviated representation of the functional concepts regarding the 6-way valve from both TSO and FSO is shown in Figure 5.

In summary, the ontologies strongly differ in regards to the representation of the functional concept. To describe the intended flow, TSO introduces the concept of states and defines matter, energy, and data as classes, which can be linked to the states. The object properties, which link the states of the systems directly, can be inferred. On the other hand, FSO represents the intended transfer of fluids, heat, or electric charge between different systems via object properties that link these systems directly. As such, FSO does not consider the fact that the internal connectivity and function of systems and therefore the flow of matter, energy, and data can vary depending on aspects such as control values, while TSO enables expressing this with states.

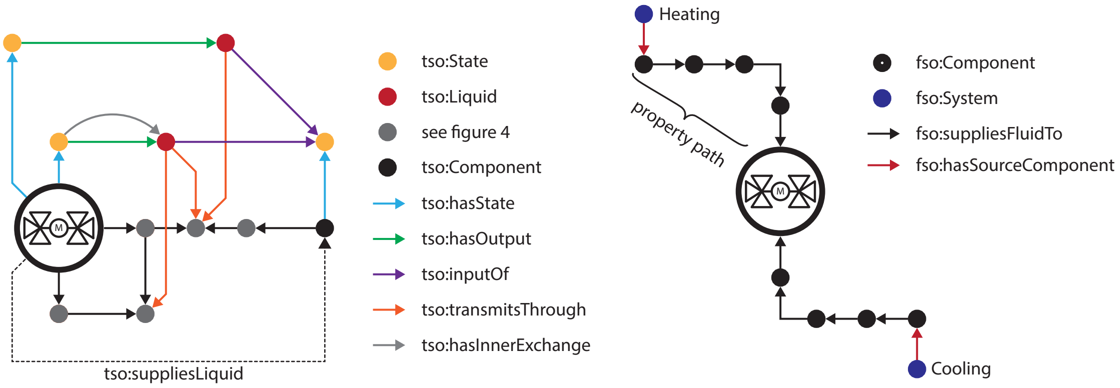

**Figure 5:** Functional concepts and CQ3 in FSO and TSO

### 3.4. Alignment

In order to concretely evaluate and describe the current compatibility of the two ontologies, an alignment of them is presented to the extent possible. Several concepts can be used to set up an alignment for TSO v0.3.0 and FSO v0.1.0. Distribution Systems, Supply Systems, and Return Systems are described as equivalent since they represent a nomenclature for systems on a certain level of hierarchy in both ontologies. *fso:System* is defined as a subclass of the abstract *tso:System* with *fso:isSubSystemOf* and *fso:isComponentOf* as subproperties of *tso:subSystemOf* - the inverse properties *fso:hasSubSystem* and *fso:hasComponent* being subproperties of *tso:hasSubSystem*. A Component in FSO is equivalent to the one in TSO, but a *tso:Component* is always an abstract *tso:System* in itself, while *fso:Component* and *fso:System* are disjoint classes.

A topological connection is categorized with *tso:connects* and *fso:connectedWith* respectively, hence they are defined as equivalent properties. Further topological concepts cannot be matched, since they do not exist in both TSO and FSO. The functional concepts are too diverse to propose a direct alignment between the two ontologies. The alignment is summarized in table 1.

## 4. Roadmap to a unified representation

A unified representation for interconnected building service systems which combines the concepts of TSO and FSO needs the expressiveness to model those systems throughout their whole life cycle and the simplicity to make them usable in day-to-day operations. Since the complexity and general structure of building service systems differ widely regarding the various disciplines and building types that are served by the systems, a modular ontology structure needs to be implemented.

The (lightweight) core module(s) need to contain hierarchical, topological, and functional aspects, which are valid for all disciplines. This includes classes such as System and State as well as object properties as connected and supplied. The degree to which these aspects should be included is yet to be defined based on application examples ranging various trades and levels of complexity and in communication with approaches such as SEAS [10], BRICK [11], and

**Table 1**
Alignment between FSO v0.1.0 and TSO v0.3.0

| FSO | | TSO |
|---|---|---|
| *fso:System* | *rdfs:subClassOf* | *tso:System* |
| *fso:DistributionSystem* | *owl:equivalentClass* | *tso:DistributionSystem* |
| *fso:SupplySystem* | *owl:equivalentClass* | *tso:SupplySystem* |
| *fso:ReturnSystem* | *owl:equivalentClass* | *tso:ReturnSystem* |
| *fso:Component* | *owl:equivalentClass* | *tso:Component* |
| *fso:isSubSystemOf* | *rdfs:subPropertyOf* | *tso:subSystemOf* |
| *fso:isComponentOf* | *rdfs:subPropertyOf* | *tso:subSystemOf* |
| *fso:hasSubSystem* | *rdfs:subPropertyOf* | *tso:hasSubSystem* |
| *fso:hasComponent* | *rdfs:subPropertyOf* | *tso:hasSubSystem* |
| *fso:connectedWith* | *rdfs:equivalentProperty* | *tso:connects* |

SAREF [12].

Further hierarchical, topological, and functional aspects which are valid for all disciplines but should not be included in the core module(s) for the sake of usability could be defined in a hierarchical, topological and functional ontology pattern, which enhances the expressiveness of the unified representation. The hierarchical pattern could include different levels of system hierarchy and the object properties to link these. Concepts such as connection points and connections could be contained in the topological pattern and the definition of different forms of matter, energy, and data as well as properties describing the input and the output of systems could be included in the functional ontology pattern.

Classifications of systems and concepts which are necessary to describe specific aspects of disciplines could be defined in separate domain ontologies. These should not describe concepts out of the scope of building service systems but to be aligned with existing approaches such as BOT [9] which reached are high level of shared conceptualization in the context of linked data in the AECO industry.

## 5. Conclusion

This paper presented a structured comparison of the two recent ontologies aimed at describing building service systems and their flow of matter, energy, and data: the TUBES System Ontology (TSO) and the Flow Systems Ontology (FSO). The ontologies were compared in terms of their overall goal and size and through the lenses of hierarchical, topological, and functional concepts. It was shown that while the ontologies have similar goals, their conceptualizations differ substantially. TSO is a considerably larger ontology, enabling more detailed descriptions of interconnected systems and their functional relationships based on their states. FSO offers a more limited set of concepts and properties and lacks particularly in terms of qualifying connections through concepts such as connections and connection points. Further, FSO does not enable the description of flow states that may vary, such as in the example of a six-way valve. On the other hand, the expressivity of TSO comes at the cost of complexity, and a more modular approach could make it more useable.

High-level alignments for the ontologies were proposed, which are admittedly rather superficial. This is primarily due to the incompatible functional concepts of the two ontologies, and is further complicated by the disjointness of *fso:System* and *fso:Component*. Finally, the roadmap towards future unification was presented, with ideas for further development and structuring of the ontologies to support shared conceptualizations.

The structured comparison of the two ontologies highlighting their respective strengths and weaknesses is useful for future efforts to refine the ontologies or build new ones. Aligning the ontologies and showing the friction points for the alignment is useful for future refinements to make the ontologies more compatible. By proposing a roadmap for the unification of the included concepts in TSO and FSO the authors pave a path for a harmonized and shared ontology that better serve use cases related to building service systems.

## Acknowledgments

The research within the project EnergyTWIN leading to these results has received funding from the German Ministry for Industry and Energy under grant agreement no. 03EN1026A.

## References

[1] P. Pauwels, A. Costin, M. H. Rasmussen, Knowledge Graphs and Linked Data for the Built Environment, in: M. Bolpagni, R. Gavina, D. Ribeiro (Eds.), Industry 4.0 for the Built Environment: Methodologies, Technologies and Skills, Structural Integrity, Springer International Publishing, Cham, 2022, pp. 157–183. doi:10.1007/978-3-030-82430-3_7.

[2] N. Pauen, D. Schlütter, J. Siwiecki, J. Frisch, C. van Treeck, Integrated representation of building service systems: Topology extraction and TUBES ontology, Bauphysik 42 (2020) 299–305. doi:10.3217/978-3-85125-786-1-59.

[3] N. Pauen, D. Schlütter, J. Frisch, C. van Treeck, TUBES System Ontology: Digitalization of building service systems, in: Proceedings of the 9th Linked Data in Architecture and Construction Workshop, 2021.

[4] V. Kukkonen, A. Kücükavci, M. Seidenschnur, M. H. Rasmussen, K. M. Smith, C. A. Hviid, An ontology to support flow system descriptions from design to operation of buildings, Automation in Construction 134 (2022) 104067. URL: https://doi.org/10.1016/j.autcon.2021.104067. doi:10.1016/j.autcon.2021.104067.

[5] J. Beetz, J. Van Leeuwen, B. De Vries, IfcOWL: A case of transforming EXPRESS schemas into ontologies, Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM 23 (2009). doi:10.1017/S0890060409000122.

[6] J. Flore, T. Djuedja, Integration of environmental data in BIM tool & Linked Building Data, in: Proceedings of the 7th Linked Data in Architecture and Construction Workshop, 2019.

[7] A.-H. Hamdan, M. Bonduel, R. J. Scherer, An ontological model for the representation of damage to constructions, in: Proceedings of the 7th Linked Data in Architecture and Construction Workshop, 2019.

[8] M. Niknam, S. Karshenas, A shared ontology approach to semantic representation of bim data, Automation in Construction 80 (2017) 22–36. URL: https://www.sciencedirect.com/science/article/pii/S0926580517302364. doi:https://doi.org/10.1016/j.autcon.2017.03.013.

[9] 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 (2021) 143–161. doi:10.3233/SW-200385.

[10] M. Lefrançois, Planned ETSI SAREF extensions based on the W3C & OGC SOSA/SSN-compatible SEAS ontology patterns, CEUR Workshop Proceedings 2063 (2017).

[11] B. Balaji, A. Bhattacharya, G. Fierro, J. Gao, J. Gluck, D. Hong, A. Johansen, J. Koh, J. Ploennigs, Y. Agarwal, M. Bergés, D. Culler, R. K. Gupta, M. B. Kjærgaard, M. Srivastava, K. Whitehouse, Brick: Metadata schema for portable smart building applications, Applied Energy 226 (2018) 1273–1292. URL: https://doi.org/10.1016/j.apenergy.2018.02.091. doi:10.1016/j.apenergy.2018.02.091.

[12] L. Daniele, F. den Hartog, J. Roes, Created in Close Interaction with the Industry: The Smart Appliances REFerence (SAREF) Ontology, in: Lecture Notes in Business Information Processing, volume 225, 2015. doi:10.1007/978-3-319-21545-7_9.

[13] M. Poveda-Villalón, R. García-Castro, Extending the SAREF ontology for building devices and topology?, CEUR Workshop Proceedings 2159 (2018) 16–23.

[14] IEC: International Electrotechnical Commission, DIN EN IEC 81346-2: Industrial systems, installations and equipment and industrial products – structuring principles and reference designations – Part 2: Classification of objects and codes for classes, 2020.