Article

A Semantic Approach for Building System Operations: Knowledge Representation and Reasoning

Parastoo Delgoshaei [1,*]**, Mohammad Heidarinejad** [2] **and Mark A. Austin** [3]

1   Mechanical Systems and Controls Group, Engineering Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA
2   Department of Civil, Architectural, and Environmental Engineering, Illinois Institute of Technology, Chicago, IL 60616, USA; muh182@iit.edu
3   Department of Civil and Environmental Engineering, University of Maryland, College Park, MD 20742, USA; austin@umd.edu
*   Correspondence: parastoo.delgoshaei@nist.gov

Abstract: Artificial intelligence is set to transform the next generation of intelligent buildings through the application of information and semantic data models and machine learning algorithms. Semantic data models enable the understanding of real-world data for building automation, integration and control applications. This article explored the use of semantic models, a subfield of artificial intelligence, for knowledge representation and reasoning (KRR) across a wide variety of applications in building control, automation and analytics. These KRR-enabled applications include context-aware control of mechanical systems, building energy auditing and commissioning, indoor air monitoring, fault detection and diagnostics (FDD) of mechanical equipment and systems and building-to-grid integration. To this end, this work employed the Apache Jena Application Programming Interface (API) to develop KRR and integrate it with some domain-specific ontologies expressed in the Resource Description Framework (RDF) and Web Ontology Language (OWL). The ontology-driven rules were represented using Jena rule formalisms to enable the inference of implicit information from data asserted in the ontologies. Moreover, SPARQL (SPARQL Query Language for RDF) was used to query the knowledge graph and obtain useful information for a variety of building applications. This approach enhances building analytics through multi-domain knowledge integration; spatial and temporal reasoning for monitoring building operations, and control systems and devices; and the performance of compliance checking. We show that existing studies have not leveraged state-of-the-art ontologies to infer information from different domains. While the proposed semantic infrastructure and methods in this study demonstrated benefits for different building applications applicable to mechanical systems, the approach also has great potential for lighting, shading and security applications. Multi-domain knowledge integration that includes spatial and temporal reasoning allows the optimization of the performance of building equipment and systems.

Keywords: building management system; context-aware control; knowledge representation and reasoning; intelligent buildings; Web Ontology Language

1. Introduction

Modern buildings are complex multi-disciplinary systems that need to operate in an efficient and sustainable manner. To formally reason about building performance and sustainability, underlying models need to include material and geometry properties as well as their dependencies. Notions of sustainability evolve from processes that seek to incorporate and evaluate a broad range of stakeholder and operator concerns over the complete building system lifecycle. Given the growing necessity of the design of sustainable buildings, and in support of these ends, we seek data-driven approaches to the measurement of performance in the building environment and identification of trends and patterns in behavior. Long-term solutions need to account for the unique physical, economic and social

characteristics of individual buildings and building systems. A key challenge in developing the required capability is finding new ways to work with interacting and interlinked domains that span the building enclosure, mechanical equipment systems and services, stakeholders (e.g., owners, facility managers) and occupants. Unfortunately, state-of-the-art procedures in building control, operation and system simulation fall short in their ability to represent and reason with the knowledge of the interlinked and interacting building domains. This knowledge can be as explicit as the representation of well-known facts but can also include knowledge derived implicitly through chains of reasoning spanning multiple domains.

The World Wide Web is expected to use the Semantic Web to enable a pathway for machine-to-machine interaction and the development of enhanced knowledge-based applications. Ontologies are an essential concept of the Semantic Web. An ontology is a formal and explicit representation of data that conceptualizes information in a particular domain for consistent definition and access [1]. Knowledge graphs are created when ontologies are populated with data points (individuals). During the past twenty years, Semantic Web technologies [2,3] and associated knowledge representation languages and tools have matured to the point that it is now feasible to consider their use in applications beyond those normally associated with computer science. One such opportunity is assistance in the design and operation of large-scale building systems. To this end, a recent study [4] reviewed the state-of-the-art ontologies that can be used for smart and ongoing commissioning (e.g., SAREF, ifcOWL, SEAS). The review was based on common data types used (e.g., energy use, maintenance, building systems and components), and applications (e.g., fault detection and diagnostics), building performance improvement and performance indicator calculations since 2014. This study suggested that future research directions include the identification of data types, assessment of ontology performance and creation of open-source approaches. A second recent study [5] proposed a conceptual system architecture for semantic data mining and linked data for a system that generates recommendations for performance-oriented design in architecture, engineering and construction. Themes that emerge from the literature include the following: (i) semantic queries can enable access to knowledge graphs; (ii) traditional data mining techniques can allow sensor data mining; (iii) the use of a linked building data graph empowers semantic data mining; and (iv) the use of pattern matching in combination with Resource Description Framework (RDF) stream processing can occur with RDF graph mining [5].
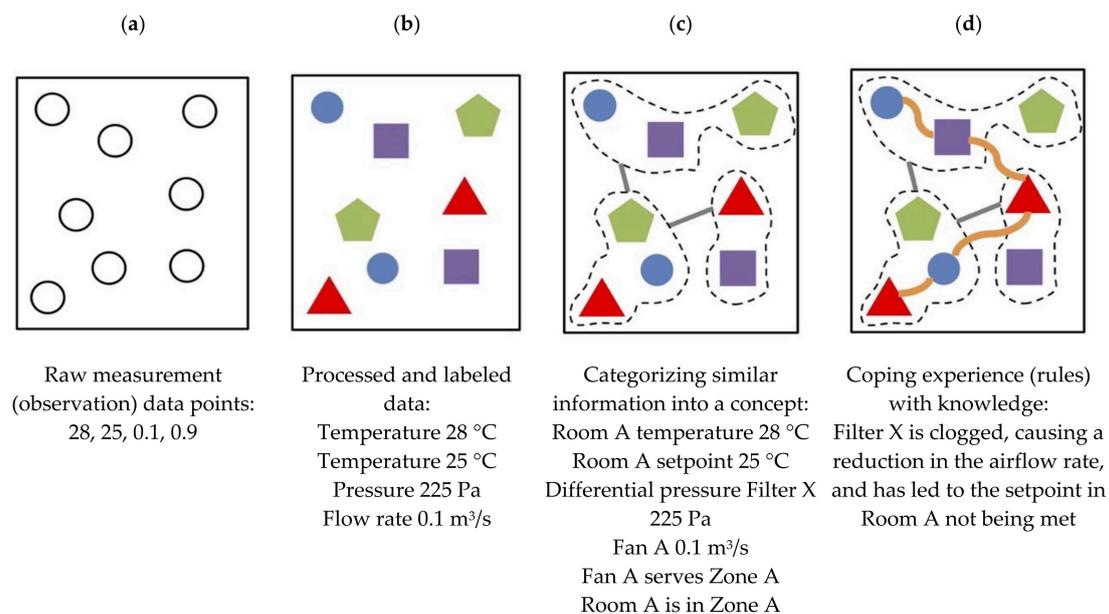
Another set of studies have focused on the investigation and development of ontology-based approaches to building automation systems (BASs) [6–11]. This body of work concludes that a significant barrier to the deployment of many building analytics applications—fault detection and diagnostics (FDD), tools for system-scale optimization, automation-assisted commissioning tools—is the mundane, error-prone process of mapping the building automation system data points to the applicable software tools. These observations point to a strong need for a new generation of software tools that incorporate and work with ontologies and semantic graphs representing equipment, sensors and actuators in a machine-readable format. As an illustration of what is possible, one study used ontologies as a generic application model to facilitate the integration of heterogeneous building automation networks [9]. Their model combined a classical data-driven energy analysis with a knowledge-driven energy analysis through ontologies and rulesets. Their analysis utilized information collected from building automation devices and inferred energy inefficiencies based on the state of those devices and user behavior. In a similar effort, One study proposed an ontology development process and the delivery of an intelligent multi-agent software framework (OntoFM), both designed to support the real-time monitoring of building conditions [6]. Their multi-domain framework weaves together building, sensor and supporting ontologies (i.e., mereology and topology ontologies) to support real-time knowledge querying of the underlying multi-agent framework. A third example introduced an ontology and associated data models to represent and incorporate layers of data from multiple domains, including occupants, indoor and outdoor environmental conditions,

control systems and devices, equipment and energy flows [10]. The goal of their study was to facilitate the collection, storage, sharing and analysis of monitored data spanning building automation, facility management, building diagnostics and building performance simulation. Another study proposed a building management system architecture using ontologies and inferencing engines to gather sensor and equipment state data, decide the status of control commands and operate building systems [7]. Collectively, these studies demonstrate the effectiveness and promise of ontology-based approaches in controlling the behavior of building systems.

1.1. Proposed Approach: Data, Information, Knowledge and Wisdom

We envision the widespread use of data-driven approaches to building performance analysis and the identification of behavioral trends and patterns in buildings. For this vision to work, there needs to be a well-defined pathway from raw building data to decision making supported by the formal use of analytics. It is important to note that present-day data-driven approaches use the data without any intuition; knowledge-based (i.e., ontology-driven) approaches, on the other hand, account for the context of the data. The four key concepts of the proposed approach are data, information (context), knowledge and wisdom (analytics). These concepts are the elements from Ackoff's data-information-knowledge-wisdom pyramid [12] and have been highlighted in many publications [13]. Briefly, data are the product of observations. Information is the processed data. Knowledge further refines information by making the transformation of information into facts possible. Wisdom is the ability to see the long-term consequences of any act.

To demonstrate the use of the four key concepts in the proposed knowledge-based approach, Figure 1 shows the relationships between data, information, knowledge and wisdom for a fault detection application. It demonstrates how wisdom can help in troubleshooting the airside of forced air heating, ventilating and air conditioning (HVAC) equipment. Forced air HVAC systems usually utilize air filters to protect equipment and improve indoor air quality [14]. After the manufacturer's recommended number of hours of operation, clogged or loaded filters can lead to a significant reduction in the airflow rate and energy efficiency of the system. Without filter replacement, especially during the summer and winter seasons, there is the possibility of equipment damage with continued operation [15].

(a)	(b)	(c)	(d)
Raw measurement (observation) data points: 28, 25, 0.1, 0.9	Processed and labeled data: Temperature 28 °C Temperature 25 °C Pressure 225 Pa Flow rate 0.1 m³/s	Categorizing similar information into a concept: Room A temperature 28 °C Room A setpoint 25 °C Differential pressure Filter X 225 Pa Fan A 0.1 m³/s Fan A serves Zone A Room A is in Zone A	Coping experience (rules) with knowledge: Filter X is clogged, causing a reduction in the airflow rate, and has led to the setpoint in Room A not being met

Figure 1. Knowledge-based fault detection of a forced air HVAC system with a clogged filter: (**a**) data, (**b**) information, (**c**) knowledge and (**d**) wisdom.

Figure 1a shows the "data" for this example—numeric values with no labels, quantity type or units (e.g., 25, 28). Information is created when the raw data are processed and structured, which in this case means the assignment of temperature units to the raw data values of 25 and 28. Additionally, notice that, at this stage, different entity types can be represented symbolically by different colored shapes. Knowledge representation (see Figure 1c) entails encapsulating the properties (i.e., isolated information) into concepts (i.e., dotted shapes) of the domain and describing how those concepts are related (gray connection lines). These relationships can indicate, for example, that Fan A serves Room A. Wisdom is gained through evaluation and rule checking about relationships in the knowledge base (i.e., see orange connection lines, Figure 1d). In this case, for example, the wisdom reveals the setpoint in Room A is not being met due to a clogged filter.

1.2. Scope and Objectives

Our work is motivated by the tenet that semantic-based frameworks offer a multiplicity of benefits including, but not limited to, an automated pathway from data to wisdom, coupled with support for big data representation and reasoning operations. The fields of big data and data-driven decision making have transformed high-performance building operations by offering meaningful insight into building conditions and equipment health and behavior. As such, semantic frameworks of the type proposed can support emerging applications of BASs, such as the ability to understand, reuse and expand underlying knowledge. The proposed approach leverages the Description Logic (DL) for implementing sequences of operation (SOOs) across a variety of building systems (e.g., HVAC, lighting). Further, it serves as a gateway to smart buildings (and cities) supported by the Internet of Things (IoT). Thus, the goals of this paper are three-fold:

1. To demonstrate the application of formal information models and knowledge structures that represent the underlying knowledge bases of building systems;
2. To present a roadmap for integration of ontologies and rulesets with advanced control strategies and state-of-the-art system simulation models;
3. To reuse and expand existing building domain-specific and foundational ontologies (i.e., Time and QUDT) for different building applications.

The remainder of this paper proceeds as follows: Section 2 introduces and describes Semantic Web technologies. Collections of ontologies that enable the proposed approach are covered in Section 3. Methods and a range of building applications (e.g., detection and diagnostic analysis of faults) are introduced in Sections 4 and 5, respectively. This paper concludes with Sections 6 and 7.

2. Semantic Web Technologies

The goal of the Semantic Web is to give information a well-defined meaning, thereby creating a pathway for machine-to-machine interaction and automated services based on descriptions of semantics [2]. The Semantic Web Technology Layer Cake (see Figure 2) illustrates the technical infrastructure that accomplishes these purposes. The lower layers support addressing resources on the Web and linking documents. For example, Extended Markup Language (XML) is designed for storing and transferring data into tree structures. Unfortunately, tree structures are incapable of integrating data from multiple sources. To overcome this problem, RDF is a framework that can represent data resources on the Web in graph structures; now, XML documents from multiple sources can be gathered and integrated into a graph format. One shortcoming of RDF is that it has no semantic restrictions, and any statement that follows the triple structure is valid. This makes it possible to assert statements in the model that are semantically wrong. An RDF Schema (RDF-S) takes a step toward solving this problem by providing the basic vocabulary for RDF statements and the constructs to create hierarchies of classes and properties.
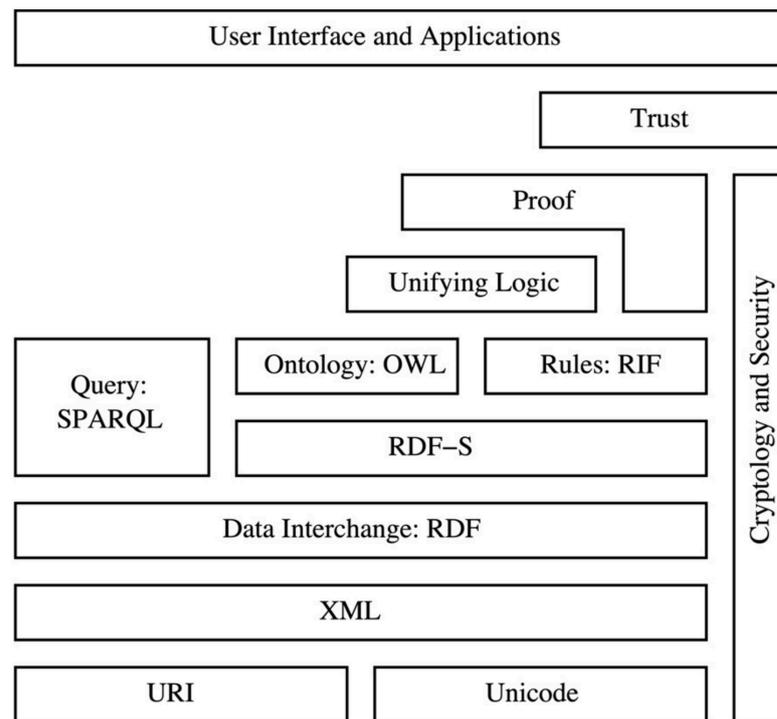
Figure 2. Technologies in the Semantic Web Layer Cake (figure is redrawn from [1]).

The Web Ontology Language (OWL) is a description logic-based (DL-based) mathematical theory language with the highest level of expressivity for constructing ontologies. OWL enhances expressiveness and support for richer property declarations (e.g., transitivity), class property restrictions (e.g., someValuesFrom), quality cardinality restrictions (e.g., owl:maxCardinality), equality between classes (e.g., sameAs) and relations between classes (complementOf) [16]. These additional capabilities allow software applications to use reasoning to infer new facts (triples) from existing ones with first-order logic (FOL) as a mathematical foundation.

As mentioned in Section 1, in addition to ontologies, Semantic Web standards also support languages for describing ontology-driven rules (i.e., mechanisms to infer data) and queries (i.e., mechanisms to access data). For example, the shapes constraint language (SHACL) and SPARQL are semantic rule and query languages for ontologies, respectively. SHACL is a relatively recent W3C (World Wide Web Consortium) recommendation for defining constraints that can be validated against RDF graphs. The interactions between SHACL and other Semantic Web technologies are a matter of ongoing research [17]. The higher layers of the Semantic Web infrastructure include the mechanisms to ensure the proof and trustworthiness of models. Trust ensures that the source is trustworthy, and proof checks that the data come from the original source [18].

3. State-of-the-Art Building Ontologies

This section provides an overview of state-of-the-art ontologies (many still being actively developed and extended) that can be used to enhance building automation, data integration, analytics and decision making. The ontologies reviewed include: (i) SAREF4BLDG, (ii) RealEstateCore, (iii) SOSA (Sensor Observation Sampling Actuation), (iv) QUDT (Quantity, Unit, Datatype), (v) Brick Schema, (vi) BOT (Building Topology Ontology) and (vii) Time. Some of these ontologies are designed specifically for buildings, i.e., Brick, while the others, such as the Time and QUDT ontologies, represent foundational concepts that apply to multiple domains. Table 1 summarizes these ontologies.

Table 1. A summary of existing ontologies that can be used within the building domain.

Name	Ontology Description	URLs to the Ontologies
SAREF4BLDG	An ontology pattern for systems, connections and connection points	https://saref.etsi.org/saref4bldg (accessed on 1 November 2021)
RealEstateCore	An ontology for smart buildings, based on established practices in real estate	https://w3id.org/rec/full/ (accessed on 1 November 2021)
SOSA (Sensor Observation Sampling Actuation)	Semantic description of sensors and observations	https://www.w3.org/TR/vocab-ssn/ (accessed on 1 November 2021)
QUDT (Quantity, Unit, Datatype)	Semantic description of units of measure, quantity type, dimensions and data types	http://qudt.org/2.1/schema/qudt (accessed on 1 November 2021)
Brick Schema	Semantic description of physical, logical and virtual assets in buildings and the relationships between them	https://brickschema.org/schema/Brick (accessed on 1 November 2021)
BOT (Building Topology Ontology)	Semantic description of concepts in buildings such as sites, zones and rooms	https://w3c-lbd-cg.github.io/lbd/bot/ (accessed on 1 November 2021)
Time Ontology	Semantic description of time	https://www.w3.org/TR/owl-time (accessed on 1 November 2021)

RealEstateCore is an ontology for smart buildings derived from established practices in real estate and construction. It provides building owners and operators a platform for the management, storage and sharing of building data. RealEstateCore is a modular ontology, that is, a collection of data schemas that describe concepts and relations for the data related to modeling buildings and building systems, or that are sourced from such systems [19].

SAREF4BLDG was developed based on the Industry Foundation Classes (IFC) standard for building information and topology. The scope of this extension includes concepts around devices and appliances and their location within the building [20].

The SOSA ontology is a lightweight semantic core module that specifies the semantics of sensors, observations, samplers and actuators. A more expressive extension module is called the SSN ontology (Semantic Sensor Network). The combined SOSA/SSN model provides a semantic description of systems of sensors and actuators, observations, measurement procedures, the subjects and their properties being observed or acted upon, samples and the process of sampling [21]. SOSA and SSN are now widely accepted as standards and are W3C recommendations.

QUDT is a semantic model for the various Quantity, Unit and Datatype standards. Operations supported by this model include: mechanisms for conversion between single and complex unit types, dimensional analysis of equations, determination of equivalent units in different systems of units and determination of equivalent quantity types in different systems of quantities [22].

Brick Schema is a collection of semantic representations for the physical, logical and virtual assets in buildings. The Brick Schema defines a concrete ontology for sensors, subsystems and relationships among them. Moreover, it enables cross-vendor representation of the various subsystems in modern buildings in the different domains of HVAC, lighting, fire and security [23]. One study tested Brick for building management systems (BMSs) in six diverse buildings with sensors and equipment from different vendors [24]. The test coverage systematically included a range of completeness and expressiveness criteria to represent all metadata information (e.g., location of sensors, type contained in the building's BMS) and capture all important relationships between data points (e.g., explicit or implicit) mentioned in a building BMS.

BOT is a core ontology for describing the topological concepts of a building. It is a minimal ontology for defining relationships between the sub-components of a building but can be extended with more domain-specific ontologies [25]. For example, EEPSA (Energy Efficiency Prediction Semantic Assistant) takes advantage of Semantic Web technologies to represent knowledge in the domain of energy efficiency and thermal comfort in buildings. This approach allows for the customization of ontologies for similar problems in different

types of buildings. The EEPSA and BOT ontologies were tested for an office building and a farm building to help with knowledge discovery in databases (KDD) [26,27].

Mechanisms for the representation of points, intervals of time and time-based reasoning are handled by the cross-disciplinary Time ontology and backend supporting rules and software. The Time ontology encapsulates a semantic definition of time (e.g., instance, interval), temporal properties (e.g., begins, ends) and relationships among them. To support time-based reasoning, the model stores the topological (ordering) relationships among instants and intervals, and information about the temporal position including date–time information. This collection of features is a key enabler of building analytics working with time series data [28]. One study used the Time ontology in their framework to support decision making in cyber-physical systems (CPSs) [29].

A few recent studies have focused on the use of the ifcOWL ontology, which is the transformation of the IFC schema to an OWL representation (i.e., ifcOWL is the OWL representation of the IFC schema). The IFC standard is designed to represent building architecture and construction data [30]. In one study, ifcOwl was used to query a wall panel production model (e.g., activity sequence and cost estimation) [31], and in other studies, it was used to modify an existing geometry model [32,33]. Another study employed the IFC model to convert building data into a node2vec graph model designed to capture semantic similarities among different building components [34]. Raghavi and Gowtham summarized current efforts to integrate the BIM (building information model) with external domain querying and reasoning systems to retrieve details of a building from the underlying building model [35]. The goal of the work was to make the IFC model available to domains other than architecture, construction and engineering. As a case in point, SimpleBIM is an open BIM-ifc application that is designed to extract information from ifcOWL to enrich BIM models [36].

These state-of-the-art ontologies can be leveraged to develop an integrated roadmap. The survey in [27] reviewed and compared existing ontologies for observation and sensing using a set of competency questions to verify the models in a simple use case. Ontologies were selected based on factors such as online availability, existing metadata and documentation. Many of the ontologies covered in the review are well established and actively maintained by important consortia: ifcOWL [33] of BuildingSMART, BOT [25] of the W3C Linked Building Data community group, SAREF4BLDG of ETSI SmartM2M and Brick. The authors in [37] proposed an alignment of BOT to five domain ontologies: SAREF, ifcOWL, Brick, DogOnt, ThinkHome. Consequently, this study used a framework to integrate different ontologies from different domains for knowledge representation and to develop rulesets across domains to perform reasoning.

Project Haystack was initially proposed as a tagging system to capture metrology building information in a lightweight strategy [38]. In the work documented in [39], the authors argued for a formal semantic model by identifying a family of inherent interpretability and consistency issues in the tagging model that emerge due to the lack of a formal definition in Project Haystack. It is important to note that Haystack 4 implements a graph model to store the Haystack tags within Haystack systems [40].

To summarize, the smart building domain spans concepts that include devices, observations, occupants, building physical information and units of measurements. While some ongoing studies in the AEC and facilities management (FM) communities focus on the use of a specific ontology to capture knowledge for a specific use case, others determine the correspondence between the concepts of different ontologies for alignment or matching.

The focus of the state-of-the-art literature is on using the ontologies as a taxonomy that can replace semantic tagging. However, the current literature does not explore how the represented knowledge and reasoning capabilities can be used as powerful tools for different building applications. Understanding the relationship between the heterogeneous data from different domains is one of the key aspects of enhancing design, maintenance and other building analytics applications, including, but not limited to, HVAC, lighting, access control, shading devices, FDD and IAQ monitoring. Existing knowledge gaps from this

literature review indicate that, first, in most cases, only one or a limited number of domain-specific ontologies have been used to represent the data. This is partly because there are many overlapping and often conflicting concepts in the different ontologies requiring alignment between the concepts of the ontologies. Second, the studies did not focus on how these ontologies and the reasoning and querying capabilities can assist or enhance a variety of building applications. This paper explored how knowledge graphs can be used to integrate data from multiple sources to enhance semantic definitions, FDD, spatial reasoning, temporal reasoning, asset management and maintenance and context-aware control. While this paper primarily demonstrates the proof of concepts for HVAC systems, these enhancements are applicable to other building domains. To address the issues from the identified knowledge gaps, the semantic framework aims to:

- Serve as conceptual modeling for buildings and their systems;
- Provide easier data interoperability at a semantic level from multiple domains (i.e., occupant, building, weather, equipment);
- Enable the reuse of existing best practice schemas and standards.

4. Proposed Semantic Infrastructure and Methods

The examples in this work utilize proposed languages from W3C standards for constructing, constraining and querying the ontologies for different building applications. From a productivity standpoint, these standards are designed to enhance the heterogeneous data in such a way that they can be used for effective discovery, automation, integration and reuse across multiple domains in various building applications.

To achieve these purposes, the proposed semantic infrastructure is organized into three layers: (1) ontology—represents a snippet of the domain knowledge, including the relevant concepts and their relationships; (2) rules—represent the mechanisms to formally infer new information based on the existing data across multiple domains; and (3) queries—enable the retrieval of both explicitly and implicitly derived information. The framework employs Apache Jena—referred to simply as Jena [41]—for ontology creation and reuse. Jena is an open-source Java [42] framework for creating Semantic Web applications. It supports forward chaining and back chaining rule engines and is targeted mainly at RDF processing. Therefore, in this work, RDF was used to describe the ontology, and Jena was used to describe the rules. Furthermore, SPARQL was used to express queries against our model [43].

This work leveraged some of the existing building ontologies discussed in Section 3 to represent knowledge across different domains of the building (i.e., sensors, equipment, spaces, occupancy). The goal is to demonstrate that these multi-domain semantic models simplify deployment in building control, analytics, automation and integration applications while also lowering operational costs. We consider the following definitions:

- Facts are logical statements that specify the relationships between properties and/or classes in the domain.
- Rules are mechanisms that allow for the specification of additional logical constraints. They explain the conditions under which new implicit knowledge can be derived from existing facts that are stored in the ontology, such as "if <Axiom conditions> then <Axiom consequence>".
- The reasoner is an algorithm that infers logic by executing rules. The reasoner infers implicit knowledge from explicit knowledge. In building applications, the reasoner can serve as the building analytics (wisdom).

Figure 3 illustrates the proposed framework for KRR in building applications. In this framework, there is a central core piece that encapsulates concepts and properties that do not exist in existing ontologies. For the proposed framework, the Core Ontology stores concepts and their associated properties for geometry and times series equipment properties that are not part of the Brick model. Then, with the Core Ontology in place,

ontology-driven rules and software support for graph queries are developed to support the building application services shown along the top of Figure 3.
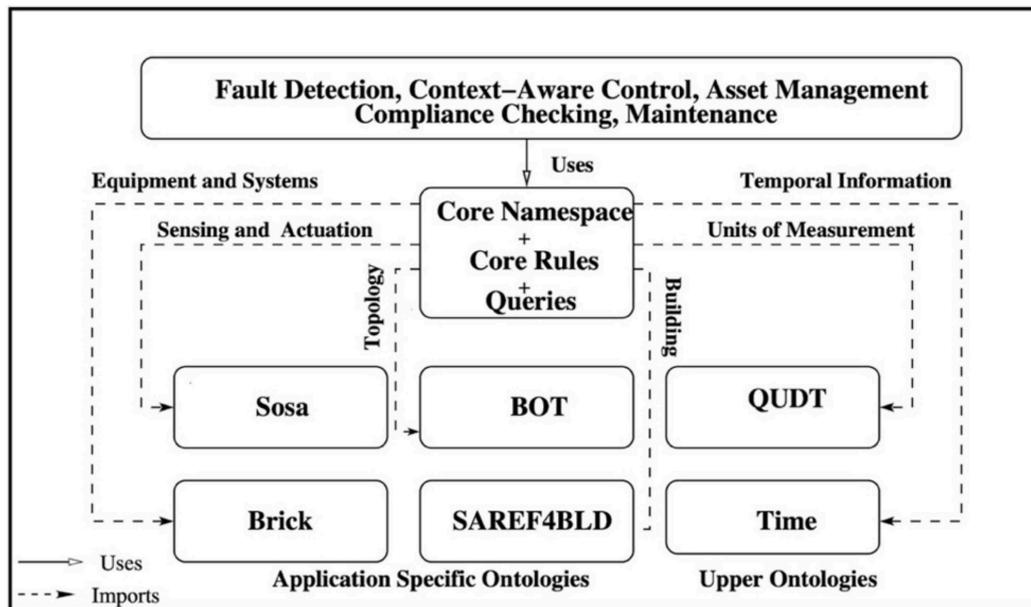
Figure 3. The proposed framework for knowledge representation and reasoning for building applications.

The core ontology namespace imports the concepts from two different classes of ontologies. One class represents the application-specific ontologies that are well established and maintained within the building domain. The second class is composed of more general, domain-independent ontologies, such as those for Time and Units. In this framework, concepts about building equipment and building subsystems are imported from the Brick ontology, while the concepts required for sensing, observation and actuation are borrowed from the SOSA ontology. The BOT ontology provides concepts relevant for describing building topology (i.e., zone containment, or adjacency). Lastly, the required concepts for building spaces and the location of equipment within those spaces come from the SAREF4BLDG ontology.

5. Building Applications

The semantic infrastructure framework and methods specified in Sections 3 and 4 allow for inferencing new statements from existing data in the multi-domain ontologies. Semantic Web features and language capabilities provide the foundation for representing knowledge bases (e.g., physical building information, HVAC equipment, indoor conditions) and reasoning over that knowledge to detect faults and systematically verify hypotheses through the evaluation of supporting evidence. To demonstrate the effectiveness of the proposed methods, this section deploys KRR for case study problems chosen from different building applications to demonstrate event-driven, spatial and temporal reasoning, behavior modeling of component dynamics and modeling of HVAC components in buildings with ontologies (e.g., Jena) and rules (e.g., Jena rules). The applications presented in this paper are:

1. Semantic Definitions: The examples demonstrate proper use of semantic definitions to represent a supply air temperature sensor and a downstream temperature sensor, and to locate the draw-through fan in the system.
2. Fault Detection and Diagnostics: The examples demonstrate the use of a semantic framework for event-based fault detection and diagnostics that mimics human thinking in detecting a fault and diagnosing the underlying causes. One example

demonstrates the detection of a faulty valve. A second example shows anomaly detection triggered by sensor readings falling outside an acceptable range.

3. Spatial Reasoning: Buildings typically require a significant amount of spatial reasoning to identify spaces, zones, regions, connections and relationships among connections and regions. The examples presented here demonstrate the use of inference to identify neighboring zones and determine the location of sensors relative to the regions and zones in which they are embedded.

4. Temporal Reasoning: Temporal representation and reasoning capabilities play a central role in reasoning about temporal variations in building occupancy and operational scheduling.

5. Asset Management and Maintenance: Maintenance management tracks the device and equipment to make important decisions for improving the performance and efficiency of the device. The example provided in this section describes a mechanism to keep track of resources that are due for maintenance service.

6. Context-Aware Control: Context-aware control sits on a layer above rule-based, model-based and data-driven control. This strategy considers the context that the equipment or the space is being used for in making decisions. The example implements a portion of the sequence of operations for a garage space.

5.1. Semantic Representation

To demonstrate semantic representation, this section shows three different usecases focused on defining and identifying supply air temperature sensors in two examples and also identifying the location of a supply air fan in an air duct system.

5.1.1. Supply Air Temperature Sensor

Table 2 lays out a simple definition of a supply air temperature sensor. As the example shows, if there is an air temperature sensor that is connected to an AHU (air handling unit) and a VAV (variable air volume) box, it is marked as a supply air temperature sensor. These semantic definitions, if used properly, can reduce the size of the model because a specific categorization, such as supply air temperature sensor, is not needed. Note that the "hasPoint" property is from the Brick ontology.

Table 2. A rule example on defining the supply air temperature.

```
@prefix ns: <http://example.org/ex#>.
@prefix brick: <https://brickschema.org/schema/1.1/Brick#>.
@prefix xs: <http://www.w3.org/2001/XMLSchema#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

// Definition of Supply Air Temperature Sensor
[SupplyAir: (?ahu rdf:type brick:AHU)(?vav rdf:type brick:VAV)(?satv rdf:type
ns:Air_Temperature_Sensor)
          (?sata rdf:type ns:Air_Temperature_Sensor)
          (?vav brick:hasPoint ?satv)(?ahu brick:hasPoint ?sata) equal(?satv,?sata)->
          (?sata ns:isSupplyAir "true"^^xs:boolean) print('Supply temperature', ?sata)]
```

5.1.2. Downstream Temperature Sensor

Figure 4 illustrates an air duct example with different components such as heating and cooling coils, supply and return air fans, dampers, sensors, humidifier, and filters. A human expert can look at the schematic and deduce that there is a Tempertature-Sensor_1 exactly downstream of a HeatingCoil_1. As Table 3 shows, the ontology al-lows querying the model to obtain the result of TempeSensor1. The select statement chooses all temperature sensors that have a measurementLocation in the duct of HeatingCoil_1. As the figure shows, in an air duct system, there are usually several sensors, including temperature, differential pressure and/or airflow stations, Carbon Dioxide (CO_2) and smoke detector and utilization of the ontology for the model repre-sentation provides a unique solution to

simplify the entire knowledge representation and reasoning of different systems. Interested readers are encouraged to look at the ASHRAE Guideline 36 for additional examples of different air duct systems [44].
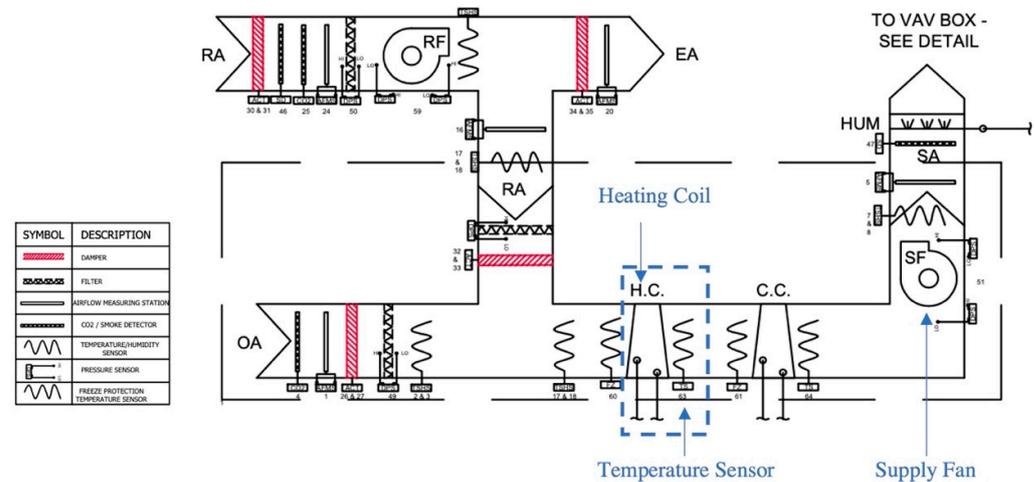
Figure 4. An air duct system with highlighting the heating coil, temperature sensor, and supply air fan in the image with a blur font color.

Table 3. A SPARQL query on finding defining the supply air temperature sensor after the heating coil.

```
SELECT ?downStream
WHERE{
BIND(:heatingCoil1> AS ?d)
{
?d s223:hasConnectionPoint ?cp1.
?cp1 a s223:OutletConnectionPoint.
?cp1 s223:connectsThrough ?downcon.
?downStream a s223:TempSensor.
?downStream s223:hasMeasurementLocation ?downcon.
}
Result:
[downstream]
ex:TemperatureSensor1
```

5.1.3. Draw-through Fan

In many applications such as FDD, commissioning and maintenance, it is important to know the sequence of the equipment after the system configuration. As a case, the supply fan can be placed before the cooling coil, which is the blow-through configuration, or after the cooling coil, which is the draw-through configuration shown in Figure 4. Having a semantic model of this system, we can query the model to see whether or not it is a draw-through system. Table 4 depicts a query that asks the model if the cooling coil is connected to (draw-through) or connected from a fan (blow-through).

Table 4. A rule example on identifying the location of the fan in the system.

ASK	ASK
WHERE {	WHERE {
?d a:Coil.	*?d a:Coil.*
?d:hasRole role:Cooling.	*?d:hasRole role:Cooling.*
?d:connectedTo* ?fan.	*?d:connectedFrom* ?fan.*
?fan a:Fan.	*?fan a:Fan.*
}	*}*
Result:	**Result:**
true	*False*

The three examples mentioned in this section show how semantic definitions could be used in different building applications.

5.2. Fault Detection and Diagnostics

Table 5 contains two examples of the application of reasoning to FDD: (1) detect a cooling coil valve leak when the valve is shut but the supply temperature is lower than the mixed air temperature, and (2) recognize that a sensor reading is greater than its maximum reading limit. These rules use concepts from the SOSA and Brick ontologies. The SOSA ontology associates a sensor with an observation. An observation has a Feature of Interest: e.g., a zone, a result, e.g., 24 degrees Celsius and a time associated with the observation. One feature of Jena rules is built-in functions such as compare(), used in this example. Jena rules employ built-in functions to perform more complex actions than simple triple operations.

Table 5. Two examples of rules in an FDD application.

@prefix ns: <http://example.org/ex#>.
@prefix sosa: <http://www.w3.org/ns/sosa#>.
@prefix brick: <https://brickschema.org/schema/1.1/Brick#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix qudt: <http://qudt.org/1.1/schema/qudt#>.

// Detect a cooling coil valve leak
[(?ahu rdf:type brick:AHU) (?ahu brick:hasPoint ?v) (?ahu brick:hasPoint ?s1) (?ahu brick:hasPoint ?s2)
(?v rdf:type brick:Cooling_Valve) (?v ns:isShutOff true) (?s1 rdf:type
brick:Mixed_Air_Temperature_Sensor)
(?s2 rdf:type brick:Supply_Air_Temperature_Sensor) (?s1 sosa:madeObservation?ob1) (?s2
sosa:madeObservation ?ob2) compare(?ob1,?ob2,?c) equal(?c,"2"ˆˆxs:int)
-> RaiseAlarm ("Alarm Valve is Leaking")]

// Detect a faulty sensor
[(?z sosa:isFeatureOfInterestOf ?ob)(?s rdf:type ns:Air_Temperature_Sensor)
(?s ns:maxRange ?max) (?s sosa:madeObservation ?ob) (?ob sosa:resultTime ?t) (?ob sosa:hasResult ?r) (?r
sosa:hasSimpleResult ?sr) (?r rdf:type qudt:Quantity) (?r qudt:unit ?u) equal('DEG_F',?u)
*ge(?sr,?maxRange) -> RaiseAlarm ("Sensor Reading Greater than Maximum Possible Value")]*

Built-in functions can be invoked in the head of forward rules. As a case, compare(?ob1,?ob2,c) performs a comparison based on the values of the observations, ?ob1 and ?ob2, and returns 1, 2 or 0 if the first, second or none of the input parameters are greater than the other.

5.3. Spatial Reasoning

Formal approaches to spatial reasoning are concerned with inference on points, lines and regions and their connectivity. The mathematical foundations for reasoning with spatial elements—points, lines and regions—are covered by the region connection calculus [45]. From a computational standpoint, the Java Topology Suite (JTS) provides a

robust implementation of two-dimensional spatial concepts and supports spatial reasoning procedures [46].

Figure 5 is an abbreviated view of the prototype ontology and associated data and object properties for the representation of a two-dimensional spatial geometry. To facilitate the implementation of our software prototype, specific types of geometries (e.g., polygon, multipoint) mirror the class structure found in the JTS. The high-level class AbstractGeometry contains a datatype property, hasGeometry, which stores a string representation of the JTS geometry. The string representation of the geometry acts as input to algorithms for analysis of the two-dimensional geometry. To see how this works in practice, the first rule in Table 6 systematically infers if two HVAC_Zone individuals are adjacent. The second rule identifies if a sensor is inside a zone based on its location. This is possible because Brick concepts such as HVAC_Zone and Sensor have been extended to have the hasGeometry property. The built-in functions getAdjacency() and getPointInPolygon() evaluate the geometric relationship between pairs of spatial entities (e.g., to determine if a point is contained within a polygon or if two polygons are adjacent). The results of reasoning procedures are saved regions of the semantic model (i.e., namespaces) that follow terms from the Brick and SAREF4Building ontologies.
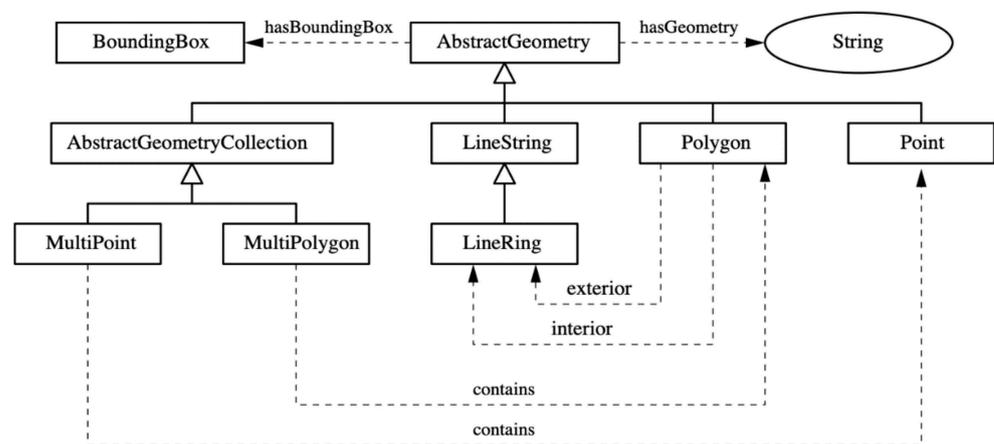
Figure 5. High-level ontology for a two-dimensional geometry and associated data and object properties [47].

Table 6. Two examples of spatial reasoning.

```
@prefix ns: <http://example.org/ex#>.
@prefix brick: <https://brickschema.org/schema/1.1/Brick#>.
@prefix bot: < http://www.w3id.org/bot/bot.ttl#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix s4bldg: <https://saref.etsi.org/saref4bldg/>.

// Identify Adjacent Zones:
[(?r1 rdf:type brick:HVAC_Zone) (?r1 ns:hasGeometry ?r1jts)(?r2 rdf:type brick:HVAC_Zone)(?r2
ns:hasGeometry ?r2jts) getAdjacency(?r2jts,?r1jts,?t) equal(?t,"true"^^xs:boolean) -> (?r1
bot:adjacentZone ?r2)]

// Locate a Sensor in a Zone:
[(?r1 rdf:type brick:HVAC_Zone) (?r1 ns:hasGeometry ?r1jts) (?satv rdf:type
ns:Air_Temperature_Sensor)(?satv ns:hasGeometry ?sajts) getPointInPolygon(?sajts,?r1jts, ?t)-> (?satv
s4bldg:isContainedIn ?r1)]
```

A natural extension of this capability is built-in functions that can work with libraries for representation and reasoning and advanced 3D geometries (e.g., scan-to-graph [48] or Linked Data-based BIM [49]). While this study used JTS, the long-term goal for this framework is to also consider more robust building geometry models such as IFC, BIM [50]

and Green Building XML (gbXML) [51] to expand our knowledge representation, querying and reasoning capabilities.

5.4. Temporal Reasoning

Formal approaches to temporal reasoning are concerned with the evaluation of relationships among entities of time. There are four time concepts that support ontological representations of time in engineering based on the temporal theories identified in [52]: (1) time interval, (2) time duration, (3) time point and (4) time dimension. The OWL-Time ontology uses instant and interval as foundational temporal entities [28]. Table 7 shows two examples of temporal reasoning. The first rule indicates if two intervals meet (i.e., when the first interval ends, the second one begins). This reasoning can be used in different applications within the building domain. For example, this rule can ensure that the interval for preconditioning a building happens before the interval when the building is occupied. The second rule implies that if a time instant is between the beginning and the end of an interval, that instant is inside the interval. This rule also has implications for different building applications, such as the case of time-of-use utility programs to identify if the time of consumption is during the peak time of day. In the operation of buildings, temporal reasoning is a key aspect to understand the interaction of occupants, equipment, systems and other building components. This example is a proof of concept on how temporal reasoning could be used in building applications.

Table 7. Two examples of temporal reasoning.

@prefix time: <https://www.w3.org/TR/owl-time#>.

// When Two Intervals Meet
[(?x rdf:type time:TimeInterval) (?y rdf:type time:TimeInterval) (?x time:endsAt ?t) (?y time:beginsAt ?t) ->(?x time:intMeets ?y)]

// When an Instant Falls in an Interval
[isInInterval: (?x rdf:type time:Interval) (?y rdf:type time:Instant) (?x time:hasBeginning ?t1) (?x time:hasEnd ?t2) lessThan (?t1, ?t2) (?y time:inXSDDateTimeStamp ?t3) lessThan(?t1,?t3) greaterThan(?t2,?t3) -> (?y time:inside ?x)]

5.5. Asset Management and Maintenance

Building equipment and systems usually have a typical life expectancy, which is an important factor for facility managers in terms of asset management and maintenance. For example, most major HVAC equipment has a life expectancy of 20 to 25 years [53]. During the operation of a building, facility managers need to understand and plan for the replacement of equipment and systems. This example demonstrates the benefits of using KRR for asset management and maintenance. Table 8 shows how the facility manager can identify the valves that are due for service or replacement.

Table 8. An example of asset management and maintenance.

@prefix ns: <http://example.org/ex#>.
@prefix brick: <https://brickschema.org/schema/1.1/Brick#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

(?valve rdf:type brick:Valve) (?valve ns:hasReplacementTime ?rt1) (?valve ns:hasRunTime ?rt2) greaterThan(?rt2,?rt1) -> (?valve:dueForService true)

5.6. Context-Aware Control

By taking context into account, context-aware control promises a richer and more customized way of controlling building spaces. As an example, the rule in Table 9 implements a segment of the sequence of operations for controlling an underground parking garage

using a carbon monoxide (CO) linear demand control strategy [54]. This is a practical example that shows how context-aware control could support building managers. In this strategy, the exhaust fans operate at 25% of their full capacities when the average CO concentration in the parking garage is less than 10 ppm_v. After the average CO concentration in the parking garage increases above 10 ppm, the fans gradually increase capacity by 3% for every additional 1 ppm of average CO concentration, operating at maximum speed (100%) when the CO concentrations are 35 ppm or higher. While this example shows demand-controlled ventilation based on CO, similar approaches could be used for other ventilation strategies in buildings using KRR.

Table 9. An example of context-aware control.

@prefix ns: <http://example.org/ex#>.
@prefix brick: <https://brickschema.org/schema/1.1/Brick#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix s4bldg: <https://saref.etsi.org/saref4bldg/>.
@prefix qudt: <http://www.ontology-of-units-of-measure.org/resource/om-2/partsPerMillion>

// Utilize the Demand Control Ventilation
(?s rdf:type s4bldg:BuildingSpace) (?ef rdf:type brick:Exhaust_Fan)(?ef brick:hasPoint ?exfc)(?s ns:usage "parking garage") (?cs rdf:type ns:CO_Sensor)(?s s4bldg:contains ?ct) (?s sosa:madeObservation ?ob) (?ob sosa:hasResult ?r) (?r sosa:hasSimpleResult ?sr) (?r qudt:unit ?u) equal('ppm',?u) ge(?sr,35) -> (?exft ns:hasValue 100)

---

6. Discussion

This paper has shown that while many state-of-the-art domain ontologies have a role to play in the semantic modeling of building operations, present-day practice is to develop ontologies and rules for specific AEC and FM domains and use cases. Too often, these ontologies are not developed from an ontology design pattern perspective, and with support for integration and alignment in mind. As a result, we are still a long way from the sustainable design and operation of building systems. Solutions to this problem are complicated by the diversity of perspectives needed to achieve a long-term balance in the three E's of efficiency, economy and equity. We surmise that in the long term, the development of a comprehensive, unified ontology for all aspects of buildings will be challenging and most likely fail. In a first step to working around this difficulty, this work explored an alternative approach where domain-specific and upper ontologies are combined (integrated) for specific building applications.

An underlying tenet of our modeling work is that a wide range of building system applications can benefit from the deployment of KRR. For example, to implement the SOO of building mechanical systems or implement commissioning guidelines, ASHRAE Guideline 36 [44] can be translated to ontology-driven rulesets. KRR also has implications for the U.S. Department of Energy's Building Energy Data Exchange Specification (BEDES) that includes a dictionary of terms and definitions applicable to tools and activities to track building energy data and assist building stakeholders [55]. Although there have been technological advances to modernize building control, operation and simulation systems, the topics and solutions covered in this paper have received less attention and have rarely been adopted in the building industry for reasons including the lack of a generic interlinked descriptive framework model that would ease the deployment of the framework in different buildings and building applications with minor adoption costs.

Several efforts are underway to advance the state-of-the-art capability for ontologies and data models in building applications. One such effort is the new ASHRAE proposed standard committee, SSPC 223P, which stems from BACnet SI-WG and was formed to develop a standard semantic model that would formally define the required knowledge concepts and a methodology to apply those concepts to create interoperable, machine-understandable semantic models that represent building system information for use in analytics, automation and control [56].

Overall, although recent studies discussed in this work have sought to address the development and utilization of a semantic framework for buildings through integration of existing metadata schemes, the literature lacks a uniform schema for representing metadata in buildings and providing core ontologies that define the fundamental concepts and their relationships.

7. Conclusions

This paper proposes a semantic infrastructure and methods that leverage W3C (World Wide Web Consortium) standards to construct, constrain and query the ontologies for different building applications and integrate concepts from various models. Semantic Web technologies enable sharing, accessing and integrating building data in a machine-understandable format. The power of precise semantics captured in the ontologies eliminates ambiguity in the model. Moreover, the formal logic foundation allows for inference to extract new information. While there are existing domain-specific ontologies that represent the knowledge needed for smart buildings, major setbacks have limited the use of them in different building applications due to: (i) large and complex bodies of terminology that are not easy to use and (ii) many overlapping and often conflicting concepts in the different ontologies. Since much of the existing literature focuses on a limited number of ontologies for knowledge representation, this study proposes a semantic infrastructure and methods to leverage existing domain-specific ontologies for various building applications. As a case in point, we used BOT (Building Topology Ontology) for knowledge representation of building topology, QUDT (Quantity, Unit, Datatype) for units of measurement, SAREF4BLDG (Smart Appliances REFerence) for building systems and spaces and Brick for building equipment. To demonstrate the proposed semantic infrastructure and methods, we implemented specific semantic rules to describe inference mechanisms for building applications ranging from semantic representation, fault detection and diagnostics, spatial reasoning, temporal reasoning, asset management and maintenance and context-aware control. While the examples target building mechanical systems, a similar approach could be used for other building applications such as lighting, shading devices and building security. The next step in this work is to deploy the approach to different building systems, including a multi-domain building system, as well as deploying knowledge representation and reasoning for different building applications in a real building using real data.

8. Disclaimer

Certain commercial equipment, instruments or materials are identified in this paper to foster understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Feigenbaum, L. Semantic Web Technologies in the Enterprise. 2006. Available online: http://www.thefigtrees.net/lee/blog/2006/11/semantic_web_technologies_in_t.html (accessed on 1 November 2021).
2. Lee, E.A. Model-Driven Development—From Object-Oriented Design to Actor-Oriented Design. In Proceedings of the Workshop for Software Engineering for Embedded Systems, From Requirements to Implementation, Chicago, IL, USA, 24 September 2003.
3. Guarino, N.; Oberle, D.; Staab, S. Handbook on Ontologies. In *What Is an Ontology?* Springer: Berlin/Heidelberg, Germany, 2009. [CrossRef]
4. Gilani, S.; Quinn, C.; McArthur, J. A review of ontologies within the domain of smart and ongoing commissioning. *Build. Environ.* **2020**, *182*, 107099. [CrossRef]
5. Petrova, E.; Pauwels, P.; Svidt, K.; Jensen, R. Semantic data mining and linked data for a recommender system in the AEC industry. In Proceedings of the 2019 European Conference on Computing in Construction, Chania, Crete, Greece, 10–12 July 2019. [CrossRef]
6. Dibley, M.; Li, H.; Rezgui, Y.; Miles, J. An ontology framework for intelligent sensor-based building monitoring. *Autom. Constr.* **2012**, *28*, 1–14. [CrossRef]
7. Han, J.; Jeong, Y.-K.; Lee, I. Efficient Building Energy Management System Based on Ontology, Inference Rules, and Simula-tion. In Proceedings of the CSIT, Yerevan, Armenia, 26–30 September 2011; IACSIT Press: Singapore, 2011.
8. Kučera, A.; Pitner, T. Semantic BMS: Allowing usage of building automation data in facility benchmarking. *Adv. Eng. Inform.* **2018**, *35*, 69–84. [CrossRef]
9. Wicaksono, H.; Aleksandrov, K.; Rogalski, S. *An Intelligent System for Improving Energy Efficiency in Building Using Ontology and Building Automation Systems, Chapter Automation*; Spon Press: London, UK; New York, NY, USA, 2012.
10. Mahdavi, A.; Taheri, M. An ontology for building monitoring. *J. Build. Perform. Simul.* **2017**, *10*, 499–508. [CrossRef]
11. Puttonen, J.; Lobov, A.; Soto, M.A.C.; Lastra, J.L.M. A Semantic Web Services-based approach for production systems control. *Adv. Eng. Inform.* **2010**, *24*, 285–299. [CrossRef]
12. Ackoff, R.L. *Ackoff's Best*; John Wiley & Sons: New York, NY, USA, 1999.
13. Bernstein, J.H. The Data-Information-Knowledge-Wisdom Hierarchy and its Antithesis. *NASKO* **2009**, *2*, 68. [CrossRef]
14. Stephens, B.; Novoselac, A.; Siegel, J.A. The Effects of Filtration on Pressure Drop and Energy Consumption in Residential HVAC Systems (RP-1299). *HVAC&R Res.* **2010**, *16*, 273–294. [CrossRef]
15. DOE. Maintaining Your Air Conditioner. 2020. Available online: https://www.energy.gov/energysaver/maintaining-your-air-conditioner (accessed on 1 November 2021).
16. Grau, B.C.; Horrocks, I.; Motik, B.; Parsia, B.; Patel-Schneider, P.; Sattler, U. OWL 2: The next step for OWL. *J. Web Semant.* **2008**, *6*, 309–322. [CrossRef]
17. Pareti, P.; Konstantinidis, G.; Norman, T.; Şensoy, M. *SHACL Constraints with Inference Rules*; Springer: Cham, Switzerland, 2019; pp. 539–557. [CrossRef]
18. Koivunen, M.-R.; Miller, E. W3C Semantic Web Activity. 2001. Available online: https://www.w3.org/2001/12/semweb-fin/w3csw (accessed on 1 November 2021).
19. RealestateCore RealestateCore (REC). 2021. Available online: https://www.realestatecore.io (accessed on 1 November 2021).
20. Poveda-Villalon, M.; Garcıa-Castro, R. Extending the SAREF ontology for building devices and topology. In Proceedings of the Proceedings of the 6th Linked Data in Architecture and Construction Workshop, London, UK, 19–21 June 2018.
21. Haller, A.; Janowicz, K.; Cox, S.J.D.; Lefrançois, M.; Taylor, K.; Phuoc, D.; Lieberman, J.; García-Castro, R.; Atkinson, R.; Stadler, C. The SOSA/SSN Ontology: A Joint W3C and OGC Standard Specifying the Semantics of Sensors, Observations, Actuation, and Sampling. *Semant. Web* **2018**, *56*, 1–9.
22. QUDT. 2020. Available online: https://qudt.org/ (accessed on 1 November 2021).
23. Brick, Brick: A Uniform Metadata Schema for Buildings. 2022. Available online: https://brickschema.org/ (accessed on 1 November 2021).
24. Balaji, B.; Bhattacharya, A.; Fierro, G.; Gao, J.; Gluck, J.; Hong, D.; Johansen, A.; Koh, J.; Ploennigs, J.; Agarwal, Y. *Brick: Towards a Unified Metadata Schema for Buildings*; BuildSys '16: Palo Alto, CA, USA, 2016.
25. BOT, Building Topology Ontology. 2020. Available online: https://w3c-lbd-cg.github.io/bot/ (accessed on 1 November 2021).
26. Esnaola-Gonzalez, I.; Bermúdez, J.; Fernandez, I.; Arnaiz, A. EEPSA as a Core Ontology for Energy Efficiency and Thermal Comfort in Buildings. *Semantic Web* **2016**. Available online: http://www.semantic-web-journal.net/content/eepsa-core-ontology-energy-efficiency-and-thermal-comfort-buildings (ac-cessed on 1 November 2021).
27. Esnaola-Gonzalez, I.; Bermúdez, J.; Fernandez, I.; Arnaiz, A. Ontologies for Observations and Actuations in Buildings: A Sur-vey. *Semant. Web* **2016**, *1*, 1–5.
28. OWL, Time Ontology in OWL. 2006. Available online: http://www.w3.org/TR/owl-time (accessed on 1 November 2021).
29. Petnga, L.; Austin, M. An ontological framework for knowledge modeling and decision support in cyber-physical systems. *Adv. Eng. Inform.* **2016**, *30*, 77–94. [CrossRef]

30. IFC, Industry Foundation Classes (IFC). 2021. Available online: https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/ (accessed on 1 November 2021).

31. Kalemi, E.; Cheung, F.; Tawil, A.-R.; Patlakas, P.; Alyania, K. ifcOWL-DfMA a new ontology for the offsite construction do-main. In Proceedings of the 8th Linked Data in Architecture and Construction Workshop—LDAC, Dublin, Ireland, 17–19 June 2020.

32. González, E.; Piñeiro, J.D.; Toledo, J.; Arnay, R.; Acosta, L. An approach based on the ifcOWL ontology to support indoor navigation. *Egypt. Inform. J.* **2020**, *22*, 1–13. [CrossRef]

33. ifcOWL, buildingSMART: ifcOWL. 2020. Available online: https://github.com/buildingSMART/ifcOWL (accessed on 1 November 2021).

34. Abdelrahman, M.; Chong, A.; Miller, C. Build2Vec: Building Representation in Vector Space. *arXiv* **2020**, arXiv:2007.00740.

35. Raghavi, V.; Gowtham, R. AI based Semantic Extensibility and Querying Techniques for Building Information Model. In Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 15–17 May 2019. [CrossRef]

36. Pauwels, P.; Roxin, A. SimpleBIM: From full ifcOWL graphs to simplified building graphs. In *EWork and EBusiness in Architecture, Engineering and Construction*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2016.

37. McGlinn, K.; Bonsma, P.; Wagner, A.; Pauwels, P.; McNerney, L.; O'Sullivan, D. Interlinking Building Geometry with Existing and Developing Standards; 2017. Available online: https://linkedbuildingdata.net/ldac2017/files/Presentations/ldac_geom_2017_11_13.pdf (accessed on 1 November 2021).

38. Haystack, Project Haystack. 2020. Available online: https://project-haystack.org/ (accessed on 1 November 2021).

39. Fierro, G.; Koh, J.; Agarwal, Y.; Gupta, R.K.; Culler, D.E. Beyond a House of Sticks: Formalizing Metadata Tags with Brick. In Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, New York, NY, USA, 13–14 November 2019.

40. Petze, J. Haystack 4. 2021. Available online: https://marketing.project-haystack.org/blog/haystack-4-is-coming-what-it-is-and-why-it-matters (accessed on 1 November 2021).

41. Apache Jena. 2020. Available online: https://jena.apache.org/ (accessed on 1 November 2021).

42. Java. 2020. Available online: https://www.java.com/en/ (accessed on 1 November 2021).

43. SPARQL. SPARQL Query Language for RDF. 2008. Available online: https://www.w3.org/TR/rdf-sparql-query/ (accessed on 1 November 2021).

44. ASHRAE. *ASHRAE Guideline 36-2021: High-Performance Sequences of Operation for HVAC Systems, American Society of Heating*; Refrigerating and Air-Conditioning Engineers, Inc.: Atlanta, GA, USA, 2021.

45. Randell, D.A.; Cui, Z.; Cohn, A.G. *A Spatial Logic Based on Regions and Connectivity*; Division of Artificial Intelligence, School of Computer Studies, Leeds University: Leeds, UK, 1994.

46. JTS. JTS Topology Suite. 2020. Available online: https://github.com/locationtech/jts (accessed on 1 November 2021).

47. Delgoshaei, P.; Austin, M.A. Framework for Knowledge-Based Fault Detection and Diagnostics in Multi-Domain Systems: Application to Heating Ventilation and Air Conditioning Systems. *Int. J. Adv. Intell. Syst.* **2017**, *10*, 393–409.

48. Werbrouck, J.; Pauwels, P.; Bonduel, M.; Beetz, J.; Bekers, W. Scan-to-graph: Semantic enrichment of existing building geometry. *Autom. Constr.* **2020**, *119*, 103286. [CrossRef]

49. Bonduel, M.; Wagner, A.; Pauwels, P.; Vergauwen, M.; Klein, R. Including widespread geometry schemas into Linked Data-based BIM applied to built heritage. *Proc. Inst. Civ. Eng.-Smart Infrastruct. Constr.* **2019**, *172*, 34–51. [CrossRef]

50. Schultz, C.; Bhatt, M. A Multi-Modal Data Access Framework for Spatial Assistance Systems: Use-Cases with the Building Information Model (BIM/IFC). In Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness, Association for Computing Machinery, New York, NY, USA, 2 November 2010; pp. 39–46. [CrossRef]

51. gbXML, Green Building XML Schema (gbXML). 2020. Available online: https://www.gbxml.org/ (accessed on 1 November 2021).

52. Hayes, P. *A Catalog of Temporal Theories*; UIUC-BI-AI-96-01; University of Illinois: Champaign, IL, USA, 1996.

53. ASHRAE. ASHRAE: Service Life and Maintenance Cost Database. 2019. Available online: www.ashrae.org/database (accessed on 1 November 2021).

54. Faramarzi, A.; Lee, J.; Stephens, B.; Heidarinejad, M. Assessing ventilation control strategies in underground parking garages. *Build. Simul.* **2020**, *14*, 701–720. [CrossRef]

55. LBNL, Building Energy Data Exchange Specification (BEDS). 2020. Available online: https://bedes.lbl.gov/bedes-compliance (accessed on 1 November 2021).

56. SI-WG, ASHRAE BACnet Committee Semantic Interoperability Working Group (SI-WG). 2022. Available online: http://www.bacnet.org/WG/SI/index.html#:~:text=SG%20Working%20Group&text=The%20Semantic%20Interoperability%20Working%20Group,associated%20building%20systems%20and%20components (accessed on 3 April 2021).