Semantic Web X (2017) 1–27
IOS Press

Semantic-Graph based approach to BIM, resilience, and spatial data interoperability

Editor(s): Name Surname, University, Country
Solicited review(s): Name Surname, University, Country
Open review(s): Name Surname, University, Country

Holly Ferguson [1,*], Charles Vardeman II [1,**], Adila Krisnadhi [2] and Michelle Cheatham [2]
[a] *Center for Research Computing at the University of Notre Dame,*
Notre Dame, IN, United States of America
E-mail: hfergus2@nd.edu, cvardema@nd.edu
[b] *Computer Science Department at Wright State University,*
Dayton, OH, United States of America
E-mail: michelle.cheatham@wright.edu, krisnadhi@gmail.com

Abstract. Interoperability within the Building Information Modeling and resilience analysis domains is essential for positively influencing the built environment, but is still not sophisticated enough for optimal data and model evaluation with regards to modern, multi-disciplinary questions. Fully interoperable data schemas and tools mean that communications from adjacent domains are also necessary, such as compatibility with geography data schemas or other geometry relationships. Dispersed resources that are required for this analysis cannot be centrally located, rather, they need to be made accessible to users through mediums such as Linked Data Platforms that do not alter existing work flows. This research provides a data pattern, called the Spatial Graph Adapter Pattern, and includes use cases that allow spatial relationships to be captured at a more granular and descriptive level than has been possible thus far; it is implemented to unify several other areas of research including an ontology-based Linked Data Platform and Linked Data Views for proprietary data processing. This pattern is an extension to existing work and is now hosted by a modular and extensible framework used for translating between different Building Industry data formats without needing to change current schemas or work flows.

Keywords: Semantic Graphs, Linked Data Platform, Big Data Interoperability, Building Information Modeling, Linked Data Views, Spatiotemporal Data, Spatial Graph Adapter Pattern

1. Introduction

Building Information Modeling (BIM) has entered an era where Linked Data (LD) and Linked Open Data (LOD)[1] on the Semantic Web[2] and Decision Support (DS) [25] technologies can be leveraged to better-inform currently isolated, domain centered applications in wide use within the building industry. Decisions are being made based upon output from these tools and are greatly impacted due to a limited picture provided by our current, disconnected, computational capabilities. For example, a Life Cycle Analysis (LCA)[3] [28] of a building should include building materials from a range of manufacturers at several locations and include shipping costs and environmental trade-offs as well as material lifespan information.

Interdisciplinary collaborative teams within the building industry are becoming the new norm and, we

[*]Corresponding author. E-mail: hfergus2@nd.edu.
[**]Corresponding author. E-mail: cvardema@nd.edu.
[1]Linked Data: https://www.w3.org/standards/semanticweb/data
[2]Semantic Web: https://www.w3.org/standards/semanticweb/
[3]GS & LCA: http://www.greenscale.org/

believe, LOD based technologies can allow us to begin establishing mappings and translations for models and data sets to start informing one another [29] using vocabularies that are natural for each different domain practitioner [17,18]. For example, within the biomedical research community, a concerted effort has been made to create structured metadata to facilitate information sharing between the diverse disciplines that constitute the field of research. Given the successes demonstrated in other fields, this approach of creating "smarter data" would be of great value to the highly multidisciplinary building industry. To bring BIM data to this level of accessibility, resolving the differences in semantics across formats is essential. We are utilizing and reusing work such as Spatial Information Theory [26], modular scientific ontologies such as Semanticscience Integrated Ontology (SIO) [13], work on formalizing Mereotopology [22], and geospatial data efforts via the Open Geospatial Consortium (OGC)[4] standards.

However, without expanding upon existing methods of representing geometry structures and associated spatial relationships, too much context can be lost in BIM translations due to a limited granularity in description options, if the translations can be done at all. Existing data models typically describe geometries and basic connections plus adjacency of physical elements within a drawing specification. It is our observation that BIM schema are typically created with the intention of conveying "instance" information without broader context even though there may be an implicit context contained in schema "tags" that limits the transferability of the data model to different schema. This has the effect of limiting application of the data to the intended domain or results in a simplistic data model that is insufficient for capturing all of the context needed for true interoperability. This paper presents a summary of our existing work as well as a proposed Spatial Graph Adapter (SGA) ontology design pattern that facilitates lifting of existing schema while retaining proper spatial and parametric geometric information necessary for BIM.

Graph-based semantic models are becoming the focus and often a viable solution not only for the presented work and related research level efforts, but also on the scale of enterprise projects.[5] This article is an extension and full implementation of the contents of several previous papers set in a larger scope of research; one is an in depth description of the architecture of the Linked Data Platform components' functionality [17], and the second is a detailed methodology using Linked Data Views (RDF[6] Semantic Mappings) to extract spatial data into Semantic Web compatible structures [18]. The Semantic Graph approach presented combines the work from these other two papers and extends it to include techniques for combining the functionality of these components within a platform architecture; we also provide several example use cases (with populated patterns and code samples) answering industry questions using these semantic alignments through modular SPARQL endpoints.

We also present the ontology-based data pattern that connects domain vocabularies (that interpret the otherwise proprietary BIM or resilience data) utilizing W3C Linked Data Platform (LDP) components to publish data as LOD objects as well as a methodology to make these connections a possibility. The remainder of the paper is organized to explain how these pieces of software infrastructure come together while maintaining Linked Data Principles and providing opportunities for capturing data provenance for the benefit of BIM. We begin with a review of the existing work and related projects including the development and use of Linked Data Views and the structure of the LDP in question leading to the need for an ontology-based connection method. This is followed by the methodological development of the LDP structure and SGA Pattern, as well as the axiomatization in first order logic. Next are the case studies that apply and validate the approach; these are diagrammed and explained for scenarios that have arisen in this ongoing research, especially when connecting BIM data to resilience-based engineering data. Finally, this project is analyzed against other similar works and future prospects that may influence the overall vision.

2. Motivation

The focus of this research is based around the idea modularizing semantic graph construction and usage such that it is not another stand-alone standard or converter for limited use, but instead is an intelligent data generator that can understand any one of the already

[4]OGC:http://www.opengeospatial.org/standards/geosparql

[5]Graph-Based Semantic Models: http://www.datasciencecentral.com/profiles/blogs/introducing-a-graph-based-semantic-layer-in-enterprises

[6]RDF: https://www.w3.org/2001/sw/wiki/RDF

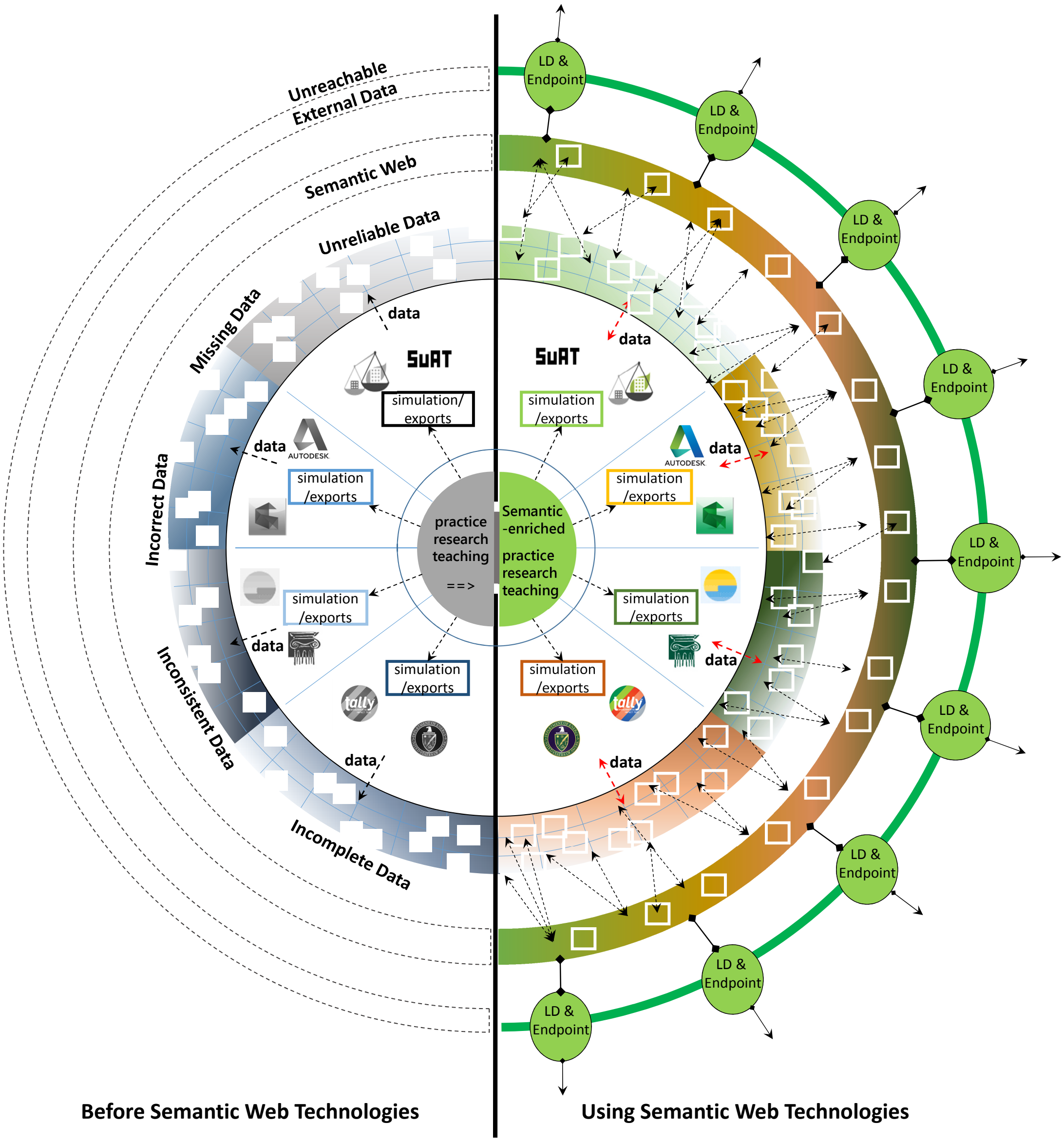


Fig. 1. Depiction of BIM Data Currently Compared to the Additions Presented in this Research

standardized views of the world (in particular, this implementation handles spatial/GeoSpatial/Environmental data). This project does not debate the need for these standards and types of structured data, but it is unique in the fact that it interprets these different standards in such a way that new standards can be modularly incorporated as they are created by diverse communities. Patterns (like the ones designed below) and ontologies give metaphorical crosswalks to automate understanding between the different standards and associated local semantic mappings which often have numerous discrepancies, even when referring to the same items or concepts. Typically, current converters use only XML[7] data tagging and it is not done using any formal semantics thus providing no possibilities for mapping to and from different applications. The solution presented provides a queryable approach to build Semantic Graphs independent of local definitions. These Semantic Graphs provide, in effect, a higher view of a schema level mapping around which to build in additional knowledge. Furthermore, this work takes advantage of the strategic application levels from which to extract information since it is neither advantageous nor practical to format every piece of data in JSONLD or triples. For example, if a simulation only needs to know the semantics of wall material information, processing is not required to transform all structural information into triples as well. This research also makes use of lifting relevant semantic relationships and tracking provenance details while still maintaining mappings to the underlying, original representations.

After studying many industry standard tools (Revit[8], Ecotect[9], Athena Impact Estimator[10], U. S. Department of Energy (DOE-2)[11], etc.) and even after developing our own multi-metric application called the Green Scale Tool (GST)[12], it was still evident that there were inherent issues with the raw data consumed in the tools that even the best multi-metric tools couldn't overcome in terms of automatic interoperability. The best-made simulation tools are still only providing results based upon information that is often not sourced, incomplete, inconsistent, and missing [16] [Figure 1, left]; making simple fixes in the software pertinent to only one tool is not a sufficient fix for getting more accurate analysis when it now possible to make smarter data. More specifically, the issues with the data come from the fact that there are too many differences between entries for the "same" data when comparing it from different industry data bases (for example material density values). This may seem negligible if looking at one wall's worth of a material, but when these numbers are extrapolated over entire buildings or city infrastructure, then the impacts are too large to ignore. Therefore, our goals of making smarter data means that instead of relying, for example, on only the energy provided by a non-sourced set of tool output that we are able to verify the values for energy use during every step of the lifespan of building materials from mining to manufacturing to shipping and throughout material life spans for engineering risk and resilience calculations.

There was also a desire to eventually expand this work and analysis to include complete supply chain information for manufacturing and building materials as well as build a knowledge base to facilitate data aggregations and connect various types of building information. In order to do this, however, the approach to this work needed to be changed to ensure that the project would be modular, extensible, scalable, and Semantic Web compatible since the amount of data that needs to be processed is potentially exponential in complexity and extensive in quantity. These challenges can be met by using Linked Data approaches to handling BIM data [Figure 1]. Using smarter, semantically enhanced data in more robust tools is advantageous to both the architecture and engineering fields as well as all of the decisions made to produce a more sustainable built environment.

Since data generated by tools is often "siloed" in proprietary structures or at least in various formats, it does not always contain the semantics necessary to function outside of specialized, domain centric, schema such as CityGML [21], Green Building XML (GBXML)[13], Industry Foundation Classes (IFC)[14], and others should a modular or extensible semantic mapping method exist. Fortunately, these data standards contain recognizable information patterns that can be computationally predicted, making automatic data extraction a possibility. The efforts of the proposed project caused the realization that using smarter data techniques means that our multi-metric simulation set no longer has to be limited to one type of energy and one type of thermal calculation. Instead, processing semantic mappings within a scalable Linked Data Platform became the more generalizable solution to the specific goal of simulations that can include and be simultaneously cross-examined with any chosen design metric, such as energy performance or thermal gains/losses, but also have metrics for analysis types of Life Cycle Assessment (LCA)[15], structural simulations, analysis of components from the city and land surroundings, and other resilience models.

[7] XML: https://www.w3.org/XML/
[8] Revit: www.autodesk.com/products/
[9] Ecotect: usa.autodesk.com/ecotect-analysis
[10] Athena: www.athenasmi.org/our-software-data
[11] DOE-2: http://energy.gov
[12] GS & LCA: http://www.greenscale.org/

[13] GBXML: http://www.gbxml.org/
[14] IFC: http://www.buildingsmart-tech.org/ifc/IFC4/
[15] LCA: http://www.nrel.gov/lci/assessments.html

2.1. *Novelty of the Semantic Mapping Approach to Data Integration*

All three data standards studied currently each provide part of a more complete picture and in fact feed back into one another; resilience modeling data can be found in CityGML, energy performance data is often in gbXML, structural or engineering information comes from IFC, and finally any missing or incorrect data can be resolved through LD-based solutions presented in this article once initial data is mined and examined [Figure 1]. Ideally and as available, all three types need to be mined and analyzed together in order to determine any long term effects or to understand how all of these building perspectives influence the built environment relative to one another based on a user's individual design choices or decision support system priorities. Therefore, to offer more accurate multi-criteria decision analysis results to users, these different data formats all need to be interoperable in a modular fashion so as to continue to expand upon these research ideas in an organized manner. The explored solution for the combination of several research pursuits has led to the following:

- Implementation of a spatial Ontology Design Pattern and associated Linked Data Views (RDF mappings) [18] within a Linked Data Platform [17] (which is a Linked Data standard, is extensible, is scalable, and gives compatibility when implementing additional Ontology Design Patterns).
- This work generates the link between the domain data with which this research group works and the LDP which assists the creation of cloud-based smart applications via modular Docker[16] Containers; presented are the bridging components and use cases on current domain expert questions.
- While the individual previous works [17] [18] were designed to work as self-describing entities for ease in automating knowledge, the connecting Ontology Design Pattern presented here unites these two previous efforts such that self-description is both possible and tractable, and older vocabularies can also be included using LD Views.
- This solution for data interoperability in particular allows the automation of connections with the Linked Open Data Cloud[17] and linking together previous work means we are therefore functionally able to find and process both RDF and non-RDF data and can start to bring that information into our simulations.
- Use of patterns, like the one demonstrated below, additionally helps to generate the unified Semantic Graphs that allow us to verify that all the data and provenance information is present in the model, connect to other data stores to fill in what might be missing (or search for external data as needed), and to validate the existing data and relationships against other industry (or user preferred) informational resources (eventually using other patterns to provide reliability metrics).

[16]Docker: https://www.docker.com/
[17]Linked Open Data Cloud: http://lod-cloud.net/

Linked Data principles used with the Semantic Web allow challenges in these domains to be resolved; the approach presented is implemented within a cloud-based infrastructure called a Linked Data Platform. As an important note, this approach to resolving this set of data challenges within the architecture and engineering domains is achieved by combining the efforts of two existing works from this same research as mentioned earlier. The first is the LDP Architecture itself with components in Docker Containers using RESTful interfacing and design for data discovery [17]; the second is a method called Linked Data Views [18] which are RDF files that map how the LDP should interpret specific industry schema standard information (IFC, GBXML, CityGML) such that semantic graphs be generated in interchangeable Linked Data formats. The implementation that brings these two separate projects together and creates the fuller picture of making smarter data and applications is the focus of the work to follow since understanding industry data is one challenge but using it as semantic graphs in a LDP is another. As an extension and facilitator to existing work, this article explains the implementation of several areas of our research as well as describes and demonstrates how these connect into a functional computational ecosystem by applying a newly proposed and axiomatized Ontology Design Pattern and explanation through use cases that need to be solved by combinations of these different types of data. For example, some of these situations arise when asking about how your building (GBXML, IFC) performs considering the impact of the surrounding environment (CityGML, etc.).

2.2. Domain Expert-Based Competency Questions

With the help of various domain experts, competency questions have been developed that are used to guide the development of the research. Our proposed pattern and query automation parser automatically capture and extract geometry, spatial relationships, structural elements, floor levels, and additional related information using LD principles [27] - at times as extensions of existing SIO [13] or other mereotopological concepts. It is important to point out that these calculations currently can be completed in individual tools; however our goal is to be able to automatically set this information up and validate the numbers to then automatically complete multi-criteria analysis without manual effort editing files and without introducing a lot of the error that comes from exporting and importing 3D models between tools (as is a common situation today). This approach allows domain-related and multi-disciplinary questions such as the following to be answered more holistically:

1. What spaces (GBXML) is this damaged structural beam (IFC) a part of (without needing to iterate through every existing spatial component or coordinate set)?
2. Exactly how much of structural element A (IFC) is exposed and to what spaces (GBXML)?
3. What are all of the spatial elements in this graph that contain void spaces (air spaces in assemblies that could be susceptible to damage)?
4. What thermal or occupancy zones is this assembly associated with (these may or may not have to do with adjacency or locality)?
5. What sets of columns are responsible for supporting floors X and Y (IFC) and how are these chosen elements (GBXML) impacting the environment (CityGML)? (Is there a more environmentally-friendly option for these structural elements given the specific environmental surroundings?)
6. For the green-resilience engineering calculations for my building, what is both the structural integrity (IFC) under wind loads (ASCE 7[18]) and the thermal performance (GBXML) over its lifecycle (External LCA Simulations) analyzed at the same time?
7. What are the peak wind speeds (ASCE 7) and weather effecting my building (GBXML or IFC) considering all of the surrounding city buildings (CityGML) within N radius using a specific set of weather data (ex. Energy Plus[19])?
8. Within N-mile radius of my building (IFC or GBXML), what other structures (CityGML) are most likely to effect both the thermal energy (GBXML) and wind speed values (ASCE 7) being used in my simulations?

3. Existing Research and Related Projects

Simulation tools have begun to resolve many of the issues [16] raised when examining the current state of BIM and sustainable data and at times they do work to comparatively analyze data and computational results. For example, newer semantic-based techniques such as semantic rules are gaining attention for solving challenges where synthesizing different types of information now requires additional context [32]. In terms of data analysis, there are also now projects that have focused on converting certain aspects of BIM data into RDF [10] utilizing Semantic Web technology. These give valuable insights into the value proposition associated with Semantic Web technologies that allow the possibility of creating new web services, enhance data discovery, and automate computational execution [35]. In particular, BIM and resilience data can be enhanced with essential 3D spatial contextual knowledge which is absolutely necessary for this data to be successfully processed or translated [14] or to be reapplied to describe building objects. There have been major efforts to automatically align information between various data formats; even though this is typically worked out between a set of two data formats, it is no small endeavor [12]. Due to the complex nature of such tasks, the value of being able to automate these processes quickly becomes important and also was for the research presented in this paper as well since we were designing not only to be able to translate between two formats but modularly by mapping semantic graphs between any number of spatial representations. While use cases are provided later to address the limits of solutions, it should be noted that we do acknowledge the efforts of alignment are immense and that there are several pieces to the whole challenge, such as resolving certain discrepancies in coordinate sets [12].

Systems are needed to integrate different data formats for many reasons; one example using two data

[18] ASCE 7: http://windspeed.atcouncil.org/

[19] Energy Plus: https://energyplus.net/

formats enabled the translation between GIS data and CityGML data such that systems can be built to improve the richness of navigational data. This was executed in one tool called BlindAid [36] which improved data such that the visually impaired have access to better spatial references. Data platforms that are ontology-based [8] not only allow proper transitions between spatial and building material information, but also make it possible to find vulnerabilities and calculate data uncertainties which are critical to provide context for any decision support application for building professionals [1]. Several software or web service efforts are attempting to help BIM become more open and transferable[20] [33], and can ultimately lead to more reliable building calculations, and help make general geospatial knowledge[21] more accessible for decision support [25].

Our approach allows data compatibility for the inclusion of modular decision support frameworks that are a major player when tools are being unified and simulations to simultaneously calculate Life-Cycle Analysis [19] or environmental impacts [30] are run. One of the difficulties in performing this Big Data analysis or handling BIM and resilience data is utilizing Linked Data methods in order to benefit from the Semantic Web; some tools are working towards this goal as well. The Google spin-off project, Google Flux (Google 2014) now known as FLUX [22], is one project that works with BIM data in an accessible platform geared toward tool interoperability; however, it does not make use of Semantic Web technologies – and thus does not benefit from them – in the way the research presented in this article proposes. Other projects, such as SEMERGY [33], do make use of the Semantic Web but these use practices like taxonomy-based design optimization instead of an extensible and modular approach to data extraction, as our research demonstrates.

Interoperability is no easy goal, but is essential for effective tools in the future. GeoSPARQL[23] helps increase data interoperability to an extent as an OWL[24] compatible RDF[25] ontology and extension to SPARQL used to store and query geospatial information; the Open Geospatial Consortium (OGC)[26] standards themselves are useful spatially, but are not yet fully linked with the full BIM picture. Apache Marmotta[27] is an accessible open data platform effort for translatable data representations across domains (soon compatible with GeoSPARQL) that has an implementation of the LDP recommendation; however, modeling the relationships among structures within a building requires something more geometrically expressive than parts and wholes. RCC [37], another spatial representation, prevents certain 3D cases but does provide an expanded set of eight spatial relationship descriptions allowing expressive distinctions other than coordinates for making connections between spatial entities. SIO allows more granularity, but is made for biomedical fields and while it uses mereotopological relations, there are other geometric concepts that cannot be captured (more in section 7).

Ontology Design Patterns[28] like Agent-Role or Collection-Entity patterns can manage groupings and memberships of entities but not at the spatial granularity intended by the SGA pattern which also allows us to extend and further support the quality and openness of city data such as in the PolisGnosis project[29]. The further goal of the SGA is to help create, predict, and support spatial knowledge graphs [8,15] to ultimately better support DS analysis.

Other projects take another step and work to build an additional interoperability layer to connect BIM ontologies to the larger Semantic Web. Of particular interest to this work is one alignment effort that uses different perspectives (quantitative/qualitative, abstract, domain-specific, multimodal) [23] to organize computational understanding for different alignment focus points [24]. This solution uses a light-weight mereology that describes how the components fit together as well as a basic RCC-8 description plus some mereology[30]; these are aligned with DOLCE-Lite and appear to have the intention of creating an ontology that handles physical entities. This instance is also not the first time there has been a goal to separate out the social intent of a space using non-physical-endurant to represent building and room types (OfficeBuilding, LivingRoom, etc.) [2]; while it has its merits, this approach (using a collection of categories) encounters many of the same limitations that a taxonomy-based approach encounters, including but not limited

[20] FLUX: https://flux.io/
[21] GeoKnow: http://geoknow.eu/Welcome.html
[22] FLUX: https://flux.io/
[23] OGC: http://www.opengeospatial.org/standards/geosparql
[24] owl: http://www.w3.org/2002/07/owl
[25] rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns
[26] OGC: http://www.opengeospatial.org/
[27] Apache Marmotta: http://marmotta.apache.org/
[28] ODP: http://ontologydesignpatterns.org/wiki/Main_Page
[29] PolisGnosis: https://thebridge.cmu.edu/organization/SUDS
[30] http://www.ontospace.uni-bremen.de/ontology/modSpace/

to having restricted, if any, compatibility with the Semantic Web. Certain other implementations such as with OWL[31]demonstrate other conceptual issues for Architectural Design modeling such as the often-encountered issue of solutions only working for select case studies and nothing more. Extensive analysis of *IfcSpace* (spaces) objects and *IfcRelSpaceBoundaries*[32] (walls, etc.) has been completed to be able to convert between formats such as BIM to GIS [6]. While this may only be handling a single data format translation, it demonstrates how the spatial fields are finding it drastically necessary to be able to have data interoperability between several formats and querying tools automatically to answer multi-disciplinary questions [6]. Our solution using Semantic Graphs provides a higher level method to achieve these mappings; other researchers agree that successful and full integration of BIM, GIS, and other spatial data must have systematic semantic mappings to resolve a variety of data structures [4], which our methods offer.

The goal for the implemented Semantic Graph approach presented in this article is to maintain Linked Data principles as well as set up a modular approach that is easily integrated with GeoSPARQL, extensible, and scalable. It was designed to be part of a LDP useful for accommodating, interpreting, and semantically enhancing any BIM, spatial ontologies, or other data formats/taxonomies (and by extension the underlying data) [Figure 2]. To explain it in terms of other approaches of modular ontologies [23], the modules in our solution are essentially at the levels of conceptualization throughout a qualitative and quantitative layering. Further use cases with code as well as alignment explanation is provided in the discussion section of this article.

3.1. Study of Existing Industry Standard Models

The SGA pattern is modeled based on needs for the building domain, however the goal was to make a pattern that can describe any spatial object or space - a generic solution to increase compatibility in the future. Because coordinates and simple bounding boxes cannot represent the level of complexity needed to process and query data efficiently at the pace modern tools need or with the variety of tagging structures used, the geometry types and the SGA resolve the differences found between spatial data standards.

[31]OWL: https://www.w3.org/TR/owl-features/
[32]buildingSMART: http://www.buildingsmart-tech.org/

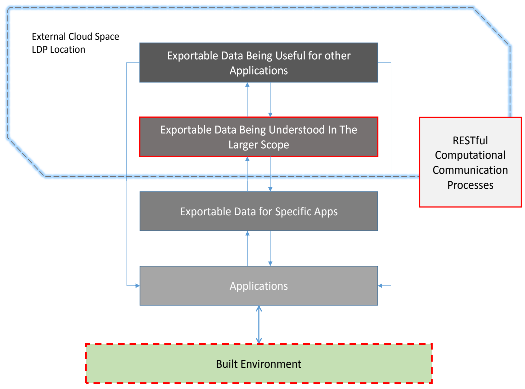


Fig. 2. General BIM Data Processing Hierarchy for the LDP

Section 6 below shows the relationship between three commonly used Building Information Modeling (BIM) standards: IFC, GBXML, and CityGML. Industry Foundation Classes (IFC) is the open standard for any data describing BIM data. It includes ways to determine boundaries, related objects, connection geometries, related building elements, related spaces, and so on. GBXML is the Green Building XML schema and like IFC, it is also open sourced. It has a building component tagging system including building, space, surface, material, etc. breakdown as well, but in a different relational structure than IFC. CityGML, which is used for Virtual 3D City Modeling, can represent not only building information but also full city and landscape models in an XML-based encoding capturing geometry, topology information, and a five-level thematic hierarchy information about object relationships.

This graph structure of Linked Data can be used as a data filter and easily queried to fill in the missing pieces, to verify the information against the larger semantic web of data, connect the data to as many other patterns as is useful, and to finally be able to use the data in many other simulation tools without the need for manual translations. Retrieving relevant data from a BIM instance file requires automation in representing cases where different source schema formats do not necessarily decompose geometries into traditional parts. For example in GBXML, Spaces are directly linked (through object properties) to the surfaces that make up that space. However, to collect those same surfaces from IFC, we have to find them based on multiple searches and matching because the original representation fails to directly connect some of those surfaces with the space level data as traditionally expected (shading surfaces, for instance). The SGA is well-suited for resolving these issues and this is important because all building components and relevant relations need to be captured regardless of what tool generates them to reach further levels of data interoperability.

4. Spatial Graph Adapter Pattern: Link Between BIM Data and Unified Processing LDP

4.1. Spatial Graph Adapter Structure and Extents

The organizational spatial hierarchy for capturing relationships between entities, at a minimum, is a generalized organization of spatial hierarchies seen in the schemas, and we chose to keep the raw geometries separate from the relationships they can embody for modularity in LDP design and also because there are a lot of inherent inconsistencies in these schemas. There needed to be a way to structure the difference in geometry types under some unifying concepts tractable by the SGA. This was the chosen method [Figure 3][Figure 4] because it pertains directly to handling BIM data and also because it easily aligns to other spatial schemas, as discussed further in the discussion sections.

In addition to further defining concepts like geometry alignments and spaces versus voids, a SGA solution to bridge the gaps that exist between BIM and energy schemas was developed after analyzing current major BIM industry standards including CityGML, GBXML, and IFC, their data, and the nature of the proprietary applications that use them. The developmental approach for the SGA was to address the needs of the modular translation adapter applied in the LDP infrastructure, as discussed. To simplify the explanation and scope for this approach, we separate the schema geometry types from the core SGA pattern; while we have fully axiomatized the core pattern in the following section, we only show domain and range restrictions in the accompanying OWL file for the schema geometry types [Figure 3]. For the schema geometry types in Figure 3, we recognize there are mappings to deeper mereotopological relations, but for flexibility, we are only interested in the domain and range restrictions at this time so it is possible to demonstrate how these two spatial mappings work together to extend spatial concept description abilities for our tools.

The SGA pattern is part of a larger, higher level ontology for spatial schemas that talks about semantically similar items from differently contextualized schema standards in a unified manner that corresponds to how a specific domain describes these concepts. It is firstly used to capture the main spatial arrangement of geometric elements extracted automatically from a spatial instance file; in other words, how the geometries are handed to you from an application [Figure 3]. Secondly, the pattern allows us to contextualize additional relationships to capture schema-specific ideas - all in a LD format that lets us ask all the data questions simultaneously. Minimally, a “Spatial Object” is recorded in terms of a location (SpaceCollectionLocation), and at least one spatial element (SpaceCollection); this spatial element can be broken down into 3D or 2D entities, depending on the data provided in the instance file.

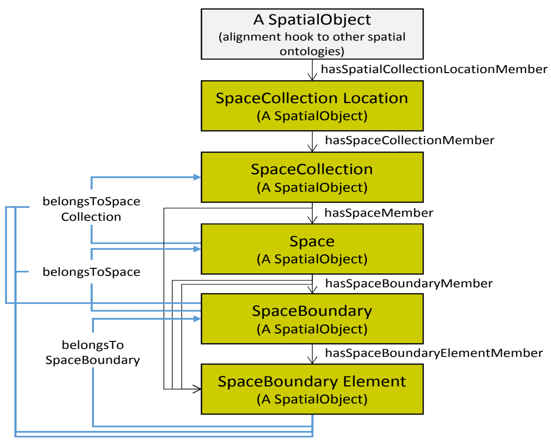
BIM entities are captured as sub-layers of spatial objects, down to material assemblies (or further). This geometry classification structure is crucial in terms of using it in conjunction with the SGA such that we can record any building industry or other spatial information, understand the difference or similarities between them, and know about the surrounding context. Additionally, because we are using a comprehensive and unifying LD graph, our concurrent work can include a joint data parser that automates the queries needed to get the appropriate file data and store this information in the SGA, of course depending on what is provided by a particular data schema.

Using a graph with extended geometric and spatial relations also means there is a solution to not only query forward through a tree of data as many current tools are capable of doing, but the SGA lets us bypass this otherwise forced forward searching, which can be very computationally intensive and limiting in terms of inferencing ability. Furthermore, once there is a mapping for a type of data schema, then that can be stored and referenced to enhance future training sets for machine learning.

4.2. SGA Pattern Entities and Axioms (See Appendix A)

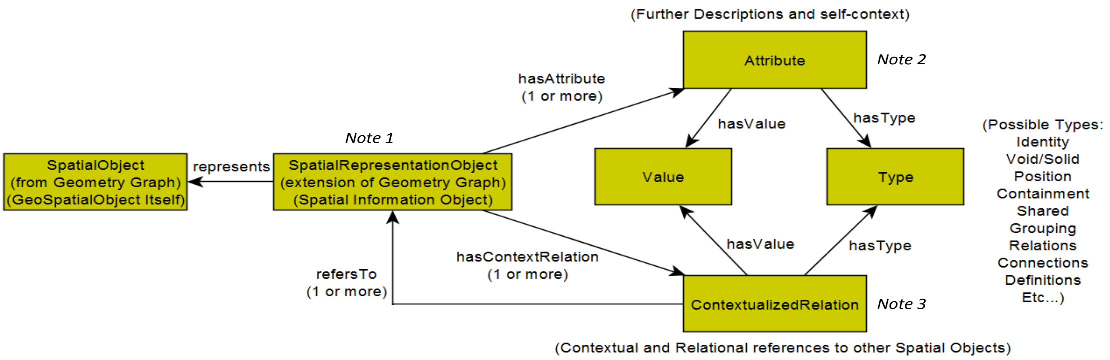
In addition to describing what the 2D and 3D geometries are for a given location, our research starts to build in layers of context for those geometries by mapping this ODP to other spatial schemas. This is where the core SGA pattern [Figure 4] can be derived for these research needs. Since each of the types of geometries represented in Figure 3 are subclasses of Spatial Things, this means that any or all of the classes and

Top Level Spatial Hierarchy for the Organization of BIM Related Geometric Elements



Spatial Object Types	Definition in Context	Example Use Scenario
Space Collection Location	Location of the geometries described in the graph instance.	Commonly the Latitude, Longitude, and Elevation.
Space Collection	Identifier of relevant geometry groupings (3D, 2D, points).	For example, a Building in its entirety.
Space	3D geometry elements in the SpaceCollection.	3D Enclosure in a Building (Room, Elevator etc.).
Space Boundary	2D or 3D geometries 1) creating the Space or 2) part of the SpaceCollection, depending on the schema.	In BIM, typically the Walls, Ceilings, and Floors, or even Shading or Retaining Walls.
Space Boundary Element	2D or 3D geometries 1) creating the SpaceBoundary or 2) part of the SpaceCollectionLocation, SpaceCollection, or Space.	In our research, these are Material Layers within Walls or Landscaping.

Fig. 3. Spatial Hierarchy for Organizing Common Geometry Types within BIM



Note 1: **SpatialRepresentationObject** can represent any of the Classes in the Top Level Spatial Hierarchy
Note 2: Using this SGA Pattern, **Attributes** can be used to capture data like coordinates, and,
Note 3: **ContextualizedRelation** can be used to capture relationships to other spatial objects such as how surfaces align (inner, outer, center line) or how they should be perceived by other systems (void, solid, etc.)

Fig. 4. Spatial Graph Adapter Pattern: Core Ontology Design Pattern for Spatial and BIM information

properties diagrammed in Figure 4 can be employed to further contextualize each of the Figure 3 geometries. Additionally, these allow a greater granularity of spatial relationships including the ability to potentially reduce processing efforts. For example, depending on the schema being mapped, we would need to be able to capture the fact that they each use their own coordinate system and the geometric properties alone from Figure 3 cannot explain this concept. However, if the geometry types from Figure 3 are used in conjunction with the core SGA pattern, then we have a method to fully contextualize the spatial objects from a variety of domains; it also helps to consider some of these classes to be roughly analogous to features in GeoSPARQL. The full set of axioms are described in Appendix A and the OWL file can be found either on the ODP website[33] or on GitHub[34].

[33]SGA-ODP: http://ontologydesignpatterns.org
[34]SGA-GitHub: https://github.com/HollyFerguson/Spatial-Graph-Adapter-Pattern

5. Use Case Scenarios for BIM Data

5.1. *Case Study 1: Populated SGA Pattern Informing Structural Dependencies in the Event of Damage*

The SGA allows us to capture term-to-term comparisons as well as information about how pieces fit together and relate to other proprietary schemas and in-file contexts. For example, a Wall in a given GBXML file may refer to the same Wall in the same building of a CityGML file, but you cannot say they are equal or have a direct comparison of coordinate points because walls in GBXML are referenced differently in terms of joining on the center line of the wall versus the outer or inner side. However, if we can enter both of these data models into the SGA graph with the relation distinction "connectedBy" with type being "centerline" for these adjoining parts, then a fair distinction can be made and recorded. Consider a more in-depth example use case for the SGA pattern (green/yellow/soild) extending the original geometry types (blue/dashed) [Figure 5]. How would we use the geometry information provided to populate the SGA pattern such that we could ask questions such as: What exposed columns are holding up Floor 3 (or would not hold up Floor 3 if damaged in a natural disaster)?

Instead of needing to filter through and process all coordinate information to see what parts of this model might effect other parts or assume we can use something like elevation (which is not helpful if based on starting position as seen in the example study), this knowledge is represented explicitly in the SGA pattern, and we can therefore automatically make queries to produce all of the relevant column-type structures, per this example. This delineation is needed because we may need to include columns that are in adjacent spaces or even in other walls, but are technically part of the same group holding up, in this case, the floor in question. Other columns may be considered spatially adjacent or in closer proximity than the ones we need to get back as query results, or they may not be structurally holding up this example floor. Asking coordinate data what building elements may exist at a certain elevation is also not enough to gather the specific columns needed as a group for this scenario because there is missing context or perhaps even columns that may extend through other surfaces. From this graph snippet, we can ask directly what columns are solid, what floors they are in fact connected to, and then retrieve the specific 3D coordinate details desired - no need to evaluate all geometries and do advanced and complicated coordinate processing [Figure 5].

Furthermore, in building design, there are often geometries called "adjacent" to other geometries such as SpaceCollection2 in Figure 5. This type of situation is often where architecturally designed elements act as landscape framing devices or even as wind breaking or retaining walls, for example, but are not always organized in an XML output file as relating at all. Grouping objects as adjacent is not even a possibility depending on the type of data we are reading or depending on the exact locations of these geometries. If this same second collection of geometries were a retaining wall it would greatly effect wind direction and thus pressure on a building, making it vital information to capture for engineering simulations. The SGA can build and capture these relationships that other spatial organization schemas cannot capture with the same level of specificity. Now answering questions such as "to what extent are these pieces considered adjacent" is possible and more accurate.

5.2. *Case Study 2: Query Example to Competency Question: Within N-block radius of my building (IFC or GBXML), what other structures (CityGML) are most likely to effect both the thermal energy (GBXML) and wind speed values (ASCE 7[35]) being used in my simulations?*

First, consider what types of information are needed to answer this question; there need to be engineering models of the building of interest, city information, climate/shading information (ex. Energy Plus[36]), elevations from at least two different sources to compare, and so on. Furthermore, to avoid errors in manual translations of data to and from several disjoint analysis tools or resources, this information should be able to be used automatically through semantic translations (which our solution offers). Since models often come in different formats, this method allows for data validation and omits human error as well as does not require the need to change engineering practices or native data formats - i.e. practice can keep the engineering models in Revit, for example, and then still analyze that model with CityGML data, etc.

Specifically, this question is relevant to engineers, for instance, who want to know accurately how a newly proposed building (IFC/GBXML) will be im-

[35] ASCE 7 Windspeed: http://windspeed.atcouncil.org/
[36] Energy Plus: https://energyplus.net/

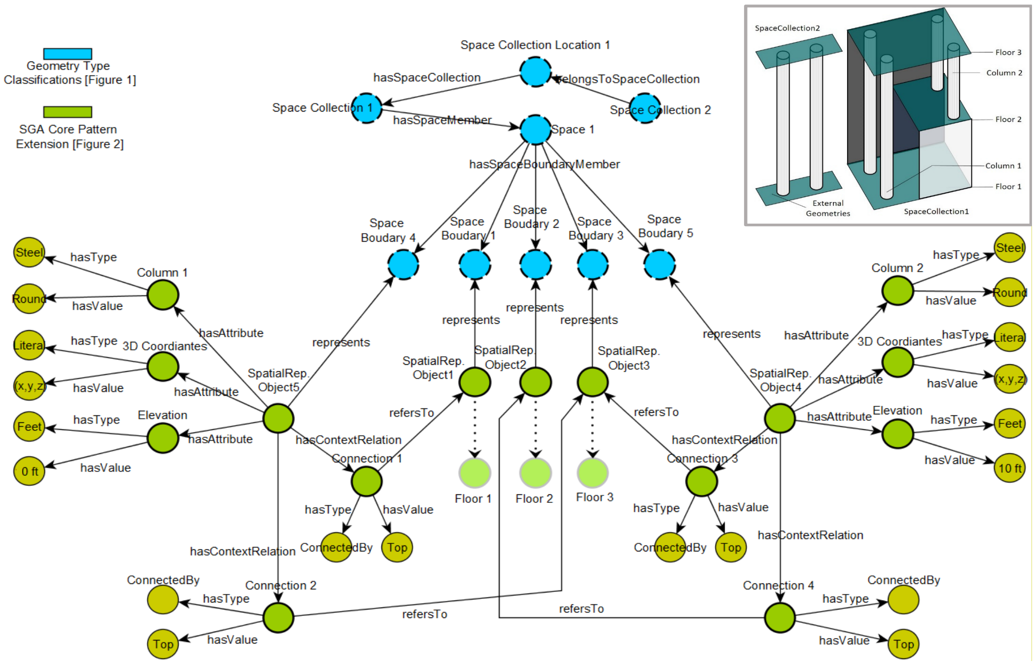


Fig. 5. Example Application for Describing Column Spanning and Adjacency

pacted by wind speeds given precise surroundings (CityGML). A typical practice is to use ATC wind speeds for a given latitude and longitude; however, a more intricate breakdown about the surrounding elevations (wind effects) and material information (thermal effects) can be acquired through semantic analysis. Additionally, more context in the data used for analysis means other types of risk can be calculated such as potential damages given certain environmental disasters where elements of surrounding structures could damage the building in question. If your building is shorter than nearby structures there can be more surrounding direct risk of damages from other buildings in a hurricane, but there can be more risk of stronger wind (effects both thermal and structural wind loads) if your building is taller than your surroundings. Semantic analysis allows all of these perspectives to be considered together based upon sequences of query results.

Using the Linked Data Platform [17] and LD Views [18], the data needed for the Semantic Graphs to be constructed to provide such answers can be gathered from potentially a variety of distributed resources or user-based input since it is all transformed into Linked Data. Consider the example query sequence used to answer such a question as that posed in this section; for the sake of this example, an auto-generated set of city information[37] is generated to test the sequence and validate its findings against the results (other types of CityGML 2.0 models can be used as well) [3]. Figure 6 shows a cluster of simple buildings that surround what is identified as the newly proposed building to be evaluated. Semantic Graphs are constructed for the CityGML space, the IFC model of the new construction and the GBXML of the new construction. Semantic alignments can then be established based upon the location of the new structure placement in the CityGML and the set location of the new structure given in the IFC or GBXML model using constructs such as *owl:sameAs* [Figure 6].

Based upon the Semantic Graphs containing all of the spatial data needed to answer this question, queries can be automated that help reach a more contextual understanding to answer this question. Using a GeoSPARQL[38] enabled graph database and server framework (Virtuoso[39] and GraphDB[40] have been investigated) combined with the Semantic Graph data se-

[37]Random3Dcity: http://github.com/tudelft3d/Random3Dcity
[38]GeoSPARQL: http://www.opengeospatial.org/standards/geosparql
[39]Virtuoso: https://virtuoso.openlinksw.com/
[40]GraphDB: http://ontotext.com/products/graphdb/

rialized in Turtle[41] (or RDF), it is possible to retrieve all of the City Structures that are available to consider in the scenario, called “SGA:SpaceCollection” from a given Level of Detail [Figure 6]. In this case, the surrounding building footprint geometry is the only part needing to be considered since the corresponding elevations and other information are already stored in our Semantic Graph, reducing the amount of building data to query over (See first query example: Figure 6, right).

Next, an example GeoSPARQL query is shown with code and results that performs a routine elimination of all structures from the previous query that are outside of a given distance [Figure 7]. This example uses a radius of 100 meters, however this can be defined user input as a specific application needs. This chosen distance allows consideration of all of the nearby structures that are adjacent to the newly proposed building known in IFC formatting. Based upon this second query, we show a subset of the results returned from GraphDB (in our example there are 24/100 total building returned).

This gives insights into not only what is contained in the original IFC/GBXML model, but it also provides external context to the surroundings from data of different formatting such that we can easily know what surroundings are more likely to influence our structure, and by how much [Figure 8]. For example, the query returning elevation information from our semantic graph based upon the location of our IFC model shows that the surrounding buildings that are 3-4 proper levels taller [Figure 8] will probably highly influence the shadows cast upon the proposed building (influencing thermal calculations) as well as alter the wind pressures (influencing structural calculations), and so on. Ultimately, this aides in the users ability to accurately consider such knowledge when making choices for the proposed building.

As an important note, it is acknowledged that there are still certain types of imperfections when translating coordinates between different reference systems, in terms of preciseness of geometry connections [12]. This contribution is instead about automating semantic translations and gaining knowledge through semantic mappings and semantic resolution of concepts useful for various spatial formats.

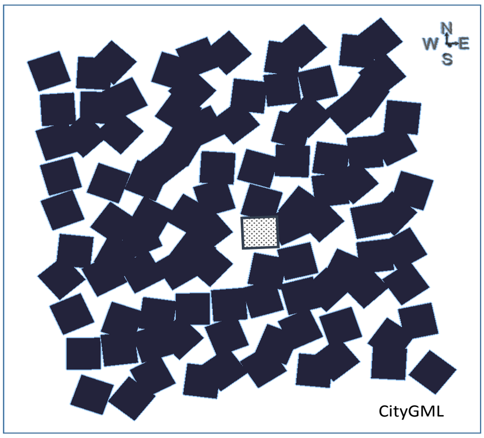
6. OGC Compatibility with Existing Standards

At this point, our implemented pattern (plus joint parser) is capable of formatting data organized according to the SGA schema into the Open GeoSpatial Consortium representation, compatible with the Simple Feature Access profile.[42] The framework can be easily extended to produce output in additional formats; OGC was tackled first because it is of special interest to our efforts due to its capability to perform spatial analysis processing involving properties such as union, intersection, difference, etc. When converting data organized according to the SGA pattern to the OGC file format, the relationships necessary to understand are the OGC geometry classes, such as Point, Polygon, etc. [Figure 9]; it is also possible to include literals, comments, and so on as well as to relate elements to one another via references and grouping IDs. The SGA pattern allows us to capture more information; it is able to capture these types of geometries but also to describe the relationships between these different OGC parts. Furthermore, it is necessary to store the relation data that is in the original SGA graph instance somehow in the OGC standard so that 1) re-creation of the SGA for reverse compatibility is available in the future and 2) full and generic vocabulary generation is possible. [Figure 9].

Translation into the OGC standard was a major part of this proof of concept because it creates interoperability between the building domain of knowledge and other spatial analysis efforts. The OGC contributes a major amount of effort toward creating high quality open sourced standards for the geospatial community worldwide and works to unite professionals from many disciplines or organizations including research, government, commercial, and more. Compatibility with OGC allows building and energy simulations to have access to a much larger scope of data, web map services, geography markups, and observational features to name a few examples. It also will eventually give tools a way to get OGC formatted data from anywhere and use the associated SGA instance to form the data into any file type usable in proprietary applications (part of ongoing research efforts). Furthermore, our data can now be compatible with the GeoSPARQL query language and other RDF stores in an automated or semi-automated manner. OGC compatibility resulting from schema format translations also benefits DS

[41] Turtle: https://www.w3.org/TeamSubmission/turtle/

[42] OGC Standards: http://www.opengeospatial.org/docs/is

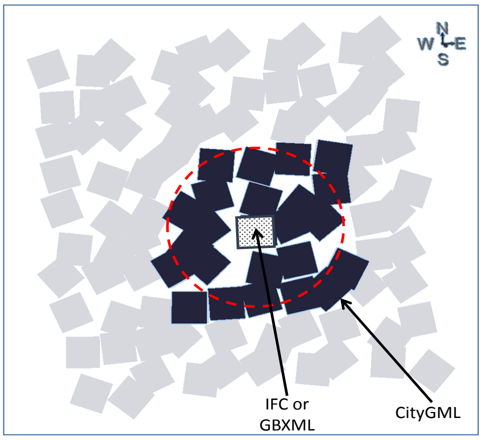


Ask Semantic Graph for City geometry around IFC / GBXML building of interest, noted in center of CityGML Schema (left)

```
for row in CityGML_Graph.query("""SELECT ?s ?o ?v
        WHERE { ?s ?rdf ?o .
            ?s ?hasProperty ?prop .
            ?prop ?hasType ?literal .
            ?prop ?hasValue ?v . }""",
    initBindings={ 'rdf' : rdf.type, 'o' : 'SpaceCollection'),
                'hasProperty' : property,
                'hasType' : type,
                'literal' : 3D_Coordinates,
                'hasValue' : value ) } ) :
    # For each city Building ("row"), collect geometries
    # Then translate WKT triples in GraphDB Triplestore
```

Fig. 6. Query Example 1: Ask Semantic Graph for relevant SpaceCollection level geometries.

GML: "POLYGON((300.0 240.0 0.0,353.981 238.595 0.0,355.385 292.5776078 0.0,301.404133639 293.981741438 0.0,300.0 240.0 0.0))"
= owl:sameAs IFC Model of New Construction at this Location, & owl:sameAs GBXML Architectural model at this lat. / long.




```
PREFIX geosparql: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX wkt: <http://www.opengis.net/ont/sf#>
PREFIX units: <http://www.opengis.net/def/uom/OGC/1.0/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?s (geof:distance(?o, "GML POLYGON ((300 240 … , 240 0))"
                ^^geosparql:wktLiteral) as ?distance)
        (geof:distance(?o, "GML POLYGON ((300 240 … , 240 0))"
                ^^geosparql:wktLiteral) < 100 as ?true)
WHERE
  {
    ?s rdf:type    wkt:Polygon .
    ?s geosparql:asWKT    ?o .
  FILTER ((geof:distance(?o, "GML: POLYGON ((300 240 … , 240 0))"
                ^^geosparql:wktLiteral) < 100))
  }
```

	s	distance	true
1	http://www.sw.org/UBO#SpaceCollection36	"48.79412023374766"^^xsd:double	"true"^^xsd:boolean
2	http://www.sw.org/UBO#SpaceCollection64	"4.843206906306947"^^xsd:double	"true"^^xsd:boolean
3	http://www.sw.org/UBO#SpaceCollection66	"8.584224346653093"^^xsd:double	"true"^^xsd:boolean
4	http://www.sw.org/UBO#SpaceCollection33	"93.83190065740051"^^xsd:double	"true"^^xsd:boolean
5	http://www.sw.org/UBO#SpaceCollection77	"93.38503150685162"^^xsd:double	"true"^^xsd:boolean
6	http://www.sw.org/UBO#SpaceCollection65	"0.0"^^xsd:double	"true"^^xsd:boolean
7	[illegible]	[illegible]^^xsd:double	[illegible]^^xsd:boolean

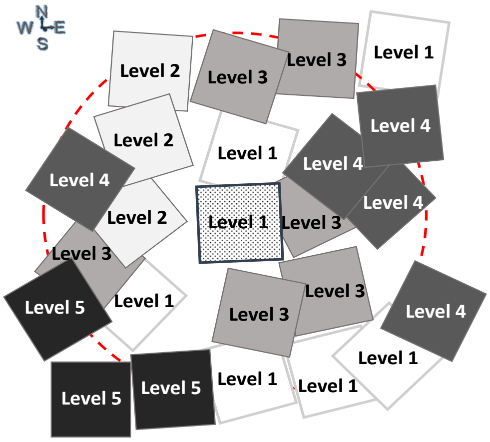
Fig. 7. Query Example 2: Reduce the whole set of buildings to include only ones adjacent to the new model.

efforts. Bridging various data types and across several domains means that DS and Machine Learning can have a broader set of information and knowledge with which to work and make conclusions.

7. SGA Functioning in a Larger LDP Context

7.1. LD Views To Map BIM to SGA Instances

Certain BIM schemas - in particular CityGML, GBXML, and IFC - only handle data effectively for a



```
Ask Combined Semantic Graphs of IFC/GBXML with CityGML
subset of Buildings from query results for Elevations

for row in Combined_Graph.query("""SELECT ?B ?type ?val
        WHERE { ?s ?rdf ?o .
            ?B ?hasProperty ?prop .
            ?prop ?hasType ?type .
            ?prop ?hasValue ?val . }""",
      initBindings={ 'rdf' : rdf.type, 'B' : 'SpaceCollection'),
                'hasProperty' : property,
                'hasElevationType' : hasType,
                'hasValue' : hasValue ) } ) :
      # Where ?val is 'elevation_value_meters'
      # Then can build subset .gml files, visualizations, etc.:
      #     Level 1:    3.048 m
      #     Level 2:    6.069 m
      #     Level 3:    9.144 m
      #     Level 4:   12.19  m
      #     Level 5:   15.24  m
      # (1 Level = 1 Building Storey)
```

Fig. 8. Query Example 3: Populate and visualize a new set of data by retrieving the contextual data for the new subset of city structures: Elevation/Shading influences from lower levels to higher levels are represented from lighter coloring/labels to darker coloring/labels, respectively.

OGC Object Options		IFC (Examples)	CityGML (Examples)	GBXML (Examples)
•	Point	***IfcCartesianPoint*** or ***IfcSite*** (where there is a Lat/Long)	***gml:pos*** or ***gml:posList***	***CartesianPoint***, ***Location*** (Lat/Long)
	LineString	***IfcQuantityLength*** or ***IfcAxis2Placement2D***	***gml:posList***	***PlanarGeometry***
	Polygon	***IfcRectangleProfileDef*** or ***IfcRelSpaceBoundary***, ***IfcMaterial***	***gml:Polygon*** or ***gml:surfaceMember***	**Surface**, ***SpaceBoundary***, ***Construction***, ***Layer***, ***Material***
	MultiPoint	ex. set of ***IfcRectangleProfileDef***	ex. set of ***gml:pos*** or ***gml:posList***	ex. set of ***CartesianPoint***
	MultiLineString	(ex. from combinations)	***gml:posList*** or ***gml:LinearRing***	(ex. from combinations)
	MultiPolygon	***IfcWallStandardCase*** + ***IfcWindow***	***gml:MultiSurface*** or ***bldg:Building***	***Space*** or ***Construction***
	GeometryCollection	***IfcSpace*** or ***IfcBuilding***	***cityObjectMember*** , ***bldg:Building***, ***grp:CityObjectGroup***, or ***gml:CompositeSurface***	***Space*** or ***Building***
	PolyHedron	***IfcBuilding***	***cityObjectMember or gml:Envelope***	***Building***
	Tin	-	-	-

Fig. 9. Example of Initial Studies of Alignments and Translation Possibilities between various BIM Schema Formats, Including Adapting to the OGC Standards (Note: Final OGC representations can have several correct forms, depending on what relations are intended.)

subset of applications or for scenarios tailored specifically to one of these data structures. LD Views can be used once to establish a set of terminology and relationships that are relevant for a particular schema while maintaining LD standards and modularity; however, since there are a lot of overlapping concepts between schemas, the exact way we make these connections and alignments is more involved because while two schemas may talk about the same *surface* or *wall*, they do not describe it in exactly the same way or with the same descriptors [Figure 10]. One example is where normal alignment can state that walls in two different models are the same geometry, but our work also lets us establish if the given coordinates are referencing different coordinate systems or are based on different axis of the wall (inner edge, outer edge, or center-line). The first step is to identify what part of a computerized building design is important for simulations (similar to how geometries are essential to thermal changes) and then to extract the more minor differences that will affect automated understanding. For example, prior to our work using LD Views, individual parsers would

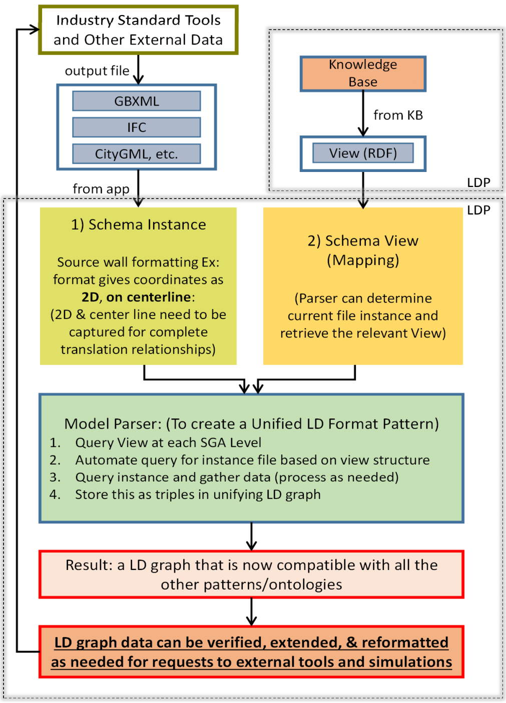


Fig. 10. LDP Processing Schema Using LD Views

have been needed to partially understand every data type encountered. Views allow us to capture all the additional spatial or material information that basic term matching cannot provide.

Most importantly, the Views or *windows to understand data formats* mean that only one parser or data translation processor is needed to query any View since they are following the same spatial graph. The effects of this are twofold: 1) it means that this data can be seamlessly extracted into an ontology design pattern for SW compatibility and 2) we have a way to establish the otherwise untraceable differences that exist when comparing data from different schemas (yielding a way to express a more nuanced spatial granularity) [Figure 10]. The ontology design pattern that is used in this new parser (and associated LDP) is referred to as the Spatial Graph Adapter (SGA) pattern. It allows data to be passed back and forth through the REST APIs of the LDP in a format that is compatible with the LDP itself. Because there is a common parser for all Views with a cloud-based infrastructure, it means all desired data can be parsed automatically, reformatted for use in simulation models at external locations, and can be automatically queried to allow any gaps to be filled in and the rest of the data to be verified. The SGA pattern is intended (and implemented) to function as a "bridge" or translation-enabling data structure which semantically unifies proprietary spatial data formats, BIM, geometric representations, dimensional representation gaps, spatially extended geographical contexts, domain specific needs within the building industry, and finally to allow the level of spatial granularity needed such that our information can be properly consumed in Linked Data Platforms (LDP)[43] enhanced with provenance layers [34]. This SGA graph structure itself and its integration with LD Views and the larger infrastructure is part of the extended work and takes the form of an axiomatized ontology design pattern and modularized schema parser that can read and automate queries from individual LD View files.

While the SGA research addition can handle information sets that include buildings themselves, it is also able to capture spatial information pertaining to a building's surroundings, topography, city-scape or other larger regional entities. The standard used for this set of broader city level information is CityGML, as mentioned earlier, and this schema uses a concept called Level of Detail (LOD) to organize their spatial information. There are 5 LODs ranging from regional topography with basic massing models to fine grained material and furniture data (level 4). Each LOD can be considered a sub-feature in GeoSPARQL and can also be assigned as a spatial attribute in the SGA pattern. Capturing the LOD data in the SGA graph means it can then be linked through other patterns and associated by that particular LOD to models that come in formats such as IFC or GBXML, creating the data interoperability sought. For instance, city level information can be linked with the Relative Relationship Pattern (RR)[44] to building models in schemas such as IFC or GBXML in order to give city-level context per the LOD provided where there would otherwise be none. Furthermore, LOD in CityGML can be aligned as a type of Level of Detail in the RR pattern and then it is possible to further extend the pattern instance to resolve alignment differences such as relating models that are described using different scaling systems or for translating between differing coordinate systems or frames of reference as is often the case with data from various industry tools.

While the SGA pattern has its own goals and usage, we acknowledge that in several cases there could be benefits to aligning it with other mereotopological or

[43] LDP: https://www.w3.org/TR/ldp/
[44] RR: https://curate.nd.edu/show/9k41zc79w4r

SIO concepts. In any case, consider an example: defining two spaces and spaces between them for building industry data. RCC [37] lets us state that one thing is within another (non-tangential proper part (NTPP)), but neither to what extent, to what purpose, or if it should be handled as a space, void, or something else. Some ontologies propose what a void is [22,7], but in BIM we need more context because any space could technically be a void. We must consider occupyable spaces differently than air spaces within wall assemblies which are closer to voids as per building industry uses, and furthermore we need to resolve confusions for scenarios such as mechanical HVAC spaces which are both (voids and occupied in some manner). SGA in conjunction with techniques such as Views[45] let us create computationally automated understanding of these and other types of distinctions.

7.2. Query Automation Opportunities

A feature of this methodology that is of particular importance is the ability to fully automate all of the querying that needs to take place when mining information from any of the BIM formats in question. Because we are working with LD Views and therefore have consistent, encoded patterns of mapping allowing a computational understanding of each spatial data format, it is possible to include automatic query parsing and processing. That is, the LD Views tell us where to look in a BIM document for relevant data, and the single parser can therefore use the given structure (that comes in one of several patterns) to set the variables needed to make each SPARQL query, then execute the script, and finally collect the results. Consider the example shown in Figure 11 (which is further explanation of Figure 10 part 2 in the Model Parser).

Figures 10 and 11 demonstrate how the single parser requires the specific instance file (typically describing a building and its surroundings in energy and resilience simulations) and the respective LD View File. Figure 11 gives an example snippet from the IFC version of a View in the form of RDF where it records where and how to find the spatial location information (only part is shown for simplicity). For this example, the LD View itself is parsed into a Python Readable RDF graph and a separate SGA instance graph is created to record and organize the data that is mined. This is performed as JIT population since it is intended to

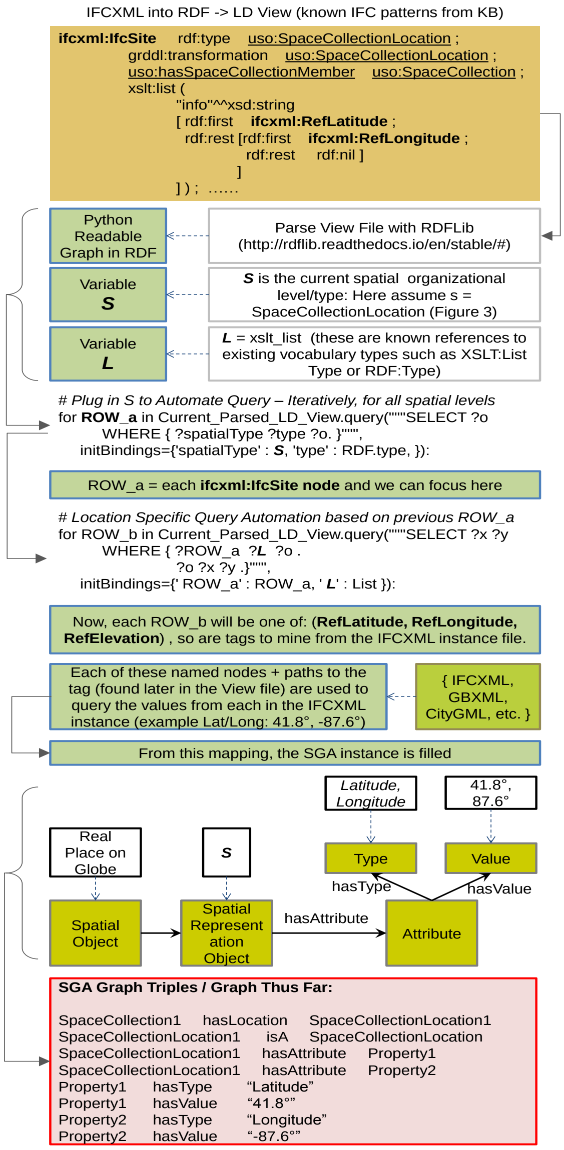


Fig. 11. Reusable Method for Automating Spatial Queries

be dynamic mappings as data is available; the resulting Semantic Graph is an in-memory structure for answering further queries as needed. Other data is visible at this point such as the spatial levels (Figure 11 "S" which are the same as Figure 3 hierarchical levels) and external vocabulary references (such as RDF, GRDDL[46], etc.). These references and known spatial types are combined as they are processed into the vari-

[45] Views: http://ceur-ws.org/Vol-1461/WOP2015_invited-talk-abstract.pdf

[46] GRDDL: https://www.w3.org/TR/grddl/

ables needed for SPARQL queries; they can be reused for the next set of spatial types encountered in the View as relevant items. There are two examples of how the BIM standard information is combined with the known SGA structure to automatically determine what to gather from the BIM instance file (here the IFCXML file), where to find it, and how to place it into the new SGA graph based on the View mapping for the current spatial type being parsed [Figure 11].

Once the parser knows what spatial items are important and where and how to find them, searching and retrieving that specific data from the BIM specific instance is as straightforward as reading any XML file, with the results stored in the new SGA graph as Linked Data. Figure 11 shows what data would be assigned to the different parts of the SGA pattern for this simple example, and then what triples that might produce toward the final SGA graph structure. Since these queries are being made as the processing happens, it is easy to computationally re-construct them for any spatial type; generally, it is only necessary to add a new query when a new XML informational structuring pattern is found outside of the known set, else they are automated and reused as explained in the image. In a similar set of processing routines, queries are automated to extract the data now in a common RDF graph to send to external formats and simulation tools; OGC is one of the output types handled thus far. The intent is for a user to present a question then only perform this processing on the necessary pieces of instance data, using only as much processing power as required to build the graphs. Additionally, the decisions of this approach can be thought of as an ETL lifter using RDFLib[47] which also makes future application enhancements using SPIN[48] rules a possibility.

7.3. LDP, LD Views, and SGA Working Together

To recap, the SGA Pattern and associated implementation of this component is the main addition to existing work as a facilitator between the Linked Data Platform and the individual Linked Data Views for each version of BIM data. This pattern provides a technique to achieve a common Linked Data structure and semantic mappings that are able to capture all of the relevant data and object relationships that can be extracted from a set of BIM or spatial data. The SGA is the link between using the LD Views (that can be stored in and retrieved from a Knowledge Base) and the more extensive LDP components that can now be utilized due to the common formatting of the BIM and resilience data [Figure 12].

The SGA is the mechanism within the LDP that works directly with a single parser; having a single parser and a single data structure that can be formed from any LD Views means that all of the querying and data communications between these components can be easily created as needed automatically. While, to the best of the author's knowledge, there is not a similar system to compare processing, the construction of these Semantic Graphs currently takes between 5.78 seconds and 12 minutes for a range of a single room building throughout a multi-story, 60+ room office building, respectively. In terms of the complexity this involves, graph construction depends upon the selection of relevant components to add to the Semantic Graph including but not limited to modular sets of base geometries, windows and doors, individual material information, structural components such as beams and columns, and levels (or stories for targeted engineering wind load simulations). In reference to Figure 12, the SGA and associated parser implementation now allows the following infrastructure to become functional and fully automated:

1. There is a place for the various types of BIM and resilience data to be brought for processing as well as there is a place to store references to the various storage locations of the LD Views and useful ontology design patterns such that they can be accessed and pulled into the parser as needed for automation [Figure 12 (1)]. Note: While the LD Views and other platform components are implemented in this work, their development and full explanation are outside the scope of this article but are part of a concurrent pending publications.
2. The processing parts of the overall infrastructure can be seen in Figure 12 (2). This section explains the function of the automation process using BIM data and LD Views. The end result is the filled SGA pattern instance; this instance is now independent from the original proprietary data formatting and can be easily queried further to add any additional data that may be relevant to begin using it in external simulations or filtered to be used with other external tools. A complete infrastructure makes existing work flows and preferred industry tools remain in place for

[47] RDFLib: https://rdflib.readthedocs.io/en/stable/
[48] SPIN (SPARQL Inferencing Notation): http://spinrdf.org/

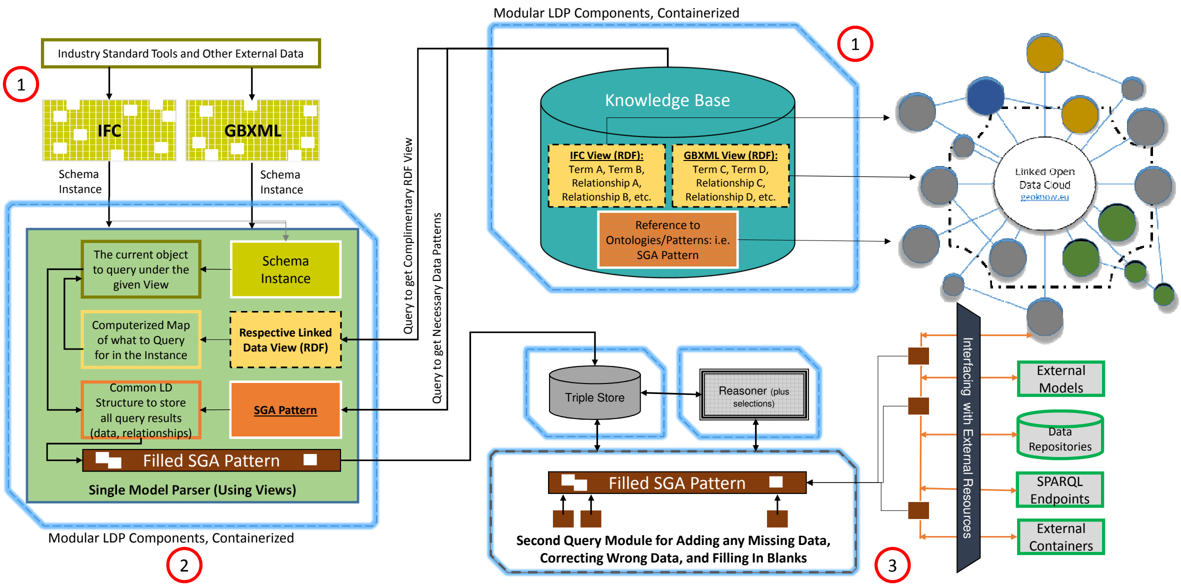


Fig. 12. How the SGA enables LD Views to work through a LDP

use, but means that additional simulations and data analysis are now possible without needing to change those existing work flows - but they can still benefit from a much more complete and Semantic Web compatible data analysis.

3. An additional benefit to this solution is based on the fact that a common SGA instance means it is easy to also automate queries that can be used to find the missing pieces of information in the SGA instance as well as ask external endpoints for data to fill in what is missing or verify the data being used in the instance (for example material property values) [Figure 12 (3)]. It is possible through this method as opposed to other solutions because the SGA instance is already and instantly compatible with any other external LD.
4. Beyond this, HYDRA is useful for incorporating and validating data that may not yet follow LD principles [17]; this means it handles data that is both RDF and non-RDF. It can also set up sets of services [11] for applications though semantics. For this, semantically enabled data and metadata are needed for achieving truly interoperable cross-domain communication and HYDRA gives machine interpretable service descriptions for useful development implementations. HYDRA and JSON-LD allow the automatic composition of services, i.e. it is possible to computationally automatically find options to assemble a set of semantically annotated services [11], which means that a set of best-fitting services can be searched and found based upon requirements. This is possible because modern services are API-driven so mappings can be created and used to retrieve data objects [18]; HYRDA helps this goal because many of these modern services we are trying to exchange information with are already using REST APIs which means there can be meaningful connections with the underlying structures.

8. Discussion and Critical Evaluation

8.1. Alignment Possibilities

Several entities within the ontology design pattern that forms the core of the Spatial Graph Adapter [Figure 3][Figure 4] have intentionally been left vague. This enables the pattern to remain relevant for a wide variety of applications. Such entities can be then modeled to an arbitrary degree of fidelity, from a flat “strings-as-things” approach, to complex ontology snippets in their own right. This follows best practices described in works depicting the capabilities and limitation of OWL [31]. Of course, for the pattern to be useful in a particular application, these loose threads must eventually be pinned down. In this section we do just that, by aligning the model to existing related ontologies in a way that supports the overall project described in this paper.

The need for modular alignment between resources is not a new idea, although one that still needs quite a bit of development. Certain projects already have worked towards aligning IFC to the Semantic Sensor Network (SSN/SSN-DUL)[49]; however, there are still certain limitations to this and other approaches to modularizing spatial information such as falling into the limitations that restrictive taxonomies introduce when used alone. The project utilizing organizational perspectives (quantitative/qualitative, abstract, domain-specific, multimodal) [23] starts to complete these upper level alignments by defining their own RCC8 ontology whereas the presented approach began by modularizing LDP components and by using GeoSPARQL as well as the existing RCC-8 ontologies plus other relationships that the proposed SGA pattern can establish.

SSN typically describes sensors, their observations, and similar concepts. The SGA was designed to be flexible enough to incorporate sources and descriptors from several ontologies or new ones that are not in use yet. For example and depending on the spatial qualities being described, SGA attribute:type and attribute:value can capture any relationship or spatial distinction, and align to existing concepts such as DUL (hasPart, hasQuality, isObjectncludedIn, etc.). However, the SGA allows spatial descriptors from many resources (RCC-8, SSN, GeoSPARQL, SGA, etc.) to be connected within the LDP and therefore analyzed together, instead of being forced to use only one resource or format for the build environment. Mereology in this solution is also relevant since we want to be able to align to existing RCC representations. To this end, Building Element as seen in Figure 13 is also a subClassOf each spatialThing.

Following in the footsteps of other ontological BIM efforts (we can reference the Modular Ontologies for Spatial Information book here), we align many of these entities to corresponding concepts within the DUL module of the SSN/SSN-DUL ontology. However, there are some key differences between our alignment work and that of other approaches. In particular, some other solutions make simplifying assumptions such as âĂIJthe x-dimension in an architecturally rendered building always corresponds to the height or width of a modeled geometryâĂİ when designing their alignments. A guiding principle of the work presented here was to avoid such assumptions at all costs. Additionally, we do not define our own RCC-8 ontology, but rather model the spatial relationships at a more general level (based on how the model was actually drawn) and use GeoSPARQL and the existing RCC-8 ontologies to achieve the same end. Other existing data patterns, such as the Relative Relationship (RR) Pattern33, can then be used to resolve the remaining discrepancies between different schemas, for example where we see models that have to be analyzed together but come from different Cartesian coordinate systems where reference points cannot be guaranteed to be the same.

At first glance, aligning against the SSN ontology might seem like an odd choice. This ontology was originally designed to model sensors, their observations, and similar concepts. However, the DUL module within the SSN has several relevant properties, including hasPart, hasQuality, isObjectIncludedIn, among others. Further, many important applications related to the built environment involve a variety of sensors measuring everything from which parts of a building are occupied to structural tension on key components of the structure.

Additionally, we align relevant SGA entities to two more general upper level ontologies: the Descriptive Ontology for Linguistic Cognitive Engineering (DOLCE) [5] and the Basic Formal Ontology (BFO/BFO-SNAP) [20]. Aligning to DOLCE allows us to model the social implications of a spatial object, such as zoning, occupancy, etc. Aligning to BFO-SNAP allows us to associate entities with temporal features. This is an important extension because buildings analyzed for energy and resilience exist as they are for a certain amount of time - either for a lifespan or while experiencing the effects of natural disasters or other discrete temporal events. Figure 13 shows the alignment of the SGA model to relevant portions of SSN-DUL, DOLCE, and BFO-SNAP.

At a conceptual layer, Dolce:physicalProperty aligns to SGA:attribute, where the referenced physical entity is the SGA:spatialRepresentationObject [Figure 4], and these entities can inherit any SGA:Attributes or SGA:Conceptualized Relations that are present in the SGA instance. For example, DOLCE:feature gives a place to connect to GeoSPARQL Spatial-Thing as an endurant and then pyhisical-endurant provides a place which lets us add social implications to a spatial object such as zoning, occupancy, etc. BFO is used to

[49]SSN: https://www.w3.org/2005/Incubator/ssn/ssnx/ssn

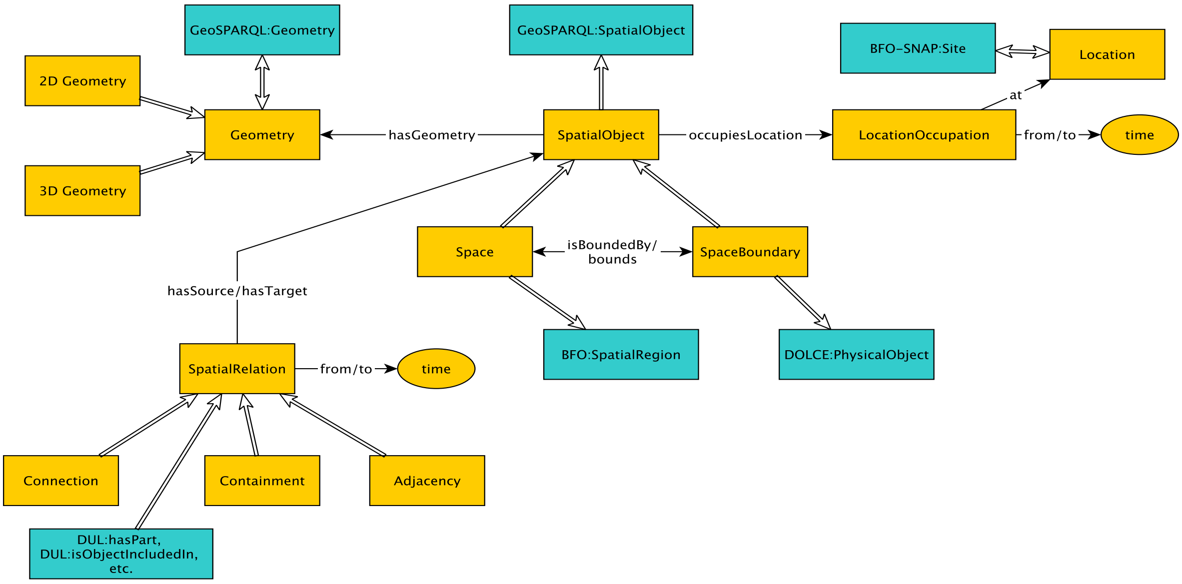


Fig. 13. Linking and Alignment to Upper Level Ontologies

describe basic physical objects as well as spatial parts and locations, although not at the level of granularity in description necessary for our work. The SGA pattern entity SGA:SpatialObject in particular can be aligned with BFO to identify entities in reality as well as associate items in our solution with temporal features. This is an important extension because buildings analyzed for energy and resilience results exist as they are for a certain amount of time - either for a lifespan or as of the effects of natural disasters making the temporal aspects useful to record. BFO-SNAP uses SpatialRegion for spaces, which can be aligned to our work, but our work allows a larger set of distinctions, including breakdown into 2D and 3D geometries. Also, we can align our class for describing locations of architectural models with BFO-SNAP Site.

Interactions between layers in the presented LDP require a bit of additional framework to make transitions effectively [Figure 12]. Building Element corresponds to SGA:spatialRepresentationObject [Figure 4] which is the pattern representation of any of the organization entities in the corresponding spatial hierarchy [Figure 3]. Those geometries seen in Figure 3 are Dolce physical-endurants and physical objects (Spatial Things), where the SGA provide additional relations and information that align to Dolce qualities for those geometries or other features. In terms of linking to upper ontologies, Building Element corresponds to an assembly (Wall, Floor, Ceiling, Roof, etc.) or a component (Door, Window, etc.) which captures the information that a taxonomy can, but is neither limited to a restrictive set of building elements nor separated from Linked Data compatibility benefits [Figure 13]. Our definition of Building Element is conflated so that it can be either an assembly or a component; this solved the problem seen in other approaches where assemblies are defined as a set but in reality they can be one or more components, and vice versa. Building Element aligns to IFCOwl using the classification "element." Functional_Structure and Architectural_Feature are modeled as roles so this solution is also extensible to engineering needs where it is necessary to figure out if an Assembly is Structural or Architectural. Building, BuildingAssembly, and BuildingComponent would all be SpatialObjects in GeoSPARQL, and the RCC8 relations are now accessible by using GeoSPARQL. For example, geo:sf-contains property connects building to site and we can reuse the Place Pattern from geolink [Figure 13]. Now, the top level is Object such that it can be aligned to patterns being developed concurrently to advance this approach.

There are even other data patterns, such as the Relative Relationship (RR) Pattern[50], that can be used to resolve the remaining discrepancies between different schemas, for example where we see models that have

[50]RR Pattern: https://curate.nd.edu/show/9k41zc79w4r

to be analyzed together but come from different Cartesian coordinate systems where reference points cannot be guaranteed to be the same. For instance, some solutions make assumptions such as the x-dimension in an architecturally rendered building is always corresponding to the height or width of a modeled geometry [23]. This article acknowledges that information such as that cannot be assumed and so our solution is more general, pulling out the information from the rendering space and then using the RR Pattern that can align them based upon how a model was actually drawn. Beyond what is capable by other works, the solution in this article also allows for integrating building and city level objects with OGC/W3C standards that are (soon to be) LD approaches for expressing this information in LD formats while maintaining ontological modularity for depicting architectural object hierarchies beyond what is capable by other solutions.

8.2. *Evaluation of the LDP, LD Views, and SGA Pattern Combination*

While the full architecture for this project is described in another paper, the condensed version is and the focus in this paper and includes the SGA, REST Communications used for data handling, and BIM interoperability through LD Views and ODP Structuring. For instance, compared to Apache Marmotta[51], which is an open data platform used to translate data across domains, the solution presented in this article is more general in terms of being able to handle any type GeoSpatial with options for connecting to other data stores depending on the simulation and type of analysis being performed, even though it is specifically implemented and demonstrated for BIM and resilience data. It can similarly handle GeoSPARQL queries, which should at least support the decision to carry this feature into the LDP presented above. Being more general in extensibility, modularity, and semantic capability, the LDP using LD Views and the SGA Pattern are also set up such that other issues can be tracked and resolved going forward, such as vulnerability, uncertainty, and risk analysis within modern practices [1].

The Google Flux (Google 2014) spin-off project FLUX[52] is another relevant project to discuss. This project does work with BIM data and saw the merits of working with that data in an accessible data platform. Also being concerned with data interoperability, this project took large steps forward in the domain of building tool functionality for design efforts but does not focus on LD. The approach using SGA pattern instances is not only compatible with the Semantic Web, but is able to look for additional data outside of the set of industry tools used for building design and this has the potential for major impacts on the data analysis relied on for design decisions. The SGA also makes it possible to capture and utilize any spatial information, as Views are added.

SEMERGY [33] is a related project with a similar set of goals. It does focus on design data interoperability using Semantic Web Technologies, which does open up a lot of potential in that direction; however, instead of the full capacity of semantics, it uses taxonomy-based building design optimizations. It is contested that taxonomies have a valuable place in research, but will not provide the multitude of benefits that ontology design patterns offer such as being able to capture the relationships that exists within BIM files. Without a rich set of relationships even between simple geometries, data analysis becomes far too limited as happens with taxonomies alone. RDF View extension files informing SGA pattern instances in a LDP means information can be queried more efficiently. For example, typical BIM file structures are tree-like in terms of how you search for information (i.e. component break-down); SGA instances mean that searching can happen in any direction and for any number of geometry groupings since it is a graph instead of a tree. When there are thousands of building components, multi-directional searching within the data can be vital for running simulations.

This is not intended to say that these are the only parts of a LDP that could be useful for BIM interoperability. It is true that getting BIM and resilience data into a common format that is compatible with the Semantic Web is no small endeavor, but it is also part of an even bigger goal. Further steps to this research would include evaluating the benefits of Reasoners, Decision Support, and recording Provenance [9] information from these SGA instances, and these steps are now a practical possibility, since our data is in a common LD format. Furthermore, it is important to note that because of the structure of this LDP and its compatibility with the Semantic Web, it is possible to add all of these features without needing remake any of the components that are already in place.

[51] Apache Marmotta: http://marmotta.apache.org/
[52] FLUX: https://flux.io/

8.3. Modularity in a Distributed LDP Framework

From the start of this project, the goal was to insist LDP components be designed to be modular in nature, to ensure the most reuse and ease platform growth over time. Modularity is a key factor for BIM and resilience data translations because it ensures that a different parser does not need to be coded for each type of data formatting that exists, which is impractical and not a sustainable approach. Furthermore, and perhaps more importantly, is that modularity is key for finding, using, and analyzing data from geographically distributed and remotely located data sets and models. Since all data cannot and should not exist in one location, modularity in the LDP approach also means that this project can be scalable as work progresses and that individual SGA instances can be stored for machine learning as they are generated at their own organized SPARQL endpoint(s).

8.4. Further SGA Research and Implementation

Additions to this work could be in several areas. The first area of work is planned for adding additional schema types to the set of existing ones available to the LDP. That is, additional Views will be added to the set of working ones, and others will follow as they are needed or as data formats are discovered. Due to the structure of the LDP and SGA, they will instantly be able to be consumed by the platform, just by having the additional View; thus, making feature expansion simple. Other potential areas of work that would be beneficial to test with this platform include specialized GIS software and data formats such as Point Clouds.

At this point, there are new questions that will need to be resolved as this research continues. Some of these areas would include additional external endpoint discovery, LDP components, user-generated Views for new data types, further Provenance information layers, etc. There is also more research that could be done in terms of ways to use the SGA instances in external tools, or rather how to extract the relevant information from a particular SGA instance to be useful in said tools. Nevertheless, the LDP presented makes it possible to begin to translate between select proprietary formats. These new areas of work are all relevant to the future of the use of the SGA in the LDP, but are outside of the current scope of this article.

9. Conclusion

The multi-disciplinary questions being asked of energy, design, and engineering professionals currently are requiring increasing amounts of multi-disciplinary sets of data in order to provide accurate and complete answers. This article presented a solution to this challenge using ontology-based Semantic Graphs for automating spatial data integration. From high-level views of this semantically mapped spatial information, knowledge is accumulated in order to automate queries and then generate data-based conclusions utilizing many types of standard data. Currently, both this type of approach and diversity of data sets included for analysis has previously not been possible, especially implemented with Linked Data principles for the Semantic Web.

In this article we presented an extension to an existing framework that was intended to use Linked Data Views to bring Semantic Web compatible interoperability to BIM and resilience data. The Spatial Graph Adapter Pattern along with its axioms, platform implementation, and use case scenarios provide the necessary connections within the platform that allow the other two existing areas of research to be functionally joined together, making a great stride forward toward the vision for completely interoperable data and industry applications. It is achieved in a manner that is already interchangeable with the Semantic Web and thus can harness the benefits therein, unlike other existing projects.

Per domain-specific research needs, this work flow has been useful and implemented for IFC, GBXML, and CityGML standards through the use of Linked Data Views that are used to map each of these formats to the SGA pattern automatically. There will be opportunities to utilize the SGA instances in the future as they can be used in cooperation with other ODPs to record provenance information from schema versions over time as well as connect this data to the larger Linked Data Cloud; this is expected to be a contributing factor in improving learning and knowledge stores as time progresses. Modularity and scalability were key factors driving the outcome of the current project as well as the now having the ability to generate automated Semantic Graph instances at separate SPARQL endpoints for an easier building block framework. This framework embedded in a linked data platform is ideal for storing, querying, and analyzing BIM and resilience data.

It is now possible to strategically add additional BIM schemas and or other forms of spatial data such as GIS software or Point Cloud representations of spatial collections using LD Views to automatically generate SGA-structured Semantic Graphs as needed. Additional future work will involve further alignment of the SGA pattern to spatial descriptions outside of the building domain schemas so that the SGA can be used for spatial constructs of any kind (risk, chemistry, etc.). The Spatial Graph Adapter Pattern is a scalable, modular, extensible approach to schema interoperability and translation methods currently used to bring BIM into a common and open triple format for further use as Linked Data within the Semantic Web and other modeling and simulation applications.

Acknowledgments. Thank you to the Center for Research Computing at the University of Notre Dame, NSF CMMI-1537652: "A Green Resilience Framework to Support the Design of Sustainable Buildings Under Multiple Hazards," and Civil Engineering domain experts Tracy Kijewski-Correa and Alexandros Taflanidis as well as our Architecture domain expert Aimee Buccellato.

References

[1] B. M. Ayyub. Vulnerability, Uncertainty, and Risk: Analysis, Modeling, and Management. In *First International Symposium on Uncertainty Modeling and Analysis and Management (ICVRAM 2011); and Fifth International Symposium on Uncertainty Modeling and Anaylsis (ISUMA)*, Reston, VA: American Society of Civil Engineers, 2011. Reston, VA: American Society of Civil Engineers.

[2] M. Bhatt, J. Hois, and O. Kutz. Ontological modelling of form and function for architectural design. *Applied Ontology*, 7(3):233–267, July 2012.

[3] F. Biljecki. Random3dcity: Synthetic CityGML data and procedural modelling engine.

[4] T. Bittner, M. Donnelly, and S. Winter. Ontology and semantic interoperability. In *Large-Scale 3D Data Integration: Challenges and Opportunities, CRCPress, Boca*, volume 139-160.

[5] S. Borgo and C. Masolo. *Ontological Foundations of DOLCE*. Theory and Applications of Ontology: Computer Applications. Springer Link, Springer Netherlands, 2010.

[6] G. Boyes, C. Thomson, and C. Ellul. Integrating BIM and GIS : Exploring the use of IFC space objects and boundaries. In *GISRUK 2015*.

[7] B. Brodaric and T. Hahmann. Toward a foundational hydro ontology for water data interoperability. In *HIC 2014*.

[8] G. P. Cimellaro, A. M. Reinhorn, and M. Bruneau. Framework for analytical quantification of disaster resilience. 32(11):3639–3649, 2010.

[9] M. Compton, D. Corsar, and K. Taylor. Sensor Data Provenance: SSNO and PROV-O Together At Last. In *SSN 2014*, Trentino, Italy, 2014. ISWC.

[10] O. Corby, C. Faron-Zucker, and F. Gandon. A Generic RDF Transformation Software and Its Application to an Online Translation Service for Common Languages of Linked Data. In *Lecture Notes in Computer Science*, volume 9367, pages 150–165. Springer, 2015.

[11] F. Diniz. Composition of semantically enabled geospatial web services.

[12] S. Donkers, H. Ledoux, J. Zhao, and J. Stoter. Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings. 20(4):547–569.

[13] M. Dumontier, C. Baker, et al. The semanticscience integrated ontology (SIO) for biomedical research and knowledge discovery. 5(1):14, 2014.

[14] E. Eaton, C. P. Gomes, B. Williams, and et al. Vol 35, No 2, Summer Issue. *AI Magazine*, 35(2), 2014.

[15] C. Esteban, V. Tresp, et al. Predicting the co-evolution of event and knowledge graphs [1512.06900]. *arXiv.org - Computer Science - Learning*, 2015.

[16] H. Ferguson et al. Capturing an architectural knowledge base utilizing rules engine integration for energy and environmental simulations. In *SimAud 2015*, pages 17–24. Society for Modeling & Simulation International (SCS).

[17] H. Ferguson, C. Vardeman II, and J. Nabrzyski. Linked Data Platform for Building Resilience Based Applications and Connecting API Access Points with Data Discovery Techniques. In *IEEE Workshop on Big Data Metadata and Management (BDMM '2016) in conjunction with the 2016 IEEE International Conference on Big Data*, Washington D.C., USA, Nov. 2016. IEEE Conference Publications and Sponsored by IEEE Big Data Initiative (BDI).

[18] H. Ferguson, C. Vardeman II, and J. Nabrzyski. Linked Data View Methodology and Application to BIM Alignment and Interoperability. In *1st IEEE International Workshop on Big Spatial Data in conjunction with IEEE International Conference on Big Data*, Washington D.C., USA, Nov. 2016. IEEE Conference Publications and Sponsored by IEEE Big Data Initiative (BDI).

[19] M. Flint, L. Dhulipala, Y. Shahtaheri, and et al. Developing a Decision Framework for Multi-Hazard Design of Resilient, Sustainable Buildings. In *Iconhic 2016 | International Conference on Natural Hazards and Infrastructure*, Chania, Greece, 2016.

[20] P. Grenon and B. Smith. SNAP and SPAN: Towards Dynamic Spatial Ontology. *Spatial Cognition and Computation*, 4-1:69–103, 2004.

[21] G. GrÃűger and L. PlÃijmer. CityGML âĂŞ interoperable semantic 3d city models. 71:12–33.

[22] T. Hahmann, B. Brodaric, and M. Gruninger. Voids and material constitution across physical granularities. In *IOS Press*. FOIS-2014.

[23] J. Hois. *Modular Ontologies for Spatial Information*. Dissertation, Berlin, 2013.

[24] J. Hois, M. Bhatt, and O. Kutz. Modular Ontologies for Architectural Design. In *Proceedings of the 2009 conference on Formal Ontologies Meet Industry*, pages 66–77, The Netherlands, 2009. IOS Press Amsterdam.

[25] A. Ishizaka and P. Nemery. *Multi-criteria Decision Analysis: Methods and Software*. John Wiley & Sons, Ltd., 1 edition

edition.

[26] Ittingen, Kartause. *Spatial Information Theory : Foundations of Geographic Information Science*. LNCS ; 2825. Springer: International Conference, COSIT 2003.

[27] K. Janowicz, P. Hitzler, B. Adams, D. Kolas, and C. Vardeman II. Five stars of linked data vocabulary use. *Semantic Web Journal*, 2014.

[28] K. Janowicz, A. Krisnadhi, Y. Hu, and et al. A Minimal Ontology Pattern for Life Cycle Assessment Data. In *6th Workshop on Ontology and Semantic Web Patterns (WOP 2015)*, PA, USA 2015, 2015.

[29] K. Kaneiwa, M. Iwazume, and K. Fukuda. An upper ontology for event classifications and relations. In *AI 2007: Advances in Artificial Intelligence*, volume 4830 of *LNCS*, pages 394–403. Springer Link.

[30] T. Kijewski-Correa, N. Smith, A. Taflanidis, A. Kennedy, C. Liu, M. Krusche, and C. Vardeman II. CyberEye: Development of integrated cyber-infrastructure to support rapid hurricane risk assessment. *Journal of Wind Engineering and Industrial Aerodynamics*, 133:211–224, Oct. 2014.

[31] A. A. Krisnadhi, P. Hitzler, and K. Janowicz. On the capabilities and limitations of OWL regarding typecasting and ontology design pattern views. In *Ontology Engineering - 12th International Experiences and Directions Workshop on OWL, OWLED, co-located with ISWC 2015*, LNCS 9557, pages 105–116. Springer, 2015.

[32] P. Pauwels, D. Van Deursen, R. Verstraeten, J. De Roo, R. De Meyer, R. Van de Walle, and J. Van Campenhout. A semantic rule checking environment for building performance checking. *Elsevier*, 20(5):506–518, 2011.

[33] U. Pont, J., N. Ghiassi, S. Fenz, J. Heurix, et al. Semergy: Application of semantic web technologies in performance-guided building design optimization. 107âĂŞ20.

[34] B. Regalia, K. Janowicz, and S. Gao. VOLT: A provenance-producing, transparent SPARQL proxy for the on-demand computation of linked data and its application to spatiotemporally dependent data. In *The Semantic Web*, pages 523–38. Springer.

[35] D. Ventura, R. Verborgh, V. Catania, and E. Mannens. Autonomous Composition and Execution of REST APIs for Smart Sensors. In *Joint Proceedings of the 1st Joint International Workshop on Semantic Sensor Networks and Terra Cognita and the 4th International Workshop on Ordering and Reasoning*, 2015.

[36] B. Wu. Development of MIT geospatial data to CityGML workflows.

[37] S. Zhang and O. Bodenreider. The body region connection calculus RCC-8, 2005.

Appendix

A. Spatial Graph Adapter Pattern Axioms

SpatialInformationObject represents exactly one SpatialObject, where a SpatialObject can align to GeoSpatialObject from GeoSPARQL. The SpatialObject is the actual spatial entity and the SpatialInformationObject is the data object representing that actual spatial entity, to clearly define the difference between spatial things and representations of those spatial things.

$$SpatialInformationObject \sqsubseteq (=1\ represents.SpatialObject) \tag{1}$$

$$\exists represents.SpatialObject \sqsubseteq SpatialInformationObject \tag{2}$$

$$SpatialInformationObject \sqsubseteq \forall represents.SpatialObject \tag{3}$$

A SpatialInformationObject may have Attribute and may have ContextualizedRelation. Since Attributes and ContextualizedRelations have different descriptions in different data schema, these model classes were also chosen to work for any of the schemas used by our domain tools. So we simply state the domain and range for the corresponding properties.

$$\exists hasAttribute.Attribute \sqsubseteq SpatialInformationObject \tag{4}$$

$$SpatialInformationObject \sqsubseteq \forall hasAttribute.Attribute \tag{5}$$

$$\exists hasContextRelation.ContextualizedRelation \sqsubseteq SpatialInformationObject \tag{6}$$

$$SpatialInformationObject \sqsubseteq \forall hasContextRelation.ContextualizedRelation \tag{7}$$

Types and Values are needed because as seen Figure 9, the semantics for the same types of spatial objects are labeled differently schema to schema. Attribute must have exactly one Value and exactly one Type. It must be attached to exactly one SpatialInformationObject. Note that it is possible for two different SpatialInformationObjects to have attributes of the same Type and Value.

$$Attribute \sqsubseteq (=1\ hasValue.Value) \sqcap (=1\ hasType.Type) \tag{8}$$

$$Attribute \sqsubseteq (=1\ hasAttribute^{-}.SpatialInformationObject) \tag{9}$$

Our use of hasType makes the model more amenable to resolving the large variation of schema differences that exist; instead of subclassing we consider types to be instances of individuals so that we are not burdened by the decision to say that something is a subtype or not. Doing so would cause more hierarchical confusion instead of help in terms of its application. Nevertheless, having typing through properties does not preclude us from adding in the function of such a hierarchy later on [31]. ContextualizedRelation must have exactly one Value and exactly one Type. ContextualizedRelation refers to at least one SpatialInformationObject. The refersTo relation is an inverse of hasContextRelation.

$$ContextualizedRelation \sqsubseteq (=1\ hasValue.Value) \sqcap (=1\ hasType.Type) \tag{10}$$

$$ContextualizedRelation \sqsubseteq \exists refersTo.SpatialInformationObject \tag{11}$$

$$refersTo \equiv hasContextRelation^{-} \tag{12}$$

Domain and range restrictions for hasValue and hasType:

$$\exists hasValue.Value \sqsubseteq Attribute \sqcup ContextualizedRelation \tag{13}$$

$$\exists hasType.Type \sqsubseteq Attribute \sqcup ContextualizedRelation \tag{14}$$

$$Attribute \sqsubseteq \forall hasValue.Value \sqcap \forall hasType.Type \tag{15}$$

$$ContextualizedRelation \sqsubseteq \forall hasValue.Value \sqcap \forall hasType.Type \tag{16}$$

Pairwise class disjointness:

$$\mathsf{alldisjoint}(SpatialObject, SpatialInformationObject, Attribute, ContextualizedRelation, Value, Type) \tag{17}$$