CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Leveraging rules on integrated 3D semantic city models for validating scenarios of urban evolution

Diego Vinasco-Alvarez[1,2] ![ORCID], John Samuel[3] ![ORCID], Sylvie Servigne[1] and Gilles Gesquière[2] ![ORCID]

[1]LIRIS, UMR 5205, INSA Lyon, CNRS, Université Claude Bernard Lyon 1, Villeurbanne, France
[2]LIRIS, UMR 5205, Université Lumière Lyon 2, Universite Claude Bernard Lyon 1, Lyon, France
[3]LIRIS, UMR 5205, CPE Lyon, CNRS, INSA Lyon, Université Claude Bernard Lyon 1, Villeurbanne, France
Corresponding author: Diego Vinasco-Alvarez; Email: diego.vinasco-alvarez@cnrs.fr

Abstract

The process to better understand the intricate evolution of our urban territories requires combining urban data from different or concurrent instances of time to provide stakeholders with more complete views of possible evolutions of a city. Geospatial rules have been proposed in the past to validate 3D semantic city models, however, there is a lack of research in the validation of multiple, *concurrent* and *successive*, scenarios of urban evolution. Using Semantic Web Ontologies and logical rules, we present a novel standards-based methodology for validating integrated city models. Using this methodology, we propose interoperable rules for validating integrated open 3D city snapshots used for representing multiple scenarios of evolution. We also implement a reproducible proof of concept test suite for applying the proposed rules. To illustrate how these contributions can be used in a real-world data validation use-case, we also provide example queries on the validated data. These queries are specifically used to construct a 3D web application for visualizing and analysing urban changes across multiple scenarios of evolution of a selected zone of interest.

Impact Statement

This research introduces a novel standards-based methodology for validating integrated 3D semantic city model snapshots that represent concurrent and successive urban evolution scenarios. By leveraging Semantic Web Ontologies and logical rules, the study enables the temporal validation of these urban evolution scenarios. A reproducible proof-of-concept test suite and example queries demonstrate practical applications, including constructing a 3D web application to visualize and analyse urban changes. This innovative approach empowers urban planners, policymakers, and researchers with robust tools to model and understand complex urban transformations, facilitating data-driven decision-making. The methodology's emphasis on interoperability and real-world applicability positions it as a significant contribution to advancing urban evolution analysis.

1. Introduction

In recent years, 3D semantic city models have become more frequently adopted as tools to understand long-term changes in urban landscapes. Here, temporal information can be used to represent the historical

---

This research article was awarded Open Data and Open Materials badges for transparent practices. See the Data Availability Statement for details.

evolution of a city as well as its imagined future. This information could be related to changes in the structure, geometry, or semantic information used to describe city snapshots across different instances of time. In addition, these changes could be analysed at different scales, from the changes of individual urban objects over time to the development of entire districts and cities (Chaturvedi and Kolbe, 2019).

These 4D (3D + Time) city models may also be extended to represent concurrent or competing representations of city evolution. For instance, historians may discuss different hypotheses of how cities evolved in their studies to understand city urbanization (Kutzner et al., 2020; Samuel et al., 2020). Similarly, urban planners may propose new buildings and districts in existing zones or territories when designing new construction projects (Chaturvedi and Kolbe, 2019; Nguyen and Kolbe, 2021). To this end, the study of urban evolution and the associated hypotheses related to changes in urban objects or features requires the complex integration of data from multiple sources. This process of combining data from different sources *requires data validation* to ensure the data can be holistically exploited to provide users with more complete views of city evolution.

The major contribution of this article is a proposed novel standards-based methodology for validating integrated 4D urban data as concurrent scenarios of urban evolution (Section 3). Additionally, we present the following contributions created thanks to this methodology:

- **Formalization of rules** for a standardized 4D urban data conceptual model for representing concurrent viewpoints of urban changes (Section 3.3).
- **Logical validation** of urban scenarios of evolution (Section 3.4) using a reproducible proof-of-concept test suite.
- **Formulation of queries and data views** for retrieving concurrent points of view of evolving cities using the proposed data model and generated data (Section 4).

The article is presented as follows: Section 2 details related works in modeling, integrating and logically validating 4D urban data; Section 3 details the aforementioned methodology for reusing standards in 4D urban data integration and eventual data validation; Section 4 presents the results of integrating and validating standardized open urban data snapshots of the Gratte-Ciel neighborhood of Lyon, France with this methodology; Finally, Section 5 offers our perspective future works and concludes the article.

2. Related works

The evolving 3D urban data used to create 4D city models is frequently complex. In this context, a process of combining data from multiple sources (i.e., *data integration* Tran et al. (2016)) can be used to construct these 4D city models and provide users with more complete views of city evolution. This section will discuss existing approaches for integrating these data at the data instance level, existing approaches for representing city evolution (both real and imagined), and previous works in logical 4D data validation. This section also positions the contributions of this article among these works.

2.1. Representing urban data evolution

Many approaches have been proposed for portraying different digital representations of the same real-world city object (over time). In geospatial data, distinctions between these representations are often *identity-based*, i.e., where identity is the characteristic that distinguishes one object from another over time, and where a change to the object may or may not change its identity (Hornsby and Egenhofer, 2000; Stefani et al., 2008; Van Ruymbeke et al., 2015).

In this context, city object changes are often either semantic and/or geometric. Semantic changes are often straightforward to identify, but may be challenging to categorize depending on the user's perspective, application, or use-case (Nguyen and Kolbe, 2021). Certain geometric or spatial changes may also complicate the definition of an object's identity, such as the splitting of buildings into two or more buildings or the fusion of multiple distinct buildings (Stefani et al., 2008). Furthermore, one major requirement for identity-based changes is to ensure the use of unique identifiers for a given city object.

This is not guaranteed as city object identifiers may be different between independent data providers, may be created through the use of different data capture techniques (e.g., by using different LIDAR, arial photography, and/or photogrammmery technologies). Even the same data provider may use different identifiers for the same urban feature over time due to various administrative or technical reasons (Nguyen and Kolbe, 2021).

Several works have been proposed to extend these representations to facilitate the comparison of different historical or future evolutions of urban spaces (Pfeiffer et al., 2013; Chaturvedi et al., 2017). Notably, the versioning module in CityGML 3.0 (Kutzner et al., 2020) has been proposed to provide a standard for representing multiple versions of a city object and, at a larger scale, multiple versions of a city or district. Extensions to these standards have been proposed that permit the comparison of alternate or concurrent scenarios of urban evolution, such as the Workspace model proposed in Samuel et al. (2020); Vinasco-Alvarez et al. (2024f). The specific concepts used in this article are discussed in detail in Section 3.1
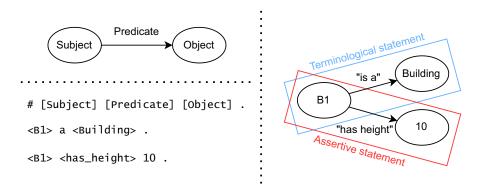
2.2. *4D urban data integration*

In the domain of 4D data integration, approaches may propose applying *feature matching* or *entity resolution techniques* that compare the spatial and/or semantic characteristics of city objects to determine when two digital representations are modeling the same real-world object (Sehgal et al., 2006; Pédrinis et al., 2015; Bernard, 2019; Balsebre et al., 2022). These characteristics may include a representation's geospatial coordinates (in 2D or 3D) and/or non-spatial attributes such as unstructured text, identifiers, administrative information, and so forth By incorporating the temporal dimension into their proposed 3D city models, these approaches can provide a continuous view of urban evolution instead of isolated, interrupted snapshots.

These techniques can be applied to stitch together static digital city models from different points in time to provide digital views of how their real-world counterparts have changed over time (Nguyen et al., 2017; Jaillot et al., 2020). These approaches can also be used to integrate data based on frequently used 3D urban data standards such as Industry Foundation Classes (IFC) and CityGML (buildingSMART, 2024; Kutzner et al., 2020; Diakite and Zlatanova, 2020; Liu et al., 2017; Liu et al., 2021; Hijazi et al., 2020).

2.3. *Semantic Web technologies for urban data integration*

New research works have turned to Semantic Web data standards such as the World Wide Web Consortium's (W3C) Semantic Web technology suite (Lassila et al., 2001; W3C, 2025) have been adopted to facilitate urban data modeling and integration on the web. Here, expressive and formal modeling languages are proposed, such as Ontology Web Language (OWL) family of languages that can be used to create models that permit the description of complex data models while being machine-readable through flexible graph data formats such as Resource Description Framework (RDF) (Malinverni et al., 2020). RDF permits the definition of knowledge graphs through triple statements that are composed of a subject, predicate, and object (Figure 1). OWL builds on this format to permit the creation of ontological knowledge graphs that are sufficiently formal to be considered logical theories and can be used to logically infer new information (Uschold and Gruninger, 2004). These ontologies can store the concepts, relationships, and constraints of a data model (i.e., the collection of terminological knowledge or *TBox* of the ontology) as well as information regarding the data instances of these concepts (i.e., the collection of assertions or *ABox* of the ontology) (Figure 1). Several data transformation approaches have been proposed to integrate 2D, 3D, and 4D urban data using these data formats (Kyzirakos et al., 2018; Tran et al., 2020; Chadzynski et al., 2021; Hor and Sohn, 2021; Usmani et al., 2021; Vinasco-Alvarez et al., 2021).

Dynamic knowledge graphs (Pujara and Getoor, 2014; Trivedi et al., 2017; Aparicio et al., 2024) have been explored to understand the evolution of knowledge graphs over time and to integrate data from different heterogeneous sources with different levels of data accuracy and completeness. In case of urban evolution studies, urban digital twins (Grieves, 2014; Batty, 2018) have been explored and implemented

Figure 1. *An illustration of the RDF triple structure (top left). An example of RDF triples in the Turtle syntax (bottom left). An illustration of the RDF triple example (right).*

as a method for integrating and maintaining spatio-temporal urban data. These approaches propose maintaining a digital representation (or "twin") of an evolving urban environment up to date through a "twinning" processes. This process often involves applying changes to the digital twin—either periodically or continuously—with real-world data from the physical twin. Furthermore, dynamic knowledge graphs such as The World Avatar (TWA) have been implemented as approaches to integrate cross-domain digital twins by levraging Semantic Web technologies for interoperability (Eibeck et al., 2019; Akroyd et al., 2021; Chadzynski et al., 2021). In this domain, the digital twin may be refered to as the 'base world' and the twinning process is effectuated through a set of software agents that can update, retrieve information from, and restructure the "base world." Some dynamic knowledge graphs also permit the modeling of hypothetical scenarios, such as the Parallel World Framework (Eibeck et al., 2020), which proposes storing secondary "parallel base-world for simulating changes to the primary base-world."

While these approaches have been proven to be effective for integrating heterogeneous urban data, the validation of this integration remains necessary and complex for ensuring a high-quality representation of the urban environment. Here, the formal nature of languages such as OWL lends themselves to validation approaches using logical rules.

2.4. Logical consistency validation of 4D urban data

Many validation approaches exist for urban data, each implemented to identify different aspects of data quality in 4D urban data applications. For example, 4D geospatial data applications using Semantic Web technologies may use data model constraints and rules formalized in languages such as OWL and SWRL (Semantic Web Rule Language) to verify that their integrated data is logically consistent (referred to as consistency checking) (Tran et al., 2016; Batsakis et al., 2017; Samuel et al., 2020). This verification step is especially important in applications that propose the automatic and logical inference of new information from existing data using reasoners (Tran et al., 2020) or spatio-temporal analysis of territorial events (Batsakis and Petrakis, 2011; Harbelot, 2015; Tran et al., 2016; Bernard, 2019).

In the case of SWRL, its language is based on several sublanguages from OWL and Rule Markup Language that permit Horn-like rules to be written in OWL ontologies (Horrocks et al., 2004). These rules are composed of an antecedent (or the rule body) and a consequent (or the rule head). In SWRL, the antecedent and the consequent are sets of atoms such that if the conjunction of all the atoms of the antecedent holds true, then the conjunction of all the atoms of the consequent must hold true.

$$antecedent \rightarrow consequent \tag{1}$$

For example, the rule *if a is a building, it must have a height greater than 0*, can be written as:

```
building(?a) -> height(?a,?h), greaterThan(?h, 0).
```

Inferencing can be performed by using reasoners such as Pellet, HermiT, or Fact++ (Tsarkov and Horrocks, 2006; Sirin et al., 2007; Glimm et al., 2014), through queries using the SPARQL query language (Buil Aranda et al., 2013; Tran et al., 2016), or with tools that can parse SWRL rules with SQWRL, such as the SWRLTab plugin in the Protégé ontology editor or OwlReady2 (O'Connor and Das, 2010; Musen, 2015; Lamy, 2017). In the context of integrating open data conformant to international standards, approaches that rely on purely *syntactic* transformations—i.e., conversion of data format as opposed to semantics (Kutzner, 2016)—may implement validation steps to ensure that the integrated data rests conformant to the initial standard and thus internationally interoperable.

For example, Chadzynski et al. (2021) proposes a similar validation to preparing the CityGML 2.0 ontology proposed by the University of Geneva in Zalamea et al. (2016) for integration in TWA that illustrates the importance of this validation. This ontology was created using XSLT transformations of the CityGML 2.0 XML Schema and evaluated according to several Semantic Web metrics proposed in Hlomani and Stacey (2014). It was determined that the ontology was inconsistent (the ontology contained logical contradictions), insufficiently concise (the ontology contains irrelevant or redundant semantic elements concerning a domain of knowledge), and incomplete (the ontology is missing appropriate semantic elements to sufficiently answer questions in a domain of knowledge) in the context of their use-case. The ontology was thus manually corrected to enable reasoning over it, simplified by removing unnecessary classes and properties leftover from the XML Schema in OWL, and extended with the help of domain experts as necessary to cover missing concepts required by the use-case.
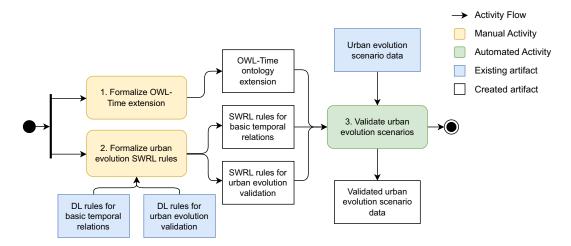
2.5. Summary

Taking these approaches into consideration illustrates how complex the 4D urban data integration process can be for providing users with more complete views of the urban landscape and its evolution. It also underlines the importance of data validation during these processes. However, while standards such as CityGML 3.0 do provide general constraints and abstract normative tests (Kolbe et al., 2021, Annex A), they leave the interpretation and implementation of these constraints and tests to the users of the standard. Additionally, as standards themselves evolve over time, the proposed rules for testing the conformance of data to the standard may also require periodic updates to work with the latest versions of the standard. Therefore, manual intervention is required to translate abstract tests, constraints, and rules to more machine-readable languages and formats.

Furthermore, research in the domain of concurrent scenarios of urban evolution is relatively new. As far as we know, there is currently a lack of existing works in the domain of validation of standardized data used to create these urban scenarios of evolution. Building on these proposed works, the next section presents our methodology for validating integrated heterogeneous data in applications that require temporal and analysis of urban phenomena.

3. Methodology

The methodology of our approach is decomposed into 3 main activities (Figure 2) as detailed in this section: formalize an extension for OWL-Time, formalize rules for validating urban evolution, and use the results of these activities to validate urban evolution scenarios. To validate whether a proposed urban evolution scenario is temporally logical, we first propose an extension of the OWL-Time model, presented in Section 3.2 for inferring temporal relations between temporal entities (such as *instants* and *intervals*). This extension is based on the temporal relations originally defined in Harbelot (2015) for temporal intervals. Subsequently, we adapt the rules proposed in Samuel et al. (2020) to infer when scenarios of urban evolution and their versioned entities are temporally valid. These rules, originally formalized in a description logic (DL) language, are translated to SWRL (presented in Section 3.3) following the practices described in Vinasco-Alvarez (2023, Chapter 5). Additionally, as the CityGML standard has evolved since the original proposal of these DL-rules, they are updated to conform to the newer CityGML 3.0 Conceptual model.

Figure 2. *UML activity diagram of proposed methodology. External artifacts or data sources are located on the bottom of the diagram while artifacts created from an activity are placed in the center.*

Finally, these contributions are used to validate existing scenarios of urban evolution with a reproducible proof-of-concept test suite presented in Section 3.4. Unlike the first two activities, which manually produce artifacts in Semantic Web formats, this final activity takes advantage of this fact to fully automate the data validation process.

3.1. Preliminaries for representing scenarios of urban evolution

As discussed in Section 2.1, many works define concepts for representing urban evolution. This section presents the concepts used in this article for this purpose, as defined by CityGML 3.0 and the Workspace model. *Feature with a lifespan* (or a "versioned" feature): a representation of a real-world phenomenon with a specific existence time. *Version*: a set of versioned features such that the existence time of the version occurs during the existence time of all of its versioned feature members (i.e., the interval of time when the feature members are *stable*, i.e., it does not undergo any modification). *Version Transition*: a set of *Transactions* or individual changes between the corresponding versioned features of two consecutive Versions (e.g., whether a feature is added, modified, or removed from the Version); the existence time of a Version Transition occurs between the existence time of its corresponding Versions. In addition, a string of sequential alternating versions and version transitions can be composed into a *Scenario* for representing a specific evolution of the city, hypothetical or otherwise. These scenarios exist either within either a *Consensus Space or Proposition Space*—where concensus space contains the (official or generally accepted) scenario (and all the versions and transitions) that represent the agreed upon real-world evolution of a city (or district), and proposition space contains all the scenarios that include hypothetical or imagined evolutions of the city. Finally, a *Workspace* is composed of a single proposition and consensus space and can be used to store the relevant representations of city evolution for a specific application or community of users. These concepts are exemplified in Figure 3.

Furthermore, two major temporal intervals or instants are often used for representing the evolution of cities and city objects:

- **Existence time** (or validity time) for representing when urban data and urban objects exist or are relevant in the real world
- **Transaction time** for recording when the data is stored, updated, or deleted from a data source (e.g., a file or database)
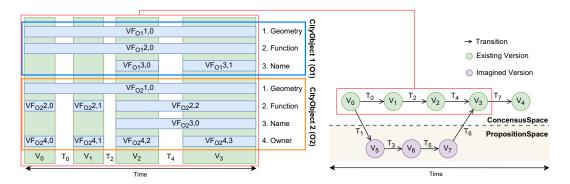
Figure 3. *Illustration of a scenario of evolution containing two city objects (O1, O2) and their properties (P₁–P₄) that change between versions (V₀–V₇) or "snapshots" of an urban area (shown above). These scenarios are also composed of Transitions (T₀–T₇) that contain the changes between each successive Version. These scenarios can intersect one of 2 abstract spaces (shown below): a consensus space containing an agreed upon real-world representation of urban evolution, and a proposition space containing 0 or more hypothetical or unverified scenarios of evolution (Samuel et al., [2020]).*

These are modeled with start and end timestamps, indicating the beginning and completion of said interval or instant (where in a temporal instant, the beginning and end are concurrent). Approaches, such as those that follow the European INSPIRE model (INSPIRE Drafting Team "Data Specifications", 2014), support both of these representations, including the CityGML standard.

The rest of this article uses the namespace prefixes available in Table A1 of the Appendix section when referring to these concepts in the context of existing or proposed RDF/OWL identifiers. E.g., `time:Instant` references a temporal instant as defined in the OWL-Time ontology.

3.2. *Extending OWL-Time for validating urban evolution scenarios*

As a part of the first activity of our methodology (2), we propose an alternative interpretation of an existing extension of OWL-Time for modeling relationships between temporal intervals and instants (Harbelot, 2015) (Figure 4). We use OWL-Time, since it is a recommended W3C standard for representing temporal relations and is a widely accepted temporal model for representing the basic temporal relations proposed in Allen (1984). As detailed in (Vinasco-Alvarez, 2023, chapter 5), our proposal consists of 3 definitions of temporal instant-interval relations (`time_ext:in`, `time_ext:during`, `time_ext:fin-ishes`) and 1 relation between instants (`time_ext:equals`). In addition to the temporal relations, we propose two object properties as subproperties of `time:hasTime` for distinguishing between the existence time (`time_ext:hasExistenceTime`) and transaction time (`time_ext:hasTransactionTime`) of versionned urban entities.

To simplify the existing definition of these relations, we use a *left-closed, right half-open* interpretation of temporal intervals, whereas the original extension proposes a purely closed interpretation. For example, given two temporal intervals T_1 and T_2, where T_1 defined as $T_1 = [t_{b1}, t_{e1}), T_2 = [t_{b2}, t_{e2})$ where $t_{b1}, t_{e1}, t_{b2},$ and t_{e2} are temporal instants, the temporal rule *meets* can be defined as the following DL statement:

$$
\begin{aligned}
Interval(T_1) \wedge Interval(T_2) \wedge \\
hasEnd(T_1, t_{e1}) \wedge has\,Beginning(T_2, t_{b2}) \wedge \\
equal(t_{e1}, t_{b2}) \\
\rightarrow meets(T_1, T_2)
\end{aligned}
\tag{2}
$$

This reduces the number of extended temporal relations from 4 to 3 and affects when instants and intervals *meet* and also permits a *continuous* representation of time. Under the previous interpretation, a

Relation Name		Instant-Instant Relations	Instant-Interval Relations	Interval-Interval Relations
A disjoint B B disjoint A	A before B B after A	○	○	
	A meets B B metBy A			
	A overlaps B B overlappedBy A			
A in B	A finishes B B finishedBy A			
	A during B B contains A			
	A starts B B startedBy A			
	A equals B B equals A			

Figure 4. *A synthesis from (Vinasco-Alvarez, 2023) of the temporal relations as defined in OWL-Time according to Allen (1984) with our alternative interpretation of the Instant-Interval relations proposed in Harbelot (2015) (highlighted in orange).*

discrete interpretation of time is required where $meets(T_1, T_2)$ is true if $t_{e1} + 1 = t_{b2}$ as defined in Samuel et al. (2020). Also, unlike Harbelot (2015) we use the vocabulary from OWL-Time to refer to temporal instants instead of points.

These relations are first formalized as object properties in OWL with constraints on their domain and range. For example, the instant-interval relation, `time_ext:in`, is defined as follows in the RDF Turtle syntax:

```
time_ext:in    a owl:ObjectProperty;
  rdfs:domain    time:Instant;
  rdfs:range     time:ProperInterval;
  rdfs:comment   "If an instant T1 is in a proper interval …
```

The next section illustrates how we translate the DL representation of these rules to SWRL.

3.3. Rules for the temporal validation of urban evolution scenarios

Within the second methodology activity (2), we propose a translation of the DL rules from Samuel et al. (2020) to SWRL for representing concurrent scenarios of urban evolution. These rules can be divided into two subsets. The first ruleset is a set of basic rules for validating temporal entities as temporally consistent, largely based on the rules described in Allen (1984); Harbelot (2015) and is closely interoperable with OWL-Time. The second ruleset can be used specifically for validating urban evolution, as it is an extension of the CityGML 2.0 model rules for structuring versioning and Workspace ontologies. Figure 5 illustrates the proposed rule translation process in three steps.

First, the class and property definitions proposed in Samuel et al. (2020) are updated to reflect the recent changes to CityGML between versions 2.0 and 3.0, such as the adoption of the Versioning extension (Chaturvedi and Kolbe, 2019) as an official CityGML module. Additionally, Vinasco-Alvarez et al. (2024f) proposes the "CityOWL" ontology network with OWL definitions for CityGML 3.0 and Workspace class hierarchies, and class and role restrictions. By reusing these ontologies, these
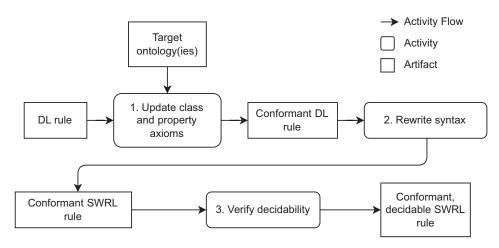
Figure 5. *3-step DL to SWRL rule translation process. In our context, the process is applied to the DL rules proposed in Samuel et al. (2020) and the target ontology is composed of the "CityOWL" ontology network Vinasco-Alvarez et al. (2024f) and the proposed OWL-Time extension (Section 3.2).*

declarations can be ignored in the translation of the original DL ruleset. For example, the Turtle definition of a Version is as follows (Vinasco-Alvarez et al., 2022):

```
vers:Version a owl:Class;
   rdfs:label "Version"@en;
   rdfs:subClassOf.
     [ a owl:Restriction;
         owl:allValuesFrom vers:ADEOfVersion;
         owl:onProperty vers:Version.adeOfVersion],
     [ a owl:Restriction;
         owl:allValuesFrom core:AbstractFeatureWithLifespan;
         owl:onProperty vers:Version.versionMember],
     [ a owl:Restriction;
         owl:allValuesFrom xsd:string;
         owl:onProperty vers:Version.tag],
      core:AbstractVersion;
   skos:definition "Version represents a defined state of a city
      model  consisting  of  the  dedicated  versions  of  all  city
      object  instances  that  belong  to  the  respective  city  model
      version.  Versions can have names, a description and can be
      labeled with an arbitrary number of user defined tags."@en .
```

In the second step, the DL rules are rewritten to align with a concrete SWRL syntax that can be interpreted by a parser. This work uses the OwlReady2 SWRL syntax since it closely resembles the abstract SWRL syntax (Horrocks et al., 2004), and OwlReady2 has support for parsing and inferencing, and both SWRL and OWL (Lamy, 2017). This translation requires replacing conjunctions ($\wedge$) with commas and removing the universal closure ($\forall$) from the antecedent (as it is assumed in OwlReady2). For example, following the first two steps, the DL rule 33 from Samuel et al. (2020) states that *"the time limits of [a] version transition must be greater than or equal to the end time of the previous version and the start time of the following version"*:

$$\forall vt \big(hasPreviousVersion(vt, vt_p) \wedge$$
$$has\,New\,Version(vt, vt_n) \wedge$$
$$has\,Existence\,Time(vt_p, t_1) \wedge$$
$$has\,Existence\,Time(vt_n, t_2) \wedge \qquad (3)$$
$$has\,Existence\,Time(vt, t) \big)$$
$$\rightarrow meets(t_1, t) \wedge meets(t, t_2)$$

This rule is rewritten as the following SWRL rule, taking into consideration "CityOWL" and the OWL-Time extension as defined in 3.2:

```
vers:VersionTransition.from(?vt, ?v1),
vers:VersionTransition.to(?vt, ?v2),

time_ext:hasExistenceTime(?v1, ?i1),
time_ext:hasExistenceTime(?vt, ?i2),

time_ext:hasExistenceTime(?v2, ?i3),
time:hasEnd(?i1, ?e1), time:inXSDDateTimeStamp(?e1, ?t1),
time:hasBeginning(?i2, ?bt), time:inXSDDateTimeStamp(?bt, ?t2),
time:hasEnd(?i2, ?et), time:inXSDDateTimeStamp(?et, ?t3),
time:hasBeginning(?i3, ?b2), time:inXSDDateTimeStamp(?b2, ?t3),
-> equal(?t1, ?t2), equal(?t3, ?t4).
```

Here we see that, instead of inferring that two historically successive versions must *meet*, we instead infer that if their respective beginning and end timestamps must be equal.

Finally, rewriting these rules as "safe" or "DL-safe" is required as the expressivity of OWL 2 and SWRL permits the definition of computationally expensive, undecidable logical statements for logical reasoning (Horrocks et al., 2004; Hitzler and Parsia, 2009). To ensure this, first SWRL rules with conjunctions in the consequent must be split into multiple rules using the Lloyd-Topor transfomation (Lloyd, 2012) such that only one atom remains in the consequent. Additionally, no variables may occur in the consequent that are not defined in the antecedent. For example, the SWRL rule provided above violates the former of these principles and should be rewritten as:

```
vers:VersionTransition.from(?vt, ?v1),

time_ext:hasExistenceTime(?v1, ?i1),
time_ext:hasExistenceTime(?vt, ?i2),

time:hasEnd(?i1, ?e1), time:inXSDDateTimeStamp(?e1, ?t1),
time:hasBeginning(?i2, ?bt), time:inXSDDateTimeStamp(?bt, ?t2),
-> equal(?t1, ?t2).

vers:VersionTransition.to(?vt, ?v2),

time_ext:hasExistenceTime(?vt, ?i2),
time_ext:hasExistenceTime(?v2, ?i3),

time:hasEnd(?i2, ?et), time:inXSDDateTimeStamp(?et, ?t3),
time:hasBeginning(?i3, ?b2), time:inXSDDateTimeStamp(?b2, ?t3),
-> equal(?t3, ?t4).
```

Additionally, we can rewrite these rules as their contrapositives we would like to infer an inconsistency. In this case, *if the time limits of a version transition are not greater than or equal to the end time of the previous version, that version transition is inconsistent*:

```
vers:VersionTransition(?vt),
    vers:VersionTransition.from(?vt, ?v1),
    time_ext:hasExistenceTime(?v1, ?i1), time_ext:hasExistenceTime
    (?vt, ?i2),
    time:hasEnd(?i1, ?e), time:inXSDDateTimeStamp(?e, ?t1),
    time:hasBeginning(?i2, ?b), time:inXSDDateTimeStamp(?b, ?t2),
    notEqual(?t1, ?t2),
-> owl:Nothing(?vt).
```

In SWRL an empty consequent could also be provided to induce an inconsistency; however, this manner allows the inconsistent instance to be directly tagged as inconsistent, facilitating the identification of errors.

3.4. Validating concurrent scenarios of urban evolution

We developed a proof of concept SWRL test suite (Table 1) to validate integrated datasets as the final activity of our methodology (2). Through reasoning, the suite can then infer new RDF triples based on our proposed ruleset and verify if the given ontology is logically consistent. Figure 6 shows an activity diagram of how the test suite is utilized. In order to reuse existing ontological models such as "CityOWL," the input ontology is divided into its TBox (the ontological data model) and its Abox (datasets conforming to the model). In addition to the ontology and the ruleset, a configuration for these tests can be provided for orchestrating which TBoxes, ABoxes, and rulesets to use for each test.

These elements are passed to the test suite, which uses RDFLib and OwlReady2 (Lamy, 2017) to read OWL ontologies and SWRL rules, reason upon them, and output newly inferred triples. These triples can be added back to the input ontology. In the case inconsistencies are detected for any test, these entities are logged in an output file.

4. Results

To illustrate how the proposed methodology can be used in a real-world use case, we integrated concurrent and successive open data city snapshots of a zone of interest and validated them using the methodology. Additionally, we created a web application for visualizing the integrated and validated data. This application was created using the UD-Viz urban data visualization framework (Samuel et al., 2023) for

Table 1. *The URLs and SWHID (Software Heritage identifiers) of the SWRL rules, test suite, and web application in this article. SWHIDs begin with "swh" and can be used at https:// archive.softwareheritage.org/ to view the archived resources*

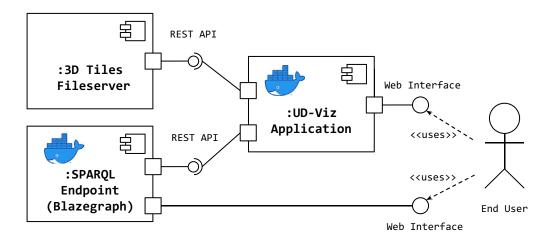| Technical contribution	Software Heritage ID with URLs
1. SWRL Rule Test Suite	swh:1:dir:0f251971359e155a07d6a67ab964d6c7f4494d3d
2. Proposed SWRL Ruleset	swh:1:cnt:86a3d1a823655c0f960ffad4331111482b075cbe;path=/
3. SWRL Rule Test Suite Configuration	swh:1:cnt:769953472c5d23e534f45f88a520ec0924bf5308;path=/
4. nD urban data application (UD-Viz + Blazegraph)	swh:1:dir:1d8a7797a11def9b85b36630d53ce61f32522740
5. Dockerized SWRL rule test suite	swh:1:dir:3d629b3b25060a53fb1c94f72b8c8fae62fdb036

Figure 6. *A UML activity diagram of the input and output artifacts used and produced by the test suite. These elements are contextualized within the M1 and M0 metamodeling layers proposed in Atkinson and Kuhne (2003).*

displaying the 3D environment, a Blazegraph SPARQL endpoint (SYSTAP, LLC, 2013) that stores and serves the validated ontology (TBox and Abox), and a fileserver for serving 3DTiles (Cozzi and Lilley, 2022) containing the 3D geometry of the urban features (Figure 7). Links to these reproducible components are available in Table 1.

The use case arises from an urban planning project of the Gratte-Ciel neighborhood of Villeurbanne, France, from 2009 to 2018, at different levels of detail (Figure 8). These skyscrapers, developed in the 1930s and some of the first of their class at the time, are today considered built cultural heritage. The datasets used to represent proposed scenarios of evolution of this zone of interest come from the Lyon, France, metropole open urban data repository (Métropole de Lyon, 2021a, 2021b, 2021c, 2022) and are integrated to conform to CityOWL and the Workspace model using the approach proposed in Vinasco-Alvarez (2023).

Once validated, we use SPARQL queries to provide views of the workspace data and the evolving urban features contained within. For example, the following query permits to construction a knowledge graph of *all the versions and version transitions of a workspace and their immediate neighbors.* Figure 9 provides an illustration of the results of this query in an urban data web application.

```
CONSTRUCT {
    ?subject ?predicate ?object.
}
WHERE {
```

Figure 7. *UML component diagram of the services used for the proposed web application for visualizing concurrent scenarios of urban evolution.*

Figure 8. *An aerial photo of the Gratte-Ciel neighborhood from 1936 Commons (2020) (left) and from 2018 as a 3D city model (Vinasco-Alvarez, 2023) (right).*

```
    ?subject a vers:VersionTransition;
    (vers:VersionTransition.to|vers:VersionTransition.from) ?object;
    ?predicate ?object .
}
```

While the previous query is useful for presenting an overview of the evolution of the zone of interest over time, queries can also be proposed at a more granular level to retrieve the individual changes features may have undergone between two specific versions. The query below can be used to construct a graph of all the transactions between version 2015 and 2018 of the neighborhood in the 1st proposed scenario of evolution. Figure 10 provides an illustration of the results of this query in an urban data web application.
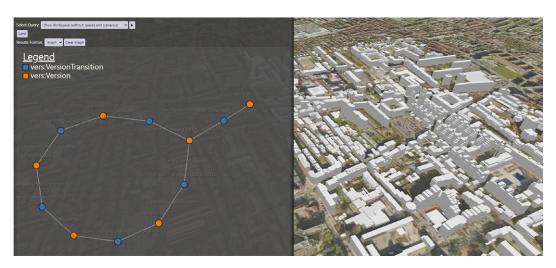
```
CONSTRUCT {
    ?subject ?predicate ?object.
}
WHERE {
    {
        ?subject a wksp:Scenario;
            ?predicate ?object .
```

Figure 9. *A screenshot of the 3D urban data application visualizing the state of the Gratte Ciel neighborhood in 2018 (right) and the versions and version transitions of the workspace (left).*

Figure 10. *A screenshot of the application visualizing the state of the Gratte Ciel neighborhood between 2015 and 2018 (right) and the transactions (or changes) between each building from each of these versions. The type of transaction is also used to color the buildings in the right view using the 3D tiles extension proposed in Jaillot et al. (2021).*

```
      FILTER(?subject = ws:scenario_1).
   } UNION { .
     ?subject a vers:VersionTransition;
        ?predicate ?object .
   FILTER(?subject = vt2015_2018:versionTransition_2015_2018).
   } UNION { .
     ?subject a vers:Version;
        ?predicate ?object .
   FILTER(?subject = v2015:version_2015 ||
        ?subject = v2018:version_2018).
```

```
    } UNION {
      vt2015_2018:versionTransition_2015_2018.
         vers:VersionTransition.transaction? subject .
      ?subject a vers:Transaction;
         ?predicate? object .
    }
  }
```

Taking into account the work of Samuel et al. (2020), we focused on generalizing the Workspace model to work for any urban data. The original workspace model focused on rules for the documentation of historical evolution, which were not translated. Therefore, we did not propose rules for the *existing* or *imagined* version types and the *influenced* version transition type. We determined that these two aspects of the Workspace model were conceived for a specific use case in cultural heritage documentation and fall out of the scope of general urban data evolution. However, these rules could be translated using the methodology proposed in our work for documentary use cases that require them. Additionally, our translated rules do not include the subsumption and equivalence DL rules proposed in Samuel et al. (2020) as they are already defined by `rdfs:subClassOf` and `owl:equivalentClass` axioms in the CityOWL ontologies.

In our context, we focused on official datasets from the Lyon Metropole, which are updated at a low frequency (e.g., every 3 years). However, the proposed ruleset can be used to validate datasets that are updated at a higher frequency (e.g., daily, weekly, monthly, etc.) as long as the data is structured according to the CityOWL and Workspace models. The datasets used in this work were validated for conformance to the CityOWL and Workspace models, and did not contain any inconsistencies; hence, we do not encounter any data quality issues. However, we note that the proposed ruleset is not exhaustive and does not cover all the possible inconsistencies that may arise in urban data. Furthermore, we do not take into account the probabilistic nature of dynamic knowledge graphs Pujara and Getoor (2014), as this work focuses on deterministic scenarios of urban evolution. But we have not yet tested this in practice as well as the performance of the ruleset on larger datasets. An exhaustive evaluation of the performance of the ruleset on dynamic and larger datasets is left for future work.

To promote the reproducibility of this work, links to all artifacts, applications, and tools discussed here are available in Table 1. As is denoted in Figure 7, containerization technologies (i.e., Docker (2025)) are used to allow the components of the web application and the proof of concept test suite to be easily installed and used through portable 'containers' independent of hardware architecture or installed programs.

5. Conclusion

This article puts forth a novel methodology for validating integrated 3D city models for illustrating concurrent and successive views of urban evolution. This methodology also capitalizes on new standards and works for representing concurrent scenarios of urban evolution. Thanks to this methodology, we also propose a formalization of machine-readable SWRL rules for validating these evolution scenarios proof of concept tests suite. An integrated dataset based on open urban data was validated for conformance and temporal consistency using these rules. To demonstrate how these validated data can be exploited, they were visualized through a proof-of-concept 4D urban data web application with an extension to the UD-Viz urban data visualization framework. Using SPARQL queries, we also demonstrated how such validated datasets could be used in a 3D virtual environment for visualizing these scenarios of urban evolution. Additionally, the technical contributions of this work are reproducible, including the proposed test suite and web application. Our methodology is well-documented, modular, and makes use of open-source components and libraries. Thanks to these modular components for validating, storing, and visualizing ontological graph data of urban evolution, these components can be replaced as necessary.

When comparing this work to other Semantic Web approaches for integrating 4D urban data, there are several complementary aspects to note. In particular, approaches such as the Parallel World Framework with dynamic knowledge graphs are also capable of comparing concurrent or successive—real and imagined—urban evolution scenarios. One advantage of a versioning approach is that it can also be used to compare a real multiple imagined successive and concurrent states of the city, whereas the Parallel World Framework only focuses on one real and one imagined world state. These agent-based approaches also lend themselves well to implementing high-frequency urban digital twins, such as those that integrate streaming sensor data, which has not been done with our approach. Future work could explore using the proposed SWRL rule test suite as a validating agent for permitting more expressive data constraints. In addition, implementing a snapshot-based versioning approach within a dynamic knowledge graph can be a strategy for efficiently storing a history of states within Parallel Framework-based approaches. This could permit existing dynamic urban knowledge graph approaches to more easily compare concurrent AND successive urban states of the city. While within the context of this work, the proposed rule test suite performs adequately for our use-case with the low frequency updates (Section 4), applications that require validation in near real-time pose more demanding data validation needs.

Additionally, we are currently exploring how to easily translate rules from standardized abstract test suites such as the CityGML 3.0 Conceptual Model test suite, which defines rules through unstructured text. To make inference on the proposed rules more optimal at scale, we are working on assuring all translated rules are DL-safe according to the proposed approach. We note that there may be applications to using unsafe rules on small-scale datasets; however, this was not examined in this work. Future works will include more quantitative studies, exploring the limitations of our approach at different urban scales and levels of detail (e.g., from buildings to large-scale territories) with respect to related works. Finally, we would like to complete the proposed ruleset for Workspaces, including rules for validating proposed changes with additional heterogeneous data (e.g., multimedia documents, cultural heritage thesauri).

References

Akroyd J, **Mosbach S**, **Bhave A and Kraft M** (2021) Universal digital twin-a dynamic knowledge graph. *Data-Centric Engineering 2*, e14.

Allen JF (1984) Towards a general theory of action and time. *Artificial Intelligence 23*(2), 123–154.

Aparicio JT, **Arsenio E**, **Santos F and Henriques R** (2024) Using dynamic knowledge graphs to detect emerging communities of knowledge. *Knowledge-Based Systems 294*, 111671.

Atkinson C and Kuhne T (2003) Model-driven development: A metamodeling foundation. *IEEE Software 20*(5), 36–41.

Balsebre P, **Yao D**, **Cong G and Hai Z** (2022) *Geospatial Entity Resolution. In Proceedings of the ACM Web Conference 2022, WWW '22*. New York: Association for Computing Machinery, pp. 3061–3070

Batsakis S and Petrakis EGM (2011) SOWL: A framework for handling spatio-temporal information in OWL 2.0. In Bassiliades N, Governatori G and Paschke A (eds), *Rule-Based Reasoning, Programming, and Applications*. Berlin/Heidelberg: Springer, pp. 242–249

Batsakis S, **Petrakis EGM**, **Tachmazidis I and Antoniou G** (2017) Temporal representation and reasoning in OWL 2. *Semantic Web 8* (6), 981–1000.

Batty M (2018) Digital twins. *Environment and Planning B: Urban Analytics and City Science 45*(5), 817–820.

Bernard C (2019) *Immersing Evolving Geographic Divisions in the Semantic Web*. Phd thesis, Université Grenoble Alpes.

Buil Aranda C, **Olivier C**, **Das S**, **Feigenbaum L**, **Gearon P**, **Glimm B**, **Harris S**, **Herman I**, **Humfrey N**, **Michaelis N**, **Ogbuji C**, **Perry M**, **Passant A**, **Polleres A**, **Prud'hommeaux E**, **Seaborne A and Williams GT** (2013) *SPARQL 1.1 Overview*. https://www.w3.org/TR/sparql11-overview/.

buildingSMART (2024). *Industry Foundation Classes (IFC) - buildingSMART International*. https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/.

Chadzynski A, **Krdzavac N**, **Farazi F**, **Lim MQ**, **Li S**, **Grisiute A**, **Herthogs P**, **von Richthofen A**, **Cairns S and Kraft M** (2021) Semantic 3D city database — An enabler for a dynamic geospatial knowledge graph. *Energy and AI 6*, 100106.

Chaturvedi K and Kolbe TH (2019) A requirement analysis on extending semantic 3d city models for supporting time-dependent properties. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences IV-4/W9*, 19–26.

Chaturvedi K, **Smyth CS**, **Gesquière G**, **Kutzner T and Kolbe TH** (2017) Managing versions and history within semantic 3D city models for the next generation of CityGML. In Abdul-Rahman A (ed.), *Lecture Notes in Geoinformation and Cartography, Advances in 3D Geoinformation*. Springer, pp. 191–206

Commons W (2020) *File:vue aérienne de la cité des gratte-ciel en 1936.jpg — wikimedia commons, the free media repository*. https://commons.wikimedia.org/w/index.php?title=File:Vue_a%C3%A9rienne_de_la_cit%C3%A9_des_Gratte-Ciel_en_1936.jpg&oldid=477031065

Cozzi P and Lilley S (2022) *3D-Tiles Specification*. https://github.com/CesiumGS/3d-tiles/tree/main/specification.

Diakite AA and Zlatanova S (2020) Automatic geo-referencing of BIM in GIS environments using building footprints. *Computers, Environment and Urban Systems 80*, undefined–undefined.

Docker Inc (2025) *Docker: Accelerated Container Application Development*. https://www.docker.com/.

Eibeck A, **Lim MQ and Kraft M** (2019) J-park simulator: An ontology-based platform for cross-domain scenarios in process industry. *Computers & Chemical Engineering 131*, 106586.

Eibeck A, **Chadzynski A**, **Lim MQ**, **Aditya K**, **Ong L**, **Devanand A**, **Karmakar G**, **Mosbach S**, **Lau R**, **Karimi IA** *et al* (2020) A parallel world framework for scenario analysis in knowledge graphs. *Data-Centric Engineering 1*, e6.

Glimm B, **Horrocks I**, **Motik B**, **Stoilos G and Wang Z** (2014) HermiT: An OWL 2 reasoner. *Journal of Automated Reasoning 53*(3), 245–269.

Grieves M (2014) *Digital Twin: Manufacturing Excellence Through Virtual Factory Replication*. [White paper].

Harbelot B (2015) *Continuum : un modèle spatio-temporel et sémantique pour la découverte de phénomènes dynamiques au sein 'environnements géospatiaux*. Theses, Université de Bourgogne

Hijazi IH, **Krauth T**, **Donaubauer A and Kolbe T** (2020) 3DCITYDB4BIM: A system architecture for linking BIM server and 3D CITYDB for BIM-GIS-integration. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences 4*, 195–202.

Hitzler P and Parsia B (2009) Ontologies and rules. In *Handbook on Ontologies, International Handbooks on Information Systems*. Springer, pp. 111–132.

Hlomani H and Stacey D (2014) Approaches, methods, metrics, measures, and subjectivity in ontology evaluation: A survey. *International Journal on Semantic Web and Information Systems 1*(5), 1–11.

Hor A-H and Sohn G (2021) Design and evaluation of a BIM-GIS integrated information model using RDF graph database. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences VIII-4-W2-2021*, 175–182.

Hornsby K and Egenhofer MJ (2000) Identity-based change: A foundation for spatio-temporal knowledge representation. *International Journal of Geographical Information Science 14*(3), 207–224.

Horrocks I, **Patel-Schneider PF**, **Boley H**, **Tabet S**, **Grosof B and Dean M** (2004) *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. https://www.w3.org/submissions/SWRL/.

INSPIRE Drafting Team "Data Specifications" (2014) *INSPIRE Generic Conceptual Model*. https://knowledge-base.inspire.ec.europa.eu/publications/inspire-generic-conceptual-model_en.

Jaillot V, **Servigne S and Gesquière G** (2020) Delivering time-evolving 3d city models for web visualization. *International Journal of Geographical Information Science 34*(10), 2030–2052.

Jaillot V, **Rigolle V**, **Servigne S**, **Samuel JS and Gesquière G** (2021) Integrating multimedia documents and time-evolving 3D city models for web visualization and navigation. *Transactions in GIS 25*(3), 1419–1438.

Kolbe TH, **Kutzner T**, **Smyth CS**, **Nagel C**, **Roensdorf C and Heazel C** (2021) *OGC City Geography Markup Language (CityGML) Part 1: Conceptual Model Standard*. https://docs.ogc.org/is/20-010/20-010.html.

Kutzner T (2016) *Geospatial Data Modelling and Model-Driven Transformation of Geospatial Data Based on Uml Profiles*. PhD Thesis, Technische Universität München.

Kutzner T, **Chaturvedi K and Kolbe TH** (2020) CityGML 3.0: New functions open up new applications. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*.

Kyzirakos K, **Savva D**, **Vlachopoulos I**, **Vasileiou A**, **Karalis N**, **Koubarakis M and Manegold S** (2018) GeoTriples: Transforming geospatial data into RDF graphs using R2RML and RML mappings. *Journal of Web Semantics 52–53*, 16–32.

Lamy J-B (2017) Owlready: Ontology-oriented programming in python with automatic classification and high level constructs for biomedical ontologies. *Artificial Intelligence in Medicine 80*, 11–28.

Lassila O, **Hendler J and Berners-Lee T** (2001) The semantic web. *Scientific American 284*(5), 34–43.

LIRIS (2023) *Virtual City Project.* https://projet.liris.cnrs.fr/vcity/.

Liu X, **Wang X**, **Wright G**, **Cheng J**, **Li X and Liu R** (2017) A state-of-the-art review on the integration of building information Modeling (BIM) and geographic information system (GIS). *ISPRS International Journal of Geo-Information 6*(2), 53.

Liu AH, **Ellul C and Swiderska M** (2021) Decision making in the 4th dimension—Exploring use cases and technical options for the integration of 4D BIM and GIS during construction. *ISPRS International Journal of Geo-Information 10*(4), 203.

Lloyd JW (2012) *Foundations of Logic Programming*. Springer

Malinverni ES, **Naticchia B**, **Lerma Garcia JL**, **Gorreja A**, **Lopez Uriarte J and Di Stefano F** (2020) A semantic graph database for the interoperability of 3D GIS data. *Applied Geomatics*.

Métropole de Lyon (2021a) *Maquettes 3D texturées 2009 des communes de la Métropole de Lyon.* https://data.grandlyon.com/portail/fr/jeux-de-donnees/maquettes-3d-texturees-2009-communes-metropole-lyon/donnees.

Métropole de Lyon (2021b) *Maquettes 3D texturées 2012 des communes de la Métropole de Lyon.* https://data.grandlyon.com/portail/fr/jeux-de-donnees/maquettes-3d-texturees-2012-communes-metropole-lyon/donnees.

Métropole de Lyon (2021c) *Maquettes 3D texturées 2015 des communes de la Métropole de Lyon.* https://data.grandlyon.com/portail/fr/jeux-de-donnees/maquettes-3d-texturees-2015-communes-metropole-lyon/donnees.

Métropole de Lyon (2022) *Maquettes 3D texturées 2018 des communes de la Métropole de Lyon.* https://data.grandlyon.com/portail/fr/jeux-de-donnees/maquettes-3d-texturees-2018-communes-metropole-lyon/donnees.

Musen MA and **Protégé Team** (2015) The protégé project: A look Back and a look forward. *AI matters 1*(4), 4–12.

Nguyen SH and Kolbe TH (2021) Modelling changes, stakeholders and their relations in semantic 3D city models. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VIII-4-W2-2021, 137–144. Copernicus GmbH. ISSN: 2194-9042.

Nguyen SH, **Yao Z and Kolbe TH** (2017) Spatio-semantic comparison of large 3D city models in CityGML using a graph database. In *Proceedings of the 12th International 3D GeoInfo Conference 2017*, pp. 99.

O'Connor MJ and Das AK (2010) A method for representing and querying temporal information in owl. In *International Joint Conference on Biomedical Engineering Systems and Technologies*. Springer, pp. 97–110.

Pédrinis F, **Morel M and Gesquière G** (2015) *Change Detection of Cities, Chapter 9*. Cham: Springer, pp. 123–139

Pfeiffer M, **Carré C**, **Delfosse V**, **Hallot P and Billen R** (2013) Virtual leodium: From an historical 3d city scale model to an archaeological information system. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W1*, 241–246.

Pujara J and Getoor L (2014) Building dynamic knowledge graphs. *NIPS Workshop on Automated Knowledge Base Construction 9*.

Samuel J, **Servigne S and Gesquière G** (2020) Representation of concurrent points of view of urban changes for city models. *Journal of Geographical Systems*.

Samuel J, **Jaillot V**, **Colin C**, **Vinasco-Alvarez D**, **Boix E**, **Servigne S and Gesquière G** (2023) UD-SV: Urban data services and visualization framework for sharing multidisciplinary research. *Transactions in GIS 27*(3), 841–858.

Sehgal V, **Getoor L and Viechnicki PD** (2006) Entity resolution in geospatial data integration. In *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems, GIS '06*. New York: Association for Computing Machinery, pp. 83–90.

Sirin E, **Parsia B**, **Grau BC**, **Kalyanpur A and Katz Y** (2007) Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics 5*(2), 51–53.

Software Heritage (2025) *Software Heritage.* https://www.softwareheritage.org/.

Stefani C, **De Luca L**, **Véron P and Florenzano M** (2008) Reasoning about space-time changes: An approach for modeling the temporal dimention in architectural heritage. In *IADIS International Conference*. Netherlands, pp. 1–6.

SYSTAP, LLC (2013) *Blazegraph Database.* https://blazegraph.com/.

Tran B-H, **Plumejeaud C**, **Bouju A and Bretagnolle V** (2016) Towards a semantic framework for exploiting heterogeneous environmental data. *International Journal of Metadata, Semantics and Ontologies 11*.

Tran B-H, **Aussenac-Gilles N**, **Comparot C and Trojahn C** (2020) Semantic integration of raster data for earth observation: An RDF dataset of territorial unit versions with their land cover. *ISPRS International Journal of Geo-Information 9*(9), 503.

Trivedi R, **Dai H**, **Wang Y and Song L** (2017) Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In Precup and Teh YW (eds.), *Proceedings of the 34th International Conference on Machine Learning, Volume 70 of Proceedings of Machine Learning Research*. PMLR, pp. 3462–3471

Tsarkov D and Horrocks I (2006) FaCT++ description logic Reasoner: System description. In Furbach U and Shankar N (eds.), *Automated Reasoning, Lecture Notes in Computer Science*. Berlin/Heidelberg: Springer, pp. 292–297

Uschold M and Gruninger M (2004) Ontologies and semantics for seamless connectivity. *ACM SIGMOD Record 33*(4), 58–64.

Usmani AU, **Jadidi M and Sohn G** (2021) Towards the automatic ontology generation and alignment of BIM and GIS data formats. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VIII-4-W2-2021, 183–188.

Van Ruymbeke M, **Carré C**, **Delfosse V**, **Pfeiffer M and Billen R** (2015) *Towards an Archaeological Information System: Improving the Core Data Model.* Archaeopress

Vinasco-Alvarez D (2023) *Model-Driven Integration of Heterogeneous n-Dimensional Urban Data.* Theses, Université Lyon 2 Lumière, Lyon, France.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2021) Towards limiting semantic data loss in 4D urban data semantic graph generation. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VIII-4/W2-2021, 37–44.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2022) *CityGML 3.0 OWL Versioning Ontology.* https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Ontologies/CityGML/3.0/versioning.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024a) *Dockerized SWRL Rule Test Suite.* https://archive.softwareheritage.org/swh:1:dir:3d629b3b25060a53fb1c94f72b8c8fae62fdb036.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024b) *nd Urban Data Application (UD-Viz+ Blazegraph).* https://archive.softwareheritage.org/swh:1:dir:1d8a7797a11def9b85b36630d53ce61f32522740.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024c) *Proposed SWRL Ruleset.* https://archive.softwareheritage.org/swh:1:cnt:86a3d1a823655c0f960ffad4331111482b075cbe;path=/.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024d) *SWRL Rule Test Suite.* https://archive.softwareheritage.org/swh:1:dir:0f251971359e155a07d6a67ab964d6c7f4494d3d.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024e) *SWRL Rule Test Suite Configuration.* https://archive.softwareheritage.org/swh:1:cnt:769953472c5d23e534f45f88a520ec0924bf5308;path=/.

Vinasco-Alvarez D, **Samuel J**, **Servigne S, and Gesquière G** (2024f) Towards an Automated Transformation of an nD Urban Data Model to a Computational Ontology Network: From UML to OWL, From CityGML 3.0 to "CityOWL". *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **X-4/W4-2024**, 231–238.

W3C (2025) *W3C.* https://www.w3.org/.

Zalamea O, **Orshoven J and Thérèse S** (2016) From a CityGML to an ontology-based approach to support preventive conservation of built cultural heritage. In *AGILE 2016.*

Appendix: URI prefixes

Table A1. *The prefixes used in this article with their respective URIs (Vinasco-Alvarez, 2023)*

Prefix	URI
skos	http://www.w3.org/2004/02/skos/core#
time	http://www.w3.org/2006/time#
time_ext	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Ontologies/Time/time-extension#
core	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Ontologies/CityGML/3.0/core#
vers	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Ontologies/CityGML/3.0/versioning#
wksp	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Ontologies/Workspace/3.0/workspace#
ws	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Datasets/GratteCiel_Workspace_2009_2018/3.0/GratteCiel_2009_2018_Workspace#
vt2015_2018	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Datasets/GratteCiel_Workspace_2009_2018/3.0/Transition_2015_2018#
v2015	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Datasets/GratteCiel_Workspace_2009_2018/3.0/GratteCiel_2015_split#
v2018	https://dataset-dl.liris.cnrs.fr/rdf-owl-urban-data-ontologies/Datasets/GratteCiel_Workspace_2009_2018/3.0/GratteCiel_2018_split#