Conversion of XML Property Schema according to ISO 23386 into RDF Graphs

Kilian Kandt[1] and Sven Zentgraf[1]
[1]Chair of Computing in Engineering, Ruhr University Bochum, Universitätsstraße 150, 44801 Bochum, Germany
E-mail(s): kilian.kandt@rub.de, sven.zentgraf@rub.de

Abstract: A fundamental part of the infrastructure is civil engineering. Due to its huge economic and resource relevance optimizing the resource efficiency in this sector has high societal relevance. The German research project RekoTi is initiated to increase this efficiency by closing the mineral material cycle in municipal civil engineering. The project aims to create resource strategies with stakeholders from different disciplines using BIM. In terms of resource aspect, the description of objects in BIM models needs to be formulated in a standardized, maintainable, and exchangeable way. ISO 23386 defines such a data format and methods, for combining information from different sources. As XML is an established format for describing data, the underlying structures of the developed resource efficiency strategies are modeled in XML compliant with ISO 23386. To provide the modeled data in a more queryable way it should be converted into a graph-based representation. So this paper aims to transform XML documents, compliant with ISO 23386, into RDF/XML documents. This is done by examining the current research to find a suitable variant for this transformation. The result of this study is a concept for transforming XML documents conform to ISO 23386 utilizing the XSLT framework. Saved into a database the transformed RDF/XML data is queried to evaluate the resulting data quality and thereby the validity of the created stylesheet.

Keywords: BIM, RDF/XML, XSLT, Resource Efficiency, Energy Simulation

1 Introduction

The central part in infrastructure is the civil engineering, due to its huge economical and resource relevance. Therefore, optimizing the resource efficiency in this sector has a high societal relevance. To increase resource efficiency by closing the mineral material cycle in municipal civil engineering, the German research project Ressourcenplan kommunaler Tiefbau (RekoTi) was formed. The project examines how resources can be used efficiently through the entire life cycle of the infrastructure, regarding the legal and administrative requirements. Subsequently, the results of this investigation should be analyzed with focus on the deficits and potentials about their possibilities to increase the resource efficiency in the mineral material cycle.

For creating resource strategies, Building Information Modeling (BIM) is of immense relevance, as it acts as a linking instrument between stakeholders from different disciplines. The key stakeholders in planning, building, operating and maintaining infrastructure projects are the public authorities. To describe the properties of objects in BIM models, e.g. in terms of resource aspects, in a standardized way, a data format for describing, maintaining, and exchanging information is needed. ISO 23386 defines such a data format and methods, especially for combining information from different sources or stakeholders [1]. Within this data format, information is recorded in properties that are themselves organized in a tree-shaped group of properties hierarchy.

An established format to describe data in a standardized manner is eXtensible Markup Language (XML). This data representation makes the gathered data human and machine-readable and is easily maintainable, and exchangeable. Thus, the underlying structures of the developed resource efficiency strategies are modeled in XML complaint with ISO 23386. A problem in modeling the data in this structure is that the queryability of XML is neither efficient nor feasible. To overcome this issue, Semantic Web technologies can be used. Resource Description Framework (RDF) is an easy queryable graph-based data representation that describes resources (metadata), is the basis of the Semantic Web and offers an XML-based data serialization the RDF/XML format [2].

This paper aims to transform XML documents, which are defined compliant with ISO 23386, into RDF/XML documents efficiently. An ontology that maps the data format of the standard and introduces the functionalities to attach the created properties to a Feature of Intrest (FOI) is presented in [3]. Therefore, the converted documents should conform with this ontology. A concept to convert XML serialized documents into RDF/XML serialized documents is presented. For this purpose, the current research is examined to find suitable variant for this transformation (c.f section 2). In section 3 the results of this research are analyzed first, and the most suitable technology is selected. With a focus on the syntactic structure of the input data given, a concept for the conversion process from XML to RDF/XML is presented. Finally, the results are present and evaluated by reviewing the created eXtensible Stylesheet Language Transformations (XSLT) stylesheet and querying the transformed RDF/XML data(see section 4). A conclusion and further outlook are given in section 5.

2 Background

This section emphasizes the main terms, concepts, and previous research relevant to the subsequent conceptualization. ISO 23386 introduces a methodology adopted by the International Organization for Standardization (ISO) to describe, manage and maintain properties in a linked data network in construction [1]. Through this standard, the groups of properties and properties that are important for the description of resource efficiency parameters can be divided and described unambiguously in a standardized way [1]. However, this methodology is still rarely used in the infrastructure area. Alani, Dawood, Rodriguez, *et al.* [4] implemented a framework based on Semantic Web technologies to convert Product Data Templates (PDTs) into a graph-based representation and used the ISO 23386 data model to support the linked data creation in the water infrastructure sector. The authors emphasize the need for a standardized description format for describing building elements in this sector. In addition, the authors recommend manufacturers and suppliers in the construction industry provide their PDTs in RDF-based formats in accordance with the ISO 23386. Furthermore, buildingSmart international, which is focused on the standardization process in the construction industry, provides an online service for classifications, standards, dictionaries, and their properties, referred to as buildingSMART Data Dictionary (bsDD) [5]. buildingSmart announced that bsDD will support the ISO 23386 data model as an export format soon. This extension enables BIM modelers to enrich their models with properties modeled complaint to the standard.

RDF is a World Wide Web Consortium (W3C) framework for representing resources (metadata) in triples (subject, predicate, object). In these, a predicate relates a subject representing a resource and an object representing a resource or a literal. RDF can be complemented through vocabularies such as the Web Ontology Language (OWL) to represent ontologies and can be serialized in different formats such as RDF/XML, Turtle, or N-Triples. RDF/XML allows RDF graphs to be represented in an XML-based serialization.

XML is a widely used text-based format for representing ordered information in a logical tree structure [6]. One extension is the XSLT based on the so-called *stylesheet*, in which the desired output format is described in references to the original XML document [7]. With XSLT stylesheets, tags and their contents from the XML document are localized by XPath expressions, a language for navigating to specific tags within XML documents [8], and placed in a new context. Templates can be used to describe a default structure that is called each time the tag belonging to the template occurs in the XML document.

The conversion of XML documents, which are clearly defined by XML Schema, into documents in RDF/XML format is gaining a broad range of applications in research due to the increasing use of Semantic Web technologies. The Gleaning Resource Descriptions from Dialects of Languages (GRDDL) working group from the W3C is working on the process of *lifting* XML data [9]. The Term *Lifting* describes the process of converting XML to RDF/XML data. For these transformations the GRDDL group uses XSLT since it was developed to express transformations between XML formats such as RDF/XML.

Bohring and Auer [10] proposed a transformation from XML to OWL data. They make the assumption that an XSD document must first be generated from their XML document to derive an OWL model as well as a suitable XSLT stylesheet from this schema. Here, it was noticed that the transformation from XML to OWL using XSLT, which can be seen as a similar process to the transformation from XML to RDF/XML, can produce an uncertain result, especially in the absence of a schema. The assumption XSLT may provide a less accurate result is also confirmed in [11] and [12]. However, Akhtar, Kopecký, Krennwallner, *et al.* [11] also indicate that since XSLT is Turing-complete, this uncertainty can be removed by describing the transformation in more detail in the stylesheet. Bischof, Decker, Krennwallner, *et al.* [12] also notes that *lifting* can be done precisely with XSLT, but that the underlying stylesheet must become correspondingly more detailed. Both papers address XML Query Language SPARQL RDF Protocol and Query Language (XSPARQL) as an alternative for XSLT conversion. XSPARQL is supposed to be a combination of XQuery and SPARQL. XQuery and SPARQL are both database query languages, with XQuery used for queries and transformations within XML documents and SPARQL used for querying RDF documents. Components of SPARQL have been subsumed into XQuery, and thus should guarantee error-free lifting [12].

3 Concept

This paper focuses on the conceptualization of a transformation process of XML documents compliant to ISO 23386 into RDF/XML documents. As denoted in section 2, XSLT and XSPARQL are two possible frameworks for fulfilling this task. XSPARQL offers in comparison to XSLT a better time complexity, resulting from embedded XML Query Language (XQuery) functionalities [13]. Compared with XSLT, XSPARQL is less prone to errors whenever the structure of the input documents differs. Therefore, the transformation with XSPARQL is a bit more robust against deviations in the input data [12]. This robustness can be considered subordinate because the input data is always first validated against the standard's data format and then transformed after the validation was successful. The validation in advance guarantees the correctness of the input data. A limitation of XSPARQL is its much higher complexity compared to XSLT resulting in a higher workload. Based on these considerations, both technologies can be used for transforming XML documents to RDF/XML. Nevertheless, XSLT is used within this paper, as it provides a more straightforward approach for the transformation and is more suitable for small to medium data sets according to Sherman [13], which are the main data sets used in this project.

An XSLT transformation relies on an underlying XSLT stylesheet that interprets an XML document and transforms its content into the desired format. If the given XML document contains an XML tag, which is defined in the stylesheet, an eXtensible Stylesheet Language (XSL) template is applied to convert the XML tag from the input format to the output format. To support the conversion of the input XML documents into RDF/XML documents using an XSLT stylesheet, it was necessary to analyze the structure of the input data.

The analysis of the underlying XML format showed that there are three basic types of data descriptions used within the input data. These are plain literals, elements that refer to resources like properties, and collections of literals. For each type a basic template concept is created. The resulting templates than can easily be reused by adjusting the according attribute names.

Most attributes in the input XML represent plane literals. These elements are converted with the template shown in listing 1. The shown example implements the transformation of the Globally Unique Identifier (GUID) attribute of a group of properties.The template consists of an XML Path Language (XPath) expression that reads out the value stored in the `match`-path and places it in the corresponding `value-of` tag in the RDF/XML representation.

Listing 1: XSL template for transforming literals using the example of the GUID attribute

```
<xsl:template match="Container/propertyGroup/guid">
    <bimstruct:GloballyUniqueIdentifier>
        <xsl:value-of select="."/>
    </bimstruct:GloballyUniqueIdentifier>
</xsl:template>
```

The second group of attributes represents dependencies between properties or groups of properties instances. For the transformation of these elements the GUIDs stored in the corresponding XML tag forms the basis of a new Uniform Resource Identifier (URI) which references the `rdf:resource` the dependency points to. All relationships specified in the data format of the standard can be transformed in RDF/XML accordingly.

The last template type is used for literals appearing as a list within properties and group of properties. Because they appear several times in an object, they are placed within `Collection` tags in the RDF/XML document. In addition, it is important that every element of the collection still can be uniquely identified to make appropriate querying possible after the transformation. Therefore a new identifier is created out of the GUID of the corresponding parent object and the position of the element in the list. An example for transforming the `namesInLanguage` collection of a property is shown in listing 2.

In summary, all attributes of the XML documents conforming to the standard can be converted to RDF/XML using the conceptualized templates, with one exception. For the transformation of the optional attribute `BoundaryValues` the `Collection` template has to be extended. This adaptation will not be discussed further in this work.

Listing 2: XSL template for transforming a list of literals using the example of the *Names in Language* list of a property

```
<xsl:template match="Container/property/namesInLanguage">
        <xsl:variable name="nameCounter">
                <xsl:value-of select="count(preceding-sibling::namesInLanguage)"/>
        </xsl:variable>
        <xsl:variable name="GUID">
                <xsl:value-of select="../guid"/>
        </xsl:variable>
        <bimstruct:NameInLanguage rdf:ID="NameInLanguage_{$GUID}_{$nameCounter}">
                <bimstruct:Name>
                        <xsl:value-of select="./name"/>
                </bimstruct:Name>
                <bimstruct:Language>
                        <xsl:value-of select="./language"/>
                </bimstruct:Language>
        </bimstruct:NameInLanguage>
</xsl:template>
```

4 Evaluation

In the following, the created XSLT stylesheet is presented and evaluated. The stylesheet converts XML documents based on the data model of ISO 23386 into a graph-based representation in RDF/XML format. The conversion of an XML document compliant with the standard can be performed with the implemented templates introduced in section 3. The Evaluation has shown that checking the XML document in advance syntactically makes the implemented XSLT robust against deviations in the document structure. One issue that occurs while testing is the possibility of empty lists in the input documents. The attribute list `descriptionsInLanguage` for example is denoted as optional in the standard and can be omitted in properties in the input document. The current version of the stylesheet converts such documents without any error but produces empty collection elements in the output data. These empty collection elements don't lead to errors but consume disk space without offering any value. Therefore this issue needs to be fixed in future work.

The Resulting RDF/XML documents are inserted into a graph database to make them queryable and thus attachable to FOIs to use them for rule checking, quantity or cost estimations, or validation purposes. Therefore, in figure 1, a query for gathering the names of groups of properties is presented. Within the SPARQL query, all objects were collected belonging to the `rdf:type bimstruct:GroupOfProperties`. In the next step the literals `bimstruct:GloballyUniqueIdentifier` and `bimstruct:VersionNumber` are gathered for each object. The consecutive step searches for the related `bimstruct:NamesInLanguage` collection and extracts the first name out of it. The corresponding result of the query is shown at the bottom of figure 1. One limitation that became apparent when querying the data was that converted dependencies pointing to resources that do not exist result in errors. Such an error can occur if mistakes arise when creating references between properties and groups of properties. From this, it can be concluded that the input data also needs to be checked for semantic correctness before the transformation. However, this cannot be realized within the XSLT stylesheet and must therefore be validated in advance similar to the syntactic validation. In summary, the evaluation states that the data converted with the developed XSLT stylesheet is suitable to describe resource efficiency specific BIM objects. However, future testing using data from real infrastructure projects must further examine the suitability and robustness of the system.

5 Conclusion

This paper provides a conceptualization for the transformation of XML documents containing properties according to ISO 23386 into the RDF/XML format to support the description of objects in BIM Models through properties, e.g. in terms of resource aspects, in a standardized way. Possible technologies to realize the transformation of XML documents are considered from literature according to the robustness, accuracy, complexity, and performance of the different approaches. As a result of this consideration, the XSLT framework was determined as the most suitable. Requirements from the XML document structure based on the standards data format were systematically analyzed to identify patterns within the data to improve the XSLT template creation by defining reusable patterns for the

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX bimstruct: <http://www.bimstruct.de/v0/bimstruct-schema#>
SELECT ?guid ?name ?version
WHERE
{
    ?propGroup rdf:type bimstruct:GroupOfProperties.
    ?propGroup bimstruct:GloballyUniqueIdentifier ?guid.
    ?propGroup bimstruct:VersionNumber ?version.
    ?propGroup bimstruct:NamesInLanguage ?namesInLanguage.
    ?namesInLanguage rdf:first ?nameInLanguage.
    ?nameInLanguage bimstruct:Name ?name.
}
Limit 3
```

QUERY RESULTS
Showing 1 to 3 of 3 entries

Search: Show 50 entries

	guid	name	version
1	"422B41EA-C177-4798-874E-36D4E0256435"	"ATB-BeStra_08"	"1"^^xsd:integer
2	"67BB1B2F-9F88-4700-B49C-8B6D56FFF279"	"Ausstattung"	"1"^^xsd:integer
3	"227FE59D-124F-48FD-89ED-0BE244F03EF4"	"Fahrbahn"	"1"^^xsd:integer

Showing 1 to 3 of 3 entries

Figure 1: Example SPARQL Query for getting all groups of properties with their `guid`, `name` and `version` and the corresponding results from the database

transformation. Out of this analysis three basic templates were created to transform plane literals, dependencies and lists of literals (cf. listing 1 and 2). The result of the concept's implementation is an XSLT stylesheet for transforming XML documents containing properties according to ISO 23386 into semantic graphs in RDF/XML format.

The feasibility of the XSLT stylesheet concerning the requirements is successfully demonstrated by querying a transformed data set (see figure 1). The assignment of queried properties to BIM Objects for describing and maintaining information about e.g. resource aspects can be done according to Zentgraf, Hagedorn, and König [3]. Although it was proven that the developed stylesheet is feasible for the desired use case, the limitations regarding the processing of empty lists and the semantic stability must be improved in the future. An evaluation of the XSLT stylesheet with complex data from real worlds infrastructure projects is still pending and needs to be done in further research.

References

[1] ISO 23386, *Building information modelling and other digital processes used in construction*, Vernier, Geneva, 2020. (visited on 11/24/2021).

[2] P. Pauwels, A. Costin, and M. H. Rasmussen, "Knowledge Graphs and Linked Data for the Built Environment", in *Industry 4.0 for the Built Environment*, Cham: Springer International and Springer International Publishing, 2022, pp. 157–183, ISBN: 978-3-030-82430-3. DOI: \url{10.1007/978-3-030-82430-3_7}.

[3] S. Zentgraf, P. Hagedorn, and M. König, “Multi-requirements ontology engineering for automated processing of document-based building codes to linked building data properties”, in *Proceedings 39th CIB W78 Conference 2022*, ser. CIB W78 Proceedings, 2022. DOI: accepted.

[4] Y. Alani, N. Dawood, S. Rodriguez, and H. Dawood, “Whole life cycle construction information flow using semantic web technologies: A case for infrastructure projects”, in *Proceedings 37th CIB W78 Conference 2020*, ser. CIB W78 Proceedings, 2020, pp. 141–155. DOI: 10.46421/2706-6568.37.2020.paper011.

[5] buildingSMART, *Bsdd buildingsmart data dictionary (bsdd)*, buildingSMART, editor, 2021. [Online]. Available: https://github.com/buildingSMART/bSDD (visited on 11/29/2021).

[6] L. R. Quinn, *Xml essentials*, 2015. [Online]. Available: https://www.w3.org/standards/xml/core.html.

[7] Kay, Michael, *XSL Transformations (XSLT) Version 3.0 W3C Recommendation*, 2012. [Online]. Available: https://www.w3.org/TR/2017/REC-xslt-30-20170608/ (visited on 11/14/2021).

[8] Clark, James and DeRose, Steve, *XML Path Language (XPath) Version 1.0*, 2016. [Online]. Available: https://www.w3.org/TR/1999/REC-xpath-19991116/ (visited on 11/13/2021).

[9] Hazaël-Massieux, Dominique and Connolly, Dan, *Gleaning Resource Descriptions from Dialects of Languages (GRDDL) W3C Recommendation*, 2007. [Online]. Available: https://www.w3.org/TeamSubmission/2005/SUBM-grddl-20050516/ (visited on 11/17/2021).

[10] H. Bohring and S. Auer, “Mapping xml to owl ontologies”, in *Marktplatz Internet: Von e-Learning bis e-Payment, 13. Leipziger Informatik-Tage (LIT 2005)*, K. P. Jantke, K.-P. Fähnrich, and W. S. Wittig, editors, Bonn: Gesellschaft für Informatik e. V., 2015, pp. 147–156.

[11] W. Akhtar, J. Kopecký, T. Krennwallner, and A. Polleres, “Xsparql: Traveling between the xml and rdf worlds – and avoiding the xslt pilgrimage”, in *The Semantic Web: Research and Applications*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 432–447, ISBN: 978-3-540-68234-9.

[12] S. Bischof, S. Decker, T. Krennwallner, N. Lopes, and A. Polleres, “Mapping between rdf and xml with xsparql”, *Journal on Data Semantics*, vol. 1, no. 3, pp. 147–185, 2012.

[13] G. Sherman, “A critical analysis of xslt technology for xml transformation”, Citeseer, Tech. Rep., 2007.