CONTEXT CAPTURING OF MULTI INFORMATION RESOURCES FOR THE DATA EXCHANGE IN COLLABORATIVE PROJECT ENVIRONMENTS

Raimar J. Scherer, Peter Katranuschkov,
Institute of Construction Informatics, TU Dresden, Germany
Raimar.Scherer@tu-dresden.de, Peter.Katranuschkov@tu-dresden.de

Abstract

This paper introduces a framework for context capturing of multi information resources and is based on the three major approaches which have been developed over the last decade. Having similar goals in mind they have come up with partially different objectives, methods and representation technologies. The three are, respectively, the Multimodel (MM) approach of the Mefisto project in Germany, the COINS approach in the Netherlands of linked documents and models and the Linked Building Data (LBD) approach of the Linked Data Group in Europe. They were developed in recent years to tackle the central problems of management and exchange of various types of data in a single package expressing a certain context and goal in an overall digital AEC environment. The differences and the commonalities of the first two approaches are addressed and the more future oriented LBD approach is compared to them. The related upcoming standard ISO 21597, pursuing a merging of the first two approaches in its Part 1, is discussed, whereas the third approach, targeting reasoning, will be a subject of Part 2 that is still under discussion.

Introduction

With the growing use of BIM in collaborative design, various questions regarding AEC communication and coordinated collaboration arise concerning the management of the continuously growing amount of data and the varios data structures and formats that have come up in the last years. While BIM data finds a central data model in the Industry Foundation Classes, IFC (ISO 16739-1), there is still a need for a suitable solution for the management and exchange of distributed BIM and non-BIM data from different information sources for tasks that go beyond traditional BIM, and without preharmonizing them in one common data model. Such non-BIM data can be found in a variety of tasks from different domains such as energy-efficient design, fire safety, cost estimation, environmental engineering, construction management, facility management, etc.

Our focus is on capturing context, i.e. a context model, which formalizes the context and makes it exchangeable and sustainable, without altering the information resources and without coming up with one harmonized big data model, as various and repeated attempts have been made over the last 4 decades of digital data exchange. Our attention is mainly focused on link information, which means maintaining the process context. The different possibilities of experessing links and their semantic expressivness distinguish the three approaches and determine the level of the captured context.

Besides the context issue, the quality of the data is of interest. This covers not only the completeness of the data, on which quality is mainly focused, but includes also correctness and conflict free use in the context of the particularly addressed information environment. It cannot be formally achieved by IT tools but needs the active involvement of expert knowledge of engineers and architects. As it is long traditional practice, the responsible author from a certain AEC domain has to release and undersign her/his information for publishing legally, which means that this information is proven by the author as the domain expert and that s/he takes over the full responsibility for it. In former times such a legal release was done on paper, which is still common practice. However, on the paper only the domain information is communicated with some informal data from related domains indicating some basic interrelationships. For digital data an equivalent but improved procedure is needed. Each author is responsible for her/his information and for ensuring that this information is conflict free with regard to the related information of the other authors. This is necessary because, if not guaranteed, digital data may become corrupt, which may lead to severe consequences. It requires either powerful filter tools to extract from the big common data model only the domain specific data needed, e.g. based on many MVDs, or that the domain model stays separately managed by the responsible author and is only linked to other information and hence will be only accessed for common use on demand by reference or copy.

In this paper we focus specifically on the second idea. In the next sections we outline the three approaches for the management of interlinked multi information

 DOI:10.35490/EC3.2019.173

resources, compare them to one another and explain at length the upcoming ISO 21597 standard together with its two representational options (RDF/OWL and XML/XSD) and the developed tool enabling lossless data conversion between them. We also briefly present currently completed and ongoing implementations of the multi model concept and draw first conclusions from the work done so far.

The Three Approaches for Context Capturing of Multi Information Resources

The Multimodel Approach

The Multimodel (MM) approach provides a concept for context building in distributed information environments, which explicitly supports responsebility as it imposes that each model is separately stored and maintained, preferably by the related author. It was originally developed in the Mefisto project (Scherer et al., 2011) for linked building model data (MMC Version 1), where documents have been treated as a specialized subissue of a single item model. The concept is inspired by federated database technology. Its essence is in the connection of separate elementary models through ID-based links (Fuchs et al., 2011). The defined elementary models are not further spezialized in the scope of the actual application, e.g. a building model in IFC or the tender, assignment and invoicing (TAI) model in the German GAEB standard. They are independent models in their own domains and usually have their own processing tools. Links describe relations between individual elements of different models and are captured in separate Link Models. By interlinking the elementary models cross-model interrelationships can be represented, which enables operations and queries over the entire information space created through the achieved interrelationship of the models.

The Elementary Models and the Link Models of a MM are stored in a MM Container (MMC) for data exchange (see Figure 1). The links themselves are simple relations, expressed by the relating objects' IDs using XML/XSD representation, i.e. links do not express any additional meaning or transport additionnal information. Such information is foreseen to be attached to the Link Model to not overload the MMC by a mass of repeating information as the approach was designed for deep and intensive linking of documents and models in a specific process context. Hence, as many as necessary Link Models are allowed in one MMC to express the different semantics of the links. However, additional attribiutes can be added on each level, i.e. the MM, the Elementary Models and the Link Model level.

The MM approach addressed the problem of integration of heterogeneous data models. In it, different data models are kept in their original state and can be connected through separate Link Models. Thus, it provides an interoperability strategy with high generalitiy and modularity (Katranuschkov et al. 2014). A prerequisite, however, is the use of appropriate parsers for each of the data models used to enable uniform data access at the MM level.

The success of the Mefisto project resulted in the further development of the MM by buildingSMART as MMC Version 2 and the standardization of a BIM-LV-Container in the DIN SPEC 91350. It defines a practical data exchange of BIM and TAI data using the MM approach. Some minor adjustments have been undertaken to align the MMC to the ISO 21597 specification resulting in MMC Version 3.

The COINS Approach

In the Dutch COINS project a version of the LBD approach has been used to create a comprehensive information platform for data exchange and integration for the entire life cycle of buildings (Van Nederveen, Beheshti & Willems, 2010). The developed framework consists of a project specific BIM, the COINS Building Information Model (CBIM), and the corresponding working methods and processes summarized as the COINS Engineering Method (CEM). The CBIM model uses geometry as the central reference model, where building data is extended using ontological concepts and LBD.

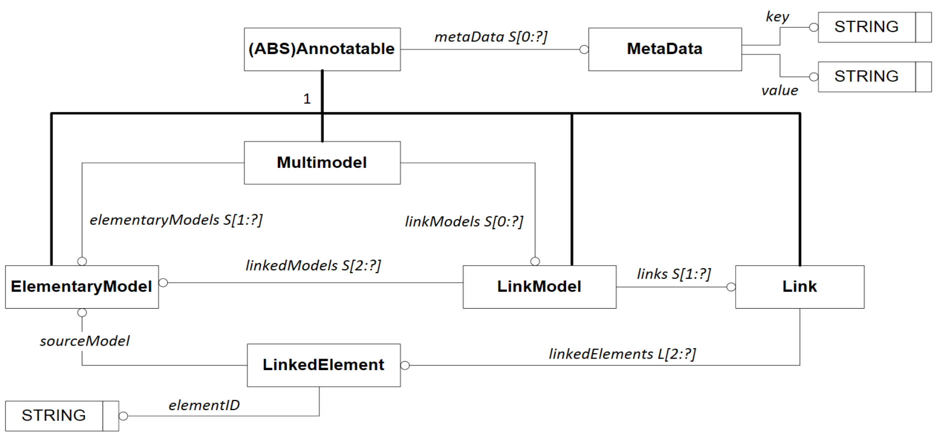

Figure 1: Express G Diagram of the Multimodel Elements

The Linked Building Data Approach

The principles of LBD stem in the Semantic Web, with the goal to connect AEC data semantically and make them machine interpretable throughout the Web. Accordingly, data is stored in triples utilizing the W3C Resource Description Framework, RDF. A triple consists of subject, predicate and object. It describes a directed relation between two entities, which are defined by a Uniform Resource Identifier (URI) or a relation between an entity and a literal value (data property). The triples form a directed graph, with subjects and objects as nodes and predicates as relations. The RDF graph can then be queried using the SPARQL Protocol and RDF Query Language. Further, a more extended vocabulary and rules can be built up, utilizing the Web Ontology Language, OWL. This in turn enables the use of powerful reasoning engines to infer implicit knowledge. Thus, in its essence the LBD approach is the application of the Linked Data concept of the Semantic Web in the AEC domain. It has been intensively investigated by many researchers in the last years (Pauwels et al., 2017).

A complementary step in the direction of LBD was the development of ifcOWL, which is an OWL representation of the IFC schema (Beetz et al. 2009). It enables the development of further ontologies such as the Building Topology Ontology (BOT), an ontology for building products (PRODUCT), or for related properties (PROPS), which were all initiated by the W3C LBD Community Group (Bonduel et. al., 2018).

Comparison of the Three Approaches

Comparison of the Multimodel and the COINS Approach

At the first glance both approaches appear very similar. However, the COINS approach was first of all directed to link docoments to models and hence a shallow non-intensive linking was in mind. Data overload from links had not been to be expected and therefore it was decided to reperesent a link as a self-standing information item defined in OWL/RDF with all related information, i.e. attributes attached. Different Link Models have not been seen necessary. In addition, in COINS there has to be a central model, which is the geometrical model, whereas in the MM approach there is no mandatory central model necessary, which gives more flexibility to align the MM to domain specific demands.

Comparison of the Multimodel and the Linked Building Data Approach

Both approaches follow the same concept, the creation of an information platform for coordinated and interoperable data exchange in projects. The goal is to unite separate information resources by linking them. In the MM approach, ID-based links are used to link the models. Thus, a link consists of a set of IDs referencing elements from the respective models. In the LBD approach, no separate link object is required for linking the information resources. A connection can be directly created by referencing through the use of URIs, which means, link information become part of the models.

A main difference of the MM approach is to consider already existing data models, e.g. IFC, CityGML, GAEB or proprietary ones originating from established domain authoring tools. For this reason, in the MM approach there are no restrictions on data format or data structure, except that elements must show a valid ID. In contrast, applying the LBD approach requires the use of RDF or standards based on it, such as the Web Ontology Language (OWL) as data format, which in turn also requires URIs for the identification of the elements. This imposed representation format for the links has also its advantages. While different parsers have to be used to integrate the heterogeneous data into the MM context, data integration with the LBD approach can be virtually automatic. However, in LBD all information links have to be preharmonized, i.e. they have to be completely transformed from their original format in an RFD/OWL representation. In the MM approach original formats can be kept untouchen and parsers or filters are needed to find the elements in the data models that should be linked via their element IDs.

There are also different principles regarding the persistence of the data. In the MM approach, the Elementary Models are static and therefore cannot be changed. In the LBD approach, on the other hand, such changes are possible. This difference is especially important when practical AEC scenarios are considered. For example, in the exchange of BIM and TAI data the unchangeability and the unambiguous assignment of responsibility to the data are mandatory in the legal sense. On the other hand, a more dynamic approach can be more useful in an open project environment where freely accessible data, e.g. weather data or building product catalogs, are important. However, even in open projects, working tohether in a collaborative team requires guaranteed information content as a main basis for collaboration. Information changes, from which new information may be automatically deduced without user notice and hence may remain unrecongnized, can lead to wrong decisions and can reduce or even destroy the benefits of open data access.

For the automated processing of cross-model links, machine interpretable semantics are necessary. The meaning of the links is provided by definition in the LBD approach. In addition, the LBD approach also enables mechanisms for the extension of semantics and data definition. Another advantage is the application of inference. It enables the generation of new knowledge through predicate logic, making implicitly available knowledge explicit and therefore accessible. In the MM approach, links have no directly embedded semantics that can be interpreted, but only an indirect one on the Link Model level. This semantic must first be deduced from Link Model metadata, which is a considerable technical difference.

The ICDD Standard

The upcoming ISO 21597 describes an Information Container for Data Delivery (ICDD), which mainly combines two functional approaches, i.e.: (1) the MM approach and (2) the LBD approach. The first enables structures and methods for interlinkage of heterogeneous data and their exchange between different applications as a closed solution of an engineering problem. The second enables an environment for integration of and reasoning on data on the Web, as well as dynamic semantic through ontology-based link model extensions. Both approaches have their preferred application fields in the AEC industry.

The ICDD standard was initiated by the COINS Project in 2016. In the very beginning, it was streamlined to the COINS approach and had little similarity to the MM approach. Thus, at first the goal was to develop a standard ontological platform that would enable models to be enriched with new data and to generate added value with the help of inference mechanisms, rules, etc. In the following two years both approaches developed towards a converging ICDD concept that adopted the virtues of each.

As a result, for the ICDD an ontological approach was chosen for the conception, i.e. the ICDD is described using RDF/OWL for the ICDD structure and the Link Models, but is not manadotry for the data models in part 1. The standard comprises two parts, as follows:

- Part 1 Container, in which the structure and the concepts of generic linkage are defined
- Part 2 Dynamic Semantics, in which the ontological concepts are introduced allowing extension of the basic Container structure described in Part 1.

An ICDD comprises the relevant linked models and documents in a Payload Document folder and the Linked Models in a Payload Triples folder. In addition, each container has an index file that specifies its particular content, which is a concept taken from the MM approach. The schemas for the index file as well as for the Link Models are also defined in Part 1 of the standard via the two ontologies Container.rdf and LinkSet.rdf (s. Figure 2).

The second part of the standard is mainly concerning the LBD approach. It introduces new and more complex concepts by extending Part 1 elements. This includes the extension of classes, properties and relationships as well as an extended concept for defining metadata within the ICDD. It allows the integration of application-oriented semantic into the generic information container and thus represents a generalized meta schema for the application-oriented MM in the sense of the German DIN SPEC 91350 standard.

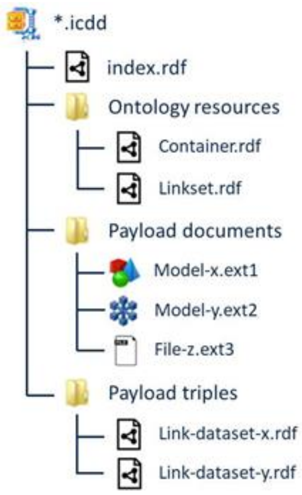

Figure 2: Structure of the ICDD in the official RDF/OWL representation according to ISO 21597

The ICDD XML Converter

Combining the MM and the LBD approach in ISO 21597 offers many advantages. The only restriction is that the LBD approach can only be used in RDF representation whereas the MM schemas are originally defined using XML/XSD. Therefore a converter was developed and is published via the ISO server to provide the use of both possibilities to end users and software developers in the AEC community, i.e. they are not forced to use the more complex RDF/OWL representation in case they are not intersted in benefitting from the Semantic Web technologies but prefer a lean data format.

ICDD Representation in XML

The ICDD XML is based on the MMC Version 2 (github of BuildingSMART) and is slightly modified as Version 3 to meet all ICCD requirements. The main part of the Container schema is depicted in Figure 3.

Compared to the RDF/OWL representation the resulting data structure is leaner and can thus be processed faster. This is important in cases where masses of links are needed. Furthermore, XML/XSD is better supported through a variety of tools and frameworks and is more widely spread in the industry today. The price to pay is relinquishment of the reasoning abilities and the less convenient and less flexible representation of Web resources.

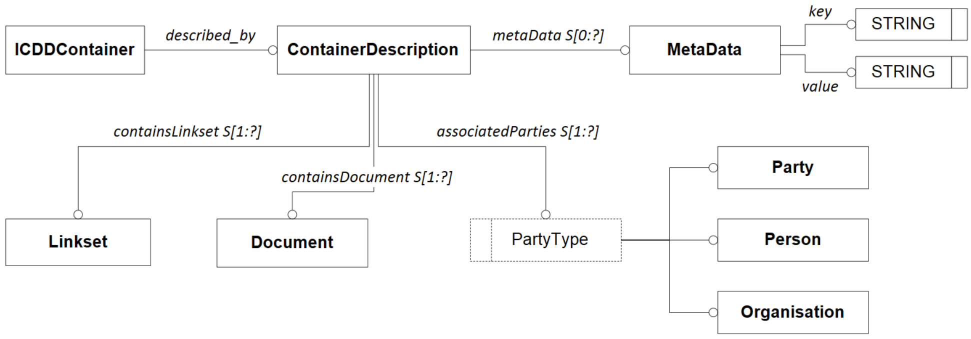

Figure 3: ICDD XML Container File Schema in Express G

The related ICCD structure of the Container is shown on Figure 4. The similarity to the RDF/OWL structure shown on Figure 2 is obvious. Software developers may choose ICDD XML for implementation when they do not have sufficient expertise in RDF or are not interested in the use of semantic web technologies, and would thus prefer a less complex, easier to understand and implement specification. ICDD XML may also be used in closed project environments where robustness and simplicity are preferred instead of the open Semantic Web approach where responsibility and data security can be an issue of greater concern.

On the other hand, software developers may choose ICDD RDF in project environments that require deeper semantics, dynamicity, and the reasoning capabilities offered by the RDF/OWL specification. Furthermore, ICDD RDF is of direct benefit if the required data sources are available as ontologies that can be integrated directly into the LBD approach.

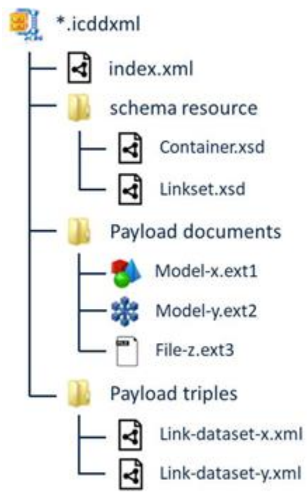

Figure 4: Container Structure of the ICDD XML

Conversion Method

A converter for lossless conversion between the two representation formats of ICDD was developed and published as part of ISO 21597 Part 1. It allows selecting the appropriate representation depending on the field of application and at the same provides for conformance with all ISO 21597 propositions.

For the transformation between the two ICDD variants it is only necessary to transform the Document Set description and the Link Models. For this purpose, the Extensible Stylesheet Language Transformation (XSLT) is used. The converter consists of the following four XSLT stylesheets:

- Document Set transformation from XML to RDF
- Document Set transformation from RDF to XML
- Link Model transformation from XML to RDF
- Link Model transformation from RDF to XML.

These stylesheets can be interpreted by a XSLT processor and create the file in the desired format from an outgoing file (see Figure 5). Subsequently, the transformed files can be replaced by their predecessor or can be stored redundantly in the ICDD Container. In the latter case, it must be ensured that both serializations are updated before each data exchange.

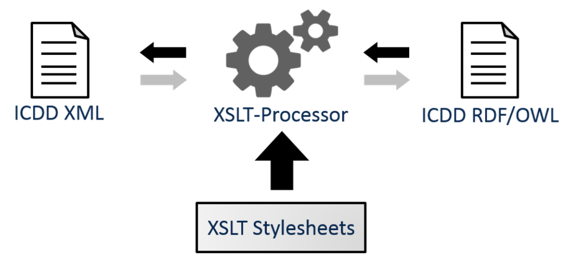

Figure 5: Conversion between ICDD XML and ICDD RDF/OWL through XSLT

Conversion of ICDD Containers

The developed XSLT stylesheets for transforming have been applied on several test cases, in which multiple documents and references stored in an ICDD Container have been linked via one ICDD Linkset. For convenience, a simple test case with one IFC model and one PDF document is shown here to demonstrate the transformation. The overall structure of the transformed Container and Linksets are identical, however small representational differences exist in the resulting ICDD XML. Listing 1 shows a data snippet of the Container consisting of one IFC model and one PDF document, which are both defined as internal documents. According to ISO 21597 each document definition is started via the tag ct:containsDocument followed by the document type (in this case ct:InternalDocument).

```
<rdf:RDF
…
 <ct:ContainerDescription rdf:about="">
  <ct:containsDocument>
   <ct:InternalDocument
   rdf:ID="id99090c14-2935-4c76-9545-8345d8861861">
    <ct:pending
     rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</ct:pending>
    <ct:filetype>ifc</ct:filetype>
    <ct:filename>48D-100.ifc</ct:filename>
    <ct:description>An IFC/BIM model</ct:description>
    <ct:name>48D-100.IFC</ct:name>
   </ct:InternalDocument>
  </ct:containsDocument>
…
 <ct:containsDocument>
  <ct:InternalDocument
  rdf:ID="id9b7145c0-123f-41f0-9801-82f03e96e523">
   <ct:priorVersion
   rdf:resource="#idcfd6eb56-a20f-40b4-bfbf-929bc88eea76"/>
   <ct:pending
   rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
   >false</ct:pending>
   <ct:format>application/pdf</ct:format>
   <ct:filetype>pdf</ct:filetype>
   <ct:filename>IDSV2.pdf</ct:filename>
   <ct:description>IDS Version 2</ct:description>
   <ct:name>IDSV2.pdf</ct:name>
  </ct:InternalDocument>
 </ct:containsDocument>
 …
<ct:conformanceIndicator>part1</ct:conformanceIndicator>
<owl:imports rdf:resource="/draft/Container.rdf"/>
…
```

Listing 1: ICDD Container defining an IFC model and a related PDF document

The first document in Listing 1 is the IFC model, named “An IFC/BIM model”, which has the status “pending”. The second document is the PDF document named “IDSV2.pdf”, which is a pending document that has a predecessor. Both documents are linked togetether on document level and not at element level, as it is current praxis in document management systems.

For a lossless XML conversion, the file must conform to Part 1 of ISO 21597. The Identifier (rdf:ID) and metadata such as filetype (ct:filetype), filename (ct:filename) or description (ct:description) are defined by using the corresponding data and object properties. Additionally, ct:conformanceIndicator shows which part of ISO 21597 is used. When converting the RDF file via a XSLT Processor using the developed XSLT Stylesheet, equivalent data in XSD representation format is generated, as shown in the following Listing 2.

```
<icdd:ICDDContainer
…
<icdd:ContainerDescription
 ID=""
 isExternal="true" >
 <icdd:containsDocument>
  <icdd:Document
   ID="id99090c14-2935-4c76-9545-8345d8861861"
   isExternal="false"
   description="An IFC/BIM model"
   name="48D-100.IFC" pending="true">
   <icdd:InternalDocumentProperty filename="48D-100.ifc"
   filetype="ifc"/>
  </icdd:Document>
…
  <icdd:Document
  ID="id9b7145c0-123f-41f0-9801-82f03e96e523"
  isExternal="false"
  priorVersion="#idcfd6eb56-a20f-40b4-bfbf-929bc88eea76"
  description="IDS Version 2"
  name="IDSV2.pdf" pending="false"
  format="application/pdf">
  <icdd:InternalDocumentProperty filename="IDSV2.pdf"
  filetype="pdf"/>
 </icdd:Document>
…
```

Listing 2: Snippet of a converted resulting ICDD XML

The resulting ICDD XML contains all RDF properties as XML attributes in a single XML element of the type icdd:Document, which results in a much shorter filesize. The document type (in this case icdd:InternalDocumentProperty) is defined in a separate element containing attributes that define the filename and the filetype. Thereby, an additional attribute named isExternal is inserted for each specified Document to enable differentiation between document or model data physically contained in an ICDD (isExternal="false" corresponding to a local identifier rdf:ID) and referenced via an URI (isExternal="true" corresponding to a global identifier rdf:about). The attribute is defined as optional and defaults to "false", i.e. internally contained data in the ICDD, which ist the normal case in practice.

When converting the ICDD XML back to the RDF format (see Listing 3), the test case data look similar to the original one shown in Listing 1. The small representational differences between the two listings are inessential and do not lead to data loss. They are mainly due to the more verbose document definitions in Listing 1, which are not mandatory and are therefore dropped in the translation process. Specifically, the document type is not defined by using the corresponding property, but is instead defined through rdf:type, which refers to the document class (in this case InternalDocument). Thus, despite using different ways for defining the Container, the original ICDD file as well as the converted RDF file are both valid ICDDs and no information loss occurs during the two-step conversion process.

```
<rdf:RDF
…
<ct:ContainerDescription rdf:about="">
<ct:containsDocument>
 <ct:Document
  rdf:ID="id99090c14-2935-4c76-9545-8345d8861861">
  <ct:description>An IFC/BIM model</ct:description>
  <ct:filename>48D-100.ifc</ct:filename>
  <ct:filetype>ifc</ct:filetype>
  <rdf:type rdf:resource="http://www.iso-
  icdd.org/draft/Container.rdf#InternalDocument"/>
 </ct:Document>
</ct:containsDocument>
…
<ct:containsDocument>
 <ct:Document
  rdf:ID="id9b7145c0-123f-41f0-9801-82f03e96e523">
  <ct:priorVersion
  rdf:resource="#idcfd6eb56-a20f-40b4-bfbf-929bc88eea76"/>
  <ct:description>IDS Version 2</ct:description>
  <ct:filename>IDSV2.pdf</ct:filename>
  <ct:filetype>pdf</ct:filetype>
  <rdf:type rdf:resource="http://www.iso-
  icdd.org/draft/Container.rdf#InternalDocument"/>
 </ct:Document>
</ct:containsDocument>
```

Listing 3 Snippet of the converted ICDD RDF back from the ICDD XML representation

Conversion of ICDD Linksets

Similar to the ICDD Container conversion, a XSLT stylesheet has been developed to map an embedded Linkset from RDF to XML. Listing 4 below shows a snippet of an original Linkset used in the testcase.

```
<rdf:RDF
…
 <owl:Ontology rdf:about="">
  <ct:creator
  rdf:resource="index.rdf#id08da3a34-1ad4-4cb0-aaa0-4d56e7079cf8"/>
  <owl:imports rdf:resource="index.rdf"/>
  <owl:imports rdf:resource="/draft/Linkset.rdf"/>
 </owl:Ontology>
…
 <ls:Link rdf:ID="19b09117a-1f9c-404f-8d0c-aac51a56149a">
  <ct:creator
  rdf:resource="index.rdf#id08da3a34-1ad4-4cb0-aaa0- 4d56e7079cf8"/>
   <ls:hasLinkElement>
    <ls:LinkElement rdf:ID="1eb0b1f73c-7db9-41e9-88c4-cfc58c02316b">
     <ct:creator
     rdf:resource="index.rdf#id08da3a34-1ad4-4cb0-aaa0-
4d56e7079cf8"/>
     <ls:hasDocument
     rdf:resource="index.rdf#id9b7145c0-123f-41f0-9801-
82f03e96e523"/>
    </ls:LinkElement>
   </ls:hasLinkElement>

   <ls:hasLinkElement>
    <ls:LinkElement rdf:ID="1e4023c2fd-4cf3-49e1-9ff9-b6229a0b80ab">
     <ct:creator
     rdf:resource="index.rdf#id08da3a34-1ad4-4cb0-aaa0-
4d56e7079cf8"/>
     <ls:hasDocument
     rdf:resource="index.rdf#id99090c14-2935-4c76-9545-
8345d8861861"/>
    </ls:LinkElement>
   </ls:hasLinkElement>
  </ls:Link>
…
```

Listing 4: Linkset defining a Link between an IFC model and a PDF document

The IFC Model and the PDF document defined in Listing 1 are assigned to two Link Elements (ls:LinkElement) which are linked together in one Link (ls:Link). To refer to these elements, the previously defined URIs are used.

Each Link Element refers to the corresponding document through the property ls:hasDocument, where the URI is referenced. Additionally, each Link Element and the Link itself can be referenced through an URI.

A snippet of the resulting XML data after converting the RDF file is shown in Listing 5. Similar to the XML version of the ICDD container, the ICDD Linkset uses the prefix “icdd:” for the definition of its contained elements.

```
<icdd:Linkset
…
<icdd:Link
 ID="19b09117a-1f9c-404f-8d0c-aac51a56149a"
 creator="index.rdf#id08da3a34-1ad4-4cb0-aaa0-4d56e7079cf8">
 <icdd:hasLinkElement>
  <icdd:LinkElement
   ID="1eb0b1f73c-7db9-41e9-88c4-cfc58c02316b"
   isExternal="false"
   creator="index.rdf#id08da3a34-1ad4-4cb0-aaa0-4d56e7079cf8"
   document="index.rdf#id9b7145c0-123f-41f0-9801-82f03e96e523"/>
 </icdd:hasLinkElement>
 <icdd:hasLinkElement>
  <icdd:LinkElement
   ID="1e4023c2fd-4cf3-49e1-9ff9-b6229a0b80ab"
   isExternal="false"
   creator="index.rdf#id08da3a34-1ad4-4cb0-aaa0-4d56e7079cf8"
   document="index.rdf#id99090c14-2935-4c76-9545-8345d8861861"/>
 </icdd:hasLinkElement>
</icdd:Link>
…
```

Listing 5: Converted XML version of the ICDD Linkset

From Listing 5 the main difference in formalizing links can be seen. In RDF the information of a link element has to be expressed in separate entities to enable distinguishing between a RDF subject and a RDF object. This discrimination does not exist in XML and hence all link information is attached to the link element as attributes. For intensive and deeply linked models this results in valuable reduction of data which can be up to dozens of Mbytes in many everyday scenarios.

When the XML file is transformed back to RDF, the result in the test case is similar to the data shown in Listing 4. It is possible that the structure of the transformed instances is in different order than in the original RDF but this has no implication on the content and does not mean any data loss. Here, again, the inserted additional attribute isExternal helps to distinguish correctly between local (internal) and global (external) identifiers.

Current Implementations

The presented approach has been already practically verified in several scenarios. The BIM-TAI integration based on DIN SPEC 91350 is currently implemented by about 20 software vendors in Germany.

In the Mefisto project, which dealt with the collaboration of building owners and contractors, BIM (represented in IFC), bills of quantities and task schedules (represented in XML) and work specifications (represented in the German standard GAEB) have been linked and exchanged (Scherer et al. 2011). In the EU project eeEmbedded, which dealt with energy-efficient building design by examining various parametric design variants using energy simulation, a climate model, an energy system model, occupancy models and a life cycle cost model have been used together with architectural and MEP BIM (Kadolsky & Scherer, 2016). In the German iSiGG project, which deals with the analysis of the effects of spreading fire and contagious gases and their interaction with the occupants, a numerical CFD and a crowd simulation are combined and performed as a co-simulation using dynamically changed BIM data along with various simulation-specific inputs and occupancy profiles (Scherer et al., 2018). Finally, in the German wiSIB project, which deals with the identification of the actual system performance of bridge structures subject to damages due to cracks, the BIM model of the bridge, the damage model, a monitoring model (sensor database) and a load model are combined using the developed MM framework. The available data of the container is then utilized by using an inference mechanism of the LBD approach to identify possible structural damages derived from the available data and from expert knowledge using an ontology knowledge base (Hamdan et al., 2019).

Conclusions

The forthcoming ISO 21597 standard, which describes the ICDD for data exchange of linked multiple data in one container. It combines concepts of LBD with those of the MM and the COINS approach. Although both approaches essentially follow the same ideas, i.e. (1) the definition of a model-level metadata set that allows the use of fully computer-interpretable MM containers, and (2) a linking mechanism introduced via one or more Link Models to represent inter-model relationships, they differ above all in the requirements placed on the models. Thus, the LBD approach requires the use of Semantic Web technologies such as RDF, OWL, SPARQL and pre-harmonized homogeneous data models. In contrast, the MM approach supports fully heterogeneous data models where data links are resolved with parsers and filter. An alternative formalization for immediate practical use was developed, the ICDD XML. It is conceived for pure MM application with advantages through its simplicity and robustness.

Acknowledgments

This research work was enabled through the support of the German Federal Ministry of Education and Research to the funding and performing of the projects mefisto (grant 01IA09001), wiSIB (grant 01|S16031C), and iSiGG (grant 01|S16009B), and the European Commission to the project eeEmbedded (FP7 grant 609349).

References

Beetz, J., van Leeuwen, J. P. & de Vries, B. (2009). IfcOWL: a case of transforming EXPRESS schemas into ontologies, *AI for Engineering Design, Analysis and Manufacturing* 23(1), 89-101.

Bonduel, M., Vergauwen, M., Klein, R., Rasmussen, M. H. & Pauwels, P. (2018). A novel workflow to combine BIM and Linked Data for existing buildings. In: Proceedings of *"eWork and eBusiness in Architecture, Engineering and Construction", (ECPPM 2018),* CRC Press.

DIN SPEC 91350 (2016). *Linked BIM data exchange comprising building information model and specified bill of quantities*, Beuth Verlag, Berlin.

Fuchs, S., Kadolsky, M. & Scherer, R. J. (2011). Formal Description of a Generic Multi-Model. In: Proceedings of *"Enabling Technologies: Infrastructure for Collaborative Enterprises", (WETICE2011)*, Paris, France.

Hamdan, A., Bonduel, M. & Scherer R. J. (2019). An ontological model for representation of damage to constructions. In: Proceedings of the *7th Workshop on Linked Data in Architecture and Construction,* Lisbon, Portugal.

ISO/NP 21597 (2019*). Information Container for Data Delivery (ICDD)*, International Organization for Standardization, Available at (req.login): https://www.iso.org/standard/71207.html?browse=tc; to be published 11/2019.

Kadolsky, M. & Scherer, R. J. (2016). Task-Specific Linking for Generating an eeBIM Model based on an Ontology Framework. In: Proc. of "*eWork and eBusiness in Architecture, Engineering and Construction" (ECPPM 2016)*, CRC Press.

Katranuschkov, P., Scherer, R. J., Weise, M. & Liebich, T. (2014). Extending BIM for energy simulation and design tasks. In: Proceedings of the *2014 International Conference on Computing in Civil and Building Engineering,* 625-632.

Pauwels, P., Zhang, S. & Lee, Y. C. (2017). Semantic web technologies in AEC industry: A literature overview, *Automation in Construction*, 73, pp. 145-165.

Scherer, R. J. & Schapke, S.-E. (2011). A distributed multi-model-based management information system for simulation and decision making on construction projects, *Advanced Engineering Informatics*, 25(4), Elsevier.

Van Nederveen, S., Beheshti, R. & Willems, P. (2010). Building information modelling in the Netherlands: A status report. In: Proceedings of the *18th CIB World Building Congress, W078 Special Track,* Salford, UK.