Linking BIM and GIS models in infrastructure by example of IFC and CityGML

S. Vilgertshofer[1], J. Amann[1], B. Willenborg[2], A. Borrmann[1] and T.H. Kolbe[2]

[1]Chair of Computational Modeling and Simulation, TUM Department of Civil, Geo and Environmental Engineering, Technical University of Munich, Arcisstr. 21, 80333 Munich; email: {simon.vilgertshofer | julian.amann | andre.borrmann}@tum.de
[2]Chair of Geoinformatics, TUM Department of Civil, Geo and Environmental Engineering, Technical University of Munich, Arcisstr. 21, 80333 Munich; email: {b.willenborg | thomas.kolbe}@tum.de

ABSTRACT

This paper demonstrates how Linked Data can be used to provide a semantically rich connection between the domains of Building Information Modeling (BIM) and Geographic Information Systems (GIS). Expertise from both domains is necessary for the planning of infrastructure projects such as roads, tunnels, bridges, and railways, as these projects usually require the consideration of widely diverging scales. As BIM and GIS rely on different types of data modeling standards, a conversion between the data models will inevitably result in data loss. To overcome this problem, we propose to utilize the concept of Linked Data which allows the original data to coexist and provide coherence by establishing references between the corresponding entities of both standards. The approach is illustrated using exemplary models of shield tunnels in both, the IFC and the CityGML format.

INTRODUCTION

Currently, some proposals for extending BIM concepts and standards for the infrastructure domain are being developed. buildingSMART is working on extending the Industry Foundation Classes (IFC) for representing infrastructure facilities, while in the GIS sector, standards such as InfraGML are developed, and the international standard CityGML is already well established.

At the moment, IFC is primarily focused on the buildings domain and comprises a wide range of classes which enable the exchange of detailed BIM models. However, the current version IFC4 Add2, which was released in 2016, is not well suited for modeling infrastructure facilities. Several proposals have been published such as IFC Tunnel, IFC Rail (bSI SPEC) and IFC Road (bSI SPEC) to resolve this limitation and corresponding standardization efforts are underway.

As the design and engineering of infrastructure projects require the involvement of experts not only from the building domain but also from the GIS domain, an approach for integrating the respective data sets with each other is required. As the main data standards of both worlds, namely IFC and CityGML, cannot be mapped onto one another in every aspect, an alternative approach is presented here. Instead of converting a set of information represented by one data model into the other one, we propose to utilize the concept of Linked Data (Bizer et al., 2009) to let the original data models coexist and provide coherence by establishing links between the corresponding entities and classes of both standards.

In this paper, an IFC-based data model for shield-tunnels is described and its possible interoperability with the GIS domain is demonstrated. In the first part, the integration of an IFC shield tunnel data model with IFC 4 and IFC Alignment is discussed. The chosen spatial structure and an approach for linear referencing by using IFC-Alignment are introduced. Furthermore, relevant parts of the CityGML data model will be briefly described regarding its applicability to shield tunnels. The second part of the paper describes a linked data approach that connects the BIM-based tunnel model with the corresponding CityGML GIS model. It is shown how a Linked Data based schema can be derived from the IFC tunnel data model and how it can be linked with a CityGML ontology. Thereby, so, the information contained in two model instances of the same shield tunnel (IFC and CityGML) is linked, and a query on combined CityGML and IFC data can be executed. In particular, it is possible to identify clashes of existing buildings/infrastructure with a planned tunnel or to query all signal lamps from the IFC tunnel data model using the corresponding CityGML element.

INFRASTRUCTURE MODELING IN CITYGML AND IFC

The CityGML standard is a widely accepted data model for the representation and exchange of 3D city and landscape models comprising information on geometry, topology, appearance, and semantics of the most common urban features.

The CityGML format is based on XML. It is an application schema of the Geography Markup Language version 3.1.1 (GML3) (Cox et al., 2004) which implements the ISO 19107 standard (Herring, 2001). Its latest issue, version 2.0.0, was released in 2012 as an official standard of the Open Geospatial Consortium (OGC). The CityGML model covers much more than 3D visualization tasks. Additionally, the semantics, thematic properties, taxonomies, aggregations, and interrelations of city objects are represented. The city model objects are decomposed by logical criteria, which can be observed in the real world.

Multi-scale modeling is one of the core features of CityGML. The standard implements five well-defined Levels of Detail (LoD) to optimize city models for different application scenarios (Gröger et al., 2012; Kolbe, 2009).

In the following, the thematic extension module *Tunnel* introduced in the CityGML standard version 2.0.0 is described. According to the specification, CityGML employs a consistent LoD concept through all its thematic extension modules. Hence, from LoD1-3 only the outer shell, the tunnels' boundary surface with its surroundings, can be described. LoD4 introduces the modeling of the tunnel interior.

The central class of the tunnel module is *_AbstractTunnel*, which is refined by the classes *Tunnel* and *TunnelPart*. *Tunnel* is used to represent one homogeneous tunnel only. If a tunnel consists of semantically or structurally different segments such as sections for different driving lanes, an individual *TunnelPart* is used for each section. The different *TunnelParts* are aggregated in a *Tunnel* object.

On LoD1, a tunnel model only represents the tunnel volume without further semantic classification. Starting from LoD2, the outer tunnel shell can be semantically differentiated into surfaces with special functions like roof, wall, ground, closure, outer ceiling, and *OuterFloorSurfaces* by applying the concept of boundary surfaces.

LoD3 adds openings like doors and windows in boundary surfaces to the tunnel model. LoD4 finally, permits the modeling of the tunnel interior. The free space inside the tunnel is composed of *HollowSpaces*. Normally, a *HollowSpace* is represented by a *gml::Solid* geometry. However, the visible surface of a *HollowSpace* can be modeled as specialized boundary surfaces such as an interior wall, floor, ceiling or *ClosureSurfaces*. Non-movable interior tunnel installations with special semantic meaning like stairs, rails, radiators or pipes, that are permanently attached to the tunnel structure can be modeled as *IntTunnelInstallation* objects.

The IFC standard is a comprehensive data model for the exchange of information in the domain of building design, engineering, and construction. It is being developed by the international organization buildingSMART and aims at being a vendor-independent open data format to support the interoperability in the AEC industry. The IFC data model is defined by the data modeling language EXPRESS, which is a part of the ISO standard 10303 STEP (Laakso and Kiviniemi, 2012).

There is only little research that addresses the extension of the IFC model to enable the representation of shield tunnels. The first contributions date back to a study on the development of a shield tunnel product model (Yabuki et al., 2007). Under consideration of the preliminary work by Yabuki et al., a concept for a shield tunnel product model has been developed (Amann et al., 2013; Borrmann et al., 2014; Borrmann and Jubierre, 2013). This concept aims to fulfill the demands of data exchange in the context of the design and engineering of large infrastructure projects. It is not yet part of the current IFC standard, but its applicability is demonstrated by introducing a low-level extension that can be used with current IFC viewers (Vilgertshofer et al., 2016). An important aspect of this proposal is the integration of multi-scale modeling into an infrastructure product model, a concept well established in the GIS domain. In compliance with the IFC standard, the tunnel product model is based on a clear separation between semantic objects and the geometric representation of those objects. A key point is the association of each semantic entity with a particular LoD to achieve semantic-geometric coherence of the overall model.

We extended and updated this IFC Tunnel data model (IFC Tunnel without LoD) with the upcoming extension of IFC for infrastructure proposes by buildingSMART. In detail, we replaced the *IfcTunnelAxis* with *IfcAlignment*, which provides an *Axis* attribute element, which is of type *IfcBoundedCurve* [1]. The upcoming IFC4x1 standard will also provide an *IfcAlignmentCurve* element that can be used to describe vertical and horizontal alignment elements (Figure 1).

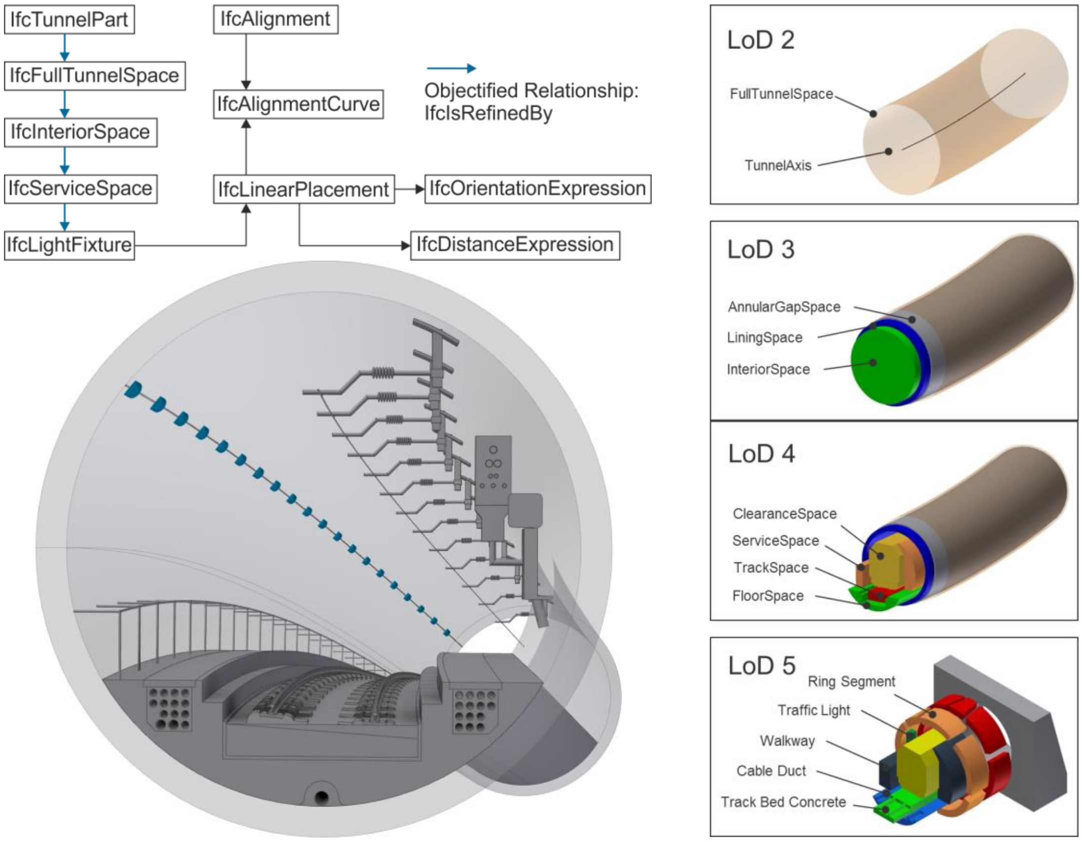


Figure 1: Placement of lighting objects in the visualization of an IFC tunnel model, 4 of the 5 LoDs described in the IFC Tunnel proposal (Borrmann et. al, 2014)

[1] http://www.buildingsmart-tech.org/ifc/review/IFC4x1/rc3-20161105/html/link/ifcalignmentcurve.htm

A LINKED DATA APPROACH FOR COMBINED QUERYING OF IFC AND CITY GML

Approaches for converting IFC models into CityGML and vice versa have been addressed by several research teams in recent years (Deng et al., 2016; El-Mekawy et al., 2011). A general problem is the fact that these two data models cannot be comprehensively mapped onto one another without data loss. This problem is caused by the very nature of the two standards: IFC aims at the detailed modeling of buildings for design, tendering, construction and facility management, whereas CityGML targets and covers a much wider field of representation in multiple scales.

There are several examples of this problem. When considering shield tunnels, IFC allows the modeling of particular physical objects to describe the various types of tunnel installations such as lighting, traffic lights, railings, drainage or cable ducts. As a corresponding semantic entity in CityGML only *IntTunnelInstallation* exists, which can be further classified by the attributes *class*, *function* and *usage*. However, if entities are semantically or spatially disaggregated in more detail, information is lost during conversion, as IFC provides a much finer granularity of the objects than CityGML. The result is that all the different semantic objects are described by the same coarse entity after a conversion. It is therefore not possible to query the CityGML model in regard to the specific entities that were available in the corresponding IFC model. On the other hand, IFC does not provide the possibility to model aspects such as terrain or land use.

Therefore, we propose not only to convert the data from one standard to the other but to enrich the conversion by using the method of Linked Data. Thereby, the connection of data from the two different sources allows us to execute queries concurrently accessing both data pools. Thereby, it is possible for an expert working with one data format in a respective application to access information that was previously lost in the conversion process.

PROOF OF CONCEPT: LINKED DATA FOR TUNNEL MODELS

The IFC Tunnel proposal (Borrmann et al., 2014; Vilgertshofer et al., 2016) was combined with the current ongoing work of IFC buildingSMART on IFC 4.1 in an EXPRESS schema (named IFC4x1_RC3_withTunnel.exp). The IFC Tunnel additions are shown in Table 1. In comparison with the previous proposal, the *IfcTunnelAxis* element has been refactored by an *Axis* attribute which is now located in the *IfcTunnelPart* element.

Table 1. IFC Tunnel extension

```
TYPE IfcTunnelSpaceEnum = ENUMERATION OF
         (FULLTUNNELSPACE, INTERIORSPACE,
         ANNULARGAPSPACE, LININGSPACE,
         CLEARANCESPACE, SERVICESPACE,
         FLOORSPACE, TRACKSPACE, RING,
         USERDEFINED, NOTDEFINED);
END_TYPE;
TYPE IfcTunnelInstallationEnum = ENUMERATION OF
         (WALKWAY, TRAFFICLIGHT, TRACKBEDCONCRETE,
         TRACKBEDRAILS, CABLEDUCT, DRAINAGE,
         USERDEFINED, NOTDEFINED);
END_TYPE;
ENTITY IfcTunnel
 SUBTYPE OF (IfcSpatialStructureElement);
         Identification : IfcIdentifier;
END_ENTITY;
ENTITY IfcTunnelPart
 SUBTYPE OF (IfcSpatialStructureElement);
         Identification : IfcIdentifier;
         Axis: IfcCurve;
END_ENTITY;
ENTITY IfcTunnelAxis
 SUBTYPE OF (IfcElement);
         PropertySet : OPTIONAL PropertySet;
END_ENTITY;
```

```
ENTITY IfcTunnelSpace
 SUBTYPE OF (IfcSpace);
         SpaceType : IfcTunnelSpaceEnum;
         Identification : IfcIdentifier;
         PropertySet : OPTIONAL PropertySet;
END_ENTITY;
ENTITY IfcTunnelElement
 SUBTYPE OF (IfcElement);
END_ENTITY;
ENTITY IfcRingSegment
 SUBTYPE OF (IfcTunnelElement);
END_ENTITY;
ENTITY IfcTunnnelInstallation
 SUBTYPE OF (IfcTunnelElement);
    TunnelInstallationType:
                   IfcTunnelInstallationEnum;
END_ENTITY;
ENTITY IfcRelIsRefinedBy
 SUBTYPE OF (IfcRelAggregates);
END_ENTITY;
ENTITY IfcLoD
 SUBTYPE OF (IfcRelAggregates);
         Level : IfcInteger;
END_ENTITY;
```

In the next step, this EXPRESS schema is translated to an OWL document according to the proposal described by (Pauwels and Terkaj, 2016). This ontology is identical to the ifcOWL ontology (IFC4) (Beetz et al., 2009) but adds tunnel and infrastructure elements. Figure 2 gives an overview of this conversion process. In the first step, a parser based on the lexer flex and

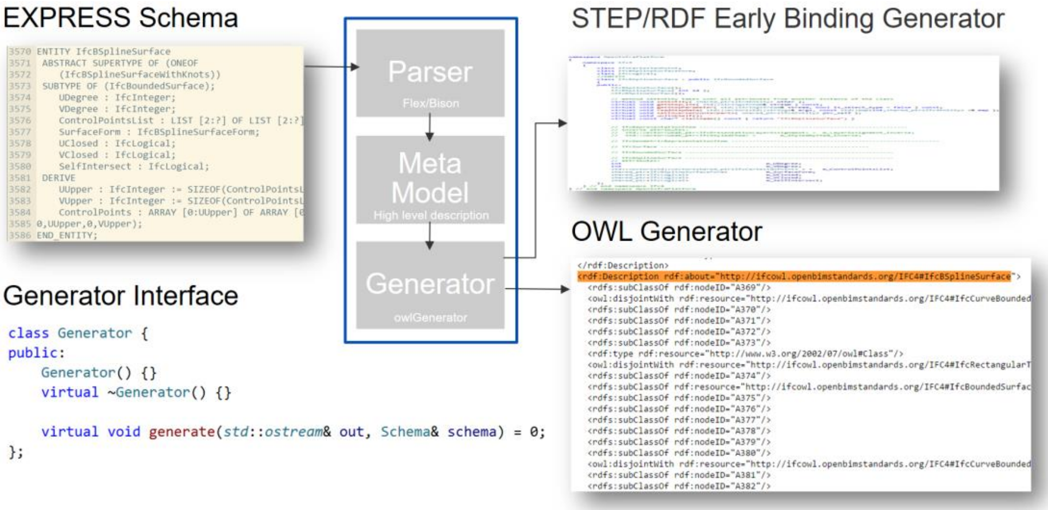


Figure 2. IFC Tunnel EXPRESS Schema conversion to OWL

the grammar parser Bison is used to read the EXPRESS schema and converts it to an intermediate representation (metamodel). The metamodel allows easy access from a programming language to the underlying classes and types of the EXPRESS schema.

Our prototypically implemented framework allows having different generators, which can access the metamodel. We implemented one for the generation of the OWL description (OWL Generator) and another one for generating an RDF early binding that is used to store IFC instances in the RDF format. Since there is also a STEP early binding generator, we can use this binding to load IFC data in the STEP format and store it as an RDF file and vice versa. Table 2 shows a very basic example of an RDF IFC instance file exported with the generated early binding generator for RDF.

Table 2. RDF IFC Instance file

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ifc="http://ifcTunnelOWL.cms.bgu.tum.de/IFC4_RC3_Tunnel#">

         <ifc:IfcApplication>
                  <ifc:applicationFullName>oipExpress Generator</ifc:applicationFullName>
         </ifc:IfcApplication>

         <ifc:IfcTunnel>...</ifc:IfcTunnel>
</rdf:RDF>
```

The same procedure is required on the CityGML side. First, the CityGML XML schema has to be converted to an OWL representation. A direct translation of the CityGML 2.0 schema to an OWL description (Métral et al., 2014) can be found at http://cui.unige.ch/isi/onto/, which we reused. Also, a CityGML 2.0 instance file has to be converted to the corresponding RDF representation.

In the last step, we tried to find similar concepts in both OWL descriptions of IFC Tunnel and CityGML. This was achieved by manually inspecting both schemas and creating an OWL (*SemanticInfrastructureLinkage.owl*) mapping that contains the similarities on the schema level of IFC Tunnel and CityGML. Furthermore, it defines a link element that is used later to make connections at the instance level. Table 3 shows a small excerpt of the created OWL mapping for semantic linking of infrastructure elements. The mapping on schema level is not required for the use case presented here but is needed for documentation and communication purposes.

Table 3. Mapping of concepts (*SemanticInfrastructureLinkage.owl* in RDF format)

```
...
<!-- schema level -->
    <owl:Class rdf:ID="TunnelPart">
        <owl:equivalentClass rdf:resource="http://www.opengis.net/citygml/tunnel/2.0/TunnelPart"/a>
        <owl:equivalentClass rdf:resource="http://ifcTunnelOWL.cms.bgu.tum.de/IFC4_RC3_Tunnel#IfcTunnelPart"/a>>
        ...
    </owl:Class>
...
<!-- types used at instance level -->
    <rdfs:Datatype rdf:about="http://SemanticInfrastructureLinkageOWL.cms.bgu.tum.de/IFC4_RC3_Tunnel#Link">
        ...
        <rdfs:label>Link</rdfs:label>
        <rdfs:comment>Element used to link infrastructure elements</rdfs:comment>
    </rdfs:Datatype>
</rdf:RDF>
```

Finally, we created an instance file that connects the corresponding entities of the IFC and GML instance files. This connection is achieved by mapping the *globalID* of a specific IFC element to the *gml:id* of the associated element in CityGML. For instance, we mapped the *IfcTunnelPart* element (id=*danmpWwv7wRgdJPvdkEy2*) to the CityGML entity *TunnelPart* (id=*GMLID_BUI180021_673_14143*) (see Table 4).

Table 4. Mapping of concepts (SemanticInfrastructureLinkage.rdf)

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:gml="http://ifcowl.openbimstandards.org/IFC4_ADD1#"
         xmlns:ifc="http://ifcTunnelOWL.cms.bgu.tum.de/IFC4_RC3_Tunnel#"
         xmlns:lnk="http://SemanticInfrastructureLinkageOWL.cms.bgu.tum.de/IFC4_RC3_Tunnel">

  <gml:GMLID_BUI180021_673_14143>
         <lnk:Link rdf:resource="ifc:danmpWwv7wRgdJPvdkEy2">
  <gml:GMLID_BUI180021_673_14143/>

  <ifc:danmpWwv7wRgdJPvdkEy2>
         <lnk:Link rdf:resource="gml:GMLID_BUI180021_673_14143">
  <ifc:danmpWwv7wRgdJPvdkEy2/>
  ...
</rdf:RDF>
```

A general overview that shows the structure of the different files involved in our proposed process is shown in Figure 3.

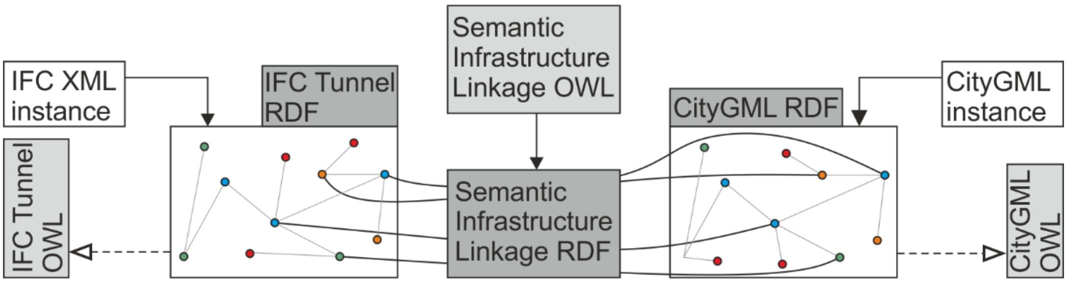


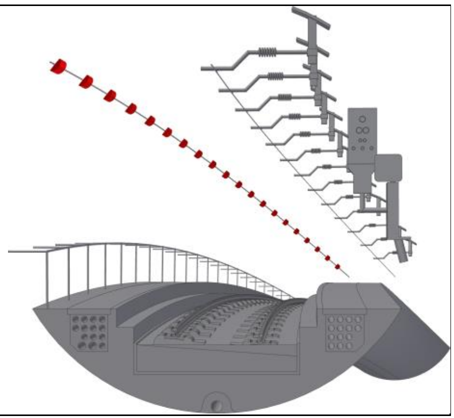
Figure 3. Structure of the Linkage concept and the generation the OWLs and RDFs

By using OWL as a common basis for the description of BIM and GIS related infrastructure elements, it is possible to use semantic web technologies such as the query language SPARQL. As a possible use case, this allows querying a CityGML model for all lighting fixture elements described in the related IFC tunnel model, which are not described directly by CityGML as it does not provide a particular class for it. The exemplary SPARQL query is shown in Table 5.

Table 5. SPARQL example query for tunnel lamps

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ifc:<http://www.cms.bgu.tum.de/schema/IFC4X1_RC3_Tunnel#>
PREFIX gml:<http://cui.unige.ch/isi/onto/citygml2.0.owl#>
PREFIX lnk:<http://www.cms.bgu.tum.de/schema/SemanticInfrastructureLinkage#>

SELECT ?gmlTunnelId ?ifcTunnelId ?ifcTunnelLightId
WHERE {
   ?gmlTunnel a gml:Tunnel.
   ?ifcTunnel a ifc:IfcTunnel.
   ?gmlTunnel gml:id ?gmlTunnelId.
   ?ifcTunnel ifc:globalId ?ifcTunnelId.
   ?gmlTunnelId lnk:Link ?ifcTunnelId.
   ?relCSS a ifc:IfcRelContainedInSpatialStructure.
   ?ifcTunnelLight a ifc:IfcTunnelLight.
   ?relCSS ifc:relatedElements ?ifcTunnelLight;
            ifc:relatingStructure ?ifcTunnel.
   ?ifcTunnelLight lnk:globalId ?ifcTunnelLightId.}
```



The proposed linkage approach may also be used to connect other BIM and GIS related formats such as InfraGML or national standards. For instance, the German OKSTRA standard (based on GML) provides a unique *gml:id* that enables the same linkage possibilities as described above. The approach relies on the existence of a unique id attribute in the standards that are linked. If no such attribute is available, a hash function can be employed to create a unique id for identifying a specific entity.

CONCLUSION

This paper discussed how the Linked Data approach could be applied to combine data models and instance data from the GIS and the BIM domain to provide comprehensive data analysis facilities. The concept was illustrated by the representation of a tunnel in both the CityGML and the IFC data model and their linkage by applying semantic web technology (OWL and RDF). The approach can be further improved by automating the mapping process through machine learning techniques.

Our main contribution is the presented approach of implementing the concept of Linked Data to connect the data in corresponding IFC and CityGML models. We showed how we semantically link those data models on instance and schema level as well as presenting an exemplary query. As mentioned above the approach may be extended to support other data models. Additionally, we integrated the IFC Tunnel schema with the latest infrastructure developments of buildingSMART and extended an early binding approach to generate RDF as well as STEP data.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support of the German Research Foundation (DFG) for funding the project under grant FOR 1546.

REFERENCES

Amann, J., Borrmann, A., Hegemann, F., Jubierre, J.R., Flurl, M., Koch, C., König, M., 2013. A refined product model for shield tunnels based on a generalized approach for alignment representation, in: Proc. of the ICCBEI 2013. Tokyo.

Bizer, C., Heath, T., Berners-Lee, T., 2009. Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems (IJSWIS) 5, 1–22. doi:10.4018/jswis.2009081901

Borrmann, A., Jubierre, J.R., 2013. A multi-scale tunnel product model providing coherent geometry and semantics, in: Proc. of the 2013 ASCE International Workshop on Computing in Civil Engineering. Los Angeles, pp. 291–298. doi:10.1061/9780784413029.037

Borrmann, A., Kolbe, T.H., Donaubauer, A., Steuer, H., Jubierre, J.R., Flurl, M., 2014. Multi-Scale Geometric-Semantic Modeling of Shield Tunnels for GIS and BIM Applications. Computer-Aided Civil and Infrastructure Engineering 30, 263–281. doi:10.1111/mice.12090

Cox, S., Daisey, P., Lake, R., Portele, C., Whiteside, A. (Eds.), 2004. OpenGIS Geography Markup Language (GML3), Implementation Specification Version 3.1.1.

Deng, Y., Cheng, J.C.P., Anumba, C., 2016. Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison. Automation in Construction 67, 1–21. doi:10.1016/j.autcon.2016.03.006

El-Mekawy, M., Östman, A., Shahzad, K., 2011. Towards interoperating CityGML and IFC building models: a unified model based approach, in: Advances in 3D Geo-Information Sciences. Springer, pp. 73–93.

Gröger, G., Kolbe, T.H., Nagel, C., Haefele, K.H., 2012. OGC City Geography Markup Language (CityGML) Encoding Standard.

Herring, J., 2001. OGC Abstract Specifications Topic 1 -- Feature Geometry (ISO 19107 Spatial Schema).

Kolbe, T.H., 2009. Representing and Exchanging 3D City Models with CityGML, in: Lee, J., Zlatanova, S. (Eds.), Proceedings of the 3rd International Workshop on 3D Geo-Information, Seoul, Korea, Lecture Notes in Geoinformation and Cartography. Springer Berlin Heidelberg. doi:10.1007/978-3-540-87395-2

Laakso, M., Kiviniemi, A., 2012. The IFC Standard - A Review of History, Development and Standardization. ITcon Journal of Information Technology in Construction 17, 134–161.

Métral, C., Ghoula, N., Silva, V., Falquet, G., 2014. Innovations in 3D Geo-Information Sciences, Innovations in 3D Geo-Information Sciences, Lecture Notes in Geoinformation and Cartography. Springer International Publishing, Cham. doi:10.1007/978-3-319-00515-7

Vilgertshofer, S., Jubierre, J.R., Borrmann, A., 2016. IfcTunnel - A proposal for a multi-scale extension of the IFC data model for shield tunnels under consideration of downward compatibility aspects, in: Proc. of the European Conference on Product and Process Modeling (ECPPM). Limassol, Cyprus.

Yabuki, N., Azumaya, Y., Akiyama, M., Kawanai, Y., Miya, T., 2007. Fundamental Study on Development of a Shield Tunnel Product Model. Journal of Civil Engineering Information Application Technology 16, 261–268.