LOSS- AND DISTORTION-FREE TRANSLATION BETWEEN DATA MODELS IN OPEN BIM

Galina Paskaleva[a], Manuel Wimmer[a], Thomas Bednar[b]

[a]E194-3 - Institute of Information Systems Engineering, Business Informatics Group, at TU Wien
[b]E206-2 – Institute of Building Construction and Technology, Research Center of Building Physics and Sound Protection at TU Wien

INTRODUCTION

Building Information Modelling (BIM) was developed as a method of information exchange in the Architecture, Engineering and Construction (AEC) and Facility Management (FM) industries. In its simplest form it is a data model that couples geometry with non-geometric information (e.g. material properties of a wall)[1]. However, BIM can also be regarded as a method of interaction between all stakeholders in the above mentioned industries, in which case it aims at not just information exchange but at providing feedback and enhancing real-time communication (e.g. collision detection between the ducts in a ventilation system and the pipes in a water supply system and automatic generation of wall penetrations for both). The complex interactions within the data model of BIM are software-dependent and not standardized. Some data models behind BIM – in particular the Industry Foundation Classes (IFC)[2] standard for Open BIM – are publicly available[2] and used as a data exchange standard between applications. The main challenge with the implementation of IFC is the loss and / or distortion of the information during the exchange process. One of the major reasons for it are translation problems between the internal data model of the application and the IFC data model, which in its IFC4 version contains more than 600 type and entity definitions. This work aims at defining a loss- and distortion-free translation method between data models as well as techniques for user control and verification of the translation results.

EXPERIMENTS / FUNDAMENTAL OF THE PROBLEM / EXAMINATIONS

There are two main challenges in designing a loss- and distortion-free translation between two data models – the lack of precision in the natural language and discrepancies in the level of detail between data models.

Behind each data model there is a taxonomy based on a natural language. For example, the term “local axes” can hold different meaning depending on the domain – for an architect it could be the local coordinate system for drawing a wall, whereas for the structural engineer it could be the load bearing axes. This can result in information distortion - the wall changing its position after translation from the architectural to the structural domain, because a local coordinate system can be placed anywhere, even outside of the object, but the structural axes are generally, but not always, placed symmetrically within the object. This challenge can be met by providing user control of the translation or by replacing the natural language term by a data model which encodes the dependencies described above in a graph-like structure.

This leads us directly to the second challenge – the level of detail. Each application requires a level of detail (or information granularity) in its data model that is adequate to its function. For a software in the architectural domain it is sufficient to represent a wall construction as a sequence of material layers. A software in the building physics domain, however, may require further subdivision of the material layers if they are inhomogeneous or if, for example, a simulation of the heat flux within the wall is to be performed. This becomes problematic if the translation path leads from the more detailed model (building physics) to the less detailed (architecture) to the more detailed model (building physics) again. To prevent information loss we need a data model of the translation process itself, which defines transition rules between, for example, the grid size of the battens layer of a roof construction in the architectural domain and the individual geometry and placement of each batten in the building physics domain.

RESULTS AND DISCUSSION

The translation methods described above present their own challenges – mainly an increased complexity in all data models involved. A major example of that is the IFC data model which incorporates structures for all domains in the AEC and FM industries. As of version IFC4 it no longer contains the parameters of its own data model (e.g. the U-value for a wall construction), but instead requires a dedicated parameter data base that can be queried by elements of the data model (e.g. the wall construction can look for its U-value). This development shows that the complexity of a data model can be reduced by separating it into groups or layers depending on its level of detail and specificity.

The methods of Model Driven Engineering (MDE)[3] and Multi-Level Modelling (MLM)[4] are particularly well suited to this challenge. They enable a formally defined separation of a data model into an arbitrary number of data model layers, in which each element serves as the type for multiple elements in the layer below[5]. In this way it defines the type and format of information they are allowed to carry and provides automatic verification of those elements in a translation context.

CONCLUSION

Reduction of complexity and control of the quality of information flow are easier to achieve in a highly structured environment (e.g. a multi-layered data model with few elements per layer) as opposed to a flat environment (a single-layered data model containing all elements).

The aim of this work is to provide guidelines for discovering imprecise definitions in data exchange standards and replacing them with data models, to develop algorithms for separation of large and complex data models into multiple data model layers and for translating between such multi-layered data models. The IFC standard, Drawing eXchange Format (DXF)[6] and a simple data model developed in a previous project, Simultan[7], will be used as case studies.

REFERENCES

[1] Laakso, M., Kiviniemi, A.: “The IFC Standard - A Review of History, Development, and Standardization” in Information Technology in Construction. ITcon, vol. 17, pp. 134-161, 2012.

[2] http://www.buildingsmart-tech.org/specifications/ifc-releases/ifc4-release

[3] Brambilla, M., Cabot, J., Wimmer, M.: “Model-Driven Software Engineering in Practice”, Morgan & Claypool, USA, 2012.

[4] Atkinson, C., Gerbig, R., Kühne, K.: "Comparing multi-level modeling approaches" in Proceedings of the 1st Workshop on MLM at the 17th ACM/IEEE Internat. Conference MODELS 2014, Vol-1286, pp. 53-61, 2014.

[5] Atkinson, C., Kühne, T.: “Reducing accidental complexity in domain models” in Software & Systems Modeling, vol.7, No. 3, pp. 345-359, 2008.

[6] https://www.autodesk.com/techpubs/autocad/acad2000/dxf/

[7] https://nachhaltigwirtschaften.at/de/sdz/projekte/simultan-simultane-planungsumgebung-fuer-gebaeudecluster-in-resilienten-ressourcen-und-hoechst-energieeffizienten-stadtteilen.php