TUM Open Infra Platform: an open source package for simultaneous viewing and analysis of digital models in the civil engineering domain

Štefan Jaud[*1], **Helge Hecht**[1], **Jonas Schlenger**[1], **and Julian Amann**[1]

1 Chair for Computational Modelling and Simulation, Technical University of Munich

Summary

The Architecture, Engineering and Construction (AEC) is in its transition from 2D design processes to 3D object-oriented modelling. Building Information Modeling (BIM) is steadily gaining importance replacing the conventional Computer-Aided Design (CAD) practices and getting implemented in every aspect of the very complex software and stakeholder landscape (Jaud et al., 2019). As one of the main principles, BIM describes the idea of integrating all information relevant to the life cycle of a structure, such as a tunnel, bridge, building or road, in a digital (BIM) model. The digital model is to ensure, among other things (Amann, 2018):

- that all relevant data is available to all project participants;
- that all data is in a consistent state (data integrity should be guaranteed);
- that the data can be used efficiently.

TUM Open Infra Platform (OIP) is an open source application for viewing and analysis of different BIM models used in the civil engineering field. OIP supports reading, visualization, navigating, and handling of:

- Industry Foundation Classes (IFC) models as specified in ISO 16739 (ISO, 2018);
- Point Cloud Data (PCD) models as supported by the Cloud Compare library (*Cloud-Compare*, 2016).

Multiple models can be loaded at once and compared between each other (see Figure 1). Their absolute position is accounted for, so the models can be checked against one another for internal differences based on location of elements.

Additionally, OIP incorporates its own EXPRESS parser, that consumes and evaluates data models (like IFC) specified with a schema following the ISO (2004) (see Figure 2). This enables automatic code generation for IFC early-binding library (Amann, 2018; Amann et al., 2018; Hecht & Jaud, 2019). As such, complete contents of IFC files produced following the ISO (2016) can be interpreted and analysed. Moreover, type safety is guaranteed at compile time thus reducing the risk for bugs and errors. This is achieved by a carefully designed schema-agnostic template library.

Let it be stated here that OIP serves as a prototypical playground for developments. The software architecture, features, and functionalities have been changed, added or removed as required along the way. Nowadays, OIP uses the IFC schema and CloudCompare's model

---

*corresponding author

36 as independent internal data models – explained in detail by Hecht & Jaud (2019). Major
37 elements from previous developments were already realigned, while some (previously available)
38 functionality is still considered as work-in-progress.
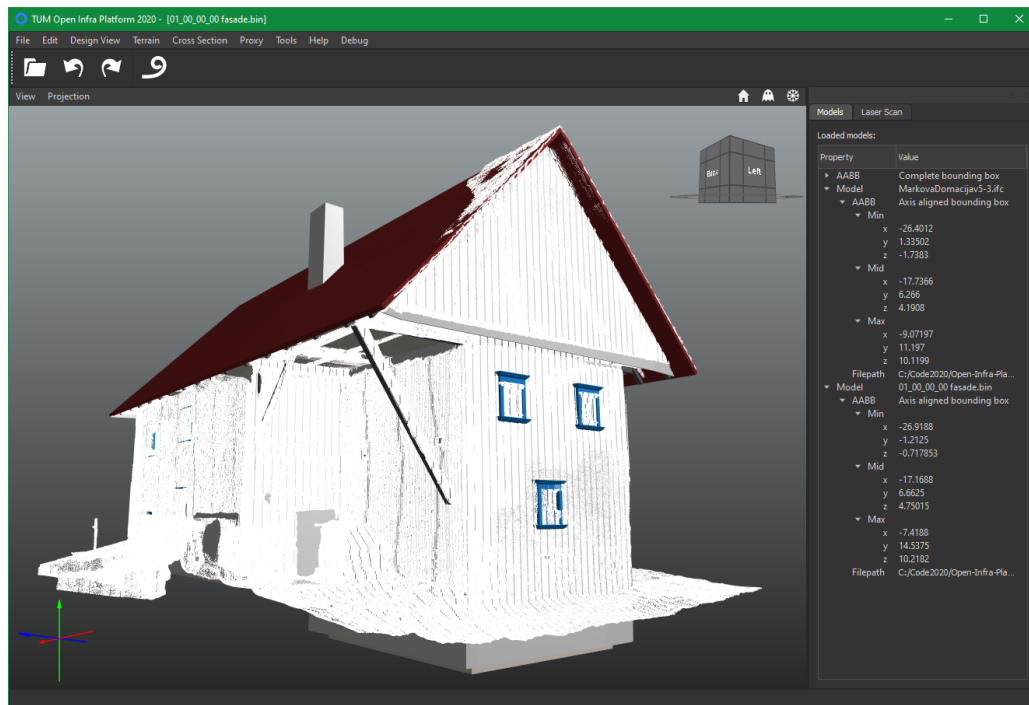
Figure 1: A PCD model together with an IFC model loaded in OIP. (PCD model courtesy of Grega Indof, IFC model courtesy of Laurens Oostwegel.)
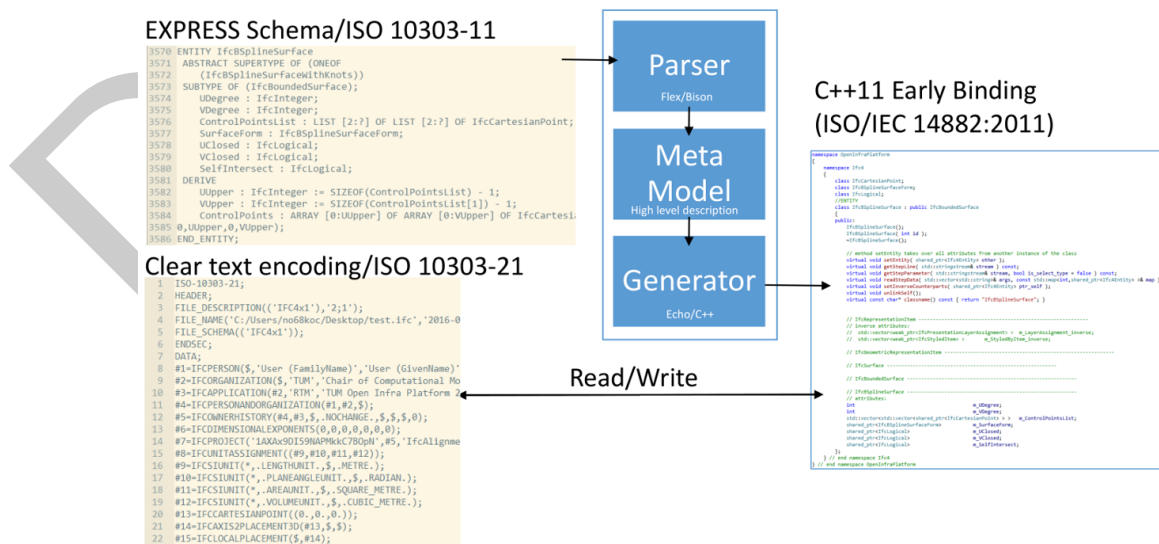
Figure 2: An overview over the EXPRESS parser's architecture (Amann, 2018).

Statement of Need

During the development of the IFC standard, a clear need for an independent software capable of consuming and producing IFC files according to the newly developed schemas was determined. Additionally, an interface to other infrastructure data model standards like OKSTRA and LandInfra was needed (Amann et al., 2015; Amann & Borrmann, 2015a; BASt, 2021; OGC, 2016). OIP fulfils this role by being open source, using CMake as a build system generator to support various platforms, and being based on well known libraries (like Qt, Eigen, Carve and Boost). Additionally, the EXPRESS parser provided proved useful for automatically producing source code and data models from newly developed schemas as needed (Amann, 2018; Vilgertshofer et al., 2017).

There are similar open source libraries and viewers available out there. For a matter of fact, the geometry handling of IFC content has been taken and improved upon from the versatile *IFC++* library (*IFC++*, 2021). Note that this is not an exhaustive list – a more thorough comparison has been conducted by Hecht & Jaud (2019) and Valero et al. (2020).

On the one hand, *IfcOpenShell*, *IFC++*, *IFC.js* and *XBim toolkit* focus primarily on handling IFC content and geometries (*IFC.js*, 2021; *IFC++*, 2021; *IfcOpenShell*, 2021; *Xbim Toolkit*, 2021). They provide an API to implement against (in various programming languages) together with a viewer. However, they focus mainly on the stable versions of the IFC standard (like IFC2x3 and IFC4), while OIP focuses primarily on the newer developments. Additionally, OIP supports DirectX versions 11 and 12 simultaneously through the use of BlueFramework library.

On the other hand, *ParaView*, *Point Cloud Library* and *CloudCompare* focus on handling PCD (*CloudCompare*, 2016; *ParaView*, 2021; *Point Cloud Library*, 2021). These provide more functionality for PCD analysis than OIP, however cannot handle BIM models. With the emergence of Scan2BIM and ScanVsBIM research field (deriving from or merely comparing PCD and BIM models), a tool that supports both PCD and IFC data is a welcomed addition to the research processes. Valero et al. (2020) selected OIP among many other software solutions as best suitable for the development of a Scan+BIM platform.

Recently, Blender received a *BlenderBIM Add-on* and can now support both IFC and PCD models, using *IfcOpenShell* and *ParaView*as supporting libraries, respectively (*BlenderBIM Add-on*, 2021). The add-on currently supports only the IFC4 version of the IFC data model, which is a major drawback if one wishes to compare PCD and IFC data of infrastructure objects. OIP bridges this gap by providing a viewer for both recent IFC versions and PCD (Valero et al., 2020).

Research Projects

The origins of the software date back to the first projects expanding IFC for infrastructure at the Technical University of Munich (Amann et al., 2014, 2015; Amann & Borrmann, 2015a, 2015b; Singer & Amann, 2014). There, first implementations of roads' geometric concepts and interfacing between multiple standards (like IFC, OKSTRA and LandXML) have been explored (see Figure 3).

Further on, many functionalities were added during the course of these projects:

- support for first experimental IFC schema additions for tunnels (Vilgertshofer et al., 2017);
- support for PCD and their analysis (Hecht, 2018);
- support for IFC Programming Language (IFC-PL) (Amann, 2018);
- support for linked data approaches (Beetz et al., 2019);
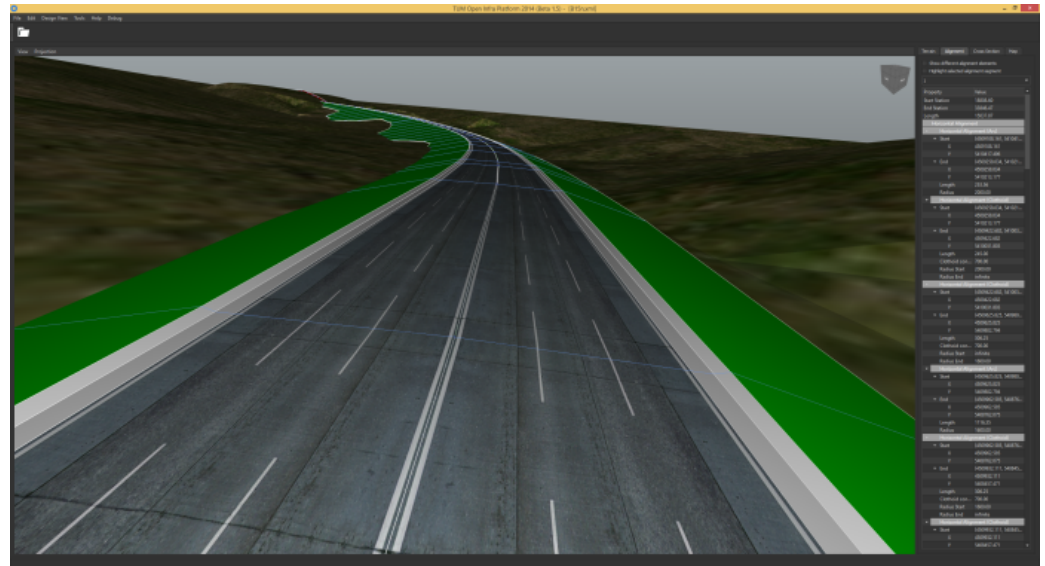
Figure 3: A perspective view of road geometry on terrain's surface with clearly marked fills.

Acknowledgements

References

Amann, J. (2018). *An object-oriented language for embedding interpretation semantics in digital building models* [PhD Dissertation]. Technische Universität München.

Amann, J., & Borrmann, A. (2015a). *Creating a 3D-BIM-compliant road design based on IFC alignment originating from an OKSTRA-accordant 2D road design using the TUM open infra platform and the OKSTRA class library* [Technical report]. Technische Universität München.

Amann, J., & Borrmann, A. (2015b). Open BIM for Infrastructure – mit OKSTRA und IFC Alignment zur internationalen Standardisierung des Datenaustauschs. *Tagungsband Zum 6. OKSTRA-Symposium*.

107 Amann, J., Flurl, M., Jubierre, J. R., & Borrmann, A. (2014, September). An alignment
108 meta-model for the comparison of alignment product models. *Proceedings of the 10th*
109 *European Conference on Product & Process Modelling.*

110 Amann, J., Preidel, C., Tauscher, E., & Borrmann, A. (2018). BIM programming. In A.
111 Borrmann, M. König, C. Koch, & J. Beetz (Eds.), *Building information modeling.* Springer.

112 Amann, J., Singer, D., & Borrmann, A. (2015). Extension of the upcoming IFC alignment
113 standard with cross sections for road design. *Proceedings of the ICCBEI 2015.*

114 BASt. (2021). *OKSTRA: Objektkatalog für das Straßen- und Verkehrswesen.* Bundesanstallt
115 für Straßenwesen. https://www.okstra.de/

116 Beetz, J., Amann, J., & Borrmann, A. (2019). *Linked Data – Analyse von Ein-*
117 *satzmöglichkeiten von verbundenen Informationen (Linked Data) und Ontologien und*
118 *damit befassten Technologien (Semantic Web) im Bereich des Straßenwesens* (pp. 82
119 pp.) [Technical report]. https://bast.opus.hbz-nrw.de/opus45-bast/frontdoor/deliver/
120 index/docId/2181/file/LinkedData_Schlussbericht_01.0195.pdf

121 *BlenderBIM add-on.* (2021). https://blenderbim.org/

122 *CloudCompare.* (2016). https://www.cloudcompare.org/doc

123 Hecht, H. (2018). *From Point Cloud Data to IfcAlignment* (pp. 26 pp.) [Technical report].
124 https://www.cms.bgu.tum.de/images/teaching/abim_seminar/Report_Hecht_From_
125 Point_Cloud_to_IfcAlignment.pdf

126 Hecht, H., & Jaud, Š. (2019). TUM OpenInfraPlatform: The open-source BIM visualisation
127 software. *Proceedings of the 31st Forum Bauinformatik.* https://publications.cms.bgu.
128 tum.de/2019_Hecht_Jaud_FBI.pdf

129 *IFC.js.* (2021). https://agviegas.github.io/IFC.js/docs

130 *IFC++.* (2021). https://ifcquery.com/

131 *IfcOpenShell.* (2021). https://ifcopenshell.github.io/docs/

132 ISO. (2016). *Industrial automation systems and integration – Product data representation*
133 *and exchange – Part 21: Implementation methods: Clear text encoding of the exchange*
134 *structure* (Vol. 2016) [Standard]. International Organization for Standardization. https:
135 //www.iso.org/standard/63141.html

136 ISO. (2004). *Industrial automation systems and integration – Product data representation*
137 *and exchange – Part 11: Description methods: The EXPRESS language reference manual*
138 (Vol. 2004) [Standard]. International Organization for Standardization. https://www.iso.
139 org/standard/38047.html

140 ISO. (2018). *Industry Foundation Classes (IFC) for data sharing in the construction and facility*
141 *management industries – Part 1: Data schema* (Vol. 2018) [Standard]. International
142 Organization for Standardization. https://www.iso.org/standard/70303.html

143 Jaud, Š., Donaubauer, A., & Borrmann, A. (2019). Georeferencing within IFC: A Novel
144 Approach for Infrastructure Objects. In Y. K. Cho, F. Leite, A. Behzadan, & C. Wang
145 (Eds.), *Computing in civil engineering 2019: Visualisation, information modelling, and*
146 *simulation* (pp. 377–384). American Society of Civil Engineers. https://doi.org/10.1061/
147 9780784482421

148 Jaud, Š., Esser, S., Muhič, S., & Borrmann, A. (2020). DEVELOPMENT OF IFC SCHEMA
149 FOR INFRASTRUCTURE. *Proceedings of 6th International Conference siBIM: Structured*
150 *Data Is the New Gold*, 27–35. https://publications.cms.bgu.tum.de/2020_Jaud_siBIM.
151 pdf

152 OGC. (2016). *OGC Land and Infrastructure Conceptual Model Standard (LandInfra)* (Vol.
153 2016) [Standard]. Open Geospatial Consortium Inc. https://www.ogc.org/standards/
154 infragml

155 *ParaView*. (2021). https://www.paraview.org/

156 *Point cloud library*. (2021). https://pointclouds.org/

157 Singer, D., & Amann, J. (2014). Erweiterung von IFC Alignment um Straßenquerschnitte.
158 *Proceedings of the 26th Forum Bauinformatik*.

159 Valero, E., Mohanty, D. D., & Bosche, F. (2020). Development of an open-source scan+BIM
160 platform. In "Furuya. "Osumi Hisashi" (Ed.), *Proceedings of the 37th international
161 symposium on automation and robotics in construction (ISARC)* (pp. 223–232). Inter-
162 national Association for Automation; Robotics in Construction (IAARC). https://doi.org/
163 10.22260/ISARC2020/0033

164 Vilgertshofer, S., Amann, J., Willenborg, B., Borrmann, A., & Kolbe, T. H. (2017). Linking
165 BIM and GIS models in infrastructure by example of IFC and CityGML. In *Computing in
166 civil engineering 2017* (pp. 133–140). https://doi.org/10.1061/9780784480823.017

167 *Xbim toolkit*. (2021). https://docs.xbim.net/