IFC-Based Product Modeling for Tunnel Boring Machines

F. Hegemann, K. Lehner & M. König
Chair of Computing in Engineering, Ruhr-Universität Bochum, Germany

ABSTRACT: The Industry Foundation Classes (IFC) standard has proven to be very successful in the exchange of data used in Building Information Modeling (BIM). Although the main focus of IFC is the definition of building entities, such as walls or columns, the IFC specification also includes more abstract classes to define relationships, properties, elements or resources. Based on the framework of classes as defined by current IFC standard, this paper explores the possibilities of extending the established IFC class hierarchy to additional engineering fields, in this case, mechanized tunneling and underground engineering. In particular, a set of new IFC-compatible classes are introduced to model tunnel boring machines (TBM). Additionally, a work flow is described to preprocess a TBM model designed in common CAD software in order to store it in a Product Server System. Finally, a case study of an underground engineering project is presented that uses the introduced IFC classes to model a TBM and employs a software tool to visualize IFC-based models.

1 INTRODUCTION

Due to the ongoing expansion of urban areas worldwide, solutions must be found to handle the increasing traffic in an efficient and environment friendly manner. One option is the extension of underground transportation systems using tunnels. In mechanized tunneling, tunnels are often constructed by large Tunnel Boring Machines (TBM). The risk involved in mechanized tunneling is related to various factors like settlements, gap grouting, face stability etc. All these factors are studied and simulated by separate research teams which focus on one scientific area. Since risk is a combination of the aforementioned factors, their interaction has to be considered while designing the tunnel. Therefore, the team members of these research teams should collaborate intensively to enable a successful tunneling project. The use of a product model as a tool will help in achieving an advantageous interaction.

2 MOTIVATION

In other civil engineering areas, for example residential building, bridge, and road construction, building information modeling (BIM) techniques are frequently used to support collaboration throughout the construction life-cycle. BIM is a valuable methodology to support communication, understanding, and visualization within the project team (Kymmell 2008). The most common format to model, exchange and share project-related BIM data between several applications is the IFC (Industry Foundation Classes) (BuildingSMART 2011). Although, the IFC-definition focuses on the building domain, other extensions are currently under development, for example IFC-Bridge and IFC-Road (Rebolj et al. 2008, Yabuki et al. 2006, Ji et al. 2011). In addition, a first approach was proposed for shielded tunnel structures (IFC-ShieldTunnel) (Yabuki 2008). Unfortunately, the established product models do not cover tunnel boring machines Detailed information about TBM are important for many project members. Examples include modeling the shield to plan the annular gap grouting, the locations of the hydraulic jacks for process simulations, and the backup train of the tunnel boring machine for the logistics design.

In this paper a detailed IFC product model for mechanized tunneling projects including the tunnel boring machine is presented. Depending on the conditions encountered in the ground, different types of tunnel boring machines are needed for the excavation. This paper will focus on Earth Pressure Balance (EPB) shield machines, which are frequently used for tunneling in unstable ground conditions. EPB shield machines consist of several machine components such as the cutting wheel, the excavation chamber, screw conveyors and others. Since the current IFC notation does not provide classes to capture these elements, one possibility would be to represent these elements by so called proxy classes. A proxy class can be understood as a container defined by associated properties (BuildingSMART 2011). Furthermore, a geometric representation can be assigned to it. Therefore, proxy classes can be used as substitutes for each element which is not captured by the IFC notation. Indeed, this could be an appropriate solution if only few undefined elements exist, but to represent a TBM many elements must be modeled. A huge number of proxy classes could lead to conflicts regarding element identification, since different team members could use different names for identical or similar elements. Additionally, individual aspects of a tunneling project, for example. the ground, the tunnel or the TBM should be integratable into one IFC file. If each element of each aspect is represented by seperate proxy class this can lead to great confusion. By using seperate models (e.g. a tunnel model, TBM model or ground data model) each aspect of tunneling can be easily identified. Therefore, the individual elements of a TBM were added to an IFC-based product model, in order to exchange important information easily and to provide this information to team members to support their planning tasks.

At this point it should be emphasized that, although the IFC standard has been extended by new IFC classes to model mechanized tunneling (TBM), it is not the aim of this research effort to standardize this extension and to include these specific classes into the current IFC notation. Rather, these classes have been defined and implemented to simplify their use and enable the exchange of data between software components (CAD programs, simulation packages, visualization components, etc.) Thus, the major aim of the proposal presented in this paper is to employ IFC technology to *exchange* (and not store or process) relevant underground engineering information. This is the strength of the current IFC standard, as witnessed by the strong support of major engineering software developers. That being said, should the proposed IFC classes establish themselves in the future as being well-suited to modeling underground engineering projects, the authors will not, of course, hinder the IFC community from adapting ideas taken from this proposal.

3 BACKGROUND

In mechanized tunneling there exist a huge number of individual tunnel boring machines. Each of them is designed for different soil conditions. In the field of unstable ground conditions shielded machines have proven their value, in particular, the Earth Pressure Balanced Shield Machine. By its shield, the EPB shield machine is able to withstand the forces of an unstable tunnel face und therefore can operate

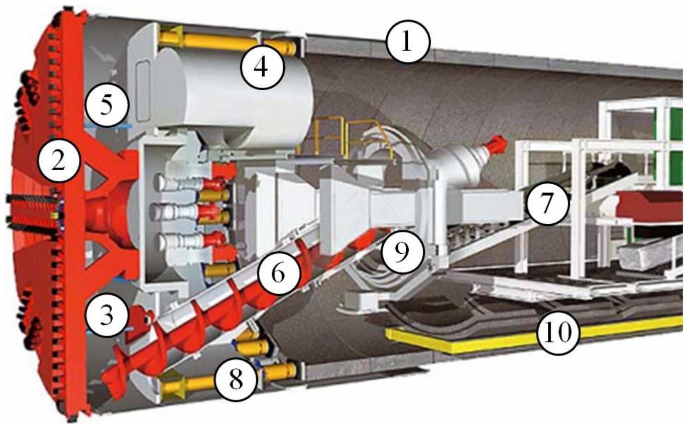

Figure 1. Abstract model of an EPB Shield Machine

in many different types of soil. It advances like every other shielded machine in a cyclic operation. Therefore, after the machine has advanced for a specific length it stops and installs segments of reinforced concrete assembled to form an enclosing ring to prevent the collapse of the overlying soil formation. This reveals two main processes of shielded machine, the soil excavation process for the machine advancement and the segment erection process to establish the tunnel (Rahm 2011).

During the soil excavation process several components work together. The basic components are illustrated in Figure 1. To support the EPB shield machine head from the surrounding earth pressure during the excavation process, it is surrounded by a thick shield (1) of steel. This shield is the reason why it is called “shield tunneling”. To excavate the ground in front of the EPB shield machine the cutting wheel (2) rotates and cuts the ground which is forced into the excavation chamber (3) through many holes in the cutting wheel. Inside the excavation chamber, pressure is created to compensate the pressure of the tunnel face and prevent fluids in front of the TBM from entering. Pressure in the excavation chamber can be built up because of a bulkhead (5) which separates the excavation chamber from the remaining TBM. Through a man lock (4) the staff can pass the bulkhead and enter the excavation chamber for maintenance issues. To remove the ground out of the excavation chamber a conditioning unit changes the incoming muck to a slurry through the injection of special fluids. A screw conveyor (6) takes away this slurry and puts it on a belt conveyor (7) which is positioned on the backup equipment that forwards the soil above ground.

To execute a forward motion and to bear the cutting wheel against the soil the EPB shield machine possesses a number of thrust jacks (8) (hydraulic cylinders). These jacks push against the already installed tunnel segments using them as counter bearings to move the TBM forward. The segments are set by a segment erector (9) which is installed at the end of the TBM head (segment erection process). A segment feeder (10), positioned on the backup equipment, supplies the erector with the necessary segments.

4 CONCEPT

In this section, an IFC-based model for TBM is introduced. The following approach is, however, just a first step of an evolving model and will be incrementally improved in ongoing research. Furthermore, as mentioned before, this paper focuses only on the modeling of an EPB shield machine.

The new TBM model is built upon the IFC standard. It extends the existing IFC structure by adding additional classes for modeling TBM and differentiates between spatial classes and element classes. While spatial classes define the different physical sections of a TBM, comparable to the *IfcBuilding* or *IfcBuildingStorey* classes, the element classes capture the specific components of a TBM such as the cutting wheel or the thrust jacks. To characterize the spatial sections the elements are assigned to their corresponding spatial section. These element classesare comparable to IFC classes such as *IfcWall* or *IfcColumn*. An overview of both spatial and element classes and their structure is illustrated using EXPRESS-G schemes shown in Figure 2 and 3. Figure 2 illustrates the overall structure of the model. It shows existing classes of the IFC standard (gray) which are adapted and the newly added classes. Figure 3 explicitly shows some of the newly developed element classes.

To represent the spatial structure of a soft rock compatible TBM, three classes were added: the *IfcTunnelBoringMachine*, the *IfcTbmHead* and the *IfcTbmBackupEquipment* class. Each class inherits from the IFC class *IfcSpatialStructureElement*. The TBM and the organization of the

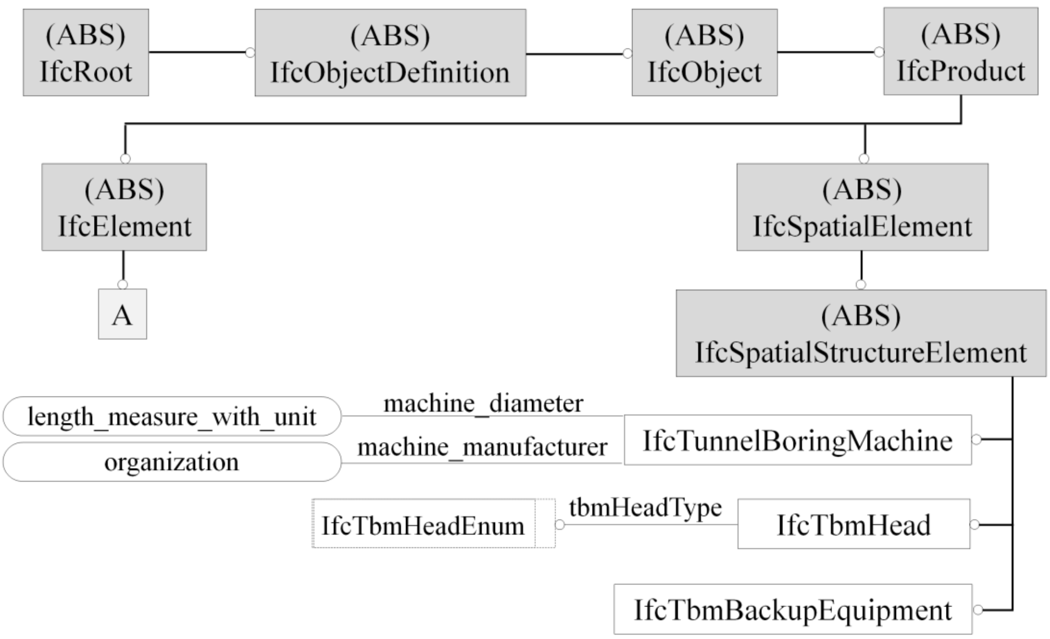

Figure 2. EXPRESS-G chart of the new TBM model

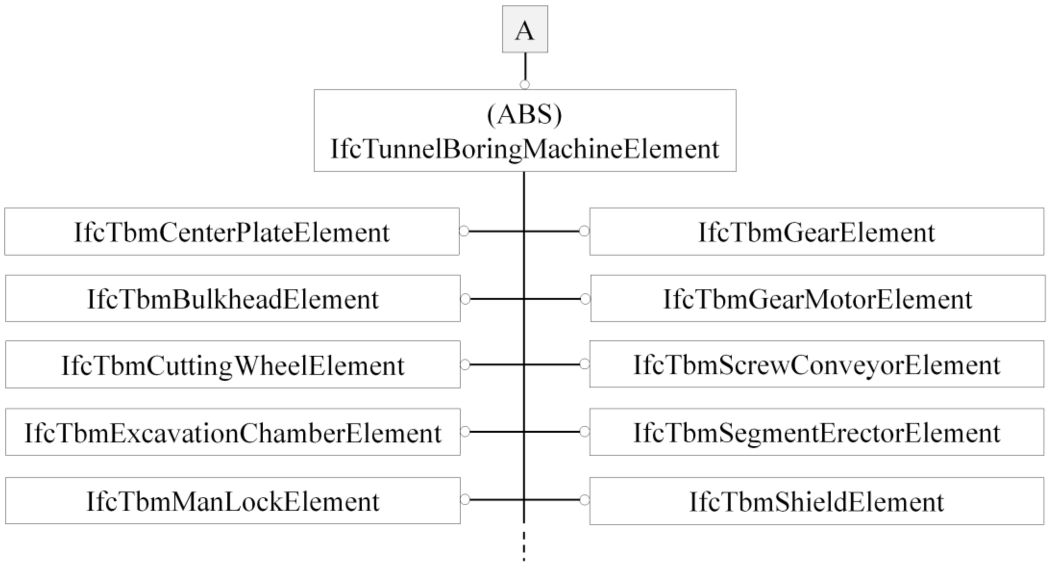

Figure 3. Extract of the new element classes of the TBM model

IfcTunnelBoringMachine class represents the whole TBM and captures the other two spatial classes *IfcTbmHead* and *IfcTbmBackup Equipment*. It contains two attributes, the excavation diameter of the manufacturer of the TBM. These attributes should help the user to get a first impression of the TBM.

In a next step, the spatial model is divided into the two spatial sections "TBM head" and "backup equipment", represented by the classes *IfcTbmHead* and *IfcTbmBackupEquipment*, respectively. The division is due to the different soil conditions found in tunneling projects with soft rock conditions: Whereas the TBM head varies from project to project, the backup equipment almost always remains the same except for the supply system for the supporting infrastructure elements. As a result, the TBM is designed in a way that the head can be easily exchanged in accordance to current soil conditions. This is why the *IfcTbmHead* class contains an attribute of the type *IfcTbmHeadType* to define the type of the TBM head. This type points to the enumeration *IfcTbmHeadEnum*, which includes various types of heads such as EPBSHIELD, SLURRY or MIXSHIELD.

The element classes are structured a little different compared to the spatial classes. An abstract super class *IfcTunnelBoringMachineElement* represents the parent of all element classes of the TBM model. Due to the fact that it inherits from the *IfcElement* class, each TBM element class can contain a geometric representation of its element. A geometric representation is needed for example by planners to identify spatial conflicts during the planning process. Additionally, due to the restricted amount of available space of a TBM, free areas can be spotted to store materials or tools. Furthermore, the geometric representation gives evidence on the dimension and the position of the specific element. Besides the geometric representation of an element there is also the meta information of an element. These information contain moreover numerical values in a key-value style (e.g. the torque of the cutting wheel). They are stored in *IfcPropertySet*s and attached to the element. The *IfcPropertySet* class is provided by the IFC standard and can contain a varying amount of values of various kinds (e.g. enumerations, lists or single values).

Some of the new element classes of the TBM model are shown in Figure 3. Due to the inherent complexity of current TBMs, the entire TBM model cannot be explained in one step. Thus, in the following the essential classes of the TBM are described with respect to their high significance in the excavation process.

One of the essential element classes is the *IfcTbmCuttingWheelElement*. It contains information on the cutting wheel of the TBM, including the excavation diameter and the shape which has a huge effect on the excavation rate (e.g. by the number of excavation tools on the disk or the number of openings to the excavation chamber). The excavation chamber is represented by the *IfcTbmExcavationChamberElement* class. This class has no geometric representation at the beginning because it is defined by its surrounding elements (e.g. bulkhead or shield). It is comparable to the *IfcSpace* class. Therefore, the geometric representation is created based on its surrounding components when it is requested. This is due to consistency reasons. Otherwise, a designer could change the position of a surrounding component but forget to adjust the shape of the excavation chamber. Nevertheless, it is an essential class to capture crucial meta information for individual process simulations, such as the total volume or the position of stators or rotors. The bulkhead component is represented by the *IfcTbmBulkheadElement* class. Its walk through the man lock is captured by the *IfcTbmManLockElement* class.

Responsible for the advancement of an EPB shield machine are the thrust jacks. Therefore, the new class *IfcTbmThrustJackElement* has been added. In the TBM model a thrust jack is defined as an element including several components. It consists of one or more (usually three) hydraulic cylinders and a stomp at the end which bears against the already installed segments. By this combination of components the machine is pushed forward through the tunnel by thrusting against the installed segments.

To execute the drilling motion of the cutting wheel several components have to play together. The most essential ones have been captured by IFC classes. One of these is the *IfcTbmCenterPlateElement* class. The center plate of a TBM consists of the gear of the cutting wheel and several connections leading into the excavation chamber. The gear is captured by the *IfcTbmGearElement* class and can be assigned to the *IfcTbmCenterPlateElement* class. To drive the gear several motors are installed in front of the center plate which are represented by the *IfcTbmGearMotorElement* classes.

Further important classes include the *IfcTbmScrewConveyorElement* and *IfcTbmSegmentErectorElement*. They also have an essential influence on the driving process, either by transporting soil out of the excavation chamber or installing supporting segments in the tunnel. To allow process simulation components to calculate accurate results additional meta information must be provided for these components. These are attached to the newly developed classes.

The last essential class of the TBM head is the *IfcTbmShieldElement* class. It represents the shield of an EPB shield machine which covers the TBM and the tunnel until the segments have been installed. Knowing the dimension of the shield is important to calculate the amount of mortar needed to fill the gap resulting by the shield. This is crucial to prevent the soil above the TBM settling down and creating large settlements on the surface.

5 IMPLEMENTATION

During the planning phase of a tunneling project, several team members often work together to create an efficient tunneling design. As a result, the accumulated data of the TBM model must always be up-to-date and readily available to all partners, for example by using an Internet-enabled database. Therefore, to realize this consistent storage of the new IFC-based TBM model specific preprocessing steps have to be executed. These steps are described in Figure 4.

In a first step the model of the TBM must be created. Nowadays, designing a TBM is done by using various modeling software. Specifically, common modeling software, including AutoCad (Byrnes 2011) and SolidWorks (Planchard et al. 2012), offer an IFC interface for importing and exporting internal models. However, they do not, of course, provide the newly developed classes of the TBM model introduced in this paper, but only stick to the standard IFC EXPRESS scheme. The IFC EXPRESS scheme defines the classes and functions which can be used in an IFC STEP file depending on the used version of IFC (e.g. version IFC2x3). Thus, in the case of an IFC export the elements of a TBM model can only be exported as IFC proxy classes. However, to be able to identify the type of an element, additional information has to be added to each element of the model using options available within the designer software. Due to the differentiation of the spatial sections of a TBM, two attributes must be attached in key-value style to an element. First of all the element itself has to be identified by adding the name of the IFC class (e.g. *IfcTbmCuttingWheelElement*). Secondly, the spatial section the element belongs to is determined by appending another attribute, such as *IfcTbmHead*. This way each element of the model can be assigned to its spatial section and be identified by its class name. Now, the newly created model is exported as an IFC STEP file based on the standard IFC EXPRESS scheme.

In the next step of the work flow, a newly developed conversion application processes the exported file by eliminating the IFC proxy classes and replacing them with the newly defined classes of the TBM model. Therefore, the conversion application consists of two parts, the import and export part. First the import part reads the exported IFC file, identifies the components by its attributes and stores the information for the export. Then, the export part is used to create a common IFC file readable by IFC compatible software knowing the extended IFC TBM EXPRESS scheme. It accesses the stored information of the import part and writes a new IFC STEP file. Therefore, the spatial sections of a TBM are processed one after another. More precisely, each element of a spatial section is added to the model together with its corresponding geometric representation and meta information. The meta information are attached by *IfcPropertySet*s. At the end the spatial section is created and the associated elements are linked to it.

In the last step of the work flow the created IFC STEP file containing the TBM model is uploaded to a Product Server System knowing the TBM extension. This Product Server System can be, for example, a BIMServer (Beetz et al 2010) or another IFC compatible Server System. The technological possibilities of storing BIM related data into an adequate server system is a complex challenge and is therefore not further discussed in this paper. Due to this server system the TBM model is readily available for all team members of a tunneling project and for all kinds of simulations requesting TBM data.

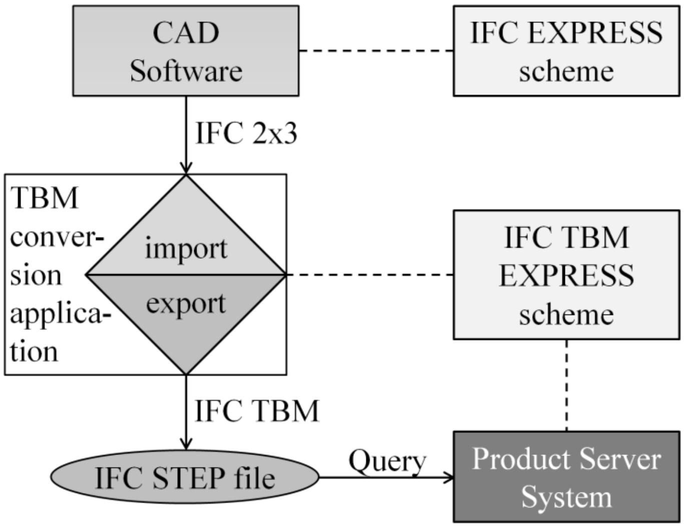

Figure 4. Chart of the preprocessing work flow necessary for a consistent storage of the TBM model.

6 CASE STUDY

In this case study the feasibility of the introduced approach is presented. The whole process cycle of how the model of the TBM can be used is demonstrated by a small example.

For the purpose of employing various simulation components used in the ongoing research currently underway at the Ruhr-Universität Bochum, an abstract model of an EPB shield machine has been modeled in the SolidWorks software. This software has been used due to its supporting tools for constructing machines, including the large amount of modeling tools, the design validation option and the parametric model function. Due to this function, the same model can easily be generated in different sizes. The newly constructed model is shown in Figure 5. The model consists of the two spatial sections,

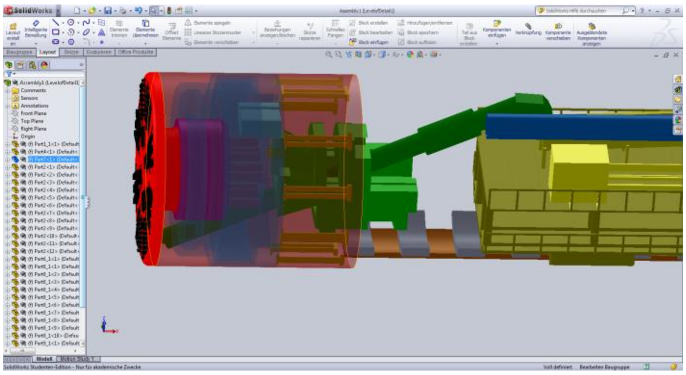

Figure 5. TBM model designed in SolidWorks

TBM head and backup equipment. The TBM head is represented by 41 different components of 10 individual types, whereby the backup equipment is defined by 53 different components of 10 individual types. All these elements are represented by a 57206 triangles.

During the design phase, the designer has assigned, among other information, the necessary attributes to define the element classes and the spatial sections. After the design process has been completed the model is exported as an IFC file. Hereby, the individual elements of the TBM model are exported as *IfcBuildingElementProxy* classes. Due to the added identification attributes these proxy classes are now identifiable.

Next, the conversion application is used to upgrade the exported IFC STEP file by the knowledge of the IFC TBM EXPRESS scheme. During this process a new IFC STEP file is created containing the new developed IFC classes of the TBM model. This file is uploaded to a Product Server System. The Product Server System used for this case study is a special database system set up for the ongoing research and is not further discussed.

The newly created model of the EPB shield machine is now successfully stored in the Product Server System and readily available. Thus, individual simulations can access these data to run their calculations. In addition, this model can be used to visualize the current status of the current project. For example, the engineer in charge of a tunneling project may want to see the current position of the TBM in the entire ground environment. In particular, he can check if there will be a change of ground conditions in front of the TBM which has a massive influence on the driving process. To read the current status, a web-service is used, including the export application which accesses the Product Server System. It reads the actual position of the TBM and automatically generates an IFC step file containing all the information about the ground, the tunnel and the TBM. In this case study a virtual model from the previous research project TunConstruct (TunConstruct 2010) is used. It contains a geological model with a size of 10 by 10 kilometers and fifteen different layers. Additionally, the track of a tunnel lining is provided. These data are used for this case study to generate the current status of this virtual project, in this case, after excavating 100 meters.

The generated IFC step file has to be read in an IFC compatible tool. Here, Open IFC Tools is used, which has been developed by the chair of Informatics in Construction at the Bauhaus University Weimar, Germany (Tulke et al. 2011). Open IFC Tools is a fully object oriented Java based tool to read and write IFC STEP files. An integrated STEP parser

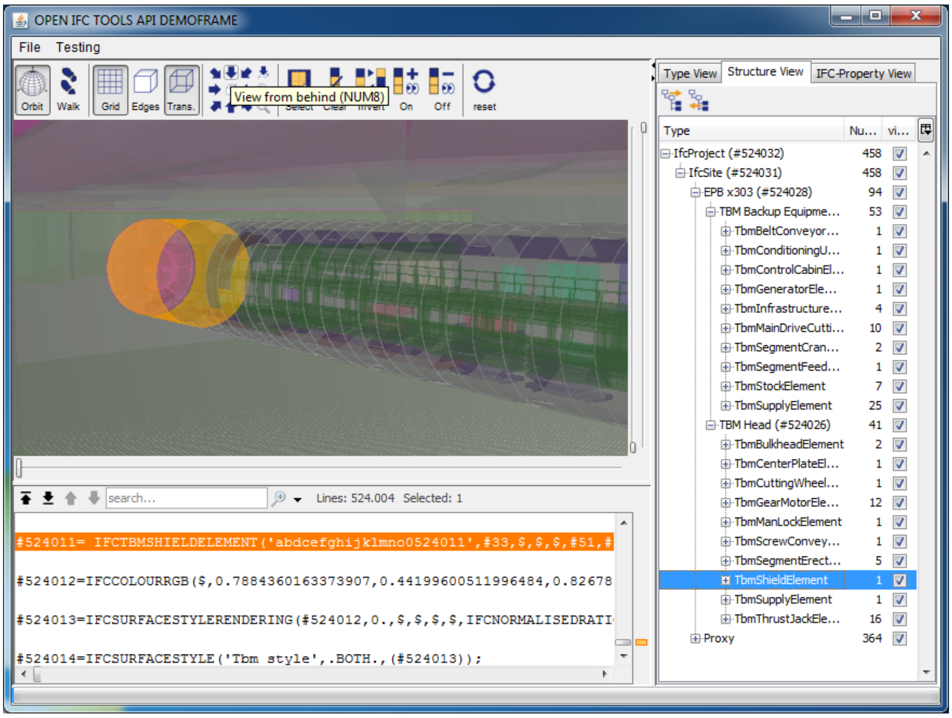

Figure 6. Status of a tunneling project loaded into Open IFC Tools

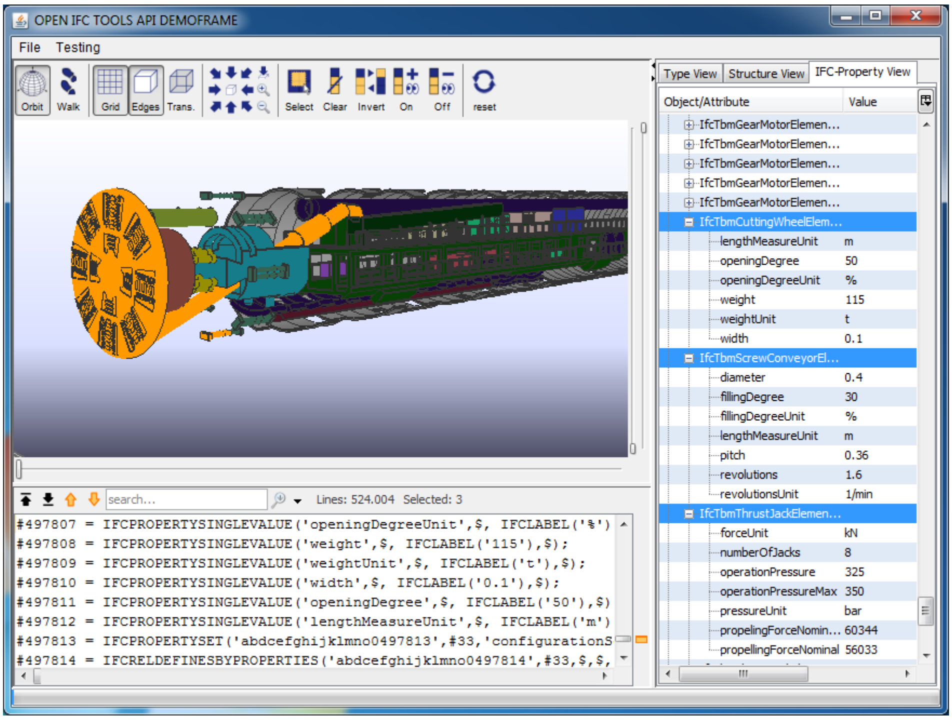

Figure 7. TBM model loaded into Open IFC Tools

creates a Java object for each IFC instance (Theiler 2010). Furthermore, the read files are shown in different structured views, so specific objects can easily be found and selected. The visualization is realized with Java3D so that it can support nearly all 3D representation types. Additionally, it provides functionality for the visualization of objects such as color settings, transparency, visibility and other appearance settings.
The results of loading the status of a tunneling project into Open IFC Tools are presented in Figure 6 and 7. Therefore, the Open IFC Tools has been structured into three windows. The first windows visualizes the model and provides tools to watch it in all kinds of perspectives. On the right, the second window provides three kinds of views displaying the structure of the IFC STEP file. In the last window the IFC STEP file itself is printed. Figure 6 shows the extracted model of the tunneling lining after excavating 100 meters. To have a better visibility, all elements have a transparent visualization. The tunnel is positioned under three separate layers. It has a round structure and the TBM is shown in front of the tunnel face during the excavation process. On the right-hand side the structure of the IFC step file is illustrated in the "Structure View" view. Here, all types of the used IFC classes are listed as well as the newly added classes of the TBM model. The view clarifies the structure of the TBM model. The two spatial sections of the TBM are captured by the TBM itself. In addition, the element classes are attached to their corresponding spatial section. On the bottom the figure shows the extract of the read IFC STEP file. To be more precise, it shows the creation of the *IfcTbmShieldElement* class which has been picked in the tool. The layers and the tunnel are represented by proxy classes because momentarily these types are not supported by the IFC notation.
Figure 7 focuses on the TBM. Here, on the right-hand side the property view is shown. The property view displays all meta information attached to an element of the model. This excludes the identifying attributes of the TBM model because the identification process has already been completed during the import. In Figure 7 the meta information of three element classes are displayed. The meta information consists of a list of values in a key-value style belonging to an *IfcPropertySet*. These property values deliver significant information to the user (e.g. to get an impression of the dimension of the element). Additionally, these meta information is usually read by simulations, accessing the Product Server System, as input data. The STEP file view shows how the meta information of the cutting wheel is stored in the IFC STEP file. The different key-value pairs are listed in individual *IfcPropertySingleValue* classes and assigned to the *IfcPropertySet* "configurationSet". Then, this *IfcPropertySet* is connected to the *IfcTbmCuttingWheelElement* class.

As a conclusion, this case study has proven the feasibility of the presented approach. Created models of TBMs are preprocessed and then uploaded to a Product Server System to be readily available for project members or process simulations. Furthermore, the uploaded model can be exported, converted to an IFC step file containing the new IFC classes and be read by IFC compatible tools.

7 SUMMARY AND OUTLOOK

This paper presents a first approach of an IFC-based model for TBMs, in particular, for EPB shield machines. Several classes based on the IFC standard have been developed to capture the essential components of an EPB Shield Machine. The model consists mainly of two kinds of classes, the spatial classes, representing the spatial sections of a TBM, and the element classes to specify the TBM components. A preprocessing work flow has been introduced. It changes an IFC STEP file containing a TBM model and exported by a common modeling software so that it applies the newly developed classes. This conversion is executed by a special conversion application consisting of an import and export part. To import a TBM model specific attributes have to be added to each element so that these are identifiable because the elements are exported as proxy classes by the modeling software. Thus, the import part is able to identify each element and to store this information for the export. The export part can read the identified model. Knowing the newly added classes the export part writes a new IFC step file containing the exported model. This new IFC STEP file is then uploaded to a Product Server System to be readily available for planers or process simulations. From there the model can be exported as IFC STEP file and imported by IFC compatible tools. In a case study the feasibility of this approach has been proven. Hereby, a new model has been created using the SolidWorks software and been uploaded to a Product Server System. In the next step the model has been converted, exported and loaded into the Open IFC Tools.

In the future the model will be incrementally improved in the ongoing research. Further classes will be added to create a detail model of an EPB shield machine. Additionally, the model must be extended for modeling further TBM. Another extension is the integration of process data. This data occurs during simulations or the real advancement process. These are measurements of the state of a specific element at a specific point or period of time. This kind of data is important to conclude why specific events occurred and why they lead for example to a system failure. In a next step a comprehensive tunnel model is needed which should be developed based on existing approaches.

8 ACKNOWLEDGEMENT

Financial support was provided by the German Science Foundation (DFG) in the framework of subproject "Multi-level product and interaction modeling in tunneling" of the Collaborative Research Center SFB 837. This support is gratefully acknowledged. In addition, the authors want to thank Eike Tauscher for his great support during the implementation and integration of the TBM model into Open IFC Tools.

9 REFERENCES

Beetz, J., van Berlo, L., de Laat, R. & van den Helm, P. 2010. Bimserver.org - an Open Source IFC model server. In Proc. of. 27th International Conference on Applications of IT in the AEC Industry CIB-W78. Cairo, Egypt.

BuildingSMART. 2011. Industry Foundation Classes (IFC) - The buildingSMART data model. http://www.buildingsmart.com

Byrnes D. 2011. "AutoCAD 2012 for Dummies", Wiley Publishing Inc., ISBN 978-1-118-02440-9.

Ji, Y. Borrmann A. & Obergrießer, M. 2011. Towards the Exchange of Parametric Bridge Models using a Neutral Data Format". In Proc. of the ASCE International Workshop on Computing in Civil Engineering. Miami, Florida, USA

Kymmell W. 2008. Building Information Modeling – Planning and Managing Construction Projects with 4D CAD and Simulations. McGraw Hill Construction Series.

Planchard D. & Planchard M. 2012. Engineering Design with SolidWorks 2012. Schroff Development Corporation, ISBN 978-1-585-03697-4.

Rahm, T. 2011. Process Simulation of Earth Pressure Balanced Shield Machines. In Proc. of 23rd Forum Bauinformatik. Cork, Ireland.

Rebolj D., Tibaut A., Čuš-Babič N., Magdič A. & Podbreznik P. 2008. Development and application of a road product model. Automation in Construction 17, pp. 719 – 728

Theiler M. 2010. Interaktive Visualisierung von Qualitätsdefiziten komplexer Bauwerksinformationsmodelle auf Basis der Industry Foundation Classes (IFC) in einer webbasierten Umgebung. Master thesis, Bauhaus University Weimar, Germany

Tulke J., Tauscher E., Theiler M. & Riedel T. 2011. Open IFC Tools-Processing / Visualisation / 4D. http://www.openifctools.com

TunConstruct. 2010. TunConstruct: Advancing the European underground construction industry through technology innovation. http://www.tunconstruct.org/

Yabuki N. 2008. Representation of caves in a shield tunnel product modeling. In Proc. of the 7th European Conference on product and Process Modeling. Sophia Antipolis, France.

Yabuki, N., Lebeque, E., Gual, J., Shitani, T. & Li, Z. T. 2006. International Collaboration for Developing the Bridge Product Model IFC-Bridge. In Proc. of the International Conference on Computing and Decision Making in Civil and Building Engineering. Montreal, Canada.