

Interoperability Challenges of an Engineering Software Provider

Tamás RÁCZ

Luleå University of Technology, Luleå, Sweden

Thomas Olofsson

Luleå University of Technology, Luleå, Sweden

ABSTRACT: The purpose of the paper is to give the view and experiences of a building analysis and design software provider on interoperability matters. The paper intends to show the difficulties the company faces during selecting the right approach for handling interoperability tasks especially cross-discipline data transfer tasks. It also will show how other influential solution provider's approach and global interoperability alliances like the International Alliance of Interoperability (IAI) affects the software design strategy. Obstacles in the way of providing better quality implementation of interoperability solutions and in formulating a transparent and reliable interoperability strategy for the future are going to be identified. The paper also takes a glance at the company's vision on the future of software interoperability for engineering and the topics considered as target for future research.

1 INTRODUCTION

Case studies of Building Information Modeling (BIM) applications are currently being presented in journals, conferences and workshops worldwide (Olofsson et al. 2008). The biggest BIM initiative is the International Alliance of Interoperability-initiated Building Smart forum for the industry and governments to identify, test, review, recommend and implement smart ways of delivering quality buildings and services to the facility owners using BIM technology (BuildingSmart 2009). Despite the serious efforts case studies in the application of the Industry Foundation Classes, shortly IFC (BuildingSmart 2009) standard to real projects and technical comparisons, such as that by Pazlar & Turk (2008), show significant weaknesses that currently exist. These results indicate that no software implementation of interoperability is fully satisfactory and that interoperability is still a concern in the advance of BIM.

In the following paper we would like to present a small software vendor's view on the issue of interoperability implementations and the challenges it brings to the company.

2 OBJECTIVE AND METHODOLOGY

The objective of the paper is to analyse the requirements and solutions of interoperability between modelling and analysis software systems. The analy-

sis is based on the experiences of the first author who is a graduated structural engineer working in the company in subject as software engineer in the area of interoperability solutions. He has been actively involved in engineering software development for the past 12 years with major part spent on interoperability. The paper will be presented from the viewpoint of the first author and the company. Using the word 'we' in the followings will refer to the company and the first author as the representative of the company.

First a short presentation of the company and its product portfolio is given before the interoperability situation is described in more detail. Different type of interoperability connections for FEM-Design, a software package in the company's product portfolio, is analyzed in more detail. Finally, the findings are summarised and concluded in the last chapters.

3 PRODUCT PORTFOLIO OF THE STRUSOFT

Structural Design Software in Europe AB (Strusoft 2009) is a software vendor specialized in building engineering software development and distribution. Its main office is located in Malmö, Sweden. It has development offices in Budapest, Hungary, in India and sales representatives in the United Arab Emirates.

The company has the following main product packages:

- FEM-Design: generic FEM analysis and building design software for whole buildings and building elements
- WIN-Static: software package for the analysis and design of buildings' structural elements
- IMPACT: precast building design and reinforcement detailing products
- VIP-Energy: product for calculating the building energy balance

The development of the engineering tools started independently from each other and their history goes back to more than 20 years.

The integration of building design software was not a requirement in the beginning. The demand for connections and interoperability between the own and to other vendors' software tools has increased gradually over the years. Since the company is small compared to the big international software vendors, the cooperation and the handling of interoperability issues has become strategically important.

In this paper the interoperability situation for the FEM-Design (2009) package will be analyzed in more detail since this package has the longest history of implemented connections, and the most of the interoperability aspects are experienced with this software package. The experiences gained from interoperability solutions implemented in FEM-Design are believed to be valid for other software packages.

4 INTEROPERABILITY OVERVIEW

FEM-Design is an analysis and structural design tool built on the top of discontinued CAD software called MrCAD. The CAD influence can still be found in the working environment of the FEM-Design. During the extension of MrCAD with FEM and other calculation functionalities the 3D oriented modelling and navigation as well as object-oriented design was the favoured approach. The program has a custom made toolset that makes it possible to create the analysis model from scratch with the help of typical CAD-like operations (copy, mirror, rotate, etc.). As a consequence of the 2D CAD heritage and the 3D object approach, the program has a classic 2D as well as modern BIM style connections to other software systems.

The connections of the FEM-Design can be categorized according to the followings:

- Based on the routing mechanisms of information:
 - File-based exchange
 - Add-on
 - Direct link (peer to peer)
 - Database connection (client-server)
- Based on the exchange format of information:
 - Proprietary format
 - Open standard format (IFC)

Since the program's purpose is to support the structural engineering activities, and since engineering is acting dominantly in the receiver's role in the building design process chain, the program's most important connections are import functionalities. Table 1 shows the publicly available connections today.

Table 1. Available connections in FEM-Design today

Direction	Routing	Format or target
Import	File-based	DWG / DXF, ArchiCAD, Xsteel, Revit Structure, IFC
Import	Direct link	Tekla Structures, Revit Structure
Export	File-based	DWG / DXF, XSteel

5 ANALYSIS

5.1 Requirements on interoperability

The requirements on the FEM-Design connections are practically the same as for any other connections in the building industry:

- Reuse of available information: this is the main purpose of any connections. The exchange procedure should deliver already defined concepts between the systems with the least loss of information, avoiding re-definition of the same information in the building design process.
- Speeding up the processes: not independent from the previous point since avoiding the repetition of tasks results in accelerated processes.
- Improved quality: this is also a possible consequence of the information sharing since it could eliminate some of the human mistakes possibly introduced by the repetitive work.
- Change management: to follow and to manage the changes occurred in any part of the connecting building design activities. It should be made on a way that helps preserving the greatest extent of added information compared to a previous state of incoming design but updating and eliminating invalidated parts at the same time.
- Easy translation of information: the conversion and translation of the compatible concepts of the connecting systems should be easy.
- Automation: there should be a set of tools to help the users in tedious high volume interoperability-related operations. These tools might help both the conversion of information and the addition of new information based on incoming data. Powerful tools may not only speed up the processes but could also help to raise quality. (Note: this might be debatable if this point really belongs to the requirements of the interoperability or it is more

like a user requirement on the engineering tool independently from the connection feature).

- **Navigation and aiding tools:** provide functionalities to group, view and filter the information under transfer to help the users in complex interoperability situations. Functionalities include the selection of partial models, building element type, filtering of out of scope elements.

The above list of requirements has been formed and modified during the years of interoperability developments in the company and hasn't been considered in full in all of the situations.

5.2 Routing scenarios

The selected routing scenario of the information has various consequences on the usability and on the available features of the connection. For the available routing scenarios see Figure 1.

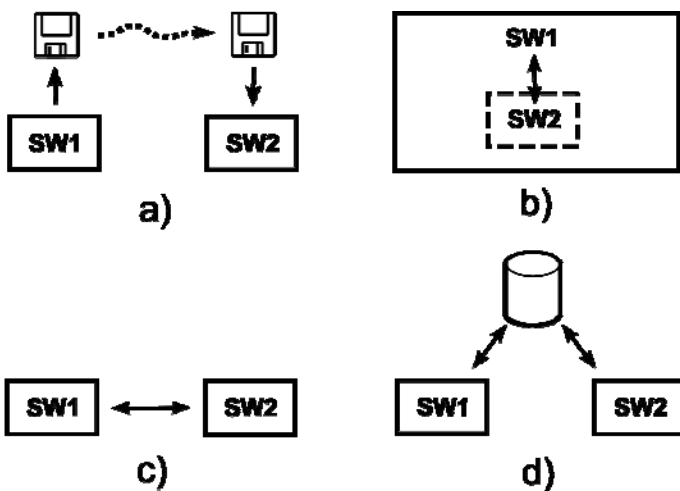


Figure 1. Routing scenarios: a) file-based b) add-on c) direct link d) database

5.2.1 File-based

This is the most common way of information exchange. The information is delivered by file in a selected format, which might be a format of one of the communicating systems or might be an intermediate format.

This way of information exchange is relatively easy and flexible to implement but also provides fewer features and requires more attention in case of recurring exchanges in design iterations.

It is important to choose the right format for the file. Intermediate formats make the two software applications more independent but requires two different conversions. It requires one conversion from the source format into the intermediate format and another conversion from the intermediate format into the destination format. Since any conversion of data is a likely – probably unavoidable – source of data loss, the two times conversion only increases the risk. Choosing the native format of one end reduces the risk of data loss but increases the dependency of

the two systems on each other; this has unwanted consequences on the software maintenance issues.

The actual way the data travels between the systems is not specified, it could be any way a file could travel – both online and offline.

5.2.2 Add-on

This is the extension of software features of an existing system with other tool's features where the interoperability nature might be well hidden or encapsulated.

In the FEM-Design package there is no example for add-on type connection but the VIP-Energy's analysis engine is built into the ArchiCAD (Graphisoft 2009) architectural design software.

This way the information conversion is well integrated into the parent system allowing smooth user experience and operation. Only a subset of functionality is available in ArchiCAD from the VIP-Energy package, but it might be an advantage in this case since the users of the ArchiCAD are from different profession. With the definition of a limited set of additional energy-related information the add-on is capable of providing quick estimation from the energy domain in the architectural domain tool.

The results of the analysis is presented in the host system.

For more reliable and accurate calculations the add-on falls back to the file-based operation by exporting the full input data for extended analysis in the main VIP program.

5.2.3 Direct link

We call it a direct link when two standalone software exchange information in real time. The means of data transfer is hidden from the user (unlike a file-based solution).

The Tekla Structures (Tekla 2009) – FEM-Design and by some extent the Revit Structure (Autodesk 2009) – FEM-Design connections are considered to be direct links.

The operation is initiated from the source system just like at the add-on type but the calculation is performed and the results are available in the destination system.

The advantage is a tighter integration compared to the file-based solution and an accelerated exchange; but it also has procedural implications: the connected tools should be available at the same time.

In case the two interoperable tools are installed on the same computer it requires more knowledge from the users; they should be familiar with both systems. Depending on the licensing conditions it also might be more costly.

In case of remote connection the requirement on software tool knowledge is less demanding but organizing the exchange process requires more effort.

The two users of the connected tools should be available for the task simultaneously.

5.2.4 Database

In this case the tools are sharing, rather than exchanging information.

The IMPACT precast design system stores its model information in a local or remote database; the FEM-Design analysis tool can connect to and read relevant information to its local model (this functionality is currently under development at the time the paper is written).

Just like the add-on and direct link types this type requires no data delivery operation from the user but at the same time there are less process constraints than at direct link.

5.3 Conversion categories

Based on the necessary cross-discipline information format conversions and translations we could have the following groups:

5.3.1 Manual conversions

The destination tool receives and presents the received information without alteration. The user should perform the interpretation of the incoming data manually for the scope of the receiving end. This is only possible for the data that has matching functionality for storage and presentation in the two systems.

In our practice the DWG based exchange could be considered as fully manual. Thanks to the CAD foundation of the FEM-Design the CAD information can be presented in its original form. The users interpret the CAD information and create corresponding analytical BIM objects manually with the native FEM-Design tools.

5.3.2 Automatic conversions

In this case the destination system performs conversion on the incoming data automatically based on predefined rules.

This kind of operation has positive impact on the usability of the tools however defining the conversion rules can be difficult and often results in incomplete conversions: usually there is a limit of the scope where the automatic conversion can be performed.

In case the users are unaware of the limits of the scope the outcome of the conversion can be unpredictable which in turn affects the reliability of the conversion results negatively.

All the FEM-Design's BIM based connections belong to this group of conversion. The architectural import type of the IFC import is a classic example of the importance of the awareness of limitations. The architectural import can process regular shaped simple buildings, but processing buildings with more

complex shape than the limits of the conversion usually ends up in incorrect model.

5.3.3 Semi-automatic mode

The original model is presented in its original form. The operations are performed with the help of predefined algorithms and rules but the user selects the scope of the operation by selecting on which part of the model it should be executed.

5.4 Implementation difficulties in the practice

5.4.1 Different views

The various building modeling and engineering tools are representing the same building products from different perspectives. The differences not only lie in the different form of geometry, but in the different level of details and the handling of concepts, such as materials or cross-sections.

The 'reuse of available information' requirement of Chapter 5.1 between different views means more than a "copy and paste" operation. Ideally based on the data of the source tool the corresponding data in the destination tool should be generated by conversion or translation.

The conversion of geometry is a typical example for conversion difficulties (see Chapter 5.4.2). The way a physical model should be converted into an analytical model is different for the various possible situations; for example, it may depend on the level of model maturity of the design process: in a conceptual design phase the simplifications of the model and the requirements against the data exchange might be very different compared to a detailed design phase.

Sometimes the users implicitly expect that 'intelligent' computer systems are able to identify the incoming model's nature and to select the right approach automatically for creating the correct analysis model. Partially the software vendors might be blamed for this expectation. Since software vendors compete with the effectiveness of their interoperability solutions they sometimes hide the limitations of the conversion behind the scenes. The first attempt to 'import physical models into the analytical tools' in FEM-Design proved to be simple and efficient but only for a very limited case of applicability. The limitations might not be emphasized enough in communicating the feature towards the users.

A fully automatic solution that can be used for any situation – even with customization parameters – is difficult to design and implement.

5.4.2 Conversion of geometry

FEM-design has its own way of handling the geometry to support the main objective: the creation of the analysis model. The handling and the approach to

create model geometry might be quite different in design tools, especially in design tools of other disciplines, e.g. in architectural systems where the task is to create physical model.

The main difference between physical models and analytical models is that the physical model is composed of 3D objects without clashes, while in the FEM-Design 1D and 2D representation of the 3D objects should form a continuous analytical model. The clash of the generated 3D extent of the analytical objects – by visualizing the cross-section parameter – in the analysis model is not relevant. The positioning of the model elements in space might have apparent incompatibility between physical models and analytical models. For example the extent of the analytical model elements may not match the relevant physical model elements' boundary. See Figure 2.

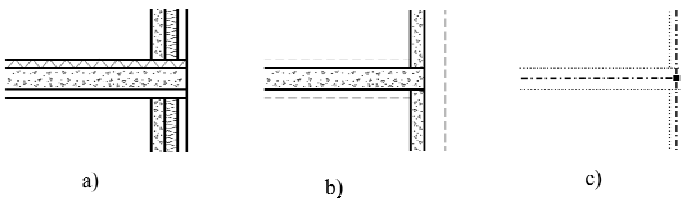


Figure 2. a) architectural model, b) structural model c) analytical model (based on Rönneblad 2002)

The conversion of the physical model into analytical model is difficult for generic cases (Rönneblad 2002). Traditionally this is the task of the structural engineer. There are situations where the conversion can be automated, but this would most likely require limitations and the implementation of guidelines for the modelling activity.

The knowledge of the structural engineer is difficult to put into algorithms and applicability conditions for the generic situation. So far no generic methods have been found for the conversion of every occurring 3D solid model situations into analytical model during the history of FEM-Design connections. Special purpose connections have been implemented where the transmitting modelling tools had built in limitation that simplified the conversion process. The extension of special purpose approach to more generic situations became difficult and expensive to implement, also the results were not satisfactory enough.

5.4.3 Translation of compatible information

Between disciplines, but also sometimes between sub-disciplines (between structural modeling and structural analysis) the handling of the same concepts might be different; therefore the information transfer may become difficult. A typical example is the handling of associated material attributes. “Concrete” in the architectural domain should be inter-

preted as “C25/30” or “K20” or other more specific material grade for the analysis domain.

Between structural design and structural analysis software, where the handling of material attributes are supposedly the same the problem might arise in the identification of the same material grades (“C20/25” or “C 20/25”, please note the space) or in the details of defining material properties. Some properties might be used in one application and not in the other, or the parameters might be used different ways. An example of different approaches is a) defining of standard material parameters from tables of the building code or b) only the main parameters are defined and other dependent values are calculated, based on the building code's regulations. The two approaches are different but the result must be the same. However, the use of tabulated versus calculated values can in some instances lead to slight differences due to rounding errors, and eventually to translation difficulties, or even to data loss during transfer.

Example: In the Table 3.1 of the EN 1992-1-1:2004 for the material grade C60/75 the code defines the following material characteristics: $f_{cm} = 68$ MPa and $f_{ctm} = 4.4$ MPa. However the analytical relation for the f_{ctm} value is also given:

$f_{ctm} = 2.12 \times (\ln(1 + f_{cm} / 10))$. Calculating the f_{ctm} value based on the formula gives $f_{ctm} = 4.354742$ for the first six decimal digits. Even this small difference in the values between the two situations makes the parameter based pairing difficult.

The situation gets even more complex by allowing the definition of custom materials, which is normally possible in the design tools.

Since no generic solution has been found for the problem in the Strusoft's practice the chosen strategy is to give the professional user the responsibility of reviewing the materials at the two ends of the connections and defining pairing explicitly between materials in the two systems. The pairing is done using a conversion table between identified materials.

5.4.4 Recurring exchange

In most cases, building design is not a single run linear process; it contains iterations, loops and changes. This leads to the need for recurring exchanges between the interconnected software tools.

During these recurring exchanges the task is to identify changes in the transferred information compared to the previous state, invalidate the changed and the dependent added information while keeping unaffected parts intact in the destination system.

The first difficulty occurs in the identification of changes; either the source or the destination system should store information about the state of the model. Ideally, the source system should inform the destination system about new, changed or deleted information, since normally the sending system has

the full and direct access to the source model's design history.

Unfortunately, it is rare that the source system provides this functionality at the required level. Often it is completely missing, or the change management is not detailed enough. For example when the fact of a change of an object could be identified but the nature of the change is unknown then the destination system is required to identify it: in case where the re-exported model contains no structural analysis related changes (e.g. change of visualization related attributes), there is no need for updating the dependent analysis model. Where the change management is not detailed enough and the source system provides change notification only on the object level, there the destination system has to perform object attribute level comparison to identify the exact nature of the change.

After the identification of the relevant changes the necessary model updates should be performed on the receiving end. Firstly, it is imperative to repeatedly translate and convert the changed incoming model information format into the corresponding destination model's format. Previously-met conditions of automatic conversion may fail, requiring some manual operation, i.e. the interoperability workflow may change too. Secondly the additional model elements of the receiving software should be updated. For example, after geometrical changes to a slab contour the relevant loads in the analytical model may be out of its boundary, or non-relevant loads could become active on the new extent of the slab. If the analytical tool has not been designed for change management related dependency handling, then this can only be corrected manually.

Due to the complexity of the change management operations, the handling of changes requires manual intervention, or at least manual revision in generic cases.

The lack of harmonized change management procedures amongst construction design tools makes it even more difficult for the software developers to provide effective automated support for recurring exchange. As a consequence and workaround the Strusoft tools today require considerable amount of manual operations for following the changes in the interconnected design tools, including manual comparison of incoming data and manual adjustments on the existing destination model.

5.4.5 Procedural uncertainties

When and where specific piece of information should be created is not always obvious: should it be defined in the sending or in the receiving system? Concerning structural analysis there is a trend today to define more and more structural analysis information in the structural modeling systems. Both Tekla Structures (Tekla 2009) and Revit Structure (Autodesk 2009) provide the opportunity for the user to

define analytical information in their system. However for the actual analysis it is required in both tools to transfer the information to another system that is capable of performing the analysis.

The difficulty with this solution is that two different tools are needed to perform the analysis. The adjustment of the analysis model requires adjustments in both systems, which is more work and puts more demand on the data exchange operations. The need for data transfer can be avoided in case of add-on type of connection but this case would require matching analysis model manipulation and results representations in the structural analysis and structural modeling tools which is normally not the case today. The structural modeling tools are not made for full scale of structural analysis operations today.

Another procedural issue is that the efforts of providing new software tools for the design processes could result in, and sometimes even requires changes to the current industry practice.

However industry practice does not change easily. Available staff is already trained and existing organizations are built upon the current practice. It is difficult to predict how much a specific new or improved functionality will affect industry practice. The provision of new powerful functionalities might meet with rejection by the users due to the uncertainties of the changed practice, regardless of how promising and beneficial those improvements could eventually be.

5.4.6 Data format

The experiences of interoperability developments in the Strusoft show that every software applications have their own form or own flavor for defining building information. The same information could be expressed on various ways; the one that suits the best the objectives of the software and the algorithms used will be selected. Even a simple concept like slab contour definition has various alternatives (not a comprehensive list):

- The contour can be defined as a series of points. The curved edges are marked in supplementary data.
- The contour could be defined as series of 2D primitives: line sections and arcs
- Contour arcs could be defined by two end points and a third point in the middle
- Contour arcs could be defined by two end points and an angle
- Contour arcs could be defined by two end points and center point
- Contour arcs could be defined by circle plus start and end angles to a reference direction
- The start and end points of a closed contour might or might not be duplicated

For the development of each new connection between two applications the developers need to learn the actual flavor of the foreign information format

and find effective ways of conversion. This knowledge needs to be maintained for the lifespan of the connectivity. Consequently each new data format puts additional load on interoperability developments.

5.4.7 *Different level of features*

Every software application has features not included in others. For example implementing the analysis and structural design of – possibly intersecting – curved surface objects (walls, shells) is typically more complex and consequently takes longer in structural analysis tools than in modeling tools. This leads to a temporarily situation where the modeling software has a feature that cannot be matched in the analysis software package. Other examples for feature differences include the already mentioned possible difference in information level of materials, availability of arbitrary openings and cuts in beam or column objects (easier in physical modeling but the handling in analysis might vary depending on the situation), compound objects like trusses, just to mention a few.

Various possibilities are available for the handling of feature level differences:

- Drop unsupported concepts. Easy to implement but leads to data and feature loss. The users should be notified about the data losses but it is problematic to report on a non-existing concept effectively.
- Find the closest substitution to the feature. Still data loss will occur but on a lower level compared to the previous solution. The nature of the data loss depends on the ability to mimic the desired solution. The developers should seek after to minimize the overall effect of the data loss, which might be difficult to accomplish. For example substituting one curved object with several linear sections might result in acceptable error margin in the analysis but might lead to confusion and errors in consolidating information from various sources for coordination; i.e. one object in one system will become several in the other.
- Synchronize the level of features. Theoretically the synchronization could mean both degradation and elevation of features, but since the degradation means loss of functionality, which is not welcome amongst the users, therefore the implementation of missing features is the desired solution. However, the elevation of feature level is not always possible or not possible in an acceptable timeframe.

5.4.8 *Software technologies*

The applied software technology is rarely the same across the software tools, which is true for the connection interface technology too.

In the interoperability developments of the FEM-Design the underlying operating systems are domi-

nantly the same (Windows family), which eliminates some cross-operating system difficulties. But still the various programming languages (C, C++, C#) and technologies (file formats, COM, .NET) and the various level of incompatibility between the subsequent versions of developer tools (Visual Studio 2003, Visual Studio 2005, Visual Studio 2008) put considerable additional demand on the developer team of the connections.

The modularization – reuse of similar practice – is difficult among different software technologies. An existing, functionally compatible geometric conversion routine may be required to redesign for a new connection due to the different handling of information in the underlying technologies. Otherwise identical user interfaces should be ported, and unified approach – like steps of material translation – might be changed. When correction or improvement is necessary on similar concepts across different technology connections then the required efforts get multiplied because of the need of synchronization.

Practically each and every connection is unique to a great extent. The development team must learn every new combination of solutions and technology, and maintain the knowledge for the whole lifespan of the connection.

Last, but not least the interoperability situation is rarely static. The connected software tools are in a constant state of improvement; when the underlying technology or modeling approach is changed in one of the software tools or new versions of the developer tools are utilized, then the changes need to be followed up in the other software too. The evolution of the software technology requires continuous follow up in the connection, and it requires continuous education of the development team for ensuring stable and reliable connections.

5.4.9 *Standardized solution*

The IFC format (BuildingSmart 2009) is a big promise for the Strusoft's product lines. Instead of expending more and more resources on the implementation and maintenance of numerous connections based on various technologies the IFC format promises a unified, cheaper, single thread operation.

However the IFC format is huge and compared to application specific connections it takes more effort to learn and implement. Its application details are complex: implementers' agreements limit and select 'valid' techniques, sometimes more than one for the same concept (for example several kinds of geometry definition for walls). Implementers' agreement also might contain workarounds, which become obsolete in later versions of the format. At the same time the IFC format is not complete, there are still some features missing. It will probably never be complete since the connecting tools and the related practice will change gradually so the information to deliver will be changed too. The IFC format is al-

ready evolving continuously: version 2.0, 2x, 2x2, 2x3 and the upcoming 2x4 to mention only the latest five. The tracking and the maintenance of the IFC connections require dedicated effort. Sometimes the changes are minor or not relevant but still testing is needed to ensure that the connection works as before.

Actually the IFC format has no mainstream practical usefulness for the Strusoft today, it still shows promise for the future. This has two main reasons:

- The widely supported physical model information of the IFC format cannot be used directly since it requires geometric conversions into analytical models (see Chapter 5.2.2). For its practical use improvements and related research are necessary.
- The IFC information that the FEM-Design can handle natively – i.e. without any or with minor adjustments – is the structural information. This is the content of the so-called “structural view” of the IFC format, which does not yet have wide support amongst the up-stream players (physical modeler tools) in the IFC community.

Additionally the procedural aspects of the interoperability are not adequately regulated and supported in the IFC format (one aspect of this is analyzed in Pazlar & Turk 2008), however some work has begun with the introduction of the Information Delivery Manuals (IDM). There are basic concepts in the IFC to support procedural – recurring – exchange scenarios (for example “owner history” concept) but the IFC certifications and workshops are still not concentrating on the regulation of these concepts. Without synchronized development and improvements in this area the recurring exchange remains uncertain in its implementation details.

Also the industrial momentum in the usage of the IFC format is apparently missing. Users of the Strusoft tools like the idea of the IFC, but the tests of current IFC implementations frequently show that the technology is not ready for every day practice (Pazlar & Turk 2008, Hejnfelt & Øksengaard 2007). Meanwhile, the developers wait for signs of more usage since no business wants to invest into something that will not be used.

The reluctant implementation by other developers causes further holds back in the interoperability developments of the Strusoft. Both Tekla Structures and Revit Structure have IFC connection and some analytical modeling concepts that our users want to utilize. However the vendor of both Tekla Structures and Revit Structure prefer their respective proprietary solutions to the IFC format to deliver analytical information. During numerous IFC community meetings we experienced that both vendors are reluctant to start the IFC structural view certification process and joint implementation process of the analytical view of the IFC. Down-stream applications like FEM-Design need to connect to up-stream applica-

tions; future efforts from the Strusoft depend on the serious commitment of the up-stream vendors.

Since many independent parties are involved in the development of the IFC format and it is supposed to satisfy the most of common needs the evolvement of the standard is slow. In many cases, business needs require quicker responses than the standards evolution can satisfy. This could result in selecting alternative solutions or non-standard extension of the format.

Due to the difficulties and the changing situations the FEM-Design developments ended up in a complex IFC situation. The FEM-Design’s IFC import has three modes:

- Solid shape import
- Architectural import
- Analytical import

The ‘Architectural import’ was the first implemented. It was designed for a project where well-constrained types of building were transferred; the conversion of physical model to analytical model was unambiguous. The IFC’s coordination view was chosen for transferring information as an already existing exchange format, instead of inventing a new one. Transferring the physical shape and converting it into analytical objects for the well-defined types of structure was a feasible solution.

Later when the analytical view was developed in the IFC format the analytical import for the same purpose was implemented.

The solid shape import is used as a template model import showing the 3D shape of the IFC model (its so called “coordination view”). It can be considered as the 3D alternative of the traditional DWG/DXF import. The purpose of the solid shape import is to give the structural engineer an aid to start defining the analysis model on top of the physical model. So far, supports for other type of information delivery than geometric information are not included in this mode – e.g. assigned material. Until the commitment is higher from other vendors and the strategy for cross-discipline operations is better defined the situation may remain as diverse as it is today.

Probably the complex IFC format development and implementation situation might partially be blamed for some misunderstanding in the usage of the IFC import amongst FEM-Design users. Novice users seem to think that the existence of an IFC connection in architectural tools and the existence of IFC connection in analytical tools will result in flawless data flow between the two disciplines through IFC automatically. However in reality the IFC format allows the transfer of both physical and analytical information, but it does not deal with the conversion between them, as this is functionality outside of the IFC scope.

5.4.10 *Business aspects*

Optimal and desired solutions might take longer to develop. However long-term developments might cost more than the available resources available in the developer company. Additionally, concentrating on only long-term solutions might lead to the loss of user confidence: they might select alternative tools if the tools are not improved continuously. Therefore, short-term solutions may drive the business of a small company, even though it might introduce unwanted practice and divert resources from better long-term solutions.

Cooperation with influential partners might give advantages over standardized solutions through pre-developed and likely supplied solutions and the stability of the massive market of the partner. But it also imposes the risk of dependency on the fortune or will of the partner not only for the small software provider but also for the user community.

6 DISCUSSION AND CONCLUSION

The interoperability situation is diverse and complex today for the Strusoft, and we believe it will remain so in the future.

Efforts are put into standardization because of all its benefits, despite the serious efforts needed for reaching its goals. When parties speak the same language, and can interact in a synchronized fashion, then the productivity soars.

However, the standardized solutions might have disadvantages in particular situations when the scope of the standard is limited and custom solutions provide more benefits. Careful consideration should precede the implementation of unique solutions. Losing the advantages of the standardized solution should be compensated with significant advantages of the custom solution. Standardized solutions are easier to reuse and maintain but custom solutions have advantages on the performance and feature side.

Example: the generic IFC import in the FEM-Design suffers from difficulties in the conversion of geometry for the analysis model while the IMPACT precast connection could exploit the advantage of predetermined and limited geometry situation. If the IFC format were utilized for the IMPACT connection then that were involving additional work and difficulties by its generic nature; the advantage of knowing the restrictions of the model were lost during the conversion to generic format, while complications of conversion from generic format to analytical model were introduced (see Chapter 5.4.2).

The standardized solution is better when there is more than one tool to connect and the information to deliver has wider scope. In case of limited scope of information exchange the standardized solution introduces quite an overhead. It is also possible to use

a subset of the standardized solution but in this case it should be clearly separated from the standard solution, avoiding the confusion of non-conformant and full conformant formats. The advantage of standard subset usage might be in the possibility of reusing already available knowledge and system components.

The selection of custom solutions introduces additional requirements on the staff and resources during the whole lifetime of the connection, multiplying the efforts needed for interoperability. Therefore one should be very considerate in selecting custom solutions

Open solutions should be favored over propriety technologies in interoperability. As one of the FEM-Design's user expressed during discussions: "what is important for us is the information, not the software". This user prefers to have the same level of access to the building information regardless of the tools used. Open solutions enable the use of several tools for the same purpose based on their strengths and weaknesses. For example, in structural analysis one software package might be specialised in steel frame structures while the other package might be better in the design of reinforced concrete slabs. Having the complete building information available in open format makes it possible to use the best available software in the design process. The risk of losing information because it only can be accessed and managed in the authoring tool can be eliminated. For the users the open format may put less constraints on software usage policy and consequently on trained professional personal. The choice of the tools and consequently the choice of personal and partners can be more flexible.

Having the information available independently from the software solutions provides greater safety for the information property owners. It would make it possible to access and reuse the information property in the future with less risk of vanishing software versions or even software vendors when open solutions are used.

It is no surprise that the only available neutral format of IFC gets high attention from design and construction companies; regardless of disappointing adaptation details compared to other means of connections (Hejnfelt & Øksengaard 2007).

Considering software technology the file-based exchange is a quite straightforward solution and is a familiar way of operation without the need of interoperability process management system but the future is going in the direction of collaborative system usage. In case of more than two connected application the distribution of the data, the merging of deliveries and the coordination of the interoperability process might be as important as the information itself in transfer.

Interoperability solutions will probably remain diverse and also will keep changing continuously.

Therefore more flexible solutions would be better to have than those are available today. Adaptation is necessary not only because of new functionalities and changed conditions but because of the customization need to specific user requirements. The formats and technology used in interoperability should be possible to extend and customized quickly and cheaply to enable the adaptation to new and custom situations in the future. It is possible to change the format and process of interoperability for the new requirements, and convert the existing system – like converting BIM model database from 2x3 to 2x4 version of the IFC format – but it is even better if the system is made for changes initially and there is no need for conversions: the information is available regardless of the technical versioning. In the experiences of Ku et al. (2008) it can be seen that the liabilities in the design process have influence even on the very fundamental geometric modeling concepts of the BIM. There are various ways of organizing the collaboration based on the liability needs and probably all the potential situations are not possible to foresee and put into predefined interoperability concepts. The flexibility could help the handling of liability issues too.

6.1 *The implications for Strusoft*

The efforts spent on software interoperability should be maintained or even increased in the future. Even the most advanced engineering tool can lose its competitive edge if it can't be used together with other design tools.

Standardized and open solutions will have the priority but particular situations might justify the use of special and closed solutions when the advantages are overwhelming. Closed and special solutions could also be used in cases where no proper level open standard solution is available.

The interoperability developments should also focus on supporting the management of the information exchange processes in addition to the pure information delivery.

Otherwise it is not possible to define more accurate and detailed interoperability strategy, it should continuously be revised and refined for the actual situation.

6.2 *Future research*

There are some research questions that would be beneficial to investigate:

- How can the automation level of cross-discipline geometry conversion between physical – both architectural and structural – and analytical model be increased? A generic approach should be developed that could be used for as many cases as possible through customization. In case of no ge-

neric solution is available then partial solutions and their combination rules should be developed.

- How can heterogeneous and frequently changing information be stored effectively without the need for conversion of the whole dataset? Adding new and restructuring existing aspects in running projects and existing building design data are likely situations where the involved tools are from various independent sources and may be changed because of various reasons. A less conservative data management would make easier the reuse of parts and wholes of previous solutions and the combination of information from various sources.

REFERENCES

- Autodesk 2009. Software vendor of the Revit Structure structural design software. Home page: <http://www.autodesk.com/>
- BuildingSmart – International Alliance for Interoperability 2009. Home Page: <http://www.buildingsmart.com/>
- EN 1992-1-1:2004. Eurocode 2: Design of concrete structures – Part 1-1: General rules and rules for buildings
- FEM-Design 2009. Structural analysis and design software. Home page, <http://www.fem-design.com/>
- Graphisoft 2009. Software vendor of the ArchiCAD architectural design software. Home page: <http://www.graphisoft.com/>
- Hejnfelt, T. & Øksengaard, R. 2007. The use of 3D and BIM technology for structural analysis and design, Master's Thesis. Lyngby, Denmark: Technical University of Denmark. (<http://www.s-bim.com/>)
- Ku, K., Pollalis, S. & Fischer, M. & Shelden, D. 2008. 3D Model-based Collaboration in Design Development and Construction of Complex Shaped Buildings, *ITcon Vol. 13*, Special Issue Case studies of BIM use, pg. 258-285, <http://www.itcon.org/2008/19>
- Olofsson, T., Eastman, C. & Lee, G. 2008. Editorial - Case studies of BIM in use, *ITcon Vol. 13*, Special Issue Case studies of BIM in use, pg. 244-245, <http://www.itcon.org/2008/17>
- Pazlar, T. & Turk, Z. 2008. Interoperability in Practice: Geometric Data Exchange Using the IFC Standard, *ITcon Vol. 13*, Special Issue Case studies of BIM use, pg. 362-380, <http://www.itcon.org/2008/24>
- Rönneblad, A. 2003. Product Models for Concrete Structures, Licentiate Thesis. Luleå, Sweden: Luleå University of Technology.
- Strusoft (Structural Design Software in Europe AB) 2009. Home Page: <http://www.strusoft.com/>
- Tekla 2009. Software vendor of the Tekla Structures structural design software. Home page: <http://www.tekla.com>