

BCFViewer – BIM Collaboration Format tool development

F. Lorenzi and A. Geiger

¹ Karlsruhe Institute for Technology, Hermann-von-Helmholtz-Platz 1 76344 Eggenstein-Leopoldshafen

fernanda.lourenziv@gmail.com

Abstract: Building Information Modelling (BIM) is the foundation of a digital workflow between different fields in the architecture and construction industry. It establishes a method that integrates multidisciplinary data to create a digital twin of a building throughout its lifecycle, going from its initial planning and construction phases to the final execution. With the BIM Collaboration Format (BCF), buildingSMART provides a standard for the cross-application communication within an openBIM process. Therefore, the data can be exchanged and opened by different programs without the additional need to exchange the entire model. As part of an internship semester, a stand-alone application is being developed for managing BCF files. The BCFViewer, as the application is called, is designed to read data in the BCF standard and shows its content without losing information. New data including issues, comments and images can be created and saved. In addition, it is possible to share the data with project partners via integrated email notification. The resulting files can be read by other software applications for BCF management. The viewer is a stand-alone tool just for BCF management, as it does not interpret IFC files. That means an interactive view of the object is not possible. In the present paper the development as well as the functionality of this stand-alone software tool is described. The plan for additional complementary features is discussed and how it was designed as a module for the software tool FZKViewer. Finally, it is shown which features the application offers compared to similar tools.

Keywords: BIM, BIM Collaboration Format, Tool Development, BCF Viewer, BCF

1 Introduction

BIM is undeniably important to projects nowadays in a variety of fields in the AEC industry. It can help to see through details in the whole process of multidisciplinary work. Workflow methodologies are however essential to this and sometimes what the team needs for communication is rather a reference to the data-model containing the required information than the big-scale project. BIM Collaboration Format (BCF) comes in handy for this, as it is an open file format that raises issues efficiently while avoiding heavy file exchange [1].

Today it is possible to find BIM software with tools or complementary modules that support BCF files. What is required most of the time is to have said programs to be able to visualize the issues or information. With stand-alone BCF applications, it is possible to easily view those files when the goal is only to have knowledge of it or make a couple of notes on its content.

This paper intends to show a free stand-alone application for BCF management which has been implemented as part of an internship semester for a bachelor civil engineering course. It was initially conceived in order to complement some of the missing features of the FZKViewer tool. The application covers the basic needs for an easy reading and writing of a BCF file and focuses on sharing its content between interested people.

1.1 BIM Collaboration Format

The BIM Collaboration Format (BCF) was developed by BuildingSMART as an international open standard that allows different BIM applications to communicate modal-based issues improving the IFC-based process. BCF bases the model workflow in three stages: Define (what is the issue?), specify (what is the original problem that caused it?) and delegate (what is your suggestion and to whom are you assigning the action?) [1].

BCF was meant for a better communication over issues, and that's why most of the BIM Softwares offer BCF coordination through collision detection or model checking tools. It can also be used to exchange information, expectations, comments or remarks on the project. The data can be shared in two formats: XML, where the information is written in XML files exported to a ZIP container; or the BCF API [2], where the information can be exchanged via a webserver based on cloud synchronisation.

1.2 BCF Structure

When passed through the XML-format, the basic content of a BCF file includes the issue itself with comments and references of spoken objects (possible through its Global Unique ID in IFC). It is possible to add a Status-information, making it easier to track the development when discussed by multiple users, along with a viewpoint or screenshot of the situation in the application where the issue was created [3].

The structure of the BCF Version 2.1 is a ZIP file (Figure 1) containing one folder for each *topic* and XML-files defining the extension (extensions.xml) and the details (project.bcfp) of a project, the documents (documents.xml) in it as well as a XML file with the information of the BCF schema used (bcf.version). The name of the issue folder is the GUID of the topic and must be written in all-lowercase. Inside the folder are the following files [4]:

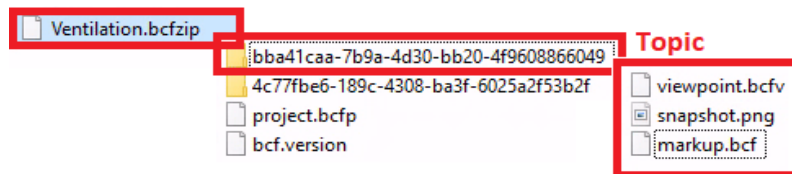


Figure 1: BCF ZIP file

- markup.bcf - an XML following the markup.xsd schema with text information about the topic. The information is separated into the nodes *Header*, *Topic*, *Comment* and *Viewpoints*. The *Header* contains attributes about the IFC file from where the issues are from. The *Topic* contain attributes to identify it (*GUID*), the *TopicType* and *TopicStatus*. The last two includes a predefined list to help categorize the topic. Besides that, the *Topic* contain several elements such as *Title*, *CreationDate* and *CreationAuthor* (obligatory); and *Priority*, *Labels*, *Description*, etc. (optional). The *Comment* node contains comments related to the *topic*. Each one has an unique identifier as attribute and may refer it to a *viewpoint*.
- Viewpoint files - an XML conforming the visinfo.xsd schema with information of components related to the topic, camera settings, and possible markup and clipping information.
- Snapshot files - PNG or JPG related to the object

2 BCF Software Tools

In some cases, BCF is used not only for reporting issues. The direct and simple structure is easy to implement and also very useful when it comes to documentation. There is a study, for example, that evaluates the methodology for image-documentation in the early phases of existing building through a BCF API, using different search parameters for image localization [5]. Keeping the simple specification in mind, BCF was also thought to be expanded to attend to the demands of task management in the building industry [6], by creating a data schema architecture that supports the

requirements such as topic definition, users involved or dates. Through the viewpoints components in the BCF parameters, which are able to show data-model relationships, it is also easy to spatialize information. For example, the damage mapping on-site for a later conversion to the BCF parameters is a cost-effective and simple solution for small and medium enterprises to make damage documentation and facilitate their workflows with a better accessibility to the provided data [7].

There are many tools available for the managing of BCF files that can assist the workflow and provide help in the desired domain. Some are integrated with BIM software or are modules that help with the Issue handling, which means that in most cases a license or purchase is required to see or edit the BCF. Even though the BCF standard is intended to be software independent, it is sometimes necessary for a casual viewer to go through this process and own one of these programs just to visualize the project issues or, as stated earlier, documentation. Software tools just for BCF viewing with different functions are available on the market. However, only a few of them are free. The BCFViewer as a freeware will be an additional option to fill this role of software-independent BCF viewer and/or editor. Later, a few more options for free stand-alones will be presented.

3 BCFViewer

The BCFViewer is a Windows desktop application able to read and create BCF files. The implementation of the tool was developed with Microsoft Visual Studio. The C++ programming language and Microsoft Foundation Classes (MFC) were utilized for this purpose. MFC is a library that wraps parts of the Windows API in C++ classes, including functions that allow them to use a standard application framework.

The goal of the tool was initially to be a built-in module for an IFC visualization Software, the FZKViewer. The FZKViewer was able to create BCF files through a model checking when errors were identified. The Issues were created automatically when selected with the information from the checking tool with no editing possibilities. For a more detailed issue handling with more freedom, a new module with some complementary features were desired. With the BCFViewer, now it was possible to create not only its own issues, but also new remarks or requests and send it to someone. There is also the possibility to add several comments and/or snapshots to each topic.

3.1 Features

The BCFViewer as a stand-alone tool is just for the viewing and creation of BCF and does not interpret IFC files. The goal is an easy and fast way of viewing and editing topics without the need of a BIM Software. That means that an interactive view of a viewpoint object is not possible.

In the main dialog (Figure 2), there is the possibility to open BCF files and view its content. There is a list of topics and a window containing information about the selected topic, comments and, if

available, a snapshot for the selected comment. It is possible to create and/or edit topics or comments and, if so desired, upload images to each comment.

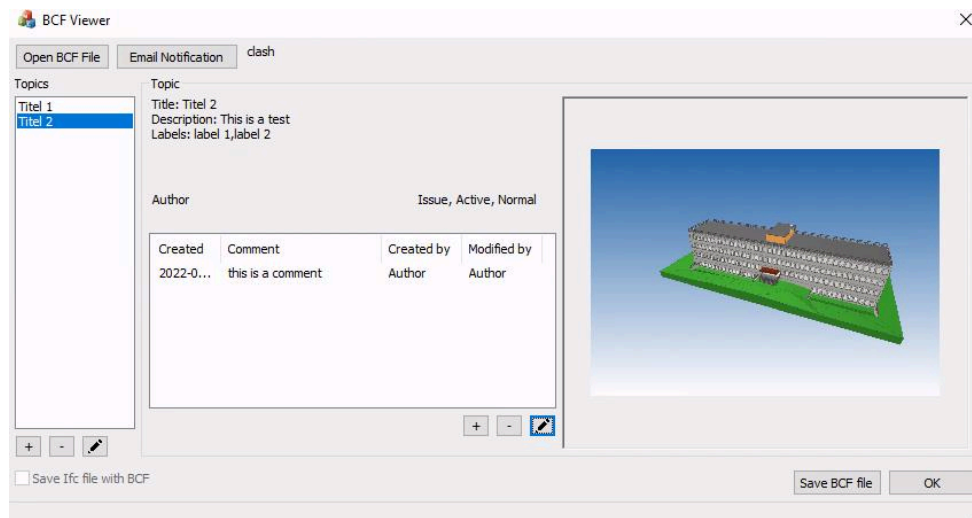


Figure 2: BCFViewer features

Each time a change is made in BCF it is possible to send a notification to any person. In each topic or comment dialog there is a notification section (Figure 3) where you can select who should be sent an email to and a checkbox that defines if the BCF file should be sent as an attachment or not. Since the BCF file has not been saved yet, a temporary file of the current state will be created if the checkbox is activated. The selected people will receive an email with the most important details about the changed topic and the attachment, which can be loaded and visualized either in the BCFViewer or any other program that supports BCF.

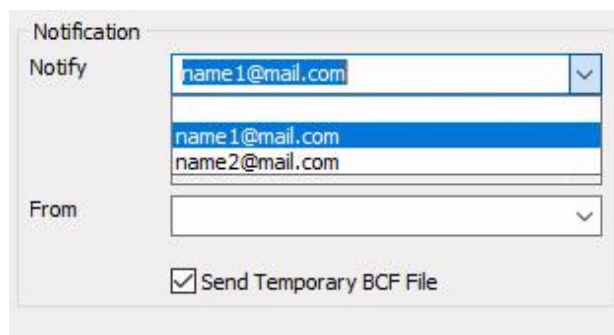


Figure 3: BCFViewer Notification

BCFViewer offers the functionality to save regular contacts to which the user wants to send notifications. The list can be edited in a separate dialog (Figure 3). When the list is saved, an external XML file is created, which is read each time the tool is initialized.

3.2 Future implementation plans

The next step for the BCF viewer would be the editing of the Snapshots images, where drawings on the pictures could be made, and the possibility of adding attachments to each topic. These attachments can be any kind of document that contributes to the understanding of the topic, illustrates it or adds something that was mentioned in the comments. For example, this could be a PDF or an XML file.

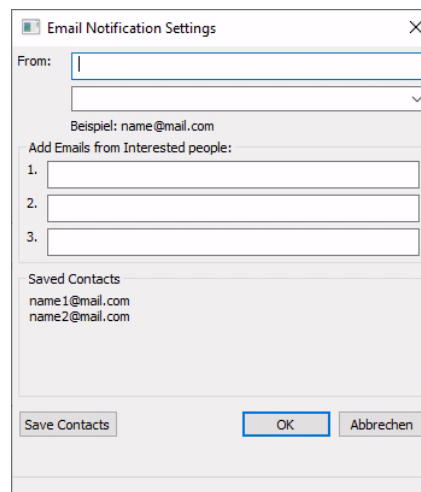


Figure 4: Notification Settings

It is planned for further functions to be implemented to complement the module for the FZKViewer, such as the ability to save the IFC project with the BCF zip file, take screenshots of the current viewpoint, and save a viewpoint file with the appropriate components.

3.3 Comparison with similar tools

There are a number of similar free tools for managing BCF data, and here BCFViewer is compared to three of them: the free version of BIMCollab ZOOM [8], a tool for model viewing and BIM validation that is fully integrated with problem management workflows; BCFier [9], a set of plugins and apps (modules) that handle BCF and integrate directly with BIM tools; and Sloth [10], an online BCF report generator.

BIMCollab ZOOM works with version 2.1 of BCF and creates files with the extension ".bcf". It is possible to add or edit different snapshots and viewpoints and upload IFC files. When dealing with older BCF versions, where just one snapshot per topic was allowed, there is no possibility of replacing it. BIMCollab has a cloud support where it is possible to log in and have an overview of the issues and send notifications. It is not possible to get a local copy of the BCF file once the user is

logged. With the BCFViewer the notifications are sent offline. That means that an account for the app is not necessary.

BCFier is currently composed of Autodesk Revit 2015 and 2016 add-ins and stand-alone Windows Viewer. It creates files with the extension ".bcfzip" and allows the user to have more than one BCF report at the same time. When adding a new view from BCFier Viewer no viewpoint will be added in the view therefore it will not contain 3D information.

The Sloth web-app offers a way to upload a BCF file and see the resulting report in your browser, showing the main information in your issue, along with the first screenshot. Sloth has the feature to export your report to Word or Excel format. It uses basic Styles in Microsoft Word to highlight titles and dates.

Table 1: BCF tool comparison

	BCFViewer	BIM Collab ZOOM	BCFier	Sloth
IFC viewer as Stand-alone	No	Yes	No	No
Read BCF files	Yes	Yes	Yes	Yes
Edit BCF files	Yes	Yes	Yes	No
Upload snapshots	Yes	Yes	Yes	No
Edit Image from Snapshot	No	Yes	Yes	No
Send notification	Yes (offline)	Yes (through cloud)	No	No
Report export	No	No	No	Yes

4 Conclusion

In this paper, the development of a new tool for collaborating with BCF data is presented and it is shown how this tool can be used in a BIM workflow. The features are shown and compared to other similar tools. The BCFViewer provides reading and edition of BCF data, but the lack of an IFC viewer, that is, a lack of 3D graphical representation, can make it difficult to visualize some of the issues listed in the file. Although if the goal is a quick and easy file manager, BCFViewer provides the basic necessities and allows a way to exchange content via Email notifications without further work. It has a simple, self-explanatory user interface and it's more convenient to use than an integrated module within more complex BIM software. There are still some steps to be taken in the development of the

BCFViewer, such as the plan for new implementations as attachments of files to the BCF or editing of the Snapshot image. The IFC interaction integration for the FZKViewer is also in development, so it will be possible for example to save the object components in the viewpoint file or take screenshots. The implementations should however be made without sacrificing the premise of simplicity so that in the end a complete yet user-friendly tool is offered that enhances the workflow.

References

- [1] BuildingSMART International. Available: <https://www.buildingsmart.org/standards/bsi-standards/bim-collaboration-format-bcf/>. [Accessed March. 15, 2022].
- [2] Github, "buildingSMART/ BCF Rest API" [Online]. Available: <https://github.com/BuildingSMART/BCF-API/>. [Accessed 3 September, 2021].
- [3] K. Linhard, R. Steinmann, "BIM-collaboration processes – from fuzziness to practical implementation", in *eWork and eBusiness in Architecture, Engineering and Construction: ECPPM*, Vienna, Austria, 2014.
- [4] Github, "buildingSMART/ BCF-XML/ Documentation" [Online]. Available: https://github.com/buildingSMART/BCF-XML/tree/release_3_0/Documentation/. [Accessed 3 September, 2021].
- [5] O. Schulz, J. Beetz, "Image-documentation of existing buildings using a server-based BIM Collaboration Format workflow", August 2021. [Online], Available: <https://www.researchgate.net/>. [Accessed March 23, 2022].
- [6] N. Trelidal, H. Parsianfar, J. Karlshoj, "Using BCF as a mediator for task management in building design", Proceedings of the International RILEM Conference Materials, Systems and Structures in Civil Engineering, Denmark, August 2016.
- [7] O. Schulz, "Spatialization of damage to buildings using a BIM collaboration Format approach," September 2021. [Online], Available: <https://www.researchgate.net/>. [Accessed March 23, 2022].
- [8] BIMcollab [Online]. Available: <https://www.bimcollab.com/>. [Accessed 20 September, 2021].
- [9] BCFier [Online]. Available: <http://bcfier.com/>. [Accessed 23 March, 2022].
- [10] Sloth [Online]. Available: <https://www.bim42.com/2017/07/sloth/>. [Accessed 25 March, 2022].
- [11] L. van Berlo, T. Krijnen, "Using the BIM Collaboration Format in a server based workflow", presented at 12th International Conference on Design and Decision Support Systems in Architecture and Urban Planning, DDSS 2014.