Journal of Building Engineering 85 (2024) 108647

Contents lists available at ScienceDirect

Journal of Building Engineering

journal homepage: www.elsevier.com/locate/jobe

An OpenBIM-based 4D approach to support coordination meetings in virtual reality environments

Aissa Khorchi, Conrad Boton *

Research Laboratory on Information Technology in Construction (LaRTIC), Department of Construction Engineering, École de technologie supérieure, 1100 Notre-Dame Street West, Montreal, Canada

ARTICLE INFO

Keywords:
Construction virtual reality
4D simulation
BIM collaboration format (BCF)
OpenBIM
Collaboration

ABSTRACT

Notwithstanding the emergence of Virtual Reality technologies, the solutions available on the market for integrating 4D and Virtual Reality are not well adapted to the principles of BIM and only propose certain visualization functionalities. Indeed, generally, such solutions are mainly simple visualization tools which do not offer BIM-based collaborative tools such as BIM Collaboration Format (BCF)-based exchanges. Our research aims to develop a method and subsequent prototype to integrate an IFC-compliant 3D model and a planning schedule in order to perform a 4D simulation in a Virtual Reality environment having BCF-based information exchange capabilities. The prototype developed achieved its theoretical and technical objectives, namely, to validate our hypothesis that the BCF can be implemented in a VR environment and support 4D simulation-based collaboration. The proposed method allows to identify some lessons learned, in terms of success factors as well as areas for improvement.

1. Introduction

While construction significantly impacts a country's economy, it is unfortunately characterized by low productivity, a lack of technological advancement and low efficiency [1]. In a world where information technology is increasingly present in almost all sectors, construction has only just begun working to establish a vision for its 4.0 digital transformation, with a focus on digitization and productivity improvement, as well as the automation of production environments [2]. Given all the capabilities it offers, Building Information Modeling (BIM) is considered as capable of playing a central role in the digitization of the construction industry [2,3]. BIM "uses a multidisciplinary object-oriented three-dimensional (3D) model of the constructed facility in order to improve and to document its design and to simulate different aspects of its construction or its operation" [4]. It adds different aspects to the basic 3D representation, including the representation of the schedule (4D), cost information (5D), energy consumption information (6D), and operations and asset management (7D). The use of 4D simulation has led to significant improvements in the planning and execution of construction projects [5]. Another emerging technology, Virtual Reality (VR), allows visualizing the 3D model in an immersive environment. The technology is increasingly adopted and applied in various industries. Many research works have explored the use of VR to visualize BIM models and 4D simulations [6–11].

Boton [5] proposed a framework to support coordination meetings by the combined use of 4D simulation and VR environments. Coordination meetings are essential to the success of construction projects. Indeed, these meetings are useful to limit errors, change orders and rework during the construction phase, by allowing the project teams to "mutually discover issues, critically engage in

* Corresponding author.
E-mail address: conrad.boton@etsmtl.ca (C. Boton).

https://doi.org/10.1016/j.jobe.2024.108647
Received 22 May 2023; Received in revised form 21 January 2024; Accepted 24 January 2024
Available online 1 February 2024

clarifying and finding solutions to the discovered issues" [12]. With the increasing complexity of construction projects and the heterogeneity of project teams, the use of virtual coordination meetings offers two major advantages: 1) allowing distributed teams to efficiently conduct BIM-based coordination meetings; and 2) taking advantage of the innovative visualization and interaction mechanisms offered by new virtual tools. However, 4D simulation in VR environments remains limited to being a means of visualization, and does not actually integrate data exchange and interoperability between VR environments and BIM applications [7]. The rare solutions that allow collaboration based on BIM models in VR are often tailor-made solutions that support a particular software vendor, which makes them less effective for collaboration with stakeholders not using the same solution provider [13]. The BCF format is one of the dominant BIM collaboration and coordination tools and considered alongside the Industry Foundation Classes (IFC) format as one of the main pillars of OpenBIM [14,15].

Technically, in the construction industry, VR environments are mostly considered end-of-the-chain environments [5,16,17]. In other words, they are mainly used for visualization purposes and no two-way link is established with the design tools. Of course, some notable research works have explored the bidirectional link between BIM and VR [3,18]. However, these works do not explore 4D and VR to support coordination meetings, and are limited to the use of the IFC format, without the integration of comments, annotations and points of view conveyed in a BCF. However, the strength of using BIM to support coordination meetings lies not only in the problem visualization capabilities, but also and above all in the possibility of the effective transfer of issue information (3D geometry, viewpoints, annotations, comments, etc.) between the VR environment and the design software for correction. It is therefore becoming important to combine the use of 4D in VR environments, with an effective possibility of using the BCF format, to ensure a two-way link between the coordination environment and the design software. Unfortunately, despite major advances in the use of IFC format in VR environments, it is not possible today to combine 4D and BCF in a VR environment.

The research question justifying the work presented in this article is therefore the following: How can we efficiently generate BCF files as part of a 4D simulation in a VR environment, to reuse them in a BIM design software? The objective of our research is to develop an approach based on OpenBIM to support 4D collaboration in VR environments. The approach must allow to propose a neutral framework to (1) generate and extract BCF files directly from the VR environment, (2) ensure that the generated BCF files can allow collaboration with BIM design tools, and (3) validate the results from the industry perspective.

This paper is presented in seven main sections. In section 2, a literature review presents the related works, including BIM applications and 4D simulation in construction, VR in construction, and OpenBIM and VR-based collaboration. Section 3 presents the research methodology, inspired by Design Science theories. In section 4, the results of the research are presented, including the proposed artifact encompassing both a method and a tool to support it. In section 5, we present the evaluation of proposals, which is an important aspect of research in Design Science. In section 6, before the conclusion, the paper presents a discussion that highlights the added value of our proposal as well as limitations and areas for improvement to be explored in future research. Section 7 concludes the work.

2. Literature review

2.1. Virtual reality in construction

VR is defined as a medium comprised of interactive digital simulations that capture the position and actions of the participant and replace or augment feedback to one or more senses, which generates the feeling of being mentally immersed or present in a simulated world, a virtual world [19]. A more technical definition, which is more expansive, refers to an integrated collection of hardware, software and content created for the user experience. As described by Fuchs [20], the purpose of VR is to allow a person (or more) to experience an immersion. In other words, to carry out a sensorimotor and cognitive activity in a digitally created world that can be "imaginary, symbolic or a simulation of certain aspects of the real world" [20]. According to Sherman & Craig [19], five main elements should be combined in an effective VR session: the participant, the creator, the virtual world, immersion and interactivity.

Some studies have covered the use of VR in the field of construction, as cited by Bouvier et al. [21]. The research studies carried out in this context include an examination of the simulation of human behavior when designing a building's security and fire systems [22]. Getuli et al. [23] have evaluated the human behavior heatmap in a construction site to identify the safe zones around handling some materials and activities. That would improve the health and safety on a construction site. Chen et al. [24,25] have simulated fire fighter intervention in interior fire hazards to improve the effectiveness of such interventions. Other researchers have found that the use of VR in safety training is more effective due to its immersion, and trainee involvement as stated by Afzal et al. [26]. There is also the use of VR in construction project management, specifically for the effective management of site space, using simulations to improve health and safety [23]. More and more research are being conducted in the field of VR for building engineering education, as mentioned by Wang et al. [27]. BIM and VR can also be leveraged in the design phases to assess cost and emission footprint as it helps the decision-making process without compromising the esthetics as stated by Kamari et al. [28]. Among the challenges of VR in the construction industry is the difficulty simulate a real construction site with all its risks and opportunities, but rather some hard coded scenarios which is limiting. In addition to the complexity of creating such applications as stated by Schiavi et al. [29]. Alizadehsalehi et al. [30] identified a lack of interoperability with different systems and exchange formats. This issue has also been highlighted by other authors [5,7]. The best way to approach the issue seems to be the use of open OpenBIM formats, in order to guarantee better freedom of use and optimal interoperability between tools and software developers [7].

2.2. OpenBIM and VR-based collaboration

2.2.1. Definitions and concepts

OpenBIM is defined by buildingSMART [31] as a collaborative process that is software vendor-neutral, consisting of shareable project information that supports effective collaboration for all project participants. It facilitates interoperability to benefit projects and assets throughout their lifecycle. To allow efficient and open collaboration, OpenBIM is based on 6 fundamental principles: interoperability, the openness of the standard and its neutrality vis-à-vis developers, reliability of data exchanges, collaboration improved by agility and support for multiple data formats, flexibility in the choice of technology by stakeholders, and sustainability by supporting long-term data standards [31].

The IFC format is a key component of the OpenBIM concept of BuildingSMART. IFC is not only an exchange format, but rather, a schema, a data structure or a specification [32]. This schema is used for organizing and transporting computer data. It can be expressed in different formats, with the most common being the IFC-SPF, which is a text format in the EXPRESS language known for its reduced file sizes [32]. It also exists in the IFC-XML format, which is less used, and finally, in the ICF-ZIP, format which is a compressed format [32]. The IFC is an object-oriented specification and describes definitions of objects. These can be real (physical) objects from the building world, such as a wall, a column, a door, etc., or abstract objects, such as a role, process, control, etc. In addition, the IFC format defines the relationships between objects (a door that is linked to a wall and is on a given floor). Moreover, the IFC defines the properties (or attributes) of objects, namely, their height, width, materials, supplier, etc. [32]. Objects, relations and properties represent the root concepts of the IFC format, and they are in turn divided into sub-concepts [32]. While the predominant approach for transferring BIM data between BIM design platforms relies on the use of the Industry Foundation Classes (IFC), modern game engines like Unreal Engine and Unity 3D do not natively support IFC, necessitating third-party libraries for IFC file analysis [33]. Further some research works have explored alternatives such as the FBX data model, natively supported by game engines, offering diverse perspectives for the effective integration of BIM data in gaming environments. Thus, it is possible to use third-party applications (e.g. Tridify) to convert BIM models from the IFC format into more game engine-compatible formats, without the risk of losing non-geometric data [33]. This enhances the overall experience by providing relevant information that can potentially be visualized or used to improve the interactions or functionality of the model in its new environment.

The BCF is a format used for collaboration and exchange of communications in BIM projects; it is based on an Extensible Markup Language (XML) schema that can be read by a human [34]. The BCF format, adopted by BuildingSMART as an OpenBIM format, aims to allow the communication of 3D coordination issues in a model from one BIM tool to another while separating communication information from the model. Being in XML format, a BCF file has a very precise tree of elements structure, consisting of a hierarchy of attributes and other tags. Each attribute is a set of keys and values allowing information to be encapsulated in a human-readable text format. Thus, the BCF format is organized according to topics, each topic representing an issue. The topics are organized in folders, each with a unique Globally Unique Identifier (GUID) or Universally Unique Identifier (UUID), which is a series of 128-bit coded numbers used to generate unique identifiers in a decentralized system based on a few components of a computer [14].

Each topic contains BCF-XML files named markup.bcf, an optional file, viewpoint.bcfv (bcf viewpoint), which contains the coordinates of the viewpoint. It thus allows the relocation of the user to the same position in the model, regardless of the BIM software

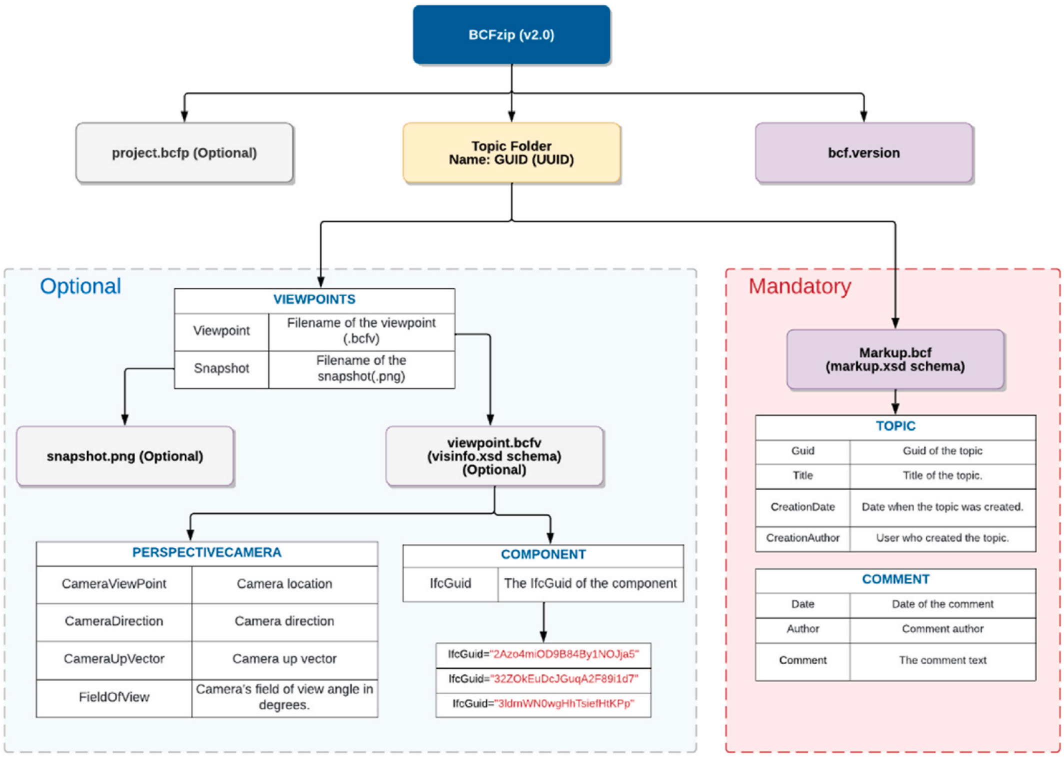


Fig. 1. Explanatory hierarchy of a BCFzip file and the BCF format.

used. In addition, it allows locating ".png" files that represent screenshots of the issue [14]. A BCF file can encompass multiple folders representing various topics, compressed into a single.bcfzip file. Based on the buildingSMART documentation for the BCF v2.0, we provide a diagram explaining the components of the BCF (Fig. 1).

2.2.2. OpenBIM-based 4D collaboration in VR environments

VR is used mostly as a visualization medium [16,17], and extracting data or information from this environment into other collaboration solutions is still rarely done. In fact, much research has been devoted to the interoperability between BIM tools and VR

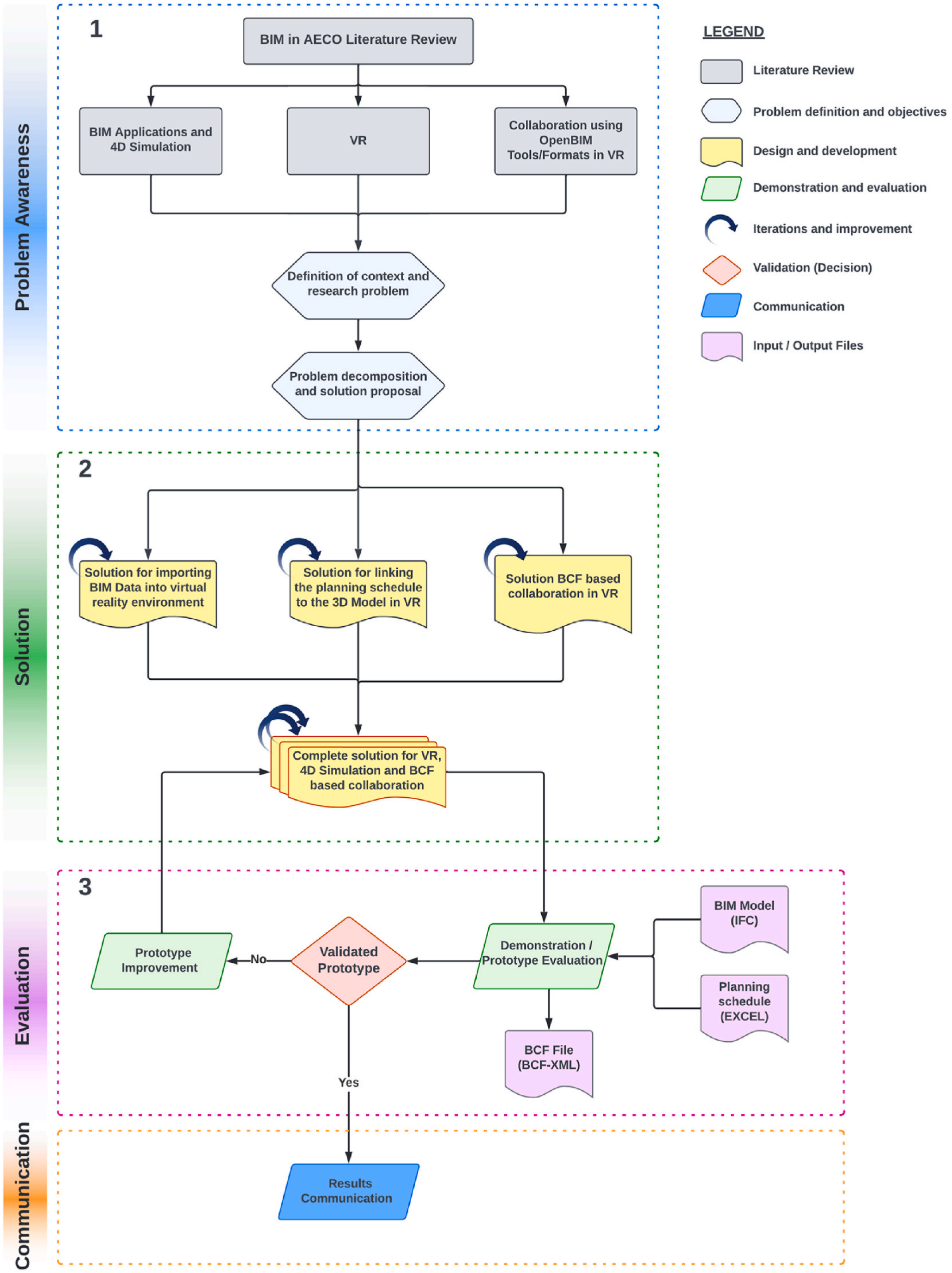


Fig. 2. Main steps of the research project.

environments [35]. However, most of this work has addressed this interoperability in one direction only, i.e. the transfer of information from BIM tools to VR environments, including using the IFC format [3,22,23,35–37]. One can note that most of these studies focus on a one-way information transfer from BIM tools and VR environments. One of the most significant works related to our research issue is the research presented by Nandavar et al. [3]. They proposed an OpenBIM-based prototype that makes it possible to integrate BCF issue mark-ups and to create some IFC objects, with appropriate semantic relationships, directly from the VR environment [3]. Although this work represents an important milestone in the integration of 4D BIM and VR, it does not address the specific issue of 4D simulation. Indeed, the integration of 4D in VR environments does not only concern IFC objects and their properties. It also concerns timeline management and the proposal of suitable mechanisms enabling users to interact with the temporality of construction activities planning, including the analysis of schedule alternatives and the dynamic visualization of site logistics.

Some notable research works have addressed the use of 4D simulation in VR environments [38–41] since the seminal works from Stanford University [42]. However, very few works have been dedicated to the bidirectional link between BIM tools and VR environments. In 2019, Dreischerf and Astour [43] propose a draft concept for creating a bidirectional link between BIM 4D planning and VR environments. Unfortunately, the proposal is not clearly formalized, and no application or evaluation is proposed.

Based on the literature review, we completed the characterization of the problem and the motivations of our research work. Depending on the context, some difficulties were identified in the literature regarding existing solutions. Among the issues raised are the inefficiency in creating 4D simulations and the difficulty of passing schedule data and BIM information related to the model in a VR environment. Once these difficulties were overcome, we saw another challenge, namely, the communication of problems identified during collaboration in a VR environment to another collaboration platform via an OpenBIM exchange format such as BCF. Interoperability to integrate VR systems with other non-immersive collaboration solutions also remains very limited.

3. Research methodology

3.1. *A research approach inspired by Design Science theories*

The present research aimed to develop the use of 4D simulation in a VR environment in order to support constructability studies based on openBIM interoperability formats such as IFC and BCF. To this end, we used a methodology based on Design Science theories. Design Science is a research approach in which the researcher creates an artifact to resolve identified problems while adding value to the body of knowledge [44]. The fundamental principle underlying this methodology is that the design problem and its solution are fully understood during the artifact design and application process [45].

Peffers et al. [46], who compared seven key pieces of research using Design Science, identified five key steps of this methodology, namely, the motivation and identification of the problem, the definition of the solution objective, the design and development, the demonstration and evaluation, and the communication of the results. In all these steps, it is required to know the state of the problem (state of the art) and the existing solutions and their effectiveness, to have a good knowledge of the theory to bring a new solution, as well as techniques for using the artifact (proposed solution) [45].

3.2. *Steps and tools used in our research methodology*

3.2.1. *Main steps*

Fig. 2 summarizes the four main steps of our research methodology: understanding the needs, development and design of the prototype, demonstration and evaluation, and finally results communication.

The artifact, i.e., a prototype to meet the objectives listed above, was developed by dividing the functionalities of this tool into modules consistent with the objectives to be achieved. In addition, we divided our process into pre-VR preparation stages (data entry into the 3D BIM model, preparation of a planning schedule), VR stages (running the 4D simulation, creation of comments and annotations), and post-VR (reuse of data and comments extracted from the VR environment). Fig. 3 summarizes the theoretical framework underlying the proposed prototype.

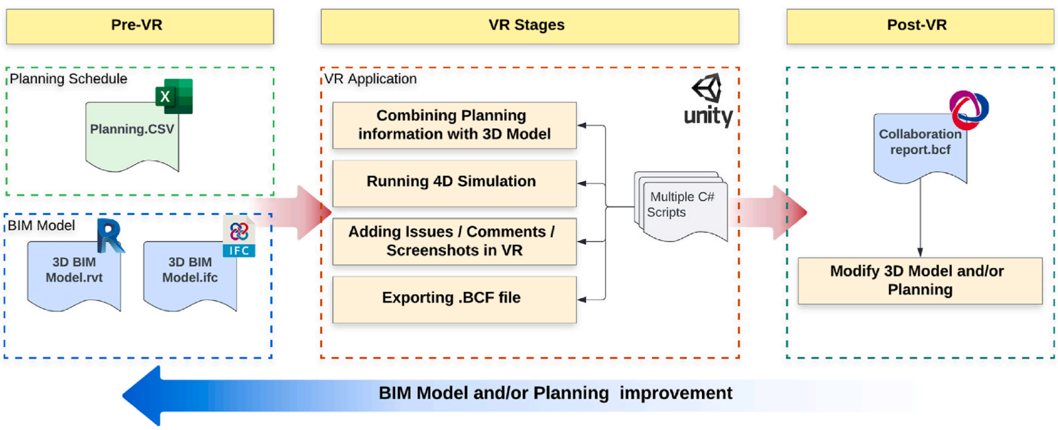


Fig. 3. Inputs and outputs of the proposed solution.

For the evaluation of the designed tool, we had it tested and evaluated according to predefined criteria based on the identified needs: (1) the ability of the system to import and execute a 4D simulation in VR and (2) the creation and effective use of a BCF file exported from the tool outbound through other collaboration solutions. Thus, the tool was evaluated according to the following criteria:

- Navigation and projection of the 3D model in the VR environment
- Import of the BIM model data and linking to the corresponding objects
- Operation of 4D simulation in the VR environment
- Addition of “topics” and comments in order to produce BCF files
- Output BCF files readable by other BCF solutions (interoperability)

Moreover, the application was tested by a third party with extensive experience in the field of engineering and construction, in addition to a significant academic background in BIM and VR. Their feedback helps us evaluate the tool against the criteria mentioned as well as the usefulness of the tool vis-à-vis the construction industry.

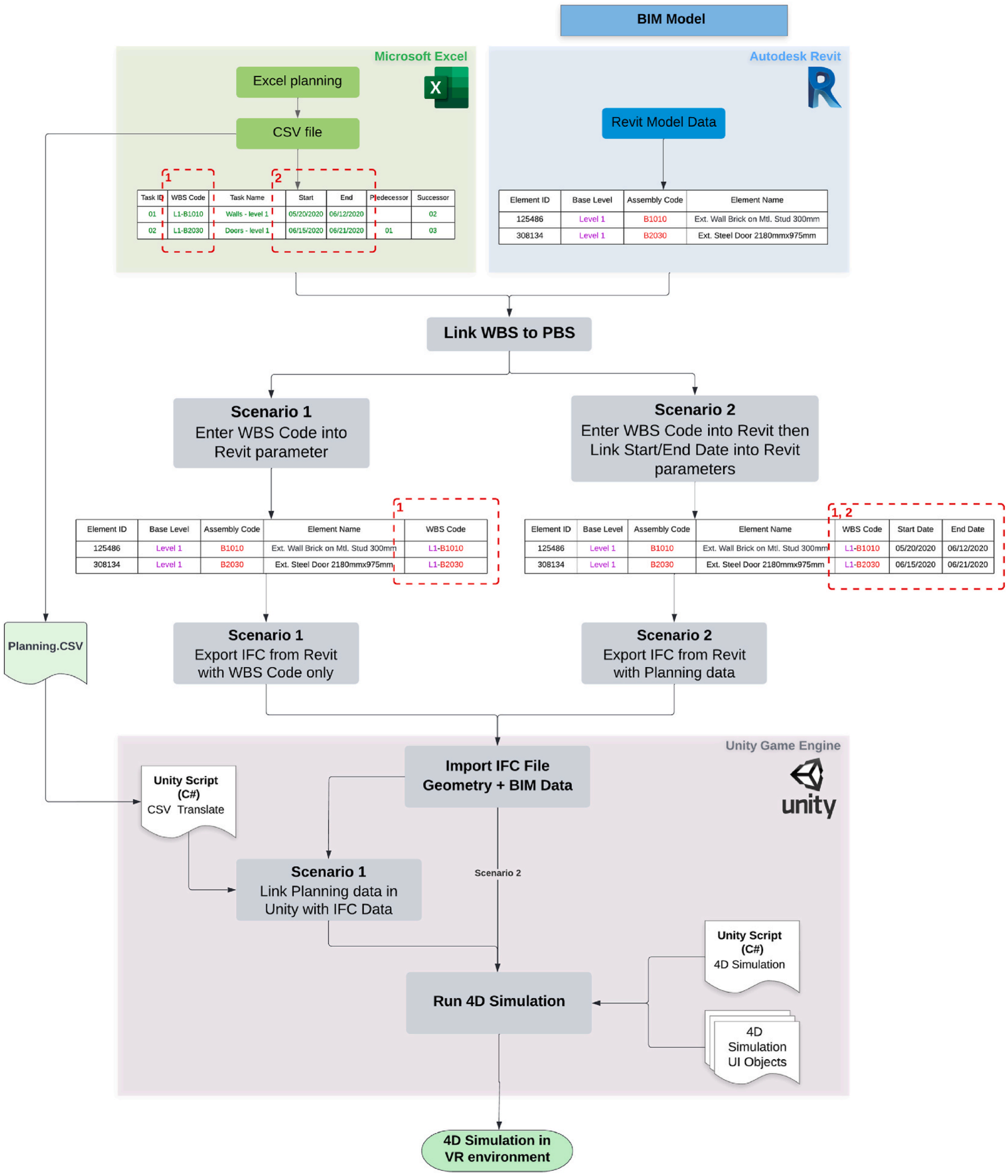


Fig. 4. Processes considered for transferring data into a VR environment.

3.2.2. Used tools

For the research project, different tools have been used, including commercially available BIM software and solutions. Autodesk Revit was used to model the building model used to test our prototype. Revit is available free of charge to students and teachers and allows to export in different formats, including the IFC format, thus providing some advantages in terms of interoperability. For construction planning, we used the Microsoft Project Professional and Microsoft Excel. Our example was simple and theoretical, and therefore, in order to simplify the transfer and processing of planning data (activities), we opted for a schedule made with Excel.

Regarding VR and the creation of applications, we chose a video game engine, Unity 3D, given the availability of its documentation and features and its wide support for VR head-mounted displays, as well as its use in the construction industry [47].

Finally, for the VR experience, we went with the Oculus Quest head-mounted display, which was already available at our research laboratory.

4. Proposed method and prototype development

We now present the proposed method and the subsequent prototype developed to address the issue of collaboration based on 4D simulation and OpenBIM in a VR environment. Particular emphasis is placed on the IFC and BCF OpenBIM formats, given their interoperability advantages across software applications available on the market.

In this section, we will focus on the three main challenges illustrated in Fig. 3: the preparation and the transfer of BIM and schedule data, the development of the 4D simulation module, and the development of the BCF module and its use in the VR environment.

4.1. Preparation and transfer of data into the VR environment

4.1.1. Preparing the model and data transfer

To perform a 4D simulation, it is essential to link the 3D model with the schedule, and therefore, certain elements of the 3D model will be attached to an activity which will determine their start and end dates, as well as the duration of the activity. Subsequently, showing or hiding these elements according to the current date will allow seeing the evolution of the construction of the building over time. The schedule has a key element, the Work Breakdown Structure (WBS) code, that serves as a unique identifier for the different tasks. According to our example, this code is composed of the level and a Uniformat code (L1-B1010: Level 1 – B1010: Shell > Superstructure > Floor Construction). This coding system is still a proposal and can be adapted according to the type, complexity and standards of the project. The 3D model in turn will have a WBS Code (parameter added to each element of the model).

We evaluated two scenarios to transfer and combine the 3D model and schedule data into the VR environments (Fig. 4Fig. 4). The first scenario (scenario 1) considers entering a single shared parameter in Revit, which represents a text-like attribute that defines the WBS code for each element in the model. This attribute will then be used as a reference to link each item to the corresponding activity in the schedule. Once the model and schedule have been transferred to the VR environment (Unity), we create a code to read and transform the different activities of the schedule directly from the CSV source file in order to link them to the models. Subsequently, a C# script to run the 4D simulation will support the linking of the WBS and the Product Breakdown Structure (PBS). The second scenario (scenario 2) also considers adding 3 shared parameters, a Start date, End Date in addition to the WBS code to each model

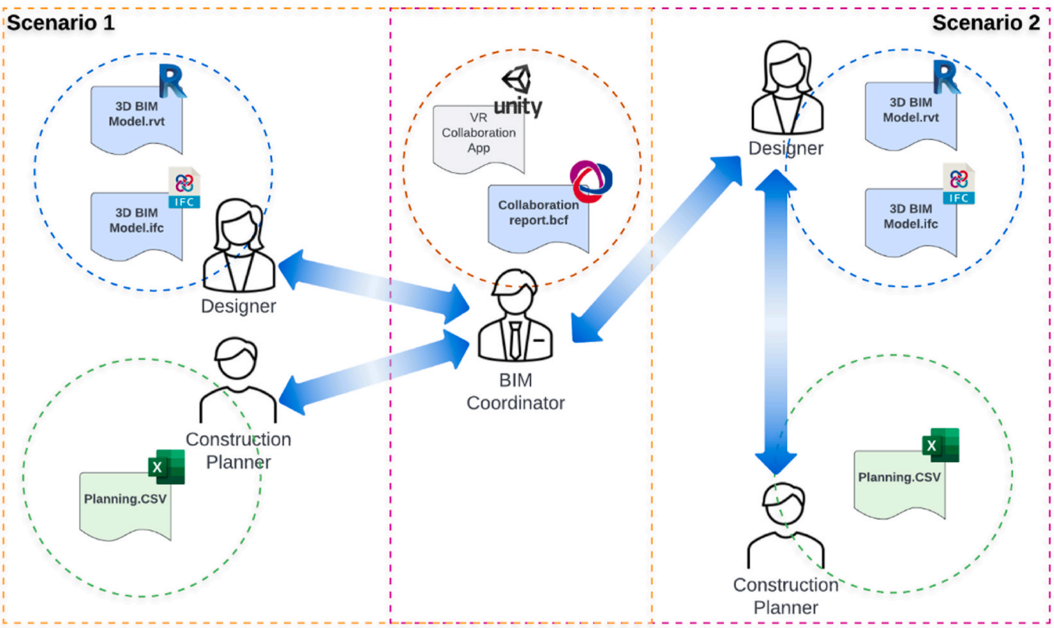


Fig. 5. Comparison of the communication flow between both scenarios.

element. Secondly, we import the Start date and End date fields for each element according to the corresponding WBS code. For example, all the exterior wall elements of level 1 will have L1-B1010 as the WBS code and therefore their start and end dates will be loaded from the schedule directly into the Revit model. This can be done manually (via arrays of elements) or automatically, based on a Dynamo script. The main issue with this method is the high probability for the start/end dates data to be converted to text instead of date format, and then not useable as temporal data in Unity.

After evaluation, scenario 1 was preferred as it requires less preparation of the BIM model before moving on to coordination. Indeed, each parameter model preparation requires additional time to collect at the level of the designers (architects, engineers, etc.) and can increase the level of complexity of the process. The coordinator plays the central intermediary role and ensures compliance with established standards. This way, construction planners no longer need to communicate any changes of dates to the designers but only to the BIM coordinator. Fig. 5 shows the flow of communication between stakeholders in both scenarios.

Shared parameters in Revit are parameter definitions that can be added to the project or project families and saved in a project-independent text file. These parameters are useful for defining an attribute to all the elements of a project. For example, in our case, we defined a shared parameter named ‘WBS Code’, which represents the activity in which the element in question will be built. This parameter will be displayed in the list of properties of each element of the 3D model, allowing necessary information to be filled in this field, as shown in Fig. 6. WBS Code parameter values can easily be filled and checked from a multi-category schedule as well. This table also allows to validate the integrity and consistency of the data entered at the item level. After this validation, the model is ready to be exported to IFC format.

4.1.2. *Importing the model and data into a VR environment*

Multiple options and tools exist to import a BIM model, including all relevant element data, into a VR environment. In this research work, we did not pursue any laborious coding activity for this phase, as solutions already exist to this effect. The solutions we tested are PiXYZ, Unity Reflect, Arcventure IFC Importer and 3D Repo, but none of them were chosen for reasons of accessibility, cost and non-availability of an educational license as well as functionality limitations. We opted instead for Tridify, a cloud-based collaborative solution that allows to share and view IFC files directly on an Internet browser. It also allows BCF file-based collaboration. In addition, IFC files can be imported into Unity, including files containing all BIM information. We started by uploading our IFC file exported from Revit after preparation for the 4D simulation to the Tridify server. Once the IFC file was loaded into the Unity project using the Tridify plug-in, we inspected the model and the BIM data, specifically, the “WBS Code” parameter that we added to the model as the parameter is used to link the elements of the model to the schedule (Fig. 7).

It is important to note that Tridify is a commercial solution that is not open source. In this sense, it is of less interest in terms of openBIM than a solution such as IfcOpenShell. However, for the purposes of our study, Tridify has the advantage of providing

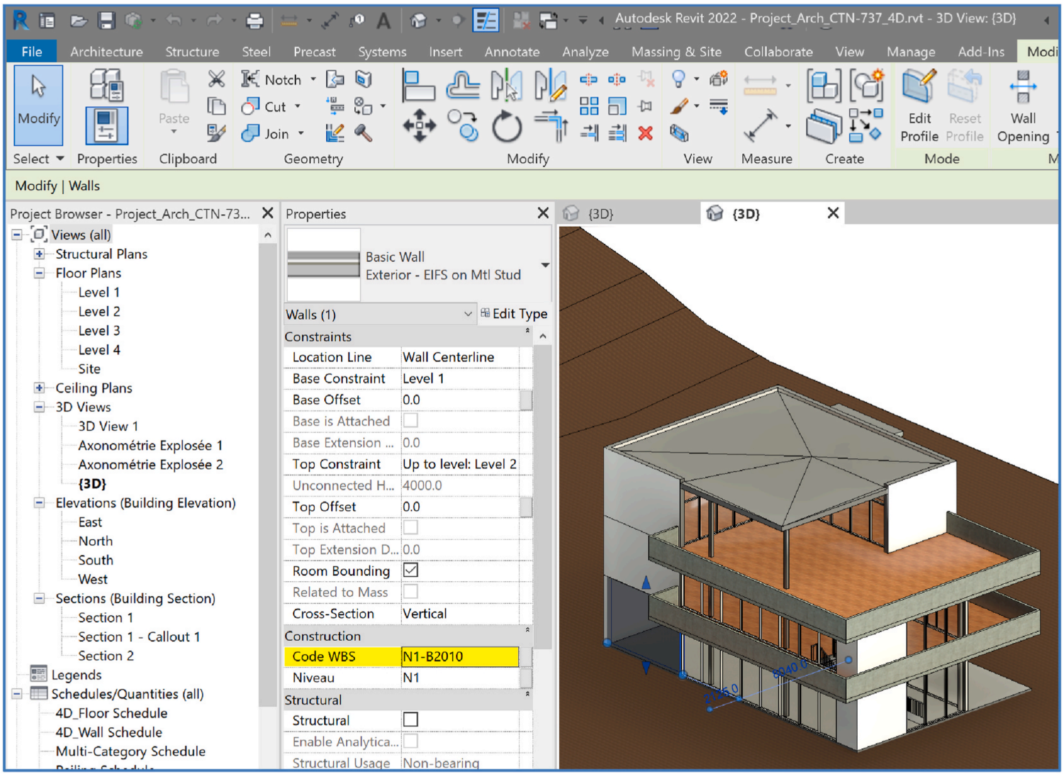


Fig. 6. WBS code shared parameter in element properties.

comprehensive and easy-to-use IFC-based transfer functionalities, enabling a proof-of-concept to be achieved within the project timeframe.

4.1.3. Importing schedule data into unity

Once the IFC model and the associated information are successfully imported into Unity, the next step is to import the schedule data. The schedule was simplified to an Excel file in Comma Separated Values (CSV) format. Clicking the “Import CSV” command in the application’s main menu launches the schedule data import. This user input in effect launches the *ImportTaskPlanFromFile* method (C# function), which consists of the following steps: 1) Creation of a dictionary with a Key (WBS Code) structure; 2) Clearing of the list of tasks in our application if tasks have already been imported; 3) Reading of the CSV file located in the “C:\BCFprj\plan.csv” path; the path can be modified to target the location of the planning file; 4) Reading of the lines of the CSV file and then separating the columns using the comma separator “,“; 5) Filling of the dictionary with the values of WBS Code (column 1) and StartDate (column 5), and 6) Instantiation of the task Prefab in the space designed for this purpose (the Prefab is shown in Fig. 8), and then displaying information taken from the CSV file in 5 main columns in the text fields of this Prefab.

4.2. Development of the 4D simulation module

Other operations are necessary to make this data functional in 4D simulation. We followed the following steps in coding the 4D simulation module:

- Definition of the start and end dates of the project according to the schedule by comparing each date in the *Start Date* column and defining the earliest date (start).
- Determination of the total duration of the project, based on the two dates calculated in step 1 (End Date - Start Date).
- Configuration of the *time slider* element to reflect the imported CSV schedule.
- Configuration of the *Play*, *Pause*, *Stop*, -1 Day, + 1 Day, and *Speed* buttons.

In order to show or hide elements according to the current date (a core principle of 4D simulation), we implemented functions to link 3D models to imported schedules. The first function runs when launching the application. In Unity game engine, the code should be under the public *void Awake()* function:

- Find the parent object with the name “model3D";
- Find the *Renderer* component that controls the rendering and display of the subobjects of the parent “model3D” and disable their rendering in the game, i.e., make them invisible;
- Access the *Tridify.IfcPropertySet,* which are components containing the BIM information of the imported IFC file. The problem encountered in this step is that the number of *IfcPropertySet* components can vary from one type of element to another, and additionally, the “WBS Code” parameter can be listed under different *IfcPropertySet* components. We developed a series of nested “*for loops*” to search for the parameter “WBS Code” in the different components (Fig. 10). As an example, the following is the procedure on a Wall element (Fig. 9); the WBS code is in *IfcPropertySet* group number 2 of the 15 *IfcPropertySet* available for this

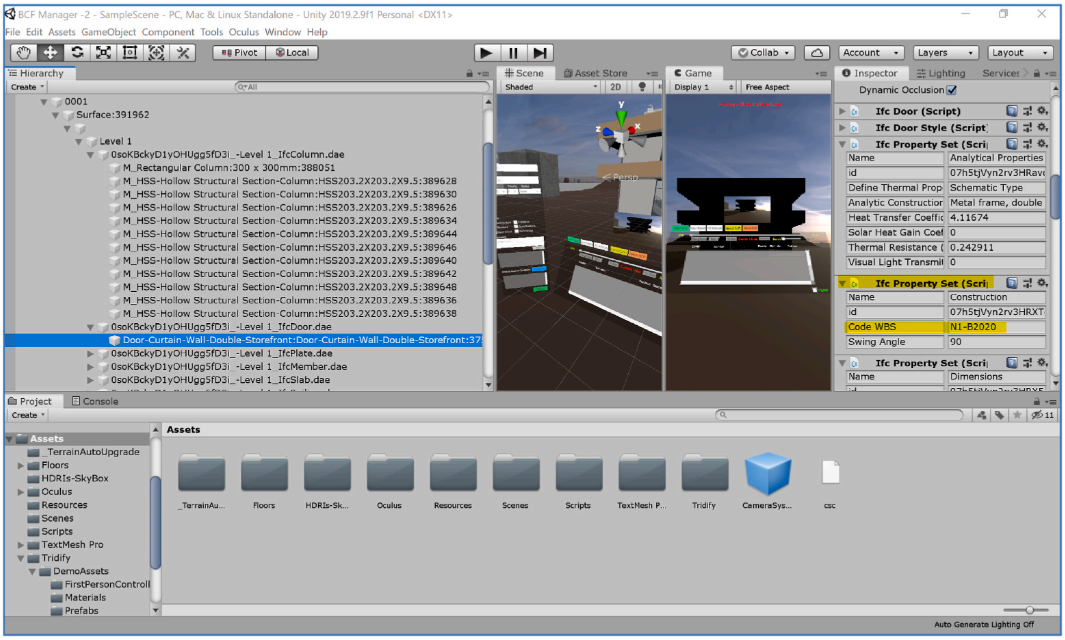


Fig. 7. Validating transfer of “WBS Code” parameter from Revit to Unity.

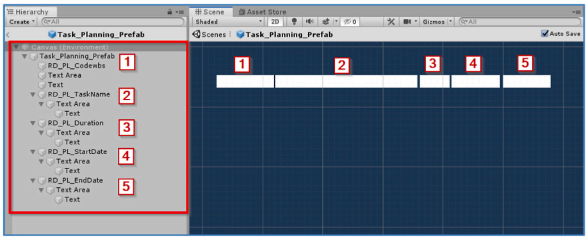


Fig. 8. Visualization of the items of the Prefab Task_Planing_Prefab

object. In another element, this parameter can be found in the *IfcPropertySet* number 6, for example, hence the need for "for loop" functions.

•Rename the 3D objects according to the value of the "WBS code" attached to them in *IfcPropertySet*. We opted for this technique in order to facilitate the selection of the appropriate elements for a task based on its WBS code and then display/hide these elements according to the current date (Fig. 11).

Once the previous functions are successfully executed at application launch, another function has been added allowing objects to be linked to start and end dates of their respective tasks in the schedule. The current date is defined as a flexible date ranging from the start date of the project to its end. Its value is taken based on the timeline slider position which in turn is read based on user input. A "for each" function is used to verify the start date value from the combinations of the dictionary < Code WBS, startDate>. If its value is less than or equal to the current date (currentDate), we add the objects with the same name as the combination key i.e., "WBS code", to a list. Thereafter, we will control the display of the objects belonging to this list and make them visible.

To sum up, the functionality of the 4D module we developed was mainly based on renaming the 3D objects of the model according to the "Code WBS" parameter entered in Revit and imported into Unity with the Tridify tool. Then, the 3D objects were shown and hidden according to the corresponding start dates in the schedule versus the current date. The current date represents the date between the initial start date (day 0) and the end date of the project.

4.3. Using the BCF format in the VR environment

In order to integrate the BCF format into the Unity game engine, we proceeded with the technical understanding of the BCF format, which is defined by schema files allowing to generate XML files. The schema file markup.xsd, which is available in the GitHub directory of the BCF format, indeed contains all the definitions of the variables and classes of the markup, which is a key file containing all the "issues" and "comments" of the BCF.

We proceeded by converting the XML Schema Definition (XSD) schema to C# classes, as it is the programming language supported by Unity. The conversion was automatically done using the tools integrated in Microsoft Visual Studio. The generated class file does not allow to create BCF files, but defines the different variables and information needed for the creation of the BCF format. Converting the schema to a C# class generates around 1700 lines of code. In the scope of our project, which aims to integrate the BCF format, we focus on the mandatory fields and parameters in the BCF format, and secondly, we integrate some optional parameters.

4.3.1. Designing the BCF input form

Following our analysis of the components of the BCF format, we chose to integrate the following components for a "topic". A "topic" is a problem detected during collaboration that we would like to document:

- The name of the "topic"
- The description of the "topic"
- The type (a predefined list: Comment, Issue, Request, Solution)
- Priority (a predefined list: 1, 2, 3, 4)
- The status (a predefined list: Open, In progress, Closed, Reopened)
- Labels (a list based on toggle buttons allowing to choose one or more disciplines; the list is predefined and consists of: Architecture, Structure, Mechanical, Electrical, Specifications, Technology)
- Comment (Comment): one or more comments can be attached to this "topic"
- Screenshot (Snapshot): screenshot of the user's virtual camera, which is attached to the comments. We can have one or more captures per comment

A BCF input form is designed to serve as an interface between the user and the application. All information entered is saved in temporary memory until a topic is saved. Afterwards this form is reset in order to fill in another "topic" (Fig. 12).

4.3.2. Data entry

Entering data into a form in a classic environment, using the screen, the mouse or the keyboard, is generally simple. However, this task is filled with challenges in a VR context. For entry in a VR environment, we explored different avenues, such as voice recognition, which is complicated to implement in Unity and is generally based on other solutions from Microsoft or Google, and sometimes requires a continuous connection to the Internet. For our project, we went with a simpler solution: the visual keyboard [48]. It consists of

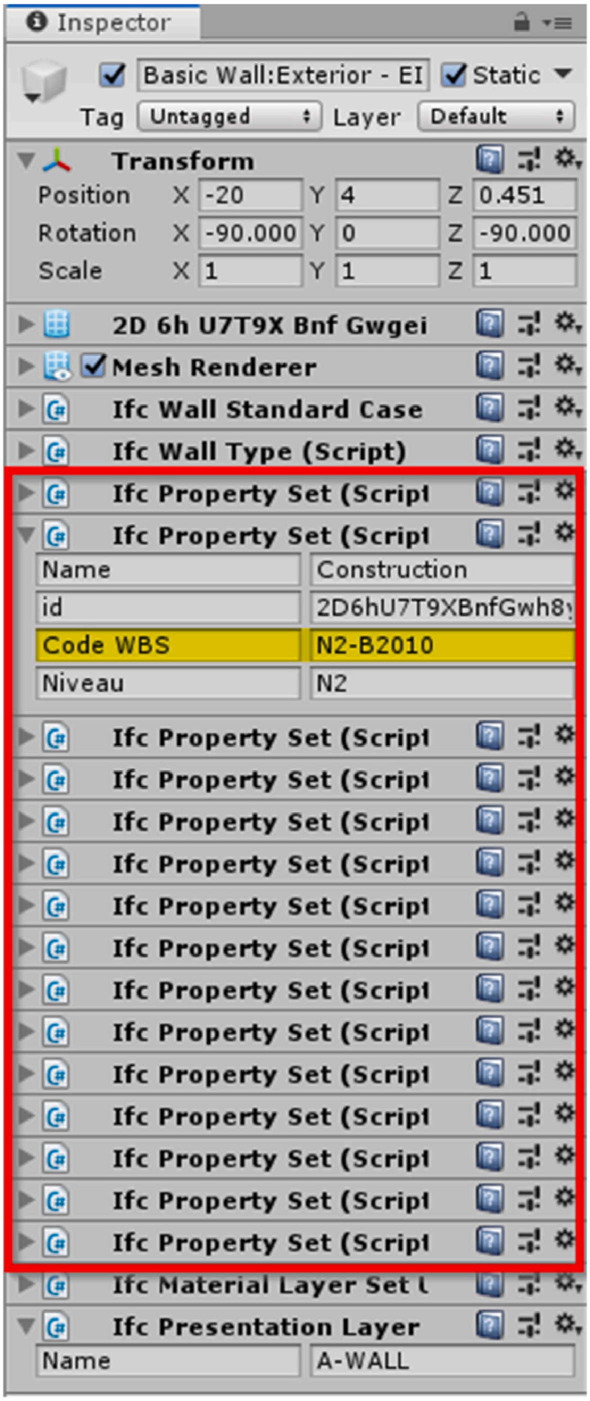


Fig. 9. IFCPropertySet and "WBS Code" attached to a 3D object in Unity.

the letters A-Z, numbers 0–9, and basic punctuation marks (comma, period, dash, colon), plus space and backspace buttons. The keyboard can be hidden or shown using a specific button. The application code includes a function to automatically show the keyboard in case the user clicks on an input text field. Each of the keys on the keyboard is a button, and the interaction is through a laser pointer which is programmed with the Oculus interaction module (the head-mounted display chosen for our project).

4.3.3. Screenshots of identified conflicts

One of the key elements of the BCF format is the possibility of taking screenshots of the model, of the conflict, or of the problem, in order to support a text comment [49]. Albeit being an optional element of the BCF format and its inclusion is not necessary in

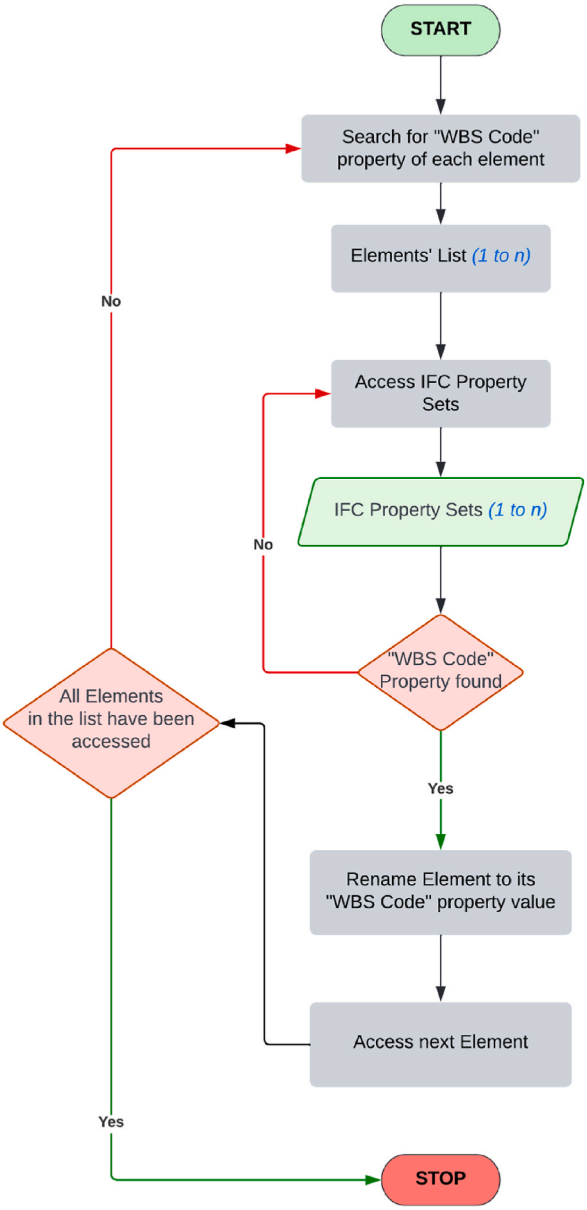


Fig. 10. "For Loop" Diagram for accessing "WBS Code" Property of each element and renaming it to its value.

compiling a valid BCF file according to the standards established by BuildingSMART [49], we find this functionality to be highly pertinent, especially when some annotations have been made, since it provides the BCF file receiver with a visual support for a better awareness of the context of the conflict.

We were able to integrate this feature into our project after overcoming a few challenges. In a desktop application that runs on a screen, it is easier to capture the user's camera image as it has same point of view and resolution as its monitor. However, in VR, the user has two cameras to achieve the stereoscopic effect, and the rendering of these cameras results in a very wide field of view (FOV) of about 130°. This results in a panorama capture in which the object taken in the photos appears very small. We solved this problem by creating another camera attached to the avatar of the user who is manipulated by the head-mounted display in terms of orientation and direction. This makes it possible to have the same field of vision of the user without the peripheral field of vision, thus, a window of the center of the stereoscopic view of the user. We get a camera with a field of view (FOV) of 60°.

In addition, in order to have a rendering of the 3D model only and to exclude the user interface, the virtual joysticks and the laser beams from this view, we excluded all elements classified as "UI" (User Interface) from the rendering of the camera, and subsequently, we assigned the Layer UI to all these elements (Fig. 13).

We also added a window showing the a preview rendering of the capture camera to the user. To help with framing the objects

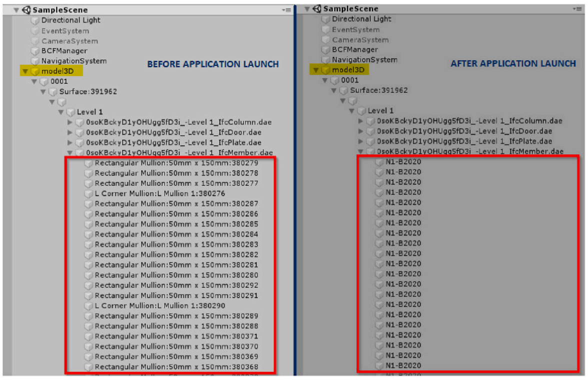


Fig. 11. Comparison of object names before and after application execution.

Fig. 12. BCF data entry form in the application (form framed in red). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

correctly prior to taking the capture, as shown in Fig. 14.

The coding of the screenshot-taking functionality is integrated into the "BCFManager.cs" script file under the Takesnapshot () method, and consists of the following functionalities: 1) Establish the link between the script and the camera in order to distinguish the camera which provides the visual rendering from the camera aiming to take captures; 2) Establish the output image resolution (we

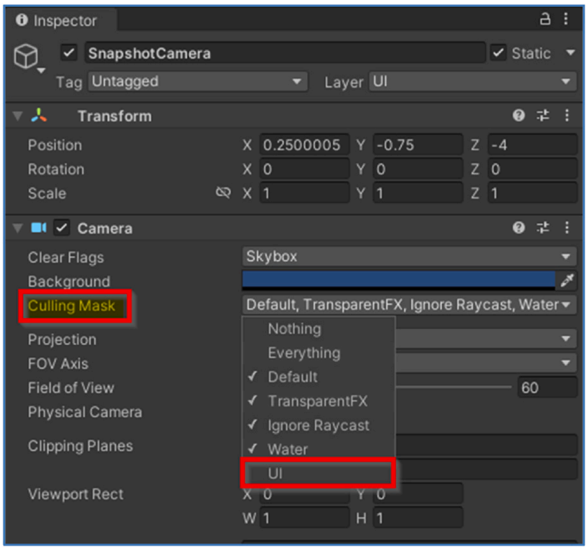


Fig. 13. Excluding the UI layer from the VR screenshot camera rendering.

chose 2100 x 1200 pixels), and 3) Take an array of "bytes" of pixels and encode it in Portable Network Graphics (PNG) format with transparency support.

Once these steps are complete, the screenshot is in the program's memory and will need to be output to an external PNG file according to BCF standards, as detailed in the following function:

- Add the captured image to the list of screenshots (snapshots)
- Identify the capture taken with a GUID
- Give a name to the capture by adding a suffix in order to avoid having multiple captures with the same name (snapshot[n].PNG, "n" being a number ranging from 0 to n)
- Creation of a capture reference file, named "viewpoint.bcfv", as required by the BCF standard [49,50]; this file allows linking the snapshots to the corresponding comments using GUIDs.

In order to add functionality allowing the user to view screenshots taken before leaving the VR environment, we created a visualization space which allows displaying a miniature of the photo taken in a GameObject generated in this space; a button is provided to delete the photo as needed. This function is done with the instantiation of a Prefab. The Delete button, in turn, allows to delete the occurrence of this Prefab and simultaneously delete the capture from the list of captures, and as a result, it does not appear in the output BCF file.

4.3.4. Compilation and BCF output file

To create a BCF file, "Markup.xml" files must be generated, which will represent the "topics". To this end, we created a class definition named "Pmfile", which allows to represent the different elements of the Markup and the functions used to generate the BCF file. The C# code file is named "Pmfile.cs" and contains the following functions:

•Definition of variables needed for markup and topic (Table 1).

- Creation of an instance of the Markup, and in this markup, a new Topic.
- Creation of a new list of comments; the object type is Comment, based on the class definitions of the BCF schema.
- Creation of a new list of screenshots linked to the comment.
- Creation of a list of "viewpoints" corresponding to the screenshots.
- Generation of a GUID for each of the topics created; which will be used in the creation of the subfolders of the BCFzip file.
- Writing of the "Markup.bcf" file in the corresponding folder of the "topic"; for this, we used the XmlSerializer and FileStream functions. We created an output folder in the path: "C:/BCFprj/(TopicGUID)/markup.bcf".
- Writing of "viewpoint.bcfv" file(s) in the same "topic" folder following the same path as in the previous step.
- Writing of the PNG files of the screenshots in the same folder of the "topic" and in the same path.
- Addition of the information needed for each of the files created, such as the GUIDs, allowing to link comments to "topics", as well as the "viewpoints" and "snapshots" to the comments. In addition to that, the name of the author (who is the user of the current session) and the date of creation based on the local time of the machine.

4.4. Interaction in the VR environment

Interaction is a very important phase of our development because it directly impacts the user experience.

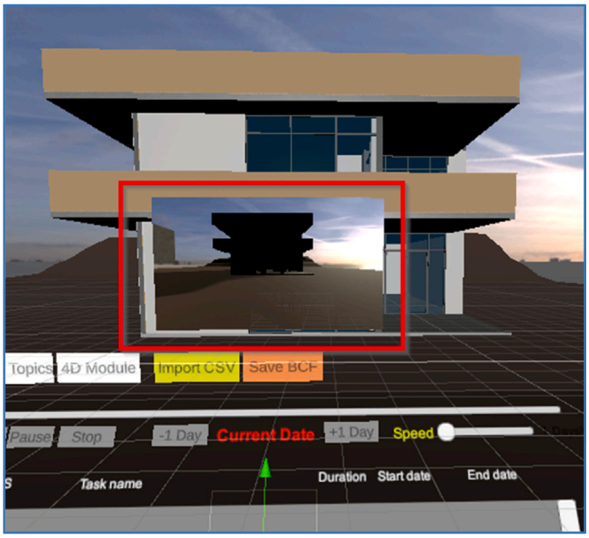


Fig. 14. Snapshot camera preview window.

Table 1
BCF markup variables with their data types.

Name of variable	Name for user	Type of data
atopicguid	GUID of the *topic*	String (Text)
ttopicstatus	Status of the *topic*	String (Text)
btopictype	Type of the *topic*	String (Text)
ctopictitle	Title of the *topic*	String (Text)
dtopicdesc	Description of the *topic*	String (Text)
etopicpriority	Priority of the *topic*	String (Text)
atopiclabel	Label of the *topic*	List<string>

4.4.1. User interface design

The user interface contains 3 main modules as well as a bar that manages the display of each module.

In VR, it is important to properly locate the interface panels in relation to the user. There are 4 main zones according to Ref. [51]: the comfortable zone, the peripheral vision zone, the curiosity zone, and the non-recommended zone.

According to Alger [51], the non-recommended zone is a sphere that extends 0.5 m from an adult's eyes. The diagram in Fig. 15 simplifies this space on a 2D plane, but all the zones are actually three-dimensional volumes. Depending on the mode of operation, it will be necessary to locate the elements of the user interface to a comfortable distance from the user. For example, for the buttons intended for hand interactions by the user, it is recommended to locate them at a distance of the avatar's arm. For elements intended for laser pointer interactions, they can be located up to 20 m from the user.

Fig. 16 illustrates the long-term content viewing angles the left and the intersection of these angles with the recommended content sphere (0.5 m–20 m) minus the unrecommended sphere (0–0.5 m). This gives a plane at a distance of about 1.3 m from the user (the image on the right). Although the results found by Alger [51] were tested for the Oculus DK 2 head-mounted display, the theory remains relevant based on our tests, and the distance and the scale simply needed to be adjusted according to the head-mounted display used for our research, the Oculus Quest. Given that these two head-mounted displays have a similar field of view (very similar FOV; DK2 has a diagonal FOV of 135.8° vs. the Oculus Quest, which has a diagonal FOV of 131.5°).

Following the principles cited above and tests in our application, we proposed the user interface in different panels. The panels can be displayed or hidden based on the user's needs, allowing efficient use of the rendering space. The user interface is attached to the user's avatar, allowing them to be moved concomitantly. The interface panels are divided by task (Fig. 17). The main panel allows the display of the different panels: a panel dedicated to the functionalities of the 4D simulation called "4D Module", a panel dedicated to the entry of "topics" and comments and details related to an entry in the BCF file named "Create Topic Module", and a last panel dedicated to the display of "topics" and screenshots created.

A text box was added at the top of the display to show relevant messages to the user, such as messages confirming the import of the CSV file or the creation of the BCF file (Fig. 18).

The main components of the navigation module and the 4D module are:

- (1) Main navigation bar: allows to show/hide UI panels, import the CSV schedule and export a BCF file
- (2) 4D simulation control bar
- (3) Preview of the CSV schedule imported into the VR application

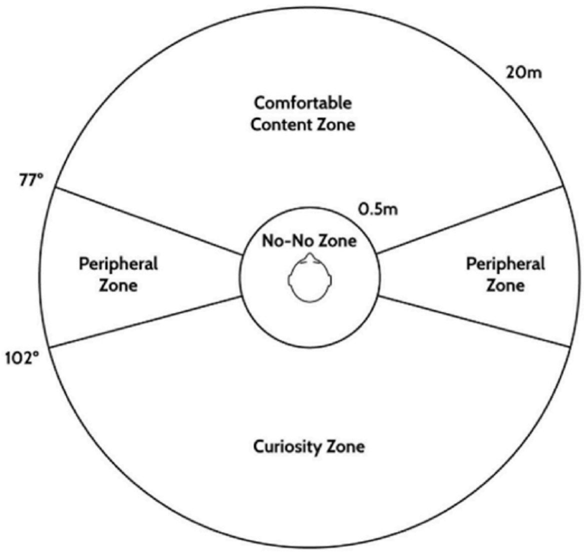
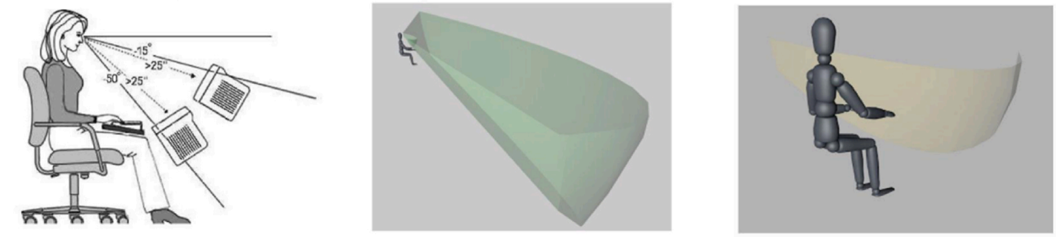


Fig. 15. Content areas in relation to the user in VR [51].



Long-term content viewing angles as illustrated by Ankrum, D.R. (1999)

Slicing these angles out of the "content zone" creates a "workspace zone"

The ideal angles and distance of 1.3 meters for working with 2D content in the DK2.

Fig. 16. Combination of recommended viewing angles with VR content area [51].

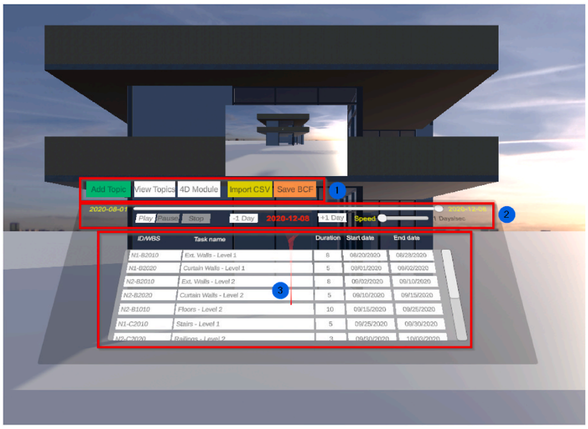


Fig. 17. Navigation module and 4D simulation module with component lists.

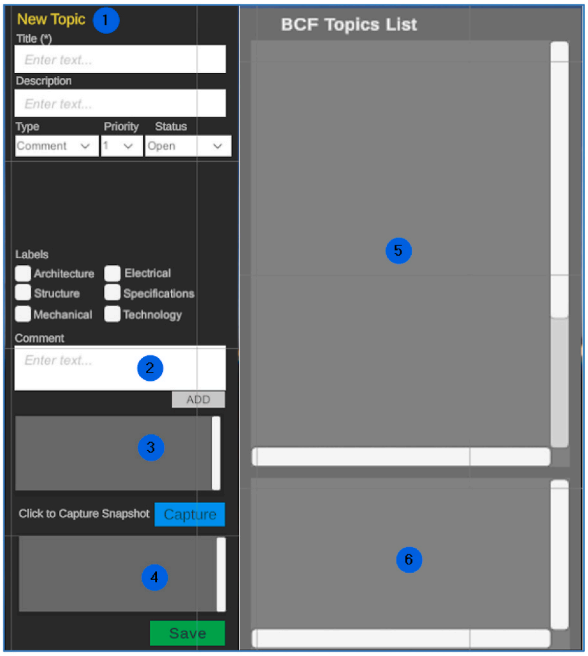


Fig. 18. BCF input module and BCF input viewer module with component lists.

As shown in Fig. 18, the main components of the BCF Input Module and BCF Input Viewer Module can be summarized as follows:

- (1) Creation of “Topic” entries and their description, labels, priority, etc.
- (2) Creation of comments of the “topic"
- (3) Viewing of created comments
- (4) Viewing of screenshots taken
- (5) Visualization of the “topics” created (see Fig. 19)
- (6) Visualization of screenshots taken when selecting a topic in field 5

4.4.2. Navigation and modes of interaction in VR

The virtual pointer is among the well-known methods for VR applications, and like a classic mouse cursor, the pointer allows the user to target different elements in the virtual world using the hand controllers [52]. The pointer has two main parts: a contact point (a small sphere) which serves as a cursor, and which collides with 3D objects or elements of the user interface (UI) and a laser beam to maintain a link between the controller and the contact point. Without the laser beam, the user easily loses the position of the contact point in a 3D environment. We compare this experience to a typical laser pointer used for wall-projected presentations (Fig. 20).

Regarding user movements, we used the default movements of the “Joysticks” of the Oculus controller. Movements consist of navigation (forward, backward, right, left), and are managed by the right joystick, while the left joystick allows to rotate the view by 45°.

The keyboard, which allows data entry in the “Topic” creation form in the BCF format is attached to the left hand controller, the user is able to write by clicking with on the different keys using the right hand controller trigger button as shown in Fig. 21.

5. Validation

As part of the validation, we first tested and evaluated the prototype against the criteria identified at the beginning of the research work. We then proceeded to have the prototype tested by five practitioners with variable experience level in construction in order to have qualitative feedback on their experience using our tool. We used a simple architectural model, representing a commercial building composed of two levels. We opted for a simple model in order to simplify the testing of our prototype version. In this sense, the construction planning schedule used is composed of 13 construction activities for the same reasons. This test can be scaled up once the proof of concept is validated.

In this section, we present the results of the two validation steps as shown in Fig. 22.

5.1. Validation against pre-established criteria

We proceeded to test our prototype in order to validate its operation according to the established criteria. First, we validated the navigation and the integration of the head-mounted display as projection support for our VR application. Afterwards, we checked the import of the BIM data of the 3D model into Unity (VR environment). Then, we proceeded to verify the 4D simulation functionalities. We then tested and validated the functionality of the data entry into the topic creation form for the collaboration in BCF format. We tested the creation of different “topics” and the information related to it such as the label, the description, priority, etc. in addition to adding comments and taking screenshots.

Finally, we tested the output BCF file to validate the integrity, file structure, and the interoperability with other BCF readers. The resulting BCF file in “. bcfzip” was extracted from the VR application. We can confirm that the structure of the markup.bcf and

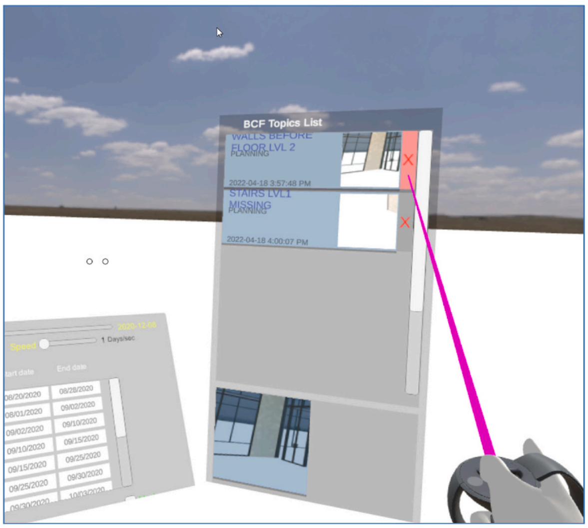


Fig. 19. Visualization panel of the “Topics” created.

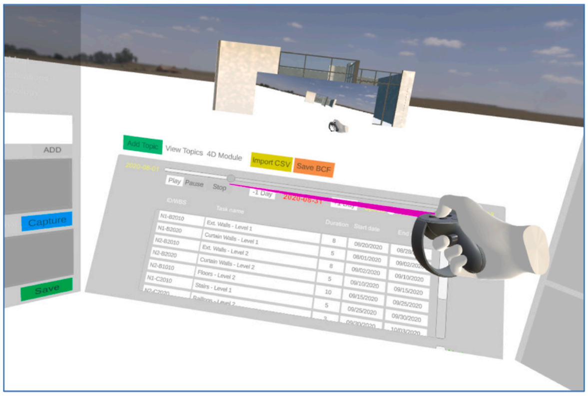


Fig. 20. Overview of joystick and laser pointer in VR app.

Fig. 21. Screenshot of the on-screen keyboard in the VR app.

viewpoint.bcfv folders and files complies with the BCF standards as defined by BuildingSMART (Figs. 23 and 24).

We opened the output file with the BCFier tool version 2.2.1.0 to confirm the interoperability of the file with other solutions. Following our tests, we confirmed that the output BCF file was readable by BCFier, and therefore the interoperability of OpenBIM was properly leveraged as mean of collaboration (Fig. 25).

This prototype can be used with other BIM models and construction planning schedules, but it will require some basic understanding of Unity, Revit and Tridify., Therfore, we recommend that BIM specialists use our prototype to prepare the project prior to sharing it to a wider audience for collaboration purposes.

5.2. *Evaluation by professionals*

In our second validation step, we approached five different industry professionals from different backgrounds to qualitatively assess our tool. All the profesionnals have limited experience with VR for construction projects. The first professional is a civil engineer with thirty years of experience in construction project management. He also holds a Master's degree in applied science, with a specialization in information technology in construction. The second and third profesionnals are Master's degree graduate and student respectively with a keen interest for BIM and the use of technology in construction, they both have experience in BIM and VR in general (limited to 3D models visualization in VR), the fourth professional is an architect and post-doctorate researcher that specialises in BIM. The fifth professional is a project coordinator and holds a Masters degree in Architectural studies with 5 years experience in the industry.

The evaluation test took place in our research lab. First, we presented the prototype, its purpose and its features. Second, the professionals used the prototype for about an hour. Subsequently, they were asked to evaluate the prototype using a questionnary, which consisted of seven questions aimed at evaluating the relevance of the tool and its usability, as well as to identify strengths and weakness in addition to any areas for improvement. The questionnary was based on a standard post-study system usability questionnaire [53] with some modifications to suite the needs of our research such as the removal of a Leikert scale to favour open answers which allows for more detailed feedback from the end-users. The questionary included the questions presented in Table 2.

Regarding the usefulness of the tool in relation to the collaboration and coordination of construction projects, most professional found the tool to be *interesting and useful for project review and coordination*. In addition, the civil engineer confirmed that although VR has seen advancements in other practices such as communication and entertainment, our tool offers a lot of potential for collaboration

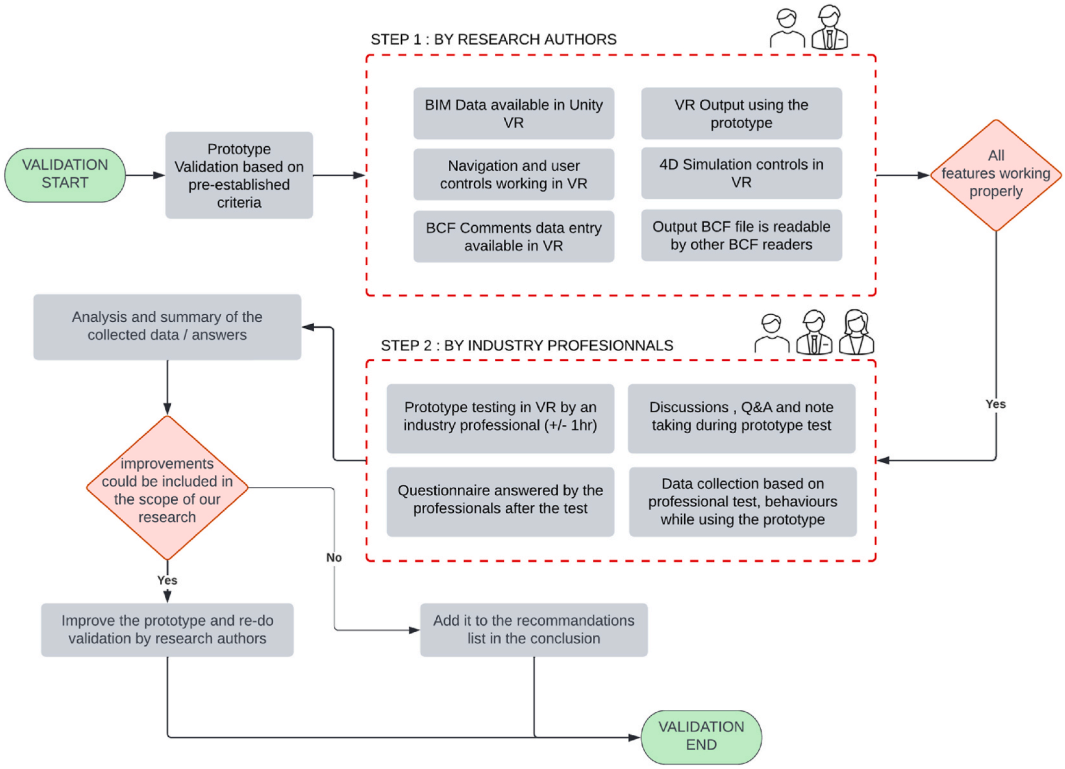


Fig. 22. Research validation process.

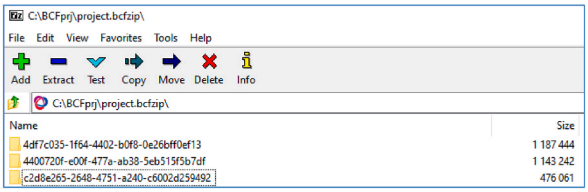


Fig. 23. Content preview of a BCFZip file containing 3 "Topics".

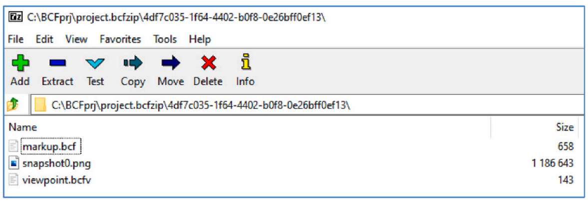


Fig. 24. Preview of the contents of a "Topic" folder in BCF format.

in a VR environment. Specifically, the professional mentioned that "… this tool can improve the state of construction projects, definitely the collaboration between planners and construction teams". The project coordinator and the post-doctorate researcher added, that the tool supports a standardized way of adding comments and documenting issues while being immersive in VR which is a new thing to their knowledge and think this can be an efficient way to add information to VR observations on the model. While the master's dergreen student and graduate disagreed with its usefulness due to the lack of seeing other users while being immersed in VR and doubted the tool's ability to support real construction meetings and coordination session. They liked however the benefits of the immersivness in VR and the BCF support none the less.

When asked about the tools usefullness for project planning, all five professionals agreed that the tool is useful when it comes to project planning and scheduling as it integrates some 4D simulation features in a VR environment. Making it easier to notice certain

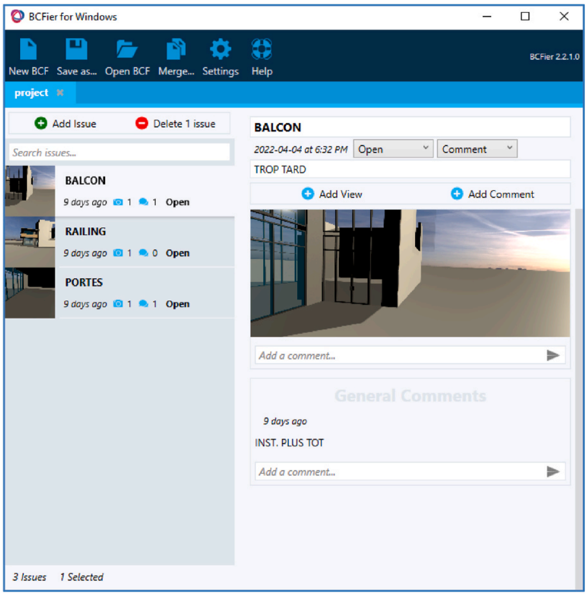


Fig. 25. Preview of outgoing BCF file opened in BCFier.

Table 2
Questionary for tool's evaluation by the professionals.

Questions	Targeted answers
- General questions to identify the profesionnals	Main profile, experience with BIM/VR.
-Do you believe that the presented tool is useful for collaboration and/or project coordination?	Evaluation of the tool's usefulness for project coordination
-Do you believe that such a tool could be useful for construction projects planning?	Evaluation of the tool's usefulness for planning (scheduling)
-Based on your experience while using the tool in the VR space, do you find it easy to use? How would you describe its ease of use?	Evaluation of the solution's ease of use in VR
-What are the strengths and weaknesses of the used tool based on your experience?	General strenghts and weaknesses
-What are the features that could be improved in the used tool?	Collecting areas of improvements regarding the tool
-What could be improved in the solution's concept or workflow or function?	Collecting areas of improvements for the concept
-What are your general appreciations of the used tool?	Collecting conclusion remarks regarding the tool's usage and limitations as well as suggestions for future improvements.

problems before hand and rectify any issues with the construction schedule. The post-doctorate researcher added that even if it agrees to the usefulness of the tool for project planning, it find its easier to do such task on a regular keyboard and mouse environment with a tradition monitor support instead of wearing a VR headset as its easier and more comfortable for prolonged periods of time and suggested that our tool could serve as an add on to the regular process to benefit for the immersivness and sense of scale.

Regarding its ease of use and user experience, All five professionals believe that the prototype is easy to use. The integration of the user interface was intuitive and easy to understand even for first use due to its simplicity, whereas the post-doctorate researcher still prefers the regular 2D user interface we find on a regular monitor. Concerning the strengths and weaknesses of the prototype, all the participants appreciate the visual aspect and the user interface. In particular, the fact that the UI is divided into different panels based on their function. And the 4D simulation controls. The master's degree student stated that "The visual side is probably the biggest point and combined with the timeline, gives an easy-to-understand animation". The civil engineer stated that a brief tutorial in the application would have helped him to navigate and know the controllers keys mapping but was able to figure it out within a minute or so of use.

In terms of functionality, we've asked the participants to states all the strenghts and weaknesses they could identify while using the app. Some of the strenghts identified by our five participants are: the ease of use. The integration of 4D animation of a BIM model in a VR environment allows for better understanding of the project construction. Immersive experience and sense of scale and presence are noticeable compared to other supports. Innovative uses of VR while integrating BIM and BCF and all participants stated that they've only seen VR for visualization purposes so far, and finally most BCF features have been integrated and the civil engineer appreciated the fact that we could take screenshot of the 3D/4D model in VR stating: "Taking a photo for the reported issue is very useful".

For the weaknesses, the participants have identified the following weakness of our tool: the lack of a multi-user system allowing for collaboration in real time, the fact that VR is HMD based creates a barrier for face to face communication with other users giving it a silo effect that could negatively impact collaboration. The user interface is less effective than the traditional 2D UI on a monitor combined with a mouse and keyboard especially the virtual keyboard for data entry which could be improved based on the civil engineer stating: “The keyboard is hard to use though, it is probably the first thing to improve. Voice recognition might be more useful for writing comments”. Similar remarks were also added by the post-doctorate researcher and the project coordinator as well.

On the other hand, in order to improve the functionality of the tool, the master’s degree graduate and the student also suggested to integrate multi-users features support multiple users simultaneously in the application, which would give more collaboration capacity to our tool. In that regard, master’s degree graduate suggested to add the support of a CAVE system for in-room collaboration while keeping visual contact between different users. Similar to that some participants, in particular the master’s degree student and post-doctorate researcher suggested to integrate the same solution in an Augmented reality support such as Microsoft HoloLens, which would improve its usability for collaboration. The post-doctorate researcher suggested to add multiples models from different disciplines with different LOD levels to have a more realistic test and even suggested to try the solution for a real project in the future. And maybe add more site logistics such as material storage zones, delivery trucks and cranes. The civil engineer as well as the master’s degree student suggested to replace the keyboard input by speech to text solutions and add 3D annotation features to draw over the model.

Table 3 summarizes the main Strengths and weaknesses of the prototypes as evaluated by the 5 participants (professionals):

6. Discussion

In this section, we propose to discuss the challenges and limitations of the different functionalities of the prototype prototype as well as our proposals for improvements to be made. The first element of discussion is related to the compliance of the produced BCF file and the presence of all standard data in the output file. All the mandatory data required in a BCF file are present, namely, the name of the topic, the labels and the comments, and the screenshots, which are optional. However, it does not include some optional components of the BCF format, such as the “BIM Snippet”, which supports a list of objects with their ifcGUID identifiers in order to identify problematic elements. Also, the position and direction of the camera at the time of the topic creation in order to return to a saved view in another collaboration or design solution. We have partially overcome the lack of this functionality by including screenshots which allow the user to communicate a view via an image instead of reproducing the position of the camera. These features of the BCF format (version 2.1) are optional according to buildingSMART [1], and do not affect the operation and integration of the BCF format, which is what we have concluded based on our testing. Regarding the use of BCF, a participant believes that for use by a larger audience, it will be necessary to add an explanatory guide to the BCF vocabulary in order to keep the data entry uniform. He believes that we should specify the kind of information the user must enter in the fields (topic title, topic description, comments, etc). We believe that these standard fields of the BCF format do not have details in order to allow flexibility for users. Because the BCF format is used by companies around the world, each company will define its own standards. However, the point raised by the professional is relevant at the project level so that the stakeholders standardize the data entry. “A reference guide or instruction for the vocabulary used would be an important point to standardize the language across the different users”, suggests the professional.

Another important aspect of the proposal is related to the preparation of the model and/or the planning schedule by the user. Indeed, it is necessary and crucial to prepare the 3D model in the design software (Revit in our case). This step ensures a link between the PBS (model elements) and the WBS (schedule tasks). This preparation consisted in introducing a WBS Code parameter containing a code in text format. The issue we encountered with this method was that all the elements of the model must be coded in the right way to match a similar code in the planning schedule, otherwise, these elements will not be displayed and will always be invisible in the virtual environment, because they do not correspond to any of the tasks imported from the schedule and therefore will not be visible on any date among the different dates of the project.

The standard commands required for a 4D simulation are present in our prototype. These includes the user reading with a variable speed ranging from 1 day/sec. to 7 days/sec, pausing on a specific date, moving in time by 1 day (forward or backward) and stopping the simulation and returning to the start date. The slider indicating the current date can be moved with an increment of 1 day. These commands are not active until the user loads the schedule. According to our tests, the tools work as expected and allow to conduct a 4D simulation in VR. The schedule can be loaded and visualized in table format in the VR environment. However, this is used for visualization purposes only, and a task start or end date cannot be modified inside the VR application, which represents a limitation if the

Table 3
Prototype’s strengths and weaknesses results summary.

Strengths	Weaknesses
Could improve the collaboration between construction planners and other teams (P1)	Less useful tool due to the lack of contact with other professionals while in VR (P3), (P4)
Support an OpenBIM standard for adding comments in VR which is a novelty at the time of the evaluation. (P2), (P3)	Less comfortable to wear VR Headset for prolonged periods (P2)
Using BCF comments is an efficient way to add information observed while in VR. (P2), (P3), (P4)	Regular On-Screen user experience is easier for new users (P2)
Usefull for project planning due to the addition of 4D in VR and ability to do scenario simulations (P1), (P2), (P3), (P4), (P5).	Lacks a tutorial to show the controller key mappings and navigation controls. (P1)
Simple and easy to use user interface and 4D simulation controls in particular, even for first time use. (P1), (P2), (P3), (P4), (P5).	Virtual Keyboard is less effective than traditional keyboard (P1), (P2), (P5).
Immersivness, sense of scale. (P1), (P2), (P3), (P4), (P5).	Lack of 3D annotations, drawing tools to annotate the 3D model. (P1), (P3).
Possibility to take screenshots of the issues found. (P1)	

Participants legend: (P1) Civil Engineer, (P2) Post-Doctorate researcher, (P3) Master’s degree student, (P4) Master’s degree graduate, (P5) Project coordinator.

goal is to try different planning scenarios directly in the virtual environment. Currently, the user has to quit the application, make changes to the schedule CSV file, and then reload it in the application.In future research we suggest the implementation of modifiable fields (task name, start and end dates, duration etc) and link these fields to an XML file in read/write modes. This way, the ability to test different scenarios would be possible. Furthermore, a standard XML structure that includes different planning data to support 4D simulations would be beneficial. This open format, similar to BCF could be supported by different Scheduling software solutions in import and export such as MS Project, Asta Powerproject, Primavera P6 and so on.

As for the immersive VR experience, while navigation in the VR environment is functional, some improvements can be added. We have implemented free mode movement with joysticks, but the movement mode does not support collisions with objects, which makes it impossible to take a staircase, for example. But the advantage is that one can freely pass through walls and doors if desired. This mode is useful for taking screenshots across multiple locations in the building during the 4D simulation. To support the constructability analysis it is best to have free access to the whole model. However when the simulation is finished and the building is in the completed state, it is preferable to have the collisions turned on in order to be able to do a walkthrough of the model in a more pleasant way. While we found the experience to be immersive, improvements can be made to the quality of the 3D rendering, the lights and the textures in order to make the project more realistic.

We found data entry to be intuitive in the sense that it is a learned method of using a computer, as the input format is a form and a keyboard. The user cannot use a physical keyboard, but rather, only a virtual keyboard, and we noticed that the use of our virtual keyboard with the help of a virtual laser pointer is difficult and slow. Sometimes, a click may not register on the keyboard, resulting in corrections to the user-entered text. This method remains useable, but is less fluid and efficient than using a physical keyboard.

7. Conclusion

The effective use of VR environments to support collaboration based on 4D construction simulation still faces significant challenges. One of the major challenges involves the bidirectional transfer of information between BIM authoring tools and VR environments. While considerable progress has been made in enhancing the use of IFC as an exchange format, efficient use of BCF is crucial for exchanging information related to identified issues from VR environments to BIM design software. Yet, there has been relatively little scientific research dedicated to this topic in the literature. The challenges identified from the literature are mainly related to inefficiencies in generating 4D simulations, hurdles in transferring schedule data and BIM information to a VR environment, and effectively communicating identified problems during collaboration in a VR environment to another platform through an OpenBIM exchange format like BCF.

In this context, our project proposes a context aimed at supporting collaboration based on the BIM model and 4D simulation in a VR environment. We developed a module allowing the entry of collaboration data such as comments and screenshots based on the BCF standard. This allows to extract data from a virtual environment to other BIM solutions with great support from the OpenBIM (BCF format). The current version of the work relies on different commercial solutions, such as Tridify. In this sense, it seems of less interest in terms of openBIM than a solution such as open-source solutions such as IfcOpenShell. However, for the purposes of our study, these solutions have the advantage of providing comprehensive and easy-to-use IFC-based transfer functionalities, enabling a proof-of-concept to be achieved within the project timeframe.

In future work, we will rely on open-source solutions such as IfcOpenShell, instead of commercial solutions.

CRediT authorship contribution statement

Aissa Khorchi: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Conrad Boton:** Writing – review & editing, Writing – original draft, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] R. Klinc, Ž. Turk, Construction 4.0 – digital transformation of one of the oldest industries, Econ. Bus. Rev. 21 (2019), https://doi.org/10.15458/ebr.92.
[2] C. Boton, L. Rivest, O. Ghnaya, M. Chouchen, What Is at the Root of Construction 4.0: A Systematic Review of the Recent Research Effort, Archives of Computational Methods in Engineering, 2020, https://doi.org/10.1007/s11831-020-09457-7.
[3] A. Nandavar, F. Petzold, G. Schubert, E. Youssef, Opening BIM in a new dimension. A simple, OpenBIM standards based Virtual reality collaboration technique for BIM, in: Intelligent & Informed, Proceedings of the 24th International Conference of the Association for Computer-Aided Architectural Design Research in Asia, CAADRIA), 2019, pp. 595–604.
[4] C. Boton, D. Forgues, G. Halin, A framework for Building Information Modeling implementation in engineering education, Can. J. Civ. Eng. 45 (2018), https://doi.org/10.1139/cjce-2018-0047.
[5] C. Boton, Supporting constructability analysis meetings with Immersive Virtual Reality-based collaborative BIM 4D simulation, Autom. ConStruct. 96 (2018) 1–15, https://doi.org/10.1016/j.autcon.2018.08.020.

[6] T. Fernando, K.-C. Wu, M. Bassanino, Designing a novel virtual collaborative environment to support collaboration in design review meetings, J. Inf. Technol. Construct. 18 (2013) 372–396.
[7] Y. Mezrag, C. Boton, Integrating 4D simulations and virtual reality environments : an innovative prototype, in: International Conference on Cooperative Design, Visualization and Engineering, Springer International Publishing, 2021, pp. 103–114, https://doi.org/10.1007/978-3-030-88207-5.
[8] K. Ku, P.S. Mahabaleshwarkar, Building interactive modeling for construction education in virtual worlds, Electron. J. Inf. Technol. Construct. 16 (2011) 189–208.
[9] T.H. Wu, F. Wu, C.J. Liang, Y.F. Li, C.M. Tseng, S.C. Kang, A Virtual Reality Tool for Training in Global Engineering Collaboration, Univers Access Inf Soc, 2017, pp. 1–13, https://doi.org/10.1007/s10209-017-0594-0.
[10] A.Z. Sampaio, 4D/BIM model linked to VR technology, in: Dep. Civil Engineering, University of Lisbon, Av. Rovisco Pais, Lisbon, 2017, https://doi.org/10.1145/3110292.3110298, 1049-001, Portugal.
[11] T.V. Cruz, Virtual Reality in the Architecture, Engineering and Construction Industry. Proposal of an Interactive Collaboration Application, Universidado do Porto, 2018. https://repositorio-aberto.up.pt/bitstream/10216/111295/2/258986.pdf. (Accessed 26 June 2018).
[12] C.S. Dossick, Messy work in virtual worlds: exploring discovery and synthesis in virtual teams, Lect. Notes Comput. Sci. 8683 (2014) 134–142. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84921741215&partnerID=40&md5=c270671d2311837b57c04c8be5480609.
[13] R. Zaker, E. Coloma, Virtual reality-integrated workflow in BIM-enabled projects collaboration and design review: a case study, Visual. Eng. 6 (2018), https://doi.org/10.1186/s40327-018-0065-6.
[14] buildingSMART International, BIM Collaboration Format v2.1 Technical Documentation, 2019. https://github.com/buildingSMART/BCF-XML/blob/release_2_1/Documentation/README.md. (Accessed 4 October 2022).
[15] L. van Berlo, T. Krijnen, Using the BIM collaboration format in a server based workflow, Procedia. Environ. Sci. 22 (2014) 325–332, https://doi.org/10.1016/j.proenv.2014.11.031.
[16] N.M. Bouchlaghem, I.G. Liyanage, Virtual Reality Applications in the UK's Construction Industry, Cib Report, vols. 89–94, 1996. http://itc.scix.net/data/works/att/w78-1996-89.content.pdf. (Accessed 26 June 2018).
[17] Y. Zhang, H. Liu, S.C. Kang, M. Al-Hussein, Virtual reality applications for the built environment: research trends and opportunities, Autom. ConStruct. 118 (2020), https://doi.org/10.1016/j.autcon.2020.103311.
[18] J. Kieferle, U. Woessner, H. RheinMain, BIM Interactive-About Combining BIM and Virtual Reality A Bidirectional Interaction Method for BIM Models in Different Environments, 2015.
[19] W.R. Sherman, A.B. Craig, Understanding Virtual Reality: Interface, Application, and Design, second ed., Morgan Kaufmann Publishers, San Francisco, CA, 2018. ISBN: 978-0128009659.
[20] P. Fuchs, Les casques de réalité virtuelle et de jeux vidéo, Presses des Mines-Transvalor, 2016.
[21] P. Bouvier, Presence in Virtual Reality, A User Centred Approach, Ph.D. Thesis, Université Paris-Est, 2009. https://tel.archives-ouvertes.fr/tel-00581550/.
[22] H. Park, D.S. Panya, H. Goo, T. Kim, J. Seo, BIM-Based virtual reality and human behavior simulation for safety design, Proc. Int. Conf. Edu. Res. Computer Aided Architect. Design Europe 2 (2018) 823–832, https://doi.org/10.52842/conf.ecaade.2018.2.823.
[23] V. Getuli, P. Capone, A. Bruttini, S. Isaac, BIM-based immersive Virtual Reality for construction workspace planning : a safety-oriented approach, Autom. ConStruct. 114 (2020) 103160, https://doi.org/10.1016/j.autcon.2020.103160.
[24] S.Y. Chen, W.C. Chien, Immersive virtual reality serious games with DL-assisted learning in high-rise fire evacuation on fire safety training and research, Front. Psychol. 13 (2022), https://doi.org/10.3389/fpsyg.2022.786314.
[25] H. Chen, L. Hou, G. Kevin, Zhang, S. Moon, Development of BIM, IoT and AR/VR technologies for fire safety and upskilling, Autom. Construct. 125 (2021), https://doi.org/10.1016/j.autcon.2021.103631.
[26] M. Afzal, M.T. Shafiq, H. Al Jassmi, Improving construction safety with virtual-design construction technologies - a review, J. Inf. Technol. Construct. 26 (2021) 319–340, https://doi.org/10.36680/j.itcon.2021.018.
[27] P. Wang, P. Wu, J. Wang, H.L. Chi, X. Wang, A critical review of the use of virtual reality in construction engineering education and training, Int. J. Environ. Res. Publ. Health 15 (2018), https://doi.org/10.3390/ijerph15061204.
[28] A. Kamari, A. Paari, H.Ø. Torvund, BIM-enabled virtual reality (VR) for sustainability life cycle and cost assessment, Sustainability 13 (2021) 1–24, https://doi.org/10.3390/su13010249.
[29] B. Schiavi, V. Havard, K. Beddiar, D. Baudry, BIM data flow architecture with AR/VR technologies: use cases in architecture, engineering and construction, Autom. ConStruct. 134 (2022), https://doi.org/10.1016/j.autcon.2021.104054.
[30] S. Alizadehsalehi, A. Hadavi, J.C. Huang, From BIM to extended reality in AEC industry, Autom. ConStruct. 116 (2020) 103254, https://doi.org/10.1016/j.autcon.2020.103254.
[31] R. Petrie, What is openBIM?, BuildingSMART Int. (n.d.). https://www.buildingsmart.org/about/openbim/openbim-definition (accessed October 4, 2022)..
[32] M. Baldwin, The BIM-Manager: A Practical Guide for BIM Project Management, Beuth Verlag GmbH, 2019.
[33] K.W. May, J. Walsh, R.T. Smith, N. Gu, B.H. Thomas, H.K.W. Kong, A.L. May, Unityrev-bridging the gap between BIM authoring platforms and game engines by creating a real-time Bi-directional exchange of BIM data, in: POST-CARBON, Proceedings of the 27th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), CAADRIA, 2022, pp. 527–536. https://github.com/kieran196/UnityRev-ProjectExchange.
[34] E. Makisha, RuleML-based mechanism of building information models verification, E3S Web of Conf. 132 (2019) 1–6, https://doi.org/10.1051/e3sconf/201913201014.
[35] A.Z. Sampaio, Enhancing BIM Methodology with VR Technology, State of the Art Virtual Reality and Augmented Reality Knowhow, 2018, pp. 59–79.
[36] A. Nandavar, F. Petzold, J. Nassif, G. Schubert, B. Ag, Interactive Virtual Reality Tool for BIM Based on IFC. Development of OpenBIM and Game Engine Based Layout Planning Tool-A Novel Concept to Integrate BIM and VR with Bi-directional Data Exchange, 2018.
[37] J.S. Goulding, F.P. Rahimian, X. Wang, Virtual reality-based cloud BIM platform for integrated AEC projects, J. Inf. Technol. Construct. 19 (2014) 308–325. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84927510857&partnerID=40&md5=620670f300ac7530513e48af57982823.
[38] L.-S. Kang, H.-S. Moon, S.-Y. Park, H.-S. Kim, C.-H. Kim, Application of VR technology linked with 4D CAD system for construction project, in: NISS '09: Proceedings of the 2009 International Conference on New Trends in Information and Service Science, IEEE Computer Society, Washington, DC, USA, 2009, pp. 1058–1062. http://proxy.bnl.lu:2080/10.1109/NISS.2009.139.
[39] N. Dawood, E. Sriprasert, Z. Mallasi, B. Hobbs, Development of an integrated information resource base for 4D/VR construction processes simulation, Autom. ConStruct. 12 (2002) 123–131.
[40] M.V. Tallgren, M. Roupé, M. Johansson, 4D modelling using virtual collaborative planning and scheduling, J. Inf. Technol. Construct. 26 (2021) 763–782, https://doi.org/10.36680/J.ITCON.2021.042.
[41] A. Rashidi, W.Y. Yong, D. Maxwell, Y. Fang, Construction planning through 4D BIM-based virtual reality for light steel framing building projects, Smart Sustain. Built Environ. (2022), https://doi.org/10.1108/SASBE-06-2022-0127.
[42] J.I. Messner, S.C.M. Yerrapathuruni, A.J. Baratta, V.E. Whisker, Using virtual reality to improve construction engineering education, Am. Soc. Eng. Edu. Annual Conf. Expos. (2003) 1–8. http://www.engr.psu.edu/ae/cic/publications/conference/messner_2003_using_virtual_reality.pdf%0A. https://www.engr.psu.edu/ae/cic/publications/Conference/Messner_2003_Using_Virtual_Reality.pdf%0A. http://www.engr.psu.edu/ae/cic/publications/conference/messner.
[43] F. Dreischerf, H. Astour, Bidirectional linking of 4D-BIM planning with virtual and augmented reality, in: Conference: ICCBEI 2019, 2019.
[44] S.T. March, G.F. Smith, Design and natural science research on information technology, Decis. Support Syst. 15 (1995) 251–266, https://doi.org/10.1016/0167-9236(94)00041-2.
[45] A. Hevner, S. Chatterjee, Design science in information systems research, in: Integrated Series in Information Systems, Springer, 2004, pp. 75–105. http://dl.acm.org/citation.cfm?id=2017217.
[46] K. Peffers, T. Tuunanen, M. Rothenberger, S. Chatterjee, A design science research methodology for information systems research, J. Manag. Inf. Syst. 24 (2008) 45–77, https://doi.org/10.2753/MIS0742-1222240302.

[47] Tredinnick Ross, Brady Boettcher, Simon Smith, Solovy Sam, Ponto Kevin, Uni-CAVE: a Unity3D plugin for non-head mounted VR display systems, in: 2017 IEEE Virtual Reality, VR), 2017, pp. 393–394.
[48] M. McGill, S. Brewster, D.P. De Sa Medeiros, S. Bovet, M. Gutierrez, A. Kehoe, Creating and augmenting keyboards for extended reality with the keyboard augmentation toolkit, ACM Trans. Comput. Hum. Interact. 29 (2022), https://doi.org/10.1145/3490495.
[49] F. Lourenzi, A. Geiger, BCFViewer – BIM Collaboration Format Tool Development, 2022.
[50] J.G. Lee, H.-S. Lee, ; Moonseo Park, W. Kim, An Interoperable Coordination Method for Sharing Communication Information Using BCF (BIM Collaboration Format), n.d..
[51] M. Alger, Visual Design Methods for Virtual Reality, 2015.
[52] J. Jung, H. Park, D. Hwang, M. Son, D. Beck, J. Park, W. Park, A review on interaction techniques in virtual environments, in: Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management, 2014, pp. 1582–1590.
[53] J.R. Lewis, Psychometric evaluation of the post-study system usability questionnaire: the PSSUQ, in: Proceedings of the Human, Factor Society, 1992, pp. 1259–1263.