Sep. 2011, Volume 5, No. 9 (Serial No. 46), pp. 787-798
Journal of Civil Engineering and Architecture, ISSN 1934-7359, USA

Linking a Game Engine Environment to Architectural Information on the Semantic Web

Pieter Pauwels[1], Ronald De Meyer[1] and Jan Van Campenhout[2]

1. Department of Architecture and Urban Planning, Faculty of Engineering and Architecture, Ghent University, Ghent 9000, Belgium

2. Department of Electronics and Information Systems, Faculty of Engineering and Architecture, Ghent University, Ghent 9000, Belgium

Abstract: Because of the importance of graphics and information within the domain of architecture, engineering and construction (AEC), an appropriate combination of visualization technology and information management technology is of utter importance in the development of appropriately supporting design and construction applications. Virtual environments, however, tend not to make this information available. The sparse number of applications that present additional information furthermore tend to limit their scope to pure construction information and do not incorporate information from loosely related knowledge domains, such as cultural heritage or architectural history information. We therefore started an investigation of two of the newest developments in these domains, namely game engine technology and semantic web technology. This paper documents part of this research, containing a review and comparison of the most prominent game engines and documenting our architectural semantic web. A short test-case illustrates how both can be combined to enhance information visualization for architectural design and construction.

Key words: 3D, BIM, construction industry, game engines, information, semantic web, virtual environments, visualization.

1. Introduction

1.1 The Extent of Interoperable Three-Dimensional AEC Information

One of the most notable efforts of recent years in the context of semantic information modeling for construction industry, is the Building Information Modeling (BIM) approach [1]. This approach is often combined with the Industry Foundation Classes (IFC) as a language to establish an appropriate level of interoperability [2]. In a BIM application, one is able to build up a 3D building model that represents the building design at hand and describes all information considered relevant to this building design, including to a certain extent material information, cost information, planning information, etc. Using the neutral IFC format to serialize this information, one should ideally be able to communicate this information to other applications that may reuse and/or interpret this information.

A parallel research and development effort towards interoperability of information is led by the World Wide Web Consortium (W3C) in the World Wide Web (WWW) domain, albeit targeting a less specialized and wider field of information. By increasingly changing the WWW into a semantic web [3], this effort aims at describing and connecting all WWW information so that its semantics are made explicit and the described information becomes reusable both by humans and machines. The usage of semantic rule engines and query environments accessible by semantic web agents, for instance, may then target an improved automatic processing of information.

Whereas the IFC standard provides a means to describe information for one domain only, namely the AEC domain, the semantic web effort targets interoperability in a wider range of domains, namely all domains that may be described in the WWW. As the

Corresponding author: Pieter Pauwels, Ir., research fields: information modeling, architecture, virtual environments, game engines, design cognition. E-mail: pipauwel.pauwels@ugent.be.

semantic web effort thus provides a more general and more widely supported and applied approach, we investigated how the building information described with the IFC standard could be brought into this semantic web and how it could be extended with other information in this semantic web, including more detailed material information, cultural heritage information, library information, etc. A large part of this research work is presented in [4], and will only be documented briefly in this paper. The result of this research is that building information can be converted or directly modeled into a semantic web representation, allowing anyone to extend it with information that falls out of scope for the IFC standard, to (re-)edit this information and to reuse it efficiently in domains and environments that were not available in the narrower context of BIM and IFC.

1.2 Offering Semantic Information through a 3D Virtual World

A second part of our research focuses on the visualization of this semantic information in a 3D virtual world. The semantic web version of the building information allows the addition of information that is not part of the IFC standard, nor of the BIM environment (e.g., geography, cultural heritage, material details, etc.). This information is hard to visualize in traditional AEC applications as it is not part of the native data structure and functionality of these applications.

We therefore started an investigation in advanced visualization environments, including Virtual Reality (VR) and Mixed Reality (MR) systems, specifically focusing on their ability to visualize the semantic web of information behind the building model. We compare the general visualization possibilities of VR and MR visualization systems in this respect. When considering the AEC domain, MR systems seem to focus on inspection and construction progress management because of their incorporation of real world objects, whereas VR systems rather focus on the earlier design phases, in which the building design is still mainly virtual. We also experience a contrast between dedicated VR environments built up using specialized VR software (e.g., VR Toolkit, etc.) and hardware components (e.g., data gloves, CAVE, etc.), as opposed to fast VR environments built up using an out-of-the-box game engine that provides a whole range of standard components with diverse complex functionalities.

Following the outcome of this general comparison, we consider the usage of game engines as they could provide us with a fast and intuitive way to generate a VR world connected to the available semantic web information. We briefly test and review several game engines, thereby taking into account parameters such as intuitiveness, API quality, graphical visualization quality, relation with AEC applications, and interactivity. Results of this study show advantages and disadvantages throughout the engines for each of these parameters. One game engine is eventually considered as the more appropriate choice for the visualization of a building model together with its underlying semantic web of information.

In a short test-case, we finally investigate how a three-dimensional building model visualized in a game engine may be connected to semantic web servers for an advanced and interactive 3D visualization of this semantic information. For this test-case, we have built an architectural semantic web for an architectural design in Antwerp, Belgium. Through a semantic query interface, queries can be constructed and run on this architectural semantic web, resulting in the specifically requested information. We connect the game engine with the query interface, so that the architectural semantic web may be queried at runtime from within the game engine environment.

This paper gives a brief overview of this end-to-end investigation. We start the paper with a brief discussion on semantics in the AEC domain, a discussion concerning visualization strategies for AEC information and a comparison of underlying game engines. We then indicate how a good combination of an information management and visualization environment could produce an effective and realistic visualization with the appropriate respect for the underlying information.

2. AEC information in the Semantic Web

Management of information in the AEC domain seems to evolve into an appropriate usage of BIM applications. Within the context of construction design, these BIM applications prove to be a very valuable tool. However, as soon as building related 3D information is to be accessed from within any other knowledge domain, such as architectural design or cultural history, the range of powerful BIM environments does not suffice. Our research therefore concentrated on how the amount and the kind of information available in traditional BIM models may be extended with information from within these other knowledge domains.

2.1 Information Management Relying on BIM Environments

The AEC domain involves all kinds of information, covering material characteristics, elementary architectural design notions, legal regulations, three-dimensional parameters, etc. Within a BIM platform, a designer is able to model a three-dimensional BIM model containing the most essential parts of this information about the building designed, including concrete information as 3D geometry, cost information, material information, etc. [1]. This improves the interoperability of information as AEC specialists presumably need to maintain only one central information model, which can then be referenced by diverse other applications for automation in construction.

This interoperability is further enhanced by the IFC format [2]. The standardized IFC schema aims at providing AEC specialists and application developers with an agreed upon information structure, fit for describing any building or architectural 3D structure. Although important limitations persist in the development of IFC [5], it provides a solid basis to rely upon within construction industry applications. Example applications relying on IFC information can be found in various corners of the AEC domain, including for instance applications for energy performance calculations or project cost analyses. As these can become rather specialized applications, these calculation and simulation applications most often require remodeling information or adding extra information next to the information coming from the BIM environment. Certain knowledge domains, covering not only energy performance simulation or cost analysis, but also cultural heritage or architectural history domains, may benefit from additional information linked to the IFC information [4].

2.2 Connecting Building Information to Information in Relevant other Knowledge Domains

As we indicated in [6], relying on the IFC schema has its advantages, but also its prices. Whereas it provides a certain level of interoperability, which is in itself an issue that still needs substantial research work [5], it restrains users to a certain level of expressiveness. One cannot focus on designing the most unique building elements, including sloped walls, three-dimensionally curved roofs with window elements, etc., and still expect this atypical information to be describable in an international standard. As the semantic web allows anyone to construct his or her own vocabulary and link it to existing information, this may encompass this situation and allow designers to step out of the IFC schema when necessary and design the information in a vocabulary of choice, while to a certain extent maintaining the link with the standardized tools.

Semantic web technology enables the description of any information using mere logical terms. The discussion of semantic web technology is out of scope for this paper, but references and a brief discussion in the light of the AEC domain can be found in [6]. Using semantic web technology, one may describe near to anything, including for instance the spatial topology of a building, building element information, historical design information, geographical information, material details, etc. These descriptions can be made using custom vocabularies or ontologies, which might for instance be considered the “vocabulary” of an architectural design firm or of a certain period in cultural history.

Semantic web technology additionally promises, or at least targets, the possibility to connect all these vocabularies and different kinds of information into one semantic web. By providing links to the underlying vocabularies and to other related information, semantic web descriptions thus always include their inherent semantics. As such, it might enable the description of information in several distinct, but nonetheless connected graphs, for instance explicitly connecting a building representation in the IFC schema to representations of the same building according to any geometric schema (e.g., X3D, STL, etc.), or to completely different information not available in the IFC schema. Applications may rely on this web of information as a central information source to provide services as needed for each of the members of a design team and to people outside the AEC domain but nonetheless interested in building information.

3. Architectural Visualization in Virtual 3D Environments

The information in a semantic web is not that easy accessible by any human user. Because of the importance of an intuitive interface in handling any kind of information, we start an investigation on how to present this information in an appropriate visualization environment capable of relating it to the respective 3D objects. We investigate the diverse environments and their inherent strategies to visualize information related to the 3D objects. Several environments already exist for the visualization of construction-related information. These environments differ mainly in the following characteristics: the quality of graphics, the level of interactivity, the information flow and reuse from within modeling environments for the AEC domain and/or the semantic web, the availability and quality of scripting functionality for the implementation of extended functionality, the intuitiveness of the resulting 3D environment, and the targeted benefits for the AEC domain.

We distinguish between VR and MR environments within this investigation. We consider VR environments as environments that consist completely of virtual objects and of no real objects, whereas MR environments are understood as environments that combine both virtual and real objects in a certain degree.

A first class of VR applications related to the AEC domain provides users with an online 3D environment in which users can lead a “second”, virtual life through their personalized avatars. Two main examples of such environments include Second Life [7] and SketchWorlds [8]. These environments typically focus more on fast geometric modeling of an attractive 3D environment and less on the information modeling typically found in AEC applications. The standard functionalities, including for instance easy collaboration possibilities over a network, intuitive 3D modeling and basic visualization functionality, provide nonetheless for a solid basis for collaborative design exploration. The main limitation in these environments seems the missing import/export functionality with respect to AEC applications. SketchWorlds provides the possibility for importing Ogre meshes, which may be exported from AEC applications using third party plug-ins, but this does not compare to the typical information exchange found in the AEC industry.

A second class of VR applications seems to focus on providing an appropriately functional virtual world for building design inspection. Examples of such applications include NavisWorks [9], Virtual Building Explorer (VBE) [10] and goBIM [11]. These applications typically focus on an optimized information exchange with major BIM environments, as opposed to the online worlds brought up above. These applications provide easy support for calculations and simulations mainly considered in a final design stage, including 4D construction planning, interference or clash detection, design reviews, 3D model checking, etc. The applications tend however not to provide an equally intuitive virtual world as is experienced in Second Life or SketchWorlds for instance. Instead, focus is put on an interactive third person overview and on the possibility to create animations with a certain lesser degree of freedom. The ICEvision application [12], for instance, focuses on the creation of such more presentation-oriented animations. These applications present a third person view that is not as intuitive and interactive as the first person view experienced in Second Life for instance.

MR applications always rely on a certain percentage of real objects. Two common examples of MR applications are the Layar Augmented Reality Browser [13] and the Wikitude World Browser [14]. Both applications are typically used on a smartphone and present information retrieved from the web as a dedicated virtual layer on top of a view recorded by the built-in camera recorder of the smartphone, based on GPS information and on the view direction of the camera. In the AEC domain, MR applications seem typically useful in the later stages of the design process, including for instance the stages in which part of the site is already under construction. A large part of the suggested applications include construction site inspection [15]. Another important application domain is project management, more specifically for construction progress monitoring [16]. Even more extremely, MR visualization can prove a useful addition in the context of facility management (FM), building maintenance and renovation [17], and building damage evaluation in the aftermath of disasters [18].

4. Suggested Strategy for the Visualization of Semantic Architectural Information

4.1 Strategy

MR applications are thus typically developed for contexts in which real objects matter for the building design. In an early design context, this includes in many cases only an empty construction site. In these cases, this MR visualization does not seem such a valuable addition to the design process, because design decisions rely almost exclusively on the virtual parts of the design. Certain, more exceptional design tasks may profit from the usage of MR visualization, including the design of underground structures [19] and exterior architectural design [20] for instance, but in mainstream design projects the percentage and value of real objects, e.g., the site, tends to be rather low in comparison with what is imagined and is essentially virtual. Because VR visualizations, on the other hand, typically involve only virtual elements, they tend to be far more useful in the early stages of the design process, in which the complete design is still mainly virtual in the designer’s mind. VR visualization environments thus typically focus on visualizing as much of this internal mental image of the design to generate an improved image of the state of the design at hand.

In our research, we target the improvement of the initial stages of the design process. Following the above considerations, we focus our research on VR visualization systems and will handle only these systems in the remainder of this paper. VR visualization systems can typically be subdivided in two types, depending on the software deployed to build the virtual environments. On the one hand, we distinguish more traditional VR systems that rely on more low-level software libraries for optimizing the interface between the application logic and the VR hardware, including tracker systems, stereoscopic displays, interactive input devices, etc. Alternatively, we distinguish VR environments that rely on standard game engines that focus on an out-of-the-box environment for a fast and intuitive development of virtual worlds that may subsequently be optimized for certain VR hardware.

A brief and comprehensive comparison between game engines and "virtual reality engines" is given in [21]. This overview indicates the most significant reasons why one typically chooses for a visualization based on a game engine. Comparing to other research efforts [22-24], we extract the following main reasons.

- Solid, high-level, and out-of-the-box functionality enables a considerably fast development of functional virtual worlds.
- Low cost.
- Mature tools for the development of extra functionality: networking utilities, physics engine, AI engine, etc..
- More compelling results in terms of interactivity and 3D graphics.
- Designed for a remarkably good minimal performance on a whole range of operating systems and hardware configurations.

Comparing this strategy overview with our focus on the initial architectural design stages, we decided to focus first on VR visualization environments based on game engine technology. Considering the large number of available game engines, it is nearly impossible to give even a brief overview of all game engines. We therefore focus on the most promising game engines in relation to visualizations in the AEC domain. An overview and qualitative analysis of such engines was recently given in [25]. Following developments in game engine technology, we give a short overview of the game engines Unity, Quest3D, Shiva, Virtools and Creator. We analyze to what extent they are appropriate for the visualization of semantic architectural information in a virtual interactive 3D environment, and in how far they compare or compete with the visualization environments discussed earlier. Traditionally very popular game engines, such as Doom, Unreal, etc., are not considered as they do not provide the required interoperability with the CAD tools typically deployed in the AEC industry, and usually require users to (re-)model all 3D in an in-house game editor to obtain sufficiently qualitative graphics.

4.2 Comparison of Game Engines

4.2.1 Creator

Esperient Creator is one of the newly emerged 3D engines explicitly targeting architectural design processes. The product's whitepaper indicates how the product fits perfectly into the typical architectural design process, indicating the position of the application in relation with major architectural design tools such as Google SketchUp, Autodesk Revit, GraphiSoft ArchiCAD, Bentley v8i and Autodesk 3DS Max [26]. A closer look at the workflow from CAD tools in the AEC domain to Esperient Creator, however, shows a not so intuitive process, involving a link to the relational database underlying the original CAD tool or connecting trough Microsoft's Distributed Component Object Model (DCOM) [26]. Alternative file-based processes seem to require a detour via 3DS Max, which is a specialized 3D visualization tool and not a standard CAD tool possessed by any AEC specialist, and the usage of the Right Hemisphere (.rh) file format, which seems not to produce the best visual graphics when testing.

On the other hand, Esperient Creator provides a significant and useful set of standard GUI components to its users, and looks a fairly intuitive tool to use [27]. Apart from the two available built-in scripting languages, it also provides an extensive C++ API for the development of advanced functionality in the visualization. Extended with the availability of ODBC database connectivity and the necessary networking functionality, Esperient Creator provides a solid basis for the advanced information visualization environment we targeted earlier.

4.2.2 Virtools

The 3DVIA Virtools visualization engine of Dassault Systèmes has evolved into a complex, but highly functional platform for 3D authoring and 3D visualization. The presented process “Import - Create – Publish” [28] gives an appropriate idea of the visualization process typically gone through when using Virtools for building a virtual world. Extensive functionality is provided for the creation of the eventually resulting world, extending the main platform with many additional functionality libraries, including a Physics Library, a Multiuser Library, a VR Library, etc. Combined with a highly functional and well-documented Software Development Kit (SDK), any user is thus provided with all the required functionality to build compelling 3D worlds.

Virtools primarily focuses on the connection with Dynamic Content Creation (DCC) applications, thereby mainly targeting with animation software such as Autodesk 3DS Max. The bridge towards Product Lifecycle Management (PLM) and BIM software is provided through the 3D XML plug-in for Virtools [28]. This plug-in is, however, mainly available for PLM applications by Dassault Systèmes, thereby somewhat excluding the easy usage of Virtools outside the Dassault Systèmes product suite. Combined with the high purchase cost, we tend not to consider the Virtools platform as an appropriate tool for developing the targeted information visualization.

4.2.3 Unity

The Unity game engine is a recent game engine under development by Unity Technologies. The engine focuses on a fast and intuitive game development for diverse hardware and software environments, including iPhone and iPad applications, immersive installations, etc. [29]. After testing, it was considered as one of the best game engines in terms of usability, intuitiveness, and resulting quality in graphics and interactivity. The game engine relies mostly on import through the FBX file format, analogous to other, similar game engines. Unity provides a useful API that is accessible through C# scripts and JavaScript for basic game engine functionality [29]. The more advanced API components, such as a VR library, a physics engine, a multiuser library, etc., are not available out-of-the-box, resulting in a compact, functional API for fast application development.

In our investigation of Unity, we experienced a remarkable support, elaborate documentation and an active user community, indicating a high level of user satisfaction. The fact that other initiatives for VR visualization, including goBIM for instance [11], rely on the Unity engine, indicates the appropriateness of the game engine. The availability of a free version adds to these advantages.

4.2.4 Quest3D

The Quest3D engine, developed by Act-3D, differs considerably from the other game engines described here. This difference is mainly caused by the heavy reliance on “channel graphs” to express interactivity in the resulting virtual world [30]. Channel graphs are graphs that are continuously called when running the virtual world, and depending on the ‘channels’ contained in the graph, one or another action / interaction is triggered.

The connection with existing CAD tools relies heavily on import/export/conversion plug-ins developed by third parties. As Quest3D information is described in an in-house format, any external 3D description needs to be converted into this data structure. An important information loss is experienced in this conversion, not to mention the hard balancing exercise between advantages and disadvantages of the several conversion alternatives. After experimenting, the Quest3D engine proved not as intuitive as was originally expected. Notwithstanding the promising character and the nice results of the Quest3D game engine, we found the game engine inappropriate for the targeted fast and intuitive visualization of 3D building models and their information.

4.2.5 ShiVa

The newly emerging StoneTrip ShiVa game engine [31] focuses on compatibility with a whole range of existing DCC applications through the DAE and FBX file formats, including Blender, 3DS Max, Maya, Cinema4D, etc. In order to get a building model from a BIM modeling environment into the ShiVa visualization environment, one is thus required to get the model in any of these DCC applications and export it again in a DAE or FBX format, which might cause a certain amount of information loss. The provided import/export functionality via DWG provides an alternative workflow, but this is not the ideal approach considering the richness of the IFC format, for instance.

Furthermore, ShiVa seems to provide all the basic functionality one could expect from a game engine nowadays. Scripting possibilities via Lua are provided, an API provides developer access to the engine, and the environment can be published into a myriad of environments, including mobile hardware systems such as an iPhone and iPad. To conclude, ShiVa can provide a solid basis for an information visualization environment. Being a young and small company compared to the companies considered above, a somewhat lesser support and maintenance may nonetheless be expected for the moment.

4.3 The Best Choice?

Purely based on the functionality provided by each of the game engines, only relatively small differences were found between the engines, of which the extent can only be experienced through a far more detailed study *and* usage of each of these engines. Every game engine can to some extent be used as a basis for implementing a visualization environment for the architectural information linked to the building models on a semantic web, and none of them seems perfectly fit. Based on the extra characteristics of the game engines, such as cost, development support and popularity, we found the Unity engine most promising. We therefore chose to use this engine for building an example virtual environment in which a 3D building model is visualized and the semantic architectural information can be interactively accessed in real-time for each of the elements of the building model.

5. Accessing the Architectural Semantic Web from Within the Unity Game Engine

5.1 Information Exchange

For testing the visualization of a building and the information available on the semantic web for this building, we modeled a design built in Antwerp by architects R. De Meyer and F. Van Hulle, using Revit Architecture 2010. Building information concerning the building design was added to the model as in every usual BIM model using the Revit BIM environment. This information was exported using the IFC export utility standard provided in any mature BIM application, including Revit Architecture 2010. The information in this IFC file was then converted into an online RDF graph using our in-house IFC-to-RDF converter [32]. References were added to other information in the online open linked data cloud, including more detailed material information and geographic information [4]. This information is now accessible through an online SPARQL endpoint. Through this endpoint, any information processing system may access the information described.

We did not found any game engine able to build a virtual world based on an IFC description. As the IFC format targets interoperability mainly in construction industry, it is near to useless for a game engine. As an alternative approach, one could choose to exchange the 3D information using the triangulated mesh representation made available through the API of the modeling application, as was done in [11]. We however chose to rely on a well used file format, ideally a formal or informal standard. Industry standards in animation industry include formats such as FBX and DAE. As Unity provides standard 3D exchange through FBX, this was chosen as the communication medium of the 3D information. Note that the amount of information that can be described using the IFC format exceeds that of FBX, which focuses solely on graphical information.

5.2 Creation of the Virtual World

After import of the 3D model in FBX into Unity, a few actions enable the creation of an interactive virtual world. The mere inclusion of a terrain, a global directional light and a First Person Controller enable a fast production of a basic virtual world. After the import of the FBX file, every building element that was modeled in Revit Architecture is available as a separate object, identified by the original CAD object ID. As this CAD object ID is also present in the IFC description and thus also in the semantic web graph (Fig. 1), a connection between the object in the virtual world and the information in the semantic web graph is available.

5.3 Implemented Functionality

Several additional scripts allow a connection from within the Unity game environment with the SPARQL endpoint on the server. When connected with this endpoint, any possible query can be sent through over HTTP. The query result received by the script can be processed or visualized as wished or required in the virtual world. At the moment we implemented functionality so that any AEC specialist can intuitively walk around in the virtual 3D world (Fig. 2, left) and select any object using the available pointer. Upon selecting an object, a query is fired towards two parallel semantic web servers, the one containing a number of IFC models in one semantic web graph, and the other

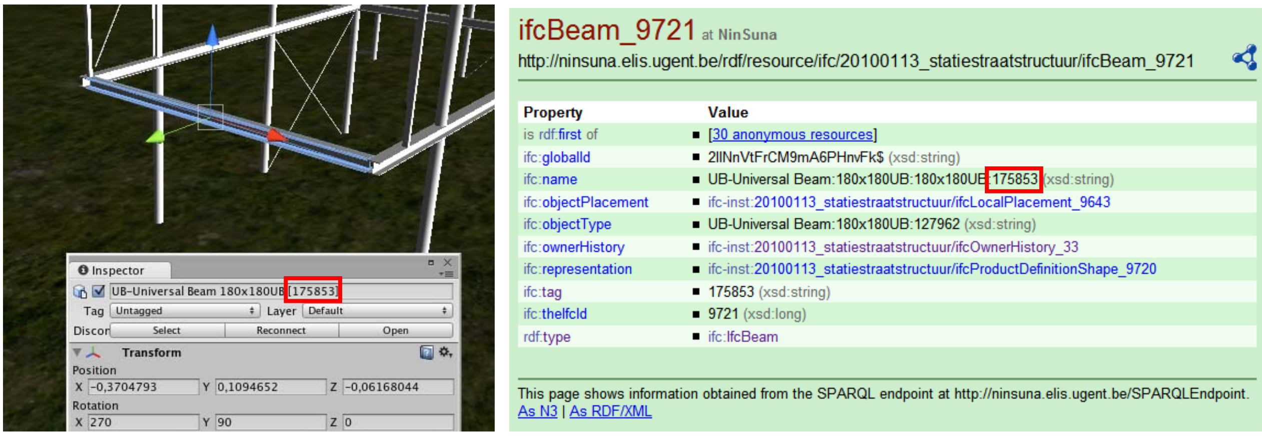

Fig. 1 **The unique CAD object ID is available in unity (left) and in the semantic IFC/RDF graph (right).**

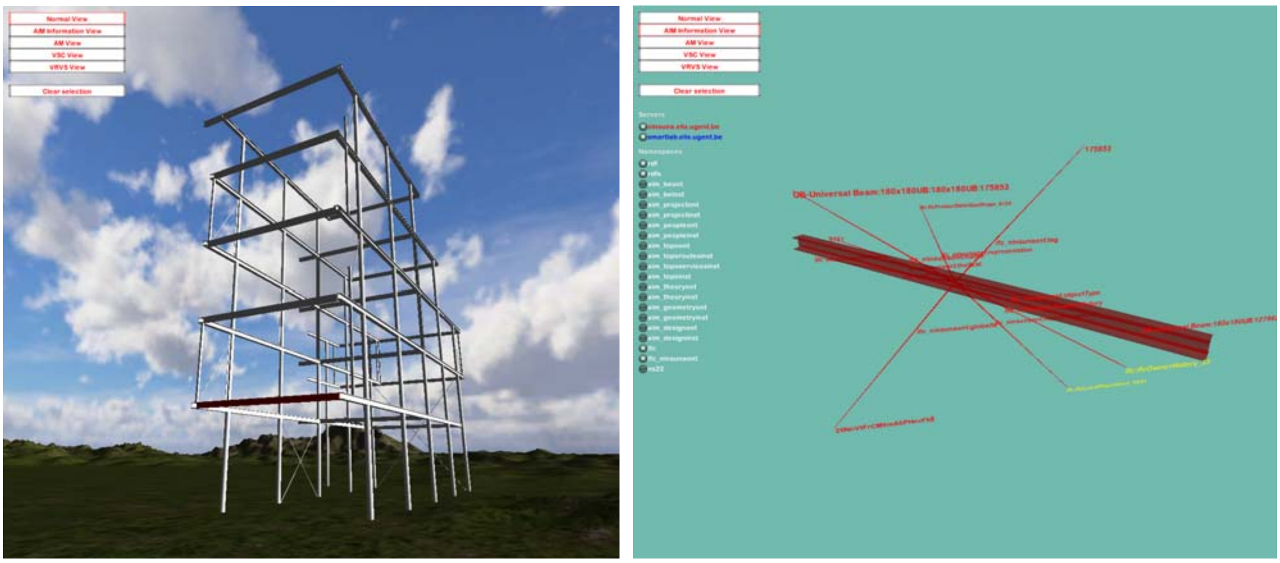
Fig. 2 **Virtual world with the building structure of the visualized design, both in an overall First Person View (left) and in an isolated view for the building element highlighted earlier in Fig. 1 (right).**

containing extra architectural information (e.g., topological information, material information, etc.) added to these original IFC models. The user view changes into a separate focus view that isolates the selected element from the rest of the building and merely displays the information available for this element (Fig. 2, right). This information is displayed as a three-dimensional graph that can be further explored by selecting the nodes in this graph.

5.4 Evaluation

The main objective of this research was to find an appropriate virtual environment allowing easy visualization of 3D building models typically developed within the AEC domain *and* the information adhered to these models using semantic web technology. The Unity game engine has indicated this functionality, and an initial implementation of the targeted functionality was implemented.

However, because research primarily focuses on the connection between a game engine and this semantic building information, we did not elaborate on the interface displaying the information. As most AEC specialists are not familiar with a 3D graph representation in a virtual world, it proved not the most intuitive interface for the targeted functionality. As an alternative, the information might be better visualized in a form-like view, similar to existing CAD applications. We also recognized the importance of selective views on the information of the building elements. No one benefits from a 3D information “explosion”, so further interface optimization is needed also on this topic. As this is ongoing research, further reports on these topics are to be expected.

6. Conclusion

This research shows how the combination of game engine and semantic web technology may enable an advanced visualization of design information in an attractive, possibly very realistic and interactive environment. Considering further enhancements, such an environment may well be an appropriate extension of the designer’s workspace, better than a combination of two-dimensional drafting technologies and a set of disparate databases. We also believe that the usage of game engine technology may provide the domain of semantic web technology a way to visualize information in a more intuitive and more attractive environment, especially when considering building and architectural information.

We are aware that this end-to-end research requires more in-depth research at certain critical points in the decision-making, but we argue that it represents useful considerations on the topic. A more in-depth investigation of game engines, of VR and MR in general, and of 3D modeling functionality in these environments would improve the system considerably, but were out of scope for this test case research. Further research could also focus on the usage of query systems or rule engines in the visualized environment to enable the appropriate visualization of an automated checking of the building designed.

Acknowledgements

This research is part of research conducted at the UGent SmartLab, supported by both the Department of Electronics and Information Systems and the Department of Architecture and Urban Planning of Ghent University. The authors gratefully acknowledge the funding support from the Research Foundation - Flanders (FWO). Equally grateful acknowledgement goes to Ir. Arch. Tom Jonckheere, who provided, during his master thesis, an important part of the research work essential for the results documented in this paper.

References

[1] C. M. Eastman, P. Teicholz, R. Sacks and K. Liston, BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Architects, Engineers, Contractors, and Fabricators, John Wiley and Sons, Hoboken, NL, 2008.

[2] T. Liebich, Y. Adachi, J. Forester, J. Hyvarinen, K. Karstila, K. Reed, S. Richter and J. Wix, Industry foundation classes IFC2x edition 3 technical corrigendum

1, available online at: http://buildingsmart-tech.org /ifc/IFC2x3/TC1/html/index.htm, accessed on 4th Nov., 2011.
[3] W3C, W3C Semantic Web Activity, available online at: http://www.w3.org/2001/sw/, accessed on 4th Nov., 2011.
[4] P. Pauwels, R. De Meyer and J. Van Campenhout, Interoperability for the design and construction industry through semantic web technology, in: Semantic Multimedia-5th International Conference on Semantic and Digital Media Technologies, Lecture Notes in Computer Science 6725 (2011) 143-158.
[5] Y. S. Yeong, C. M. Eastman, R. Sacks and I. Kaner, Benchmark tests for BIM data exchanges of precast concrete, Automation in Construction 18 (2009) 469-484.
[6] P. Pauwels, D. Van Deursen, J. De Roo, R. Verstraeten, R. De Meyer, J. Van Campenhout and R. Van de Walle, A semantic rule checking environment for building performance checking, Automation in Construction 20 (2011) 506-518.
[7] L. Khemlani, Exploring second life and its potential in real life AEC, AECBytes “Building the Future” Article, available online at: http://www.aecbytes.com /buildingthefuture/2007/SecondLife.html, last accessed on 4th Nov., 2011.
[8] Sketch Worlds, Your real-time 3d interactive virtual world, available online at: http://www.sketchworlds.com, last accessed on 13th Jan., 2011.
[9] Autodesk, Navisworks–Project review software, available online at: http://usa.autodesk.com/navisworks, last accessed on 4th Nov., 2011.
[10] Graphisoft, GRAPHISOFT virtual building explorer for ArchiCAD, available online at: http://www.graphisoft. com/products/virtual-building-explorer, last accessed on 4th Nov., 2011.
[11] I. Keough, goBIM: BIM review for the iPhone, in: Proceedings of the 29th Annual Conference of the Association for Computer Aided Design in Architecture, Chicago, IL, 2009, pp. 273-277.
[12] ICEvision, ICEvision–Spread serious fun, available online at: http://icevision.ice-edge.com, last accessed on 4th Nov., 2011.
[13] Layar, Augmented reality browser: Layar, available online at: http://www.layar.com, last accessed on 4th Nov., 2011.
[14] Wikitude, Wikitude, available online at: http://www.wikitude.org, last accessed on 4th Nov., 2011.
[15] D. H. Shin and P. S. Dunston, Evaluation of augmented reality in steel column inspection, Automation in Construction 18 (2009) 118-129.
[16] M. Golparvar Fard, P. M. Feniosky and S. Savarese, D4AR–A 4-dimensional augmented reality model for automating construction progress monitoring data collection, processing and communication, Journal of Information Technology in Construction 14 (2009) 129-153.
[17] A. Webster, S. Feiner, B. MacIntyre, W. Massie and T. Krueger, Augmented reality in architectural construction, inspection, and renovation, in: Proceedings of the ASCE Third Congress on Computing in Civil Engineering, Anaheim, CA, 1996, pp. 913-919.
[18] S. El-Tawil and V. Kamat, Rapid reconnaissance of post-disaster building damage using augmented situational visualization, in: Proceedings of the 17th Analysis and Computation Specialty Conference, St.-Louis, MO, 2006, pp. 1-10.
[19] G. Roberts, A. Evans, A. Dodson, B. Denby, S. Cooper and R. Hollands, The use of augmented reality, GPS and INS for subsurface data visualisation, in: Proceedings of the XXII Congress of the International Federation of Surveyors (FIG), Washington, D.C., 2002.
[20] B. Thomas, W. Piekarski and B. Gunther, Using augmented reality to visualise architecture design in an outdoor environment, Design Computing on the Net, Sydney, NSW, 1999.
[21] N. Al-Najdawi, Introduction to visualization using game engines, in: AHRC Methods Network Workshop “From Abstract Data Mapping to 3D Photorealism: understanding emerging intersections in visualisation practices and Techniques”, 2007.
[22] S. Marks, J. Windsor and B. Wünsche, Evaluation of game engines for simulated clinical training, in: Proceedings of the New Zealand Computer Science Research Student Conference, Christchurch, NZE, 2008, pp. 92-100.
[23] B. C. Wünsche, B. Kot, A. Gits, R. Amor, J. Hosking and J. Grundy, A framework for game engine based visualizations, in: Proceedings of the 20th Image and Vision Computing New Zealand, Dunedin, NZE, 2005, pp. 465-470.
[24] J. Moloney, R. Amor, J. Furness and B. Moores, Design critique inside a multi-player game engine, in: Proceedings of the CIB W78 Conference on Construction IT, Waiheke Island, NZE, 2003, pp. 255-262.
[25] T. Koehler, A. Dieckmann and P. Russell, An evaluation of contemporary game engines, in: Proceedings of the 26th eCAADe Conference, Antwerp, BE, 2008, pp. 743-750.
[26] Esperient, Interactive architectural visualisation with Esperient creator, Esperient Whitepaper, available online at:http://www.esperient.com/images/Whitepapers/Architectural%20Visualization.pdf, last accessed on 4th Nov., 2011.
[27] Esperient, Esperient creator-3DS Max to creator workflow, Esperient Whitepaper, available online at: http://www.esperient.com/images/Whitepapers/MaxtoCreatorWorkflow.pdf, last accessed on 4th Nov., 2011.

[28] Dassault Systèmes, 3DVIA Virtools - Dassault systèmes, available online at: http://www.virtools.com, last accessed on 4th Nov., 2011.
[29] UNITY, UNITY: Game development tool, available online at: http://unity3d.com, last accessed on 4th Nov., 2011.
[30] Quest3D, Welcome!–Quest3D, available online at: http://quest3d.com, last accessed on 4th Nov., 2011.
[31] Shiva, ShiVa 3D game engine with development tools, available online at: http://www.stonetrip.com, last accessed on 4th Nov., 2011.
[32] UGent MultiMediaLab, IFC-to-RDF service, available online at: http://ninsuna.elis.ugent.be/IfcRDFService, last accessed on 4th Nov., 2011.