Multidisciplinary Design in Virtual Worlds

ROSENMAN M.A.[1], SMITH G.[1], DING L.[2], MARCHANT D.[3] and MAHER M.L.[1]

[1]*Key Centre of Design Computing, University of Sydney*
[2] *CMIT, Commonwealth Scientific and Industrial Research Organisation, Australia*
[3] *Woods Bagot, Sydney, Australia*

The research described was carried out by the Australian Cooperative Research Centre for Construction Innovation

Keywords: Collaboration, Multiviews, Virtual Worlds, Agents, CAD

Abstract: Large design projects, such as those in the AEC domain, involve collaboration among a number of design disciplines, often in separate locations. With the increase in CAD usage in design offices, there has been an increase in the interest in collaboration using the electronic medium, both synchronously and asynchronously. The use of a single shared database representing a single model of a building has been widely put forward but this paper argues that this does not take into account the different representations required by each discipline. This paper puts forward an environment which provides real-time multi-user collaboration in a 3D virtual world for designers in different locations. Agent technology is used to manage the different views, creation and modifications of objects in the 3D virtual world and the necessary relationships with the database(s) belonging to each discipline.

1 INTRODUCTION

Large design projects, such as those in the AEC domain, involve collaboration among a large number of participants from various design disciplines. With the increase in CAD usage in design offices, there has been an increase in the interest in collaboration using the electronic medium (Kvan, 1995; Wojtowicz, 1995; Maher and Rutherford, 1996), together with the advance in electronic representation and standardization of design information (IAI, 2000). Collaboration among different participants in the design of a building involves both synchronous and asynchronous communication. It involves the ability of the different participants to work on their part of the project using their own particular ways of working yet being able to communicate with the other participants to bring about a common objective, the design of the building. Digital collaboration raises new issues such as keeping track of versions, ownership and ensuring that decisions made are recorded and transmitted to the necessary participants. Shared data models (Yasky, 1981; Wong and Sriram, 1993; Krisnamurthy and Law, 1997) have been put forward for over twenty years as the answer to many of these problems.

Traditionally, the representation of designs has been effected by each discipline producing its own set of drawings i.e. that discipline's representation (or model) of their view of the building. The various sets coexist, may share some commonalities, but are separate representations. The use of a single shared database, as usually proposed, does not take into account these different representations required by each discipline nor does it allow for synchronous real-time multi-user collaboration through electronic collaboration as the amount of information present makes such communication impractical.

This paper presents a collaborative virtual environment for multidisciplinary design based on the need for extending the shared database to take into account the needs of the various views. It focuses on the extensions to a shared model required to address the following issues: different decomposition schema of the model among the collaborators; relationships within and across the different schema; multiple representations and versioning of elements; ownership and access to elements and properties of elements and shared visual representation in a 3D virtual world

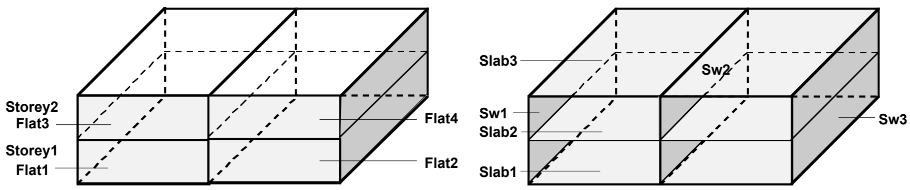
2 COLLABORATIVE VIRTUAL ENVIRONMENTS AND COLLABORATIVE DESIGNING

What is required is an environment that allows real-time multi-user collaboration by designers in different physical locations. This environment must provide 3D visualisation, walkthroughs and rendering to allow communication of the various views of the design as modelled by the different disciplines. This is of special importance at the conceptual stage of the design since much of the early collaborative decision-making is carried out at this stage. A virtual world environment based on an underlying object-oriented representations of the design is put forward here as a necessary environment for synchronous collaboration in the design of buildings. This is in contrast to the decision made by Lee et al. (2003) to use a commercial CAD system or visualisation. One of the main advantages of virtual world environments is that it allows users to be immersed in the environment, allowing for real-time walkthroughs and collaboration (Savioja et al., 2002; Conti et al., 2003). Moreover, CAD models contain a great deal of detail which makes real-time interaction extremely difficult. Agent-based technology will be used to provide the necessary communication between the users, the virtual world views and the object-oriented models.

2.1 Example Problem

This section presents a simple example scenario to illustrate the issues involved in early conceptual design involving architects and structural engineers working both asynchronously and synchronously.

The architects create their initial conceptual spatial design (model) of the building (2 storeys, 4 spaces) in own CAD space, Figure 1 a). The architects' model contains a building object as an aggregate of storey objects which are aggregates of space (flats) objects. A bulletin board entry of change to architects' model is made. An email message is sent to all participants as well.



a) Architect's initial design **b) Engineers' initial design**

Figure 1. Initial designs

The engineers view the architects model and, based on their understanding, create their initial conceptual structural system design (model) of the building (3 shear walls and 3 slabs) in own CAD space, Figure 1 b). The engineers' model contains a building object as an aggregation of slabs and shear walls. The engineers add in relationships between their elements and architects' elements, namely that the walls and slabs bound the storeys and flats. A Bulletin board entry of change to engineers' model and relationships is made. An email message is sent to all participants as well.

The architects wish to modify Flats 3 and 4 such that a change in the engineers' Sw2 would occur. This results in a notification of existing relationships between Flats 3 & 4 and walls and slabs in the engineers' model. The architects examine the relationships by viewing the engineers' model and see that they must confer with the engineers. The architects call a meeting with engineers (email message or posting on bulletin board).

A meeting is held in the VW for the discussion of desired changes. The architects' view is presented as a version in the VW and the architects' propose to increase the size of Flat3 and decrease the size of Flat4. The view is switched to the engineers' view in the VW and the ramification of the changes to the structural system are discussed. An agreement is reached to proceed with modification and permission is granted to go ahead with changes.

The architects' version is committed to the 'legal' model, by the agents. A bulletin board notification is made and email messages sent. The engineers make changes to their model (either in the VW or in their CAD system) and update the relationships between their model and the architects model. A bulletin board notification is made and email messages sent.

The above example gives an indication of the objects and relationships required as well as the notifications that would have to be made when working asynchronously. Some relationships may exist either as intra-discipline relationships and/or inter-discipline relationships, whereas others may be just inter-disciplinary. The correspond_to(a, b) relationship, Figure 2 is such an inter-disciplinary relationship. A one-to-many /many-to-one corresponds_to relationship would also be required when the architects may have several wall objects above each other on different floors and the structural engineers would have only one wall object. Little attention has been paid to inter-discipline relationships in modelling. For example, the IFC schema (IAI, 2000) do not have such relationships. Since inter-discipline relationships are an essential part of multidisciplinary collaboration, and, if IFCs are to be useful in this area, they will require to be extended.

3 MULTIDISCIPLINARY MODELLING

The views and, hence, models of different design disciplines are founded on the functional concerns of those disciplines. In a design context, the view that a person takes depends on the functional concerns of that person. A building may be viewed as a set of activities that take place in it; as a set of spaces; as sculptural form; as an environment modifier or shelter provider; as a set of force resisting elements; as a configuration of physical elements; etc. A building is all of these, and more. A model of an object is a representation of that object resulting from a particular view taken. For each different view of a building there will be a corresponding model. Depending on the view taken, certain objects and their properties become relevant. For the architects, floors, walls, doors and windows, are associated with spatial and environmental functions, whereas structural engineers see the walls and floors as elements capable of bearing loads and resisting forces and moments. Both models must coexist since the two designers will have different uses for their models. According to Bucciarelli (2003) "There is one object of design, but different object worlds." and "No participant has a 'god's eye view' of the design.".

A single model approach to representing a design object is insufficient for modelling multiple views to the views taken by the different viewers (Rosenman and Gero, 1996). Each viewer may represent an object with different elements and different composition hierarchies. While architects may model walls on different floors as separate elements, the structural engineers may model only a single shear wall. Each discipline model must, however, be consistent vis-a-vis the objects described. While Nederveen (1993), Pierra (1993) and Naja (1999) use the concept of common models to communicate between the discipline models, it is never quite clear who creates the common models and maintains the relationships between them and the discipline models In this project, this consistency will be provided by interrelationships between the various objects in different disciplines modelled by explicit (bidirectional) links from one object to another. Figure 2 shows an example of this approach, with each discipline labeling its objects according to its need. While this approach may have the disadvantage of replicating the same information, it saves the complexities of creating the common concepts and allows each discipline great flexibility in creating its model. The discipline models allow each discipline to work according to its own concepts and representations. The whole model may be seen as the union of the different models.

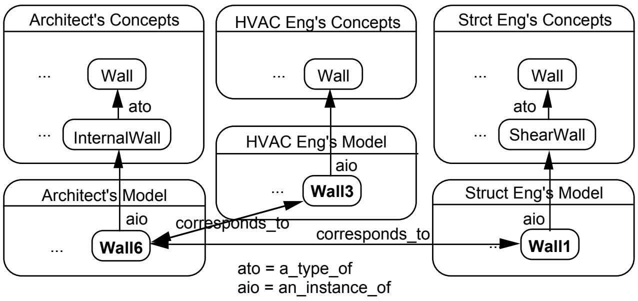


Figure 2. Discipline models and relationships

Two issues of concern, in any collaborative environment, are those of keeping multiple versions of a model and who controls what. In this work, ownership is defined by assigning purpose and functional properties to objects. Thus, an object with both spatial and structural functions, is 'owned' by both the architect and engineer. There is one 'legal' approved version of the total model (i.e. of each discipline), and any modifications to this model require approval by all the 'owners' of the objects which are the subject of modification. Versions are the property of disciplines and thus any 'what if' scenario can be carried out and presented for discussion in the collaborative environment. If approved, the modifications can be committed to the 'legal' version.

4 THE SYSTEM ARCHITECTURE AND PROTOTYPE IMPLEMENTATION

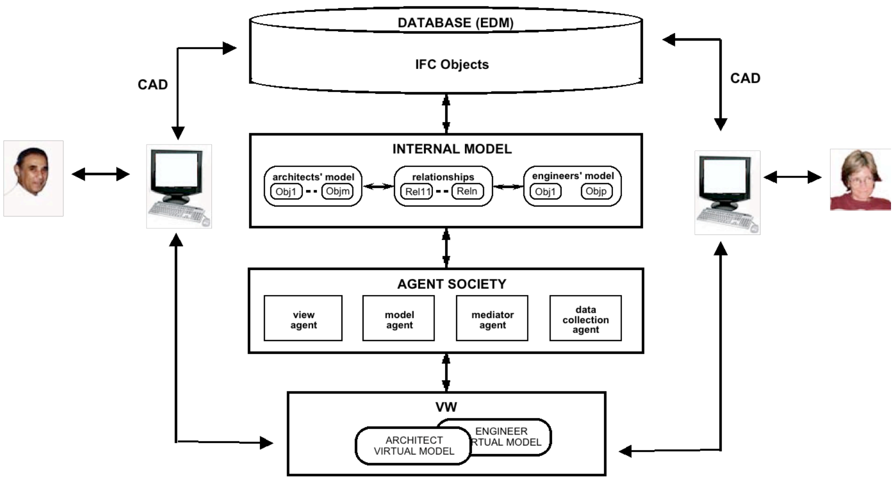


Figure 3. System architecture

Figure 3 shows the system architecture.

4.1 The Database and Internal Model

The designers may enter their design through their own CAD systems, thus populating the (external) database with information about the objects and their properties. Notwithstanding the shortcomings of IFC schemas at present, this project uses IFCs (IAI, 2000) as the standard format allowing interoperability between representations. The IFC objects will be stored in an EDM database for persistence. Since the virtual world models do not require all the IFC detailed information, this information is converted into a simpler form and stored in an internal relational database for simple communication with the agent system and the virtual world. When users request a view in the virtual world, the appropriate agent will query the internal database to extract the necessary objects to display. The one-to-one and one-to-many relationships are stored in the internal model. To date the following relationships are recognized by the internal model: Aggregates, Composes, CorrespondsTo, Connects, Bounds and Loads.

Alternatively, the designers may create or modify some objects in the virtual world during a collaborative session. Agents will create the respective objects in the internal database. When committed, the new or modified objects will be translated and transferred to the IFC/EDM database and thus be available to the various CAD systems. Versions of the model reside in the external database. The internal model always holds the current working model.

4.2 The Agent Society

The primary role of the agents is to construct and maintain multiple views of 3D objects instantiated in a virtual world. Agents also provide an interface between the 3D objects from which the 3D virtual worlds are constructed and the database objects that comprise the multiple-views model. Mediator agents associate 3D objects with designers and their 3D world avatars, handle text chat from designers to agents, handle communication between servers and agents, and control the work flow between agents. Data collector agents provide for logging and data collection for later cognitive and data mining analysis or simply as a record of important collaborative sessions.

The internal model shown in Figure 3 contains a set of instances of Component and Association classes, which are instantiated either from a CAD model (via the external database and IFCs) or from the virtual world by an agent. The Component class is specialised by classes Wall, Slab, Beam, Column, Storey and Space. Each Component can also have associations with other Components. The Association class is specialised by classes Corresponds_to, Bounds, Aggregate, and Loads. Components belong to a project, are owned by a citizen (a designer) with a personality (architect, engineer, etc), and are declared to be of one or more functional categories (spatial, structural, aesthetic, etc). Component and Association objects encapsulate tables in a relational database. This relational database both provides persistence for the agents as well as reducing the coupling between the agents and the external database.

The external database contains CAD models from different disciplines and the internal model is populated with Component and Associations accordingly. Maintaining the internal model with respect to changes in the virtual world or changes in the external (EDM) database is the role of the model agent. Views in the virtual world are constructed and reconstructed by view agents through queries on the internal model according to specified personalities, component owners, categories and so on. According to the view required, selected by clicking on a designer view in the web page builder (see section 4.3), the view agent decides which objects are relevant and converts these into AW objects. Because the geometry of the virtual world will be different to that in the external database, view agents maintain a separate set of objects. An opposite process, whereby objects are created in the AW environment, has the agents producing internal model objects which in turn create IFC objects in the EDM database. The Components in the relational database of the internal model can therefore be seen as providing persistence for the agents, reduction of coupling between the agents and the external database, and a simplified intermediate geometry that is accessible from both the external database and the virtual world.

Modifications, such as architects wishing to move a wall, are to be sent via the mediator agent to the model agent who will check whether the modification is permitted. This will be done by checking whether any relationships exist between the object and objects in other disciplines, i.e. whether the object is 'owned' by other disciplines. The model agent may then send back a notification to the mediator agent that this is not permitted and the architects will be notified that they must discuss this with, e.g. the structural engineers. Together with the owner association of a component will be a list of those permitted to make modifications to related objects. Granting permission will mean adding a discipline with the permission to manipulate the objects even though they are owned by another discipline. Alternatively, if no such relationship exists, the model agent will permit the request, update the model accordingly and the view agent will update the view. A discipline may create any number of versions of its own model, since versions are not 'legal' and may make any modifications to its objects within that version model. This version model may be presented in the VW for discussion. If agreement is reached, the version may then be committed as the 'legal' version. Related changes to other discipline models must be made by the discipline concerned.

4.3 The Virtual Collaborative Environment

Figure 4 shows the CRC Collaborative Designer (CCD). CCD is a prototype environment that supports collaboration by augmenting the inherently multi user Active Worlds (AW) platform with additional collaboration tools. Designers interact with a virtual world, other designers and agents via the different browser panels. The large 3D panel shows the 3D world, including artefacts being built and the avatars of designers. The panel below the 3D view panel facilitates chat. While streaming audio is also used by CCD, text chat provides persistence and encourages brainstorming-like interaction. The narrow panel at the far left provides for navigation between worlds, teleports, telegrams, contacts, and help pages. The large panel at the far right shows dynamically served web pages that provide more information about the design and runs interactive applications. These applications include a builder (which will be discussed later), a webcam and streaming audio, a distributed sketchpad, and a logging facility for use by cognitive experimenters.

We are using the Active Worlds (AW) platform as a basis for development. This is because the AW server provides a platform for distributed collaboration upon which we can build, and because the virtual world is constructed at runtime from within the world. We enhance the collaborative experience by driving the web panel from the server side off an Apache Tomcat HTTP server. This serves Java Server Pages and Java Servlets from designer actions on the web panel. One reason for choosing Tomcat as the HTTP server is because the agents are implemented in Java, and so agents communicate with the HTTP server using Java Remote Method Invocation. The agents also use a Java Native Interface to the AW software development kit, enabling agents both to sense the world and to effect changes to it.

Figure 4 shows the prototype in its current form.

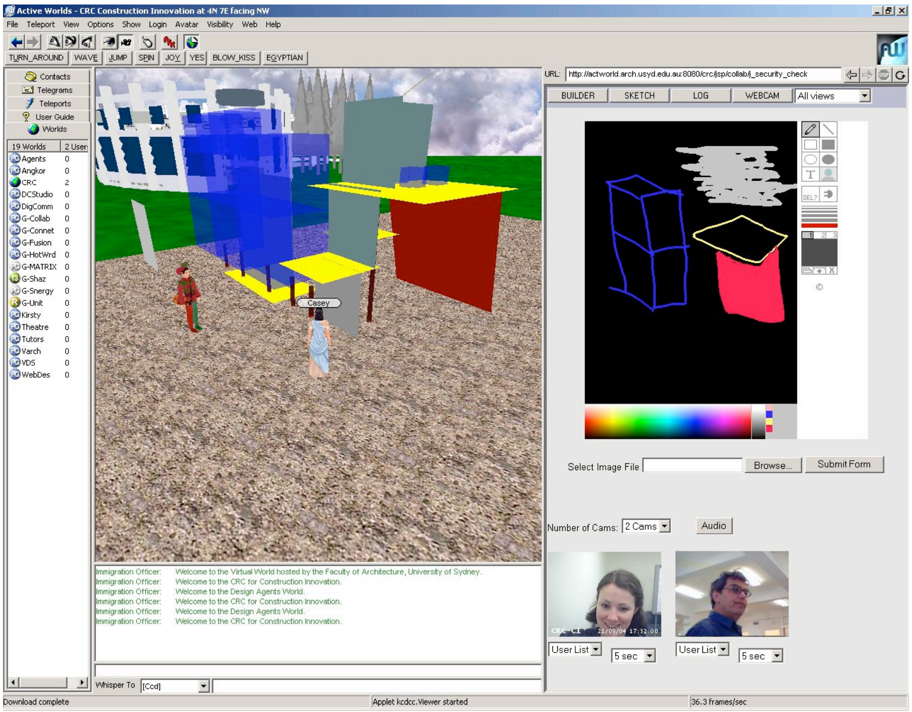


Figure 4. The 3D virtual world collaborative environment

Two designers are shown, both as webcam views and as their avatars in the 3D world. The line of buttons at the top of the web panel shows the main applications The Figure shows ‘All views’. If ‘Architect’ was selected; then the view shown would be rebuilt by an agent such that only objects declared to be of interest to architects would be shown.

To create objects in the world, the 'BUILDER' button is selected. A 'list box' appears. This listbox is used to select the personalities of interest and insert objects accordingly. For example a column may be selected from the 'Engineers palette'. Both the list of objects and the palettes are queried from the internal model. To add new objects to the world, the designer moves to a desired location and clicks on the 'INSERT' button. This results in a message being sent to the agents, and a new object being inserted in the world.

5 CONCLUSION

This paper has presented a framework for multidisciplinary collaborative design. It has discussed the need for modelling all the views of the various disciplines and the need for specifying the relationships between the various models. Collaboration takes place in a virtual world environment because of the multi-user and immersive properties of such environments. The paper presented a framework for collaborating in a virtual environment including a database, based on IFCs, containing the various models and relationships between them; a virtual world environment for collaboration and an agent-based society for handling the communication between the users, the virtual world and the database. An internal database simplifies the work of the agents and also decouples the agents from existing (IFC/EDM) technology. If the representation method for representing objects in CAD systems were to change, the agent system would not need to.

The paper has highlighted the need for extending the IFCs to include interdisciplinary relationships as well as extending the scope of IFC objects. Future work includes finalising the mapping from IFC objects to the internal database, including the notification messaging and testing the system more fully.

ACKNOWLEDGEMENTS

This work is part of the Cooperative Research Centre for Construction Innovation project, Team Collaboration in High Bandwidth Virtual Environment.

REFERENCES

Bucciarelli, L. L. 2003. Designing and learning: a disjunction in contexts. *Design Studies*, 24(3):295-311.

Conti, G., Ucelli, G., and De Amicis, R. 2003. JCAD-VR – a multi-user virtual reality design system for conceptual design, in TOPICS. *Reports of the INI-GraphicsNet*, 15:7-9.

IAI. 2000. Industry Foundation Classes-Release 2x, IFC Technical Guide. http://www.iai-international.org/iai_international/Technical_Documents/documentation/IFC_2x_Technical_Guide.pdf; accessed 10 November 2004

Krishnamurthy, K. and Law, K. H. 1997. A data management model for collaborative design in a CAD environment. *Engineering with Computers*, 13(2):65-86.

Kvan, T. 1995. Fruitful exchanges: professional implications for computer-mediated design, *The Global Design Studio*, M. Tan and R. The, eds, Centre for Advanced Studies in Architecture. University of Singapore, pp.24-26.

Lee, K., Chin, S. and Kim, J. 2003. A core system for design information management using Industry Foundation Classes, *Computer-Aided Civil and Infrastructure Engineering*, 18:286-298.

Maher, M. L. and Rutherford, J. 1996. A model for collaborative design using CAD and database management. *Research in Engineering Design*.

Naja, H. (1999). Multiview databases for building modelling. *Automation in Construction*, 8:567-579.

Nederveen, S.V. 1993. View integration in building design. *Management of Information Technology for Construction*. Mathur, K.S., Betts, M.P. and Tham, K.W. eds. Singapore:World Scientific. 209-221.

Pierra, G. 1993. A multiple perspective object oriented model for engineering design. *New Advances in Computer Aided Design & Computer Graphics*, Zhang, X. ed. Beijing: International Academic Publishers, 368-373.

Rosenman, M.A. and Gero J.S. 1996. Modelling multiple views of design objects in a collaborative CAD environment. *CAD Special Issue on AI in Design*, 28(3): 207-216.

Savioja, L., Mantere, M., Olli, I, Ayravainen, S. Grohn, M and Iso-Aho, J. 2003. Utilizing virtual environments in construction projects, ITCon, **8**:85-99, http://www.itcon.org/cgi-bin/papers/Show?2003_7;

Wojtowicz, J. (ed.) 1995. *Virtual Design Studio*. Hong Kong: Hong Kong University Press.

Wong, A. and Sriram, D. 1993) SHARED An information model for cooperative product development. *Research in Engineering Design*.

Yasky, Y. 1981. *A consistent database for an integrated CAAD system: fundamentals for an automated design assistant*. PhD Thesis. Department of Architecture, Carnegie-Mellon University.