Teaching and Learning in 3D Immersive Worlds: Pedagogical Models and Constructivist Approaches

Amelia Cheney
Appalachian State University, USA

Robert L. Sanders
Appalachian State University, USA
Chapter 11

3D Collaborative Virtual Environment to Support Collaborative Design

Mohd Fairuz Shiratuddin
The University of Southern Mississippi, USA

Alen Hajnal
The University of Southern Mississippi, USA

ABSTRACT

Constructivist learning emphasizes students’ involvement in the learning process, how they become self-directed learners and actively engaged in the learning environment. This chapter describes Collaborative Virtual Environment (CVE) and its relevance to collaborative learning and constructivism. The authors developed the Collaborative World Design Tool (CWDT) software to evaluate the benefits of CVE for architectural design students. The CWDT was developed based on the Torque 3D Game Engine, thus the appearance and functionalities of the CWDT within the CVE are similar to computer game playing environment. In the experiment, subjects designed within the CVE, were either experts or novices, worked either individually or in pairs, and constructed a virtual building in a three-dimensional outdoor environment. Results show that working collaboratively within a CVE has great potential to increase performance where teamwork is faster than individual work, and overall provide a constructive learning environment.

INTRODUCTION

Teaching and learning strategy in educational institutions has been evolving and has altered the way teachers teach and students learn. Traditional teacher-centric method of teaching has been modified and enhanced with the introduction of computer technologies. Traditionally and conventionally, knowledge is communicated by the teacher through lectures in a classroom en-
environment, where students listen and take notes. Learning tends to be passive and students play little part in the learning process because focus is emphasized on the content of the teaching material, how much material has been delivered and how much the students have learned. On the contrary, in a constructivist learning paradigm, the learning process allows students to work individually or in small groups; rather than being passive recipients, explore, investigate and solve problems, and become actively engaged in seeking knowledge and information.

Technology provides opportunities to apply a constructivist approach to teaching and learning. Modern instructional strategy and tools for K-12 and higher education are becoming more convenient and sophisticated, whether in classroom or online. Teachers use the interactive whiteboards, online Blackboard systems, internet resources, Power Point slides, etc., to effectively extend the “used-to-be only in-classroom/laboratory, spoon-fed knowledge, long-established memorization of facts, principles, or procedures of learning traditions” into the paradigm of active learning, creative thinking, analysis and evaluation, and problem solving. As such, students must play an active part in their learning process and be self-directed learners who are actively engaged in constructing new meaning within the context of their current knowledge, experiences and social environments. Students become successful in constructing knowledge through solving problems that are realistic, and even more successful when working in collaboration with others (Bruner, 1996).

The foundations of constructivist learning approach come from the cognitive approach to psychology of learning (Jonassen et al., 1999), whereas the theories are rooted in Piaget (1952), Dewey (1966), Vygotsky (1978), Papert (1980), and Bruner (1985). Constructivist learning places emphasis on learners and proposes that learning is affected by context, beliefs and attitudes. Learners are encouraged to find their own solutions and to build upon their prior knowledge and experiences. Students learn by adding and fitting new information together with what they already know and actively construct their own new understanding. As such, students gain deeper understanding of the event or knowledge, thus constructing their own new knowledge and solutions to the given problems (Duffy & Jonassen, 1992; Jonassen, 1994). Jonassen et al. (1999) propose that problems or tasks given in a constructivist environment should present certain characteristics such as interesting, engaging, appealing, authentic, personally relevant, challenging to learners, and provide a physical simulation of the real world task environment.

Virtual Environments (VE), particularly systems that are embedded with collaborative features, also known as Collaborative Virtual Environments (CVEs), provide similar characteristics for a constructivist environment in the form of virtual settings. A CVE extends a standalone VE to include real-time collaboration, interaction and sharing of the same virtual space among users across a network. The need for collaboration, sharing of information, and exchanges of experiences, led to standalone VE applications being enhanced and developed into CVE applications. Generically, a VE application is considered a CVE when information sharing, 3D visualization and real world user-interaction and object manipulation are included as built-in features (Bryson, 1996; Schuckmann et al., 1999; Theoktisto & Fairen, 2005; Liston et al., 2000).

As virtual worlds, VEs and CVEs provide teachers the opportunities to create environments or virtual settings that support cognitive occurrence through the use of near-realistic visuals, audios, real-time interactions and manipulation of objects. Students are given the opportunity to explore and navigate the virtual worlds with authentic and purposeful contexts for practice and learning that are transferable to the real world. Andolsek (1995) suggests that since virtual worlds are totally engaging, they immerse the student
both cognitively and affectively. Students will be actively engaged rather than staying passive in a virtual world. VEs have great potential in terms of increasing learner motivation. If learning is made more interesting and fun, students may remain engaged in an activity for a longer period of time.

In this chapter we describe social constructivism and collaborative learning within a CVE, and CVE as a medium for social constructivist learning. We also report an experiment that demonstrates the development of a virtual environment with collaborative features that can support collaborative learning oriented towards a social constructivist-based paradigm of learning environment. In the experiment, the collaborative architectural design task consisted of the construction of a virtual building in a three-dimensional outdoor environment. Subjects were represented by avatars capable of moving freely in the 3D space (including flying). Subjects worked alone or in pairs, and were either experts (architecture undergraduate students) or novices sampled from a general college student population. Based on the results, we proved that teamwork is more successful than individual work, and that experts perform better than novices.

**SOCIAL CONSTRUCTIVISM AND COLLABORATIVE LEARNING WITHIN A CVE**

Collaborative learning is the grouping and pairing of learners for the purpose of achieving a learning goal. It is an instruction method in which learners at various performance levels work together in small groups toward a common goal. The learners are responsible for one another’s learning as well as their own. Thus, the success of one learner helps other learners to be successful. Various theories of learning have served as the foundation for the conceptualization of collaborative learning, most notably constructivist approaches (Dewey, 1966; Piaget, 1973). The basic principle of constructivism is “learning by doing”. In other words, learning always involves exploration and constructing of objects in a variety of settings. Collaborative learning also borrows important features from various cognitive theories of social cognition (Fiske & Taylor, 1991) and joint attention (Sebanz et al., 2006). These include but are not limited to interpersonal coordination and synchronization of behavior (Marsh et al., 2009), task-sharing, reciprocity of information flow among partners, joint responsibility, etc.

Students benefit from personal and unique learning experiences when they work collaboratively. Other than constructivist learning (Biggs, 1996; Fosnot, 1996), it is observed that behaviorist aspects of learning (Bandura, 1977; Kretchmar, 2008) also occur when students work collaboratively. Behaviorism occurs when learning is achieved by changing an observable behavior, whereas constructivism occurs when learning is achieved when a participant’s knowledge is created through interaction with others and with the world around them. The occurrence of learning is thus prevalent in people who experience working within a collaborative immersive environment setting. Students can learn from one another because each has diverse skills, abilities and experiences (Hill, 1982; Fischer, 2005). In an immersive environment, McConnell (2000) and Palloff and Pratt (2003) suggest that students can work according to their strengths, develop creativity and critical thinking skills, validate their own and others’ ideas, and appreciate learning styles, skills, preferences and perspectives of others.

It is also observed that a CVE setting is relevant to the principles of social constructivism, and social context of learning. Unlike traditional tools for education, a CVE supports the social side of learning. A CVE soundly accounts for factors suggested by Sanders et al. (2007) and Bronack et al. (2008) such as, “social presence, serendipitous interaction, and informal learning.” Bronack et al. (2008) discuss the use of a VE called the AET Zone where a virtual social space is provided for
students (who are mostly K-12 educators working full-time and attending graduate school part-time), faculty, and others, to engage and interact in collaborative activities that offer the social aspects of teaching and learning. Bronack et al. characterized this approach as Presence Pedagogy (P2) where the teaching and learning in this environment is grounded in social constructivist theory.

Social constructivism is an extension and an enlarged view of cognitive constructivism. Cognitive constructivism centers on the individual’s mental construction of knowledge, whilst social constructivism places more emphasis on the social context of the learning environment. According to Vygotsky (1978), individual construction of knowledge and understanding of meaning grows out of social encounters. Vygotsky’s concept of social learning i.e. zone of proximal development (ZPD) describes circumstances where a student can master concepts and ideas with the help of teachers or more advanced peers and students that he or she cannot understand if he/she is on his/her own. Thus, in a collaborative learning community or environment such as the AET Zone or other form of CVEs, students learn by interacting and engaging with their more capable peers, teachers and experts.

CVE as a Medium for Social Constructivist Learning

A virtual world provides a medium for the creation of learning environments that envelop the students with a sense of immersion into the content of the environment, with the ability for the students to naturally manipulate and change the content into new understandings. VEs can provide enhanced visualizations and a range of perspectives into complex and abstract information (Salzman et al., 1999). For example, the ability to create, alter, and rotate an architectural structure in real time 3D can make it easier to understand abstract concepts (Perdomo et al., 2005). Bailenson et al. (2008), Stevens (2006), Aziz et al. (2006), and Arango et al. (2007) discuss the potential of VEs in providing rich and engaging learning experiences for students that include discovery, investigation, and creation. Within this environment, simulations and collaboration on learning materials are possible among students and teachers worldwide.

The condition in a VE is congruous to social constructivist learning principles (Dede, 1995; Papert, 1993), where students have the potential to discover content and create meaningful connections with the content through creativity and imagination. Lessons learned from these settings are transferred back into their real lives, thereby creating meaning with the content and ultimately connections with the concepts being taught and learnt. Jonassen (1994) proposes six principles of constructivist learning environments that are relevant to VE:

1. VE provides multiple versions of reality, thereby representing the natural complexity of the world
2. VE can be designed to focus on knowledge construction rather than reproduction
3. Authentic tasks can be presented within a VE
4. Activities within a VE can be crafted to foster reflective practice
5. VE can facilitate context and content-dependent knowledge construction
6. VE discourages competition among learners for recognition, but supports collaborative construction of knowledge.

The learning processes within a CVE are different from those of the traditional classroom-based environment (Liegle & Madey, 1997; Careless, 1998; Ahmad & Piccoli, 1998; McFadzean, 2001; Quinn, 2005; Chang et al., 2006; Aziz et al., 2006; Arango et al., 2007). A learning environment within a CVE allows multiple opportunities for students to be active with the content. By becoming active participants within their own learning process, students take control of their instructional
process and become the champions of their own learning. Learning content or problems presented within the learning space should encourage students to think, explore, discover, and manipulate the content to become better problem solvers and at the same time, learn and gain knowledge (Rice, 2007). Complexity of problems to be solved by students within a CVE can be designed to be a gradual process that requires students’ involvement, participation, and interaction. The more complex and challenging the problem encountered by students, the more interactions are required, thus the more meaningful the students’ experience will be. Interactions among participants within a CVE are done via avatars (Benford et al., 1995; Steptoe et al., 2008). Interactions occur firstly, among the students and teachers themselves, and secondly, with the content and the environment. Deeper interactions and involvement ultimately creates an opportunity for students to work through higher order learning. Through this meaningful interaction and engagement, and the exploration and manipulation of content within the CVE, students become more stimulated, thus bringing about opportunities for a more intense learning experience (Gagne, 1985).

Immersion is a distinctive characteristic of a CVE which allows users to experience interaction with objects and environment similar to those in the real world. This characteristic has some effects on learning and cognition. According to Carpenter and Anderson (1996), VE is able to improve learning performance i.e. user performance is increased significantly because of higher understanding of abstract problems and the opportunity to explore the VE with objects that represent the abstract entities. Traditionally, educational VE applications have been designed for a single user interacting with a virtual world (Jackson & Fagan, 2000). The use of CVEs opens up for studies of multi-user performance in VE and the potential impact of collaboration on learning (Jackson & Winn, 1999). There are also many effects in CVEs that are the consequence of social interactions and group behavior among avatars (Bailenson et al., 2001; McCall et al., 2009). Aziz et al. (2006) reports on the learning effectiveness using virtual engineering laboratory. The results indicate that the students’ scope of laboratory experience is expanded well beyond the confines of what would be feasible in the context of traditional laboratories. The virtual laboratories allow the students to learn at flexible times, place and pace; gain access to a large number of experiments, and cost savings through experiment sharing. The educational benefits from this are that more students can be exposed to wide-ranging experimental experiences, support for asynchronous learning, and promote self-learning. Arango et al. (2007) discusses the implementation of the same virtual laboratory concept using a multiplayer game engine from a pedagogical as well as technical point of view. The virtual setting provides students with an opportunity for exercising their problem solving skills by collaboratively interacting with each other and the virtual exercises.

According to Byrne (1995) and Kalawsky (1997), VE is a new and different world where teachers and educators are given a new way and avenue to teach more effectively and provide students with higher level of motivation and interest. Activities in a VE can be repeated for as long as it takes to master the concepts and principles (Barfield & Furness, 1995). Activities can also be stopped at any stage to allow participants to reflect on their performance (Hoffman et al., 2001). Learning improves even more if executed by students in a group. Hill (1982) suggests that interactive group learning outweighs individual performance because a group is able to accumulate team members’ resources, and team members can learn and correct their own errors or each other’s. Team members can utilize their individual experiences as well as those from the other group members to enhance their learning experiences and processes. This situation brings about unique learning experiences that would not be possible otherwise.
COLLABORATIVE LEARNING FOR ARCHITECTURE STUDENTS USING A 3D CVE

In the design and building construction domain, collaboration among design team members is limited due to the nature of current design execution which is linear (Shiratuddin & Thabet, 2007; Spillinger, 2000; Fu & East, 1999; East et al., 1995). The final design of a construction project is complete when each team member (architects and engineers) completes their specific task and component to furnish the facility’s design. Most design tools being used currently to create and communicate these designs are restricted to two-dimensional (2D) representations (Dunston et al., 2003; Shiratuddin & Thabet, 2003; Emmitt, 2002). We theorize that a CVE allows designers and engineers to collaborate concurrently rather than linearly, and at the same time provide the ability to create design in three-dimension (3D) within the VE.

CVEs thus provide an opportunity for architectural students in particular and architectural designers in general to work interactively in a virtual workplace and collaborate directly with one another on a design project. According to Arias et al. (2000) and Fischer (2005), collaboration provides the opportunity for collaborating design partners to socialize, learn to work with each other, and discover what support one can acquire from the other. Additionally, Martin and Sommerville (2004) emphasized that good rapport achieved by collaborating often led to future client referrals in both directions. The benefits of collaboration as suggested by previous works (Fischer, 2005; Martin & Sommerville, 2004; Kubo et al., 2002; Arias et al., 2000; Hill, 1982) can be achieved not only by architectural students, but ultimately the construction design domain through the implementation and use of CVEs.

With many individuals being able to work on the same project embedded in the same VE at the same time, conventional design in a linear fashion can be avoided, thus time constraints or ‘bottlenecks’ in the design development process can be removed. Within a CVE, review, alteration, and communication of a project’s design can be executed in real-time. This is in contrast to the more “traditional” collaborative styles that require long turnaround times or awkward over-the-shoulder collaboration.

We developed a software application, the Collaborative World Design Tool (CWDT) to test the benefits of CVE for architectural design. The CWDT was developed based on the Torque 3D Game Engine. The appearance and some of the functionalities of the CWDT are very game like. The CWDT allows for real-time collaboration and interaction among multiple users across the network. The CWDT is customized to support architectural design activities in a CVE. An automated data logging system was also developed to record activities during collaborative design sessions. We conducted an experiment with architectural students as the expert subjects and the general population students as the novice subjects to determine the effects of working collaboratively within a CVE, as opposed to working individually in a standalone VE.

Structure of the CWDT

In this section, we describe the utilization of GarageGames’ Torque 3D Game Engine to develop the CWDT software. In achieving the goals of the CWDT being a collaborative design and learning tool, two main features are incorporated which include real-time 3D object manipulation and visualization, and real-time collaboration in a CVE across the network.

The Torque 3D Game Engine was originally developed by a computer game development company called Dynamix, who have created computer games titles such as Earthsiege, Starsiege, Tribe and Tribes 2 (Maurina, 2006; Finney, 2007). The founders of Dynamix then created GarageGames and currently distribute the Torque 3D Game
Engine as one of the products for independent (indie) game developers. The low-cost licensing attracts many independent game developers to use the Torque 3D Game Engine to create and sell games. GarageGames’ business model is somewhat different from many other 3D game engine developers. Not only the licensing cost is affordable, the license also includes the entire source code of the engine. From our experience using other 3D game engines such as Epic Games’ Unreal engine and Valve’s Half-Life engine, we find that without access to the entire engine’s source code, further modifications of the engine to support real-world applications such as the CWDT is impossible (Shiratuddin & Thabet, 2003a). In the development of the CWDT, only suitable and usable components of the Torque 3D Game Engine were used and incorporated because not all components in the Torque 3D Game Engine were relevant. Figure 1 shows the components of the Torque 3D Game Engine that were used in the CWDT.

Three main components of the Torque 3D Game Engine are:

1. The User interface (UI) – this component consists of several sub-components i.e. the Console Terminal, Dedicated Server, the World Editor, the GUI Editor and the Play Interface. The UI component comprised of both text and graphical-based UIs. The UI allows for interaction between the application and the end-user.

   The Console Terminal sub-component uses only text-based interaction. The console terminal can be of two types: (1) a text chat window and (2) the developer’s debugging terminal. While each type represents different functionality, both types use text-based interaction with the end-user.

   The Dedicated Server sub-component allows for setting up and launching a dedicated standalone server using the Console Terminal. A dedicated server is usually setup to wait and listen, and link any incoming and outgoing connections from external clients. The server also synchronizes all the data between the clients and World/Mission Manager.

   The next UI sub-component is the WorldEditor interface. The WorldEditor is where the visual scene assembling in the VE occurred. The WorldEditor employs a more graphical approach to many 3D object manipulation functionalities such as moving, deleting, rotating, scaling etc. Besides 3D object manipulation, the WorldEditor is also linked to the GUI Editor. The GUI Editor also employs a visual approach to GUI design. It allows for the creation and modification of UIs for the CWDT application such as adding new menu items, right-click menu, the login screen etc.

   The Play Interface sub-component allows for real-time viewing of the VE. In the Play Interface mode only real-time fly or walk-through of the VE is allowed. No object manipulation exists in the Play Interface mode. This mode is suitable for users who would like to view only the 3D model with no intention of making changes to the design.

2. The Engine – this is the main component or the heart of the Torque 3D Game Engine. The Engine component consists of various managers. Each manager is responsible to handle, retrieve and process specific type of data. At the center of the Engine is the World Manager. The World Manager manages all the interaction amongst managers before passing the final required data to the end user.

   Table 1 shows descriptions of the various managers that exist in the Engine component.

3. Data – various data is stored and retrieved in different data elements. The required data for the CWDT application are various effects data, object data (textual and 3D object), world mission data, terrain data and sound data. E.g. Various Effects Data stores and
retrieves effects such as glow, display of bounding boxes, lightings, shadows etc. Table 2 describes the various types of data stored in the data component.

Another important element of the CWDT application is scripts. The 3D engine provides the code for graphics rendering, texturing, etc, and these capabilities are tied together by scripts.

Scripts are often used to bring different parts of the engine together, to provide a fully functional CWDT application.

**Characteristics of the CWDT**

We based the development of the CWDT on the work by Quinn (2005), Aziz et al. (2006), Arango et al. (2007) and Arango et al. (2008). Quinn
3D Collaborative Virtual Environment to Support Collaborative Design

Table 1. Description of the engine’s managers

<table>
<thead>
<tr>
<th>Manager</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect Manager</td>
<td>Manages visual effects such as displaying the skybox, textures, lightings, highlights around an object when selected etc.</td>
</tr>
<tr>
<td>Script Manager</td>
<td>Manages the execution of any scripts required by other managers and then passes it to the World Manager and Console Terminal</td>
</tr>
<tr>
<td>Object Manager</td>
<td>Manages all the objects in the VE which includes 3D objects, 2D GUI, avatars etc. Also manages properties such as texture mapping, geometry size, GUI size and color etc. Note that any items present in the VE are treated as objects.</td>
</tr>
<tr>
<td>World Manager</td>
<td>Tracks the location of each object present in the VE. All data must go through the World Manager and then be redistributed accordingly to other managers or to the end-user.</td>
</tr>
<tr>
<td>Terrain Manager</td>
<td>Manages the rendering of terrains. The terrain engine is part of the terrain manager and it is a sub-set of the main engine.</td>
</tr>
<tr>
<td>Network/Message Manager</td>
<td>Manages any data that requires updating across the network. Whenever real-time collaboration occurs, any changes made by any reviewers will be relayed to the hosting server and then retransmitted to other clients in the network.</td>
</tr>
<tr>
<td>Sound Manager</td>
<td>Manages all sound related data such as the sound effect when an avatar is walking in the VE.</td>
</tr>
</tbody>
</table>

(2005) suggests that computer games can be an effective medium to accomplish learning because of the game playing characteristics; contextual, goal-oriented, challenging, anchored, exploratory, attention drawing, and feedback providing. The game playing setting allows students to experience some form of constructivist learning experiences whether students work alone or in small groups. Rather than being passive recipients, students have the opportunity to explore, investigate and solve problems, and become actively engaged in the activities in the CWDT. Following Quinn (2005) and the work by Aziz et al. (2006), Arango et al. (2007), and Arango et al. (2008), the game playing characteristics included in the CWDT are discussed below:

Contextualized: Whenever actions are performed by students in the CWDT, the actions should support and make sense to them. In architectural design where 3D CAD software is primarily used to create 3D representations of buildings in the form of 3D computer models, actions such as selecting objects to either move, rotate and scale them are considered as basic required skills. Similar concepts are used in the CWDT so that students can relate to their current 3D modeling experience. The only difference is that in the CWDT students are able to “fly” freely in three-dimensional space and view from any

Table 2. Description of the types of data

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effects data</td>
<td>Stores visual effects related data such as the skybox, textures, lightings, highlight color around an object when selected etc.</td>
</tr>
<tr>
<td>Object data</td>
<td>Stores data of all the objects in the VE that include 3D objects, 2D GUI, avatars etc., and their properties such as texture mapping, geometry size, GUI size and color etc.</td>
</tr>
<tr>
<td>World data</td>
<td>Stores the world files so they can be retrieved later.</td>
</tr>
<tr>
<td>Terrain data</td>
<td>Stores terrain data such as height field elevation map, the terrain geometry, and terrain textures.</td>
</tr>
<tr>
<td>Sound data</td>
<td>Stores sound data that may include sound effects such as footsteps, weapon fire, weapon damage, background music, etc.</td>
</tr>
</tbody>
</table>
perspective without being restricted to any two-dimensional planes or viewports. Different types of 'gizmos' were designed and used to support and give meaningful gesture for different actions such as moving, rotating and scaling 3D objects (Figure 2). Since the actions also support free movements in three-dimensional space, different colors were used to represent the X, Y and Z axes.

Goal-oriented: In general, the CWDT can be used to build a fully functional 3D VE. For the purpose of this study, the goal of the CWDT is to teach architectural students to organize their thought process in design related activities in a CVE. In the study reported here students will come up with a plan on how to collaborate and build a 3D Japanese restaurant in a CVE, in real-time. A modular-based design of a Japanese restaurant provides all the essential elements of architectural design learning experiences such as how to navigate and manipulate objects in 3D space, how to coordinate the design efforts among designers, how to assist a design partner to make the right decision in terms of utilizing the correct building components and placing them in the intended space, how not to conflict with a design partner and organize the agreed 3D work space, etc. Students were informed of these learning goals early on and prior to using the CWDT.

Challenging: Some of our students are used to creating highly realistic 3D CAD models which are then used for rendering static images or generation of pre-defined path video walkthrough. However, these types of 3D CAD models are highly unsuitable for real-time rendering in 3D game engine. Therefore, students had to be taught on the limitations of real-time 3D engines in general, and how to employ some of the tricks used in the computer and video game industry. It should be understood that even when a computer with high processing power is used, there is always a limit on how much a 3D game engine can process and render all the polygons with real-time dynamic lightings at any given time. In the computer and game development world, a lot of “trickery” or techniques are employed, such as pre-baked lighting and shadows, normal map textures mapped onto low resolution 3D model to make it look more realistic. These techniques are employed to trick the audience into believing that the VE is realistic. Students at first felt challenged to learn these new concepts. However, once they understood the principles and limitations of real-time 3D engine, and were able to establish a workflow from conceptual design to 3D CAD to real-time 3D game engine, they excelled and enjoyed creating virtual worlds.

Students were then trained to use the CWDT for the sole purpose of performing design collaboratively and learning from each other in a CVE setting. We observed that students who were not used to playing computer or video games took some time to get used to navigating using the keyboard and mouse. The WSAD keys on the keyboard are the acceptable standard keys for moving the avatar forward, backward, left and right respectively in many modern first-person-shooter games, and the mouse for looking up and down. We demonstrated these navigation techniques and allowed students time to adapt to it. Once students were comfortable with navigation, the following
task the students had to learn was to manipulate 3D objects in the CVE. Non-gamer students face more challenges thus were allowed time to adapt to the manipulation techniques.

During the experiment, students were not allowed to physically speak to one another, except for text chatting using the built-in text chat feature of CWDT. The reason we disallowed them to speak was to simulate the idea of a collaborative and remote learning environment. However, students did not use the text chatting feature because of the cumbersomeness for them to switch between using the keyboard and mouse for navigation and manipulation and text chat. We observed that students who worked in pairs managed to coordinate the design process between each other without being in each other’s way. Since CWDT supports avatar representation of each student, students were able to see from a first-person-view what their partner was doing in the CVE. For example, any 3D objects being moved by a student can be seen moving in real-time by the partner. Since text chat was found to be an unproductive feature for the CWDT, in upcoming experiments, we plan to include Voice-Over-IP (VOIP) to allow students to communicate verbally using headset with built-in microphone. During the experiment, the design tasks assigned to our students were somewhat challenging especially in terms of getting the correct perspective views and performing manipulation on the selected 3D objects, and making sure the same actions were not performed twice. Majority of our students were able to complete the given tasks in approximately 30-45 minutes.

 Anchored: The goal of the CWDT is for students to collaborate in performing design tasks in a CVE, and also learn from each other. All the tasks and related actions that must be performed by students in the CWDT are part of their learning activities. By utilizing features in the CWDT and by collaborating in the CVE, students were able to achieve the final goal which was to fully assemble a Japanese restaurant with their partner.

Our students were taught the benefits of working in a group, and how to achieve that using the CWDT. The steps we described to our students prior to taking the experiment made sense, hence allowing them to grasp the concepts and fully appreciate the benefits of working together.

 Exploratory: In the CVE, students are free to decide where and when to start assembling the restaurant, and which building component to execute first. Students also have the freedom to collaborate with their design partner and decide the level of shared responsibilities with respect to accomplishing the assigned tasks. We designed the CWDT to be flexible and non-intrusive, and gave students the freedom to explore the CVE. Students were reminded that there were no absolute steps in accomplishing the end result; “Do what you think is logical and what you think is right in the least amount of time”. The CWDT was developed to monitor and record all the activities performed by students. Some of the data recorded and collected are presented in this chapter, and others will be studied in future research undertakings.

 Attention-drawing: The CWDT was developed using the Torque 3D Game Engine from GarageGames. Compared to other traditional VE development tool, the Torque 3D Game Engine (1) has the capability to render high quality life-like textures and realistic lighting, (2) allows for customization through the availability of its source code, (3) able to accept 3D CAD models generated using industry standard 3D modeling software, and (4) supports multi-users across the network. With the abovementioned built-in features, we were able to develop the CWDT and the VE that can capture the attention of our students. Since the CWDT uses a 3D real-time game engine, our students receive instant feedback on all the actions they performed while they are creating the Japanese restaurant. They no longer have to wait for an image to render for a few minutes on the screen.
The Torque 3D Game Engine, by default, does not support real-time object manipulation across the network. Therefore, to support this important architectural design functionality, extensive and customized code changes were made to the base source code. Unlike typical 3D engines or 3D rendering engines, many next generation 3D game engines such as the Torque 3D Game Engine comes with a handful of built-in tools such as a World Editor that can accept 3D CAD models, a terrain engine, a physics engine, audio and video support, support for multi-user across the network, support for pre-defined animations, a material or shader editor, a GUI editor, a script editor etc.

Feedback providing: In the CWDT, automated feedback system was not developed. However, visual cues were used to ensure that students are aware of the types of manipulation that they are performing on the selected 3D objects in the CVE. During the experiment, only the human facilitator provided verbal feedback to the students on issues such as whether the design is correct or not, whether they are taking too much time to complete a given task, or whether they are performing the tasks correctly. From this experiment, we noted that answers to queries from students while working within the CVE could have been provided in the form of automated feedback. In future, we intend to develop some form of an automated feedback response system in the CWDT.

The Experiment

The tasks in our experiment were to construct a building in a customized CVE. The 3D model of the building used in the experiment was modeled after a Japanese restaurant (Figure 3). Japanese architecture is often modular in design with repetitive segments. Aside from the roofing and stairs, we divided the building into 4’ and 8’ sections. These basic sections are shown in Figure 4. We repeated each small section to produce the complete flooring and walls. To further simplify the experiment for the novices, we grouped these smaller sections into even larger sections that comprised of six to ten smaller segments. Completion of the final 3D model of the Japanese restaurant required subjects to copy nine floor segments, move eight wall segments and three stair segments into place.

During the experiment, the automated data-mining system monitored what each subject was doing. It logged data such as: how long the overall experiment took; the number of objects moved, rotated, scaled, created, deleted, and copied. It

Figure 3. The 3D model of the Japanese restaurant used in the experiment
also logged the exact amount of time a subject spent moving, and handling objects. The exact location and orientation of each user’s avatar was recorded at a rate of fifty times per second (50 Hz).

A total of 37 students participated in the experiment, randomly assigned to either work in pairs or individually. Twenty one novice students were recruited from the Psychology department’s research subject pool (14 of them formed seven pairs, the remaining seven worked alone). 16 architecture students were recruited as the expert group (10 of them formed five pairs, the remaining six worked alone).

A 2×2 between-subjects factorial design was used with Expertise (novice, expert) and Group (single, pair) as the independent variables. There were six dependent measures that were analyzed: completion time, total duration of objects in movement, number of times objects were moved, total path length in 3D space, average speed of movement, and movement frequency (defined as the number of moves divided by the total duration of objects in movement). To be able to compare the performance of pairs with singles working alone, the dependent measures of one randomly selected member of each pair were pitted against the measurements taken from singles working alone.

The software interface seen by each member of a pair is shown on Figure 5. The computer screen is split between the two subjects (labeled as Client and Host in the software) who can monitor their partner’s activities throughout the duration of the experiment. Subjects who were assigned to work alone only had access to a single viewpoint representing their own activities.

We hypothesized that working in pairs will be beneficial for the design task, and that experts will perform better than novices.
Experiment Setup and Procedures

The experimental setup is shown in a diagram screenshot of the building from a top-down viewpoint (Figure 6). The coordinates of the main corners of the building are noted in the diagram in meters. All subjects’ avatar started the experiment from the same spatial location, on the left side of the building construction site. The coordinate system’s orientation is indicated at the bottom right of the figure (positive Z direction is up, not visible from the top-down view of the diagram). The avatar had six degrees of freedom of movement (three spatial dimensions, and three orientations – yaw, pitch, and roll). Subjects used the keyboard and mouse to move around and rotate the viewpoint as they wished. A combination of computer mouse and hotkeys were also used to select, drag, and drop building blocks into the desired locations. When working in pairs, no physical verbal communication was allowed. In fact, some of the subjects did not know their partner well. The avatar of their partner was visible. Each time a building component was moved and placed, the action was visible and shown in real time. This allowed subjects to see what the other person is doing at all times. A brief training session preceded the start of the experiment to let subjects become familiar with the CWDT interface and learn how to operate in the CVE, and how to manipulate 3D objects.

Results

A 2×2 between-subjects analysis of variance (ANOVA) was used to analyze the results with Expertise and Group as the independent variables. One of the basic measures of successful collaboration is time management, that is, how much time is needed for effective construction design to be completed, and how expertise and level of collaboration influence times savings. In order to test time management, completion time was measured, defined as the total time elapsed from the moment a person or a group log in and start the project up until they complete the task. We hypothesized that experts should finish the
project in less time than novices, and that singles 
would take up more time, because they had to do 
more work than the single members of any pair. 
The average results are presented in Figure 7. 
The only significant result was the main effect 
of Group, $F(1,21) = 4.62, p < .044$, suggesting 
that singles working alone took longer time ($M = 15.4$ minutes, $SD = 6.16$ minutes) to complete 
the task than pairs ($M = 10.6$ minutes, $SD = 4.29$ minutes). There was no difference between experts 
and novices, nor an interaction between Expertise 
and Group. Our hypothesis was only partially 
confirmed by the results.

In order to map out the spatial character of a 
subject’s performance we tracked the movement 
of the subject’s avatar in real time through the 3D 
virtual space of the design environment. Total 
path length was calculated as the sum of all dis-
placements as the person’s avatar crisscrossed 
through the CVE. Three dimensional coordinates 
were recorded at a sampling rate of 50Hz. As 
noted earlier, each pair’s total path length was 
represented by randomly selecting the measure-
ment of one member of that particular pair. We 
hypothesized that members of pairs will natu-
really cover less territory, and that experts will 
traverse through fewer places than novices. The 
average results are presented in Figure 8. Contrary 
to our hypothesis, results revealed that experts in 
fact traversed a significantly longer path ($M = 1.99$ km, $SD = 0.91$ km) than novices ($M = 0.89$ 
km, $SD = 0.65$ km), $F(1,21) = 11.8, p < .003$. No 
difference was observed between pairs and singles, 
and no significant Group × Expertise interaction 
was present.

In order to assess the speed at which subjects 
performed the task we calculated average speed 
as the ratio between total path length and comple-
tion time. We hypothesized that novices would 
be slower than experts, and that singles would be 
 faster than pairs. The average results are depicted 
in Figure 9. Experts were significantly faster ($M = 3.25$ m/s, $SD = 1.86$ m/s) than novices ($M = 1.08$ m/s, $SD = 0.64$ m/s), $F(1,21) = 21.8, p < .001$. There was a marginally significant main 
effect of Group, $F(1,21) = 3.90, p < .06$, indicat-
ing that singles were on average slower ($M = 1.68$ m/s, $SD = 0.89$ m/s) than pairs ($M = 2.42$ m/s, $SD = 2.26$ m/s). In fact, a marginally significant Ex-
pertise × Group interaction, $F(1,21) = 3.69, p < .07$, has revealed that experts are fastest when 
working in pairs, and that novice pairs and novice 
singles moved at almost identical average speeds 
(around 1 m/s).
How subjects handle building blocks in the CVE may reveal how efficient their actions are. We measured the total time subjects spent moving objects. Our hypothesis was similar to the previous one, in that we predicted that experts and pairs should spend less time manipulating objects. In addition, we predicted that expertise should impact singles more than pairs. The average results are presented in Figure 10. There was a main effect of Expertise, $F(1,21) = 5.91, p < .01$, suggesting that novices spent more time moving objects compared to experts. A main effect of Group, $F(1,21) = 8.05, p < .01$, revealed that singles spent more time moving objects than pairs. The significant Expertise $\times$ Group interaction, $F(1,21) = 4.70, p < .01$, qualified these results in the following manner: Novices who worked alone moved, handled and manipulated objects for the longest
time by far. In addition, being a novice or an expert did not make a difference for move time when subjects worked in pairs.

Another measure related to object handling, namely the total number of times objects were set into motion, was also tallied and analyzed. Our hypothesis stated that novices who work alone would be forced to move objects many times, whereas experts who work in pairs could afford to move objects significantly fewer times. The only significant result obtained was a main effect of Group, $F(1,21) = 4.33$, $p < .05$, revealing that
singles indeed moved objects more times than pairs. There was no main effect of Expertise, and no Group × Expertise interaction. The average results are presented in Figure 11.

The next measure was a combination of the latter two. Movement frequency ($MF$) was defined as the ratio between number of moves and move time ($MF = \text{moves/move time}$). This quantity had the potential to tease apart certain design strategies. For instance, handling a lot of objects in a short period of time would result in large movement frequency, whereas handling a few objects in the same (short) amount of time would be described with a low movement frequency. We hypothesized that the most efficient way to perform the design task would be indicated by a high movement frequency, most likely exhibited by experts. We did not have any prediction about how singles versus pairs would fare in terms of movement frequency. The average results are presented in Figure 12. There was a Group × Expertise interaction, $F(1,21) = 4.86, p < .04$, suggesting that expert singles moved more objects per unit of time than novice singles. No difference between expert pairs and novice pairs was observed. There were no main effects of Group or Expertise. Expert singles had the highest movement frequency, as predicted by our hypothesis.

**Discussion**

We sought to investigate the influence of (1) level of expertise and (2) the presence or absence of opportunity to collaborate on performance in a virtual architectural design task. The technology employed was a CVE that allowed for real time interactions among design partners. Performance was assessed with several kinematic and enumerative measures. We hypothesized that subjects would abide by the principles of economy of action, where they would converge towards efficient time management and at the same time minimize trial and error strategies. This simply meant doing less, in shorter time, using fewer steps in the entire design process. We predicted that experts would be more likely to exhibit such behavior. Our measures were set up to investigate what factors contribute to successful real time collaboration.

We found that experts explore the workspace more extensively, but spend about the same time
as novices. As a consequence, experts were found to work faster. Working in pairs was faster than working alone, because pairs explored the same amount of space (as measured by total path length) as singles in less time. The benefits of collaboration were facilitated by level of expertise in that experts who worked in pairs were the fastest designers. Interestingly, the advantages of collaboration were manifested mostly in temporal savings, but not so much in how much of the workspace was explored: overall singles explored the same amount of space as pairs. Future experimental work is needed to investigate how spatial and temporal aspects of performance are influenced by level of collaboration and expertise. One possible way to investigate this would be to impose either explicit time constraints (mimicking deadlines from everyday experience) or explicit spatial exploration limits (mimicking limitations of viewpoint and software computing power) in order to increase task demands.

The economy of action was also reflected in the enumerative measures (move time, number of moves, and movement frequency). Overall, pairs and experts spent less time manipulating objects. This suggests efficient use of software tools, knowledge of how to handle objects, etc. All of this cut down on the extra time taken to explore and try out the objects and figure out how they can be handled. Interestingly, experts’ performance was described as having a higher frequency of movement, but only when working alone. This may be consistent with the principles of economy of action and reflect the benefits of collaboration in the following manner: Pairs may afford to do more in less time as singles, because they can divide task responsibilities among themselves.

Overall, we observed interesting tradeoffs between various behavioral aspects of design activities (such as handling objects and time management), but also compounded benefits of expertise and team work.

**CONCLUSION**

A CVE setting is relevant to the principles of social constructivism, and social context of learning. Unlike traditional tools for education, a CVE supports the social side of learning. Within a CVE, students have the opportunity to discover the content of what is being learned and create meaningful connections with the content through creativity and interaction. By working collaboratively with other students, they become active learners and learn to work together toward a common goal. Learning content, tasks and problems presented within the CVE learning space encourage students to think, explore, discover, and manipulate the content to become better problem solvers and at the same time, learn and gain knowledge. The software we developed for this study (CWDT) includes game playing characteristics that allow students, whether working individually or in small groups, to experience a constructivist learning where, rather than being passive recipients, students explore, investigate and solve problems, and become actively engaged in the activities in the CVE.

The results and observations obtained from the experiment suggest the benefits of working collaboratively within a CVE outweighed working individually in a standalone VE. Our findings indicate that collaboration within a CVE has great potential to increase the productivity at which designs are assembled, reduce the number of errors in design, provide a constructive learning environment, reduce the overall stress levels, and increase positive thinking and a group mindset for subjects.

Future work on collaboration in virtual environments should look at the differential benefits of various technologies, such as software solutions for CVE, and also the ergonomic aspects of human-computer interactions by utilizing haptic, visual and auditory aspects of interactions in the CVE. Monitoring performance over an extended period of time would provide insight about the development of expertise. This can have implications for
designing novel educational tools such as interactive real-time CVEs for online courses tailored to the demographics of the students (from pre-K to college and beyond). On the psychological side, future work is needed to explore the influence of demographics such as age and gender on collaboration. Social components of group behavior, such as leadership roles, division of labor, and group management are also important factors that need to be submitted to rigorous empirical investigation.

ACKNOWLEDGMENT

We would like to thank Mr. Jason Breland for his assistance with the experiments. We would also like to thank the University of Southern Mississippi for providing us the support. Lastly we would like to thank all the students who were involved in this study.

REFERENCES


**ADDITIONAL READING**


3D Collaborative Virtual Environment to Support Collaborative Design


KEY TERMS AND DEFINITIONS

3D Game Engine: All computer games have some form of engines working in the background. The engines can support 2D, 3D rendition of pixels (picture elements) or both 2D and 3D on the screen. There may be more than two engines working together and these engines are the backbone of the games. Computer games that include 3D environments use 3D Game Engines to produce and display images in real-time on the display device.

Architectural Design: In this chapter, “architectural design” is used in the engineering and technical context. Architectural designs are prepared by the designer/architect/engineer and depend on the stages of design; programming, schematic, preliminary and working drawings.

Collaborative: To collaborate is to work together in scientific undertaking. Collaborative software, also known as groupware, is an application software that integrates work on a single project by several concurrent users at separated workstations (see also Computer Supported Cooperative Work or CSCW).

Constructivism: Constructivism is founded on the idea that by reflecting on one’s experiences, one constructs their own understanding of the world. One generates his/her own “rules” and “mental models”, which is used to make sense of his/her experiences. Individual learning process involves adjusting the mental models to accommodate new experiences.

Virtual Environment: VE has been variably defined by academic circles and industries. To some people, VE is a specific collection of technologies, while others stretch the term to any medium that can present an environment that draws the receiver into its world, which includes conventional books, motion pictures, radio, and so on (Isdale, 1998). Lok (2004) however sees VE as immersive VE that is made of systems that allow participants to experience interactive computer generated worlds from a first-person perspective, as opposed to pre-rendered movies, videos, or animations.