From Dungeons to Classrooms: the evolution of MUDs as learning environments

Brian M. Slator[1], Otto Borchert[1], Lisa Brandt[2], Harold Chaput[3], Kellie Erickson[1], Gabriel Groesbeck[1], Jacob Halvorson[1], Justin Hawley[1], Guy Hokanson[1], Dan Reetz[4], Brad Vender[1]
[1]North Dakota State University, Fargo ND (USA) 58105
{brian.slator, otto.borchert, kellie.erickson, gabriel.groesbeck, jacob.halvorson, justin.hawley, guy.hokanson, bradley.vender@ndsu.edu
[2]anthro@lisabrandtphd.com
[3]hchaput@ochashiyo.com
[4]danreetz@gmail.com

Abstract. The history of MUDs (multi-user domains or dungeons) goes back to the 1970s. Primarily this is a history of role-playing games and text-based virtual realities. As the decades have passed, MUDs increasingly have been developed and deployed for a wide range of applications in education and the Science of Learning. As opposed to an historical recounting of the development of MUDs, this chapter describes the evolution of MUDs as they have influenced education and learning, from simple meeting and discussion places, to simulated learning spaces, to the current state of the art, which is Immersive Virtual Environments (IVEs) for Education.

1. Introduction

MUDs (multi-user domains or dungeons) have been with us since the 1970s. They are part of the fabric of internet life, and have been more efficacious in building and sustaining social communities than many technologies we have seen come and go.

As will be explained in detail below, MUDs are multi-user places where people meet in virtual reality. Over the years a large number of MUD variations have emerged, under different names like MUSH, MUVE, and MOO, but the basic idea is the same – these are virtual places where people log on, take on a virtual persona, sometimes referred to as an avatar, and live a virtual life as a character in a virtual world. The possibilities for the characters are seemingly infinite: Butch McManly can log on as a 7 year-old girl, Poison Ivy can log on as Mr. Freeze, and regular mortals like us can log on to partake in an adventure or a learning experience that would never be available to us in the normal course of our lives.

This is the strange power of MUDs. You can visit impossible spaces, experience things not possible in real life (IRL), and be someone you could only dream of being IRL.

The MUD phenomena, and the history of MUDs, has been documented many times over the years. We review a bit of that here. But our purpose is to trace the history and evolution of MUDs as they have been used for education within a science of learning context, which is a somewhat different story.

Over the years, educators have employed MUDs to create a number of educational opportunities: meeting and discussion spaces for students, simulations for immersion in language learning, simulations for experiencing historical reconstructions, gathering spots for professionals to share ideas and teaching strategies, and in the current culmination of this evolution, immersive virtual environments (IVEs) for education.

This chapter traces the evolution of these systems from the distant past to the current state of the art as they have participated in education and the science of learning.

1.1 Technical Notes

There are some things to keep in mind while reading this chapter. First, MUDs are client-server systems, meaning the virtual environments are hosted on a single machine that many players access at the same time. Imagine the virtual environment as a wheel. The people are all out on the wheel rim while the simulation resides in the hub of the wheel. Client software is used to connect the people to the simulation. The client programs were text-based in the early days as, indeed, all computing was.

In those olden days (before, say, 1983), there was no such thing as a mouse or a click. Programs were launched from the 'command line', produced text for users to read, and responded to commands typed into a text window. In the very early days, if you made a mistake in your typing, there was no backspace button to fix your command. Instead, you waited for an error message and then retyped your command again. Imagine, if you can, that everything done on a computer was accomplished through something resembling primitive text-only chat messaging. Imagine too, that programming in these simulated worlds was accomplished in the same way, without even rudimentary text editing such as you see in Notepad or Microsoft Word. In those days, you wrote and edited code using a 'line editor' which let you list code, and replace lines that were stored in memory.

These 'stone aged' systems still were in place when the MUD phenomenon took off in the early 1980s, and this explosion would never have occurred if not for pioneering efforts by people developing and distributing tools that others could use to build their MUDs. These advancements, and all the others, are described below.

2. Medicated Goo: the 'birth' of MUD

MUDs have their roots in interactive fiction programs written for mainframe computers in the mid-1970s. These interactive fiction programs, or “text adventures,” allowed the user to explore and manipulate a virtual world entirely through text commands. Users could perform movements or actions by typing commands such as “go west” or “open box,” and the results of their actions would be described for them by the MUD program. The first text adventures were “Colossal Cave Adventure” by Crowther & Woods in 1976 and “Dungeon” by Anderson, Blank, Daniels and Lebling in 1977 [1], which was later sold commercially under the title Zork.

Text adventures were single-player games, that is, people who played a text adventure had the virtual universe to themselves. It was this limitation that Roy Trubshaw and Richard Bartle worked to remove in 1978-79 when they created their own version of Dungeon for multiple users, called Multi-User Dungeon, or MUD [2]. Working in England at Essex University, Trubshaw and Bartle used a programming language known as MACRO-10. This version consisted of little more than the text-based ability to move around to different “rooms”, and to chat with other users on the system in real-time. Bartle wrote in his article, “How It Really Happened,” however, that this was not actually called a MUD. The subsequent version, also written in MACRO-10, was a little more sophisticated, and was referred to as an actual MUD. This version had at its core an actual database of user commands, rooms, and interactive objects.

The technology advanced rapidly, and after Trubshaw left Essex University, Bartle began working on additional functionality for the users. Before Trubshaw left the university, the MUD system code had already become unmanageable. Trubshaw rewrote the entire system in BCPL, a programming language that was the precursor of the now-ubiquitous C programming language. Then Bartle started to add features into the program, such as a point-scoring system, more advanced communication commands among players, and privileged characters like wizards [3]. More features enabled even greater popularity of the system, and this new form of virtual interaction took off.

Trubshaw and Bartle based their multi-user functionality on a shared database. This database was accessed by people logged into the same DEC PDP-10 minicomputer. Soon thereafter, MUDs were created that could be accessed in several different ways: via dial-in using a modem, or from any computer connected on a network, and finally, via any computer connected via the Internet.

There were several different MUDs created over the next ten years, but MUDs remained not much more than a technological gaming curiosity until a fundamental design change was introduced with MONSTER by Skrenta in 1988 [4]. MONSTER’s invaluable contribution to MUD development was the ability to manipulate the environment while the game was still running. MONSTER became a benchmark in MUD evolution. Now players could “dig” new rooms, connect them with newly created “exits,” and create a variety of player-defined objects, all while the game continued. Once new rooms and objects were created, they instantly were available to anybody connected to the game. The only major drawback to the original MONSTER was that it was written in the less popular Pascal language on the more obscure VAX operating system. However, in 1989, MONSTER was rewritten in C for Unix by James Aspnes, a graduate student at Cernegie Mellon University. He called his work TinyMUD [3]. The TinyMUD variant of the MUD system became popular due to its simple game play, and also the fact that it was completely free to use.

TinyMUD was only up and running for seven months until it had to be brought down for overtaxing the host computer, but in that time the MUD-slide of the 1990s was born. TinyMUD gave inspiration to dozens of new MUD implementations, including TinyMush, PennMUSH, TinyMUCK and TinyMUSE, which were all based in some way on Aspnes' original TinyMUD source code. Other MUDs, such as UberMUD, TeenyMUD and MOO [5] were written by people who played TinyMUD, but were not based on TinyMUD's source code. (See [4] for an overview.) Each of these expanded on TinyMUD's feature set, adding better networking abilities, better security, or expanded customization abilities.

3. From Text to Graphics: Habitat

All of the MUDs described above are entirely text-based games. These games catered to programmers and early computer users, who preferred simple text-based interfaces over graphical interfaces. In 1985, Randy Farmer and Chip Morningstar set about creating a MUD that a broader gaming audience might enjoy. Working for Lucasfilm, Morningstar and Farmer [6] created Habitat, the first graphical MUD. Users connected to Habitat using a client program on the Commodore 64 personal computer. Habitat presented the MUD world to the user in a two-dimensional comic book style, with cartoon representations of users, a bright color palette, and user-entered chat messages presented in speech balloons. Today, graphical MUDs are known as Massive Multiplayer Online Games, and the subgenre of Massive Multiplayer Online Role-Playing Games (or MMORPGs) are a multi-billion dollar industry, including Blizzard Entertainment's "World of Warcraft" and Sony Online Entertainment's "Everquest" and "Star Wars: Galaxies."

4. Lahar in the 1990s: the explosion of MUDs, MOOs, MUSHes, MUVEs, and TINY things

Wikipedia (http://wikipedia.org/) defines *lahar* as "a type of mudflow… [that] is explosive in nature, moves quickly with little or no warning, and can have a long-lasting effect on people and communities." The explosion of textual-based virtual realities in the 1990s was like that – a lahar of MUDs.

In the beginning of the 1990s, personal computers were becoming more affordable to the middle-class, and, subsequently, more widespread. In 1990, Apple introduced the Classic, LC and IISI.. In 1992, Microsoft introduced Windows 3.1, selling 10 million units [7]. The fastest computers of those days had only tens or hundreds of megabytes of hard drive storage, four to sixteen megabytes of RAM, and not more than a 60 Mhz processor. These machines were not the graphical powerhouses that exist today, however, they sold very well and many users were tapping into this earlier and nascent version of the Internet to see what was out there.

At that time, the World-Wide Web had existed for less than a decade, and there was no widespread access to the Web with a "point and click" browser interface, such as the user interface to which we are accustomed today. To go online, one needed to connect to a specific server and have a somewhat advanced understanding of the way in which the Web was organized. The specific server to which the user was connected (typically via ftp, telnet, or Gopher) generally required a paid subscription service such as the original Compuserve (later purchased by AOL), and the user could not easily browse. Users needed to have an idea towards doing something specific once connected. From the subscription service's business point-of-view, the problem was that people, especially those outside of academia, needed something interesting to do on these servers.

MUDs were one way to keep people coming back.

Once connected to a MUD, a person entered a completely alternate reality from that which they came. Although MUDs were text-based early on, the user experience was immersive nonetheless. By immersive, we mean that the person experienced being somewhere other than where they were actually physically located and someone else other than who they were IRL. After entering this alternate world, the person could become anyone they could dream up and they would become immersed in that character. This allowed a person to break from their daily routines and live out a separate life in a separate world. Early on, the majority of MUD participants were men and teenage boys, but that slowly changed as alternative worlds were created and an entire generation of children grew up in a world where MUDs are commonplace. Today, girls and women make up over half of the online gamer group, and the majority of these females participate in role-play games, including MUDs [8].

In addition to their immersive qualities, MUDs also became popular because users could socialize in ways they never would in real life. A MUD world, due to its player anonymity, "tends to lower social barriers and encourage players to be more outgoing than in real life" [9]. While exploring rooms and manipulating objects, characters encounter each other, often are very friendly toward one another, and often strike up conversations.

These types of interactions typically don't occur quite so easily and leisurely in a real world setting. This is one significant reason why users kept coming back to the MUDs. Moreover, the anonymity of the real person in these virtual worlds produces a feeling of personal safety, both physical and emotional, to the individual user, thus allowing people to express feelings or take actions that were not as accepted in the real world. Examples of such feelings and behaviors include creating characters who are homosexual or bi-sexual, or creating characters that give the MUD player the opportunity to engage in gender reversal [9].

Many MUDs had a gaming/rewards-based feature to them to encourage long-term activity within the game. In some worlds, players battled against each other with the intent to kill or destroy the other character. By doing so, they would be rewarded with points. Of course, this created a competitive atmosphere, which some people craved. Unlike real life, physical fitness wasn't a criterion for success in hand-to-hand combat, which meant more people had a real chance at success.

Not all MUDs were competitive nor filled with violence. The TinyMUD (noted above) rewarded its users with "pennies" if they visited other people's rooms in the world. TinyMUDs created a world where people would be rewarded by freely exploring without worrying about who might be lurking in the next room. This created a more social environment for players who either disliked or were bored with the violence of other MUDs, and it filled a niche for people who were interested in seeing what was through the next door.

For some users, MUDs became an environment in which an individual could gain considerable personal power (whether good or bad) that could not be obtained in the real world. For example, some players had the capability to "toad" other players (turn another player's character description into that of a wart-covered toad) or even remove other players completely from the server [10]. A wizard is the usual example of such a powerful character in a MUD. These power-wielding players obtain a sense of power they couldn't get from reality by being able to vent their frustrations openly. Most wizards were not "evil." Most enjoyed teaching and helping people through their adventures in the MUD world, and could even "promote" other players to be wizards like themselves.

Finally, most MUD software was free and available to anyone who was interested. A person simply could download the software and double-click on the executable control to start the client. Often, these clients even had a predefined list of servers to make a connection. This allowed many people who might otherwise not be interested (or computer savvy) to develop a "what have I got to lose?" attitude.

In summary, MUDs exploded onto the scene in the 1990s and spread "lahar-like" for many reasons. The ability to safely role-play, be social, explore new worlds, be rewarded, exert power, and play for little investment were all very powerful forces that MUDs provided to anyone interested in joining. If not for this MUD-flow, we would not see the vast array of graphical-based realities today.

5. Virtual Discussion, Virtual Campuses

The next phase in the evolution of MUDs as learning environments involves the development of multi-user interaction in an online environment as a given strategy to community building, especially the concept of the virtual discussion, and its natural academic counterpart, the virtual classroom. These two technologies, along with their derivative concepts, have allowed participants to be teleported great virtual distances.

Early attempts at generating a virtual meeting place provided the foundations for modern virtual classrooms. The original MUD platforms can be viewed as a fusion of two already popular social forums. These were the BBS, or bulletin board systems, and the role-playing adventure games that were gaining popularity at the time, the most famous of which is probably Dungeons & Dragons.

As we have discussed, these early forms of the now-popular online community gathering-place proved immensely popular. Users were able to interact with each other, as well as with the virtual world in which they were immersed. Remote participants were able to "meet" inside MUDs and go on adventures together or individually. The games provided goals and metrics by which a character could advance in skill and wealth, and was also used as a social forum for the users. This technology already was significantly ahead of the previous interactive champion, the telephone. Interacting in a virtual world, even in the early text-only versions of the technology, provided a much more fulfilling experience to the participants.

Despite their relatively advanced and modern features, the basic concepts behind the MMORPG games of today are the same as that of the original MUDs. The adventuring environments now consist of a fully 3-D graphical world, with both sound and text communication between users, and between a user and the system. These games are so enthralling, that a backlash movement has emerged against what some perceive as an addiction to online gaming. Countless news articles tell of players whom have forsaken family, friends, jobs and sometimes even their own lives to continue to play, as well as committed real-world crimes to gain advantage in role-playing games [11]. In actuality, of course, the vast majority of MMORPG players are just having a fun time gaming and socializing.

The more advanced versions of MUD technology in use today provide exceptional opportunities for people looking to create virtual learning environments. There are many examples of rather advanced online communities in use, including Donut MOO in Stark County, Ohio (a K-12 oriented learning environment), to the many various types used at universities all over the world. Examples of these include MundoHispano and MOOFrancais, both of which are hosted at the University of Missouri-St. Louis. These MOOs offer learning environments using the respective languages, Spanish and French. There are MOOville at the University of Florida, PennMOO at the University of Pennsylvania, and CafeMOO at the University of California, Berkley, as well as many others. These virtual learning environments supplement, and in some cases replace, the universal model of classroom-style teaching. Several of these environments are briefly discussed in the following section.

6. Writing Labs, Language Learning, Historical Reconstructions

The ability to create entire virtual universes that present an almost infinite range of functionalities and instructional capabilities is available to instructors today. Indeed, with a generation of students raised on video games and cell phones, speaking to them in their own language becomes of fundamental importance.

Several applications stand out as immediately amenable to the new ways of creating learning environments. Some of these include writing and composition instruction, learning a new language, and the creation of historical constructs in order to make the past appear "alive" to students studying history. These applications seem ready-made for virtual learning environments for many reasons. These include the natural displacement of location of the students and/or the instructor, the fact that many of these applications are very student-driven, such as writing, requiring only rough guidance from the instructor, as well as the problem of the subject matter no longer being available directly to the students, as in the case of the subject of history.

The early adopters of virtual teaching environments were business schools. As early as the 1950s, business colleges began requiring their students to compete in simulated business environments. This was accomplished by students feeding actual punch-cards into the school's mainframe computer. The students' own programs would then compete with, and be evaluated against, other students' programs within the virtual system. These primitive methods have, of course, advanced considerably since then. The environments today routinely incorporate a graphically interactive world, sound, text, and vastly increased functionality and interactive ability. Today's MUDs and MOOs are truly interactive, artificial virtual universes.

The University of Pennsylvania employs a system called PennMOO. This MOO is loosely based on the geography of the campus of the university. It provides a virtual world in which students and teachers can "meet" and conduct educational activities. Though it is used in general for all of the university, one particular course is well structured for this type of learning style: their English 88 course. It features online assignments, virtual seminars and virtual speeches, as well as an annual event known as the MOO Poetry Slam. This environment provides a technical emphasis to the introductory English course.

Another example of writing education being taught virtually is the CafeMOO environment developed at the University of California, Berkley. This artificial environment provides virtual office hours and other supplemental material. One of the most interesting uses, however, is that of the Humanities area of their English 108 course. Students have access to the same online facilities described above, however, the course itself emphasizes composition writing about the effects of technology on society. Required reading includes Microserfs by Douglas Coupland and WIRED magazine. Assignments are completed within the virtual world itself, which reinforces the purpose of the assignments.

The process of learning a new language is immediately ready for this type of enhanced learning. The best expertise is almost always from a foreign land (that of the language being taught), and it is difficult to disseminate this information without direct contact with these cultures. A couple of examples of online language educational environments include the MundoHispano environment and the MOOFrancais environment, both developed and maintained at the University of Missouri, St. Louis. MundoHispano is a language learning environment geared towards students whose native tongue is Spanish. Within this system there are thousands of “rooms” which represent over a dozen different countries. Each room is written by native speakers of the country it represents, thus acknowledging the many national and regional dialect differences within the Spanish speaking community. This gives participants the chance to interact with real native speakers and to communicate in terms of real-world conversation.

The MOOFrancais environment is similar, but on a smaller scale. There are over one hundred rooms, each representing a different part of the city of Paris. Users can stroll virtually down various famous streets and interact with other users in French. This provides an immersive experience similar to the experience of physically traveling to the foreign country.

One of the more exciting subjects to be taught within this new paradigm is history, which is normally taught through reading and writing of prepared material with limited opportunities for hands-on or field trip experience (though trips to a local museum are common). This manner of teaching can be expanded using virtual worlds to include immersion in a simulated historical context.

An example of a system for studying history can be found in the Virtual Archaeologist/Like-A-Fishhook 3D Reconstruction, which “…[combines] immersive role-based technology with the principles and data of archaeology, ethnography, and history, … for teaching generations of students about all of these.” [12]. Participants in this environment must interact with the people and geology of the time period through the virtual 3D game-like interface.

7. LambdaMOO, MacMOOse, MOOse Crossing, High Wired and enCore

MUDs were extremely popular, but lacked flexibility for creating new virtual worlds for players to explore. Stephen White solved this problem by introducing MOOs (MUD, object-oriented), which implemented the attractive features offered by MUDs and provided players with the capability of creating new virtual environments to explore [13]. Pavel Curtis, an employee at Xerox PARC, recognized the work done by Stephen White and embraced his basic design to create LambdaMOO, a system that provided a MOO server with a MUD database [9]. LambdaMOO allowed developers to program in the MOO language to create text based virtual worlds. Since the MOO language is a non-compilation language and the LambdaMOO system has a small main memory footprint [14], developers can instantly view their new virtual worlds and program modifications. Another attractive quality the LambdaMOO server possesses is portability. LambdaMOO is capable of running on a number of operating systems, for instance, it can run on several types of Unix environments including Free BSD. It can also run in Windows and Macintosh environments. The most common Windows version is WinMOO [15].

On the client side, the most widely used Macintosh development implementation is MacMOOSE (http://www-static.cc.gatech.edu/fac/Amy.Bruckman/MacMOOSE/), developed by Amy Bruckman (discussed briefly later in this section). This is a client GUI interface that allows developers to program in the MOO language and to access properties and verbs of specified objects, which they can modify, add, and delete [16]. Since MacMOOSE provided these attractive usability features for programming, it was later embedded in the MOO client, tkMOO-light [17], which provided programmers the option of 'graphical' programming using the MacMOOSE 'out of band' protocol on Windows client machines or programming from the command line.

Users connect to LambdaMOO via telnet, ssh, or these MOO clients. Once connected, LambdaMOO provides the users with vast user control and freedom. Users have the power to create their own virtual rooms with furnishings. They also have accessibility where they can navigate from room-to room by providing text based navigational commands. Users have the ability to view other users, their description, and location. A user can communicate with other users by 'shouting', where all connected users receive the instant message, which is generally viewed as rude unless the message has system-wide implications, such as a server shut-down event. A user can also communicate with only the players residing within their current room or whisper privately to a specific user. LambdaMOO also provides emoting where users can express emotions and physical actions to other players. As a result of these appealing features, LambdaMOO increased the number of popular MOO hangouts available on the Internet [18].

LambdaMOO also provided researchers and software programmers with the ability to develop new exciting educational tools and software such as online writing centers, electronic classrooms, cyberspace campuses, and educational games. Educational tools and software implementing LambdaMOO can be text based or, with software extensions, graphical.

An example MOO educational system, the MOOSE Crossings (http:// www-static.cc.gatech.edu/ elc/ moose-crossing/) was developed as a research project and than later as a doctoral dissertation by Amy Bruckman while attending MIT [19]. MOOSE Crossings provides educators with an additional tool to help students improve their reading, writing, and programming skills by using a learn through experience approach. Students become more engaged and interested in reading, writing, and programming by using their own creative imagination to develop virtual reality worlds using words and programming.

Another freely available tool, enCORE (http://lingua.utdallas.edu/encore/) is open source software that packages LambdaMOO and MacMOOSE utilities, created by Jan Rune Homevik and Cynthia Haynes [20]. EnCORE is an online system, with a server-side client, Xpress. Xpress is an optional GUI interface that allows users to view MOO environments. However, if users do not want to use Xpress, they can connect through telnet or the other available MOO clients. EnCORE follows the same learning approach emphasized by Bruckman, the learn-by-doing approach, which provides users with the means to create and modify objects and practice programming in an object-oriented language. It empowers educators by providing an educational tool they can use in collaborative learning, such as group projects or other methods such as distance learning or online conferencing.

Author's note:

The textbook, High Wired: On the Design, Use and Theory of Educational MOOs [18], provides more information about MOOs and MUDs and the software and systems spawned from them It consists of collected essays from well-known individuals in the MOO and MUD community. This collection of essays provides answers to MOO and MUD based questions and their usefulness in education. First, it covers the history of MUDs and MOOs and then provides a technical explanation, where readers can learn the step-by-step process for installing and setting up a MOO successfully. Afterwards, it provides the reasoning behind why MOOs are useful in education and MOO communities.

8. Professional Spaces

Professionally-oriented MOOs have evlolved alongside those in academia. These MOOs offer all of the same functionality and advantages of their educational counterparts, but are geared towards a more professional user. By offering more advanced services to an experienced clientele, the professional MOOs available today will often surpass their educationally-oriented counterparts.

Professional MOOs offer a collaborative environment for people wishing to communicate with other people, when everyone involved has a rather advanced grasp of the topics discussed within the MOO. This is in contrast to the educational MOOs, in which a large number of users who are relatively naïve in the given subject receive instructions from a few users who are very knowledgeable in that subject. For instance, in the case of the TAPPED IN system, the focus is on educational professionals (teachers, lecturers, etc.) and getting them to share knowledge, skills and techniques.

There are many examples of this type of MOO, both mature (fully functional), and in development. Most of these revolve around a central theme, usually an academic field of research. A few of these are MediaMOO, BioMOO, and the TAPPED IN system. Within these environments, experts and professionals share information on research projects, as well as general information and the latest developments within the respective fields. Often, these meeting places are invitation only, and the requirements for membership generally include a relatively advanced standing within the field (for instance, graduate student, researcher, teacher, etc.). This is done to assure a higher level of expertise among the participants, which can in turn foster further advances in the field without having to "catch anyone up."

The first of these, MediaMOO was developed by graduate students for graduate students. It is associated specifically with media-related disciplines and the research being conducted within those fields. This project was started in 1994 by Bruckman while a graduate student at MIT (previously discussed in this chapter) and is now hosted at Georgia Tech. According to the MediaMOO webpage, "MediaMOO is a professional online community for media researchers. It is a place to come meet colleagues in media studies and related fields and brainstorm, to hold colloquia and conferences, and to explore the serious side of this new medium." The requirements for membership are simply that one needs to be working on research in a media-related field. "The requirement that members be actively involved in research is more rigorously enforced, as this is a professional community." [21]. Discouraging people with only a passing interest in the subject matter ensures the content of the MOO will not be diluted in some way.

Another example of a professional MOO is TAPPED IN. This system, created by SRI International, is geared towards education professionals who can collaborate and share ideas and techniques. It is not for students, but rather a place for teachers and educators to refine their knowledge. Members include an international community of K-12 teachers, librarians, teacher education faculty, and education researchers. These professionals collaborate on professional development programs and informal workgroup activities with other members from all over the globe.

Within this virtual environment, educators can plan and conduct projects with colleagues and students, participate in (or lead) topical discussions, conduct and attend courses, find resources, experts, and new colleagues, serve as resources for other educators, or try out new ideas in a safe supportive environment (http://tappedin.org/tappedin). The TAPPED IN system has many ways for its members to collaborate, including customized virtual buildings with public, group, and personal rooms [22]. There are various methods by which users can communicate with each other. These include virtual meeting rooms, private messaging, discussion boards, and a built-in message delivery system similar to email. Of immense convenience is the feature that provides for file and URL storage within the virtual meeting rooms.

Another example of the many professional MOO projects being developed is BioMOO. This MOO is an ongoing project of the BioInformatics Unit of The Weizmann Institute of Science in Israel. BioMOO is a professional community of Biology researchers. "It is a place to come meet colleagues in Biology studies and related fields and brainstorm, to hold colloquia and conferences, to explore the serious side of this new medium." [23].

This resource is a forum for professional biology researchers. The members create colloquia and conferences, as well as share research ideas and information and brainstorm about biology and related fields. This MOO claims to be the most technologically advanced MOO of its type. "The BioMOO is one of the most successful MOO for communities of researchers. On a technical level, it has one of the most (if not the most) enhanced environment. Unlike the MediaMOO, it offers a WOO environment using interactive maps, databases with very simple search systems, etc." [23]. A "WOO" is a world wide web connection to a MOO. The most successful application of BioMOO has been online seminars. These are particularly amenable to the online meeting format and have proven productive and efficient. Transcripts of these seminars are recorded, and can be accessed via the Internet by anyone.

In addition to its core function in the fields related to biology, BioMOO also operates as the virtual campus for the Global Network Academy (GNA). The mission statement of this nonprofit organization is to open the academic world to anyone, anywhere. BioMOO provides the online virtual meeting spaces used by GNA to reach students all over the world who are then provided with social networking resources as well as advice and instruction.

The future for these and similar systems is bright. As network speeds, computing capacity, and sensory peripherals (user input devices) improve, the virtual collaboration experience is likely to improve. Already, great strides have been made in eliminating the physical-distance barriers between people. In the future, we can expect to see increases in these immersive technologies as a means of staying connected to one another in professional and personal realms. The potential for bringing together different ideas from different parts of the globe in a synergistic, cooperative manner is seemingly limitless. Having disparate experts in disparate locations could become a problem of the past.

9. Immersive Virtual Environments for Education

Constructivist approaches to education hold that true learning occurs when learners construct their own meanings about a given topic. Such constructions are thought to occur best within authentic scenarios for learning. For Immersive Virtual Environments (IVEs), authenticity has two major components. First, at the system scenario level, authenticity means not duplication but reflection of the dynamics of practical contexts [24]. In this way, it does not matter that Planet Oit of the Geology Explorer (described below), is fictional. Second, authentic at the individual learning level means agreement by the student to participate in the simulation scenario by means of a social identity and its component status and role.

This is a style of learning in direct contrast to traditional classroom learning. It requires the student to act a part and to suspend disbelief in the unreality of the virtual context [25]. For some students, this non-traditional approach to learning at first can be disconcerting. Nonetheless, by assuming roles, students partake in a simulation that provides meaningful experiences at multiple levels within an educational setting. These experiences in the IVE are designed to take advantage of enculturative conditions within socially situated environments.

We view these IVEs for education as the natural evolutionary outcome arising from the historical progression described above. What follows is a description and discussion of the current state of the art in IVEs for education.

Author's note:

The Worldwide Web Instructional Committee (WWWIC) at North Dakota State University (NDSU) is a multi-disciplinary group of faculty, staff, and students, working together to study learning in the context of multi-user virtual environments. The WWWIC group is composed of computer scientists, biologists, anthropologists, geologists, statisticians, and learning specialists. They WWWIC has developed a number of research artifacts, described below.

9.1 Dollar Bay

Dollar Bay is a fictitious seaside town intended to teach the principles and practices of retailing. To join the game, a player creates a character that is then assigned retail space and a starting budget. The player's goal is simple: make more profit than other store owners in Dollar Bay. However, the simulation presents a formidable and invigorating challenge. The economic environment is sensitive to a number of factors, and players must adapt to changing market forces. Perceived demand changes as other players enter the market and the game simulates seasonal effects on consumer purchasing. Dollar Bay players must anticipate these and other trends along with socioeconomic factors in order to adjust their business and keep it thriving. Depending on the success of their business decisions, a business might fail and be reaped, or succeed, and be inducted into the Hall of Fame [26].

Dollar Bay uses an intelligent tutoring system to oversee and guide the student's learning experience. The overall goal of intelligent tutoring is to implement context-sensitive advice within multi-user distributed simulations to help provide effective learning experiences [27]. The tutor is able to both guide students having difficulties and identify students who have made lucky guesses and let them know they did not follow the proper process in getting to their answer. Dollar Bay uses both rule-based tutoring and case-based tutoring.

Rule-based tutoring in Dollar Bay functions by maintaining a simple set of rules about the domain, monitoring student actions for any indication of breaking one of the rules, and then visiting the student to present a warning. For example, one rule concerns whether a student has set their prices to an excessive markup. In such an instance, the tutor sends a message to the student informing them they may be setting their prices too high [28].

Case-based tutoring provides an analysis of student behavior based on selected attributes and classification and advice based on comparisons with previously stored student records. The case-based retrieval attends to attributes such as product spread and advertising quotients. This system provides the means to generate personalized lessons for each student participating in the Dollar Bay environment [29].

Dollar Bay is available to play at http://dbay.ndsu.edu.

9.2 Blackwood

The Blackwood Project is the NDSU World Wide Web Instructional Committee's first attempt at creating a “next generation” role-based virtual environment for education – one where the pedagogical simulation will support cross-disciplinary content and a choice of roles to promote player interaction and collaboration. The project is intended to teach Microeconomics and American History through the simulation of a typical 19th century western town.

Like Dollar Bay, Blackwood aims to provide an engaging context for immersive, role-based education in a platform that teaches business-oriented problem solving in a hands-on, learn-by-doing, pedagogical style. However, unlike Dollar Bay's static environment, Blackwood simulates the historical events surrounding a mythical small town in the American Old West during the early 1880's. By implementing these historically accurate events, users are able to experience the effects of history on the economic environment first hand.

The Blackwood game has players join the simulation and assume a specific role in the virtual environment. Instead of having all players vie for a portion of the same economic market, players choose a role and become providers of specific goods or services. For example, you can choose to be a blacksmith, leather worker, dry goods merchant or a wagon maker. This allows players to compete for market share against other players or software agents trying to learn the same role in the simulated environment.

The game is designed to impart the time-independent principles of microeconomics and the practice of retailing within a historical context, and to promote the social and symbolic relations that sustain a business culture in the long term. As the simulation progresses, players learn their role in the environment and see the results of their actions as well as the impact of other player's actions within the constraints of the simulated world. Students are immersed in an authentic context, assigned authentic goals, given an opportunity to learn to operate in a historical context, and learn about culture and society while at the same time developing historical cross-cultural awareness and understanding [27].

To implement this comprehensive economic model with an authentic historical context, the simulation covers a six year period from spring of 1880 to the spring of 1886 where events in the simulation track actual events that occurred in the 1880-1886 time frame, with one day equivalent to about three weeks in the simulation. The model's economic trends reflect the impact of western expansion, the discovery of mineral deposits, population fluctuations, weather, the introduction of new ideas and technology, and the coming of the railroads.

The Blackwood environment is populated by a number of software agents. Customer agents that represent the demand of consumer groups of the time: Settlers, Farmers, Ranchers, Soldiers, Laborers, Miners, etc. Merchant agents represent the supply of business in competition with human players. Employee agents see to the daily operation of the players’ establishments and conduct the actual transactions with the customer agents.

The Blackwood project's homepage is: http://lions.cs.ndsu.nodak.edu/~blackwood/

9.3 The Virtual Archaeologist Project: Like-A-Fishhook

The Virtual Archaeologist project is an immersive three-dimensional (3D) educational environment that presents the science of archaeology in an intellectually engaging way by simulating the real-world conditions of an archaeology excavation. The goal is to develop an authentic simulation of an archaeological site where students experience the tasks and problems archaeologists have to face while conducting field research. The site chosen for the simulation is the Like-A-Fishhook village, located in central North Dakota in the United States. The village was occupied in the mid 1800's by members of the Mandan, Hidatsa, and Arikara tribes, known today as the Three Affiliated Tribes. Through the use of the Virtual Archaeologist, students can experience both the present day excavation as well as shift to the past, when the site was actually occupied, and see their discoveries in their actual context.

Archaeology is a discipline that relies on fieldwork, and students normally learn the conduct of archaeological research by taking a field course, but there are limitations to such courses. They can be expensive, often occur in remote locations and harsh conditions, and only a limited number of students can participate. Archaeology is also known as a "destructive" science and irreplaceable archaeological sites should not be used as training grounds for novices.

The goal of the Virtual Archaeologist is to overcome these restrictions and provide a means to teach students how to “think like an archaeologist” and learn to use the tools and resources available to researchers. To achieve this goal, the simulation consists of two environments representing two different time periods at the same location. The first is a representation of the modern day archaeological site and is the primary world in which the students will operate. This world consists of a 3D representation of the Like-A-Fishhook archaeological site, complete with models of actual artifacts environmental disturbances discovered. There, students are given a series of assignments and the tools and instructions necessary to carry out these assignments. Once the students have completed an assignment, they are given an opportunity to see their discoveries, in their historical context, by giving them a glimpse into the past.

The Like-A-Fishhook/Virtual Archaeologist home page is: http://fishhook.ndsu.edu/

9.4 Geology Explorer

The Geology Explorer is one of the many multi-user immersive virtual environments produced by the World Wide Web Instructional Committee. In the Geology Explorer, players take on the role of a geologist sent to explore a mythical planet called Planet Oit. Through exploration, experimentation, and guided collaboration, students learn about geology through a process of scaffolded enculturation.

The Geology Explorer provides a vast array of geologically interesting areas to explore, with over 50 different locations and hundreds of different rock and mineral outcrops to identify using a variety of geologically relevant field instruments. Areas to explore include a variety of geologically interesting areas: Old Mountains, Young Mountains, Plains, Lake, Desert, Hills, and Seashore. There is also an interpretive module, where students apply higher order geologic knowledge to interpret how a landform was created. Students can explore numerous facets of geology education. However, simply showing them rocks and minerals will not necessarily teach them geology. They also need to know how to act like scientists – like geologists.

Geologists typically take a number of instruments with them into the field. These instruments provide a range of useful functions. Some aid in the identification of rocks and minerals, some provide information about rock formations and structures, still others are necessary to allow the geologist to get to and observe the environment.

A variety of these tools for student use are implemented in the Geology Explorer, so that it is possible to perform experiments as geologists would. For example, by using the Glass Plate, students can determine a rock's hardness, using the Pocket Transit, students can determine the strike and dip of a particular outcrop, which is useful in the interpretation of geologic landforms.

After giving students an area to explore and the tools needed to learn in that context, students need direction and instruction to maximize learning potential. The Geology Explorer directs students in two distinct, yet interdependent ways. Students are led through a series of progressively more difficult tasks, teaching them more and more about geology, and throughout these tasks, software agents tutor students on appropriate ways to approach certain tasks.

There are two main tasks in the Geology Explorer. The first, initial module is called the identification module. In this module, the student is asked to locate and identify a series of rocks and minerals. To begin with, rocks and minerals are relatively easy to find and identify. Players are given multiple hints to aid in the location and identification of their goal. Upon correctly identifying each rock or mineral, however, hints are faded, and the student is given progressively more difficult rocks and minerals. During this module, software tutors help students by pointing out what equipment the student will need to successfully identify their goal, showing the student what tests should be performed to come the correct conclusion, and explaining why their hypothesis is incorrect by pointing out contradictory experiments that would disprove its identity.

The second module is called the Interpretive Module. In this module, the student is asked to go to a different area of the planet, where an interesting geological phenomenon has occurred in the past. The student's job is to identify the outcrops in the region, create a geologic map of the area and, based on that map and other experimental data, determine the shortest length a tunnel could be driven through a particularly unstable rock area. Tutors in this module tell the students if they have missed outcrops that they should identify, help the students make a valid geologic map, and all of the tutoring help from the first module while identifying the underlying outcrops.

By giving them a vast area to explore, the tools needed to explore that area, and pedagogical support, students are able to enter the virtual world as novices, assume the role of a scientist, and come out knowing a great deal about the Earth, its geology, and how a geologist works.

The Geology Explorer web site is: http://oit.ndsu.edu.

9.5 ProgrammingLand MOOseum

The ProgrammingLand Museum [30] is a text-based multi-user immersive virtual environment, used in conjunction with a real life lecture to teach computer languages. In a real life museum, individual exhibits provide nuggets of information about a particular subject. In the ProgrammingLand museum, a series of rooms acts as those individual exhibits. To traverse these rooms, students follow a series of hyperlinks, looking at each exhibit and slowly learning more and more about the programming language being taught. As in a real museum, some of the exhibits include a hands-on activity, teaching the students how to apply the knowledge they have gained.

The ProgrammingLand museum currently has four different "wings" to explore, each covering a different programming language: C++, Java, BASIC, and LambdaMOO, the language employed by the museum itself. Instructors are able to track student progress through each of the wings, enhancing its use as a lecture tool.

The webpage for the ProgrammingLand Museum is: http://euler.vcsu.edu/pland.htm

9.6 The NDSU Virtual Cell

In the NDSU Virtual Cell game, cellular processes are taught through simulations of the cellular processes and interactive goals. In order for students to learn the cellular process effectively, simulations require both speed and accuracy.. Without these two qualities, obstacles are created to fun and learning, which may cause students to have negative attitudes toward the game.

Currently, the Virtual Cell server is transitioning from a LambdaMOO to JavaMOO server. Along with this transition come extensive modifications in how simulations are implemented. The most drastic change is moving the simulation from the server to the client. This alteration reduces network traffic between the client and server since the server no longer needs to update the client on the current state of the simulation. However, there still exists a small amount of communication between the server and client. The server provides information to the client on the number and type of molecules and complexes to reside within simulation. Once the server has supplied the initial information, communication ceases until the simulation has been manually stopped, paused, or rewound by the player or the simulation has finished on its own.

Another simulation improvement is having the client calculate the different positions for elements and molecules prior to running the simulation. This maintains simulation speed where the previous version calculated positions on the fly, causing some lag during the simulation.

Another modification addresses the creating and destroying of molecules and elements during the simulation. For example, in the Electron Transport Chain process, the NADH Dehydrogenase Complex removes two electrons from the NADH molecule. Now, the NADH molecule becomes NAD+ and two additional elements are created, electrons. During server side simulation, the electrons and NAD+ molecule would be dynamically created while the NADH molecule would be destroyed when needed. This was a complex process that created a noticeable lag in the simulation until the molecule(s) and element(s) were completely created/destroyed. In addition, destroyed molecules and elements caused another layer of complexity, due to its need for the garbage collector to collect elements after they were destroyed or remember which elements were destroyed to clean up at a later time. This is resolved by creating all required molecules and elements for the simulation before the simulation begins. Instead of destroying molecules, the molecules and elements simply become invisible and visible when notified by the client. Another improvement provides players with more control of the simulation by having additional control buttons, pause and rewind. The rewind button allows the player to re-watch a part of the simulation they might have missed or did not understand.

These new modifications create a more robust simulation. It enables speed and timing accuracy of the elements and molecules in the simulation by eliminating obstacles in the way of fun and learning and reducing players' frustration with the simulation process. Yet, new problems arise with the transition from server to client side simulation. The current state of the client side simulation eliminates multi-user experiences, because client side simulation does not allow for two players to watch the same simulation as a result of them viewing different clients that hold different simulations. This takes away from player collaboration and players asking or providing advice to their peers. In order to allow players to view the same simulation, their clients need to know the identical simulation state event to start on. This can be achieved by one client providing information to the server and the server updating the other client.

The NDSU Virtual Cell web site is: http://vcell.ndsu.edu.

10. Role Theory and Learning in WWWIC IVEs

WWWIC's immersive virtual environments continue to advance Science of Learning theory and practice. The approach adapted, extended, and implemented by WWWIC IVE developers is founded on a strong anthropological perspective. The disciplines of psychology and sociology, as well as education theory, drama theory, linguistic theory, and epistemological philosophy inform our dynamic perspective.

10.1 The Learning Paradigms Debate

To put the Science of Learning framework that is integral to the WWWIC IVEs in context, it is useful first to grasp the contemporary background of debate about learning and learning theory approaches and practices. Today there exist three major paradigms for learning and education frameworks. While initially focused on children and teaching, these paradigms have been extended in application to higher education and adult learning and are advancing knowledge about learning in virtual environments. These three paradigms are 1) the former status-quo model, known as the "instructionist," "instructivist," "behaviorist," "essentialist," "traditional," or "classical" model, 2) a transitional model, known as the "cognitive" or "comprehension" model, and 3) the model that has gained favor in recent decades, the "progressivist" or "constructivist" model, and its computer age sub-variant, the "constructionist" model (for this last, see [31]). In this chapter, we refer to the first approach as the *instructionist model*, the next as the *cognitive model*, and the newer approach as the *constructivist model*.

These three approaches have been compared and contrasted in detail by others (e.g., [32]). Here they are generalized in broad terms and briefly compared in structural, functional, processural, and ideological terms. Arguably, these three models represent points on a continuum, with the instructionist and constructivist models discrete, and the cognitive model a transitional phase in the development from instructionist to constructivist approaches. Typically, learning situations combine various aspects of each of these models. That is, constructivist learning scenarios, themselves heavily influenced by the cognitive model, often take place as discrete learning activities within a traditional classroom institution. There exist many and varied practices and research based on these models and combinations of these models, but we deal here with idealized stereotyped synopses for the purpose of simplicity.

Instructionist model.

Structure. The learning environment is a formalized hierarchical setting of social actors defined as students and teachers, each with a set of tools appropriate to their rank. The student inhabits the social place of learner and subordinate, while the teacher inhabits the place of knowledge transmitter and authority. The educational environment is separate from the context within which professionals in a given domain act upon knowledge. That is, the student is exposed to information about a given subject but typically not involved in the practice of the subject itself. Evaluation typically occurs at the end of a series of lessons.

Function. The students are to learn new material and gain new knowledge, and prove the acquisition of this knowledge to the teacher.

Process. Linear. The teacher provides organized content through use of various tools and materials, including textbooks and audiovisual supplements. The student is the receptacle (*tabula rasa*) for that content, and typically uses tools of paper and pen or digital devices to gather the information provided by the teacher and the teaching materials. In other words, the teacher acts as an active transmitter, while the student acts as a passive receiver and signal repeater. Student performances in rote memorization, nomenclature use, acquisition of taxonomies of facts, and the ability to recognize these facts at any given moment are privileged. Evaluation of acquired knowledge is isolated from the field of practice; evaluation is retrospective in the sense of determining what the student *has* learned (rather than what the student *is* learning); and evaluation is done in batch mode rather than continuous and conterminous to the learning process. Evaluation usually is accomplished through formal examinations.

Ideological Basis. Repetition is key to learning. Recall and practice (changes in behavior) are evidence of learning. Learning is the addition of more and more information and associated skills into a solid whole. Cognitive ability principally is linked to age while cultural context is ignored or de-emphasized.

Cognitive model.

Structure. The learning environment is similar to the instructionist model.

Function. The students link new material to previous knowledge, thus readying themselves for the addition of new knowledge to be gained through activities that will solidify and expand the connections between what was previously known and what was recently learned.

Process. Linear within recursivity; an oscillation between linearity and circularity. The teaching processes are similar to the instructionist model. However, the student is not an automaton but recognized as a selective filtering agent who organizes and connects new knowledge to his or her own individual existing knowledge and knowledge structures. Most crucial to the model, the student will *retain* data significant to that individual, for whatever reason that significance exists. Feedback is important to the process both as a way to impel the recursive nature of cognition and as a way to affect or counter the sociocultural context of learning. In other words, the teacher acts as an active transmitter and feedback mechanism, while the student acts as an active (rather than passive) receiver. Short-term memorization is quickly lost, while long-term comprehension focuses on a subset of the input received. This occurs, in part, through the student's use of the classical student tool kit (note-taking, oral explanation, et al.) referred to by researchers as elaboration activities. Evaluation of acquired knowledge is similar to the instructionist model with the added emphasis on feedback from teacher to student.

Ideological Basis. Thinking (cognitive processes) is key to learning. Recall, reconstruction, inference, and praxis (practical conduct) all are evidence of learning. The sociocultural context of learning will influence what is learned. Learning is the resolution of comprehended information into advanced knowledge. This involves the cognitive placement of the comprehended material into existing webs or schemata of knowledge already organized by the student in her or his mind. Overall knowledge as well as knowledge structures simultaneously are modified, either enriched or changed, by this new knowledge.

Constructivist model.

Structure. The learning environment is a simulation of real-life contexts in which problems are embedded and knowledge is gained, used, and (re)created (commonly referred to as situated, genuine, or immersive environments). Hierarchy exists but is not the only powerful factor – structures may be vaguely defined, problems typically have duration and complexity, and problems may reformulate into new variations. Often the student is placed at the rank of entry-level professional among a cohort of other entry-level participants – cooperation and shared consequences exist. Evaluation of acquired knowledge is ongoing through feedback mechanisms and student reflexivity.

Function. The students are to learn new material and gain new skills through both tactical uses of knowledge and strategic approaches to gaining knowledge.

Process. Linear within recursivity; an oscillation between linearity and circularity. Role-play and collaborative construction of knowledge. Evaluation through peer communication, self-assessment, and feedback from expert agents, as well as normative testing, is continuous and conterminous to the learning process.

Ideological Basis. Cognition is adaptive, recursive, reflexive, and socioculturally situated. Learning and practice are active social processes occurring within relevant enculturating contexts. The student is presumed to be the active constructor of knowledge. When learners dynamically engage with new resources and data in ways that require reflection and cooperation, new knowledge can be gained and acted upon. Pragmatism, especially in the sense of learning-by-doing, is the guiding principle, with the idea of "scaffolding" to increasing levels of complexity driving the learning process. The computer age sub-variant of this approach is the *constructionist* school. Constructionist approaches involve students actually creating games and other educational materials as the vehicle for learning. Inverting the equation of a student learning by playing a game, constructionists argue students learn by creating the game or other materials themselves.

The basic premise of constructivist learning theory as informed by cognitive theories (heavily influenced by the work of Swiss psychologist and philosopher Jean Piaget) is that individuals piece together understanding by reference to their own comprehension of materials and to their own individual experiences, a process involving, among other factors, qualitative transformation of individual understanding (see [33] for an edited volume considered to be the mainstay of the constructivist approach. For criticisms of the approach, see [34] and [35]).

Our version of the constructivist approach attempts to align both the socially situated nature of knowledge and the individual construction of knowledge through the use of immersive role interaction in an authentic virtual scenario setting. MUDs lend themselves to immersive experiences, and the WWWIC research agenda has focused on capitalizing on this through promoting roles within exploratory science learning experiences. In this way, performance and construction of knowledge are primarily under the control of the student, with evaluation of student learning occurring both within the IVE as immediate feedback through embedded assessments and interaction with intelligent agents or tutors, as well as later, post-IVE, through more traditional retrospective quantitative and qualitative analyses by the instructors. The driving theoretical parameters in the WWWIC IVEs for student learning are performance of role, authentic scenarios, and enculturation processes.

10.2 Role theory

Since the 1960s, roles and role-playing have become an increasingly important component of the "progressive education" or cognitive and constructivist paradigms. While roles are commonly used in a variety of teaching situations, from elementary education to on-the-job training, the theory behind roles is rarely articulated. Here we summarize briefly role theory: what roles are, how they work, and what can be expected from them.

Roles really are a component of social identity. *Social* refers to the material and symbolic interaction of the individual and the group. *Social identity* is a theoretical construct defined by status and role and is part of the theory of social organization. *Social organization* refers to groups of people interrelated in a particular arrangement and is based on ideas (shared generally by members of the group) about the group's membership and associated identities.

Social identity involves status and role. *Status* is the structural component of the identity; it refers to categories or taxonomies of group membership. Statuses, as social structural places within a group, typically have labels, such as mother, boss, significant other, student, geologist, white, Native American, woman, boy, rich, middle-class, educated, criminal, and so forth. Statuses are either ascribed or achieved. *Ascribed* status is assigned to a group member without the participation of that member, in other words, the individual has no control over the initial giving of that status to oneself. Typically, ascribed status deals with larger taxonomies in the social organization, such as ethnicity, class, or other attributes assigned at birth, depending on the social organization of the group under study. In the WWWIC IVEs, participants initially are given an ascribed status, but the remainder of their social identity is up to them. For example, in the Geology Explorer, participants initially are ascribed the status of Geologist.

Achieved status results from the actions of an individual, such as student, criminal, boss, and so forth. Achieved statuses are created within the IVE by the participants through their performance and reflection on that performance and the performance of others within the IVE. Achieved status is the primary social identity at work within IVEs, resulting in varying levels of prestige. Prestige refers to social value associated with functional and dynamic aspects of social organization, especially role.

An abstract construct or idea created and shared by a social group, a *role* is a general and flexible *set of rules for behavior associated with a particular status.* Hence, *role* is the *dynamic aspect of a social identity.* What makes role dynamic is that *it is performance of the status.* Performance of the status is the driving experience of students in IVEs.

For example, in the WWWIC IVE Geology Explorer, students take on the status of "geologist" and perform the role of "exploring geologist." However, this role performance is not as straightforward as it appears. Roles are reflexive, involve self-reference during and after an act, and can affect the actor. Because roles are reflexive, they are dynamic, and open to improvisation, to self-assessment, and to shades and degrees of variation as created and practiced by an individual.

As students new to the Geology Explorer IVE adapt to that IVE and gain increasing comfort with their status and role, they will be focused on learning the IVE tools and a specific tactical approach to their role. In contrast, experienced students who have been in the Geology Explorer several times no longer may be concerned with figuring out how to move about or communicate with others within the IVE. They have mastered those types of activities, and so they may begin to focus on developing a strategic approach to complete the overall set of assigned problems. In so doing, these students become more advanced 'exploring geologists' than the first-time users. These students often will self-assign a second status to themselves, that of advanced user, and associate with this new status a higher prestige than that which this student now associates with the brand-new user. In this way, the initial static social organization of the IVE is reconfigured to include achieved as well as assigned statuses. In this way, students effect a new social learning environment within the IVE. This newer social organization form works to make the students feel comfortable with their social identity within the IVE and advances their ability to reflect and analyze their performance based on on-going change. Students move from analyzing their utilitarian use of tools to creating a strategic approach to the problem set they are to perform.

To summarize, by relying on the dynamic and reflexive nature of role performance, IVEs teach more than "how-to" knowledge. Students gain not only a utilitarian tactical approach to a specific problem, but also a strategic approach to a class of problems. In this context, performance and construction of knowledge is primarily under the control of the student.

10.3 Performance theory

In the WWWIC IVEs, student performance depends on ever-increasing knowledge and the ability to successfully interpret and practice the role associated with the virtual environment. Performance theory in anthropology describes how social actors use the dynamics of social interaction to create or produce outcomes on others and themselves.

Performances are reflexive, where learners learn about themselves [36]:81). Performance theory is characterized by a concern with the productive force [37]:567. Role performance directly affects the learning dynamics of a student in an IVE. More than mere goal-oriented "doing" of a task, role-based simulation learning is learning-by-performance.

Role performance learning is driven by individual experiences and "other-dependent learning" in both the virtual and real worlds. Other-dependent learning involves "conditions of informally guided discovery" [38]:186. "Others" in the IVE environment include software tutoring agents whose virtual behavior takes the form of powerful hints and occasional correction that is critical to other-dependent learning. Similar learning theory is found among business management theoreticians, who emphasize the role of people who are "third party brokers" and "go-betweens" [39].

In WWWIC IVEs, learners experience reflexive cognitive performative encounters. 'Reflexive cognitive performative encounters' are social interplay that produce affects either on the performer or on the other social actors [40], such as when students are self-conscious of their language interactions and usage, and perceive their role as one "to display for others" a certain grasp of specialized concepts and language. For example, advanced students with experience and expertise may share with less-experienced students strategies for approaching and completing problem within the IVE. The teacher-student affects not only the performance of the less-experienced student, but also enriches his or her own understanding of the problems facing the IVE participant. This is similar to what educators discover as they teach a topic, that they themselves learn more about that topic through their interaction with students.

The performative aspect of the IVE experience allows us to move beyond simplistic analysis of whether or not a student's work is correct or incorrect and is indicative of change in learning levels [41]. For the virtual learner, performative social interaction develops and changes as the student progresses through levels of understanding and learning [42]:182. This type of performative encounter has been observed in other learning situations, such as mathematics classes [43] and drama classes [44],[45].

Performance of social identity in the IVE, therefore, includes not only the accomplishment of set objectives, but also the enriching of the IVE environment itself. The socially situated learning environment is performed, not merely inhabited. Performance, therefore, creates the dynamic necessary to authentic learning scenarios.

10.4 Enculturation and learning

Repeatedly, cognitive and constructivist theoreticians stress that learning is socially and culturally situated. Just what does this mean? It means that learning is not divorced from the reality of a person's life and their place in the unfolding story of humanity's existence and condition. More specifically, it means that, fundamentally, individuals have individual agency, that is, they are able to think for themselves and decide for themselves what behavior and actions they will undertake. Such agency, however, is always located within a social context that has historical influence on the actor, reflecting both informal social norms and formal rules for action shared by a given group. This social context, its material, symbolic, and dynamic aspects, can be understood as enculturative conditions.

Enculturation classically refers to the processes by which a human group's ideas and behaviors are passed from one generation to the next. It is a fundamentally social process by which individuals adapt to their environment and learn to fulfill the function of their statuses (social identities) through the learning and performing of associated role behavior (rights, duties, and obligations). In contemporary anthropological usage, enculturation generally refers to the learning of a culture in terms of behavior and symbolic content, including belief systems [46]. Enculturation combines experiential, empirical, and propositional knowledge. It is an adaptive and cognitive experience that uses material context and content and symbolic context and content to bridge the gap between cognizance of new ideas and practice relying on those ideas. [Enculturation is sometimes given as a synonym for the sociologist's term 'socialization.' The difference in these terms is beyond the scope of this chapter, but it is important to note that we use the term enculturation because we are emphasizing not a conditioning process but an adaptation process.]

For instance, students accustomed to expensive private schools share a historical situation and worldview different from students in under-financed public schools. Thus, in an IVE setting that brings people from such diverse backgrounds together, norms and rules are not evenly shared among members of the IVE social setting. On the one hand, hegemony of western educational cultures means that much behavior by individuals in the IVE can be predicted. On the other hand, there is room for much variation in behavior patterns. Original thought and behavior that may be based on intent to do what is not considered socially "normal" can occur. From this, we get creativity. This creativity is not always thoughtful nor polite nor even successful in its intent, but when it is, originality and advancement can be produced that affects the whole group, not just the individual.

Therefore, IVEs developed by WWWIC work to promote an overarching social situation by creating enculturating conditions intended to harness the creativity of individual agency and guide that creativity to efficacious outcomes. The enculturative conditions include, among others, the structural contents of the IVE itself, the structured and unstructured dynamic interactions among students and others, and the material and physical interaction of students with the system. Even in fairly passive interaction with an IVE, students are constrained to dealing with the enculturative conditions present.

The catalyst that transforms the virtual world and its component enculturative conditions into a learning experience is role performance. Because the virtual reality is embedded within everyday reality, the student's understanding of the virtual problem is transferable to real-world problems, using the same classes of psychological and social processes that are associated with individual learning through problemexposure [47], unceremonious social coaching [38], and innovation diffusion within a cultural system [48]. It is the ability of the individual to adapt to a new cultural system by assuming a new social identity (role, status, and prestige) that drives the immersive virtual environment (IVE) [49]. This choice is conscious on the part of the students, whether or not they recognize all the ways in which the virtual system has been organized to filter their attention directly to learning practices through enculturative conditions.

In summary, we are building and enhancing learning environments that takes both collective activity and the individual experience into account. WWWIC IVEs have been engineered to provide enculturative conditions that focus the students attention on learning specific practices while at the same time comprehending strategic approaches to a given discipline. Unlike life in the classroom, there is little to distract the student from their performance. Furthermore, our work advances two theoretical stances in the Science of Learning: 1) IVE cultures are not simply prototypes or mimicry of “real world” culture. Through social interaction among learners inside the IVE, the IVE becomes a culture system just as ‘real’ as cultures of the classroom. As such, it is available for critical study of learning structures and processes at both the individual and the system level. 2) IVE culture can be designed for learning sciences experimentation – we can develop hypotheses and engineer changes in the IVE culture to test hypotheses.

11. Software Tutors in WWWIC IVEs

While completing their goals within WWWIC IVE games, students may encounter difficulties. A tutoring system that provides guidance without deterring the student's ability to independently draw conclusions has been developed.

The current tutoring system consists of several software agents and a primary tutoring agent, identified as the Tutor Controller. The multiple software agents are goal-specific and handle independent tasks while the Tutor Controller distributes information regarding student activities and progress to the appropriate agent. Student history information are stored so tutoring agents may draw conclusions based on player activities, such as when students need help. Once an agent perceives a student needs help, the agent determines the type of help needed and chooses a remediation response. Typically, tutor responses are classified into several categories where each category represents a possible student behavior in attempting to complete the goal. For example, in the Organelle Identification module of the NDSU Virtual Cell game, one response category is a successful goal completion, where the players must choose the correct assay to perform an experiment on a sub-cellular organelle and then correctly identify the component. After a response has been issued, the response is used in selecting a remediation. The chosen remediation is displayed to the student, encouraging student thinking and guiding them towards using the resources of the Virtual Cell game. When repetitive student behavior occurs, the remediations become more detailed, providing additional insight for the student.

12. Technical Issues with Current IVEs

The aim of this advanced learning technology project is to move beyond the laboratory and venture into the realm of serious games and systems for real-world deployment. Research in this area is essential to this progress. To make these improvements, it is necessary to leave behind the rapid prototyping systems that got us here, and move on to more advanced learning technologies. This means moving beyond LambdaMOO [9] and onto the next generation of learning systems.

LambdaMOO is a dynamic, real-time, network based simulation environment built on an object-oriented database, with an interpreted object-oriented programming language. The LambdaMOO platform was used for WWWIC’s original suite of educational environments and has many fine qualities, including the ability to replace and compile individual methods on objects without recompiling any other parts of the application. LambdaMOO also features a fail-safe mechanism for canceling runaway processes but the overhead involved in monitoring all these executions is considerable and creates obvious inefficiencies in production systems. Further, and most serious from a performance point of view, the LambdaMOO execution environment is purely interpreted, unlike modern Smalltalk or Java environments, LambdaMOO is a basic byte-code interpreter without any support for just-in-time compilation or similar improvements. Due to the small size of its development community, it is unlikely that LambdaMOO will become competitive with other more popular interpreted languages.

Source code control is also a serious problem in development environments of this type. There is no facility for checking code in and out beyond a cumbersome and somewhat baroque system of ownerships and permissions. Nor is there is a facility for version control or change conflict resolution. In fact, LambdaMOO provides no built-in support for file handling of any sort. The only way to write and replace code is to enter it at the command line, or 'send' code with one of the freely available editing clients. Therefore, all code in LambdaMOO is equivalent to the 'stored procedures' implemented in a distributed database, where the state of the art is to fashion updates that are tested in a development 'mirror' of the database and then uploaded to the 'live' server during maintenance periods. In this mode of operation, source code and version control are relegated to external offline processes.

There are also no facilities in LambdaMOO for the type of development support that modern software developers expect: clickable reference to inspectable objects, visual interfaces for managing classes and methods, even editing applications with convenient indicators for bracket matching and variable references. In addition to the issues already mentioned, there is no facility for synchronizing mirrored MOOs, and consequently no way to fully protect against a network outage. Nor is there a way to support a distributed approach where code and data are protected against all-too-common server crashes and network accidents. These common elements are increasingly necessary as applications grow and evolve, and LambdaMOO suffers from their absence.

13. JavaMOO Technology for Education

JavaMOO is a framework for building virtual worlds used for educational purposes, intended to replace LambdaMOO, which served as the implementation platform for WWWIC's original suite of educational virtual environments.

While LambdaMOO provides the essential functionality needed for building educational virtual worlds, it has many drawbacks that inhibited the development and distribution of large-scale environments. JavaMOO was written to address those drawbacks, and further facilitate the creation and deployment of educational virtual worlds.

The JavaMOO framework consists of 1) a suite of servers providing a reliable and robust environmental simulation and managing the interaction of dozens of connected students, 2) an application programmers' interface (API) for building an application-specific client that presents the state of the environment and allows students to interact with the environment and each other, and 3) a Data Collection Server which collects server information and tracks student progress in each game. Figure 1 shows all of these components and how they communicate with each other. This framework provides teachers with a self-contained package that is easy to setup and use, along with maintenance functions that allow instructors to track student progress, and WWWIC to monitor the status of the server.

The server of a JavaMOO virtual environment actually hosts a set of servers. The central component is the JavaMOO Server itself, which is responsible for executing the simulation, handling communication with clients connected to it, and communicating with the database to store and retrieve simulation and environment data. JavaMOO provides a set of foundation classes that form the base of the JavaMOO server. The game developer builds the simulated environment on top of this base. The foundation classes handle all the client communication and data storage, so the game developer does not need to worry about these features.

The JavaMOO server package comes with an SQL database, which the JavaMOO Server uses to store and retrieve simulation data. Account information is also stored in the database. While nearly any SQL server would work with our system, JavaMOO comes with the Apache Derby SQL database. By using an external database, JavaMOO has quick and reliable access to the data it needs.

Client/Server communication is handled using Java's Remote Method Invocation (RMI) system. In order for a client to connect to the JavaMOO Server, the server must first register with an RMI Server. The client then contacts the RMI Server to establish a connection with the JavaMOO server. Once this connection has been established, the RMI Server is no longer explicitly addressed.

The JavaMOO server package also comes with Tomcat, an HTTP server, which is used to host a Registration Server and a Media Server. The Registration Server is used by the client to create new accounts, and ask students to fill out surveys with demographic information and knowledge evaluation. This information is used to track student learning and provide data for larger studies. The Media Server holds images, animations, and other media used by the client. By storing these images on the central server, they do not need to take up storage on the client computers. This system also allows the game media to be updated without reinstalling a new client or rebooting the JavaMOO Server.

Together, these components make up the JavaMOO server package. A single server is created for each game being played by each class. Currently, this server is being tested by WWWIC as the future of IVE development.

14. The Future: Group Play

As to the future, we predict the next 'big thing' in IVE research will be *group play*. Cooperative learning provides many benefits, both short and long term. In a cooperative environment, students depend on each other to do well and provide support in order to succeed themselves. Cooperative work increases self-esteem, time on task, and altruism while decreasing disruptive activity [50]. Giving students the opportunity to work in groups provides experiences in supporting skills useful throughout both personal and occupational lives. Brown, Collins, and Diguid [51] argue, "If people are going to learn and work in conjunction with others, they must be given the situated opportunity to develop those skills."

Group work in virtual worlds involves three components. First, students and/or teachers need the ability to create groups. Second, these groups need a variety of communication mediums, both synchronous and asynchronous, to allow exchange of information and ideas among present and not present group members. Third, goals must be available for groups to complete, evaluation must be ongoing and accurate, and rewards must be present for people who choose to form groups. Players with a group-centric set of goals employ constraints that map into Kagan's four primary principles for effective group learning: positive interdependence, individual accountability, equal participation, and simultaneous interaction [52].

15. Conclusion

The evolution of MUDs and MOOs extends back to the days of main frame computers, paper tape readers, punched cards and so-called dumb terminals. This chapter has traced that evolution through the successive generations of MUD/MOO use, culminating in the state of the art as it exists today in education. Immersive Virtual Environments for education hold the potential to alter the way students learn by doing. The IVEs described above provide a safe method for students to experience learning and learn through experience. These hold the further potential to bring learning opportunities to students that might otherwise lack the opportunity. One of the many opportunities involve computer mediated cooperative learning.

Cooperative learning provides many benefits, both short and long term. In a cooperative environment students depend on each other to do well and provide support in order to succeed themselves. Cooperative work increases self-esteem, time on task, and altruism while decreasing disruptive activity [50]. Giving students the opportunity to work in groups provides experiences in supporting skills useful throughout both personal and occupational lives. Brown, Collins, and Diguid [51] state it this way, "If people are going to learn and work in conjunction with others, they must be given the situated opportunity to develop those skills."

15. Acknowledgements

WWWIC research has been supported in part by the National Science Foundation EIA-0081642 and the U.S. Department of Education FIPSE P116B030120.

16. Resources

Books

Ghaoui, C., Jain, M., Bannore, V. and Jain, L.C., (Editors), Knowledge-Based Virtual Education, Springer-Verlag, Germany, 2005.

Haynes, C., & Holmevik, J. R. (eds.). (1997), High Wired: On the Design, Use and Theory of Educational MOOs. University of Michigan.

Jain, L.C., Howlett, R.J., Ichalkaranje, N., and Tonfoni, G.(Editors), Virtual Environments for Teaching and Learning, World Scientific Publishing Company Singapore, 2002.

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Oxford University Press.

Rogoff, Barbara 2003 The cultural nature of human development. New York: Oxford University Press.

Spindler, George and Louise Spindler 2000 Fifty Years of Anthropology and Education, 1950-2000: A Spindler Anthology. Mahwah, New Jersey: Lawrence Erlbaum Associates.

Slator, Brian M., Richard Beckwith, Lisa Brandt, Harold Chaput, Jeffrey T. Clark, Lisa M. Daniels, Curt Hill, Phil McClean, John Opgrande, Bernhardt Saini-Eidukat, Donald P. Schwert, Bradley Vender, Alan R. White. (2006). *Electric Worlds in the Classroom: Teaching and Learning with Role-Based Computer Games.* New York: Teachers College Press. Columbia University. 192 pages.

Book Chapters

Curtis, P. (1997). High Wired: On the Design, Use and Theory of Educational MOOs. University of Michigan.

Daniels, Lisa M, Phil McClean, Bernhardt Saini-Eidukat, Donald P. Schwert, Brian M. Slator, Jeff Terpstra, Alan R. White, Kimberly Addicott (2005). Effects of teaching Science through immersive virtual environments. In Ma, Z (ed), Web-based Intelligent e-Learning Systems: Technologies and Applications, Idea Group, Inc., Hershey, PA. Chapter XIV, pp. 271-290.

Dede, C. (1996). The evolution of constructivist learning environments: Immersion in distributed, virtual worlds. In Constructivist learning environments; Case studies in instructional design, Wilson, B. G. (ed.), (pp.165-175). Englewood Cliffs, NJ: Educational Technology Publications.

Edelson, D., Pea, R., & Gomez, L. (1996). Constructivism in the Collaboratory. In B. G. Wilson (Ed.), Constructivist learning environments: Case studies in instructional design, (pp. 151-164). Englewood Cliffs, NJ: Educational Technology Publications.

Regian, J. W., & Shute, V. J. (1994). Evaluating intelligent tutoring systems, in Baker, Eva L., and O'Neil, Harold F. Jr., eds., Technology assessment in education and training: Hillsdale, NJ, Lawrence Erlbaum Associates, Publishers, (pp.79-96).

Tedman, D. and Jain, L.C., An Introduction to Innovative Teaching and Learning, in Teaching and Learning, Springer, 2000, pp. 1-30, Chapter 1.

Key Papers

Alpaslan F.N. and Jain L.C., Virtual Al Classroom: A Proposal, Proc. 1st International Workshop on Hybrid Intelligent Systems (HIS-2001) in Advances in Soft Computing, 2002, Springer, Germany, pp.485-495.

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated Cognition and the Culture of Learning, Educational Researcher, 18(1), (pp.32-42).

Ertmer, Peggy A., & Newby, Timothy J. 1993 Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. *Performance Improvement Quarterly 6*(4):50-72. Silver Spring, MD: International Society for Performance Improvement.

Jain, L.C., Knowledge-Based engineering: An Innovative Teaching Approach, Proceedings of the

Proceedings of the Eighth Turkish Symposium on Artificial Intelligence and Neural Networks, June 1999, pp. 15-19.

Jain, L.C. and Bowden, B.S., Introductory Course on Expert Systems in the University of South Australia, Proceedings of PRCEEE, Adelaide, 1992, pp.194-200.

Kafai, Yasmin B. 2006 Playing and Making Games for Learning: Instructionist and Constructionist Perspectives for Game Studies. *Games and Culture* 1(1):36-40. Thousand Oaks, CA: Sage Publications.

Rowland, J.G. and Jain, L.C., Artificial Intelligence Languages in Engineering Education, Proceedings of PRCEE, Adelaide, 1992, pp.201-206.

References

1. Raymond, E. (1991). The New Hackers Dictionary. Cambridge, MA: MIT Press.
2. Bartle, R. (1990). Interactive multi-user computer games. Technical report, BT Martlesham Research Laboratories, Dec. 1990.
3. Bartle, R. (1996). Hearts, clubs, diamonds, spades: Players who suit MUDS, Journal of MUD Research 1(1).
4. Keegan, M. (1997). A classification of MUDs. Journal of MUD Research 2(2).
5. Curtis, P. and Nichols, D. A. (1994). MUDs Grow Up: Social Virtual Reality in the Real World, in Proceedings of the 1994 IEEE Computer Conference.
6. Morningstar, C. and Farmer, F. R. (1991). The lessons of Lucasfilm's Habitat. In M. Benedikt (Ed.), Cyberspace: First steps (pp. 273-302). Cambridge, MA: MIT Press.
7. A Chronology of Computer History. 24 July, 2006 http://www.cyberstreet.com/hcs/museum/chron.htm
8. Nielsen Entertainment 2006 Active Gamer Benchmark Study, 2006. New York: VNU Corporation.
9. Curtis, P. (1997). LambdaMOO Programmer's Manual, http://www.hayseed.net/MOO/ manuals/ ProgrammersManual_toc.html.
10. Jazdzewski, Stephen, and Jazdzewski, Charles. Cool Jargon of the Day. Retrieved 24 July, 2006 < http://www.jargon.net/jargonfile/t/toad.html>
11. Kirk, Brendon. "Digital Addiction and MMORPG Phenomenon." 2005. TheTechZone.com. <http://www.thetechzone.com/?m=show&id=463>.
12. NDSU Archaeology Technologies Laboratory. 2004. <http://atl.ndsu.edu/projects/>.
13. Riley, B., Taylor, L., and Sansone, M. (2004). MUDs and MOOs: Collaborative Narrative Play at Work in the Classroom, retrieved July 22, 2006 from http://www.nwe.ufl.edu/~ltaylor/temp/history.html.
14. Fox, K. (2004). MOO-Cows FAQ, retrieved July 22, 2006 from http://www.moo.mud.org/moo-faq.
15. Unkel, Chris (2000). WinMOO. http://www.stanford.edu/~cunkel/winmoo.html
16. McCallum, R., McGrath, O., and Rusch, J. (1997). Instructional Applications of Internet Technology: Teacher Training in the Language and Literacy MOO, retrieved July 22, 2006 from http://www.readingonline.org/electronic/moo.
17. Wilson, A. 2002 TkMOOlite for Unix, Windows, and Macintosh. http://www.awns.com/tkMOO-light/. Accessed 29July 2006
18. Haynes, C. and Holmevik, J. R. (2001). Eds. High Wired: On the Design, Use, and Theory of Educational MOOs. Ann Arbor, MI: The University of Michigan Press.
19. Bruckman, A (1997). MOOSE Crossing: Construction, Community, and Learning in a Networked Virtual World for Kids, PhD Thesis, MIT, http://www.media.mit.edu/~asb/thesis.
20. Holmevik, J. R. (2001). The enCore Advantage: An Introduction with Examples. V 1.0, retrieved July 22, 2006 from http://scholar.google.com/url?sa=U&q=http://www2.scedu.unibo.it/roversi/FormNet/enCore1.pdf.
21. Jensen, Carlos. "MediaMOO." 2001. Georgia Institute of Technology <http://www-static.cc.gatech.edu/~asb/MediaMoo/>.
22. SRI International. "About Tapped In." 2006. <http://tappedin.org/tappedin/web/about.jsp>.
23. BioMOO, The. 1996. VMDL/MOO Report. <http://tecfa.unige.ch/moo/VMDL/VMDL-26.html#HEADING26-43>.
24. Barab, S.A., K.D. Squire, & W. Dueber, 2000 A co-evolutionary model for supporting the emergence of authenticity, Educational Technology Research and Development, 48(2), 37-62.
25. Herrington, J., R. Oliver, & T.C. Reeves 2003 Patterns of engagement in authentic online learning

environments, Australian Journal of Educational Technology 19(1):59-71.
26. Mack, J., Slator, B. M., Boulais, K., Doan, V., Stanford, C., and the students of CS345. Learning by Earning: The Dollar Bay Project Proceedings of the 36th Annual Midwest Instruction and Computing Symposium (MICS-01), Duluth, MN. April 5-7. 2003
27. Slator, Brian M., Richard Beckwith, Lisa Brandt, Harold Chaput, Jeffrey T. Clark, Lisa M. Daniels, Curt Hill, Phil McClean, John Opgrande, Bernhardt Saini-Eidukat, Donald P. Schwert, Bradley Vender, Alan R. White. (2006). Electric Worlds in the Classroom: Teaching and Learning with Role-Based Computer Games. New York: Teachers College Press. Columbia University. 192 pages.
28. Slator, B. M., Farooque, G. The Agents in an Agent-based Economic Simulation Model. Proc. of the 11th International Conf. on Computer Applications in Industry And Engineering (CAINE-98) November 11-13, 1998, Las Vegas, Nevada USA. 1998
29. Regan, P. M., Slator, B. M. Case-based Tutoring in Virtual Education Environments. Proceeding of the ACM Conference on Collaborative Virtual Environments (CVE-2002). Bonn, Germany. September 30 - October 2, pp. 2-9. 2002
30. Hill, Curt and Brian M. Slator (2000). Computer Science Instruction in a Virtual World. World Conference on Educational Media, Hypermedia and Telecommunications (ED-MEDIA 2000), June 26-July 1, Montreal, Quebec, Canada.
31. Kafai, Yasmin B. 2006 Playing and Making Games for Learning: Instructionist and Constructionist Perspectives for Game Studies. *Games and Culture* 1(1):36-40. Thousand Oaks, CA: Sage Publications.
32. Ertmer, Peggy A., & Newby, Timothy J. 1993 Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. *Performance Improvement Quarterly 6*(4):50-72. Silver Spring, MD: International Society for Performance Improvement.
33. Fosnot, Catherine Twomey, editor. 1996 Constructivism: Theory, Perspectives, and Practice. Teachers College Press.
34. Suchting, W.A.: 1992, ‘Constructivism Deconstructed’, Science & Education, 1(3), 223-254.
35. Tobin, K., ed., R.G. Good, J.H. Wandersee and J. Saint Julien. 1993 Cautionary Notes on the Appeal of the New "Ism" (Constructivism) in Science Education, The Practice of Constructivism in Science Education. Washington, DC: AAAS Press.
36. Turner, Victor 1987 The Anthropology of Performance. New York: PAJ Publications.
37. Morris, Rosalind C. 1995 Annual Review of Anthropology. pp. 567-592. William Durham, ed. CA: Annual Reviews, Inc.
38. D’Andrade, Roy G. 1981 The cultural part of cognition, Cognitive Science 5, 179-195, 186.
39. Nooteboom, Bart 1999 The triangle: Roles of the go-between, Interfirm networks: Organization and industrial competitiveness, A, Grandori, ed. Groningen University, 91-119.
40. Guimarães, Mario J.L., Jr. 2001 Investigating physical performance in cyberspace: Some notes about methods. Centre for Research into Innovation, Culture and Technology, Brunel University, Uxbridge, Middlesex, UB8 3PH.
41. Schilling-Estes, Natalie 1998 Investigating ‘self-conscious’ speech: the performance register in Ocracoke English. Language in Society 27:53-83.
42. Bauman, Richard 1993 Disclaimers of Performance. In Hill, J. H. & Irvine, J. T. (Eds.) Responsibility and Evidence in Oral Discourse. Cambridge: Cambridge University Press.
43. Staats, Susan 2005 Mathematics Discourse as Performance: Perspectives from Linguistic Anthropology, The Mathematics and Teaching Learning Centre, Australian Catholic University, International Group for the Psychology of Mathematics Education.
44. McConachie, Bruce 2001 Doing Things with Image Schemas: The Cognitive Turn in Theatre Studies and the Problem of Experience for Historians. *Theatre Journal* 53(4):569-94.
45. McConachie, Bruce 2003 Theatre of the Oppressed with Students of Privilege: Practicing Boal in the American College Classroom. *Teaching Performance Studies*, eds. Nathan Stucky and Cynthia Wimmer, pp. 247-260. Carbondale,IL: Southern Illinois Univ. Press.
46. Spindler, George and Louise Spindler 2000 Fifty Years of Anthropology and Education, 1950-2000: A Spindler Anthology. Mahwah, New Jersey:Lawrence Erlbaum Associates.
47. Spindler, George 1955 Sociocultural and psychological processes in Menomini acculturation, U. of California Press, Publications in Culture and Society #5.
48. Rogers, E.M. 1962 The Diffusion of Innovations, The Free Press of Glencoe, Macmillan New York.

49. Brandt, Lisa, Otto Borchert, Kimberly Addicott, Bob Cosmano, Justin Hawley, Guy Hokanson, Dan Reetz, Bernhardt Saini-Eidukat, Donald P. Schwert, Brian M. Slator, Shannon Tomac (2006). Roles, Culture, and Computer Supported Collaborative Work on Planet Oit. Journal of Advanced Technology for Learning. 3(2), pp. 89-98.
50. Slavin, Robert E., 50-69 Cooperative Learning: Second Edition 1995. Allyn and Bacon. Boston, MA. pp. 50-69
51. Brown, Collins, Duguid (1989) Situated Cognition and the Culture of Learning. Educational Researcher, 18(1), 32-42
52. Kagan, S. Cooperative Learning (San Juan Capistrano, CA: Kagan Cooperative Learning, 1994).