

N. JENSEN S. SEIPEL W. NEJDL S. OLBRICH

COVASE: COLLABORATIVE VISUALIZATION FOR CONSTRUCTIVIST LEARNING

CSCL Conference 2003

Abstract. The paper specifies CoVASE, a software for teachers to create and view networked learning environments (VE). Students carry out virtual experiments in CoVASE, at the same time and from different places. They use the same tools and work on the same scientific problems as researchers do. Teachers create a motivating, demanding, and authentic interaction between learners and real-world problems, a premise for constructivist learning (CL). CoVASE generates and displays the result of a numerical simulation in parallel of its progress on distributed 3D graphic viewers, steered by users in real-time. VEs mediate communication between users, deictic elements, and display. Researchers and students have evaluated the predecessors of CoVASE with good results. A field study is planned for 2003.

1. INTRODUCTION AND OVERVIEW OF COVASE

The paper specifies a framework for scientific 3D visualization and collaborative work in the natural sciences and engineering. Completed examples are Finite Element Analysis, Computational Fluid Dynamics, and Volume Visualization. Teachers compose 3D 'learning worlds' where students carry out tasks. Instructors can evaluate new theories of learning, and software designers can adapt our design.

We use simulated experiments because their visualization adapts to different views, they externalize patterns in processes that are difficult and risky to carry out in reality, and because interactions can be repeated and recorded. Virtual Reality (VR) hardware increases perceptual authenticity, draws on tacit knowledge of operating entities, and preserves cues for perceiving multi-sensual stimuli. Our goals are:

- Develop a versatile visualization tool that scientists and students use.
- Evaluate if immersion, i.e. to deepen in a VE, positively affects learning.
- Identify VE components that foster collaborative, self-directed learning.

Our objectives in the scope of the VASE 3 project are as follows:

- Let users study with CoVASE (Figure 1), and identify system weaknesses.
- Examine how students perform with stereoscopic and monoscopic viewing.
- Add CSCL components to the VE, and examine how students use them.

CoVASE is a framework for mediating between users and virtual experiments. We define 'framework' as a network-distributed software for the exchange of data between users. The content and the user interface are specified in script files that describe the use of components. A 'component' is binary code linked at runtime (a DLL) that encapsulates data and exposes behavior by means of functions.

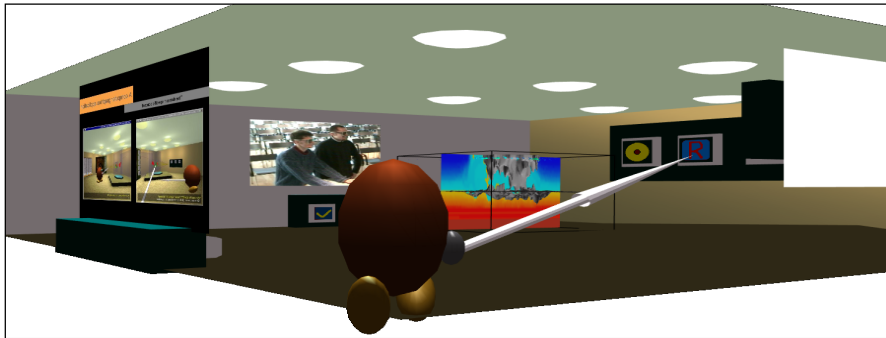


Figure 1. A synthetic 'Collaboratory' in CoVASE (Andersson, 2002; Jensen, Olbrich, Pralle & Raasch, 2002).

2. COVASE, DIDACTIC CONCEPT, RELATED WORK, AND SOFTWARE DESIGN

Users must customize CoVASE. An example is Figure 1, created and used with CoVASE. We have placed a 3D model of an oceanic convection in the middle of a virtual classroom. A slice displays low temperature in the upper part, high temperature in the lower part. More elements cover the peripheral area from right to left: a button, pointer, avatar, slide, and a video wall. Mutually aware users enter the VE together from their networked computers with mice and keyboards.

CoVASE is new because it visualizes and controls *complex* results of computer simulations in custom interfaces on remote computers. Constructivist learning (CL) with interactive expert simulations is possible (Kolb, 1984; Duffy & Jonassen, 1992). CoVASE is valuable because (i) it allows users to reuse VEs and visualizations in a structured way, and (ii) it remedies the lack of tools that support CL to facilitate students' autonomous learning in groups. CL, and therefore a CL tool, helps groups to perceive their progress in building knowledge among subjects, and to synthesize viewpoints (Scardamalia & Bereiter, 1992). CL tools foster *self-directed learning in groups* by use of *expert tools and data* that aid teachers to guide students (Duffy & Jonassen, 1992).

CSCW and CSCL tools support CL. CSCL introduces novices to a CSCW system, limits requirements of the system to make its use affordable by classes, and exposes experts' knowledge resources (Edelson, Pea & Gomez, 1996). In scientific inquiry, which is problem-driven not task-driven, we believe CSCW does not differ from CSCL except in those aforementioned points.

Why is collaborative visualization needed? McCormick, DeFanti and Brown (1987) use computer graphics to create visual metaphors to describe data related to processes and entities because users interpret graphics faster than they decode alphanumeric symbols. Collaborative visualization, in accordance to Wood, Wright and Brodlie (1997), accommodates that problem-solving by way of visualization is predominantly group-oriented, consistent with CL theory.

CoVASE: COLLABORATIVE VISUALIZATION FOR CONSTRUCTIVIST LEARNING

CoVASE implements CSCL at different places, at the same and different time, compare the time / space matrix in Baecker (1993, p. 11, Fig. 2).

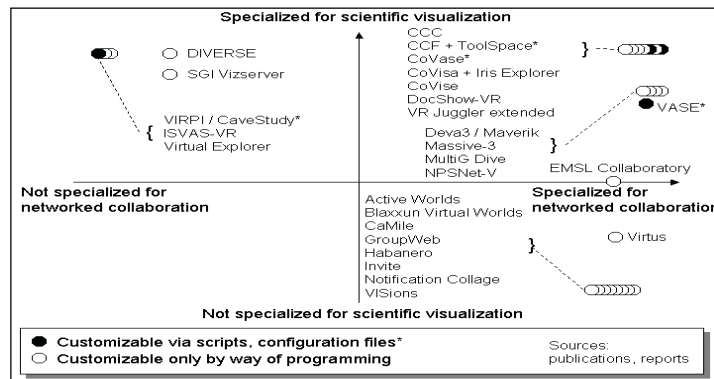


Figure 2. Classification of CSC* and visualization toolkits.

Figure 2 relates systems visually (www.learninglab.de/~jensen/ves.html specifies references). From bottom-right in clockwise order we have classified representative (i) systems that only enforce CSCW and socialization in networked VEs, (ii) unrelated systems, (iii) scientific visualization software without CSCW, and (iv) scientific visualization systems for CSCW and CSCL toolkits to manage network-distributed VEs. The latter are most similar to CoVASE but do not display time-variant complex simulation data, do not record events and replay them, or support more users than we do at the cost of making the creation of VEs difficult and prone to error (Wood, Wright & Brodlie, 1997; Benford, Greenhalgh, Reynard, Brown & Koleva, 1998). An exception is Watson (2001) but he does not specify results for the application in education, different to Edelson, Pea and Gomez (1996). Their system, CoVIS, uses Web software that does not control simulations in an interactive way, but like ours supports scaffolding, i.e. helping teachers to coach students online.

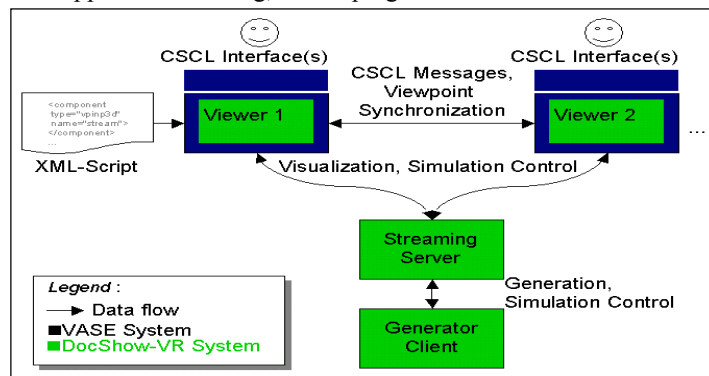


Figure 3. CoVASE diagram specifies software layers and communication paths.

CoVASE (Figure 3) combines a collaborative 3D user interface (VASE) with a networked visualization system (DocShow-VR).

DocShow-VR (Jensen, Olbrich, Pralle & Raasch, 2002) connects over TCP/IP to a simulation on a remote computer that sends 3D graphics to a streaming server for replication to each collaborative viewer. The server receives commands from viewers to control display and simulation. A Sun E450 server, for example, transfers predefined 3D sequences to a multimedia workstation at 1 mio. triangles per second over Gigabit/s network. DocShow-VR clients connect in sequence to each other.

VASE (Andersson, 2002) is a user interface and extensible messenger. An XML file contains the initialization script. We use XML because we plan to add descriptions of VEs, and develop a Document Type Descriptor (DTD) for our XML format. More sophisticated DTDs exist, but we aim for simplicity. The file holds names of components and message paths between them. Components process alphanumeric messages. For example, users press a button to start a 3D animation:

```
...<button1> DocShow-VR : play : stream </button1>...
```

```
<component type="vpinp3d" name="DocShow-VR"> </component>...
```

The message is distributed over IP as follows: each client connects to the super client, which is a client different to the streaming server and generator client. The super client holds the state of the shared VE in a repository. A client processes the message and forwards it to the super client which propagates the message to all clients. The scheme is easy to implement and accommodates that clients connect at different times. However, it is neither scalable to many viewers nor robust against failures of computers and networks.

How to use the system? A tutor writes an XML file to specify the topology and layout of a VE. The tutor sets instances of components and their attributes (for example the URL of a remote simulation). She specifies message flow between components to install 'cause-and-action' chains in the VE. She publishes the XML file, and starts CoVASE and the video conferencing hardware. The VE displays video images of students. After briefing, they play and navigate the visualization by use of a virtual control panel to move avatars, point at models, and annotate parts. Students negotiate their findings and consult their remote supervisors.

3. DISCUSSION AND CONCLUSION

Most of CoVASE is finished: exchange of messages (and events), virtual control panels, pointers, avatars, slides, video walls, whiteboards, panels for text chat, 2D desktops, rooms, and static 3D models. We will add video conferencing via web cameras, stereoscopic viewing by use of VR hardware, and embed DocShow-VR's facilities to simulate, record, and replay 3D sequences in CoVASE.

Informal pilot studies indicate the value of our concept. DocShow-VR, CoVASE's component for experimenting with simulations, transfers 3D models in an efficient way (Jensen, Olbrich, Pralle & Raasch, 2002, p. 130). VASE, CoVASE's CSCL interface, is usable. VASE users like the software and consider experiments valuable (Seipel & Lindkvist, 2001). They talk in the VE by several means, and 'walk' and point at objects in the VE with two degrees of freedom with few errors.

CoVASE: COLLABORATIVE VISUALIZATION FOR CONSTRUCTIVIST LEARNING

In summary, the predecessors of CoVASE support similar applications and have complementary strengths. We combine functions that are orthogonal to each other to improve system performance, usability, and customizability. Some problems are unsolved. Users edit XML files manually, and they have no guidelines to build VEs. The maximum of synchronous users is unknown. We have neither proven that CoVASE is useful, nor analyzed the use of the system in a formal way. We will carry out a controlled field study between June and July 2003 to quantify results.

REFERENCES

- Andersson, B. (2002). *VASE - A functional framework for flexible configuration of virtual and distributed teaching environments* (unpublished master thesis). Uppsala, Sweden: Uppsala University.
- Baecker, R. (1993). *Readings in groupware and computer supported cooperative work*. San Francisco: Morgan Kaufmann.
- Benford, S., Greenhalgh, C., Reynard, G., Brown, C., & Koleva, B. (1998). Understanding and constructing shared spaces with mixed-reality boundaries. *ACM Transactions on computer-human interaction (TOCHI)*, 5(3), 185-223.
- Duffy, T., & Jonassen, D. (Eds.). (1992). *Constructivism and the technology of instruction: A conversation*. Hillsdale, NJ: Erlbaum.
- Edelson, D., Pea, R., & Gomez, L. (1996). Constructivism in the collaboratory. In B. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design* (pp. 151-164). Englewood Cliffs, NJ: Educational Technology Publications.
- Jensen, N., Olbrich, S., Pralle, H., & Raasch, S. (2002). An efficient system for collaboration in tele-immersive environments. In D. Bartz, E. Pueyo, & E. Reinhard (Eds.), *Parallel graphics & visualization 2002: Eurographics workshop proc.* (pp. 123-131). New York, NY: ACM Press.
- Kolb, D. (1984). *Experiential learning: Experience as the source of learning and development*. New Jersey: Prentice Hall.
- McCormick, B., DeFanti, T., & Brown, M. (1987). Visualization in scientific computing. *Computer graphics*, 21, 1-14.
- Scardamalia, M., & Bereiter, C. (1992). An architecture for collaborative knowledge building. In E. De corte, M. Linn, & H. Mandl (Eds.), *Computer based learning environments and problem solving*, (pp. 41-46). Berlin: Springer.
- Seipel, S., & Lindkvist, M. (2001). Simulation based collaborative experiments for interactive learning in virtual classroom environments. *Proc. 2nd european conference on e-learning WBLE* (pp. 347-355). Lund, Sweden.
- Watson, V. (2001). Supporting scientific analysis within collaborative problem solving environments. *Proc. 34th annual hawaii international conference on system sciences (HICSS-34)* (9 pp.). Maui, Hawaii: IEEE Computer Society Press.
- Wood, J., Wright, H., & Brodrie, K. (1997). Collaborative visualization. In R. Yagel, & H. Hagen (Eds.), *Proc. IEEE visualization 1997* (pp. 253-259). New York, NY: ACM Press.

AFFILIATIONS

Wolfgang Nejdl is professor for Knowledge Based Systems, University of Hannover, Germany, and director of Learning Lab Lower Saxony [L3S].

Stefan Seipel is professor for Computer Graphics at Uppsala University and University of Gävle, Sweden. He has been principle investigator in the Swedish Learning Lab projects CVEL and VASE.

Stephan Olbrich leads the Visualization group of the Regional Computing Center for Lower Saxony (RRZN / Vis), University of Hannover.

Nils Jensen is research associate at [L3S] and RRZN / Vis.