A Collaborative Learning Environment for Data Modelling

Adrian Gordon, Lynne Hall

Department of Computing
University of Northumbria at Newcastle
Newcastle-Upon-Tyne, NE1 8ST, UK
email: {adrian.gordon. lynne.hall}@unn.ac.uk

Abstract

Data modelling is an important skill for students undertaking university courses in computing and Information Systems. It is a complex task which is not currently well provided with tool based support. This paper describes extensions to a learning environment for data modelling which is based on a text based virtual reality paradigm. This learning environment has been enhanced by the inclusion of autonomous, intelligent agents of two types. One type of agent basically has a pedagogical role, providing feedback to learners about the quality of their data models. The other type of agent effectively plays the role of a learner within the environment, and is intended to offer the opportunity for collaboration to human learners within the environment.

Introduction[1]

One of the most commonly taught data modelling formalisms in computing education is the Entity Relationship Model (Chen, 1976) and its extensions (Teorey, Yang & Fry. 1986). The Entity Relationship (ER) model is also widely used for software development in industry. As with many areas of systems analysis and design, ER modelling can only be learnt practically (McLeod, 1996). Such practical, experiential work in ER modelling is usually provided through the analysis of text based scenarios, from which the learner has to derive a model by undertaking a sequence of tasks.

ER modelling is complex. It involves the identification of salient facts from disparate information sources, many of which are text based. Novices find this task difficult and exhibit systematic errors in their models (Hall and Gordon, 1998). In learning ER modelling, learners need to have a conceptual representational tool (a notation) and a methodology for using that tool (Batra and Davis, 1992). Learners do not experience problems in using the notation (i.e. physically drawing the model), instead, errors are mostly made in relation to the application of an appropriate methodology (Hoggarth and Lockyer, 1996).

The problems of ER modelling are compounded by a widely observed anchoring heuristic (Batra and Antony, 1994), whereby learners become reluctant to review any parts of a model that they have already constructed. Frequently, learners anchor to an incorrect model. Without appropriate and timely feedback, a learner will not be aware that they have constructed an incorrect model.

In real life ER modelling is a collaborative activity, with multiple analysts gathering information from multiple users / stakeholders each of whom may have a different view of the proposed system. Analysts then integrate this information to produce the final ER model. As far as we are aware, no attempts have yet been made to support this essentially collaborative aspect of the ER modelling task for novice ER modellers.

The problem of supporting collaborative ER modelling activities in a learning environment is complicated by the changing nature of the higher education student. Rather than the 20 year old direct entrant living on campus with few family responsibilities, there is a growing requirement to support mature students who may study in a part time or distance learning mode (Jelly, 1997).

The Entity Relationship Modelling Virtual Learning Environment (ERM-VLE), described in this paper was developed to meet these challenges. ERM-VLE provides a learning environment in which learners can gain extensive experience in solving ER modelling problems. It provides learners with timely and appropriate feedback about the quality of their models as they are being constructed. ERM-VLE also supports collaborative learning, and is available to support students outside of the traditional classroom environment.

Earlier work described in (Hall and Gordon, 1998) details our first implementation of ERM-VLE. This paper describes recent enhancements to ERM-VLE. Most notably, it describes the extension of this environment by the inclusion of a number of different types of intelligent agents.

The remainder of this paper is organised as follows. In the first section we detail the structure of our learning environment. The second section describes how ERM-VLE can be used to support collaborative learning of ER modelling. The third section describes the different types of agents which can populate ERM-VLE. These agents are basically of two types: Pedagogical Agents, which provide feedback to learners about the quality of their models; and Pseudo Learner Agents, which are intended to collaborate with learners in problem solving. The final section provides some discussion of ERM-VLE, and details of proposed future work.

Entity Relationship Modelling Virtual Learning Environment (ERM-VLE)

In ERM-VLE, the task of the learner(s) is to study a textual scenario (typically a description of a business area) and construct an ER model which correctly reflects the data requirements described in that scenario. ERM-VLE is based on a text based virtual reality, the most common examples of which are seen within the Multi User Dimension (MUD) paradigm. The past decade has seen the use of MUDs for a variety of educational purposes, see for example, (Fanderclai, 1995), (Hughes & Walters, 1995).

The virtual space in ERM-VLE, which represents both the scenario and the ER modelling task, is realised as a collection of locations or 'rooms' connected to one another through the use of exits / doors in a virtual space. A number of different types of rooms exist, such as Scenario Rooms, where the various elements of the scenario are distributed; and model construction rooms, where the model is built. For example, Entity Creation rooms are locations in which the creation of new entities is permitted, by manipulation of appropriate elements from the textual scenario. To interact with the world, a restricted set of commands is provided, relating to movement, object manipulation, and communication, as described in (Hall and Gordon, 1998).

A key issue within the ER modelling task is the need for learners to adopt an appropriate task structure. For example, to limit the potential complexity of an ER model it is important to identify all entities within the scenario, prior to determining the relationships amongst them. The topological organisation of the virtual world effectively imposes this task structure on the learner, simply by restricting the navigational opportunities offered to her. Learners are simply not *allowed* to enter Relationship Creation rooms until all appropriate entities have been identified.

ERM-VLE is built using a client/server architecture. The VLE server system, in which the virtual world resides, is implemented in CLIPS (NASA STB, 1997), a rule-based Expert System programming language. The VLE client systems are implemented in Tcl/Tk, a script based language for building graphical user interfaces (Ousterhout, 1994). The client systems communicate with the server system by sending text commands, and the server system communicates with the client systems by sending Tcl/Tk scripts, which are then evaluated by the relevant clients. The TCP/IP communication protocol is used, so client processes can be either local or remote.

Although communication between client and server systems is predominantly text based, the client interface can also contain graphical elements. A graphical representation of the ER model being constructed is presented to the learner, for example.

Collaboration in ERM-VLE

The earliest iteration of ERM-VLE was effectively a single user environment, in that learners existed in the ERM-VLE in separate, though parallel, virtual spaces. However, learners could, and would communicate with one another, *across* virtual spaces. This communication resulted in learners essentially undertaking collaborative problem solving. It was decided to augment the environment to exploit the potential for enhancing learning offered by this collaborative activity.

In order to achieve this, ERM-VLE was reconfigured to enable more than one learner to be in the same virtual environment at the same time. Rooms within the multi-learner environment were of two types: common areas to which all learners had access; and private areas to which only a single user had access. Various parts of the scenario were distributed in these different rooms.

During a learning session, learners could be divided into groups. The group task would be to create an ER model for the whole scenario, that is each member would have to create their own ER model. However, due to the separation of the scenario learners would be unable to gain a view of the entire scenario unless they collaborated with their colleagues.

Learners interact with the ERM-VLE and communicate with one another, sending and receiving what we term 'faxes', which are essentially copies of elements from the client interface of each learner. Learners could, for example, fax a copy of the diagrammatic representation of their current ER model to other learners. Other learners could not directly extract model elements from these faxes, they had to build their own copy of the relevant ER model themselves. This was achieved through learners transporting certain elements of the scenario from their private areas into the common areas for their colleagues. Additionally, by using the free style natural language 'internet chat' facility included in ERM-VLE, they could, for example, ask one another for explanations of these faxes, ask one another for advice and guidance, and ask the whereabouts of various scenario elements. In this way ERM-VLE supports both pedagogical and peer learning opportunities.

Agents in ERM-VLE

One of the principal constraints on the development and deployment of Intelligent Tutoring Systems (ITSs) lies in the complexity of constructing such systems. Knowledge acquisition, knowledge encoding and system architecture design are extremely difficult tasks. This is exacerbated by the fact that most ITSs are one-off tailored applications which are difficult if not impossible to adapt to new domains. The use of autonomous intelligent agents offers a potential solution to this problem. As noted in (Atkins et al. 1996) distributing tasks to numerous specialised agents promotes modularity, flexibility and incrementality.

We are currently extending ERM-VLE by populating it with such autonomous intelligent agents. These agents are of two types, Pedagogical Agents and Pseudo Learner Agents.

Pedagogical Agents

The use of coarse grained agents within learning environments has recently received increased interest, with a number of different applications emerging where such agents are used to provide a variety of functions to learners. In MEMOLAB (Dillenbourg, Mendelsohn and Schneider, 1994) the agents operate as tutors or teachers, with the agents attempting to determine the preferred teaching style for learners in the acquisition of basic methodological skills in experimental psychology. In the GRACILE project (Ayala and Yano, 1994), a number of different agents exist to aid students in learning Japanese. The agents within GRACILE are of different types: mediator agents act as facilitators, supporting communication and collaboration among learners: and domain agents have knowledge about sentence construction and language usage.

In ERM-VLE we have included a coarse grained, domain specific pedagogical agent which provides feedback to users, only allowing them to take correct steps on their search for a solution. In this respect, it resembles the agents of (Ayala and Yano, 1994). This agent is referred to as the Gatekeeper agent, and it provides immediate feedback to learners in relation to their model and only permits learners to create correct models, thus preventing learners from ever anchoring to an incorrect solution. This agent provides limited support for the learning of declarative knowledge and considerable support for the learning of the procedural knowledge which we consider to be the task structure or the development of an appropriate methodology. The Gatekeeper navigates around the virtual world controlling the access of learners to locations and allowing or denying learners the use of particular commands.

For example, when the learner is in the process of entity creation, the Gatekeeper Agent has two main functions. Firstly, to ensure that the learner continues creating entities until all of the entities necessary for the model have been identified. Secondly, accepting or rejecting the learner's choice of entities within the Entity Creation rooms. Within these rooms the learner attempts to create and name entities. The Gatekeeper Agent only permits learners to create correct entities.

Due to the relative simplicity of the scenario there is single correct solution in terms of which of the scenario objects can be used for each of the model constructs. So, if a learner attempts to construct an entity from an object which does not represent an entity in the solution to the problem at hand the learner will be given negative feedback, and told that the item does not represent an entity. Similarly, if the learner tries to enter a Relationship Creation room she is told that she may not enter this area until all of the entities have been created.

Pseudo Learner Agents in ERM-VLE

One of our aims for ERM-VLE was to ensure maximum availability of the system outside of the traditional classroom setting. This meant that in order to allow collaborative learning, we needed to ensure that other actors were present at all times within ERM-VLE. Our proposed solution to this problem is to populate ERM-VLE with autonomous fine-grained intelligent agents, which we have called Pseudo-Learner (PL) agents.

The fine grained PL agents in our environment replicate some of the behaviour of novice ER modellers. The PL agents do not have expert knowledge related to ER model construction. They behave in a similar way to the human learners in the environment. The key characteristics of the PL agents are that they are architecturally simple and homogeneous, with each of the PL agents being identical to all others; they do not have any internal representation of the virtual world; they follow simple patterns of behaviour which can be easily programmed; and they embody the concept of cognitive economy, and as such are similar to the reactive agents of (Ferber and Drogoul, 1992)

Although the PL agents are fine grained, they do have some basic knowledge which differentiates them from the human learners (at least at their most novice stage) This knowledge is provided through some simple rules which can be applied to ER modelling. For example, one of the most basic of these rules is that an entity will be derived from a noun-phrase. Therefore, when a PL agent is in the process of entity creation it will ignore objects which are anything other than noun-phrases. However, in the same way as human learners, they do not know *a priori* which of the available noun-phrases really do represent entities within the scenario. Thus, similar to human learners they must follow the structure imposed by the virtual world, under the guidance of the feedback supplied by the Gatekeeper agent.

The use of companion agents in addition to teaching agents within a learning environment is described in a number of applications. In (Hietala and Niemirepo, 1996), EduAgents provide the teaching function within the domain of equation solving, whilst companion agents collaborate with the learner in the determination of a solution. However, a significant problem has been identified in the situation where the learner knows that their companion is a computerised collaborative learner rather than a human agent. Essentially, the learner looks to this agent not as a collaborative companion, but rather as an assistant teacher or tutor. (Hietala and Niemirepo. 1996) note that learners expected the companion agent to produce viable, correct solutions, and if they did not the learner collaborated with them less.

Similar results were found in Dillenbourg's (1994) People Power system. Dillenbourg notes that if a co-learner is not improving its suggestions quickly enough then subjects lose their motivation to collaborate. Thus, although companion agents enrich the learning situation by taking an active part in the session, learners primarily wish for such agents to provide a teaching function.

However, these findings are not duplicated in environments where learners are not aware of the computational nature of their collaborators. This use of agents which simulate humans within virtual worlds can be seen in applications such as the Soar/IFOR project (Tambe et al. 1994) and VET (Virtual Environments for Training) (Johnson, 1995). In both of these applications, human-like intelligent agents interact both with one another and with humans in the performance of tasks. These agents participate in activities, additionally they can explain and demonstrate how to perform particular tasks, and offer advice and assistance.

Although human learners collaborating in ERM-VLE have a full range of communications available to them, it is not possible to support the latter in the Agent ERM-VLE system (ERM-VLE plus PL agents). Thus, the communication possibilities have been reduced to a minimum. Essentially, the only communications allowed to human learners and cybernetic agents in Agent ERM-VLE are faxes (of the agent or learner's current model, inventory, or scenario), requests for faxes and transportation of scenario elements to shared areas. This applies both to human-human and human-agent communication. We are currently investigating the effects of these limited communication possibilities on the effectiveness of learning in Agent ERM-VLE.

Discussion

Although preliminary results with PL-Agent ERM-VLE have been encouraging, a number of issues have emerged which require consideration and a further iteration of ERM-VLE.

At present the Gatekeeper Agent provides only simple, negative feedback to learners. For example, if learners attempted to create an entity from an object which did not represent an entity, they were simply told that the object was not an entity, with no explanation as to why it was not. Although negative feedback appears to be more successful than no feedback at all, and seems to mitigate the effects of the anchoring heuristic, we feel that it is too limited.

We feel that there is considerable scope for improving the quality of feedback, including explanations, that can be provided by the Gatekeeper Agent. Our agent based architecture should allow us to easily incorporate improved pedagogical agents, or even sets of collaborating pedagogical agents, into our learning environment.

A feature that ERM-VLE shares with Microworlds (White and Horwitz, 1988) is that pedagogical activity is very much focused on procedural knowledge - on how a task should be performed. ERM-VLE does not currently offer the learner much support in learning the concepts underlying its task domain - what does or should an "entity" represent, what is a "noun phrase," what is a "customer" (where customer represents an entity), etc. We are currently looking at ways of addressing this problem.

A Tcl/Tk web browser plug in has recently been released and we are hoping that this may enable us to incorporate ERM-VLE into a website that would provide conventional computer based learning support for the underlying concepts of the ER modelling domain.

At present, many of the inadequacies of ERM-VLE are compensated for by the collaborative nature of the system. Currently, if a learner fails to understand something that occurs within ERM-VLE they may be able to consult their colleagues, or even their instructor (who may also be present within the VLE) for explanations.

The limited communication possibilities available to the PL agents obviously reduces their utility for providing this kind of support. We aim to extend their abilities to enable them to have more detailed interactions with learners, although clearly a restricted command form will still have to be used.

ERM-VLE appears to provide a supportive learning environment for practising activities traditionally learnt in a classroom environment, without incurring the high costs associated with on-line pedagogic activity (Albright and Graf, 1992). Even with the current limited Gatekeeper Agent, the ER modelling task has changed from a class based to a student based activity. Learning has become a flexible, independent activity, without imposing additional correction and feedback tasks on staff. We hope that the inclusion of simulated human learners within the environment will enable us to provide an environment in which our learners can learn in collaborative groups at any time they choose.

References

Albright. M.J.: and Graf. D.L. (eds) 1992. Teaching in the Information Age: The Role of Educational Technology (introduction). Jossey-Bass Inc.

Atkins, D.E.: Birmingham. W.P.: Durfee. E.H.; Glover. E.J.: Mullen. T.; Rundensteiner, E.A.; Soloway, E.: Vidal. J-M.; Wallace, R.: and Wellman. M.P. 1996. Toward Inquiry-Based Education Through Interacting Software Agents, in *Computer*: May

Ayala. G.; and Yano. Y. 1994. Design Issues in a Collaborative Intelligent Learning Environment for Japanese Language Patterns. In the Proceeding of ED-MEDIA 94, AACE: 67-72

Batra. D.: and Antony. S.R. 1994. Novice Errors in Conceptual Database Design. *European Journal of Information Systems* 3(1):57-69.

Batra, D.: and Davis. J. 1992. Conceptual data modelling in database design: similarities and differences between expert and novice designers. *The International Journal of Man-Machine Studies* 37(3):395-417.

Chen. P.P. 1976. The Entity Relationship Model - Toward a Unified View of Data, *ACM Trans. Database Systems* 1(1):9-36.

Dillenbourg, P. 1994. Internalization and Learning Environments, in Vosniadou, S.: De Corte, E; and Mandl, H. (eds) *Technology-Based Learning Environments*. Berlin: Springer-Verlag: 48-54.

Dillenbourg, P.; Mendelsohn, P.; and Schneider, D. 1994. The distribution of pedagogical roles in a multi-agent learning environment. in Lewis, R. & Mendelsohn. P., (eds), *Lessons from Learning*. Amsterdam: North-Holland, (1994).

Fanderclai. T.L.. 1995, MUDs in Education: New Environments. New Pedagogues, CMC Magazine 2(1).

Ferber. J.; and Drogoul. A. 1992. Using Reactive Multi-Agent Systems in Simulation and Problem Solving. In Avouris, N.: and Gasser, L. (eds), *Distributed Artificial Intelligence: Theory and Praxis*. Kluwer Academic Publishers.

Hall. L.; and Gordon. A. 1998. A Virtual Learning Environment for Entity Relationship Modelling. In Proceedings of the ACM SIGCSE Conference '98: Association for Computing Machinery.

Hietala. P.: and Niemirepo. T. 1996. Studying learner-computer interaction in agent-based social learning environments. In Brna, P.. Paiva, A. & Self. J. (eds) Proceedings European Conference on Artificial Intelligence in Education.

Hoggarth. G.R.; and Lockyer, M.A. 1996. Systems development methods guidance and CASE: integration between CASE and CAL, Software Engineering Journal, May:141-147.

Hughes. B. and Walters, J., 1995 Children, MUDs, and Learning at:
http://vmsgopher.cua.edu:70/0gopher_root_eric_ae%3a[_edir.aera.95. paps]5005bh.asc

Jelly, I. 1997. Whom are we teaching? Computer 30(11):49-50.

Johnson. W.L. 1995. Pedagogical Agents in Virtual Learning Environments. In Jonassen, D. & McCalla. G. (Eds) *Proceedings International Conference on Computers in Education*, 41-48.

McLeod Jr, R. 1996. Comparing Undergraduate Courses in Systems Analysis and Design. *Communications of the ACM* 39(5): 113 - 121

NASA STB, 1997. *CLIPS Reference Manual*. NASA Software Technology Branch. Lyndon B. Johnson Space Center.

Ousterhout. J.K. 1996. *Tcl and the Tk Toolkit*. Reading:Mass.: Addison-Wesley.

Tambe. M.: Johnson, W.L.; Jones, R.M.: Koss, F.; Rosenbloom, P.S.: and Schwamb, K. 1995. Intelligent Agents for interactive simulation environments. *AI Magazine*. 16 (1):15-39

Teorey. T.J.; Yang, D.; and Fry, J.P. 1986. A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model. *Computing Surveys* 18(12):197-222.