Collaborative Virtual Learning Model for Web Intelligence

Jinan A.W. Fiaidhi and Sabah M.A. Mohammed

Department of Computer Science, Lakehead University,
Thunder Bay, Ontario P7B 5E1, Canada
{jinan.fiaidhi, sabah.mohammed}@lakeheadu.ca

Abstract. The integration of Learning Objects Repositories, Information Visualization, Web and new Visual Interaction techniques will change and expand the paradigms of current work of learners on the Web. Virtual learning will improve visual communication that takes place in all elements of the user collaboration and provide decreased "time-to-enlightenment". Virtual learning is a process that provides information visualization technology to address the challenge of discovering and exploiting information for the purpose of learning. This article examines the issue faced by most eLearning systems - how to turn data into understandable learning knowledge, and make this knowledge accessible to peers who rely on it. It introduces a generic design model for Collaborative Virtual Learning based on the Model-View-Controller design pattern.

1 Introduction

Web Intelligence (WI) was first explicitly introduced in 2000 as a joint research effort in developing the next generation Web-based intelligent systems, through combining expertise in Intelligent Agents, Data-Mining, Information Retrieval, and Logic [1]. Broadly speaking, WI encompasses the scientific research and development that explores the fundamental roles as well as practical impacts of Artificial Intelligence (AI), such as autonomous agents and multi-agent systems, machine learning, data-mining, and soft-computing, as well as advanced Information Technology (IT), such as wireless networks, grid computing, ubiquitous agents, and social networks, on Web-empowered systems, services, and activities. WI is aimed at producing new theories and technologies that will enable us to optimally utilize the global connectivity, as offered by the Web infrastructure, in life, work, and play. As more detailed blueprints and issues of Web Intelligence (WI) were evolved and specified in recent years, numerous WI research studies and business enterprises have been established around the world. WI companies and research centers/labs have been launched in USA, Canada, Europe, Japan, and India, etc. Each of them focuses on certain specific WI issues or products/services. As a result, today WI has become a well-defined IT research field, publicly recognized as well as promoted by the IEEE Computer Society.

However, much of the WI research has been focused on mining the log file of a Web site for knowledge about how Web pages are linked to each other and how frequently links have been traversed by users. Indeed, finding the online Web pages/services perfectly suited for a given task is not always feasible even with the aid of intelligent

W. Shen et al. (Eds.): CSCWD 2005, LNCS 3865, pp. 563 – 572, 2006.

agents. In this article we will describe a framework for building flexible collaboration that handles these imperfect situations. In this framework we exploit an information retrieval system as a general discovery tool to assist finding and pruning information.

2 The Issue of Web Intelligence Learning: The Missing Element

In almost all areas of science, building computing models for prediction are considered an important technique, which has been mainly promoted by the research of machine learning and data mining. Since the 1987s, *learning from examples* has been regarded as the most promising direction [2], in which the prediction model, or learner, is trained from examples whose desired output is known. In most cases, obtaining a big training set is good news. However, although huge volume data exist in the Internet, it is not feasible to collect them together to train a global learner. One reason is these data are scattered on different sites, the style of which is different from that of distributed systems where the resources are deliberately distributed. Moreover, the huge volume of data disables any process collect them together because the overwhelming cost of communication will be a disaster. Even if these data could be exchanged, new data may be difficult to be utilized because they are accumulated on different sites every time.

Actually making use of data resources in the Internet has been investigated by the communities of web intelligence. There are many fields which contribute to Web Intelligence as defined by the Web Intelligence Consortium (http://wi-consortium.org/ aboutwi.html). Although WI incorporates fields like Web Information Retrieval, Web Mining, Web Agents, Ubiquitous Computing and Social Networks, it focuses mainly on Web Knowledge Management. Although social networking has been stated as one of WI central targets, it has not mainly been targeted towards learning. When learning is concerned, WI puts weight on Learning by Example and the use of Web agents. In this direction, the focus is to use learning agents that can adapt to its user's likes and dislikes. A learning agent can recognize situations it has been in before and improve its performance based on prior experience. The ultimate goal for intelligent agents is have them learn as they perform tasks for the user. Indeed this perspective to learning is very restrictive as it ignores the social factors in learning. People learn from each other and with each other. Research shows that talking with others about ideas and work is fundamental to learning. Thus, it is among our social responsibilities to include explaining things to others, and that leads to learning. In this perspective, learning is a social and interpretive activity in which multiple members collaboratively construct explanations and understandings of materials, artefacts, and phenomena within their environment [3]. It is the result of active engagement in and with the world coupled with reflections upon the relationship between ideas, actions, and outcomes. As such, learning-as-interpretation is deeply embedded in all activity, and experiences are part of a socially embedded active and re-active process. Collaborative activity presents an opportunity for reflection and interpretation of events by providing a shared context for the interpretation of individual experience. Interpretations evolve around artefacts and narratives [4], and experiences take on meaning within communities of practice [5]. In the Web supported collaborative work, a collaboration process is led by four sequential processes [6]; co-presence, *awareness*, communication, and collaboration. Co-presence gives the feeling that the user is in a shared workspace with someone at the same time.

Awareness is a process where users recognize each others activities on the premise of co-presence. In the communication process, the users can exchange messages. In the final process, the user collaborates on the specific task with other users and accomplishes the task and common goals. Thus, in a Web collaborative learning setting, learners have the opportunity to converse with peers, present and defend ideas, exchange diverse beliefs, question other conceptual frameworks, and to be actively engaged.

3 Collaborative Learning Environments

There are many environments that can be used for collaborative learning on the Web (e.g. Blackboard, WebCT, WebFuse, CoSE, TopClass, WebEx, VNC, SCORM, and Tango). However, all such environments concentrate on providing communication between participants and tools to facilitate collaborative activities such as shared whiteboards and shared applications. As the use of collaborative environments becomes more ubiquitous and virtual, we can expect many of the same problems facing colleagues physically meeting together to arise in cyberspace. The use of ubiquitous computing will help more in the organization and mediation of interactions wherever and whenever these situations might occur. Moreover, awareness is also another missing factor for effective collaborative learning within the traditional collaborative environments. Awareness context is used to ensure that individual contributions are relevant to the distributed group's activity as a whole, and to evaluate individual actions with respect to group goals and progress. The information, then, allows groups to manage the process of collaborative working. Although there are some collaborative environments for supporting some level of awareness (e.g. VideoWindow [7] and CRUISER [8], Portholes [9] and VENUS [10], none of these systems are very useful for the user in understanding the activities of others in ubiquitous places, they have not yet provided awareness for inducing collaboration in a shared knowledge space in a ubiquitous collaborative learning situation [11].

4 Collaborative Virtual Learning

Collaborative virtual learning (CVL) environments traditionally were studied in classroom-based environment at first for tasks such as industrial team training, collaborative design and engineering, and multiplayer games. Moreover, much work in the area of enabling effective collaboration in CVLs has focused on developing the virtual reality metaphor to the point where it attempts to completely mimic collaboration in real environments [12]. In particular, much attention has been paid to user embodiment [13]. However, much more recent work was focused on Web-Based CLEs [14]. Web-based CVL systems can be divided into two categories, one is *asynchronous* system, and another is *synchronous*, which many practical systems were developed. The influential asynchronous system includes First Class, CSILE/Knowledge Forum, Learning Space, WebBoard, and WebCT; synchronous system includes Conference MOOS, WebChat Broadcasting System, and Microsoft Netmeeting. Although the above mentioned CVL research focused on interactive instructional visualization, not much of the research work focus on ubiquity and awareness. The main two research directions for the current CVL are based on developing Open Reusable Components and having Virtual Learning Objects. In fact there are only very few research attempts which can be cited in the literature that address CVL as CB reusable systems (e.g. *multibook* CVL of the Technical University of Darmstadt [15], the WebDAV-Collaborative Desk of the Institute of Telematics [16], and *JASMINE* [17], as well as the *Java Multimedia Telecollaboration* [18]. But the issue of complying with a learning standard remains largely to be answered by variety of systems. The majority of work concerned with learning objects standards has involved on what is called "the knowledge engineering of eLearning." An international standard has been achieved for learning object metadata (see [19] & [20]), and a robust specification has been developed for content packaging [21], along with a list of related initiatives [22]. The only serious attempt for implementing such specifications came from [23] in their work entitled "Smart Multimedia Learning Objects. However, their implementation model does not support ubiquity and awareness. Such integration will change and expand the paradigms of the current work of learners on the Web [24]. We consider the process of CVL as the main framework that provides information visualization intelligence to address the challenge of discovering and exploiting information for the purpose of learning. In this direction, we believe that by having a generic model for such process is very important. The next section introduces a model for CVL based on the Model-View-Controller design pattern, which solves the Web visualization intelligence problem by decoupling data access, collaboration and business logic, and data presentation and user interaction.

5 An MVC Model for Collaborative Virtual Learning

Since collaboration is a central process in our proposed virtual learning model, we need to establish a framework for controlling events and signals on a common event bus [2]. Web-services and peer-to-peer platforms seem to be the best candidates for this framework since it can run across various platforms and is easy to be extended and understood. As components increasingly are designed to be accessed over the Internet and its ubiquitous devices, it becomes more and more important that component technologies have the openness, and use the protocols, that make up Internet infrastructure. For this reason, XML messaging is emerging as an important component technology. On one hand, there are many systems that use XML as their media of communication between peers enabling Text Chat, Instant Messenger, and White boards including sharing multimedia resources (e.g. Jaber, NaradaBrokering, JXTA). On the other hand, there is no unifying model that can be used to represent collaboration and to gear all these protocols and infrastructures to successfully aid the resource sharing and organize collaboration. For this purpose we are proposing a modeling framework that can be used to integrate all these technologies for the purpose of developing effective CVL environments. The proposed framework is based on the Model-View-Controller (MVC) design pattern which is often used by applications that need the ability to maintain multiple views of the same data. The MVC pattern hinges on a clean separation of objects into one of three categories — **models** for maintaining data, **views** for displaying all or a portion of the data, and **controllers** for handling events that affect the model or view(s). Events typically cause a controller to change a model, or view, or both. Whenever a controller changes a model's data or properties, all dependent views are automatically updated. Similarly, whenever a controller changes a view, for example, by revealing areas that were previously hidden, the view gets data from the underlying model to refresh itself. In this design pattern, the *model object* knows about all the data that need to be displayed. It also knows about all the operations that can be applied to transform that object. However, it knows nothing whatever about the GUI, the manner in which the data are to be displayed, nor the GUI actions that are used to manipulate the data. The data are accessed and manipulated through methods that are independent of the GUI. The *view object* refers to the model. It uses the query methods of the model to obtain data from the model and then displays the information. The *controller object* knows about the physical means by which users manipulate data within the model.

Part 1: The Model

There are basic characteristics that need to be available for any learning model for CVL environments which includes (1) complying with a learning object standard, (2) to have a flexible model to represent virtual graphics, and (3) to be reusable. The first is on having an XML like schema [25]. In country like Canada the standard model used is CanCore [26] (see Fig. 1).

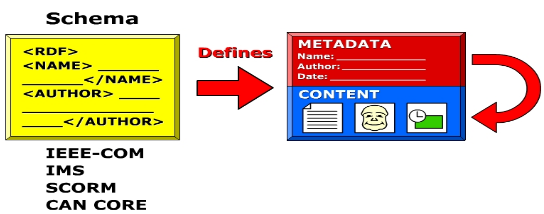

Fig. 1. The Learning Object Model

The second criterion is related to the model used to represent the virtual graphics of the learning objects contents. This model requires to be flexible and to have the ability to relate between the metadata description and its content. Flexibility ensures that the modeled graphics can be interpreted/transcoded/optimized according to the capabilities of the receiving device. The second criterion ensures that any change at the metadata must imply a change in the actual contents. The third criteria addresses reusability of learning objects which mainly means that the learning object should be an open source.

However, choosing a flexible virtual graphics model is not a big problem, since there are only two dominating models used by many programmers and programming languages: *Scene Graph* and the *DOM Tree*. However, in order to relate between the metadata and the contents described as scene graph, one need to represent the different nodes in that graph as generic 3D/virtual graphical objects/components and make those objects to be controlled and animated by behaviors as described by their metadata (e.g. using SceneBean[27], Virtual SceneBeans [28][29] to represent animated 3D learning objects). The second model used to describe the virtual scene is the Document Object

Model (DOM) tree. This model has been largely used by the W3C SVG standard [30] in which an XML document type is used for describing two-dimensional graphics and animations. SVG defines a scene using the Document Object Model (DOM), a tree structured representation of an XML document. DOM nodes are used to represent primitive shapes, styles, paths and groups. DAG structures are defined by referencing one part of the document from another with a URI. SVG defines a number of basic animation algorithms that can be declaratively applied to the properties of DOM nodes. More complex animations can be defined by embedding scripts within the SVG document. The main advantage of using SVG DOM model is that it can be transcoded easily to other forms of scene trees by using additional APIs like Batik (http://xml.apache.org/batik). Indeed the difference between Scene Graph and DOM is not that major as the Scene Graph represents the Typed version of DOM tree.

Part 2: The Controller

The controller is the code that determines the overall flow of the application mode l within the environment. Basically it comprises one or more struts actions, servlets, portlets, beans and/or Web Services that manage the accessibility of the various requested virtual leaning objects. This means the control need to have a Learning Content Management System (LCMS) which can interpret queries and return the right sequence of requested and relevant learning objects (Fig. 2).

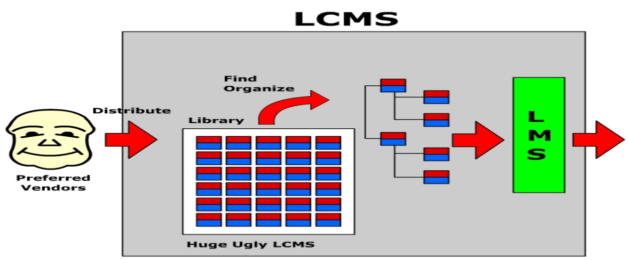

Fig. 2. The LCMS Controlling Part

At the heart of the LCMS is the Learning Object Search engine. This search engine should not be based on keywords. Since keywords does not capture the underlying semantics of Learning Objects. The search engine must have the ability to discover Learning Objects and to promote Learning Object Context Awareness. Context in LO discovery can be defined as *the implicit information related both to the requesting user and service provider that can affect the usefulness of the returned results* [32]. Learning Object *context* can be the location of the service, its version, the provider's identity, the type of the returned results and possibly its cost of use. On the other hand, each user is characterized by a *user context* that defines her current situation. This includes several parameters like location, time, temporal constraints, as well as device capabilities and user preferences. During service discovery user context is matched against service context in order to retrieve relevant services with respect to context-awareness.

Indeed, the Universal Description, Discovery, and Integration (UDDI) Project provides a standardized method for publishing and discovering information about web services and can be used for LO discovery too. The UDDI engine needs to be built upon a push model that pushes LO service information into the learning environment at a steady period of times. However, the UDDI is too primitive to capture the deep semantic structure of variety of learning objects. In this case we need to let UDDI to be aware about the ontology used for our context of search. This can be done by making UDDI an OWL aware engine [33]. Moreover, we need to support one of the delivery protocols (Axis SOAP, Jini, IETF SLP, UPnP, Bluetooth SDP, Jabber, Naradabrokering, or JXTA). In particular Axis encourages variety of intermediary services to be incorporated within the chain of the SOAP Message processing. This can be very useful to add more intelligent filters that can aid in the process of learning and the awareness of existing LOs. Having such LCMS Search Engine, learners can collaborate more successfully and create better solutions to complex, ill-defined problems by using such awareness-based search engine that support members' shared understanding of long-term goals, plans, challenges and allocation of resources. The more shared awareness among learners the more effectively a group will function.

Part 3: View

The view is the code that registers itself as a listener to certain parts of the application's underlying business and functional logic, as represented by the model. The model then notifies all registered views whenever there is a change in the data. Completing the cycle, the controller receives user actions and dispatches them to the model. This idea can be simply realized using a bean interface using for example Java Media Framework (JMF) from Sun Microsystems. But this idea will only work for devices that have at least the desktop PC capabilities. For non-PC devices, which include mobile phones, digital TV sets, car telematics, have various and limited resources compared to desktop PCs. These resources include small memory size, CPU power, small screen size, restricted input methods, and network bandwidth. For this we propose that the viewer can be modeled as a generic multimedia player based on the SVG engine like the Ikivo player (http://www.ikivo.com/02player_mmsvg.html) or the SVG engine (http://www.svgopen.org/2004/papers/ModularSVGEngineArchitectureForIA/). However, what we are proposing is a rendering engine that can work for both PC and non-PC devices. This can be done following having either a SAX or DOM processing engine for rendering the received SVG media (Fig. 3).

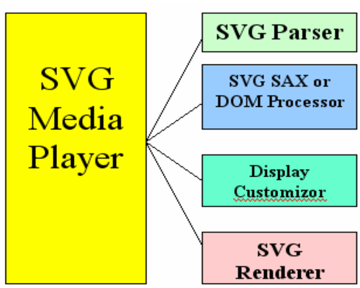

Fig. 3. SVG Generic Multimedia Player for the Viewer

SVG supports three media elements (audio, video, and animation). Media elements define their own timelines within their time container. All SVG Media elements support the SVG Timing attributes and run time synchronization.

6 Conclusions

Model-View-Controller (MVC) is a widely used software design pattern that was created by Xerox PARC for Smalltalk-80 in the 1980s. More recently, it has become the recommended model for Sun's J2EE platform, and it is gaining increasing popularity among software developers. This article introduced a first level refinement of the MVC model that can be used to design collaborative virtual learning systems within Web and ubiquitous environments. Other features that may contribute to Web Intelligence may be added as information filters to the model or controller parts of the MVC basic model. The term "information filtering" refers to both finding desired information (filtering in) and eliminating that which is undesirable (filtering out). We recently used SVG SAX filters to achieve media protection within ubiquitous environments [31] as one component to implement a search engine for learning objects [34].

References

1. Yao, Y.Y., Zhong, N. Liu, J., Ohsuga, S.: Web Intelligence (WI): Research Challenges and Trends in the New Information Age, in Zhong, N., Yao, Y.Y., Liu, J., and Ohsuga, S. (eds.) Web Intelligence: Research and Development, LNAI 2198, Springer-Verlag (2001) 1-17
2. Bareiss, R., Porter, B. PROTOS: An exemplar-based learning apprentice. Proc. of the 4th International Machine Learning Workshop Irvine, CA: Morgan Kaufmann, (1987) 12-23
3. Bruner, J. Vygotsky: An historical and conceptual perspective. Culture, communication, and cognition: Vygotskian perspectives, 21-34. London: Cambridge University Press (1985)
4. Jordan, B.: Technology and Social Interaction: Notes on the Achievement of Authoritative Knowledge in Complex Settings. IRL Report number IRL92-0027, Institute for Research on Learning, Menlo Park, Calif. 94025 (1992)
5. Lave, J. and Wenger, E.: Situated Learning: Legitimate Peripheral Participation. Cambridge, Eng., Cambridge University Press (1991)
6. Matsushita, Y. Okada, K. (Eds.): Collaboration and communication, Distributed collaborative media series 3, Kyoritsu Press (1995)
7. Fish, R., Kraut, R. Chalfonte, B.: The video window system in informal communications, Proceedings of Computer Supported Cooperative Work O92, ACM Press (1990) 1-12
8. Root, R. W.: Design of a Multi--Media Vehicle for Social Browsing, Proceedings of Computer Supported Cooperative Work 88, (1988) 25-38
9. Dourish, P. Bly, S.: Portholes: Supporting awareness in a distributed work group, Proceedings of Computer Human Interaction 92 (1992) 541-548
10. Matsuura, N., Hidaka, T., Okada, K. Matsushita, Y. VENUS: An informal communication environment supporting interest awareness, Trans. Information Processing Society of Japan, 36(6) (1995) 1332-1341
11. Ogata, H. and Yano. Y.: Combining Knowledge Awareness and Information Filtering in an Open-ended Collaborative Learning Environment, Int. J. of Artificial Intelligence in Education, 11 (2000) 33-46

12. Capin, T. Pandzic, I.S. Thalmann, D. and Thalmann, N.M.: Realistic avatars and autonomous virtual humans in VLNET networked virtual environments. Virtual Worlds on the Internet, J. Vince and R. Earnshaw, eds., IEEE Computer Society, Los Alamitos, (1998) 157-173
13. Benford, S. Bowers, J. Fahlén, L. E. Greenhalgh, C. andSnowdon, D. : User Embodiment in Collaborative Virtual Environments. ACM CHI '95 Proceedings, (1995) 242-249
14. Jianhua Z., Kedong L., Akahori K: Model P2P Structuring Algorithm. ing and System Design for Web-Based Collaborative Learning, Proc. of the 2nd Int. Conference on Information Technology based higher Education and Training, Kumamoto, Japan, (2001)
15. El Saddik, A. Seeberg, C. Steinacker, A. Reichenberger, K. Fisher, S. Steinmetz, R: A Component-Based Construction kit for Algorithmic Visualization, Proceedings IDPT, Springer Verlag, N.Y (2000)
16. Engel, T. Meinel, C.: Implementation of a WebDAV-based Collaborative Distance Learning Environment, ACM SIGUCCS 2000 Proceedings, Virgina (2000)
17. Shirmohammadi., S. El Saddik, A. Georganas, N.D. Steinmetz, R. : JASMINE: A Java Tool for Multimedia Collaboration on the Internet, Journal of Multimedia Tools and Applications, 19(1) (2003) 5-28
18. Oliveira, J.C. M. Hosseini, M. , Shirmohammadi, S.,Malric, F. Nourian, A. El Saddik, A. Georganas, N.D.: Java Multimedia Telecollaboration, IEEE Multimedia Magazine, 10(3) (2003) 18-29
19. Duval, E.: Standardized metadata for education: a status report. In Montgomerie C. and Jarmo V. (eds.) Ed-Media 2001, World Conference on Educational Multimedia and Hypermedia. AACE, (2001) 458-463
20. IEEE: IEEE Learning Technology Standards Committee (LTSC), Learning Object Metadata Working Group. (2002). http://ltsc.ieee.org/wg12/
21. 21. IMS: (2003). http://www.imsglobal.org/content/packaging/
22. CETIS (Educational Technology Interoperability Standards)(2003). http://www.cetis.ac.uk/
23. El Saddik A., Ghavam, A., Fischer, S. Steinmetz, R.: Metadata for Smart Multimedia Learning Objects, Proc. of the 4th Australasian Comp. Education Conf.. ACM-CSE, Australia, (2000)
24. Jern, M.: Visual Intelligence - Turning Data into Knowledge, IEEE Int. Conference on Information Visualization (1999)
25. Friesen, N.: Three Objections to Learning Objects, In McGreal, R. (Ed.). Online Education Using Learning Objects. London: Routledge/Falmer (2004)
26. Richards, G., McGreal, R. Friesen N. Learning Objects Repositories for TeleLearning: The Evolution of POOL and CanCore, IS2002 Proc. of the Information Science & IT Education Conf., Irland, (2002)
27. Pryce, N. Magee, J.: SceneBeans: A Component-Based Animation Framework for Java, Technical Report, Dept. of Computing, Imperial College (2001)
28. Fiaidhi, J.: Virtual SceneBean: a Learning Object Model for Collaborative Virtual Learning, Informatics in Education: Int. Journal, 3(2) (2004) 191-218
29. Fiaidhi, J. Mohammed, S. "Developing a Collaborative Virtual Learning Environments for P2P Grids Environment", IEEE LTTC Learning Technology Journal, (2005)
30. Lee, S. Fox, G. Ko, S. Wang, M., Ubiquitous Access for Collaborative Information System using SVG, SVG Open Conference, Zurich, Switzerland, (2002)

31. Fiaidhi, J., Mohammed, S., Garg, M. Arif, A.: Developing a SAX Filtering Intermediary Service for Protecting SVG Multimedia Contents in a Ubiquitous Publish/Subscribe Environment, International Conference on Internet Computing (ICOMP'05), Las Vegas, USA, (2005)
32. C. Doulkeridis C. and M. Vazirgiannis, M: Querying and Updating a Context-aware Service Directory, IEEE/WIC/ACM Int. Conference on Web Intelligence (WI'04), (2004) 562-565
33. Srininvasan, N. Paolucci,M. and Sycara,K: CODE: A Development Environment for OWL-S Web Services, 3rd International Conference on Semantic Web,ISWC2004, Hiroshima, Japan, (2004)
34. Fiaidhi J and Mohammed, S.: Developing a Search Engine for Learning Objects, Asian Journal of Information Technology, 3(7) (2004) 533-545