IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Hershey PA 17033-1117, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4108

A Web Based Model of an Adaptable Multi-Media Intelligent Tutoring System

Eshaa M. Alkhalifa and Fawzi AlBalooshi
Department of Computer Science, College of Science, University of Bahrain, PO Box 32038, Isa Town, Bahrain
Tel: (973) 449999, Fax: (973) 682582, {ealkhalifa, fawzi}@sci.uob.bh

ABSTRACT

Although the Internet offers a novel promising ground to seekers of education, it has been lacking in its generosity to foster Multi-Media. Loading time issues plus the strict content of HTML pages seemed as a prime hindrance to all those who wished to mix the two media online yet retain the flexibility of an adaptable system. As a consequence adaptable intelligent tutoring system were mostly confined to presenting textual material and simple graphs. Here, we present a model that was designed and implemented through Java Servlets that dynamically create the HTML page according to student requirements. The screen is a composite of files put together for a particular student with both an animated part and a textual one in addition to an interaction window. A student is guided through a maze of different approaches to explain the same material according to a student model. The result was a Multi-Media adaptable Intelligent Tutoring System on the Internet.

INTRODUCTION

The sudden growth of the Internet and its availability to an international population has made it especially attractive to educators worldwide. Norman (1988) indicates that each media has "affordances" and "constraints" that would be either beneficial or counter-active to educational goals. The Internet as a mediator of information, is no exception as a full and complete understanding of its "affordances" and "constraints" has not as yet been achieved.

> "Backward looking metaphors focus on what we can automate – how we can use new channels to send conventional forms of content more efficiently – but miss the true innovation: redefining how we communicate and educate by using new types of messages and experiences to be more effective." (Dede & Palumbo ,1991)

Rather than use the Internet to deliver classically stored educational material in the same way information is presented in books, some understanding of what it has to offer could be beneficial. Supporters of hyper-media as an educational media offer the following as support for their belief:

1. "The associative, nonlinear nature of hypermedia mirrors the structure of human long-term memory, empowering both intelligence and coordination through intercommunication.
2. The capability of hypermedia to reveal and conceal the complexity of its content lessens the cognitive load on users of this medium, thereby enhancing their ability to assimilate and manipulate ideas.
3. The structure of hypermedia facilitates capturing and communicating knowledge as opposed to mere data.
4. The hypermedia's architecture enables distributed, coordinated interaction, a vital component of teamwork, organizational memory and other "group mind" phenomena" (Dede & Palumbo ,1991)

However, as the authors point out, some skepticism exists about the validity of the points made. They also point out the main disadvantages of hypermedia as follows:

1. "people become disoriented when navigating through large hypermedia structures.
2. traversing a hypermedia network imposes considerable cognitive overhead on the user.
3. creating hypermedia structures involves a very large front-end investment of time and expertise.
4. "Tower of Babel" situations are likely in shared hypermedia systems." (Dede & Palumbo ,1991).

It seems obvious that the jury is still out on the final decision with respect to the utilization of the full scope of hypermedia. Therefore, a possible guideline that one may follow could be to ask how effective this media is in educating students in a particular course. A data structures course was designed as a Java Applet (AlBalooshi & Alkhalifa, 2001) shows that when students are exposed to this system following a classroom lecture, their levels increased no less than 40% in a post-test as compared to the pre-test they did following the lecture and before exposure to the system. Two control groups that were exposed only to the system or only to the lecture showed a comparable mean. This implies that a Multi-Media system does seem to have the ability to convey the information to the learner in an effective fashion.

However, the limitations of the system were immediately clear as it was extremely slow in loading and in response times on conventional browsers. Other limitations included the difficulty faced by programmers to produce the simple animations used while allowing students some flexibility to control the speed of the animation and the start. This system, did not in any way reflect any "intelligent" behavior in that it was not able to adapt its reactions to student progress. Therefore, it did not take full advantage of what the Internet offers.

ADAPTIVE TECHNOLOGIES ON THE WEB

The aim of adaptive techniques is to draw the path on which the educational media will guide a student along the learning curve. In fact, most decisions are made according to the user's state or level of expertise (Chan, Chan, & Cao, 2001). Most of the Internet based systems use a variation of conditional reasoning where a decision is made to organize the text differently from fragments or select from a group of whole pages or even groups of pages. The link structure of a hyper document can also be modified by color-coding, or sorting according to specific criteria based on student preferences or abilities. A student model is usually created and maintained past the individual session and opened when a student logs into the system. This model retains in details all required information by the system from which the various decisions could be made. The decisions are to select between lessons or pages, which are going to be presented to that particular student.

Brusilovsky (1999) conducted a review of existing Adaptive Intelligent Tutoring Systems (ITS) on the Internet. He classified them into three groups; curriculum sequencing, intelligent analysis of student's solutions, and interactive problem solving support. The goal of the curriculum sequencing technology is as the name implies, to provide the student with the most effective sequence of information tailored to that individual student. This includes the selection of examples, questions, problems etc. This can occur in two ways, either active or passive sequencing depending on the learner's role in the selection process. The sequencing is classically done using CGI scripts to select the next lessons to be presented to students and we know the problems

This paper appears in *Issues and Trends of Information Technology Management in Contemporary Organizations,* the proceedings of the Information Resources Management Association International Conference.

with CGI scripting. The second and third are concerned when students are faced with problems and investigations into possible causes of student errors are required to guide student learning.

> "The goal of the adaptive presentation technology is to adapt the content of a hypermedia page to the user's goals, knowledge and other information stored in the user model" (Brusilovsky,1999)

On the other end of the scope, Adaptive hypermedia, is a relatively new area and it means that the page content and links are different to various users usually based on decisions made according to a user model (Brusilovsky, 1999). Links can either be color-coded or some are disabled to make them invisible to users. Heuristics are used to decide which are the most appropriate pages to be visited by this particular visitor. In a strange sense, there does not seem to be a large difference between adaptive sequencing and Adaptive Navigational Support (ANS) when one looks at how sequencing is implemented on the WWW. The most popular form of ANS on the web is annotation, which was used in various systems including ELM-ART (Brusilovsky, Schwarz, & Weber, 1996).

ELM-ART (ELM – Adaptive Remote Tutor) is an adaptive intelligent tutoring system that was designed to support learning Lisp programming (Brusilovsky et al, 1996). The system presents students with a textbook style curriculum along with a reference manual. A set of similarity links, exist between the textbook and the various concepts in the reference manual in both directions. The links are therefore content based and are closely related to the structure of the Lisp language. The textbook itself has a set of interactive pages where students can try out examples by filling out forms to see the results of the instructions.

> "An important feature of the system is that all pages presented to the user are generated adaptively on the fly when the user requests them" (Brusilovsky et al, 1996)

The implementation of ELM-ART was on the Common Lisp Hypermedia Server CL-HTTP, which offers a Common Gateway Interface (CGI) to handle incoming requests. This CGI program associates this call with a Lisp function that responds to it in order to generate a HTML page as an adaptive response. The power of the language, which Lisp programming uses, is transmitted through the CGI to the world of the Internet. Espinoza and Hook (1996), on the other hand, used a Java Applet to allow interactivity with the learner. They allowed users to navigate through in the information space by clicking on the graphs or posing questions using menus. Along with this, a task analysis study was conducted and used that to construct a hierarchy of information seeking tasks. Rules were inferred from the study to guide the adaptation process and instruct a live CGI (Common Gateway Interface) program to generate the appropriate HTML page.

The two Adaptable ITS systems presented imply that the web does offer the ability to create custom pages that are generated according to certain criteria but not without an overhead cost. Although the designers of both systems did not discuss efficiency problems, the following are commonly connected to any applications linked through CGI to the Internet.

- Difficulty in maintaining session state: A connection could be lost and the session related information is interrupted.
- Performance bottlenecks: high requirements on initialization of CGI scripts.
- Can involve proprietary APIs: hard to work with more than one database at the same time or to migrate data from one to another (Kirby, Lee)

AHA (De Bra et al., 2000) is a system that is implemented in XML (eXtended Markup Language) and is therefore not subject to the above constraints. It uses a student model to filter page contents and link structures conditionally by either including or excluding page fragments. Its adaptive linking can be either "link annotation" or "link hiding" or even "link disabling". It was implemented primarily using XML technology to generate the HTML pages on the web. Since XML was originally developed to allow HTML to be more functional, it is strongly related to the way documents are presented on the screen. In order to include a fragment of a page in the text presented to the learner the following code would be used:

```
< if expr="requirement1 > 30 and requirement2 < 80 >
   < block >
        here is the conditionally included fragment
   < /block >
< /if >  (De Bra, 2000)
```

In a sense, XML allows one to manipulate how a document is displayed. A programmer can define a structure in a related document and attribute meaning to the tags that are defined. This allows the creators of the AHA system to manipulate whatever fragments are presented within the rules set by the display requirements. The texts are still essentially part of the XML document surrounded by their tags as defined by the user.

A JAVA SERVLET APPROACH

The approach introduced here takes one a step further along the road of Adaptive Technology by taking full advantage of the server side abilities of Java Servlets. The system presented here is similar in a way to the power related to Lisp in the implementation of ELM-ART with the added advantage of the high portability of the Java language.

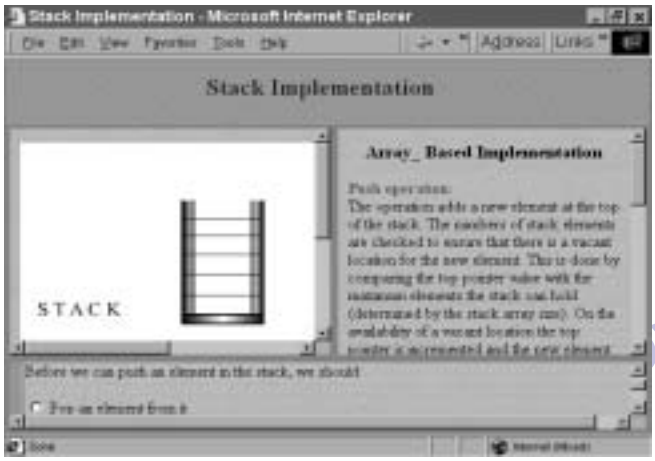
These Servlets are complete programs that are capable of creating Java Server Pages (JSPs) or even simple HTML pages. A Servlet allows a programmer to utilizes whatever functions a programmer needs including conditional branching and loops. This allows for much more flexibility in creating the page than XML. Oddly enough, Servlets do not face any of the problems faced by classical CGI programming because a Servlet has a lifecycle. Even a loss of connection may not kill it, unless the Server side program kills it. It is light weight, which means that it does not require a large amount of processing power on creation of the Servlet instance, and it accesses databases using JDBC which offers a secure way of accessing many well-established database brands. The aim of using this particular approach was to have the flexibility of isolating the control structure (Servlets) completely from the files that contain the teaching material. The only link between them is the file names. Therefore, it would be quite easy to add or alter any lessons without changing more than one value or variable in the program. Additionally, the type of files as containing textual information or animation is completely invisible to the Servlet. This allows full freedom to the system designers to include whatever types of files, and in this case it would be a combination of textual, animated, and a query in one page composed of three sections.

To test this technology, a Data Structures course was broken up into four main modules; stacks, queues, linked lists and trees. Each was then broken up into several parts; concept, implementation, examples and quiz. The first three parts were then broken up into four examples each. The initial selection of the first part to go to is made through a random choice and from then on based on a value passed to the Servlet. The commands that generate an HTML document in fact regard it as a form of output where each line 'prints' the html script line by line as in the follow code:

```
out.println("<html>");
out.println("<head>");
out.println("<title>Stack Concept</title>");
out.println("</head>");
```

A Page is broken up into two square frames as is shown in Figure 1. Below them students find a question that tests their comprehension of the presented material. The textual part is a simple text file that can be produced by any word processor while the animated part is a flash animation. Macromedia's flash technology offers a very light-weight vector graphic based medium to develop animation for the web. It only keeps minimal information about the objects presented making the files much smaller than is common for graphical representations. Additionally, shockwave, its published version is very well supported by most browsers or can be added by downloading a plug-in with a 2-minute download time. It allows for interactivity on the web, which made it an excellent programming environment for many online games.

Figure 1: A sample composite screen from the multi-media ITS system



Here, these abilities are used to give students more flexibility with controlling the presented animations.

Students are guided randomly at first to one of the types of explanations related to the same concept. A sample of the code used to do this follows:

```
i = (int)Math.floor(Math.random()*4);
switch (i) {
        case 0: {
                 out.println("<frame name=\"main1\";
                src=\"c:/javawebserver2.0/servlets/
                   scanimation1.htm\";
               scrolling=\"auto\";
                style=\"border-style: solid; border-color: #808080\"
                  target=\"_self\">");

                 out.println("<frame name=\"main2\";
                src=\"c:/javawebserver2.0/servlets/sctext1.htm\"
                 scrolling=\"auto\";
                 style=\"border-style: solid; border-color: #808080\"
                  target=\"_self\">"); break;
               }
               …
} // end switch for I
```

The power of the language should be clear from the clarity of the code. Its quite simple to add conditionals, loops and whatever mathematical calculations that are necessary to utilize the student model to predict future choices of explanation formats. The passing of the parameters could be done, from the screen to the Servlet or from one Servlet to another allowing full freedom to predict the factors that

Figure 2: An overview of the adaptive multi-media ITS

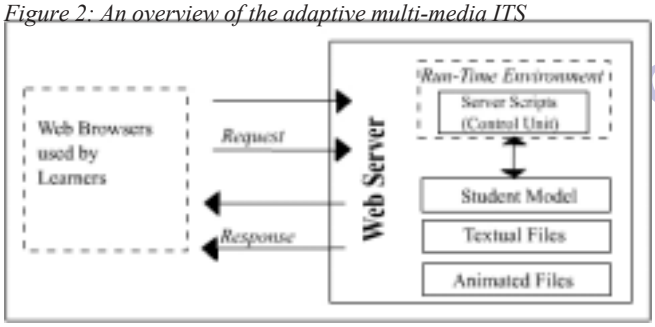
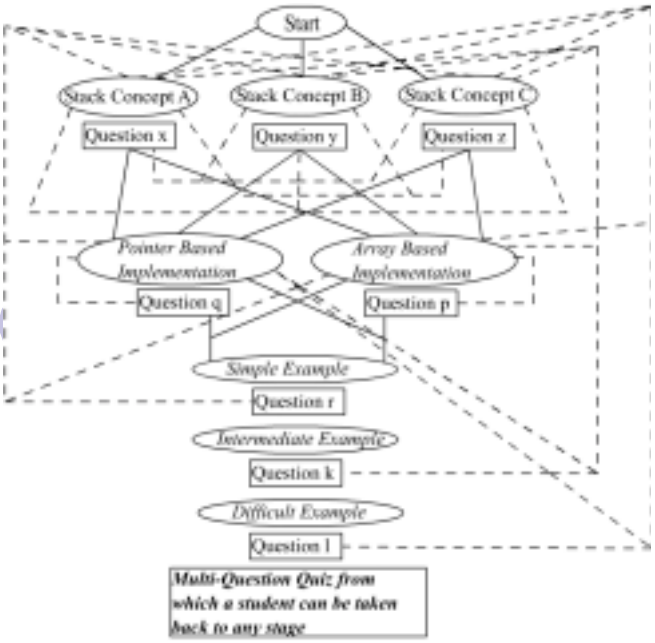


Figure 3: An overview of the possible paths that may be followed by students



permit this particular learner to learn more efficiently. The system as a whole is represented in Figure 2.

Note that the learners access the system from their own browsers and are only supplied with HTML rather than XML, which may require interpreting. The server responds to learners by creating new instances of the Servlets as and when appropriate that will tend to that learner until the Servlet lifecycle is complete. This Servlet will be responsible to supply that student with the pages and to create them by collecting information from the various data files that are stored. The various paths that students can follow in just one of the four modules that compose the system are shown in the following diagram.

The solid lines or the direct path downwards is when a student gets all the answers correctly. Errors on the other hand are not as simple to navigate. The system investigate the types of errors that students made and reroutes the student to the exact module, which is mostly likely to have caused that error. Remediation then takes place with the hope of placing a student back on the right track but note that a student is not taken back to the same explanation that the student was exposed to the first time through. This decision is made based on the information in the student model.

ANALYSIS AND CONCLUSION

This paper presented an implemented model for a Multi-Media Adaptable Intelligent Tutoring System. Most adaptable systems proposed for the Internet to this date are either purely textual or with minimal graphics due to the restrictions imposed by the implementation strategy. Here, we show the power of Java Servlets as a hidden control unit responsible for deciding the components of an HTML screen that is viewed by the learner. We show that Servlets are capable of creating a frames screen and filling the different parts with different types of information or media, all published as HTML. These Servlets are capable of eliciting student responses on more than one question and analyzing them to find out the strengths and weaknesses of that student to direct them towards remediation. Flash is used as the medium to present animation as it is published ready as a shockwave object that is inserted into an HTML document. In a previous study (Albalooshi & Alkhalifa, 2001) a Multi-Media has shown its ability to effectively transfer information to students. Therefore, the system

presented here will hopefully encompass that effectiveness with the added "adaptability" feature and quick response times on the Internet towards an even better result.

REFERENCES

Albalooshi, F., & Alkhalifa, E. M. (2001). Multi-Media as a Cognitive Tool: Towards a Multi-Media IT. Proceedings of IEEE International Conference on Advanced Learning Technologies: Issues, Achievements and Challenges. Madison, Wisconsin. pp 231-234, August, 2001.

Brusilovsky, P. (1999). Adaptive and Intelligent Technologies for Web-based Education. In C. Rollingerand C. Peylo (eds.) Kunstliche Intelligenz, Special Issue on Intelligent Systems and Teleteaching, vol.4, pp. 19-25.

Brusilovsky, P., Schwarz, E., & Weber, G. (1996). ELM-ART: An intelligent tutoring system on World Wide Web. In Frasson, C., Gauthier, G., & Lesgold, A. (Ed.), Intelligent Tutoring Systems (Lecture Notes in Computer Science, Vol. 1086). Berlin: Springer Verlag. 261-269.

Chan, A. T.S., Chan, S. Y.C., & Cao, Jiannong, (2001). SAC: A self-pased and adaptive Courseware System. Proceedings of IEEE International Conference on Advanced Learning Technologies: Issues, Achievements and Challenges. Madison, Wisconsin. pp 78-81, August 2001.

De Bra, P., Aerts, A., Houben, G.J., & Wu, H. (2000). Making General-Purpose Adaptive Hypermedia Work. Proceedings of the WebNet Conference, pp. 117-123.

Dede, C. & Palumbo, D. (1991). Implications of hypermedia for cognition and communication. Impact Assessment Bulletin, 9(1-2), 15-28.

Espinoza, F., & Hook, K., (1996). A WWW Interface to an Adaptive Hypermedia System. In Proceedings of the First International Conference of the Practical Applications of Intelligent Agents and Multi-Agents (PAAM '96), pp. 143-167, April 1996.

Kirby, J., & Lee, C. Developing Java Servlet JDBC Applications: A replacement for CGI, A white paper by Dynamic Information Systems, Boulder, Colorado.

Norman, D. (1988). The design of everyday things, New York: Currency/Doubleday.

An Adaptive Enhancement Method of Malicious Traffic Samples Based on DCGAN-ResNet System
Qiankun Li, Juan Li, Yao Li, Feng Jiuand Yunxia Chu (2024). *International Journal of Information Technologies and Systems Approach (pp. 1-17).*
www.irma-international.org/article/an-adaptive-enhancement-method-of-malicious-traffic-samples-based-on-dcgan-resnet-system/343317