Vidyakash

A project under the Ministry of Information Technology,
Government of India

A Workshop on Content Design for Online Learning

Conducted by NCST

April 2001

Interactive Learning/Testing Tool

Peter Pereira, Rhona Gamma, Acbbar Khan and Utpal Lotlikar
<peter@brainsoft.ch>, <rhona@brainsoft.ch>,
<acbbar@brainsoft.ch>, <utpal@brainsoft.ch>
Indo Swiss Financial Software Development Company Pvt. Ltd

Dr. V. V. Kamat and Dr. D. D. Kare
<vvkamat@goatelecom.com>, <ddkare@goatelecom.com>
Goa University

Abstract

Teaching finance to students who are not mathematically inclined is very challenging. Interactive Learning Tool (ILT) has been designed keeping in view the needs of a learner who may require some help in addition to the classroom instructions. It contains learner centric content with large number of simulated exercises to help the learner to understand the concepts in finance. Multiple Choice Questionnaire (MCQ) is a self-testing environment designed for the purpose of checking preparedness of learner. In this paper, we will discuss both the applications. In the current version, ILT and MCQ are not integrated.

Background

The work reported in this paper has been carried out for Training Center for Investment Professionals (TCIP), a Zurich based company conducting training programs for banking professional and portfolio managers. The actual content to be programmed is provided by the parent company. The developmental work is carried out at its subsidiary company located in Goa.

Target Audience

The course consists of set of modules. The lectures for each module are conducted on weekends at three different locations in Switzerland, namely, Zurich, Geneva and Lugano. Currently the course is taught in three different languages, i.e. German, French and Italian. The course manual is prepared in English and a hardcopy of the manual is given to each participant at the beginning of the module. University professors and practicing professionals write the manuals. The participants are mostly working professionals sponsored by their companies. At the end of the course, participants appear for a certification examination (similar to CFA) conducted by TCIP.

Motivation

TCIP desires to add value to learning by creating content that is better understood by the participants. The subject of finance contains large number of formulae and

mathematical derivations and participants often find it difficult to understand the material. One of the objectives behind programming the content is to encourage the participant to experiment, and understand the models represented by the formulae. To create a learning experience through exploration, interactivity and visualization is the primary motivation behind ILT. The programmed content is also expected to serve as a quick reference for practicing professionals for on-line help.

General Constraints

The work was started in January 1999. Currently the team contains 3 programmers and 1 subject expert. About 80 man months of efforts have gone in programming the manuals. It was a conscious decision not to use any authoring tool because it was observed that these tools are very general purpose and do not provide necessary flexibility and control over the content. Further, since it was expected that the participant could access the content either from home or office, no assumptions could be made about the availability of specific software or the platform. Finally, the content management tools available in the market were found to be very expensive.

One of the major constraints faced in development is the separation of the team that produced the content and the team that programmed the content. The original author of the content was never available to see how the content has been programmed. The feedback was primarily based on few subject experts in Switzerland. Till date the content has not been field tested for its usability or for its utility value in learning.

The current version of ILT requires on the server side Windows NT running IIS and SQL and an Internet Explorer on the client end.

Methodology Used

Whenever a manual for a module is required to be produced, a team of subject experts is formed to decide on the content to be covered in that module. This is called body of knowledge (BoK). BoK typically contains a table of content that lists different chapter headings, section and subsection headings and a list of references. BoK is then handed over to content writers. At times there are several writers for a given module. This also creates some problem in content uniformity, particularly in style and notation. The content writers than create documents in RTF file format, which is then sent across to subject experts for proof reading and corrections. Once the module content is finalized, the content programmers receive the final version of the document. The programmers than convert the RTF file into HTML and do some manual tweaking in order to make it suitable for the web. This single HTML document is than broken down into number of pages in order to store it in database.

Programmers and a subject expert then sit together and identify the content that could be programmed for interactivity. Typically, the interactive component involves a formula, a worked out example, a graph or a table. The programmers then write an applet and insert it at appropriate place in the manual. These applets are also stored in the database along with the HTML document. When a particular module is accessed,

it gives an impression that a single document is being viewed. When ILT application is started, a main page as shown in Figure 1 will appear in the browser. At this point the learner can choose a particular module of study.

On selection of a particular module, table of content is displayed in the browser. See Figure 2 below. At this point, learner can select any section or subsection for study.

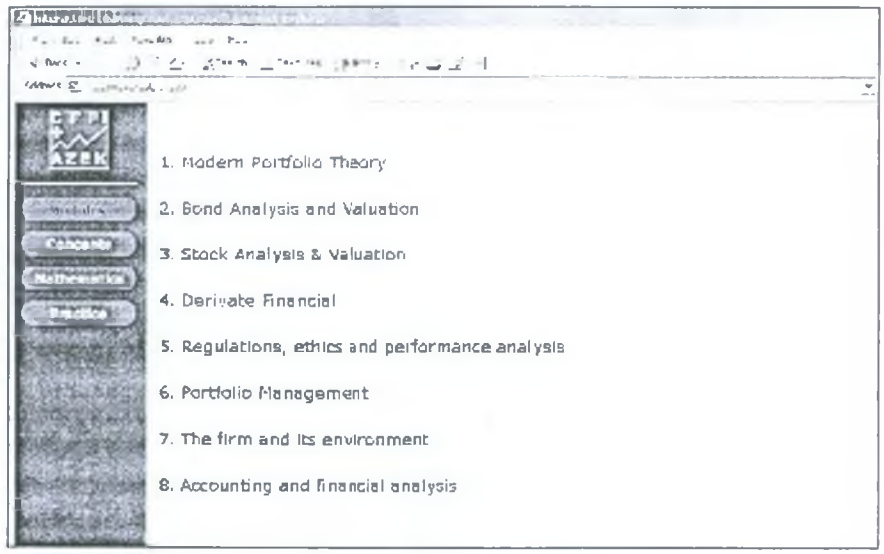

Figure 1: Main page of ILT

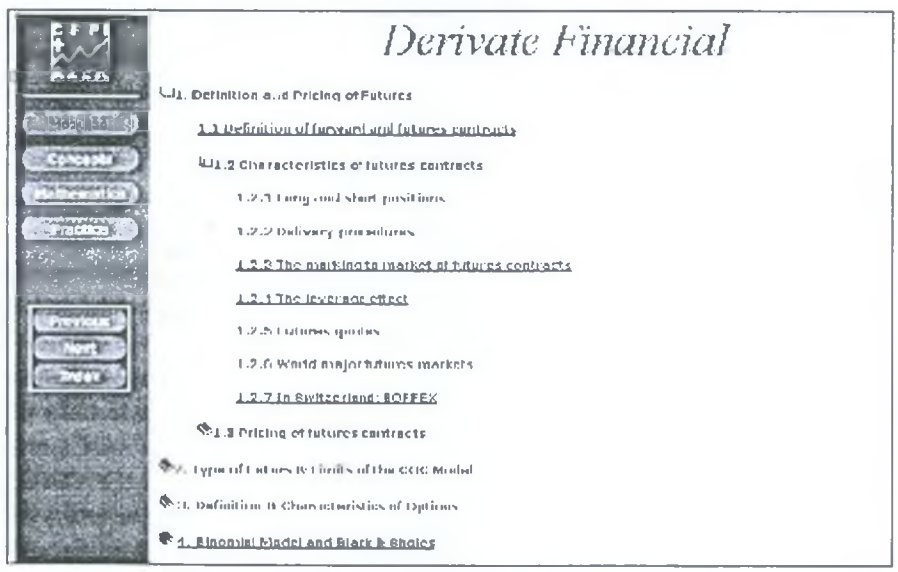

Figure 2: Table of content as seen by the learner

Figure 3 for instance, shows a typical page from the programmed manual. In this subsection, a concept is being explained with a help of a formula. The various terms in the formula are explained in the text. In case the learner wants to understand the working of the formula, s/he can click on the formula to get a template of variables as shown in Figure 4. This template can be filled in to solve class of problems as explained in the text.

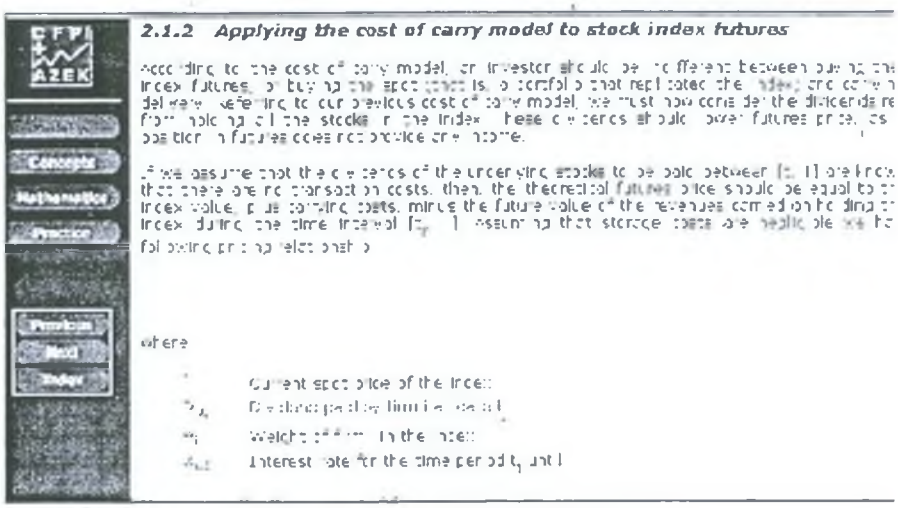

Figure 3: An example of a formula

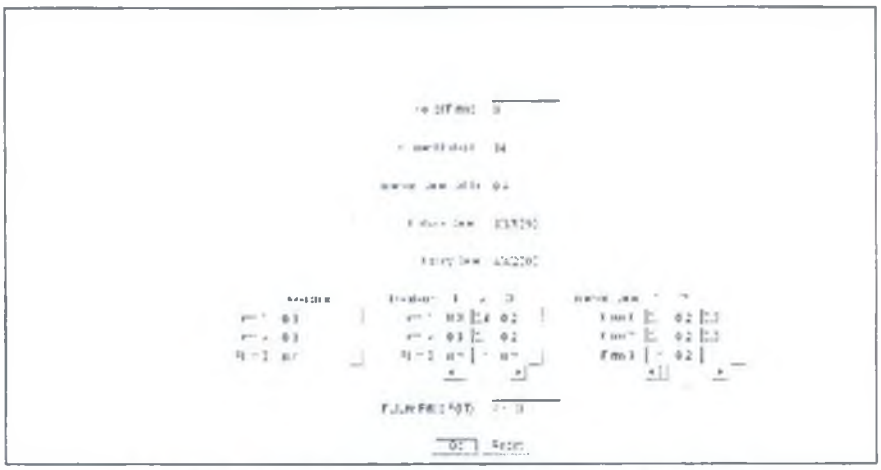

Figure 4: An applet representing a programmed formula

At times, it is preferable to visualize the data with a help of a graph and interact with a graph and the data interchangeably. This allows two-way interaction between data and the graph. For instance, a learner can directly manipulate the graph, which in turn will change the data, or manipulate the data values that in turn will change the graph. This two-way interaction is illustrated in Figure 5 and Figure 6. This kind interaction gives the learner a better feel for the data as well as the model, and adds value to learning experience.

MCQ (Multiple Choice Questionnaire) is another such application. Multiple-choice questions from the past examinations are compiled and stored in database along with answers. The questions are categorized module wise and concept wise so that learner can only take a test on concepts that s/he has learnt from a given module. Currently questions and answers are stored in database in HTML format.

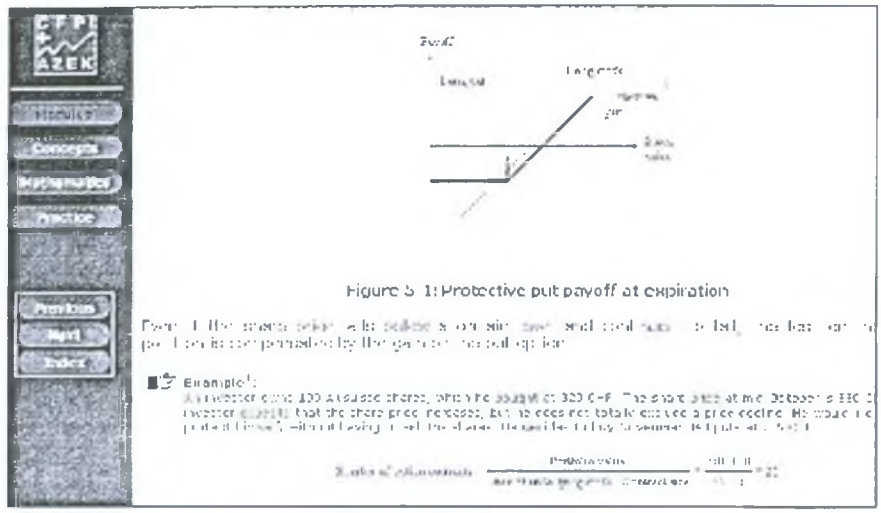

Figure 5: A simulated example

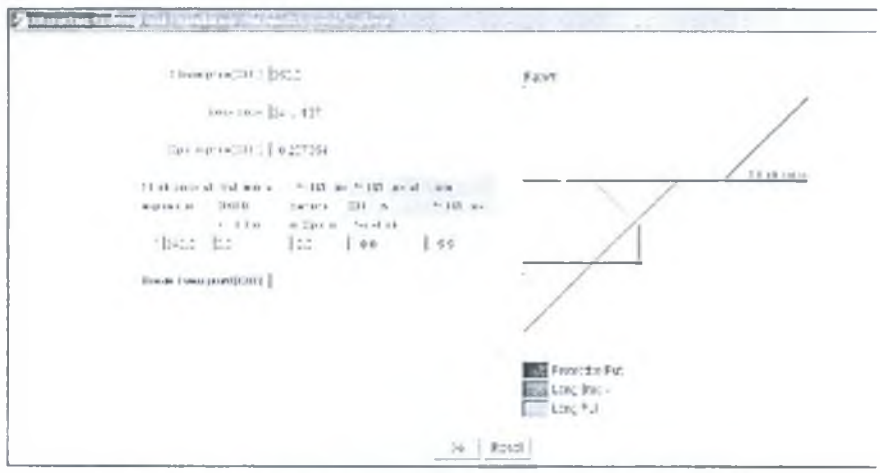

Figure 6: An applet corresponding to example in Figure 6

To use MCQ application, any learner who has registered for a course can login into the web server and can take a self-test. See Figure 7. Before taking the test, learner will have to specify the module and number of questions that s/he is prepared to answer. A sample inputs are shown in Figure 8. The system keeps a profile of each learner and randomly selects required number of questions from the database. The application is intelligent enough not to repeat a question that learner has seen earlier. It also keeps a record of individual's performance in the form of statistics.

After answering the questions, the learner can evaluate oneself by pressing the evaluate button on the form. At this moment, choices made by the learner are sent to the database for checking against the correct answers. The output of the result is immediately shown to the learner. In case, the answer selected by the user is wrong, the correct choice is indicated with an explanation. See the sample question and evaluated answer in Figure 9 and Figure 10.

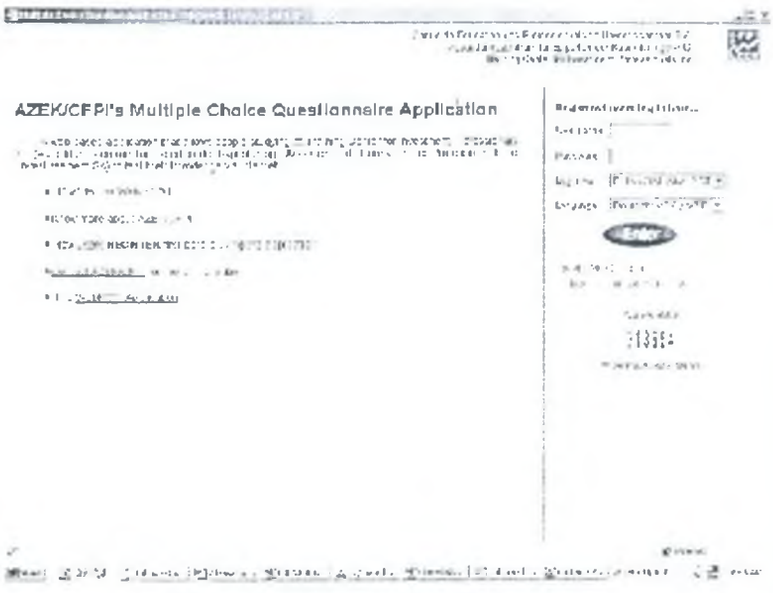

Figure 7: Login screen of MCQ application

Figure 8: Module and concept choice made by the learner

Figure 9: A sample multiple choice question

Figure 10: Evaluated question in Figure 9

Experiences with Usage

As mentioned earlier, the ILT has not yet been released for usage. It is scheduled for release only after some positive data is available through field tests. On the other hand. MCQ application has been released for usage. It is observed that MCQ application has maximum utilization during the examination period.

Lessons learnt

It was observed that the content writers used a very general-purpose tool like Word to capture the content. The Word document when simply converted into HTML document, results in loss of format information. And therefore, lots of effort goes in reformatting the HTML pages manually.

It was also observed that, the development team spent maximum time in understanding the requirements of content writer. Initially, the subject expert was not part of the content programming team. This restricted the productivity of the team. During this period, the team could only create simple applets based on straightforward concepts. After subject expert joined the team, the productivity of the team increased visibly. Also the team was able to develop more complex applets requiring subtle knowledge in finance.

Ten percent of content is expected to change every year. The present cycle of content creation is found to be very long and tedious. Further, due to low-level content production tools, the maintenance of content has become very cumbersome. Any changes to the original Word document triggers series of changes, causing a snowball like effect. The initial objective of making the content browser independent was not found to be easily achievable.

In browser, the mathematical formulae are rendered as plain images. For a human reader, the image of the formula and the underlying representation is the same but for computer these are unrelated items. This results in loss of context information.

Conclusions

Content creation involves team effort and it has lot of similarities to software development life cycle. Reusability and portability of content remain central issues of concern. The involvement of subject expert as a part of the development team can ensure better productivity. High-level of tools to capture the content can minimize the efforts involved in maintenance. One alternative that is being considered in future development is the use IMS standard with XML bindings.

Acknowledgements

All the authors are grateful to Dr. J. C. Dufournet, CEO, Training Center for Investment Professional (TCIP) for his initiative and support in carrying out this work. Second authors would also like to thank Goa University for the encouragement received in undertaking industrial consultancy.