Compact Display of Large User Models

James Uther
Faculty of Medicine, University of Sydney
A27B, University of Sydney
Sydney, Australia
+61 2 9351 7325
hemul@gmp.usyd.edu.au

Judy Kay
Dept. Of Computer Science, University of Sydney
F09, University of Sydney
Sydney, Australia
+61 2 9351 4502
judy@cs.usyd.edu.au

ABSTRACT

Scrutable (viewable) user models are known to aid learning through metacognitive reflection. In a large domain such as a university medical program, the user model grows too large for established user model viewing techniques. We present a mechanism for allowing a student to view and manipulate such a large model in limited screen space.

Keywords

Visual Representation, Modeling and Reconstruction, Interactive Exploration, Human Computer Interaction.

1. INTRODUCTION

The University of Sydney now teaches it's medical degree as a four year graduate course, using problem based learning. The course is supported by a web site supplying materials used in teaching medicine [1]. This web site also provides an 'online assessment' system the students can use to compare their knowledge to that expected by the faculty at any stage in the course. The faculty has agreed never to look at individual student marks in this system, so they can in no way affect the official marks of a student. This gives the students the freedom to try questions without fear of failure.

In order to provide useful performance data to the student, all answers are logged. These logs are then analyzed to generate user models for the students and various useful conjunctions of students. The models can then be used by individual students to see how they are going, or by staff to see how cohorts are going on their parts of the course. Given the size and complexity of the domain, it is difficult for the student to be able to see their overall progress through the learning topics, and to display the model on a normal screen along with other documents.

2. PREVIOUS USER MODEL VISUALISATIONS

User model viewers have generally run as stand alone applications on high

resolution displays [2,3]. They chose the acceptable course of displaying

models as trees, allowing expansion and collapsing of nodes and their children.

3. MODEL DOMAIN

3.1 Learning Topics

The first two years of the course are structured around a set of about 600 'learning topics', [Figure 1] concise summaries of about a lectures worth of material in a traditional course. Learning topics were submitted from across the university and cover topics ranging from basic sciences to sociology. They consist of a title, author, department, some keywords, a page or two summary of the topic, and a set of references that may be web sites, papers, books, specimens from the pathology museum, or anything else appropriate. Groups of learning topics are used as core content for each of the weekly 'problems' for the course, although discussions among the students during any week may lead them to studying any of the topics.

3.2 Online Assessment

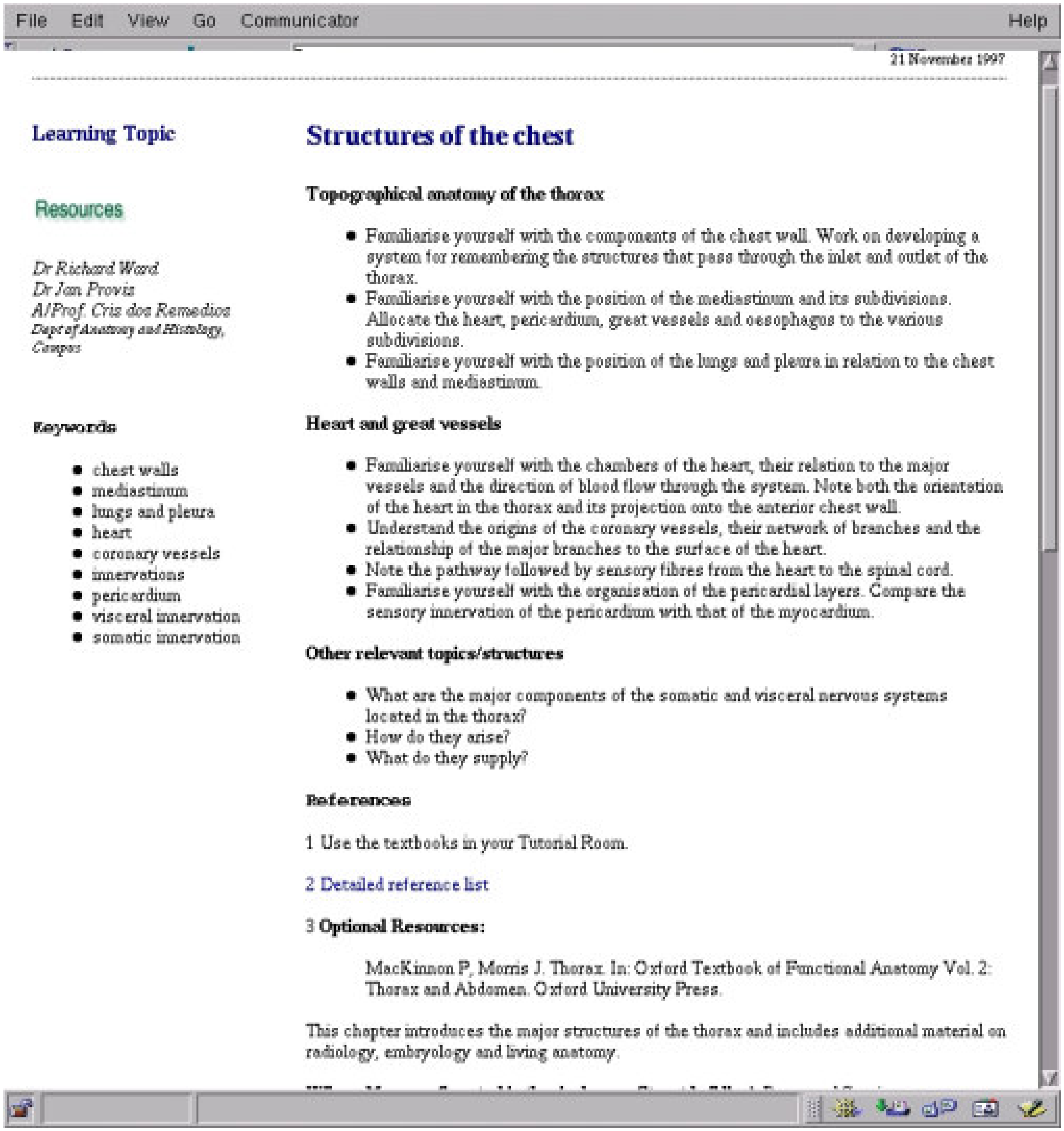

Figure 1: A Learning Topic

The author of a learning topic is asked to provide 10 questions relating to the topic. The questions may be in any of a number of styles such as true/false or multiple choice. These questions are used in the online assessment system to allow a student to test themselves against the expectations of the topic author [Figures 2, 3, 4]. By logging all responses the student makes to each question we can build a model of how much of the course the student understands by learning topic. By combining logs we can know how entire cohorts are managing, providing excellent feedback to the faculty on each of the topics, as well as feedback to students about where they may be falling behind their peers. The online assessment system is implemented as a Java servlet, with all data stored in a relational database.

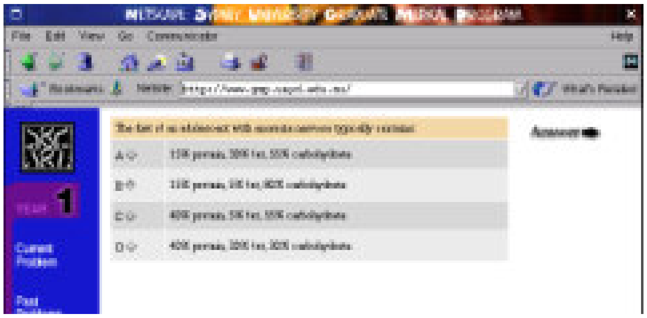

Figure 2: Asking a question in the Online Assessment system.

Figure 3: The question has been answered.

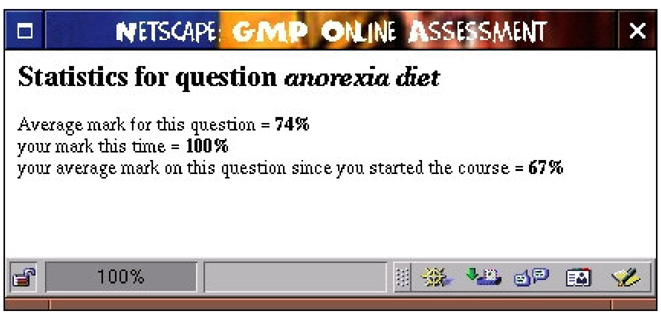

Figure 4: Some statistics.

3.3 Topic Relationships

We relate the learning topics to form a net. This provides a way for the student to move onto a reasonably well chosen 'next' topic once one has been studied. Learning topics can be related on a number of dimensions. We group topics based on the discipline to which their contained questions belong.

4. SOLUTION

We must allow the person viewing the model to get a good understanding of their progress easily. We must allow the person viewing to find detailed information about specific topics. We must show the relationships between topics. We must allow the viewer to run on a variety of platforms, possibly over modem connections, and on clients with reasonable processors (200Mhz) and XGA (1024x768) screen resolution.

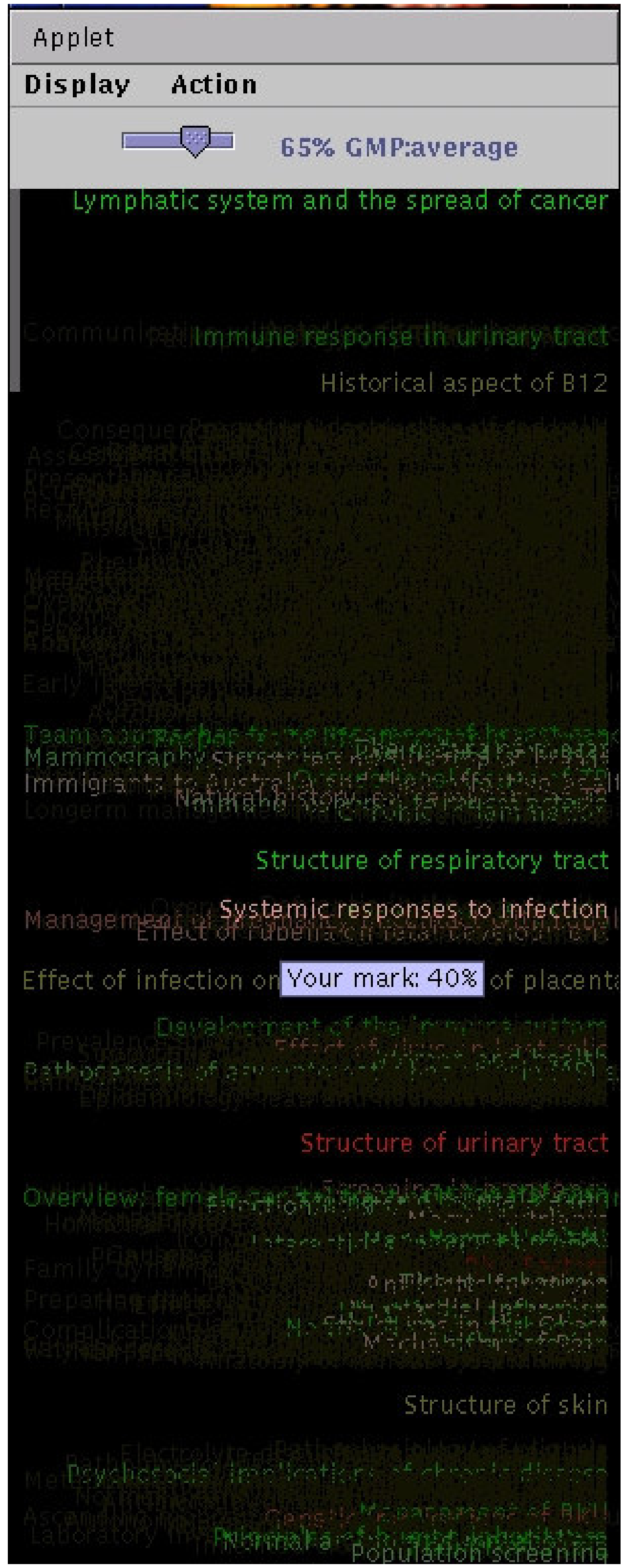

Figure 5: The model. Pass mark of 65%, displaying only the individual student marks. Selected topic at top, and space around related topics. ToolTip with detail.

Figure 6: Visual difference between low and high pass mark, or good and bad performance.

Figure 7: Mouse over shows hidden topics.

Figure 8: Stretching of closely spaced topics.

4.1 Client

The model viewer is a Java applet placed in one frame of a web page, beside a larger pane that can display a learning topic page, or online assessment questions relating to a learning topic. This leaves us with perhaps 300x550 pixels in which to display a model of 600 items each with additional information and relationships.

We used the Materials Listing display from DEXTER [5] as a starting point. DEXTER lays out a directed graph of names along the vertical axis, raising the currently selected item to the top of the list, and increasing both the surrounding space and brightness of related items. Different categories of items are shown by different hue. Some status bar information is shown about the item below the mouse on mouse-over. This approach allows the conceptual display of more items at once than there are physical vertical pixels by bunching unrelated items together and decreasing their brightness to remove them from attention.

We extended the concepts in DEXTER and adapted them to our needs. For instance, we do not have the need for categories, so we use hue to indicate the relative mark for a topic. Red indicates poor performance, green a better mark, and yellow indicates insufficient information [Figure 5]. The Saturation value of the colour is used to indicate distance of the mark from the pass mark for that topic. Standard tooltips from the Java Swing API are used to give more accurate data on each of the topics if desired. The topic under the mouse is brightened to allow reading even when dimmed [Figure 7]. If the student is interested in a topic obscured in a bunch of low relevance topics, they may drag surrounding topics up or down, thus expanding that bunch [Figure 8]. This allows access to all topics, regardless of relevance. A slider is provided to control the 'pass mark' used to separate red and green [Figure 6].

The models are sent over HTTP as RDF [4] files. RDF describes a directed acyclic graph of (entity - relationship - value) triples. At the time of writing all information is contained in one file. We plan to split this into a number of files to allow more flexible caching and decrease bandwidth needs.

4.2 Server

Generating the models from the log of answers is slow enough with the current database schema that we cannot send fresh information for each request. Instead we generate daily models for all students and cohorts each night. These are currently stored as text files and may be requested as needed through our web server. Simple access controls ensure that students cannot see the individual marks of others.

5. CONCLUSIONS

Although controlled user trials have not yet taken place, this method of displaying user models does seem to provide a groovy looking interface

6. REFERENCES

[1] Carlile S, Barnet S, Sefton A, Uther J. Medical problem based learning supported by intranet technology: a natural student centred approach. International Journal of Medical Informatics 50 (1998) 225-233.

[2] Cook R, Kay, J. The Justified User Model: A Viewable, explained user model. Proceedings of the Fourth International Conference on User Modelling (1994) pp 145-150

[3] Kay J. The um Toolkit for Cooperative User Modelling. User Modeling and User Adapted Interaction. (V2, no.3 1995) pp 149-196

[4] Lassila O, Swick R. Resource Description Framework(RDF) Model and Syntax Specification. W3C Working Draft 08 Oct 1998 http://www.w3c.org/TR/WD.rdf.syntax/

[5] Murtaugh M, The Automatist Storytelling System. Masters thesis, MIT Media Lab, 1996. http://ic.www.media.mit.edu/icPeople.hide/murtaugh.thesis/index.html