Innovating with Concept Mapping
Proc. of the Seventh Int. Conference on Concept Mapping
Tallinn, Estonia 2016

CONCEPT MAPS AS A MEANS OF UNIVERSITY TEACHING ASSISTANT

Mateo G. Lezcano Brito, Universidad Cooperativa de Colombia, Colombia,
Lydia R. Ríos Rodríguez, Universidad de Sancti Spíritus, Cuba
Yolanda Soler Pellicer, Universidad de Granma, Cuba
Email: mlezcano@gmail.com

Abstract. Concept maps are a valuable resource to support teaching, especially when this are used through a computer network or Internet. In this paper the concept maps to teach various subjects of IT careers are used. Maps nodes have various resources that support the teaching-learning process, many of them let you run simulations as an effective way to teach events that are not visible to the human eye. The subjects of study are: Operating Systems, Computer Networks, Computer Architecture, Logic Programming and Data Structures. The systems use various resources associated with the concepts, among them include: Web pages, images, text, videos and simulations outlined above. The system used to teach data structures has the added value to display, dynamically, data and programs of an algorithm programmed in a language specifically designed for this system, while the system to teach logic programming add functionality to adapt to the knowledge of students by including a set of intelligent agents that control the process of teaching and learning.

1 Introduction

The use of computers to teach began in early history of computing, the practice can be cited back to Skinner's Teaching Machine (Skinner, 1958) and the PLATO project (Bitzer, Braunfeld, & Lichtenberger, 1961), along with other results from that initial era.

To analyze the effectiveness of some of the systems of first stage of computer-assisted instruction, (CAI) several studies were made. For example, Suppes and Morningstar conducted a classic analysis based on the traditional model of an experimental group against a control group (Suppes & Morningstar, 1969). The authors cite various analyses, linking other important studies of that time, which were intended to measure the effectiveness of the systems of that time.

Naturally, the first computer teaching systems were gravely limited by the hardware of that era and the lack of computers; however, it was always thought that computers could be utilized as an aid for teaching or a means for teaching.

Ausubel's learning theory emphasizes the importance of previous knowledge on a topic in order to learn new things; this is known as meaningful learning (Ausubel, Novak, & Hanesian, 1978) and it is achieved when acquiring new knowledge is related with previous knowledge which relates new knowledge to previously acquired concepts or propositions, and these are managed explicitly and consciously (Novak, Gowin, & Johansen, 1983). Meaningful learning is contrary to memorization, since the latter doesn't form these connections.

Concept maps (Novak & Gowin, 1984) form a learning strategy that organizes interrelated concepts, supporting the ideas of meaningful learning. When the maps are accessible via computer networks, they can provide teaching resources and are an excellent option that supports the constructivist model proposed by Ausubel.

In this paper, several systems using concept maps to teach various subjects in the computer area are presented, and the systems presented use computer networks as a means of communication. The subjects of study are: Operating Systems, Computer Networks, Data Structures, Computer Architecture and Logic Programming.

2 Overview

Concept maps basically represent meaningful relationships established between two concepts to form propositions or simplified phrases (Cañas et al., 2000). Each concept is linked by a word to another concept to form a basic proposition; it can also form more complex propositions when browsing the map.

There are several tools to build concept maps, in this work the Cmap (Cañas et al, 2004) system that uses client-server architecture was chosen; where the client side is the CmapTools (or any browser), while the CmapServer is server side and has the responsibility to meet the requests generated from the connected clients, usually remotely. The choice of the tool was based on its versatility and because it is free for educational institutions.

Maps built with CmapTools can stay on a remote machine that has installed the CmapServer server or you can use the cloud service offered by Cmap; in the latter case it is not necessary to install the CmapServer. In both situations the services work well, but if Web pages cascading style (CSS) are used, it will not, the use of CSS style

is very useful because it allows separate structure and presentation of HTML, XML or XHTML documents (Duckett, 2011). For this reason, the systems presented in this article are managed from a server machine that has CmapServer as handler maps and all resources except the Web page that are managed by the Apache Web server.

2.1 Concept Maps as Teaching Support

The systems presented below were designed, advised or built by experienced teachers who relied on different degrees of student work, who had the responsibility to develop some of them locally. After a detailed review by the team, these were published with the purpose of being non-modifiable by learners; for that reason the students who use them are not authorized to make changes because the purpose is that learning is acquired when navigating well-designed maps, where the concepts and relationships that make up the propositions are true and they are formalized correctly. The systems are characterized by subliminal information that allows you to associate content covered with images showing background maps, or the header and footer of Web pages.

General aspects associated with all systems, such as colors and images that accompany each map should always be considered when a system of this type is designed. Moreover, it is very important to use simulations in the teaching of these subjects, because many of the processes that are studied are not visible to users and it is somewhat difficult to explain without adequate aid, the three systems that title this heading are aided in different simulations that fit each subject of study.

Each of the concepts or nodes in the map should be supported by additional information in the form of resources associated. We recommend using Web pages that give more beauty to what is explained and allow you to define links between different concepts, but too many link systems should not be used because students may feel lost when navigating. Simulations are also very useful because they allow students to appreciate invisible mechanisms that are difficult to explain verbally, which can be programmed in different languages.

The systems that were made following these ideas are called: SESO “System for teaching operating systems” (Garrido & Gonzalez, 2009), SERC “System for teaching computers networks” (Gonzalez, 2011; Brene, 2012), SEAC “System for teaching computer architecture” (Sori, 2011; Brene 2012), VIA-ED “System for teaching data structures” (Soler, 2009), APA-Prolog “System for teaching logic programming” (Ríos, 2009).

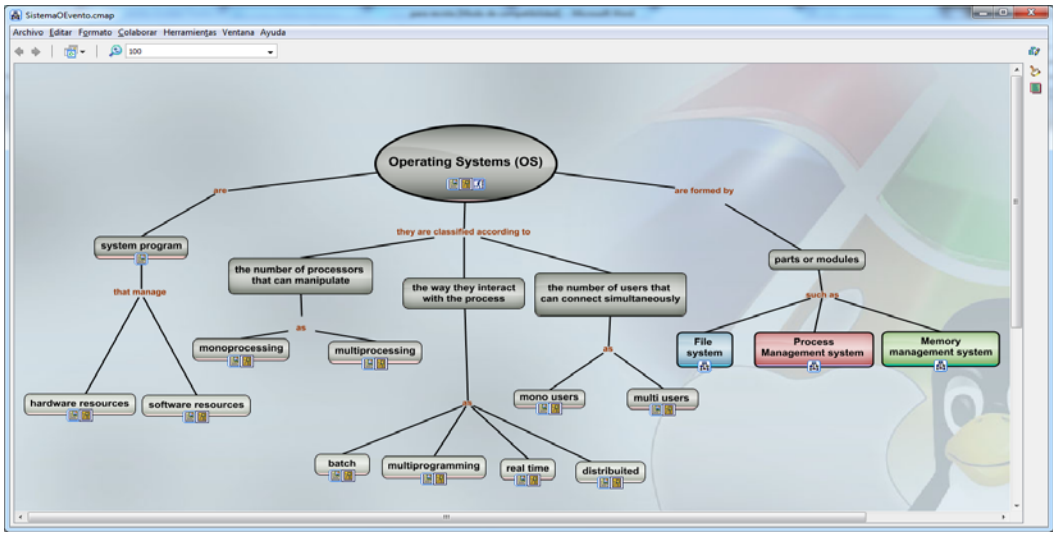

Figure 1. SESO System: main map.

Taking as an example the main map of the SESO system (Figure 1), you can see the graphic allusion to the three most used worldwide operating systems (Windows, Mac and Linux). It is observed that different colors are also used to distinguish three sub-maps that respond to three key modules of these systems, *the file system*, *the process management system*, *the memory management system*, each of these concept maps has a different color; nodes and resources derived from them must have the same color. Also, the background color of the sub-map should highlight the same color, albeit in different shades, in order to subtly remind the student what is being analyzed.

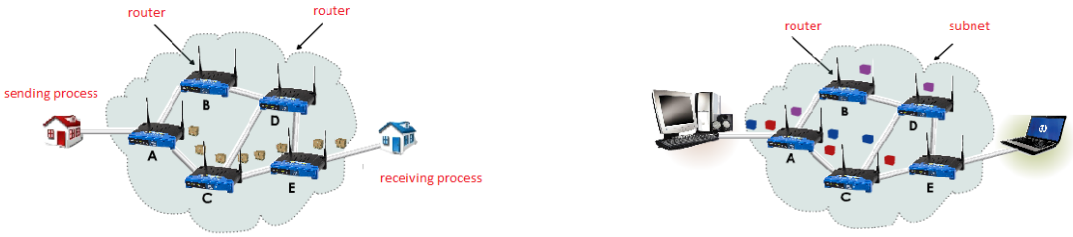

Figure 2. SERC System: simulating the transmission of a packet.

Figure 2 shows a simulation that is within one of the web pages of SERC system. The student reaches this simulation after having read an explanation about how packets are transmitted, the explanation is supported by various resources (graphics, images, etc.) displayed statically. At the end of the page the student can find and use the *play* and *stop* buttons to see how a packet is sent from a sender process to a receiver process. The packages on the left have taken the same route; while the packages on the right have followed different routes.

In Figure 3 one of the simulations SESO system can see. The simulation shows an example of a system that allocates memory spaces that need a process contiguously.

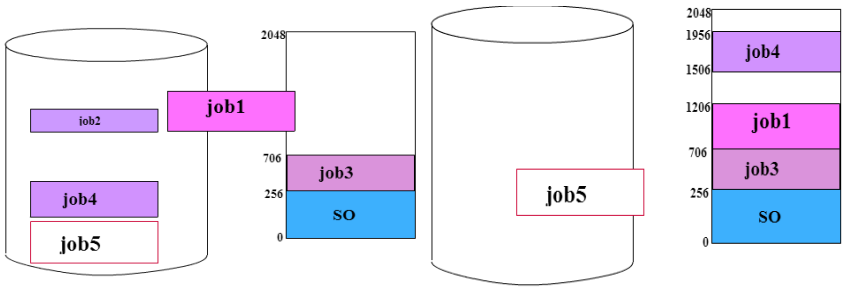

Figure 3. SESO System: contiguous memory allocation.

The simulation of figure 3 has been stopped in two instants:

- The left side of figure 3 shows the simulation when being loaded into memory a process called job1, the process called job3 had been previously loaded.
- The right side of the figure shows the moment when the OS is trying to load into memory the job5 process. The processes j3, j1, and j4 are in the memory at that moment. In addition, there are two non-contiguous free spaces which together could meet demand but as they are separated will not be usable.

Maps associated with VIA-ED system have resources similar to those discussed above and they are used with the same general design strategies of its components, but differ in one key aspect because it is based on three specially designed systems for this application (Soler, 2009): VisualProg a system to program visualization; SubC language and Pizal, a tool that can represent data graphically. The VisualProg system takes, as input, a program made in the SubC language.

The Pizal system can display the data structures associated with the program being analyzed, written in SubC, while VisualProg shows the program execution trace, all of which is very important during the learning process because students can appreciate the changes causing the code it is analyzed on the data structures used.

The APA-Prolog system is distinguished from others by being dynamic, which means that the elements associated with its nodes or concepts within the maps change according to the level of knowledge possessed by students who interact with the system. This is achieved by assigning the system a set of intelligent agents to control the behavior of resources (Ríos, 2000). The agents are:

- *Adis* performs an initial diagnosis to each student, based on several evaluative forms.
- *Tivo* offers students several links to additional information. His decision is based on identified deficiencies that are specific to each learner.
- *Teo* decides what theoretical materials are suitable for a particular student.
- *Tica* has the task of presenting practical activities, which are associated with the level of knowledge of each student.
- *Eva* is responsible for evaluating the student and deciding whether or not to change their state of knowledge. If it does decide to do so, it modifies and places them in epithets of values: good, average or poor.

3 Summary

Concept maps are a powerful tool to support teaching, their application in systems with client-server architecture provide students resources that are available at all times. They only need a network connection between the machine where the student resides and where the servers are (Apache and CmapServer, in this case).

Maps acquire greater value when you add learning resources that offer different ways of presenting the contents. Many of the problems studied in computer specialties are not visible; in that case simulations are an extremely valuable resource and learning assistance for discoveries.

The VisualProg and Pizal systems, coupled with SubC language, are an appropriate means for the study of programming, because the former allows showing the student the execution trace of a program written in SubC; which can be designed according to a target specific study, while the latter displays variations experienced by data structures used in the program.

The idea of using intelligent agents to teach, as is done in the APA-Prolog system is very useful for adapting teaching to the students' abilities, so everyone goes at the pace that suits their characteristics.

References

Ausubel, D. P. Novak, J. D., & Hanesian, H. (1978). Educational Psychology: A Cognitive View (2nd edition). New York: Holt, Rinehart & Winston.

Bitzer, D., Braunfeld, P., & Lichtenberger, W. (1961). PLATO: An Automatic Teaching Device. IRE Transactions on Education, 4(4), 157 - 161.

Brene, E. (2012). Mapas Conceptuales para la Enseñanza de Redes y Arquitectura de Computadoras. Trabajo de Diploma, Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba.

Cañas, A. J., Ford, K. M., Coffey, J., Reichherzer, T., Carff, R., Shamma, D., & Breedy, M. (2000). Herramientas para Construir y Compartir Modelos de Conocimiento basados en Mapas Conceptuales. Revista de Informática Educativa, 13(2), 145-158.

Cañas, A. J., Hill, G., Carff, R., Suri, N., Lott, J., Eskridge, T., Lott, J, Carvajal, R. (2004). CmapTools: A Knowledge Modeling and Sharing Environment. In A. J. Cañas, J. D. Novak & F. M. González (Eds.), *Concept Maps: Theory, Methodology, Technology*. Proc. of the First Int. Conference on Concept Mapping (Vol. I, pp. 125-133). Pamplona, Spain: Universidad Pública de Navarra.

Duckett, J. (2011). HTML and CSS: Design and Build Websites. Indianapolis: John Wiley & Sons.

Garrido, D., & González, L. (2009). Mapas Conceptuales para la Enseñanza de Sistemas Operativos. Trabajo de Diploma. Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba.

González, Y. (2011). Sistema de Mapas Conceptuales para la enseñanza de Redes de Computadoras. Trabajo de Diploma. Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba.

Novak, J. D., & Gowin, D. B. (1984). Learning How to Learn. New York: Cambridge University Press.

Novak, J. D., Gowin, D. B., &Johansen, G. T. (1983). The Use of Concept Mapping and Knowledge Vee mapping with Junior High School Science Students. Science Education 67(5), 625-645.

Ríos, L. (2009). Ambiente de Enseñanza-aprendizaje Inteligente para la Programación Lógica. Tesis para optar por el Grado Científico de Dra. en Ciencias Técnicas, Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba.

Skinner, B. F. (1958). Teaching Machines. Science 128, 969-77.

Soler, Y. (2009). Aplicación de la Visualización Dinámica de Programas en el Diseño de Estructuras de Datos y el Análisis de la Complejidad de Algoritmos. Tesis para optar por el Grado Científico de Dra. en Ciencias Técnicas. Universidad Central “Marta Abreu” de Las Villas, Cuba.

Sori, J. (2011). Mapas Conceptuales para la Enseñanza de Arquitectura de Computadoras. Trabajo de Diploma, Universidad Central “Marta Abreu” de Las Villas, Santa Clara, Cuba.

Suppes, P., & Morningstar M. (1969). Computer-Assisted Instruction. Science, 166, 343-350.