

Designing a ubiquitous computer environment for medical education

Hyung Hoi Kim^{1,2}, Tung Tran¹, Hwa Sun Kim¹, and Hune Cho¹

¹*Dept. of Medical Informatics, Kyungpook National Univ., Republic of Korea*

²*Dept. of Laboratory Medicine and Biomedical informatics, Pusan National Univ., Republic of Korea*
hhkim@pusan.ac.kr; ttadm20031@yahoo.com; polala@knu.ac.kr; hunecho@knu.ac.kr

Abstract

Ubiquitous access to computers and networks has led to major gains in the integration of technologies into the teaching process. The laptop requirement was initiated to produce a new generation of teachers who are well prepared to use the new tools for learning in their teaching practice. It was designed to immerse pre-service teachers in a technology-rich learning environment that provides ubiquitous anytime/anywhere access to technology tools, Internet-based resources, and an online communication environment to support and build communities of practices among all players.

1. Introduction

Various learning systems have been developed in the past decade, and the majority of these systems are implemented with a client-server or are based on a centralized server. The client-server and centralized server approaches are metaphors for student-teacher and repository-centric, which reflect real-world learning scenarios in which teachers act as the content producers, while students act as the content consumers.

The ubiquitous learning environment provides mainframe computing, which is pervasive and interoperable. Ubiquitous computing is the universal interconnection of virtually any wireless/wired device nationwide, and beyond, a condition gradually being realized by ongoing advances in the Internet, smart appliances, and related technologies. Ubiquitous communication tools can be anything from mobile devices to tiny communication-capable sensors connected to high-speed networks, enabling information to be accessed and sent anytime, anywhere, in other words, ubiquitously. Today, with the development of wireless connectivity, mobile devices such as laptops, personal digital assistants (PDAs), and phones are now able to communicate from almost anywhere, inexpensively and at very high speeds, giving users great freedom in choosing when, where, and how to be connected.

The evolution of education and training at a distance can be characterized as a move from d-Learning (distance learning) to e-Learning (electronic learning) to m-Learning (mobile learning). These three stages of

development correspond to the influence of the Industrial Revolution of the 18th to 19th centuries, the Electronics Revolution of the 1980s, and the Wireless Revolution of the last years of the 20th Century on society[1, 2].

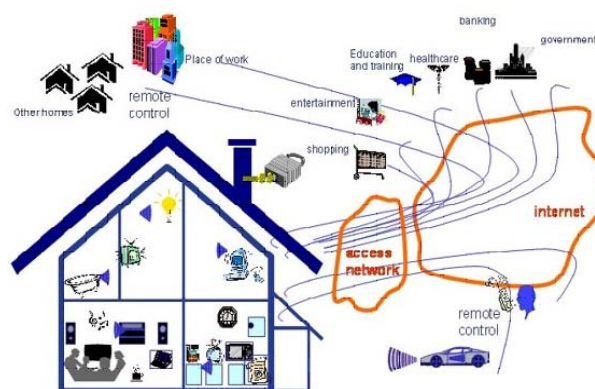


Figure 1. The ubiquitous computing model.

Currently, as more teachers and students use laptops or handheld computers on a daily basis, it is essential that policy makers and practitioners alike understand the effect that ubiquitous computing has on the classroom environment.

This paper presents the architecture of a prototype ubiquitous application for medical education. This paper is organized as follows. In the first sections, we present the general design principles used to specify the system's architecture, and the second section, we developed XML schemas that can be used to create and validate general XML documents for encoding essential data elements required for medical educations. The third section describes the process of generating XML schemas for educational templates. The next section reviews some of the issues encountered when adapting previous approaches to include additional template constraints. Finally, the fifth section includes conclusions and discussion for further work.

2. Materials and Methods

2.1 Design of system architecture

Ubiquitous computing and pervasive computing are emerging disciplines linking elements from distributed systems, mobile computing, embedded systems, human



computer interaction, computer vision, and many other fields. Their vision is grounded in the belief that processors are becoming so small and inexpensive that they will eventually be embedded in almost everything. More specifically, the system will interface with the devices shown in Figure 2. In Figure 2, the Web server is a computer, which has Tomcat version 5.5.17 installed as a Web server application. Therefore, we can develop servlets that allow students or users to access information and carry on discussions with other users. We developed web modules that can obtain information from a knowledge base as the user requires. Then, it formats the retrieved information into the following XML schema [3]. In addition, the Extensible Markup Language (XML) data are transformed to HyperText Markup Language (HTML) using Extensible Stylesheet Language (XSL) to facilitate viewing on PDAs, notebooks, and desktop computers [4].

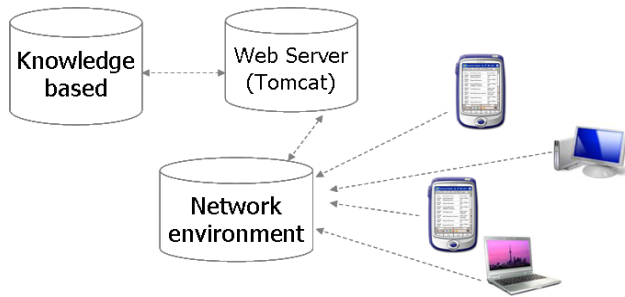


Figure 2. Ubiquitous system architecture.

Figure 3 is expressing the system architecture for medicine education based knowledge. In this architecture, client use web browser to send a request to server. And the event sender requires a reply. For example, user wants to know heart information. In the server side, Broker receive event from client request and lead Metadata operations module to query data from database.

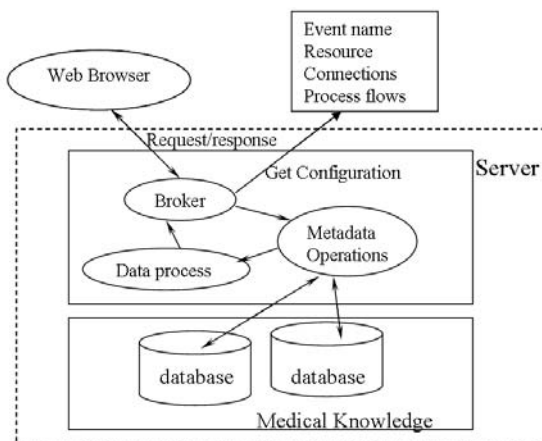


Figure 3. Education system architecture.

The database contains medical knowledge. The database receive query and return information according commands. The metadata operations module receives

data from database, and sends them to data process module. The data process module receives data and generates data to XML document following XML Schema. XML Schemas describe the structure and meaning of an XML document, in much the same way that a class describes an object in a program. To work with an XML document in a program we would like to map its components directly to a set of objects that reflect the document's meaning according to its schema. We can achieve this by compiling the schema into a set of derived content classes that can be validated. Data binding thus allows XML-enabled programs to be written at the same conceptual level as the documents they manipulate, rather than at the more primitive level of parser events or parse trees. Eventually, data process transform XML document to HTML using XSL and send it as pure HTML to browser.

2.2 XML schema for medical education

The W3C designate XML as the standard for the next –generation web document in 1998. XML is essentially a language describing another language, that is, a meta-language. XML separates the structure of the document into three parts: the part defining the structure of the document, the form of the expressing document, and the composition of the document. Thus it is important to composing the frame of the document, and Data Type Definition (DTD) solves this. However, because of the weak point that the validation of the data provided through the limited data type is impossible, the XML schema was emerged as new standard. Furthermore, XSL, XSLT and others through which transforms and transmits the mere XML documents suited to the environment of the user are considered as the related standards. Here the XSLT processor shares XPath which applies XSL to XML and with the result creates trees and can reads the desired contents using query [5].

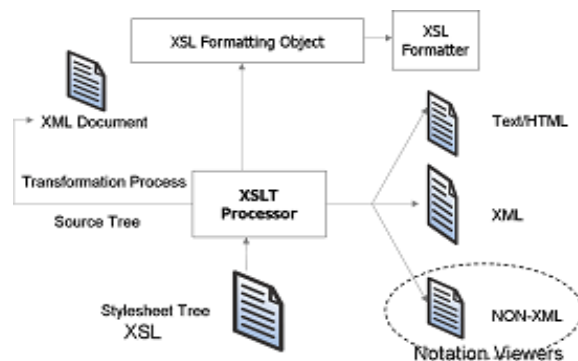


Figure 4. XML Transformation process.

Xpath provides the structural addressing mechanism for XSLT and the elements of the Xpointer-based XML documents is expressing the XML standard techniques



mentioned above and the object-oriented database storage which easily holds the meta-data of the XML document form was added. This indicates that the stored XML is shown through the web interface employing XSL, and the tasks are easily carried out through Xquery.

We can access knowledge bases for querying and obtaining information, anytime and anywhere. A personalized repository organizes the information that interests us, keeps track of how we access that information, and contains annotations about its relevance, quality, and comprehensibility. We access our repositories in our own ways. When we want more information, we can access repositories that trusted friends and associates are willing to share. Currently, the knowledge base stores information on more than 100 diseases, such as heart disease, kidney disease, and lung disease [6, 7].

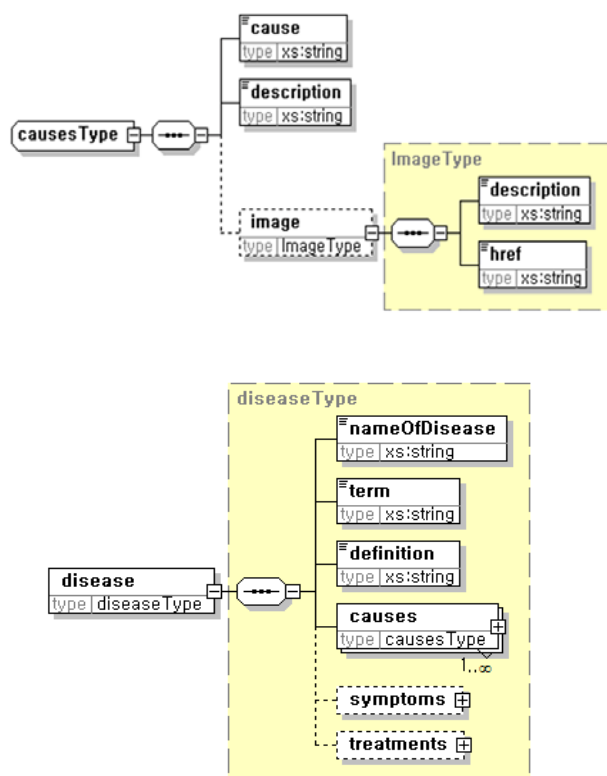


Figure 5. XML schema for medical education

We used XML Spy to display the details of the schema for the content tree, as illustrated in Figure 5, which is a useful way of visualizing schema fragments. The schema <causesType>, <disease> and its children elements serve as the meta-model for the schema of the templates.

2.3 Data Extraction Module

In the case of Oracle 9i, there needs to be a process that maps XML elements and attributes them to the table

field. Since the XML document creates tables according to its complexity, even a single XML needs table mapping. A tree-type document object model (DOM) is used for extracting data from the relational database that supports open database connectivity (ODBC) [8].

Application programs are provided so that common words can be defined using internal DOM application programming interface (API) when storing data. Where a need occurs for a standard method of interaction with the data, DOM allows access to the content regardless of the structure, providing an object model for modeling an XML document. The discussion in the last section describes the basic XML schema for medical education template generation. The process consists of Conversion and Transformation. Conversion of template tables is an intermediate XML representation that attempts to recover some of the structural information inherent in the templates (Figure 6).

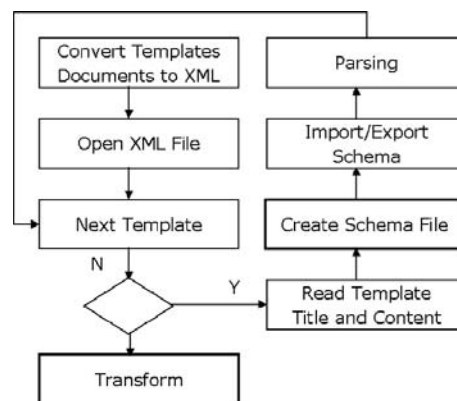


Figure 6. XML schema generation process.

This step is necessary because templates are presently defined with text and tables in a Microsoft Word document. Transformation of XML representations to XML schemas is accomplished with a JAVA program. This can also be accomplished with XSLT transformations.

3. Results

The knowledge access subsystem supports the natural ways people access information. In particular, it supports personalized, collaborative, and shared knowledge, to find the information that people need. It observes and adapts to its users, to meet their needs better. The subsystem integrates a variety components to gather and store data, to monitor user access patterns, and to answer queries and interpret data.

Data representation: The subsystem stores information encountered by its users using an extensible data model that links arbitrary objects via arbitrarily named arcs. There are no restrictions on object types or names. Users and the system alike can aggregate useful information regardless of its form (text, speech, images, video). The



arcs, which are also objects, represent relational (database-type) information, as well as associative (hypertext-like) links.

Data acquisition: The subsystem gathers as much information as possible about the topic of interest to a user. It does so through the raw acquisition of data objects, by analyzing the acquired information, by observing people's use of it, by encouraging direct human input, and by tuning access to the user.

Automated observers: Subsystems watch the queries that users make, the results they dwell on, the files they edit, the XML they send and receive, the documents they read, and the information they save. The system exploits observations of query behavior by converting query results into an XML document according to the XML schema, and using XSL to transform it into HTML to represent information (Figure 7, 8). The subsystem generates the results as an XML document after aggregating information from the database. In addition, it also creates an XSL style sheet in order to describe how the XML document should be displayed.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:fo="http://www.w3.org/1999/XSL/Transform" >
  <xsl:template match="diseaseCollection">
    <center>
      <h1 style="color:blue">{xsl:value-of select="disease/nameOIDisease"/}</h1>
      <p>Medical Terminology: {xsl:value-of select="disease/term"/}</p>
      <p>What is a {xsl:value-of select="disease/nameOIDisease"/}</p>
      <blockquote>{xsl:value-of select="disease/definition"/}</blockquote>
      <br/>
      <p>What causes a heart attack?</p>
      <blockquote>
        <ul>
          <xsl:for-each select="disease/causes">
            <li><p>{xsl:value-of select="cause"/}</p>
            <p>{xsl:value-of select="description"/}</p>
            <img alt="Image description" data-bbox="528 135 855 370" style="width: 100%; height: auto; border: 1px solid black; border-radius: 50%; margin: 10px 0;"/>
          </xsl:for-each>
        </ul>
      </blockquote>
      <br/>
      <p>What are the symptoms?</p>
      <blockquote>
        <ul>
          <xsl:for-each select="disease/symptoms/symptom">
            <li>{xsl:value-of select="symptom"/}</li>
          </xsl:for-each>
        </ul>
      </blockquote>
    </center>
  </xsl:template>
</xsl:stylesheet>
```

Figure 7. XSL for medical XML document.

In the existing medical education sites on the web, there are education data in their own way. Most of the data are stored on the certain tables of the DBMS systems. They are providing the lectures to the users through transforming the data into the form of HTML by using the server-side scripts of ASP, JSP, and PHP and sending them to the browser. In the XML system, XML schema is located on the outside, not in the lecture sites in which XML and XSL reside. In other words, the definition for the lecture data is specified by another standard enterprise. DOM loads XML and XSL located in the education sites, and before loading XML, this XML data tests the validity according to XML schema specified by the standard enterprise, and XSL is applied to XML data by the DOM's XSLT processor, with which result HTML is created and transmitted to the client browser by the web server, so that there happens no blockade in the data sharing.

the higher blood levels of adrenaline released from the adrenal glands during the morning, discussed, may contribute to rupture of cholesterol plaques. Approximately 50% of patients with warning symptoms such as exertional angina or rest angina prior to their heart attacks.

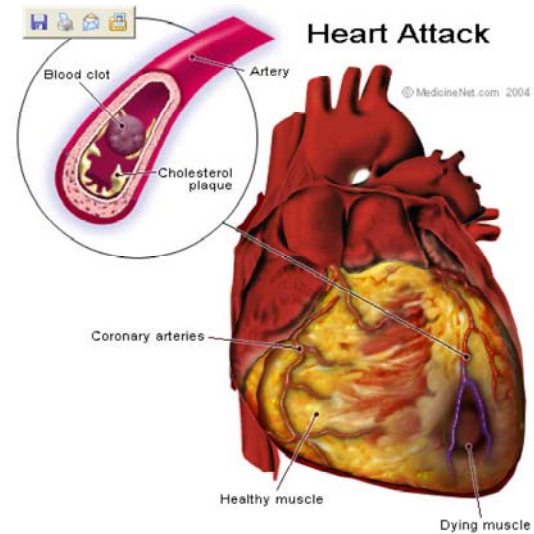


Figure 8. XSL describes how the XML document should be displayed.

4. Conclusion and Future works

Ubiquitous computing leads to ubiquitous learning and technology provide abundant opportunities for sharing information, constructing knowledge, and stimulating personal growth. It is pervasive and prevalent in many interactions among students, faculty, administration, and staff. Web-based learning is currently showing its technological potential as an environment for learning. In recent years, medical schools have responded in a variety of ways to the challenges of modernizing traditional disciplinary-based courses [9-12]. The trend has been move to various integrated curriculum models, including problem-based learning (PBL). The availability of computer-based systems to support curriculum management and the delivery of learning materials (e-learning), may provide on way of supporting students undertaking PBL without increasing existing teaching loads.

In the future, we will extend the system so that it provides more clinical data, and users who are connected to the server can discuss it in real time.

The system proposed and implemented in this study bears the following characteristics:

- Ability to understand and solve problems in healthcare for students can be expected to increase.
- Students participation to the discussion in the cyber classes can be expected to increase compared with the traditional methods



- If two research teams are successful in developing and implementing course material by using PBL approach, they can apply similar approaches to other courses offered by university for healthcare.
- Knowledge and information sharing can be achieved for the joint research team.

Acknowledgements

This work was supported by the Brain Korea 21 program from the Ministry of Education and Pusan National University Research Grant, 2005.

References

- [1] B. Van Grimbergen, G. Thienpont, G. J. De Moor, and H. Buysse, "Test-based E-Learning: a new path in training medical students," *Stud Health Technol Inform*, vol. 93, pp. 33-8, 2002.
- [2] J. M. Renard, S. Leroy, H. Camus, M. Picavet, and R. Beuscart, "Personalising e-learning modules: targeting Rasmussen levels using XML," *Stud Health Technol Inform*, vol. 95, pp. 732-7, 2003.
- [3] P. Spencer, *Professional XML Design and Implementation*, Wrox, Chicago, IL, 1999.
- [4] Extensible Style-sheet Language, available at: <http://www.w3.org/Style/XSL/>, Dec. 2004.
- [5] Developer's Guide – Chapter 4 XML Query Structure, available at : <http://umlsks.nlm.nih.gov/>, June 12, 2006.
- [6] Information technology, Knowledge base, available at : <http://kb.iu.edu/>, July 17, 2006.
- [7] eMedicine Clinical Knowledge Base, available at: <http://www.emedicine.com/>, May 23, 2006.
- [8] R. Ramakrishnan and J. Gehrke, *Database Management Systems*, McGraw-Hill International edition, New York, NY, 2002.
- [9] K. Schilling, J. Wiecha, D. Polineni, and S. Khalil, "An interactive web-based curriculum on evidence-based medicine: design and effectiveness," *Fam Med*, vol. 38, pp. 126-32, 2006.
- [10] J. G. Ruiz, M. J. Mintzer, and R. M. Leipzig, "The impact of E-learning in medical education," *Acad Med*, vol. 81, pp. 207-12, 2006.
- [11] P. Moule, "E-learning for healthcare students: developing the communities of practice framework," *J Adv Nurs*, vol. 54, pp. 370-80, 2006.
- [12] I. Heetebry, M. Hatcher, and H. Tabriziani, "Web based Health Education, E-learning, for weight management," *J Med Syst*, vol. 29, pp. 611-7, 2005.

