CSC 560: Management of Semistructured Data
Fall 2006
Course Syllabus

September 14, 2007

Instructor: Alexander Dekhtyar
email: dekhtyar@csc.calpoly.edu
office: 14-209

What	When		Where
Lecture	TR	9:10 – 11:00pm	22-218 (English)
Final Exam	Dec 4 (T)	10:10 - 1:00pm	22-218 (English)

Office Hours

	When	Where
Tuesday	11:00am - 12:00pm	14-209
Wednsday	9:00am - 12:pm	14-209
Thursday	1:00 - 2:00pm	14-209

Description

The course will be devoted to XML, its uses in the field of databases and data management and various aspects of working with XML data. We will try to concentrate on two things: (a) studying the ways of storing XML data in various kinds of databases (relational, native XML) and (b) querying XML data using XPath and XQuery. Hands-on experience with building software for management of XML data and for working with XML-supported applications will be provided throughout the course.

This is an advanced graduate-level seminar. I expect to learn from it nearly as much as I expect to teach.

Textbook, Readings

The class does not have an official textbook.

The first part of the course will be mainly taught using World Wide Web Consortium (W3C) standards documents and research papers and overviews. The materials we will be using are either publically available (e.g., W3C documents) or will be handed out in class and/or posted on the course web page.

For the second part of the course, I recommend the following book on XQuery:

> *XQuery from the Experts: A Guide to the W3C XML Query Language*, Howard Katz (ed.), D. Chamberlin, D. Draper, M. Fernandez, M. Kay, J. Robie, M. Rys, J. Simeon, J. Tivy, Ph. Wadler, 2004, Addison-Wesley.

Other books on XQuery exist, and can be purchased and used instead of the text above. (The key advantage of the recommended text is that it is written by the authors of the W3C recommendation, and includes a lot of explanations and historical perspective that makes certain aspects of XQuery more understandable).

W3C documents are available through its web page. We will use them to study the formal definition of different XML-related standards. In particular, expect to spend some time looking through the following W3C standards:

- XML 1.0. (Fourth Ed.), T. Bray, M. Paoli, C.M. Sperberg-McQueen, E. Maler, F. Yergeau (Eds.), http://www.w3.org/TR/2006/REC-xml-20060816/.
- XPath 1.0., J. Clark, S. DeRose (Eds.), http://www.w3.org/TR/xpath.
- XQuery 1.0., S. Boag, D. Chamberlin, M. Fernandez, D. Florescu, J. Robie, J. Semeon (Eds.), http://www.w3.org/TR/xquery/.

We may also look briefly at W3C documents on XSchema, XSLT and XPath 2.0.

Topics

This course takes a database perspective on XML and its uses. It will NOT cover ALL possible applications of XML and all possible standards and solutions available for XML. On the other hand, it will give you a hands-on perspective of how XML is incorporated into modern Database Management Systems, and how it is used in modern database applications.

The approximate list of topics is:

No.	Topic	Duration (weeks)
1.	Introduction: XML and Semistructured Date?	1-2
2.	Indexing XML	2-3
3.	XML Query Languages: XPath and XQuery	3-4
4.	Other topics (mining of XML data, document-centric XML, XSLT, filtering)	0-3

Grading

Homeworks	15%
Programming Project	50%
Reports	30%
Class Participation	5%

Prerequisites

I expect everyone to be reasonably familiar with the following:

- relational database model, SQL;
- context-free grammars, Backus-Naur notation (you should have had it covered it in some undergraduate programming languages or theory course);
- trees (the type of a data structure);

Course Policies

Exams

We have a scheduled time to meet for the final exam (Tuesday, Dec 4, 10:10am - 1:00pm). There will be no written exam for the course. However, there will be a group project which will involve an oral presentation. We may use the final exam time for oral presentations, or for any other end-of-semester reporting activity.

Project

The course will be built around a(n almost) quater-long group project. The exact details of the project will be announced within the first 2-3 weeks of the course; the rest of the quater, the project will proceed in stages, each, 2-3 weeks long. You are responsible for forming groups (2-3 people each) and distributing all project work among group members. Outside of truly extraordinary situations (of which I am informed well ahead of time), each member of the team will recieve the same grade for the project.

The project will involve building an XML-supported database application. We will use a database management system MonetDB as the back-end. MonetDB is an open source single-user DBMS, with both an SQL and a fully compliant XQuery query processor front ends. Each team will run its own copy of MonetDB (details of the technical setup will be distributed in the first two weeks of classes). We will only be using MonetDB in conjunction with its XQuery processor.

Homeworks

We will have occasional paper-and-pencil and/or small programming homeworks. While the project will concentrate more on the application development with XML, the homeworks will cover the main theoretical and algorithmical knowledge studied in class. All homeworks (regardless of whether paper-and-pencil or programming) are individual.

Reports

In this course, you are expected to read quite a few papers, and also, to write some, and, possibly, make an oral presentation. This is done to allow your interests to diverge from the content of the course.

You will write a short survey of literature on a particular topic related to XML/Semistructured data. If you have a topic in mind, discuss it with me. If you don't, a topic will be assigned to you. The assignment will involve three components: literature search, reading selected papers and writing a survey. The number of papers you need to read will depend on the topic and the size of the papers and will be discussed with me during the course. The reports are individual assignments.

Important! All writing assignments will be treated as academic writing. This means that the papers you write will be required to adhere to the same writing standards as the published papers and technical reports. More detailed specifications will be given at the time assignments are distributed.

Late Submissions

Typically, writing assignments are due in class and programming assignments are due at midnight of due date with a customary grace period extending until the morning of the day after. Exact programming assignment/project submission instructions will be included with the assignments. Be sure to follow exactly the submission procedures. Homework/projects submitted later than indicated will be considered *late submissions*.

Late ***writing assignment*** *submissions* will get 10% off if submitted within 24 hours of the due date and time. They will get up 30% penalty if submitted before the next class (more if the next class is a Tuesday, less, if it is a Thursday). No other submissions will recieve any credit.

Late ***project*** *submissions* are strongly discouraged. A penalty of 10 - 30% will be assessed for any submissions that are late by less than 24 hours. No credit will be given for any later submissions. You are encouraged to submit your code on time even if it is not perfect. You can then debug your code and submit a fixed version late, subject to the abovementioned rules. When more than one submission is present, I will independently grade two submissions: (i) the latest on-time submission and (ii) the latest late submission for which non-zero credit can be assessed. Your grade for the project will be the **maximum** of the two grades.

If there are some extraordinary circumstances that you feel may prevent you from submitting homework/project on-time, especially, if it is a group assignment, I want to hear about it **as early as possible**. If the circumstances warrant it, we may find a way to resolve the coursework issues, but only if I have advance warning. (I know that sometimes, it is impossible to give an advance warning, but if someone has to leave town for 2 weeks due to an emergency, I need to know that before they leave, not after they come back).

Web Page

Class web page can be found at

http://www.csc.calpoly.edu/∼dekhtyar/560-Fall2006

Through this page you will be able to access all class handouts including homeworks, project information, reading materials and lecture notes (should the latter be written).

Academic Integrity

University Policies

Cal Poly's Academic Integrity policies are found at

http://www.academicprograms.calpoly.edu/academicpolicies/Cheating.htm

In particular, these policies define *cheating* as (684.1)

> "*...obtaining or attempting to obtain, or aiding another to obtain credit for work, or any improvement in evaluation of performance, by any dishonest or deceptive means. Cheating includes, but is not limited to: lying; copying from another's test or examination; discussion of answers or questions on an examination or test, unless such discussion is specifically authorized by the instructor; taking or receiving copies of an exam without the permission of the instructor; using or displaying notes, "cheat sheets," or other information devices inappropriate to the prescribed test conditions; allowing someone other than the officially enrolled student to represent same.*"

Plagiarism, per University policies is defined as (684.3)

> "... the act of using the ideas or work of another person or persons as if they were one's own without giving proper credit to the source. Such an act is not plagiarism if it is ascertained that the ideas were arrived through independent reasoning or logic or where the thought or idea is common knowledge. Acknowledgement of an original author or source must be made through appropriate references; i.e., quotation marks, footnotes, or commentary."

University policies state (684.2): "Cheating requires an "F" course grade and further attendance in the course is prohibited." (appeal process is also outlined, see the web site above for details.). Plagiarism, per university policies (684.4) can be treated as a form of cheating, although a level of discretion is given to the instructor, allowing the instuctor to determine the causes of plagiarism and effect other means of remedy. It is the obligation of the instructor to inform the student that a penalty is being assessed in such cases.

Course Policies

First, all traditional warnings concerning cheating apply in this course. In particular, solicitation of help from people not involved in the course and submission of materials/code etc.. not developed by you are absolutely prohibited. Any outside materials used in preparation of homeworks, reports, project assignments must be properly documents. For example, you must properly cite all papers you refer to, all web resources used in perparation. You must also note any open source, off-the-shelf, etc. . . software or code fragments that you have incorporated in your solution. If you have questions concerning allowable use of such materials, please consult me **in advance**.

For example, if an assignment is to design and implement an XML parser, you are supposed to build one from scratch and not use any available parser code (which is plentiful). On the other hand, if you want to use an open-source library, or some code developed by one of the team members prior to the course as part of a project solution, this may qualify as allowable use, if the code is used in support of the main tasks of the project.