CSEP 545

Transaction Processing for E-Commerce

Course Information

Spring (March – May) 2007

Instructor

Philip A. Bernstein, philbe@microsoft.com, (425)706-2838

http://www.research.microsoft.com/~philbe/

Meeting Time

- Thursday 6:30 – 9:20 PM, March 29 – May 31
  - No class on May 10 and May 17
  - Make up lectures April 19, April 30 (tentative)
- A one-hour project demo, May 28-June 6 (exact dates TBD)
- Mid-term exam, probably May 3

Textbook

- Handouts of the revised version of:

  “Principles of Transaction Processing for the System Professional,”
  by Philip A. Bernstein and Eric Newcomer,
  Morgan Kaufmann Publishers, 1997

- Hard copy only. *PLEASE honor the publisher’s copyright.*

- Comments and corrections would be welcome.

Teaching Assistant

- Abhay Kumar Jha,

  abhaykj@cs.washington.edu
  (206) 616-7732
  Office: CSE 502

Web Site and Mailing Lists

- http://www.cs.washington.edu/education/courses/csep545/07sp
- Mailing List

  http://mailman.cs.washington.edu/mailman/listinfo/csep545

Please subscribe immediately to this mailing list

Lecture Outline

Below is the list of course topics. The exact order of topics may vary slightly from this sequence. Also, the mapping of topics to course meetings is hard to predict, since topics are of different lengths and some class time will be consumed by discussions of assignments and projects.

- Introduction (textbook, Chapter 1)
- Shadowing-based recovery, in support of the project (Chapter 7, Section 6).
- Database Concurrency Control, Part 1 (Chapter 6, Sections 1-4)
- Database Recovery (Chapter 7)
- Basic Application Servers, in support of the project.
- Two-Phase Commit (Chapter 8)
- Database Concurrency Control, Part 2 (Chapter 6 Sections 4 - 11)
- Queuing (Chapter 4)
- Replication (Chapter 10)
- Application Servers (Chapters 2 and 3)
- Advanced topics: T.B.D.

Prerequisites

- There are no specific prerequisites for the course, besides having a good general understanding of software systems.
- It’s also helpful to have some knowledge of SQL (to be able to write simple queries).

Assignments and Grading

- There are two components to your grade:
  - There will be a term exam in class, probably on May 3.
  - You will do a substantial two-person project.

- Your grade will be a weighted sum of the exam (20%) and project (80%).

- There will be weekly assignments for much of the course, covering material that’s easiest to learn by solving structured problems. The solution to each assignment will be discussed in lecture one week after the assignment is handed out. The assignments will not be graded. However, we recommend that you make a serious attempt at solving them, since it’s an effective way to learn the material. Some of the exam questions will be minor variations of some of the assignment questions.

Project

Transaction processing is a systems engineering problem, with many interacting parts. The goal of the project is to deepen your understanding of how the parts fit together. It involves implementing a distributed transaction system in Java or C#. The server will allow customers to make/cancel/query travel reservations, will allow airlines to add/cancel/modify flight information, and will allow hotels and car rental agencies to post room and car availability. We will provide clients, the interface the clients expect, a lock manager, the basic application, and a general design framework for the (ultimately distributed) server. Students will build mechanisms for persistence, recovery, and two-phase commit. The transaction system will concurrently support multiple clients and multiple servers.

Make sure you have a solid working solution of basic features, and save a snapshot of that code, before adding more advanced features. Only a short write-up is required to summarize what you have done and to provide a blueprint of the code, as long as the code is readable. A final demonstration is required.

Except in rare cases, people will work in groups of two. It helps you, to have design discussions with a partner. And it helps us, to make our reviewing load manageable.

Project (cont’d)

Due dates:

- April 5 – Email message telling who you will be working with and which language (Java or C#).
- April 19 – Milestone One. You should have at least the first couple of steps implemented.
- May 10 – Milestone Two. You should have at least half of the final project features running. Hand that in, with enough of a write-up that Abhay can understand what’s in your code and what you have accomplished.
- May 31 – Final Project is due. We would like to view as many of the demos as possible this week.

Milestone reports are largely for your benefit, to ensure you’re pacing the work and to check that you’re on the right track. Milestones will be reviewed but not graded. If you do not hand in a milestone, and we later find that you’re making some serious design errors, it will be *your* problem for not having given us the chance the flag the error early enough for you to fix it.