

University of Arizona
CSc 560: Database Systems Implementation
Spring 2009

Instructor: Bongki Moon (bkmoon@cs.arizona.edu)

Office: Gould-Simpson 746, 621-4326 (voice), 621-4246 (fax)

Office hours: Mon 9:00-10:00 and Wed 11:00-12:00, or by appointment.

Grader: TBA

Class schedule: Mon/Wed 12:00-1:15, Gould-Simpson 942

Final exam: May 15, 2009 (Fri), 11:00-1:00, Gould-Simpson 942

Overview and Topics: CSc560 is a graduate-level course in database systems that will emphasize the DBMS architecture and implementation issues as well as recent research problems. The topics to be covered include

- Storage structures (disk, records, pages and files) and buffer management strategies,
- Access methods: B⁺-tree, multidimensional indexes (k-d trees, Grid files, R-trees),
- Query evaluation and optimization,
- Transaction management: concurrency control and recovery,
- Database security and authorization,
- Parallel database systems, decision support and OLAP,
- Advanced search: *k*-NN similarity search, string databases,
- XML: labeling schemes, path joins, selective dissemination (if time permits).

Prerequisites: CSc460 (Database Design), or the instructor's permission.

Recommended texts:

- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, third edition, McGraw-Hill, ISBN 0-07-246563-8, 2003.
- Bongki Moon, *Lecture Notes on Database Systems Implementation*, available on-line.
- Most of the research articles in the reading list will be available from the ACM/IEEE Digital Libraries, which are accessible from all the computing labs on campus.

Term project: Students are required to complete a term project in a small group (two or three people) throughout the semester. The goal of the project is to build a simplified but working relational database system called MINIREL. The implementation of the MINIREL will be composed of four layers. Among the four layers, the *paged file (PF)* layer will be provided; the other layers (*heap file (HF)*, *access method (AM)*, and *front-end (FE)*) will be implemented by each project team. For each part of the MINIREL project, students are required to submit an archive of source files that will produce a function library and/or an executable program, and a short report that will explain the implementation approach such as overall techniques, important data structures, tricky design decisions, and so on.

Class URL and Newsgroup:

<http://www.cs.arizona.edu/classes/cs560/spring09> and

`news:cs.course560`. You are required to read messages posted to the class newsgroup on a regular basis. You can also send messages to the newsgroup to discuss any issues that arise in class.

Grading: This course will be offered for three credits. The grading policy will be midterm exam (20%), final exam (30%), and project (50%). Except for special circumstances, all members in a group will receive an identical grade for each part of the project.

Posting grades: In an effort to comply with the new laws prohibiting the use of any part of student's SSN to post grades, the Department has devised a scheme to generate a unique numeric identifier for each student. The six digit numbers (called CSID) will be used to post the grades in the class web page, the class listserv and my office door.

Class accounts: You will be given accounts on `lectura(lectura.cs.arizona.edu)`, Linux and Windows workstations, plus an access card to Gould-Simpson 228 Lab. The `lectura`, Linux and Windows accounts are for instructional purposes only. To obtain your accounts, you will need to go to the 9th floor of Gould-Simpson during the first week of classes. There should be signs on the 9th floor directing you to the location of the electronic account application process. You can use any of these platforms to develop your MINIREL project, but it will be tested and evaluated *only* on `lectura`.

Absence Policy: All holidays or special events observed by organized religions will be honored for those students who show affiliation with that particular religion. The memo from Associate Vice President for Campus Life and Dean of Students states: "While we realize that absence from class is sometimes unavoidable due to medical or personal reasons, it is the responsibility of the student to report the reason for his/her absence to the instructor and provide documentation if requested. The instructor will decide whether the student's absence should be excused, and make such arrangements as he/she deems appropriate. Each student is accountable for all work missed because of absence."

Code of Academic Integrity and Conduct: Students are responsible for understanding and complying with the University's Code of Academic Integrity. See <http://studpubs.web.arizona.edu/policies/cacaint.htm>. Violations of the Code will, at minimum, result in loss of credit for a graded item. An egregious first violation or any second violation will minimally result in failure of the entire course.

Policy on Threatening Behavior by Students: The University has this new policy since disruptive behavior has been on the rise. Refer to <http://policy.web.arizona.edu/policy/threaten.shtml>.

Students with Special Needs: Students with special needs who are registered with the S.A.L.T. Center (<http://www.salt.arizona.edu>) or the Disability Resource Center (<http://drc.arizona.edu>) must submit appropriate documentation to the instructor within the first few days of the course if they are requesting special accommodations.

Topics and Tentative Schedule

Date	Topic	Description
Jan/14/2009	Introduction	Course and project description, memory hierarchy and disk space management [1, ch. 9.1,9.3]
—	MLK Holiday	No Class on January 19
Jan/21/2009	Disk modeling	Disk drive modeling [2], Bandwidth vs. latency [3]
Jan/26/2009	Records and Pages	Fixed and variable length records [1, ch. 9.7], page formats [1, ch. 9.6], heap files [1, ch. 9.5, 8.4]
Jan/28/2009	Buffer Management	Basics [1, ch. 9.4], Non-stochastic buffer management [4]
Feb/02/2009	Buffer Management	Improved stochastic algorithms: LRU-K, 2Q, ARC [5, 6, 7]
Feb/04/2009	B-tree Index	Properties of indexes [1, ch. 8.2], B ⁺ -tree, duplicate keys, and bulk loading [1, ch. 10.3-10.8]
Feb/09/2009	Multidimensional index	Overview [1, ch. 28.1-28.3], Grid file [8], k-d tree, quad-tree [9]
Feb/11/2009	Multidimensional index	k-D-B tree [10] and hB-tree [11], R-tree and variants [12, 13]
Feb/16/2009	Multidimensional index	Space-filling curves [14, 15, 16]
Feb/18/2009	Query evaluation	Join algorithms: nested loop, sort-merge, hash join [1, ch. 14.4]
Feb/23/2009	Query optimization	Overview of query optimization, Algebra equivalences, Estimating the cost [1, ch. 15.1-15.4]
Feb/25/2009	Query optimization	System-R query optimizer [17]
Mar/02/2009	Transaction processing	ACID model and serializability [1, ch. 16.1-16.2], Transaction schedules: RC, ACA, ST [1, ch. 16.3]
Mar/04/2009	Concurrency control	Two phase locking, phantom problems, granular locks, SQL isolation levels [1, ch. 16.4,17.1,16.6]
Mar/09/2009	Concurrency control	B-tree concurrency: page-oriented [18, 19], ARIES/KVL [20]
Mar/11/2009	Midterm Exam	In-class Exam
—	Spring Break	No Class on March 16 and 18
Mar/23/2009	Concurrency control	B-tree concurrency: tuple-oriented ARIES/KVL [20]
Mar/25/2009	Concurrency control	Lock management [1, ch. 17.2]
Mar/30/2009	Transaction recovery	Logging and WAL, Steal and Force, checkpointing [1, ch. 18.1-18.5]
Apr/01/2009	Transaction recovery	ARIES recovery [1, ch. 18.6]
Apr/06/2009	Parallel databases	Parallel database architectures and overview [21], parallel hash join [1, ch. 22.4.3]
Apr/08/2009	Parallel databases	Multi-disk B-tree [22], Parallel R-tree [23]
Apr/13/2009	Parallel databases	Multidimensional declustering [24], and other declustering techniques for multimedia
Apr/15/2009	Decision support	Star schema, bitmap index, OLAP [1, ch. 25.2,25.6]
Apr/20/2009	Decision support	Mining association rules [1, ch. 26.2.1], Apriori algorithm [25], Iceberg queries [1, ch. 26.2.2]
Apr/22/2009	Decision support	Classification and decision trees [1, ch. 26.4]
Apr/27/2009	Text & Web search	Inverted indexes and signatures [1, ch. 27.3], HITS and PageRank [26, 27]
Apr/29/2009	Similarity search	Branch-and-bound <i>k</i> -NN search [28]
May/04/2009	String search	String B-tree [29]
May/06/2009	Conclusion	Discussions and review
May/15/2009	Final Exam	11:00-1:00, Gould-Simpson 942

Reading List and References

Ordered By Lectures

- [1] Raghu Ramakrishnan and Johannes Gehrke. *Database Management Systems*. McGraw-Hill, Inc., New York, NY, third edition, 2003.
- [2] Chris Rummeler and John Wilkes. An introduction to disk drive modeling. *IEEE Computer*, 27(3):17–28, March 1994.
- [3] David A. Patterson. Latency Lags Bandwidth. *Communications of the ACM*, 47(10):71–75, October 2004.
- [4] Hong-Tai Chou and David J. DeWitt. An evaluation of buffer management strategies for relational database systems. In *Proceedings of the 11th VLDB Conference*, pages 127–141, Stockholm, 1985.
- [5] Elizabeth J. O’Neil, Patrick E. O’Neil, and Gerhard Weikum. The LRU-K page replacement algorithm for database disk buffering. In *Proceedings of the 1993 ACM-SIGMOD Conference*, pages 297–306, Washington, DC, May 1993.
- [6] Theodore Johnson and Dennis Shasha. 2Q: A Low Overhead High Performance Buffer Management Replacement Algorithm. In *Proceedings of the 20th VLDB Conference*, pages 439–450, Santiago, Chile, September 1994.
- [7] Nimrod Megiddo and Dharmendra S. Modha. ARC: A Self-Tuning, Low Overhead Replacement Cache. In *Proceedings of the 2nd USENIX Conference on File and Storage Technologies*, March 2003.
- [8] J. Nievergelt and H. Hinterberger. The Grid File: An adaptive, symmetric multikey file structure. *ACM Transactions on Database Systems*, 9(1):38–71, March 1984.
- [9] R. A. Finkel and J. L. Bentley. Quad-Trees - a data structure for retrieval on composite keys. *Acta Informatica*, 4:1–9, 1974.
- [10] John T. Robinson. The K-D-B-Tree: A search structure for large multidimensional dynamic indexes. In *Proceedings of the 1981 ACM-SIGMOD Conference*, pages 10–18, Ann Arbor, Michigan, April 1981.
- [11] David B. Lomet and Betty Salzberg. The hB-Tree: A multiattribute indexing method with good guaranteed performance. *ACM Transactions on Database Systems*, 15(4):625–658, December 1990.
- [12] Antonin Guttman. R-Trees: A dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM-SIGMOD Conference*, pages 47–57, Boston, MA, June 1984.
- [13] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. The R^* -tree: An efficient and robust access method for points and rectangles. In *Proceedings of the 1990 ACM-SIGMOD Conference*, pages 322–331, Atlantic City, NJ, May 1990.
- [14] J. A. Orenstein and T. H. Merrett. A class of data structures for associative searching. In *the 3rd ACM SIGACT-SIGMOD Symposium on Principles of Database Systems*, pages 181–190, Waterloo, Canada, April 1984.
- [15] Christos Faloutsos. Multi-attribute hashing using Gray codes. In *Proceedings of the 1986 ACM-SIGMOD Conference*, pages 227–238, Washington D.C., May 1986.
- [16] Bongki Moon, H. V. Jagadish, Christos Faloutsos, and Joel H. Saltz. Analysis of the Clustering Properties of the Hilbert Space-Filling Curve. *IEEE Transactions on Knowledge and Data Engineering*, 13(1):124–141, Jan/Feb 2001.
- [17] P. Griffiths Selinger, M. M. Astrahan, D. D. Chamberlin, R. A. Lorie, and T. G. Price. Access path selection in a relational database management system. In *Proceedings of the 1979 ACM-SIGMOD Conference*, pages 23–34, Boston, MA, 1979.
- [18] Rudolf Bayer and Mario Schkolnick. Concurrency of Operations on B-Trees. *Acta Informatica*, 9:1–21, 1977.
- [19] Philip L. Lehman and S. Bing Yao. Efficient Locking for Concurrent Operations on B-Trees. *ACM Transactions on Database Systems*, 6(4):650–670, December 1981.

- [20] C. Mohan. ARIES/KVL: A key-value locking method for concurrency control of multi-action transactions operating on B-tree indexes. In *Proceedings of the 16th VLDB Conference*, pages 392–405, Brisbane, Australia, 1990.
- [21] David J. DeWitt and Jim Gray. Parallel database systems: The future of high performance database systems. *Communications of the ACM*, 35(6):85–98, June 1992.
- [22] Bernhard Seeger and Per-Ake Larson. Multi-disk B-trees. In *Proceedings of the 1991 ACM-SIGMOD Conference*, pages 436–455, Denver, Colorado, May 1991.
- [23] Ibrahim Kamel and Christos Faloutsos. Parallel R-trees. In *Proceedings of the 1992 ACM-SIGMOD Conference*, pages 195–204, San Diego, CA, June 1992.
- [24] Bongki Moon and Joel H. Saltz. Scalability Analysis of Declustering Methods for Multidimensional Range Queries. *IEEE Transactions on Knowledge and Data Engineering*, 10(2):310–327, March/April 1998.
- [25] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th VLDB Conference*, pages 487–499, Santiago, Chile, September 1994.
- [26] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
- [27] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30:107–117, April 1998.
- [28] Nick Roussopoulos, Stephen Kelley, and Frederic Vincent. Nearest neighbor queries. In *Proceedings of the 1995 ACM-SIGMOD Conference*, pages 71–79, San Jose, CA, May 1995.
- [29] Paolo Ferragina and Roberto Grossi. The string B-Tree: A new data structure for string search in external memory and its applications. *Journal of the ACM*, 46(2):236–280, March 1999.