

# **Transactional Memory: Concurrent Operations in Multiprocessor Systems**

MS Computer Engineering Scholarly Paper Presentation

By

Chris Mason

Advisor: Dr. David Hwang

Co-Advisor: Dr. Kris Gaj

November 25, 2008, 12:00 PM

ST II, Room 230A

## **Abstract**

As the rise of Chip MultiProcessors (CMPs) increases, application developers will face difficulties in trying to develop software that takes advantage of the parallelism available with multiple cores. Traditionally, this would require the developer to implement sophisticated memory blocking in their program through the use of locks, monitors, and semaphores in order to prevent deadlocking and race conditions. This increases the complexity and lowers the maintainability of software. Transactional memory (TM) offers a new alternative to this problem. Transactional memory systems treat finite blocks of code as a single operation, which, at the end of execution, either commits its modified data to shared memory or rolls back all changes if a conflict is detected. For the software developer, this dramatically eases the ability to develop and run an application across multiple cores because they simply have to define where the critical sections are in the code, leaving the TM system to handle potential conflicts at runtime.