MIT Department of Aeronautics and Astronautics
Reflective Memo for Spring 2006

Subject: 16.35, Real-time Systems and Software
Semester: Spring 2006
Instructors: Nicholas Roy
Grading assistance provided by Alvar Saenz-Otero

Learning Objectives

1. **What are the learning objectives (expressed as measurable outcomes) for this subject?**

   Students graduating from 16.35 will be able to:

   (a) **Demonstrate** an understanding of software development practices such as code reviews, code testing and version control

   (b) **Demonstrate** an understanding of the concepts underlying the design, development and sustainment of real-time systems, such as schedulability and execution time analysis, and apply those concepts to implementing such a system.

   (c) **Evaluate** different software architectures

   (d) **Design** and **implement** a concurrent, real-time software system using inter-process communication

   Each of the following outcomes will correspond to an assignment and the final project:

   (a) The student will have **demonstrated** the use of software development practices of version control, unit tests and program analysis.

   (b) The student will have **implemented** control system using concurrent programming.

   (c) The student will have **implemented** a real-time system using concurrent programming and real-time programming concepts.

   (d) The student will have **implemented** a distributed, real-time system system using inter-process communication.

   (e) The student will have demonstrated the ability to **choose** and **justify** an architecture and scheduling algorithm for a control system.

   (f) The student will have **implemented** a distributed, real-time control system using collaborative software development.

2. **To what extent were you able to integrate the CDIO skills specified for this subject in the Curriculum Plan of 2002 (please fill in attached table)?**

   See attachment.

Teaching & Assessment Methods

3. **What teaching and assessment methods did you use and what evidence indicates these methods were successful or not?**

   16.35 uses three main forms of assessment:

   (a) Programming assignments due every 2 weeks. These assignments were the primary form of assessment for the students' ability to implement ideas and concepts.

   (b) A final project due in the last month of classes. The project was to be implemented by teams of 2, and involved two deliverables: integrating the four programming assignments into a networked, real-time flight simulator choosing one additional functionality, such as hardware integration, or synchronization between multiple simulators for formation flight. The final project allowed us to assess the students' ability to integrate the concepts, as well as carry out the principles of design and implementation of software systems.

   The final project appeared to be very successful, as indicated by the course evaluations. The final project was deemed very effective by 5 out of 10 students, with an average score of 2.4/3.

   (c) A final exam. The exam allowed us to assess the students understanding of formal principles such as scheduling, concurrency, synchronization, inter-process communication, etc.

   In a trend consistent with last year, the course evaluations reported by 6 of the 10 respondents indicated they did not believe the course had a final exam. The 3 of the 4 students that responded otherwise indicated that the exam was ”Generally effective” with an average score of 1.75/3. Possibly one concern of the students was that the final exam was the first formal examination in the course.

Student Learning

4. **How well did the students perform on each subject learning objective? (Where possible, make reference to specific data to support your conclusion.)**

   The performance of the students on the learning objectives was generally good. One students underperformed, receiving Cs. Retention was generally high, as only two students of 16 withdrew after the first week.

   In general, performance on the assignments was strong, except for the fourth assignment which was far and away the most difficult assignment. Performance on the final project was strong (88%), consistent with previous years, reflecting the students enthusiasm for developing a complete system. Additionally, the ”Extended deliverable” component that required the students to identify and develop some additional capability for the flight simulator led to high grades (88% average).

Continuous Improvement

5. **What actions did you take this semester to improve the subject as a result of previous reflections or input from students or colleagues?**

   Following on the reflective memo from 2005, I moved the course to only support Java as the programming language. This transition was much easier than I had expected, and had no complaints whatsoever about the choice of language or the lack of support for Ada.

   I re-organized the material, moving the real-time and concurrency material to the front of the course, and de-emphasized the networking material. This led to a more natural flow of both the material and the assignments.

   I switched to lecturing almost exclusively on the blackboard. This made a huge difference to the pace of the material. The only downside is that the lectures that involve writing out code are writing-intensive for the students. I would like to reduce the amount of code that they need to take notes on.

   In the previous year, there had been problems with the pacing of the code deliverables and the written deliverables (i.e. Requirements documents). This was rectified by eliminating the Requirements documents for all but the final project, and requiring the students to submit incomplete assignments (what we termed "Pre-deliverables") a week before the final assignment was due. This allowed me to force the students to start work on the assignment early, but at the same time give an assignment that was larger than can be completed in 1 week. The complaints from the previous year were notably absent, and 4 of the 10 students rated the homework as "Very effective", with an average score of 2.44/3. Involving the additional instructor, Alvar Saenz-Otero, in this part of the course was equally effective. Unfortunately, Alvar was only able to participate for the first six weeks of the course. A full-time TA throughout the course should further help with the effectiveness of the "Pre-deliverables".

   Finally, one problem with handling the assignments last year was in how students carried out their electronic submissions. I clarified the instructions last year, providing explicit instructions about number and type of files, and this simplified submission and grading substantially.

6. **What did you learn about your teaching and assessment methods this semester?**

   I was pleased with how effective the blackboard lecturing was, and the students were generally very positive about the lecture style. I added in-class active learning exercises in the later part of the course, and these exercises were generally very effective.

   We used self-reporting to track the number of hours students spent on the coursework during the first 2 months (when the assignments were due most regularly). As the course has a 3-0-9 load, the students should have expected to spend 9 hours outside of class per week. Over the course of the first 2 months, the students were spending on average 8.66 hours per week. This is obviously not uniform, but in no two week period did the number of hours exceed 20 hours, that is, more than one additional hour per week.

The one major concern I have is that the course has a small number of impedance mismatches. The final exam is the only time the students are actually required to know some detailed factual material. While the assignments and final project require the students to be able to use certain concepts, the course evaluations reflected the fact that enumerating facts on an exam is a different task from using it in practice. Secondly, I still feel like the material is somewhat imbalanced. I would like to accelerate the material on concurrency and spend more time on real-time concepts. Additionally, I would like to incorporate more material on memory management – this seems to be a critical topic in embedded systems that was lost in the semester.

Using Stellar as the course website continued to be positive and the problems with electronic submission were fixed by more explicit instructions. The instructions need to be even more carefully specified for class inheritance and class names, however. The one major problem with Stellar was the public bulletin boards. The bulletin boards were a major technical failure, and after contacting Stellar technical support, we abandoned them mid-way through the semester.

One final major technical failure this semester was with the use of the real-time Java virtual machine. The real-time Java VM that was installed this semester had some significant (undocumented) differences from the previous year's version, leading to some unexpected difficulties in assignment 4. This will be rectified in the future by more carefully testing the installed real-time VM.

The students again had difficulties applying concepts from other courses. About half the class did not realize that a vehicle with both translational and rotational velocity could be rotating about a point outside the body of the vehicle. Without knowing this simple fact of dynamics, a flight simulator is impossible to implement correctly. Note, this is in contrast to the previous year, where no one in the class understood how to implement a P-D loop. Students did, however, regularly recognize concepts (and images!) from the Computers and Programming segment of Unified Engineering to assess algorithm performance, so it is clear that this part of Unified Engineering was starting to take hold.

I continue to believe that a good way to solve this problem is to spend some effort integrating different offerings in the Information sector of Course XVI. I have started implementing a plan with Profs. Deyst and Willcox to share material between 16.35 and 16.60. I will present two lectures on computer programming in 16.60 this (fall) semester, and have asked Prof. Willcox to return to 16.35 in Spring 2007 and give a refresher on the basics of control and dynamics as the students begin implementing their flight simulator. By presenting material in a consistent manner between courses, the students should get the benefits of regular reinforcement.

7. **What actions do you recommend to improve this subject in the future?**

   - In order to fix the problem with the mismatch between the overall style of the course and the final exam, I recommend holding an oral exam. This will allow the students to be examined on their understanding of the concepts in a setting that is more appropriate to how they have been using the material.

- I recommend that the department system administrator be involved in ensuring the installed real-time Java VM suits the needs of the class.
- I recommend having a teaching assistant to ensure the effectiveness of the "Pre-deliverables" throughout the course.
- I recommend rebalancing the material to accelerate the material on concurrency, and spend time on the material related to real-time systems.
- I recommend continuing the integration activities between 16.35 and the other Information Sector courses.

Information Sharing

8. To whom have you forwarded this reflective memo?

   Kristina Lundqvist, Brian Williams, Jon How.