Statement on Teaching
Surendar Chandra

I deeply value teaching and strive to convey my enthusiasm for Computer Science in general and experimental systems in particular to all of my students. My teaching philosophy is shaped by two trends: breakneck rate of change in hardware capability along with outsourcing and other globalization artifacts. Unlike topics that are grounded in absolute truths with mathematical proofs, systems topics emphasize simple and practical solutions that depend on the student's understanding of the contemporary system constraints. The preferred operating system scheduling policy depends on whether it is used in a smart phone, a desktop or a server. Students need a solid understanding of fundamental concepts. However, rote memorization of the underlying techniques leaves students unprepared to adapt to an ever changing technology landscape. Globalization further exacerbates this problem. Our students need to be trained to remain in the forefront of technology evolution and remain competitive in a global market.

Next, I briefly summarize an application of my teaching philosophy in a junior level core Operating Systems course. Operating systems manage system resources such as the CPU, memory and storage. Each of these subsystems use variants of the same fundamental techniques (e.g., buffering and caching). In order to make their similarities and differences clear, I organized my course around five different modules: *Process Management*, *Process Synchronization*, *Memory Management*, *Storage Management* as well as a module that focused on understanding the interplay between these various concepts, especially in the context of servers, desktops, laptops and PDAs. In each module, I showed how (for example) the same caching concepts are applied differently to solve problems unique to a particular subsystem. For example, memory caching optimizes for access speed while storage caching needs to balance with the persistence requirements. Student understanding of each module was evaluated using a quiz, homework assignment, homework project and a module exam as well as a final exam. Even though this course required continual work from the students (e.g., five module exams), it prepared them well for tackling various senior level courses such as distributed storage, image processing, data mining and cloud computing.

In a fast-moving and technologically sensitive area, I relish the challenge of maintaining the vitality of curricula using constant updates and drawing on state-of-the-art examples from topical research domains. My prior experience in building systems in academia (e.g. flockfs) and in industry (e.g., displaycast) allows me to incorporate best practices into the curriculum. I wish to strike a balance between describing fundamental concepts while introducing variants that are more appropriate for contemporary hardware. For example, over the past few years first simultaneous multithreading and then multicore architectures have become ubiquitous; making multi-threading and event based programming an indispensable concept. I adapted the operating systems course to slowly increase its focus on threading issues, more than was adapted by the course text book (*Operating System Concepts*, Silberschatz, Galvin and Gagne).

Educational Experimental Systems Lab

It is important for students to have a reasonable familiarity with the hardware and software systems that they will encounter in industry if they are to be employable upon graduation. At Notre Dame, I built, administered and managed the educational experimental systems lab which consisted of six desktops (provided by the department), a dual and quad processor Itanium2 servers (valued at $40K and $80K each and provided by HP Teaching initiative) as well as laptops (provided by a HP mobility grant). Students enjoyed full freedom to install and modify any software on these machines. I was also the site PI for Planetlab, a world-wide research infrastructure. I used these resources extensively for my course projects.

Personal lecture video capture as a course review tool

Education is a lifelong endeavor with introductory courses providing the necessary foundation for more advanced topics. However, students missed some lectures. They also forgot the topics that were covered in earlier lectures. Students desire lecture review tools for use either within the same course or for use in the future. The ability of lecture videos to capture the different modalities of a class interaction make them a good review tool. Multimedia capable devices that can be used to watch lecture videos are also ubiquitous among contemporary students.

Prior lecture capture systems recognized these advantages. However, a survey of existing capture systems showed that they were expensive to deploy and maintain. I leveraged technology trends to record and distribute my own lectures over seven semesters. All my course materials, including the lecture slides and audio/video of the lectures are publicly available at `http://www.chandrabrown.org/surendar/teach/`. Videos of the lectures are also available in YouTube, Google Video, podcasts and in iTunes U. Some alumni who had graduated and joined the workforce reported that they recently watched the lecture videos. They were able to better understand the lectures (e.g., video compression algorithms) in the context of their current work than when they were students. Similar feedback gives a hint towards the value of distributing lecture videos. I was a member of the university committee that was chartered to guide the deployment of Apple iTunes U among the campus community. My overall experiences in personal video capture is reported in a recent journal article [1].

At FXPAL, I am continuing to diversify this effort. I am currently designing and implementing a high fidelity screencast capture, distribution, archival and retrieval system. This system will complement my earlier video capture efforts. The new system will leverage newer hardware developments. It is expected that this system will be released in the public domain for use by interested universities. My experiences with the screencast streaming dynamics are also under submission to the ACM Multimedia Systems conference.

I wish to build on my prior effort at Notre Dame and at FXPAL to build a practical lecture capture and distribution mechanism that will be useful not only to me but by the academic community in general. I intend to seek support and resources from industrial as well as governmental partners.

References

[1] **Experiences in Personal Lecture Video Capture**. *Surendar Chandra*. IEEE Transactions on Learning Technologies, vol. 4, no. 3, pp. 261-274, July-Sept. 2011.