Visualizing the University

The data for this project will naturally be university specific. At UCLA the registrar's site is easily scrapable, and all the information about a class schedule for a given quarter can be extracted: department, class number and name, professor, building, room number, days of the week, times, whether it's a lecture, discussion, or other category, number enrolled, enrollment cap, and number wait listed. While we didn't need to use this data for the visualization because we were also able, for a small fee for the DB programmer's time, to get anonymized by student schedule data, running this scrape as a cron job during registration allows one to build a time series of by class enrollment data. There are two projects that can come from this data collection.

1 Visualization

Using Processing, the overall class schedule, or with a cooperative registrar, the by student schedule data, can be used to make an interesting visualization. A campus map can be transformed from some sort of mapping format into some simple format that allows simple loading into Java. From here lots of room for creativity is left. At UCLA, we had buildings that puffed up as they filled up via an algorithm to calculate ellipses for each of the building corners and turned from transluscent yellow to opaque red. The visualization cycled through an entire week, with students represented as dots moving from class to class. One got a sense of how the differences between north and south campus (the sciences are mostly south campus).

Aside from the fun of turning a large data set into an interesting visualization, this provided a good lesson in object oriented programming, as objects were needed for students, buildings, and paths from building to building, and they all needed to be able to communicate. It was also useful as an intro to Processing, giving students a look at data visualization more powerful than

what's available in R or other statistical packages. There are also possibilities for interesting analysis, looking at things like how stratified students are by how advanced they are, clustering by major or groups of majors, times of classes available in different departments, times preferred by different types of students, and the like.

2   Enrollment as Time Series

This project, and the visualization project if data directly from the registrar is not available, first requires some scraping. So an intro to tools like mechanize and BeautifulSoup (we used Python), as well as cron. Flat text files are fine, but this could also be used as an intro to working with SQL as well. Once the scraper is done, then of course you have to let it run at reasonably short intervals over the enrollment time frame. The variables of interest are of course the number of students enrolled in each class at each time point, and the enrollment cap.

Now we have a time series. For us, the most interesting question to be asked here was the relation to various professor ratings (overall, difficulty, etc.) available on the student websites. This requires more scraping. There are also interesting questions about class level vs. enrollment speed and the like, and there's an interesting economic type question about whether there's an under or over supply of classes in some departments. With the visualization tools from the project above, this could also be turned into a visualization, using the same sort of building puffery or some other animation.