CSCE 311

Project 4

Powell Fendley

This project was to build a few java classes that would run in the OSP2 simulator to emulate a disk scheduling process. First, I built a Device platform that would have its own constructor, en-queue IORB into its queue, de-queue IORB from its queue, and cancel pending I/O that the IORB Queue had in it. In order to en-queue, I placed a lock on the IORB’s page. This is so I could make sure the I/O would run smoothly. Then I incremented the IORB count of the file associated with the IORB I grabbed. Third I set the IORB’s cylinder to the cylinder that contains the IORB I needed. Then I grab the IORB and en-queue it into my iorbQueue. In order to de-queue, first I checked if the iorbQueue was empty. If it wasn’t, I returned and removed the IORB at its head. In order to cancel any pending I/O, I first checked to make sure the iorbQueue wasn’t empty. Then I iterated through the iorbQueue removing all IORBs associated to the current ThreadCB that is given as this methods argument. I did this by checking the IORBs’ Threads and comparing them to the ThreadCB I was given. If they were associated, I unlocked the IORB page, decremented the IORB file, and then closed the open file. I also had to make a disk interrupt handler. First I got some info about the current interrupt vector, like the iorb and its thread. I also got the page it was on and the frame as well. In addition to all these, I got its open-file handle. Secondly I decremented the IORB count. If its count is zero and its open file has a closePending flag, I then close the file. First I unlock the page, then use an if else statement to make sure the thread is dead and the frame isn’t null. Then a larger if statement makes sure that the reference bit of the frame is set when it needs to be and that the dirty bit is set if the operation performed was a file read, not swap I/O, and the task is dead or dying. If not, a dirty bit is not set. Finally, the frame is unreserved, the threads are notified as to which IORB process to cancel, the device is idled by using the setBusy(false) method, and the next IORB is grabbed using the de-queue IORB method. Then the new IORB is dispatched.