CS411: Individual Writeup - Group 13
Project 4: SLOB Best Fit

Trevor Bramwell

November 19, 2012

Introduction

The main point of this assignment was to gain an understanding of memory management in the Linux kernel and memory algorithms.

This was achieved by having our group write a Best-Fit algorithm for the SLOB memory manager. The Best-Fit algorithm in the SLOB performs very badly compared to the First-Fit algorithm, yet is easy to implement.

Approach

I approached the problem by first browsing the source code of *mm/slob.c* and *mm/slab.c* to get a feeling for how the Linux memory managers were setup. Second, I took an agile approach to solving the problem by first making sure I could boot into a SLOB kernel. Third, I worked on setting up the system calls, so as to first collect the performance measurements from the First-Fit algorithm. Fourth, I went about creating a program to collect this information from the system calls and output in a readable manner. Lastly, I implemented the Best-Fit algorithm in the SLOB, and recorded it's performance.

Design

Designing the Best-Fit algorithm was quite easy. In essence it was finding a minimum. Designing the testing program was even easier as I already had an idea of what information I was going to provide to user space. I assumed I had that information, and worked accordingly, only finally testing it after I had the system calls in place.

Testing

To ensure that the Best-Fit algorithm was working as expected, all that was required was to make sure the Kernel booted. A kernel can't get very far at all without being able to get and store data in memory. Thus, after I had implemented Best-Fit, and reached Init in the startup process, I knew my implementation was correct.

For testing the fragmentation, I began with assumptions. I knew the Best-Fit algorithm produced more fragmentation, and I assumed the First-Fit algorithm fragmentation had to be pretty low, or else no one would use it. Both these assumptions were correct as I ended up getting around 10% fragmentation from First-Fit and approximately 90% from Best-Fit.

Learning

A large time on this project was spent getting just SLOB to boot. After trying several configurations over two days of compiling, I finally got a Kernel with SLOB enabled to boot. The problem was not very clear, though went away once I 'reset' my files by copying in an original version of *mm/slob.c* from the 3.0.4 Kernel.

This project gave me a much greater respect for Kernel configuration. I also learned a great deal about memory management in the Kernel, and feel more comfortable working on things in the Kernel.