

Project 2: I/O Elevators

Chao-Ting Wen Chih-Hsiang Wang .Suwadi

May 6th, 2018

CS544
Operating Systems II
(Spring 2018)

Abstract

Develop the solution based off the current FIFO (noop) implementation in the block directory and write a new file titled sstf-iosched.c. Then Build a LOOK I/O scheduler.

1. The design we plan to use to implement the LOOK algorithms.

Based on the noop-iosched.c, we first understand the basic structure of the elevator algorithm. Then we search for the information about LOOK. We know that its head is moving in the constant direction and it will go through I/O queue to get the requests. The second step, it will change the direction and repeat the process when there is no data to read. We used a queue to check the requests and control the direction of the header.

2. Questions to the assignment.

A. Main point of the assignment

The main point of this assignment is to help us be acquainted with a I/O schedule algorithm, and cultivate the ability to read and design new algorithms. Also, it help us understand the Block IO Layer, in terms of:

- Different data structures involved in implementing stage.
- How scheduling is done on the block layers.
- How the LOOK, C-LOOK algorithms work in particular.
- How character devices differ from block devices.
- The use of abstraction.

B. How to approach the problem

In the first step, we spent much of time to read the materials about I/O schedulers, which helped us know more about conceptual background, essential knowledge to begin step two, analyzing the source code of the existing schedulers. Once we felt somewhat confident with the material, we began the step three, the implementation. This step was hard that we spent a lot of time finding the correct approach.

C. How to ensure the solution was correct

We successfully change the config to SSTF and make -j4. Moreover, we compile and run the scheduler without error. Besides, we print out the result at some important places to make sure each step is working as we expect. As a result, we are quiet sure our solution is correct.

D. What we learn

We learn the basic concept of I/O scheduling. However, it was a difficult process for us. We did not know where to find the instruction about how to start. We learned that how important it is to start a project without the full instruction, which is similar to what we may face in the future position. The important skill is knowing how to ask questions and searching for the crucial information. Besides, the ability to read others' code is also important for computer science students.

E. How should the TA evaluate your work? Provide detailed steps to prove correctness.

When running the VM, it will print out the queue and the position of the head. Based on the comparison of these number, it is easy to check that our solution is correct.

3. Version Control Log

Date	Arthor	Commit	Messages
05/01/2018	ChihHsiang Wang	989dc90d90b19108ba98f919018dddcf5eb65efd	first
05/05/2018	Jerrydepon	989dc90d90b19108ba98f919018dddcf5eb65efd	Create sstf-iosched.c
05/05/2018	ChihHsiang Wang	abe504c424fc19634c81f28f8072fdc84c870a81	add Kconfig and sstf
05/06/2018	ChihHsiang Wang	a1b857784d48d4b2df02932b98935f46af4d2b91	final work

4. Work Log

Date	Project Progress
04/27/2018	Group Meeting I
04/28/2018	Study what is elevator algorithm
04/30/2018	Learn how scheduling is done on the block layer
05/01/2018	Group Meeting II
05/01/2018	Discuss how to use the LOOK and C-LOOK
05/05/2018	Finish the sstf-iosched.c
05/06/2018	Delete and Rebuild the environment
05/06/2018	Upload final work to github
05/06/2018	Writing Completed