Elec/Comp 425

Fall 2010

HOMEWORK 4

Due Date: December 3, 2010

Accepted without penalty till December 15, 5pm

Overview

This homework involves using the DISKSIM Simulator to study the performance characteristics of a simple storage system.

Formal Requirements

Written reports are required. Preferably they will be short reports but should have all the information clearly described. All requested data should be plotted. Brief, precise analysis explaining the results should be provided. PDF files of the report should be emailed to **Hui Wang** (h.wang@rice.edu) by the deadline with a copy to pjv@rice.edu.

Environment Setup

Download and install DISKSIM Version 4.0 on your local computer. You may contact **Hui Wang** (h.wang@rice.edu) if there is any problem (and let pjv@rice.edu know as well). Use the guide below to get started.

Quick Tutorial of DiskSim 4.0

1. Where to I download DiskSim?

Website for DiskSim: http://www.pdl.cmu.edu/DiskSim/index.shtml

Source code download: http://www.pdl.cmu.edu/PDL-FTP/DriveChar/disksim-4.0.tar.gz

Manual download: http://www.pdl.cmu.edu/PDL-FTP/DriveChar/CMU-PDL-08-101.pdf

2. How do I install DiskSim 4.0 ?

(1) First unpack the tar file

(2) Enter the top level of the directory: disksim-4.0/

(3) "make" (Tested for both Linux and Mac, both work without any errors, **32-bit** systems are preferred.)

(4) If (3) succeeds, then you have installed DiskSim on your machine.

(If you have problem with the installation, we may install DiskSim on a server, but make sure do not change the original parameter files. Copy the parameter file and make changes on your own copy.)

3. How do I verify the install is correct?

Go to directory: **disksim-4.0/valid**

Then run: **./runvalid**

If there are no errors, then you have correctly installed DISKSIM

4. How do I run a simple "hello world" example?

(1) DiskSim requires five command line arguments:

disksim <parfile> <outfile> <tracetype> <tracefile> <synthgen>

- **disksim** is the name of the executable file.
- **parfile** is the name of the parameter file (format described in chapter 3).
- **outfile** is the name of the output file (format described in chapter 5).
- **tracetype** identifies the format of the trace input, if any **(You can use *ascii* for this experiment.)**
- **tracefile** identifies the trace file to be used as input. If synthetic workloads are used, set it 0. **(You can set it 0 for this assignment.)**
- **synthgen** determines whether or not the synthetic workload generation portion of the simulator should be enabled (any value other than "0" enables synthetic workload generation). The synthetic generator(s) are configured by values in the parameter file. **(You can set it 1 for this experiment.)**

(2) Go to ***disksim-4.0/valid*** directory.

There are many sample parameter (.parv) files for startup. Two will be enough for our experiments.

(3) You can choose one of parameter file, for example: ***3disks.parv***

To understand the configuration for this test, you can view the ***3disks.parv*** file. There is a block "**topology disksim_iodriver driver0**" which describes the architecture of the disk system. You will notice that this system contains three disks. You can check other configuration parameters in this file, such as the driver, controller and scheduler etc., to get an overall idea of the choices. You ***DON'T*** need to be an expert on this to do this assignment.

(4) Then you can run: **../src/disksim 3disks.parv 3disks.outv ascii 0 1**

The command above will use **3disks.parv** as the input parameters file and **3disks.outv** as the output file, the format is **ascii**, and the workload is **synthetic**.

(5) When it is done, you can view the **3disks.outv** to check the output performance measures such the average response time (check keyword *"Response time average*"), Throughput (check keyword "*Requests per second*"), number of requests handled (check keyword "*Total Requests handled*"), and so on.

Once familiar with setting the configuration parameters, perform the following simulation experiments, and report the results with appropriate plots and explanations.

Task 1:

Set the disk service time to a constant of **10ms**. Vary the **average request arrival rate** and obtain (i) a plot of **throughput vs average arrival rate** and (ii) a plot of **average response time vs average arrival rate**. Repeat this experiment for two different distributions: **uniform** and **exponential**. **(Please refer to the**

instruction file for parameter details).

Task 2:

Choose **HP standard disks** as part of the simulator. Study the performance **(throughput** and **average response time)** as a function of the degree of spatial locality in the workload. That is show how the performance varies as the workload is changed from **100% random to 100% sequential** keeping all other factors the same. Use **Poisson arrivals** in this experiment. **(Please refer to the instruction file for parameter details).**

Task 3 (*Extra Credit*):

Choose the same disk as in Task 2. Study the performance (**throughput** and **average response time**) as a function of the **read/write fraction** in the workload. Obtain results with both **Write Back** and **Write Through policies**. **(Please refer to the instruction file for parameter details)**