CSE141 Graded Homework #2 Solutions (Google's Computers)

February 11, 2002

1 Average CPU Time

To calculate the average CPU time needed for a query, let us assume a set of 100 queries. We know that 90 of these are "simple" queries, and the remaining 10 are "advanced". We also know that the simple queries take 1 second of CPU time, and the advanced queries take 3 seconds of CPU Time. Therefore:

$$\frac{(1.0\frac{\text{sec}}{\text{query}} * 90\frac{\text{query}}{\text{set}}) + (3.0\frac{\text{sec}}{\text{query}} * 10\frac{\text{query}}{\text{set}})}{100\frac{\text{query}}{\text{set}}} = \frac{90\frac{\text{sec}}{\text{set}} + 30\frac{\text{sec}}{\text{set}}}{100\frac{\text{query}}{\text{set}}} = \frac{120\frac{\text{sec}}{\text{set}}}{100\frac{\text{query}}{\text{set}}} = \mathbf{1.2\frac{sec}{query}}$$

2 Latency

The latency of a query is simply the amount of time that elapses between the moment when the computer begins processing a query to the moment processing is complete. In this case, the "work" involved in processing a query consists of fetching data from disk and using the CPU to operate on this data. Note that the processing by the CPU of a query cannot overlap with the disk I/O of the **same** query. The latency can therefore be calculated by simply computing the sum of I/O time and CPU time. Given that data can be fetched at a rate of 2 MB/sec, and that a simple query requires 1 MB of data, the I/O time for a simple query is .5 seconds. Similarly, since advanced queries require 10 MB of data, the I/O time for an advanced query is 5 seconds.

Simple query:

$$\text{Time}_{\text{IO}} + \text{Time}_{\text{CPU}} = .5\,\text{sec} + 1\,\text{sec} = \mathbf{1.5\,sec}$$

Advanced query:

$$\text{Time}_{\text{IO}} + \text{Time}_{\text{CPU}} = 5\,\text{sec} + 3\,\text{sec} = \mathbf{8\,sec}$$

3 A Simple Query Machine

The easiest way to understand this problem is to look at the classic Laundry Room analogy. A given load of laundry must first be washed in the washing machine, and then dried in the dryer. Let us assume that a wash cycle takes 30 minutes, and a dry cycle takes 60 minutes. If one were to use the primitive strategy of washing one load, drying it, and then washing load 2 after load 1 was completely dry and continuing in this manner, it is fairly straightforward to see that one could achieve a throughput of one load of laundry every 90 minutes. However, if one were to begin load 2 in the washer the moment load 1 was transferred from the washer to the dryer, one could increase throughput to one load of laundry every 60 minutes. The key observation is that any given load could be completely washed within the time it takes to dry the previous load.

Similarly, in processing queries, one can imagine that fetching the data from disk is like the "washing" operation, and using the CPU to finish processing the query is analagous to the "drying" operation. (Incidentally, we have chosen to execute I/O first, followed by CPU. However this decision is arbitrary in that one gets the same results regardless of which operation is done first, and which is done last. More realistically, for any given query a machine would probably find itself switching back and forth between I/O and CPU multiple times) A machine that is dedicated to processing simple queries will have to use .5 seconds for I/O time and 1 second of CPU time. Since use of the CPU and disk I/O can overlap, the I/O operations of any given query can always be completed entirely within the CPU time of previous queries. For instance, consider an arbitrary queue of simple queries. One can imagine a timeline like the following:

0 sec	Begin I/O for query 1
.5 sec	End I/O for query 1, Begin CPU for query 1, Begin I/O for query 2
1 sec	End I/O for query 2, Begin I/O for query 3
1.5 sec	**End CPU for query 1**, Begin CPU for query 2
	End I/O for query 3, Begin I/O for query 4
2.0 sec	End I/O for query 4, Begin I/O for query 5
2.5 sec	**End CPU for query 2**, Begin CPU for query 3
	End I/O for query 5, Begin I/O for query 6
3.0 sec	End I/O for query 6, Begin I/O for query 7
3.5 sec	**End CPU for query 3**, Begin CPU for query 4
	End I/O for query 7, Begin I/O for query 8
.	
.	
.	

After the initial startup time, it is clear that queries will complete at a rate of 1 query/second. Therefore the throughput of the simple query machine is **1 query/second**.

4 An Advanced Query Machine

This problem is identical to the previous one, except we now have to consider 5 seconds of I/O time, and 3 seconds of CPU time. For the purposes of our Laundry Room analogy, this is a situation where the washing machine actually takes longer than the dryer:

0 sec	Begin I/O for query 1
5 sec	End I/O for query 1, Begin CPU for query 1, Begin I/O for query 2
8 sec	**End CPU for query 1**
10 sec	End I/O for query 2, Begin CPU for query 2, Begin I/O for query 3
13 sec	**End CPU for query 2**
15 sec	End I/O for query 3, Begin CPU for query 3, Begin I/O for query 4
18 sec	**End CPU for query 3**
20 sec	End I/O for query 4, Begin CPU for query 4, Begin I/O for query 5
.	
.	
.	

After the initial startup time, it is clear that queries will complete at a rate of 1 query every 5 seconds. Therefore the throughput of the simple query machine is **1/5 query/second**.

5 Simple and Advanced Machine Clusters

We need to be able to handle 10 million queries per hour. Because we know that 90% of these queries are simple, and 10% are advanced, it follows that we need to be able to handle 9 million simple queries per hour, and 1 million advanced queries per hour. From the previous two problems, we know that a machine dedicated to simple queries has a throughput of 1 query/second, and that a machine dedicated to advanced queries has a throughput of 1/5 query/second. Therefore:

- Simple query machines:

$$\frac{1\text{ query}}{1\text{ sec}} \text{ x } \frac{60\text{ sec}}{1\text{ min}} \text{ x } \frac{60\text{ min}}{1\text{ hour}} = \mathbf{3600\ query/hour}$$

  If one simple query machine can perform 3600 queries per hour, then in order to perform 9 million:

$$\frac{9,000,000\frac{\text{query}}{\text{sec}}}{1} \text{ x } \frac{1\text{ machine}}{3600\frac{\text{query}}{\text{sec}}} = \mathbf{2500\ machines}$$

- Advanced query machines:

$$\frac{1\text{ query}}{5\text{ sec}} \text{ x } \frac{60\text{ sec}}{1\text{ min}} \text{ x } \frac{60\text{ min}}{1\text{ hour}} = \mathbf{720\ query/hour}$$

  If one advanced query machine can perform 720 queries per hour, then in order to perform 1 million:

$$\frac{1,000,000\frac{\text{query}}{\text{sec}}}{1} \text{ x } \frac{1\text{ machine}}{720\frac{\text{query}}{\text{sec}}} = \mathbf{1389\ machines}$$

6 Unified Query Processing Cluster

To answer this question, let us build another timeline. It was stated in the FAQ section of the homework that for the purposes of this problem, you can assume that queries always come in as nine simple queries followed by one advanced:

0 sec	Begin I/O for query 1
.5 sec	End I/O for query 1, Begin CPU for query 1, Begin I/O for query 2
1 sec	End I/O for query 2, Begin I/O for query 3
1.5 sec	**End CPU for query 1**, Begin CPU for query 2
	End I/O for query 3, Begin I/O for query 4
2.0 sec	End I/O for query 4, Begin I/O for query 5
2.5 sec	**End CPU for query 2**, Begin CPU for query 3
	End I/O for query 5, Begin I/O for query 6
3.0 sec	End I/O for query 6, Begin I/O for query 7
3.5 sec	**End CPU for query 3**, Begin CPU for query 4
	End I/O for query 7, Begin I/O for query 8
4.0 sec	End I/O for query 8, Begin I/O for query 9
4.5 sec	**End CPU for query 4**, Begin CPU for query 5
	End I/O for query 9, Begin I/O for query 10
.	
.	

Analysis of this timeline reveals that all computers in the cluster will process 10 queries (9 simple and 1 advanced) in 12 seconds. We can ignore the initial .5 second I/O time since we assume that this disk fetch wil always be completed within the CPU time of some preceding query. Furthermore, if one looks at the timeline closely, it is clear that CPU time is the limiting factor for throughput. In other words, any necessary I/O for a set of 10 queries can easily be completed before the CPU time for those 10 queries completes. If 9 simple queries takes 9 seconds of CPU time, and one advanced query takes 3 seconds, the total time needed to process 10 queries is 12 seconds. Given this information, we can calculate how many of these machines would be necessary to process 10 million queries per hour. We need not worry about proportions of queries, since we have already taken that into consideration in our calculation of throughput.

Mixed query machines:

$$\frac{10 \text{ query}}{12 \text{ sec}} \text{ x } \frac{60 \text{ sec}}{1 \text{ min}} \text{ x } \frac{60 \text{ min}}{1 \text{ hour}} = \textbf{3000 query/hour}$$

If one mixed query machine can perform 3000 queries per hour, then in order to perform 10 million:

$$\frac{10,000,000\frac{\text{query}}{\text{sec}}}{1} \text{ x } \frac{1 \text{ machine}}{3000\frac{\text{query}}{\text{sec}}} = \textbf{3333 machines}$$