CSE 221 Homework 3

November 18, 2008

Due: Tuesday, November 25 at 11:00 am

Submit your solutions as a hard copy at the beginning of class or e-mail them as a PDF to ebuchana@cs.ucsd.edu *before class starts* at 11:00 am. Late homeworks will *not* be accepted.

Answer each question completely, and support your answers with material from the papers and your own critical arguments, as appropriate.

1. Fast File System

    a. The FFS authors claim doubling the block size roughly doubled throughput. Of course, this trend wouldn't continue forever. Give two reasons why you'd expect continuing to double the block size to eventually experience declining returns.

    b. Cylinder groups were one of the key contributions of the Fast File System. The authors do not specify how big they should be, however. Describe the tradeoffs between large and small groups, and suggest a method to select an appropriate size for a given disk and/or file system.

    c. One thing FFS was unable to do was avoid synchronous metadata updates. Why is it that LFS can postpone metadata writes until a segment is flushed to disk (asynchronous writes), yet FFS is forced to write metadata updates synchronously?

2. Log-Structured File System

    a. What trend in hard disc performance was LFS designed to harness?

    b. Describe the tradeoff in deciding how many segments to clean at once. That is, what are the pros and cons of cleaning many segments at the same time? Few segments?

    c. Upon experimentation, Rosenblum discovered that a 'hot-and-cold' file access pattern caused the greedy cleaner to clean segments at a higher average utilization than a uniform access pattern. Why did the authors claim this occurred? Describe the 'cost-benefit' cleaning policy the authors devised to address the issue.

    d. Now suppose there was a new workload, 'cold-and-hot', which contained 90% hot files and 10% cold files (where 'cold' and 'hot' are defined as before). How would the performance of 'LFS Cost-Benefit' compare to the original two workloads?

3. Parallelization

    a. What fundamental tradeoffs do application authors consider when choosing between using kernel threads and user threads to parallelize their applications? What are the disadvantages of each?

    b. What approach does Scheduler Activations take to providing parallelization with the advantages of both models? Does the approach have costs of its own?

    c. SEDA looks at providing a parallel computation architecture in the important special case of serving Internet services. What characteristics of Internet services led them to abandon the multithreaded programming model that Scheduler Activations augmented, and develop a new architecture entirely?

    d. Considering the benefits of the new architecture, should SEDA replace multithreaded programming for other types of applications as well? Why or why not?

4. Web Server Structure

    We have read about two different web server implementations. Consider the structures of the Haboob and Flash web servers. What advantages does each architecture have over the other, in terms of efficiency, complexity and portability? If you were going to design your own web server, which of the two architectures would you use, and why?

5. Transactional Logs

    Both System R and LRVM provide transactional data storage using logs. System R's Recovery Manager has many mechanisms that the Recoverable Virtual Memory system didn't include. Select two such mechanisms. Describe what they are for, how they work, and why LRVM doesn't employ them. How would an application developer accomplish the same functionality on LRVM?