CPS 210 Final Exam

Fall 1995

Instructions. This exam consists of three parts. Answer all questions according to the instructions for each part. This is a closed book examination. You have 2.5 hours. Total points: 150.

Part 1: True or False

Indicate whether each of the following statements is true or false. Do not bother to explain your answer. Each question in this part is worth 5 points (50 points total).

1. Distributed shared memory is particularly useful for parallel programs running on a shared memory multiprocessor, because the processors do not fail independently.
2. The NFS protocol guarantees that client file caches are strictly coherent, in order to ensure that clients see a consistent view of shared files across the network.
3. File cache coherency is a concern for distributed file systems only when some file is modified by more than one client.
4. NFS server failures can be made transparent to client programs, except that file operations may appear to execute more slowly.
5. The stateless nature of the NFS service was a design choice to improve its performance and scalability.
6. The log-structured file system (LFS) was designed primarily to improve disk *write* performance by reducing seek overhead for writes.
7. NFS client programs may use different pathnames to refer to the same directory, but only if they are executing on different nodes or symbolic links are used.
8. RPC protocols typically define a common representation for arguments and results, so that RPC calls will work between machines of different types.
9. Unix programs that use NFS files must import an RPC library in order to communicate with the NFS server.
10. I pledge to spend as much time thinking about my multithreaded code as I expect to spend debugging it.

Part 2: Approaches to Reliable Distributed Computing

Write a short statement (no more than 10 sentences) about each of the system facilities listed below (11, 12, and 13). Each statement should answer the following three questions about the facility.

What problem is the facility is intended to solve?
What are the key properties of the approach?
What assumptions are made about the application, i.e., what constraints must the application adhere to in order to benefit from the facility?

Each item in this section is worth 10 points (30 points total).

11. relaxed consistency in a distributed shared memory system
12. ordered atomic multicast (CATOCS)
13. transactions in the CMU Recoverable Virtual Memory package, for a single-threaded program running on a single node

Part 3: Paragraphs

Answer each question with a paragraph. Your answer for each question should be ten sentences or less; diagrams may be useful as well. Each question in this section is worth 10 points (70 points total).

14. Describe how RPC calls differ from local procedure calls, from the perspective of the client application or interface designer.
15. Outline the protection mechanisms in NFS. In particular, what is the role of *file handles* in NFS? Are NFS file handles capabilities?
16. How do RAID filesystems increase read and write bandwidth for large, sequential files? Explain why parity calculations are necessary for RAID systems.
17. What does it mean for an RPC operation to be idempotent? Why are NFS operations designed to be idempotent?
18. It has been argued that the use of RPC makes threads more important. Outline at least two reasons why this might be true.
19. Show how to implement mutexes and condition variables as an RPC service. You may assume that message delivery is reliable. Describe the effect of client and server failures on your service.
20. One important function of file system software is to map each *(file, offset)* pair to a particular physical block on a particular server and/or disk. Outline how this is done in each of the following file systems:

    a) the Unix “fast file system”, as described in class and in Tanenbaum (5 points)

    b) NFS (3 points)

    c) LFS, the log-structured file system (1 point)

    d) Zebra (1 point)