CMU 18–746 **Storage Systems** **1 December 2005**

Fall 2005 **Exam 2**

Name: ______________________________

Instructions

There are three (3) questions on the exam. You may find questions that could have several answers and require an explanation or a justification. As we've said, many answers in storage systems are "It depends!". In these cases, we are more interested in your justification, so make sure you're clear. Good luck!

Problem 1 : Short answer. [48 points]

(a) In addition to logging writes to NVRAM, modern disk arrays use a variety of other integrity mechanisms. Explain why each of the three mechanisms below is used (1 sentence for each explanation);

- ECC on memory and bus

  ECC on memory and bus is used to guard against random bits flipping due to transient failures, solar radiation, etc.

- Scrubbing data blocks on the disk

  Scrubbing data blocks in a pro-active way of detecting errors early and (hopefully) correcting them before the data is used.

- Checksum on each data block

  Checksumming each data block is used to guard against disk defects that may corrupt data silently.

(b) Many disk arrays compute parity across disks to protect against disk failures. What makes this approach more difficult in multi-server distributed file systems seeking to protect against file server failure?

There were several answers that talked about "performance" and how it is most expensive to do this in a distributed fashion. Although performance is a problem, it was not what we were looking for. What makes this approach difficult is the amount of care that is needed to ensure ordering of requests. For example, it is important that writes to data and parity blocks are serialized and occur in order.

(c) Your friendly TA claims to have developed a new backup algorithm that backs up your system during idle time (desktop systems have plenty of idle time, so you realize that the idea is not bad). The algorithm is very simple: each time the computer is idle, the algorithm copies raw blocks from your disk to a secondary backup disk (ala physical backup). What is one serious problem with this seemingly simple algorithm?

File system consistency is a serious problem. The backup may happen while the data is on disk, but the metadata is still in memory. Snapshots are usually used to get a consistent view of the file system before doing a physical backup.

(d) Raja decided to sell his Audi car to make room for a large storage system he wants to build inside his garage. He's hoping to make big money by using it to store high-demand movies and distribute them to his neighbors for a small fee. He is very picky about designing a robust system and wants to get every little detail right. In particular, he's considering whether to buy 100 servers from a single vendor (e.g., Dell), or from 10 different vendors. Give one argument in favor of each option.

Buying machines from a single vendor has several advantages: getting a cheaper deal, one way to configure them, one company to complain to when things break. Buying machines from several vendors could make the system as a whole more robust though, because hopefully failures won't be correlated

(e) Many organizations that store highly-classified data require an elaborate process for deposing of old disks. The process sometimes involves burning the disks or overwriting them hundreds of times. Give one reason why such companies might not choose to simply encrypt the data and throw away the key when they want to get rid of the disks?

There are issues with key management (where do you keep the key) as well as cryptography issues (maybe after 10 years from now someone trying really hard to decrypt my disk will succeed)

(f) Give two reasons why call-backs are not used for client cache consistency in the World Wide Web.

There are several reasons why call-backs are not used in the WWW. One reason is that content is mostly static and changes slowly over time. Another reason is performance. Keeping track of thousands of clients and issuing them call-backs incurs resource overheads (memory and network overheads).

(g) The NFS file system was originally conceived to allow clients to use a file server in a completely transparent manner – that is, users and application programs should not notice that a remote server is involved at all. Briefly identify two ways in which it fails to do so.

Many people brought up the "performance" issue again here. A local disk can also be slow though, the network doesn't necessarily make a system slower. We were looking for semantical differences here. For example, NFS doesn't ensure the file delete semantics that a local file system does (a file is deleted only when the last open has been closed). File locking is another issue where the two may differ.

(h) Imagine a sensor monitoring program that opens a file and periodically updates it with the latest values from the sensors. If the file is stored in AFS, why would someone reading that file every hour never see any changes?

Updates in AFS are propagated when the file is closed. If the sensor monitoring program never closes the file, the updates will not get propagated.

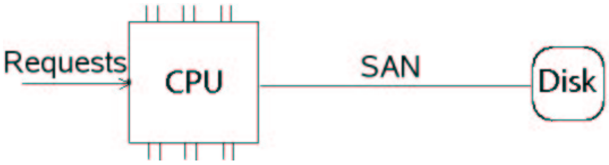

Figure 1: A simple file server.

Problem 2 : System analysis. [28 points]

(a) Company MiniMiniPC builds simple file servers as shown in Figure 1. Assume that each request needs some CPU processing and some I/O from the disk. Measurements have shown that a request in this system needs, on average, 10 nanoseconds of CPU time (CPU processing), 10 microseconds of SAN network time (propagation delay), and 10 milliseconds of disk time (reading and writing data blocks).

- Assuming a closed-loop system with zero think-time and no concurrency (i.e., only one client can send requests), what is the maximum sustainable throughput of this system (in requests/sec)

  Several of you ignored the fact that there is no concurrency here, i.e. N = 1. That single user's request has to use the CPU (10 n), use the network (10 us) and use the disk (10 ms). Hence a request requires Ts = 10ns + 10us + 10ms. The throughput (i.e. the number of requests a second) is just 1/Ts.

- What is the average response time for a request under these assumptions?

  Ts = 10ns + 10us + 10ms

(b) The company hires you as a consultant to improve the system. They are considering one of two options, and they need your help in picking the best one:

- Buy 1GB memory and place it in front of the disk (hits incur zero-cost, misses have to go to disk) OR
- Upgrade the disk with one that is twice as fast

(They cannot do both, since they really want to save money)

Do you have enough information to make a sound decision? If so, make the decision and evaluate the new throughput and response time of the system. If not, make some assumptions about the information you don't have, make the decision and evaluate the new throughput and response time.

Show all your work for full credit.

You do not have enough information to make a decision. Buying more memory will not necessarily improve performance at all. That could happen if the locality of the requests is poor, for example, or if 1GB of memory doesn't capture the working set of the workload well.

If we guess that the workload will have a probability p of hitting in the 1GB cache, then the numbers for throughput and response time change as follows:

*Ts = 10ns + 10us + (1-p)*10ms*

Throughput = 1/Ts

If we upgrade teh disk with one that is twice as fast then: Ts = 10ns + 10us + 5ms

Throughput = 1/Ts

Problem 3 : Designing for disasters. [24 points]

This question will allow you to make some real-life assumptions and compromises on what disaster prevention and recovery technique to use for your data. There is no single good answer here, rather you will need to explain your assumptions.

You are hired as a consultant by three different businesses. The first, is a university that deals with STUDENTS and their data. The second is a BANK that deals with financial information. The third is an ONLINE RETAILER that sells books online. You are to design a storage system for each of these businesses. Armed with the knowledge you learned in this class you feel ready.

(a) Fill in Table 1. Data loss cost refers to the cost of losing hours of work (e.g., if I prepare this exam and the computer crashes before I save, I just lost 3 hours worth of work) and is measured in dollars per hour. Outage cost refers to the cost of losing access to the data and is measured in dollars/hour.

	Data loss penalty $/hr	**Data outage penalty $/hr**
STUDENT	200	0
BANK	50M	10M
ONLINE RETAILER	500K	5M

Table 1: How much is data worth?

You need to make up these costs, however provide a reasonable explanation for your estimates below.

Many reasonable answers here. Make sure to read Keeton's paper (assigned as a reading) to see what she picks and why. In general, the student's data cost has a relationship to their tuition money. However if the data is inaccessible for a while that doesn't matter much. For a bank, losing data is incredibly expensive, however data outages aren't that expensive. For a retailer, data outages are directly related to loss of revenue. Data loss, however, may not be too expensive since it can be recreated again (think of Amazon.com losing the record for the Wizard of Oz book...most likely it can recreate the record quickly from another bookstore).

(b) Next, we look at some techniques to protect against data loss and outages. Fill in Table 2 and Table 3. Hints and assumptions are given in the tables' captions.

Protection technique	**Data loss (seconds)**	**.......Reason for your decision..........**
Synchronous remote mirroring	0	Data is sent instantaneously to the remote center, hence there is no data loss
Async remote mirroring with 1 GB buffer	100	In the worst case the whole buffer can be full
Backup every day on local tape	24 * 60 * 60	A whole day's worth of data can be lost

Table 2: Protecting against data loss

Note: Synchronous remote mirroring refers to writing to the local storage system and a remote replica ”at the same time”, as is done with RAID 1 in disk array systems. In the case of async remote mirroring, data is not immediately sent to a secondary server, but it is first buffered (a write doesn't wait for the secondary to be updated). Assume that 1GB can hold roughly 100 seconds worth of data.

Protection technique	**Recovery time (seconds)**	**.......Reason for your decision..........**
Failover to secondary	0	Flip a switch and the secondary takes over
RAID reconstruction at primary	3 hours	Assuming 100MB/s reconstruction bandwidth

Table 3: Protecting against outages

Assume both primary and secondary systems use a RAID-5 system and the outage is in the form of a disk in the RAID array failing. The companies are extra careful and prohibit access until the RAID is fully repaired. Assume the RAID box is holding approximately 1TB of data. Make a reasonable guess as to how long it takes to reconstruct.

(c) Now, make a decision on what protection and outage mechanisms you will use for each of the businesses. Assume that the original purchase costs of the various options are all equal. Your choices should go in Table 4.

	Data loss mechanism	**Data outage mechanism**
STUDENT	Local backup on tape	RAID reconstruction
BANK	Sync mirroring	Failover
ONLINE RETAILER	Async mirroring	Failover

Table 4: Making the final decision

Problem 4 : Extra credit. [up to 2 bonus points]

(a) What is Greg’s favorite football team?

(b) Which TA is taller?

(c) What is Greg’s standard word for “data loss” and other disastrous circumstances, this term?

(d) What was the most interesting thing you learned this term (in this class)?

This page intentionally left blank in case you need scratch space.