Quiz 2

Due Oct 26 at 11:59pm **Points** 50 **Questions** 5 **Available** Oct 26 at 12:01am - Oct 26 at 11:59pm about 24 hours **Time Limit** 30 Minutes

Instructions

You have to work on the quiz independently. **You cannot discuss with any other people.** Any form of discussion is considered cheating.

This is open book and open note quiz. But **no search on the internet**. You can access the reading list and the paper pdf files listed in the reading list on Canvas.

This quiz was locked Oct 26 at 11:59pm.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	15 minutes	42.5 out of 50

ⓘ Correct answers are hidden.

Score for this quiz: 42.5 out of 50
Submitted Oct 26 at 6:06pm
This attempt took 15 minutes.

Question 1

10 / 10 pts

Paper: The Sprite Network Operating System
Question: Name **2** benefits of using a regular file as a backing store for a process's VM

Your Answer:

1. The migration process has simplified, as backing file is now part of normal file system, every machine can easily get them and migrate the workload to themselves.

2. The remote server can cache the backing file, and as the remote file server tends to have better hardware, it may have have better performance than local drive

1. The backing file can be cached in the memory of a remote note (server node) so it is faster to fetch it in upon a page fault

2. It can make the implementation of process migration much easier

3. It allows the client side to be diskless

Question 2

10 / 10 pts

Paper: Exokernel: An Operating System Architecture for Application-Level Resource Management
Question:

(a) In Exokernel, where does the file system run, inside the kernel or a standard-alone user level process, or inside the application?
(b) Name one con and one pro of this approach over monolithic kernel (Linux)? (2 points)

Your Answer:

a. Inside the application

b. Advantage: Application will have more freedom to implement their unique way to doing file system directly to the hard-drive, e.g. database system.

Disadvantage: Although Linux limits the interface application to interact to the fs, it provide some level of protection over the data integrity, where library user will have to spend lots of efforts customize for their workload, they may also easily write bad codes where destroy all data.

The file system runs inside the application and is just a library.

Pro: faster and extensible (for example, can use a replacement algorithm based on the application's access pattern.

Con: it is hard to protect file access. And it is also hard to share files with other applications.

Question 3

10 / 10 pts

Paper: Implementing Global Memory Management in a Workstation Cluster
Question: why did it decide to use a probability algorithm with weights to approximate?

Your Answer:

Because it is too consuming to determine exactly which node, which piece of memory is the oldest, assume using LRU, while using probability, no blocking will acquire and all decision is happened independently on the node, with less communication overhead as well.

Originally they need to find the oldest global page in the cluster to evict (e.g. when they need to page in one page from the disk). This would require them to message every node to reply with each node's oldest global page, and then compare and choose the oldest one to evict. This would require 2N + 2 messages for every eviction, where N is the number of machines in the cluster. This can cause too many network messages.

To reduce the number of network messages, instead of each eviction, they want to limit the 1-to-N message communication only once during a period of time, epoch. Only at the beginning of epoch, a central node gets the age info from each node, and then base all the info, decide the weight for each node (the probably to be selected to evict an old global page). Then distribute thee weight vectors to all nodes in the cluster. When one node decides to choose a server to evict its oldest global page, it no longer requires 1-to-N communication. It simply uses the weight vector to randomly pick a server (the server with more weight is more likely to be picked), and then message that server to evict its oldest global page.

Question 4

5 / 10 pts

Paper: The Performance of Micro-Kernel Based Systems

Question: This paper tried to show that Micro-kernel can be made efficient against monolithic kernels. So why would people thought that micro-kernel is slow?

Your Answer:

Application can only transfer information via message mechanism instead of memory sharing. However, it is being proved by ukernel that with proper implementation, only less than 5% of the overhead will be added compared with monolithic kernels.

Micro-kernel, for the benefit of extensibility, puts many parts of the OS services such as file systems or device drivers or memory-paging at user level as an OS server, therefore, when an application does a file system call, or suffers from a page fault, the micro-kernel needs to deliver a request via IPC to the OS server (that runs at user level), and once the request is handled by this OS server, it needs to use IPC again to notify the micro-kernel, and then the micro-kernel returns back the application. It needs to switch between user and kernel level twice, slowing down the performance.

Question 5

7.5 / 10 pts

Paper: The Sprite Network Operating System
Question: The paper says that Sprite provides consistency but not necessarily guarantee correct synchronization. What does it mean?

Your Answer:

Consistency only guarantee that when a read happened, it will be always the latest version. However, the order of the execution, e.g. write, may not be original one, which is under the field of synchronization.

Consistency means that all clients (users) will see the same content (the content can be mixed)

Correct synchronization means that the updates from multiple clients/users should not be mixed together. So applications themselves need to use file system locks to ensure correct synchronizations.

Quiz Score: 42.5 out of 50