

CSE 451: Operating Systems

Section 9 Final exam review

Final exam review

- * Goal of this section: key concepts you should understand
- * Not just a summary of lectures
- * Slides may not cover all topics that will be on exam

12/9/10

2

Thread management

- * Queues
- * Synchronization
- * Preemption

12/9/10

3

Memory management

- * Purposes:
 - * Resource partitioning / sharing
 - * Isolation
 - * Usability
- * Paging
- * Segmentation

12/9/10

4

Virtual memory

* What happens on a virtual memory access?

12/9/10

5

Virtual memory

* What happens on a virtual memory access?

* Address translation: who performs it?

* Page table lookup

* TLB

* Page fault?

* Page replacement

* Process/queue management

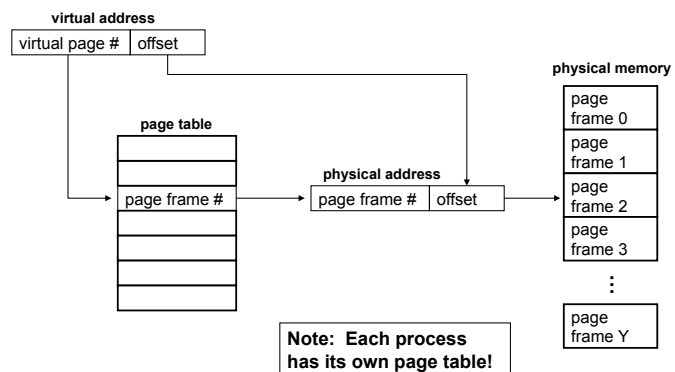
* How does all of this overhead pay off?

* Locality!

12/9/10

6

Virtual memory



12/9/10

© 2010 Gribble, Lazowska, Levy, Zahorjan

7

Page replacement

* Algorithms:

* Belady, FIFO, LRU, LRU clock / NRU, working set...

* Local vs. global

* How/why are any of these better or worse than the others?

* What happens when paging goes wrong?

* Thrashing

12/9/10

8

Advanced virtual memory

- * What problem does a TLB address?
- * What problem do two-level page tables address?
 - * What's the key concept?

12/9/10

9

Advanced virtual memory

- * What problem does a TLB address?
- * What problem do two-level page tables address?
 - * What's the key concept?
 - * Indirection

12/9/10

10

Secondary storage

- * Memory forms a hierarchy
- * Different levels of disk abstraction:
 - * Sectors
 - * Blocks
 - * Files
- * What factor most influences the ways that we interact with disks?

12/9/10

11

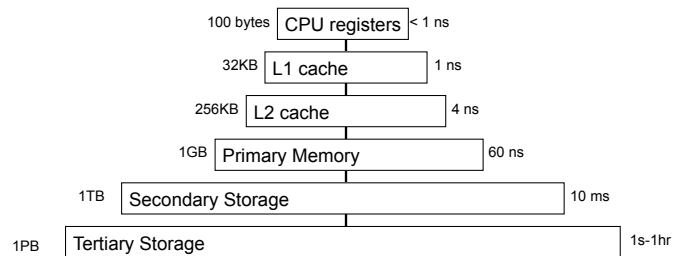
Secondary storage

- * Memory forms a hierarchy
- * Different levels of disk abstraction:
 - * Sectors
 - * Blocks
 - * Files
- * What factor most influences the ways that we interact with disks?
 - * Latency

12/9/10

12

Memory hierarchy



* Each level acts as a cache of lower levels

12/9/10

© 2010 Gribble, Lazowska, Levy, Zahorjan

13

File systems

- * What does a file system give you?
 - * Useful abstraction for secondary storage
 - * Organization of data
 - * Hierarchy of directories and files
 - * Sharing of data

12/9/10

14

File system internals

- * Directories
- * Directory entries
- * Inodes
- * Files:
 - * One inode per file
 - * Multiple directory entries (links) per file

12/9/10

15

Inode-based file system

- * Sequence of steps when I run `echo "some text" > /homes/pjh/file.txt` ?
 - * Open file:
 - * Get inode for / -> get data block for /
 - * Read directory entry for / -> get inode for /homes
 - * Repeat... -> get data block for file.txt, check permissions
 - * Write to file:
 - * Modify data block(s) for file.txt in buffer cache
 - * Close file:
 - * Mark buffer as dirty, release to buffer cache
 - * Kernel flushes dirty blocks back to disk at a later time

12/9/10

16

Other file systems

- * What problem does each of these address?
 - * BSD Unix fast file system (FFS):
 - * Performance: smarter physical disk layout
 - * Journaling file systems (JFS):
 - * Reliability: transactions prevent inconsistencies after crash
 - * Berkeley log-structured file system (LFS):
 - * Performance: even smarter physical disk layout?

12/9/10

17

RAID

- * Striping: read/write from multiple disks simultaneously
 - * Improves performance
 - * Hurts reliability
- * Parity: store redundant information to allow data recovery after disk failures
 - * Improves reliability
 - * Hurts performance

12/9/10

18

Networking

- * Layering
- * Encapsulation

12/9/10

19

RPC

- * Benefits:
 - * Low-level details taken care of for you
 - * Natural interface
- * Implementation issues:
 - * Network failures / retries
 - * Architecture differences
 - * Performance

12/9/10

20

Distributed file systems

- * Why do we want them?
 - * Location independence
 - * Large-scale data sharing
- * Why are they hard?
 - * Consistency
 - * Replication
 - * Performance
- * Understand the target workloads

12/9/10

21

Distributed systems

- * Scalability
 - * Limited by sharing

12/9/10

22

Virtual machine monitors

- * VMM is an additional layer between OS and hardware
 - * Can interpose on instruction execution, memory accesses, I/O requests, and network communication

12/9/10

23

Security

- * Symmetric (secret key) vs. asymmetric (public key) encryption
- * Privacy/confidentiality vs. integrity

12/9/10

24

Cool uses for VMs

* *Accountable Virtual Machines*

* Andreas Haeberlen, University of Pennsylvania;
Paarijaat Aditya, Rodrigo Rodrigues, and Peter
Druschel, Max Planck Institute for Software
Systems

* OSDI 2010

* [http://www.usenix.org/event/osdi10/tech/
full_papers/Haeberlen.pdf](http://www.usenix.org/event/osdi10/tech/full_papers/Haeberlen.pdf)

* [http://www.usenix.org/event/osdi10/tech/slides/
haeberlen.pptx](http://www.usenix.org/event/osdi10/tech/slides/haeberlen.pptx)

12/9/10

25

12/9/10

26