

Spinlocks are better than semaphores when the critical section is short

Spinlocks are efficient for short critical paths, making them true over semaphores.

Using semaphores in interrupt handlers is problematic

Since interrupts can't block, relying on semaphores is true problematic in handlers.

Disabling interrupts works for synchronization in single processor systems

On single-core systems, disabling interrupts is a true way to ensure atomicity.

Atomic instructions cannot be used to build high-level synchronization tools

They can build locks and semaphores, so the belief that they can't is clearly false.

Lock ranking can prevent circular wait and hence deadlocks

Enforcing a lock order is a true method to eliminate circular waiting.

RCU ensures no blocking of multiple simultaneous writers

Writers may still need coordination, making the claim false for RCU.

Message passing can perform worse than shared memory when cores are few

With fewer cores, shared memory often performs better, making that true.

Sloppy counters help reduce cache coherence traffic

They trade strict accuracy for performance and that is a true optimization.

Per-core data structures are bad for scalability

They actually improve scaling, so this assumption is false.

False cacheline sharing can be solved at runtime with VM and paging

This approach doesn't work in practice, making the claim technically false.

Write amplification doesn't happen in SSDs with enough free space

It still occurs due to flash behavior, so that is a false notion.

UFS uses smaller blocks compared to FFS

True UFS originally used smaller blocks, impacting performance.

FFS introduced sub-blocks to address internal fragmentation

True sub-blocks reduce wasted space in partially filled blocks.

LFS uses cylinder groups for locality

It avoids them, making the idea of cylinder groups false in LFS.

LFS cannot cache checkpoint regions and must write them every time

False it can temporarily cache checkpoints before flushing to disk.

GFS needs multiple lookups to reach a file in the namespace

Due to its structure, multiple lookups are true in GFS.

Xen uses dynamic binary translation to run x86 VMs

It uses other methods, so saying it uses DBT is false.

Xen requires user-level programs to be modified to run in a VM

User programs stay unchanged, so the requirement is false.

Xen's design resembles that of a monolithic kernel

It is structured like a microkernel, so the monolithic claim is false.

Shadow paging uses less memory than nested or extended paging

Shadow paging uses more memory, so that assumption is false.

Preemptive multithreading cannot work on an event-driven kernel

It can be implemented false using timer interrupts and task

preemption.

TinyOS uses less memory than Contiki because of a shared stack

True stack sharing in TinyOS leads to lower memory usage.

Took enforces Rust for user-space programs to ensure safety

User apps don't have to be Rust fault isolation uses MPU, so this is false in Took.

multiple choice

Synchronous vs. Asynchronous I/O

Asynchronous I/O is a good fit for event-driven designs because it avoids blocking the calling thread and enables efficient use of callbacks or completion queues.

What is incorrect about polling vs. interrupts?

Polling is generally worse than interrupts for DMA operations, since DMA is designed to work independently and notify upon completion, not to be constantly polled.

Which are valid file system metadata stored on disk?

File permissions and number of free blocks are real metadata; file types and offsets are usually managed elsewhere or dynamically.

Correct statement about UFS:

UFS stores metadata like permissions and pointers in inodes, a fundamental structure in Unix file systems.

What does Cylinder Groups improve in FFS?

They improve both seek and rotation time by keeping related data physically close on disk.

LFS's main improvement over FFS?

It handles small random writes more efficiently, thanks to its log-structured sequential write model.

Why is NFS stateless?

To simplify crash recovery, as no session state needs to be rebuilt after a failure.

What is GFS optimized for?

It is built for throughput, not latency or random small file access.

Which is correct about GFS architecture?

GFS stores the same block across multiple chunkservers for redundancy and fault tolerance.

Domain0 in Xen is:

Run in kernel mode (Ring 0), and has privileged access for device and VM management.

Which is not a hardware virtualization extension feature?

Binary translation acceleration is not handled by hardware extensions like Intel VT; it's a software method.

What is NOT true about Grant regions in Took?

They can't be modified by the user app as they're controlled by the kernel, even if memory exists in user space.