Operating Systems: To Be, or Not To Be?

John Ousterhout
Hollowing Out of the OS

- High OS overheads limit performance
- Functionality is moving out of the operating system
- Are we reaching a tipping point?
- What will operating systems look like in 10 years?
Network I/O: Kernel Bypass

Traditional Approach (TCP)

- Application
- Operating System
- NIC

High Latency

Kernel Bypass

- Application
- Operating System
- NIC

- NIC supports multiple sets of device registers, buffer rings
- Memory-mapped into application
Thread Scheduling

- **Traditional kernel-based thread scheduling is breaking down:**
  - Context switches too expensive
  - Applications don’t know how many cores are available (Can’t match workload concurrency to available cores)
  - Kernel may preempt threads at inconvenient points

- **Fine-grained thread scheduling must move to applications**
  - Kernel allocates *cores* to apps over longer timer intervals
  - Kernel asks application to release cores

- **Henry Qin will discuss in more detail...**
Storage I/O

- Access latencies dropping dramatically:
  - Disk: 10ms
  - Flash: 100μs
  - Future nonvolatile memories: 1μs?

- OS overheads will dominate access times
  - Can’t achieve performance potential of devices

- Applications will need more direct access to devices
  - Analogous to kernel bypass for networks?
  - What about small files?
    (high OS overhead for open: protection checks, lookups)
  - Need a new data model?
Virtual Machines

- Hypervisors taking over traditional OS responsibilities:
  - OS can use paravirtualization: much simpler
What Happens to the OS?

- Does it get dramatically smaller?
  - History suggests “no”

- Does its role change?
  - Control plane versus data plane

- Does it change shape?
  - New architectures to dramatically improve performance
  - IX talk will address this alternative
Questions/Discussion