

# Multikernel Architecture: The Future of Operating Systems

**Cong Wang**

[cwang@multikernel.io](mailto:cwang@multikernel.io)

Founder and CEO, Multikernel Technologies

# Traditional OS Are Failing

- Traditional OS was designed for a simpler era, decades ago
- Whether monolithic kernels like Linux or microkernels, struggle with modern application demands.
- They face **three critical challenges** that prevent them from meeting today's performance requirements.

# Challenge #1: Scalability Bottlenecks

- From single CPU to many-core processors, traditional kernels become performance bottlenecks
- Shared kernel resources create contention
- "One-size-fits-all" approach prevents workload-specific optimization

# Challenge #2: Noisy Neighbor Problem

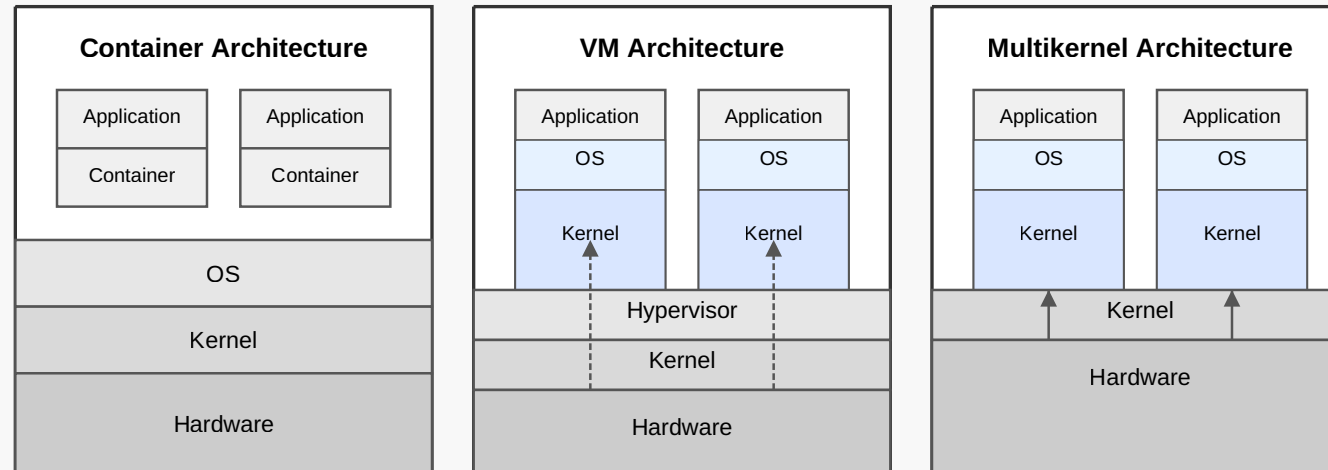
- Applications interfere with each other through shared kernel resources
- One misbehaving application can degrade performance for everyone
- Containers can't solve the fundamental kernel sharing issue

*"In containers, applications still share the same underlying kernel"*

# Challenge #3: Kernel Development Complexity

- Optimizing Linux kernel requires deep expertise
- Tailoring Linux kernel config takes hours
- Configuration kernel involves thousands of interdependent options
- Updates require reboots with significant downtime

# Multikernel Architecture



# Multikernel Architecture Overview

- Treating the system as a distributed environment
- Instead of one monolithic entity, we treat each CPU core or group of cores as running its own dedicated kernel instance.
- This creates excellent isolation while enabling application-specific optimization.

# Performance Isolation

- "It's like moving from a shared apartment to having your own house"
- Complete privacy - no interference from neighbors
- No interference - other's actions don't affect your application
- Customize everything - optimize for your specific needs

# How the Magic Happens

- The host kernel acts as a resource coordinator:
  - Dynamically allocating CPU cores, memory and I/O devices
  - Spawning kernel instances for each application
  - Enabling near bare metal performance

# Near Bare-Metal Performance

- Eliminating shared kernel bottlenecks
- Direct I/O access through modern hardware features
- Applications run at speeds approaching native hardware performance
- No more virtualization overhead, no more kernel resource contention

# Perfect Isolation

- The noisy neighbor problem disappears completely
- Each application runs in its own kernel instance
- Dedicated CPU cores and memory per application
- One application's behavior cannot affect another
- Kernel-enforced isolation without VM overhead

# Application-Specific Optimization

- Database gets a kernel optimized for I/O and memory management
- Real-time application gets a kernel with real-time CPU scheduling and latency guarantees
- Each workload gets perfectly tuned environment with AI

# The Transformative Impact

- **Cloud providers** can offer differentiated infrastructure services with guaranteed performance isolation
- **Database companies** can ship optimized kernels with their software, eliminating deployment complexity
- **Real-time systems** can achieve guaranteed latencies without expensive specialized hardware

# Zero-Downtime Kernel Upgrades

Instead of rebooting entire systems, we can:

- Spawn a new kernel instance with updated code
- Seamlessly migrate workloads from the old to new kernel
- Keep applications running while the underlying kernel is replaced

*"It's continuous deployment for the most critical system software"*

# Elegant Upgrade Process

Three simple steps:

1. Host kernel spawns a replacement kernel with updated code
2. Gradually transfers resources and state from old kernel to new one
3. Retires the old kernel seamlessly

**No single system-wide reboot needed** - effectively achieving zero downtime for kernel maintenance.

# The Future of Computing is Multikernel

Multikernel provides the solution for:

- Cloud providers looking to differentiate your offerings
- Enterprises seeking better performance and isolation
- Developers who want kernel optimization without manual efforts

# Join The Revolution

*The question isn't whether multikernel will transform computing. It's whether you'll be part of leading that transformation.*

# Contact Us

- Visit our website: [www.multikernel.io](http://www.multikernel.io)
- Contact us: [cwang@multikernel.io](mailto:cwang@multikernel.io)
- Subscribe our Youtube channel: <https://www.youtube.com/@multikernel-tech>
- Follow us on Linkedin: <https://www.linkedin.com/company/multikernel/>
- We will open source our solution in the Linux kernel community very soon, collaborations are welcome.

# Thank You!

**Cong Wang**

Founder & CEO, Multikernel Technologies

**[www.multikernel.io](http://www.multikernel.io)**

*Build The Next Generation of Operating Systems*