

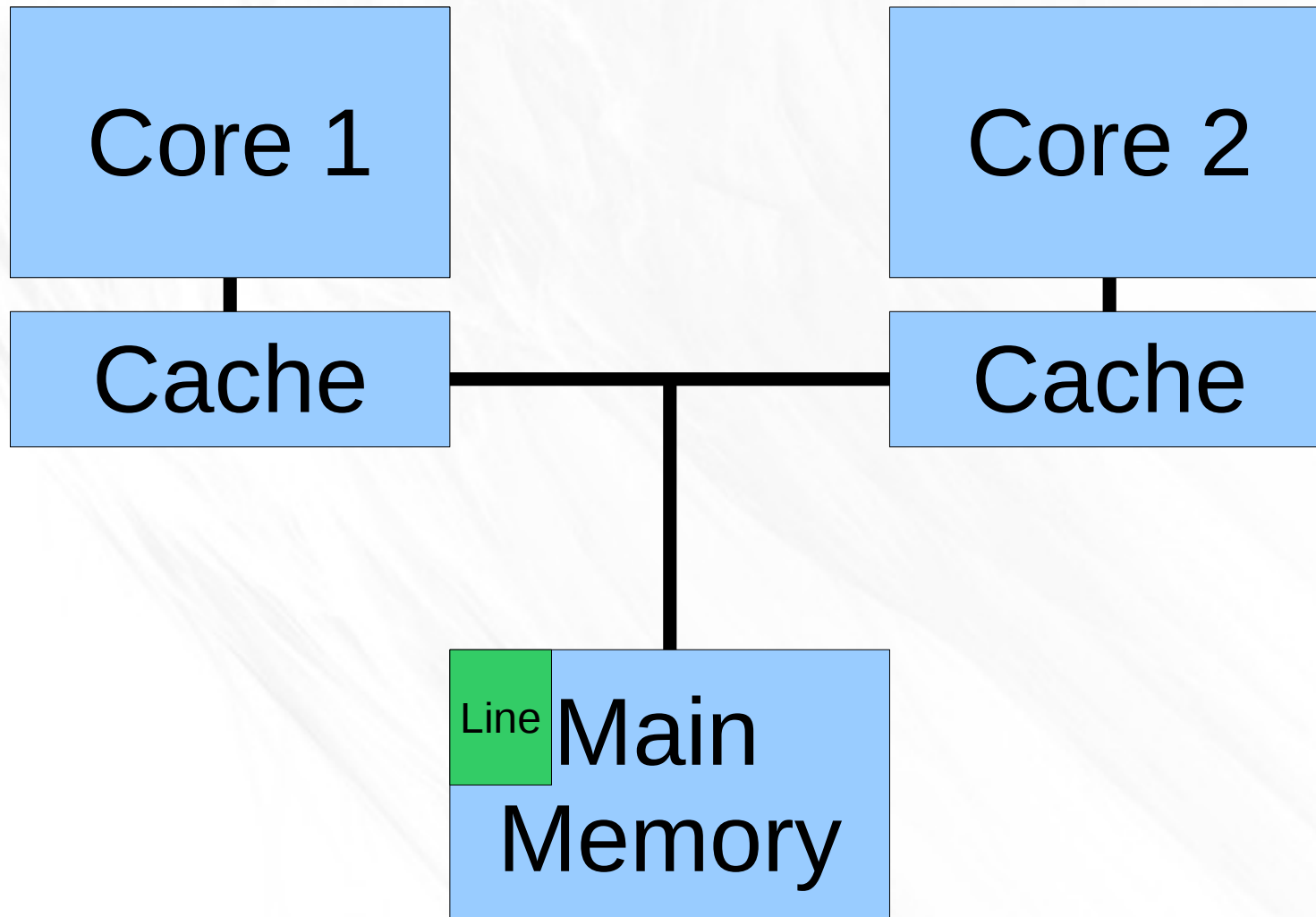
On the Glory of Corey

Christopher Smowton

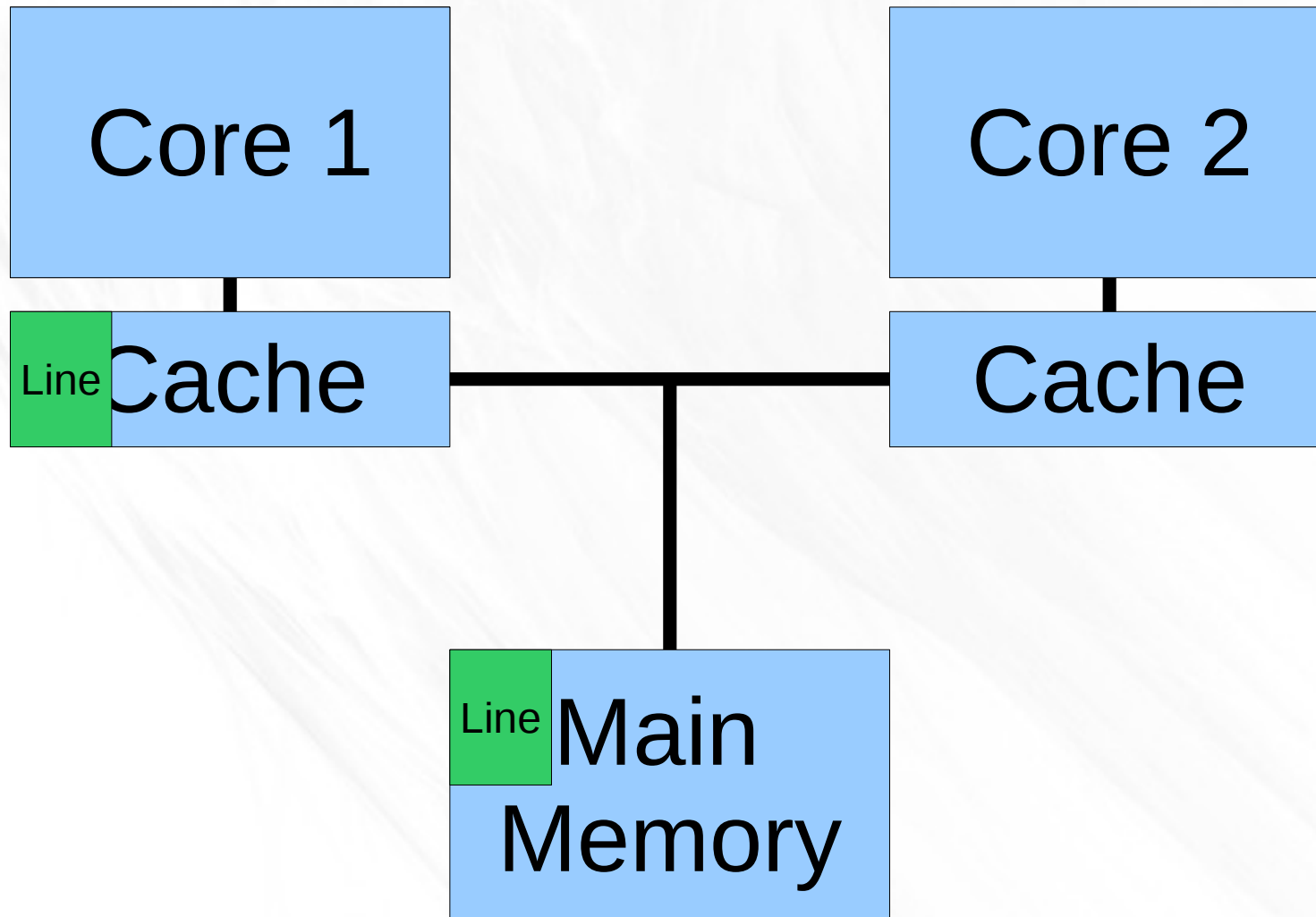
Multicore Trouble, Outlined

- Fast cores
- Interconnect bottleneck
- So, minimise traffic

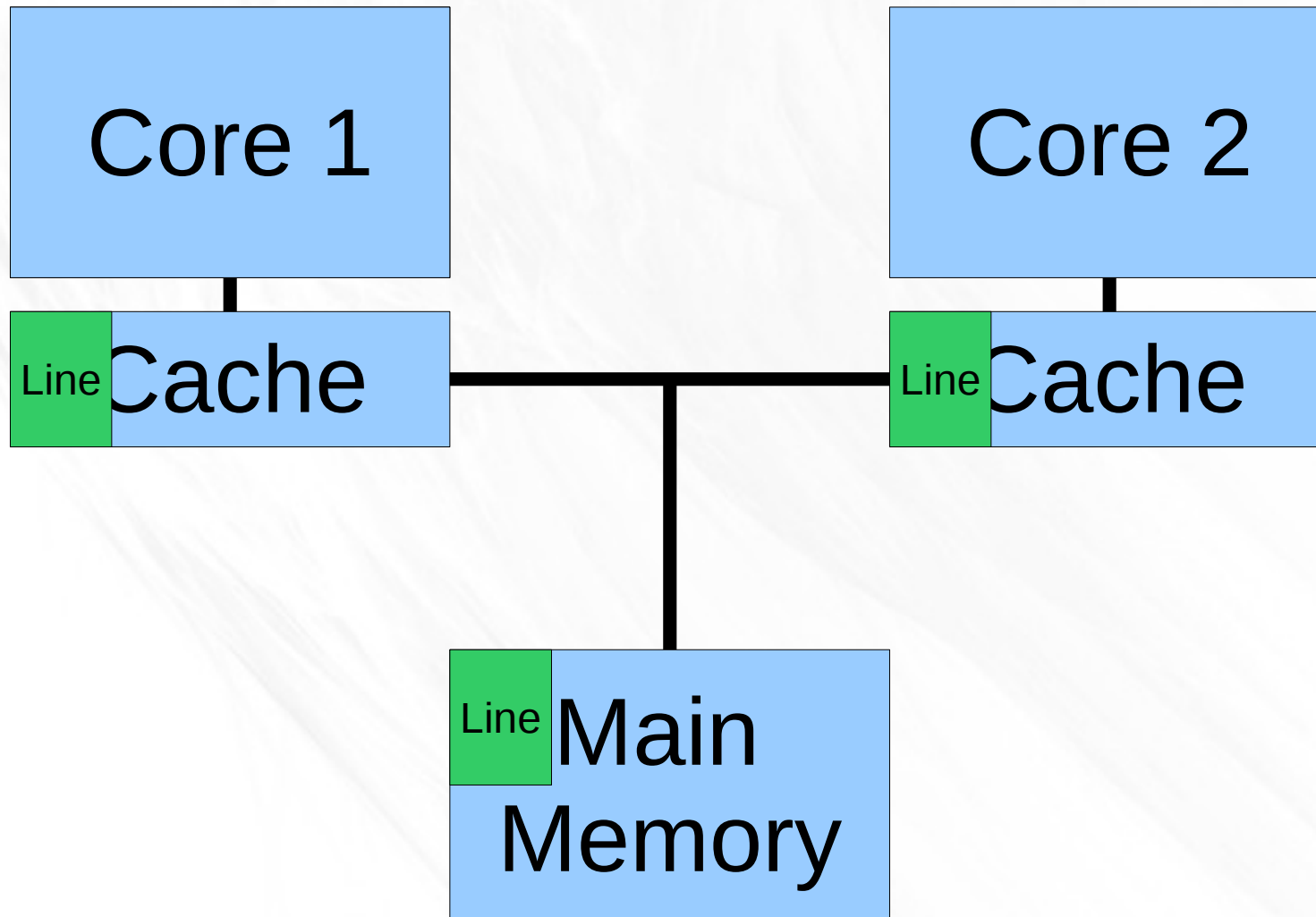
Multicore Trouble, Depicted



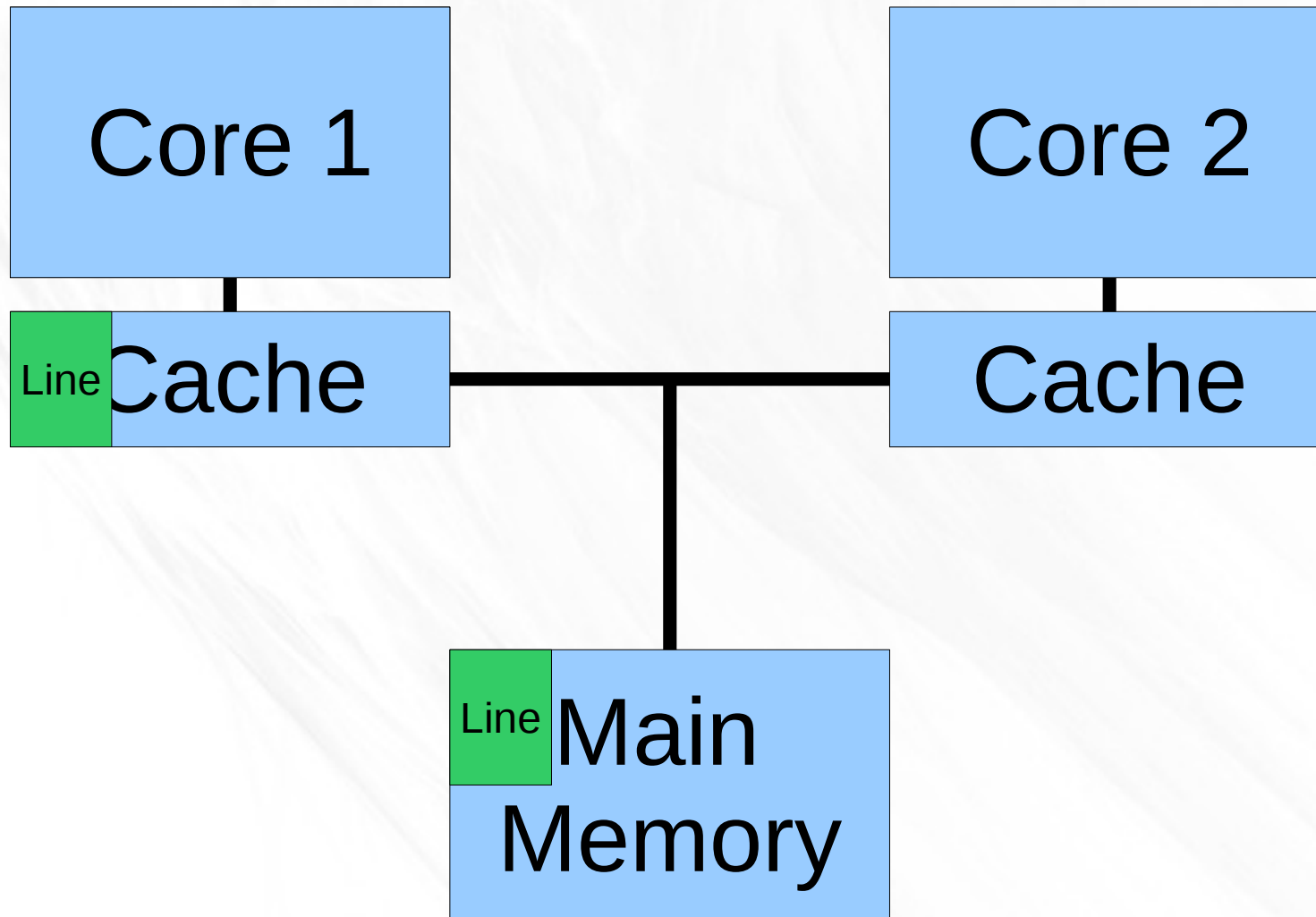
Multicore Trouble, Depicted



Multicore Trouble, Depicted



Multicore Trouble, Depicted



Two Pitfalls

- Exchanging lines between cores when we didn't need to at all
- Exchanging more lines than we needed to

The Peril of False Sharing

- POSIX file descriptor table is process-global
- Guarded by a spinlock (on Linux)
 - And contended spinlocks bounce
- So `open`, `dup`, `close` are all expensive
- Same goes for `mmap`

Optimising for Multicore

- Make most data core-private
- Exchange as little as you can
- Like a distributed system

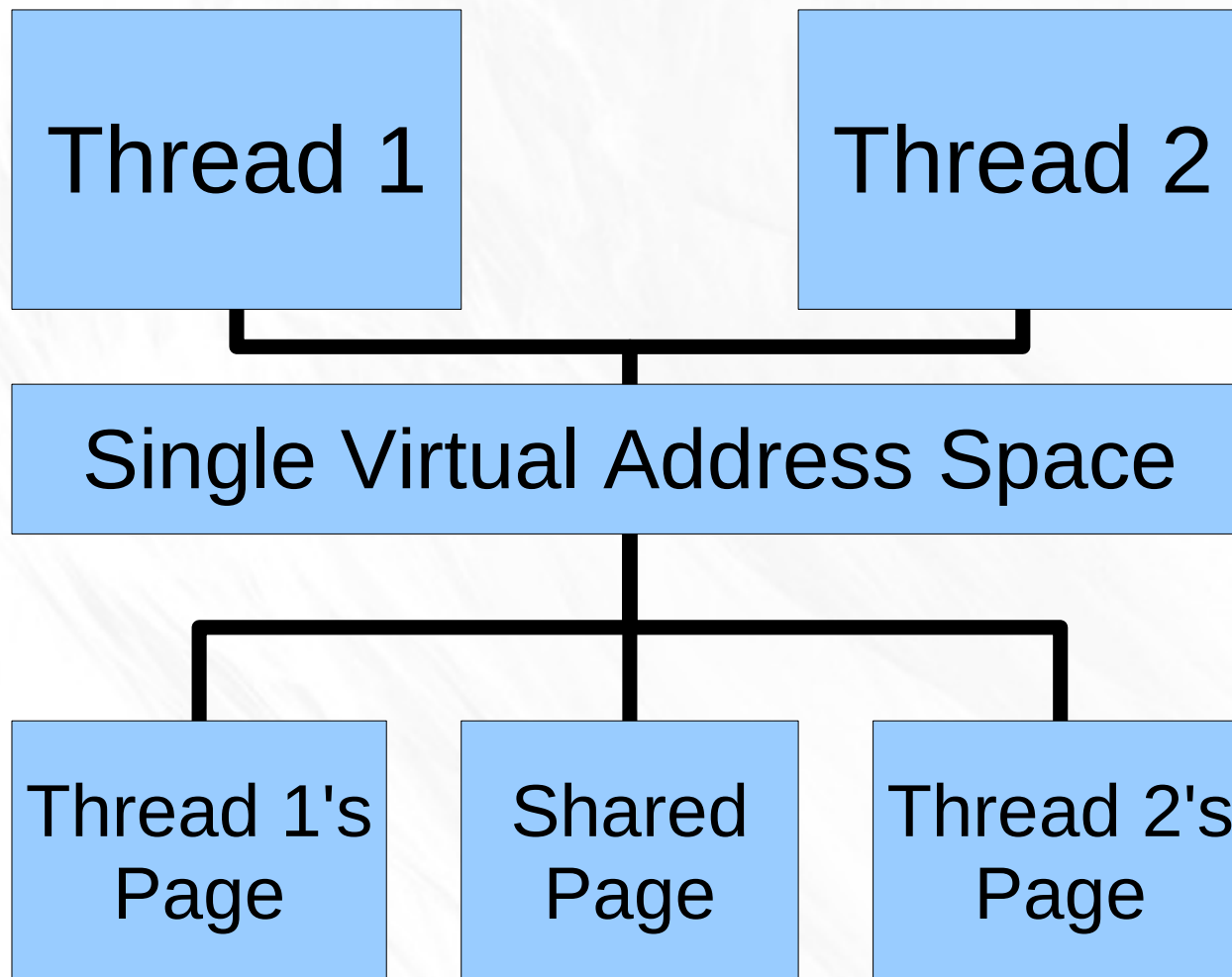
Fear Not, Here Comes Corey

- Address Ranges: private pieces of VAS
- Shares: private pieces of FD table
- Kernel Cores: a kthread per device

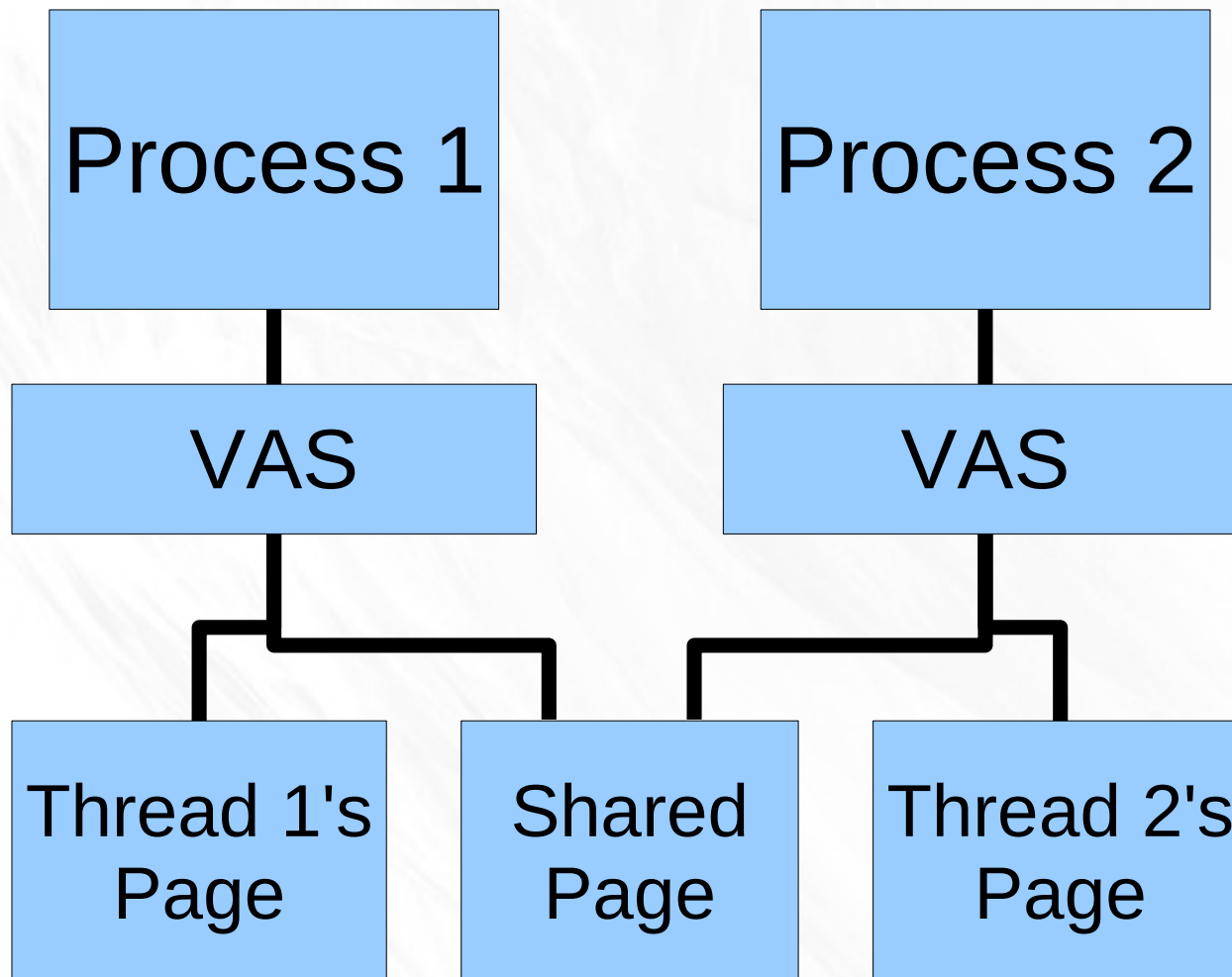
Address Ranges: Like POSIX Shared Memory, but better

- Mark some subset of your memory private
- ...and some of it public
- No locks to modify private mappings
 - And no need for expensive TLB shutdown

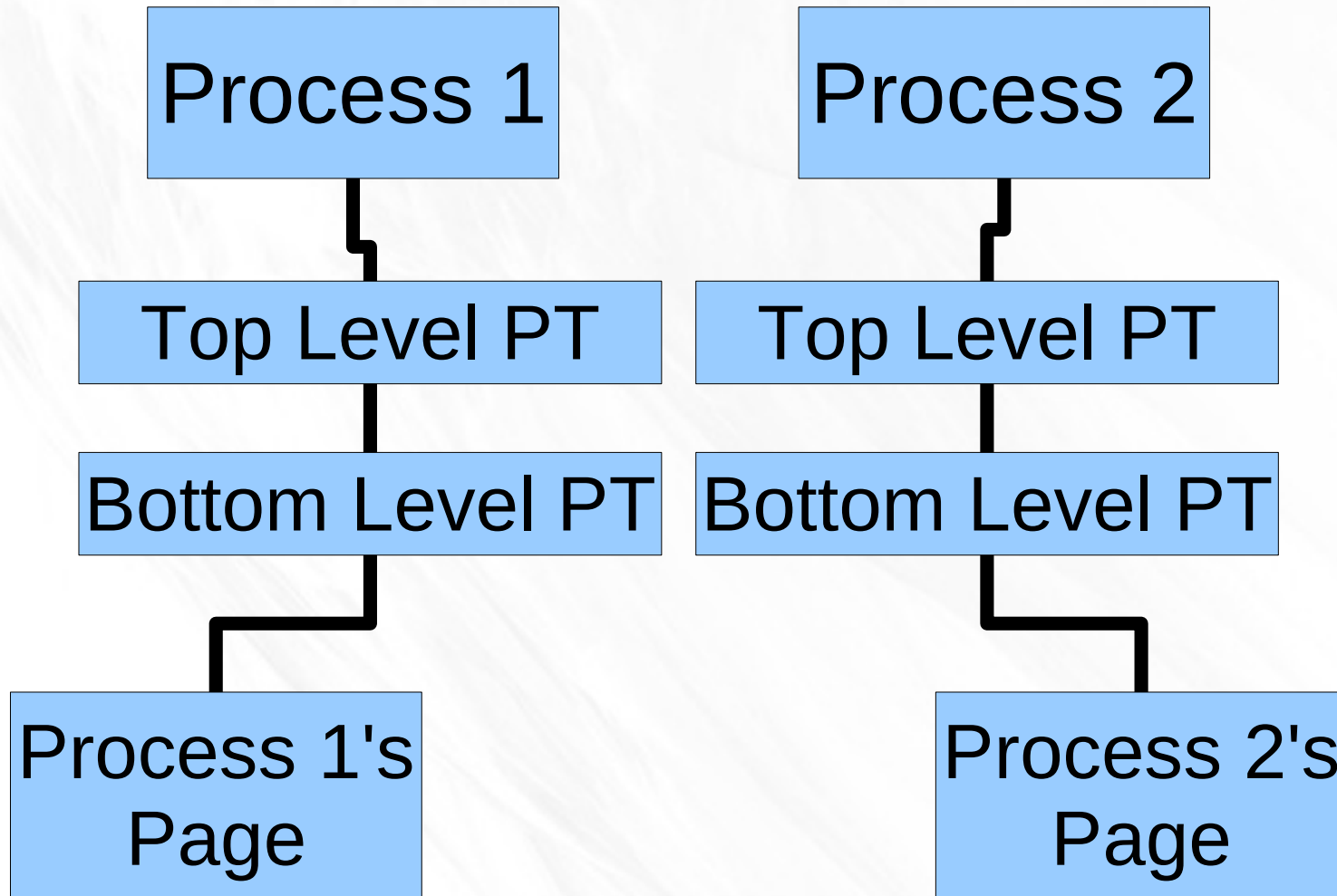
Two threads sharing a page



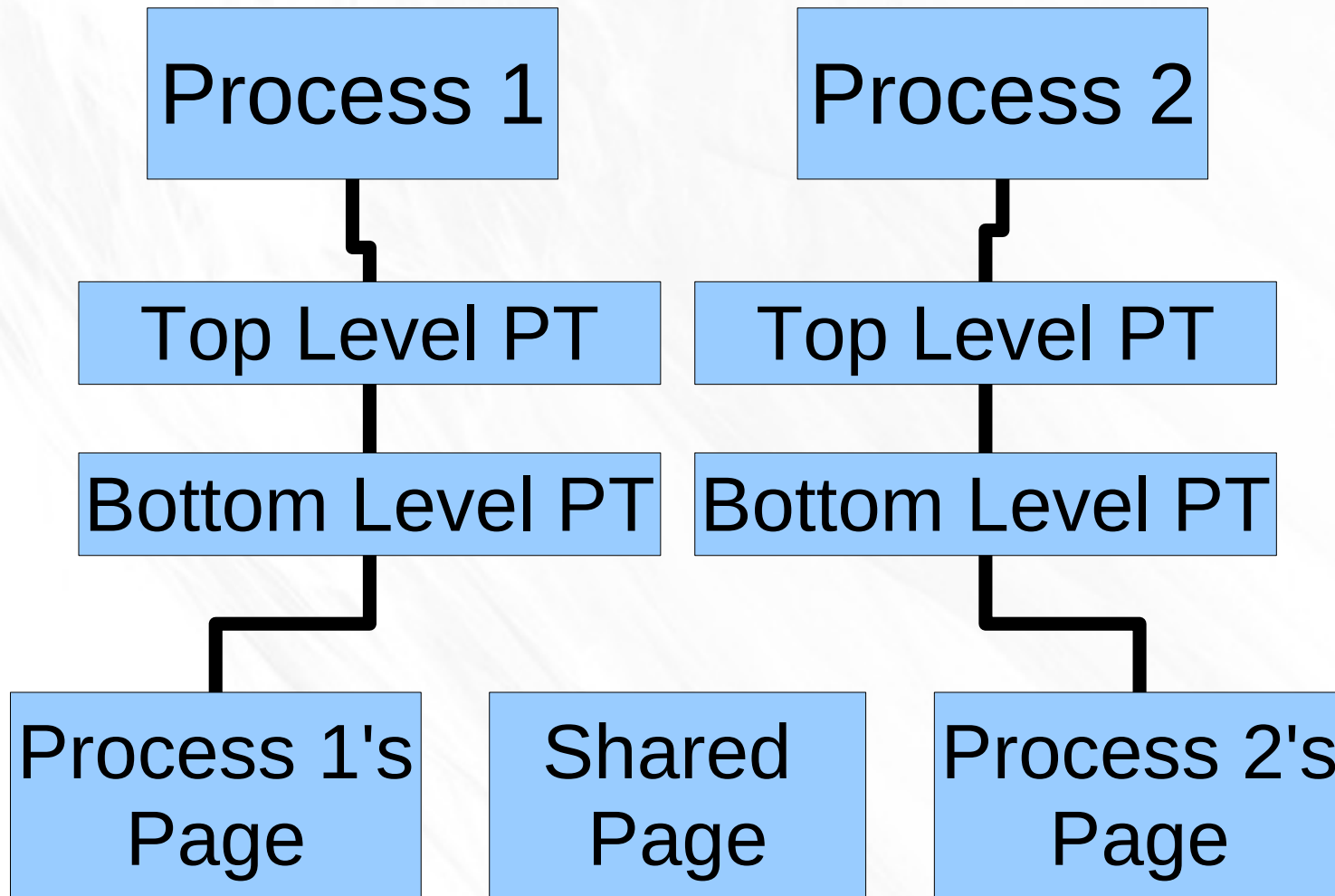
Two processes sharing a page



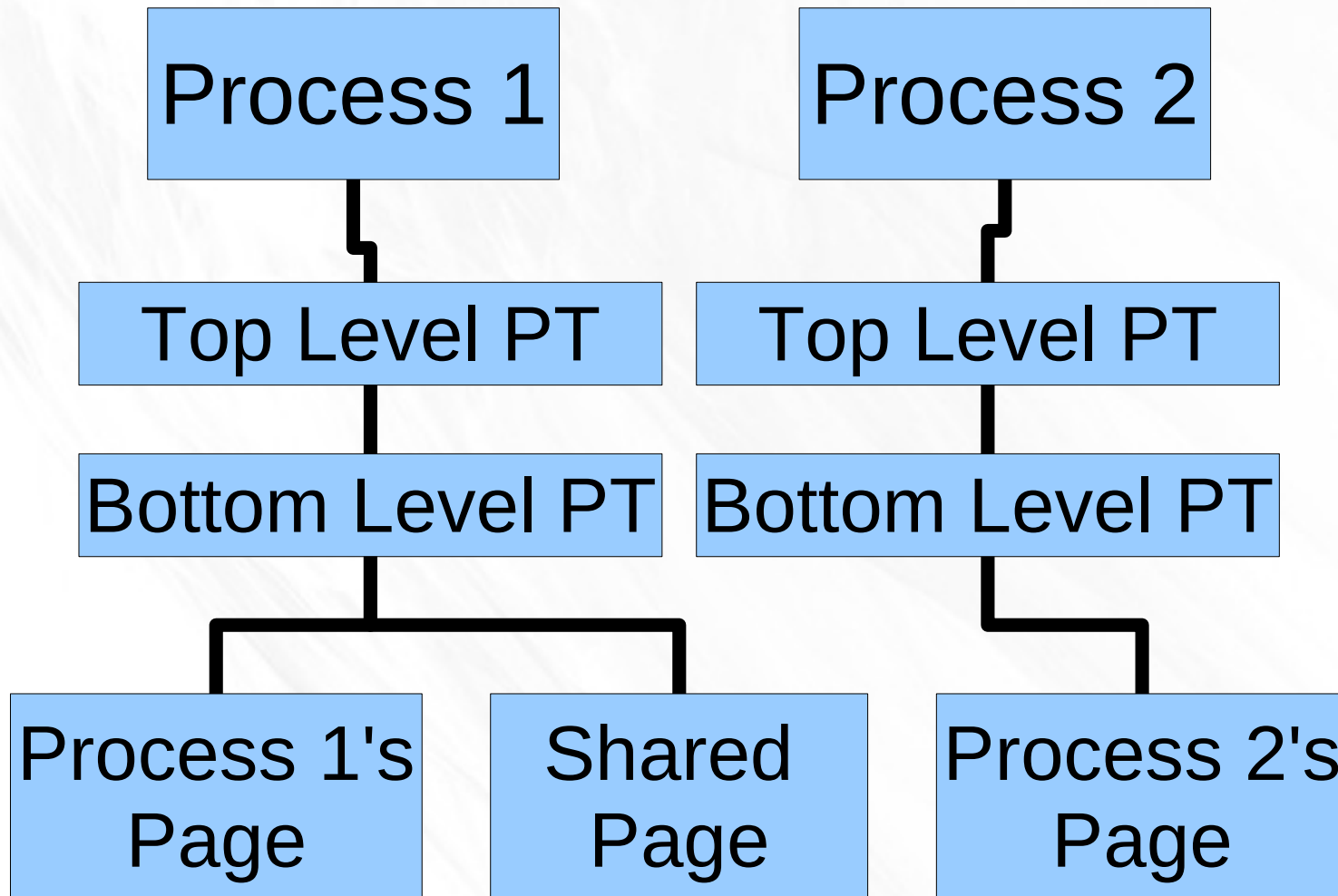
Memory might be shared, but Page Tables are not



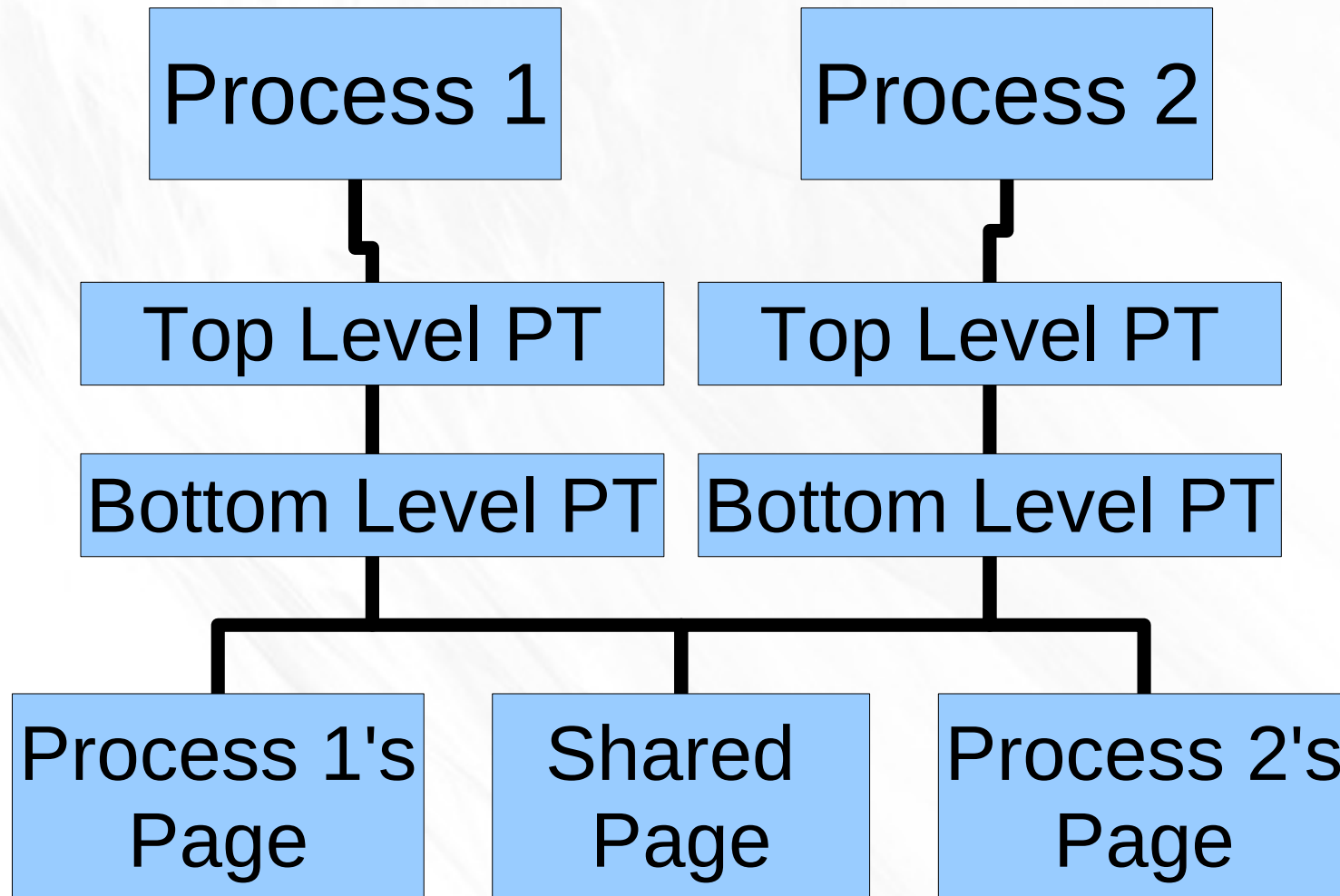
Memory might be shared, but Page Tables are not



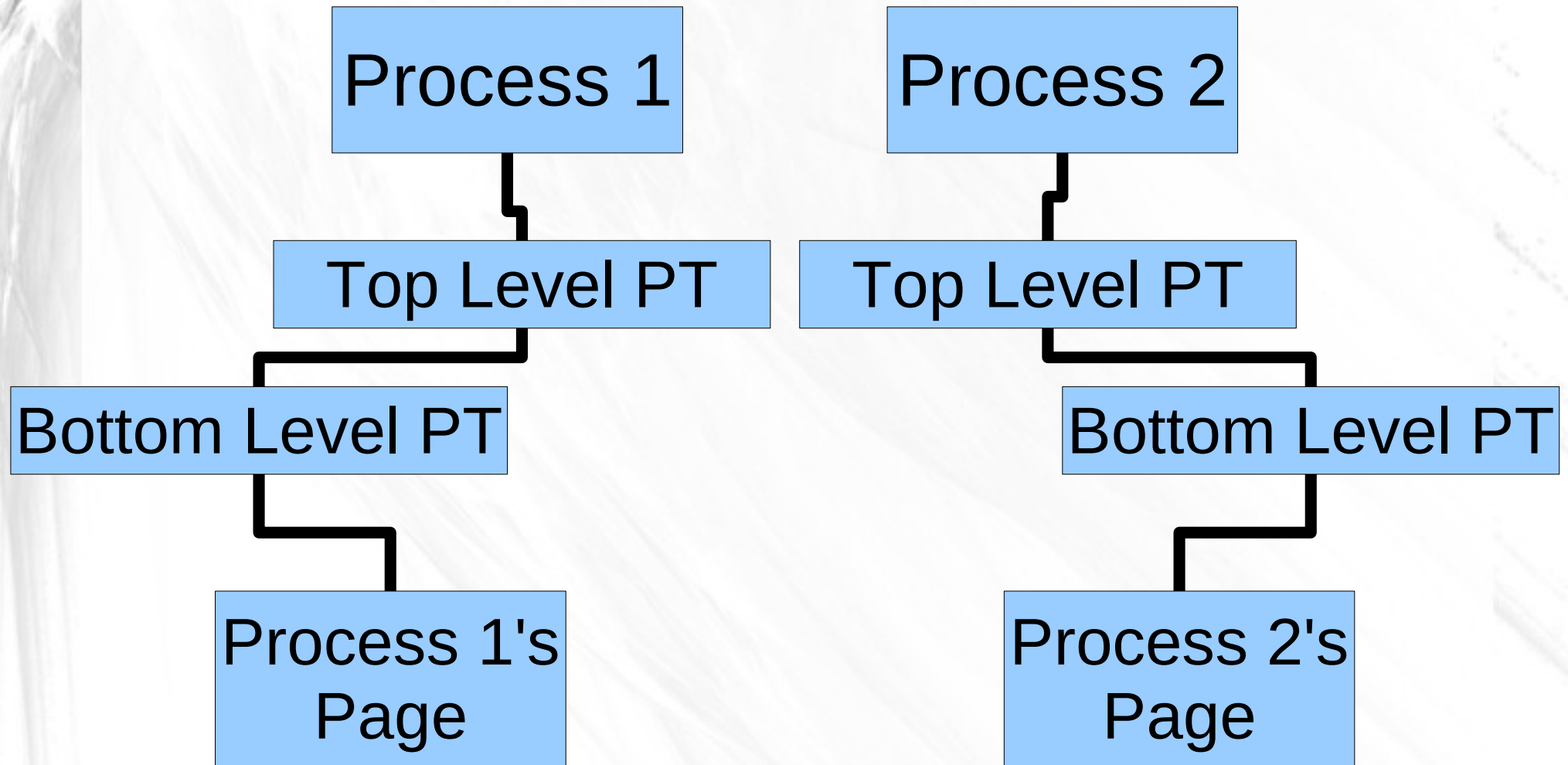
Memory might be shared, but Page Tables are not



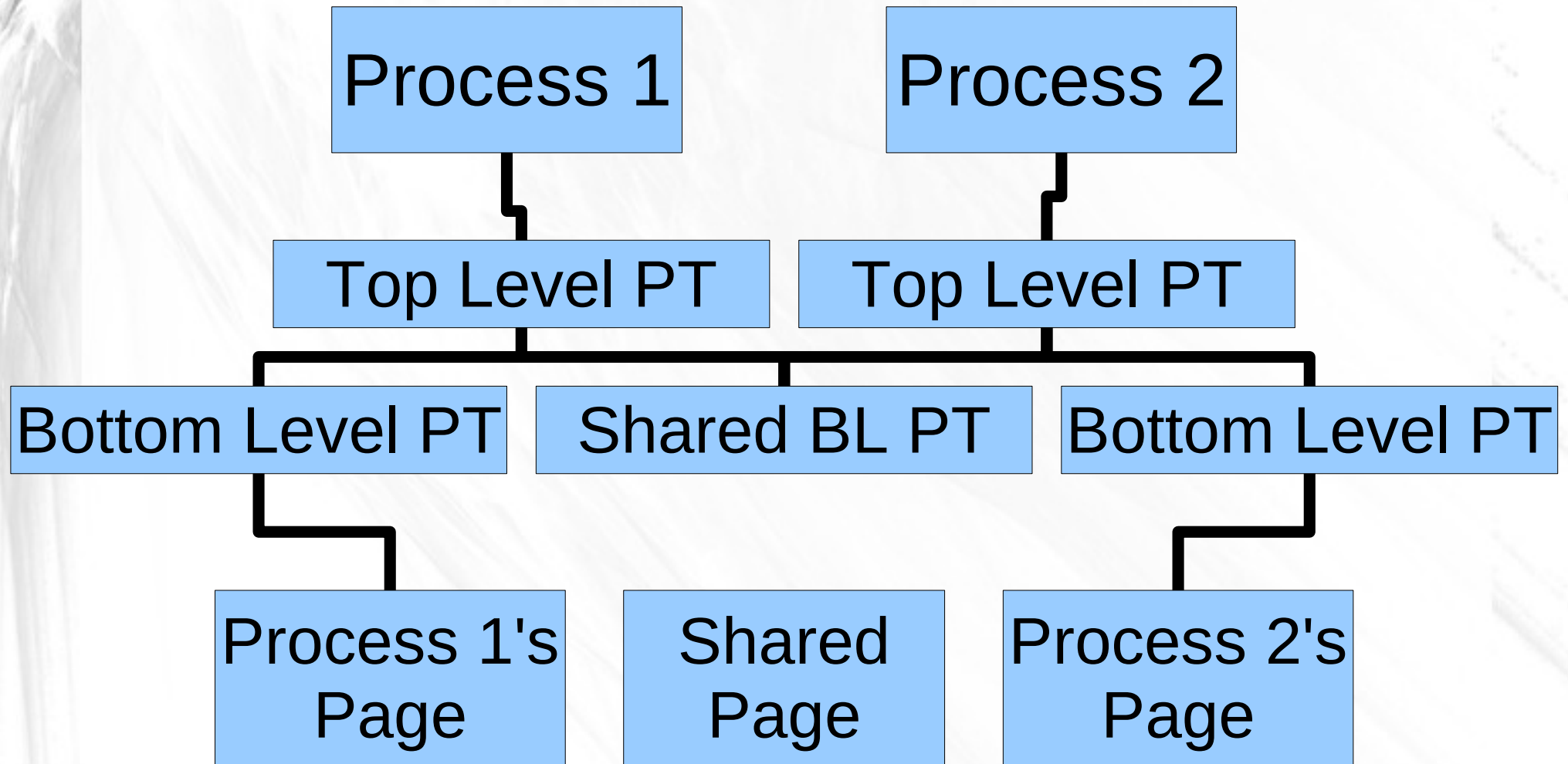
Memory might be shared, but Page Tables are not



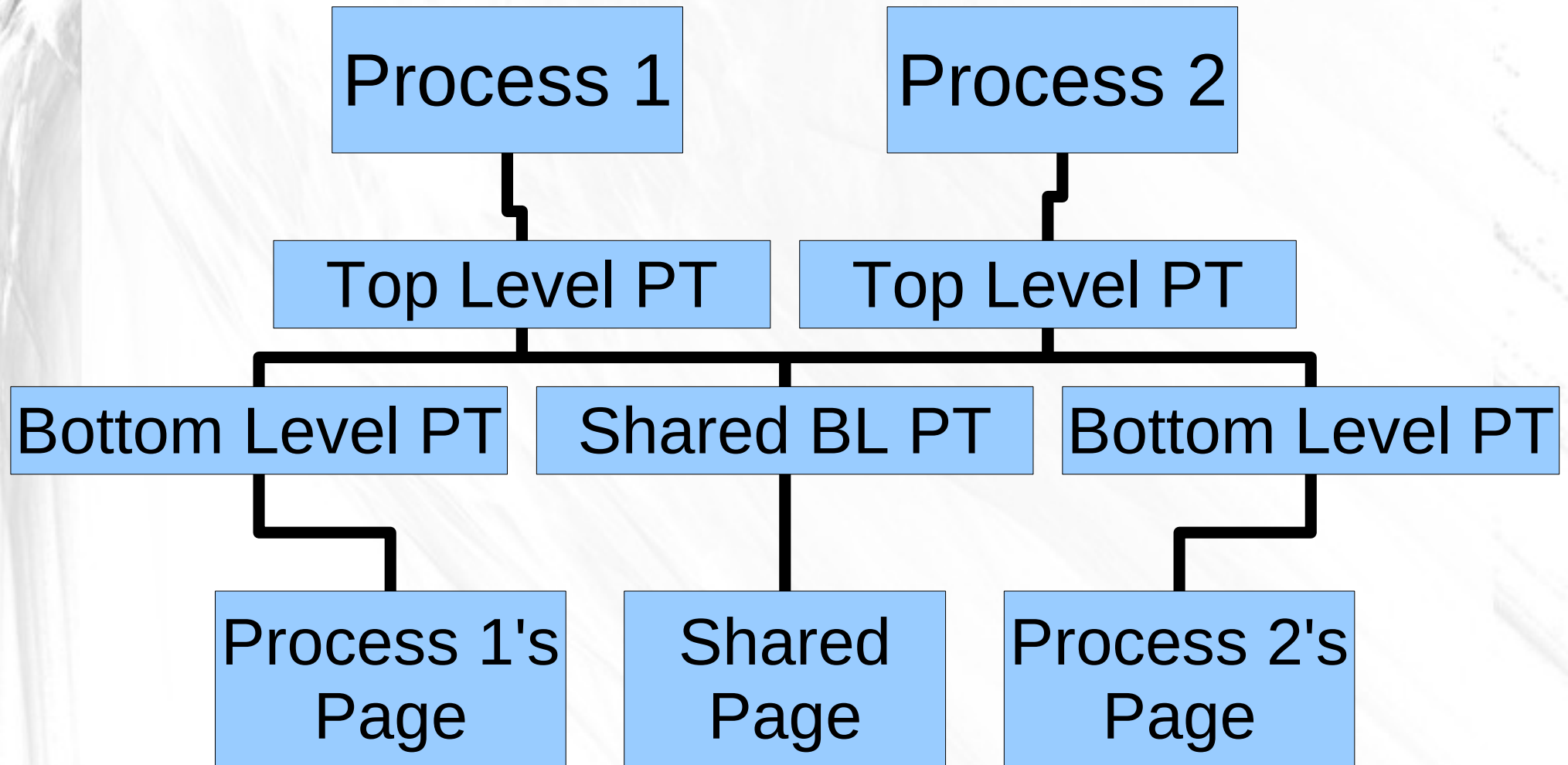
Corey PT Sharing



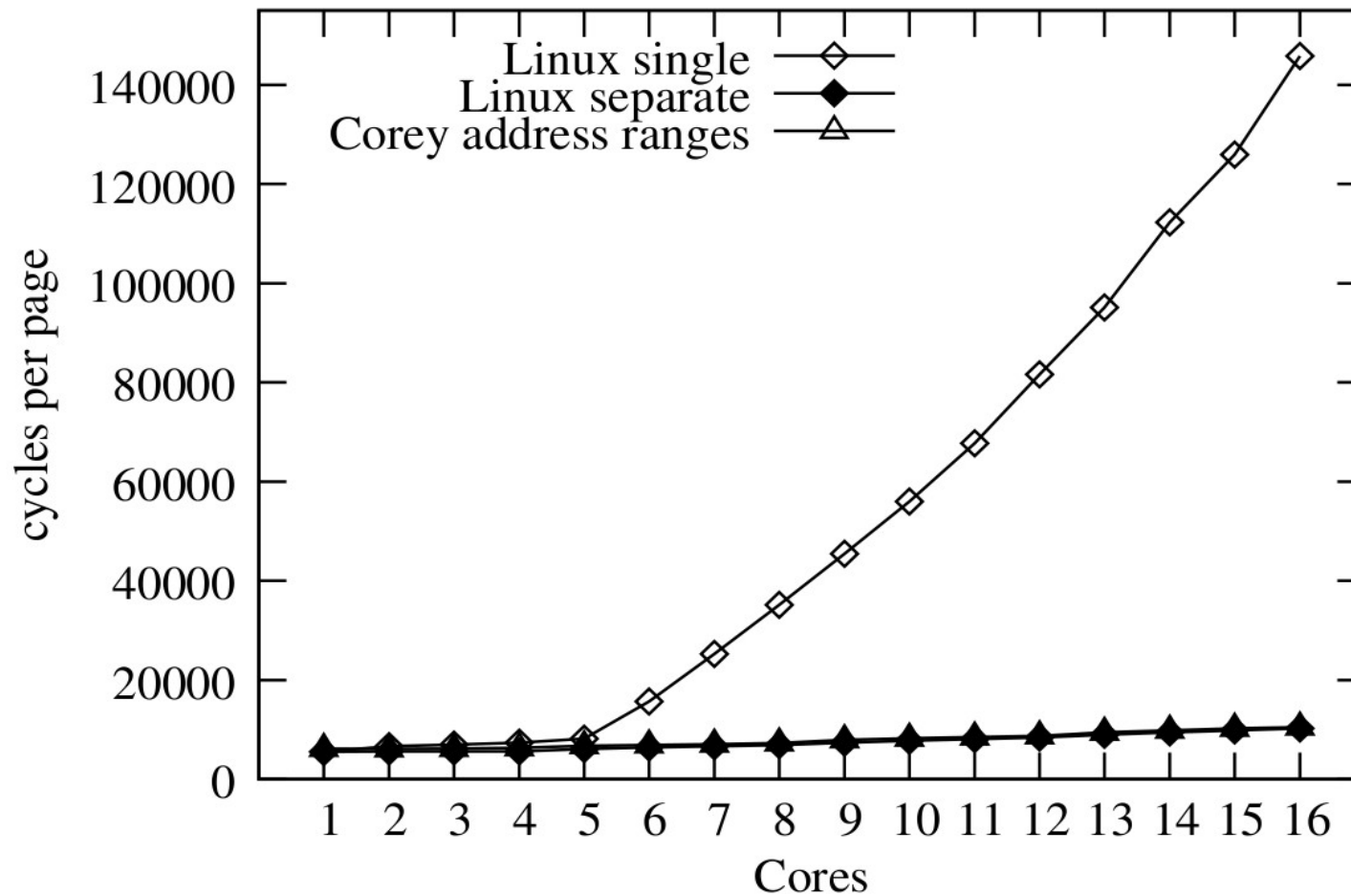
Corey PT Sharing



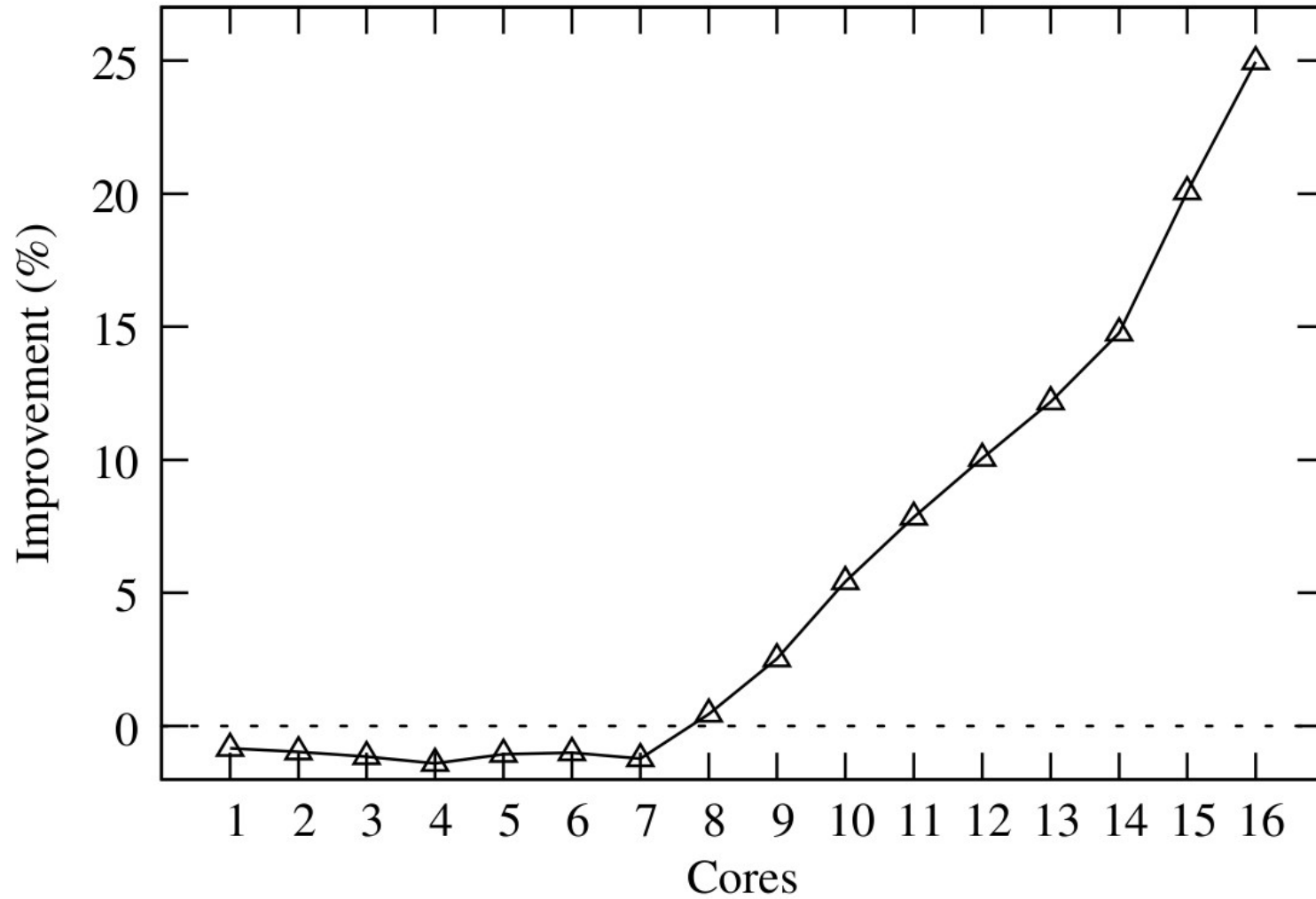
Corey PT Sharing



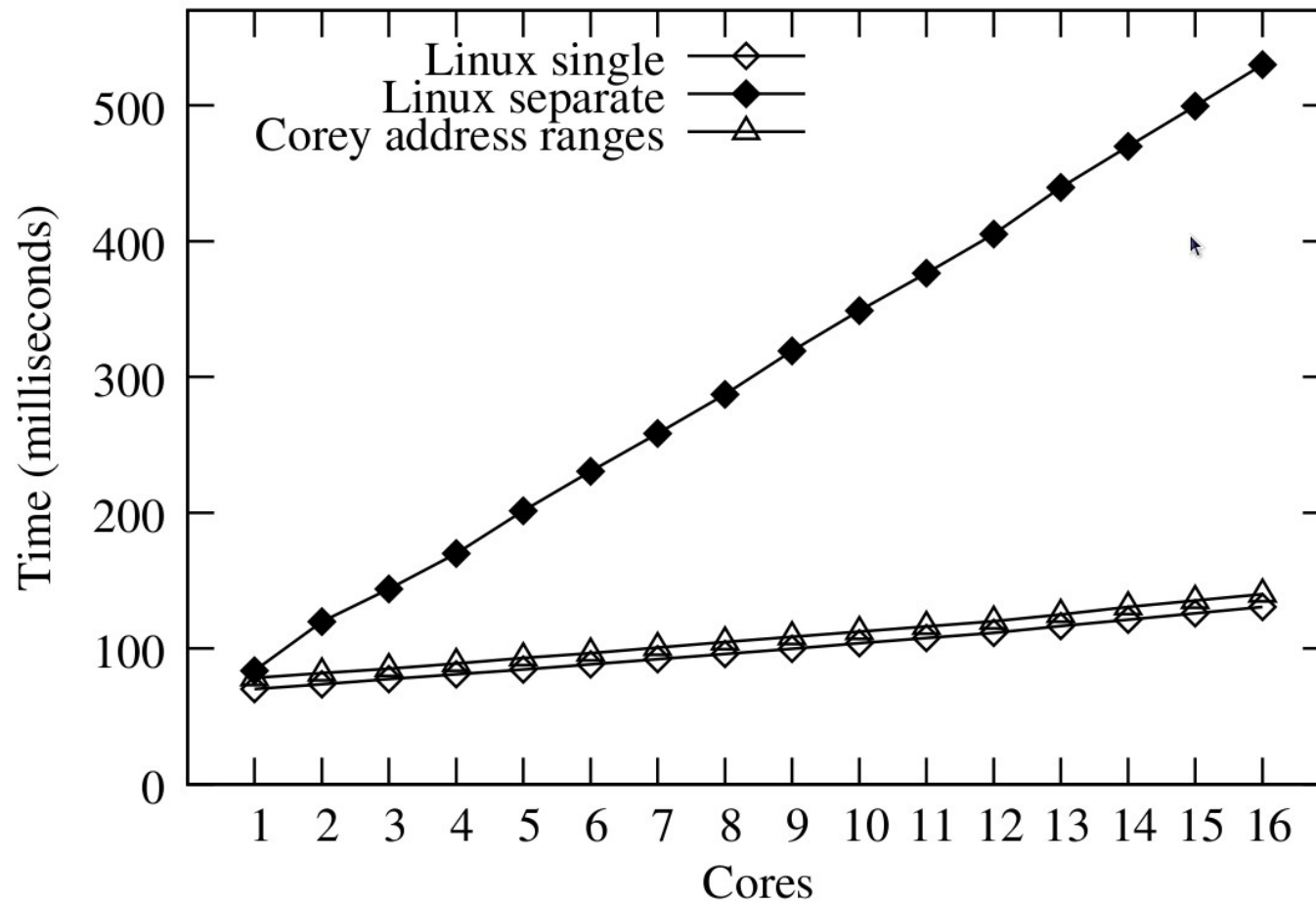
Corey MM: Evaluation



MapReduce

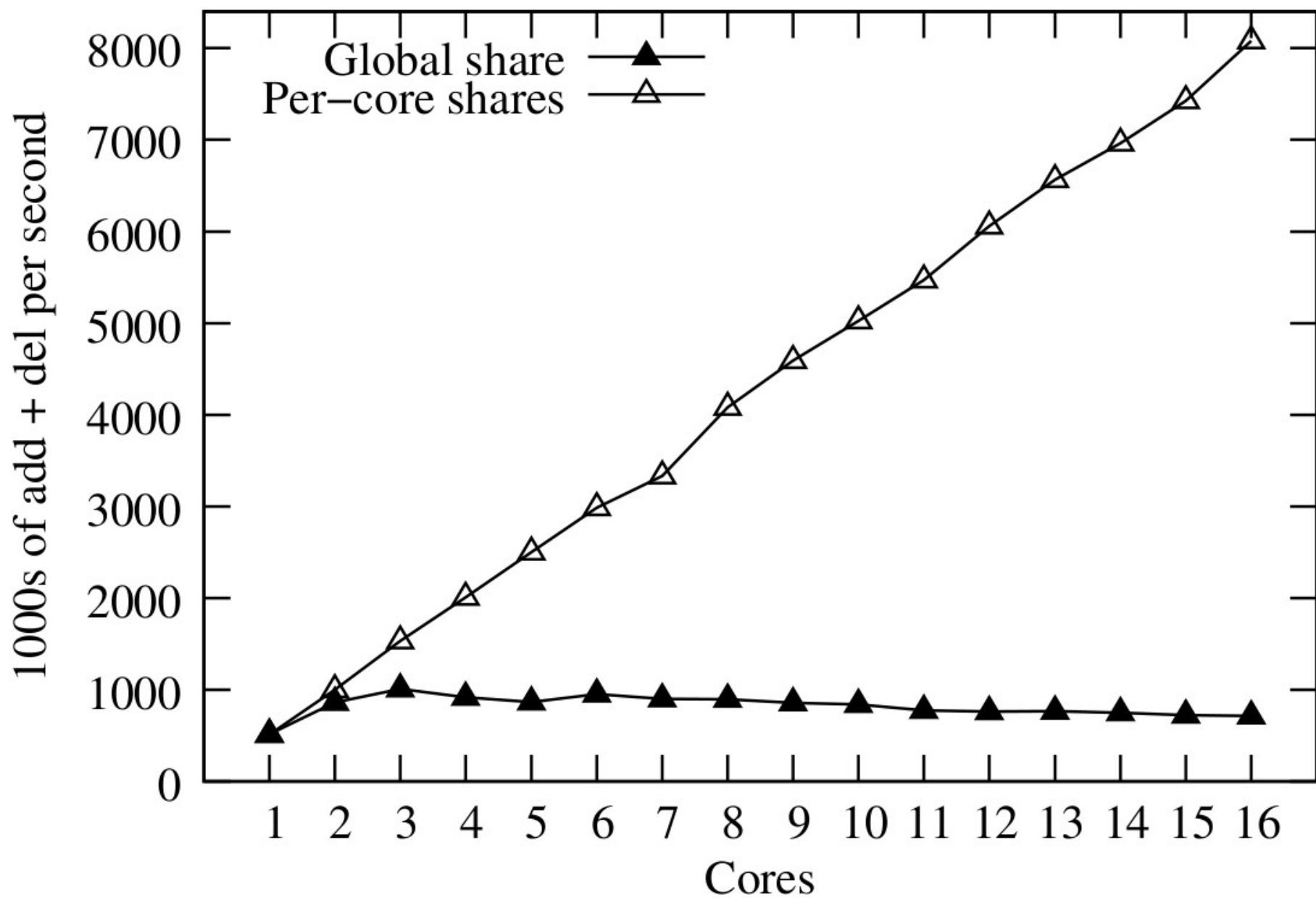


Corey MM: Evaluation



Shares: Similar, but for FDs

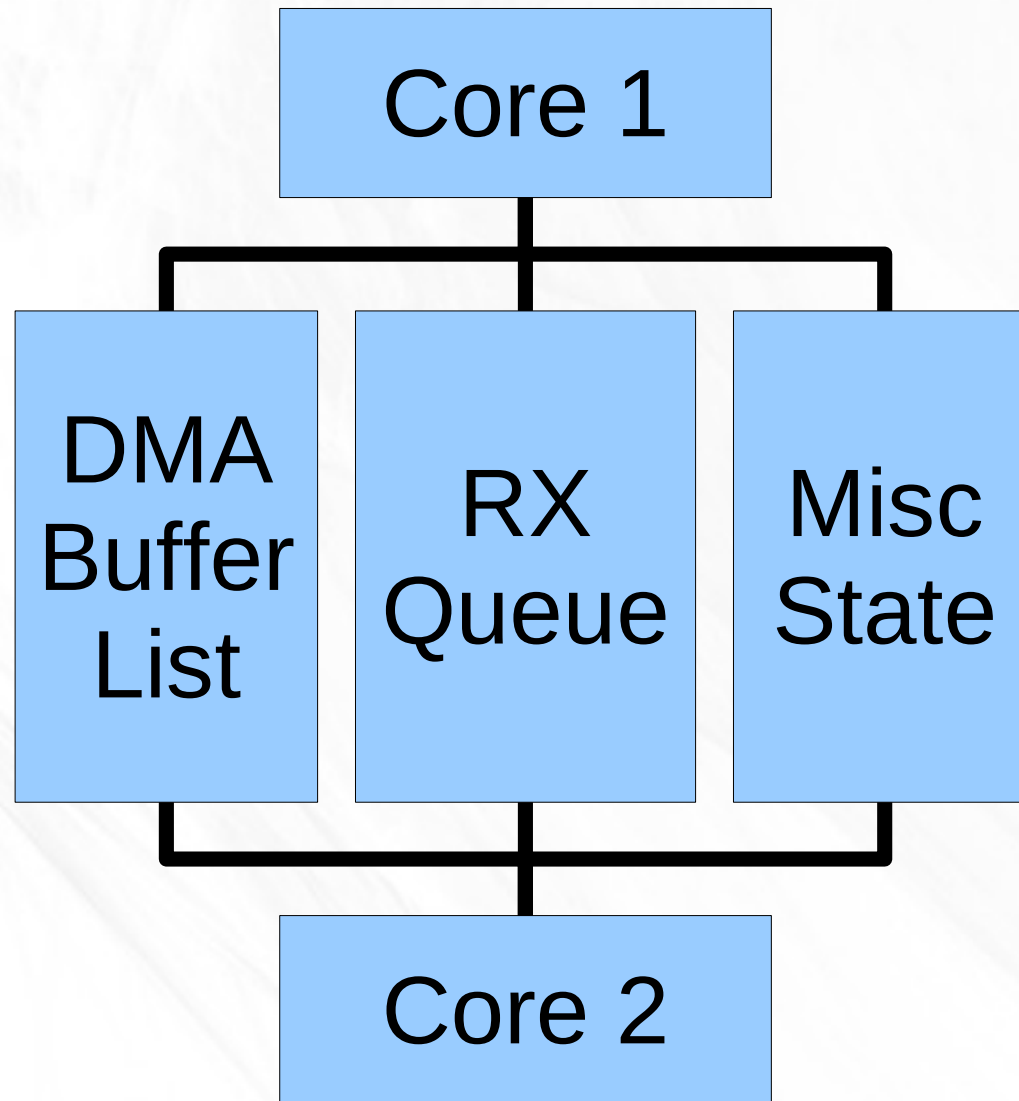
- Private and public FD tables
- Again, don't contend for the private table
- Shared shares and FD tables both serialization points



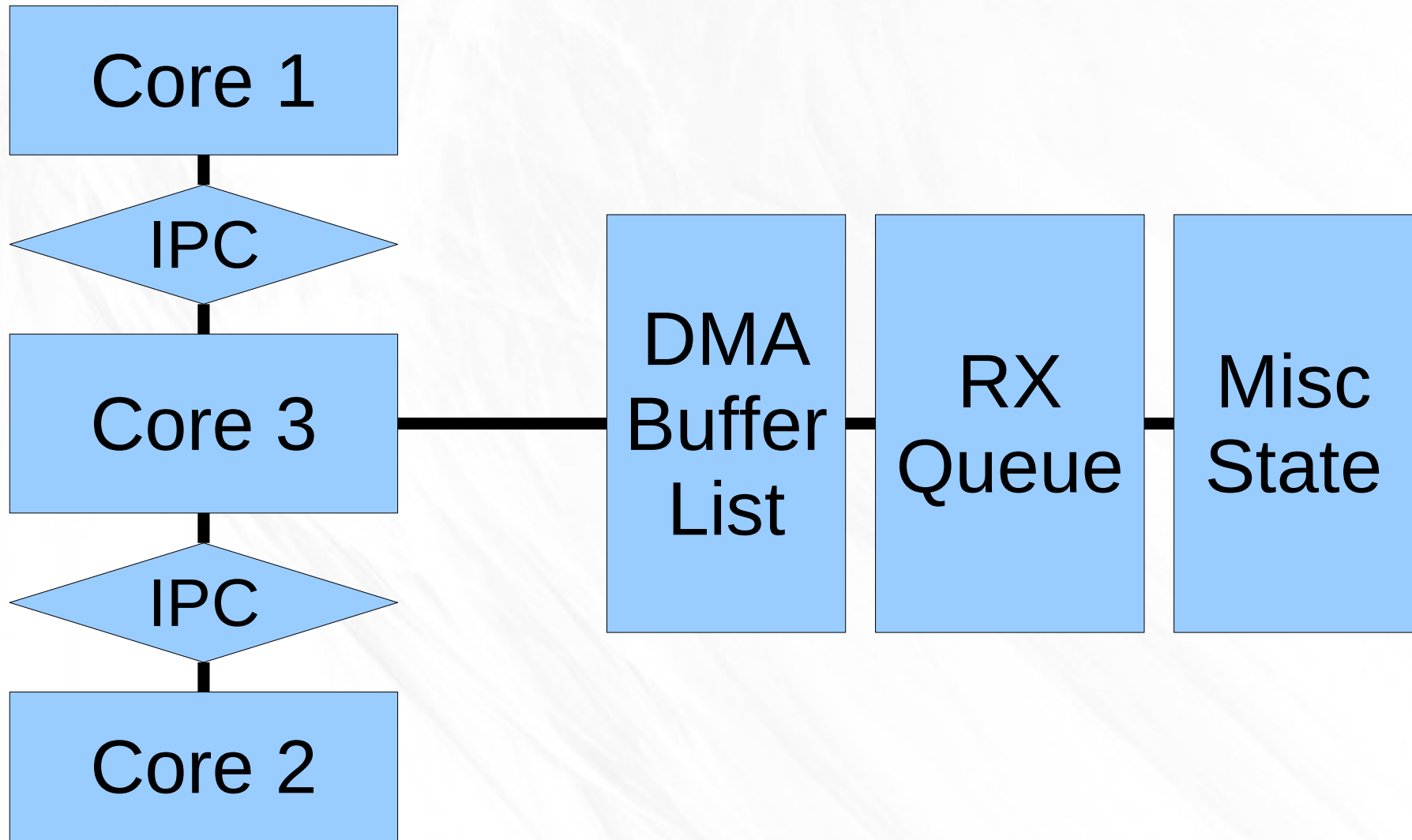
Kernel Cores

- Keep all device driver state core-local
- Ring buffer to/from kernel core
- Kernel core polls the ring and device
- User cores push requests and wait for replies

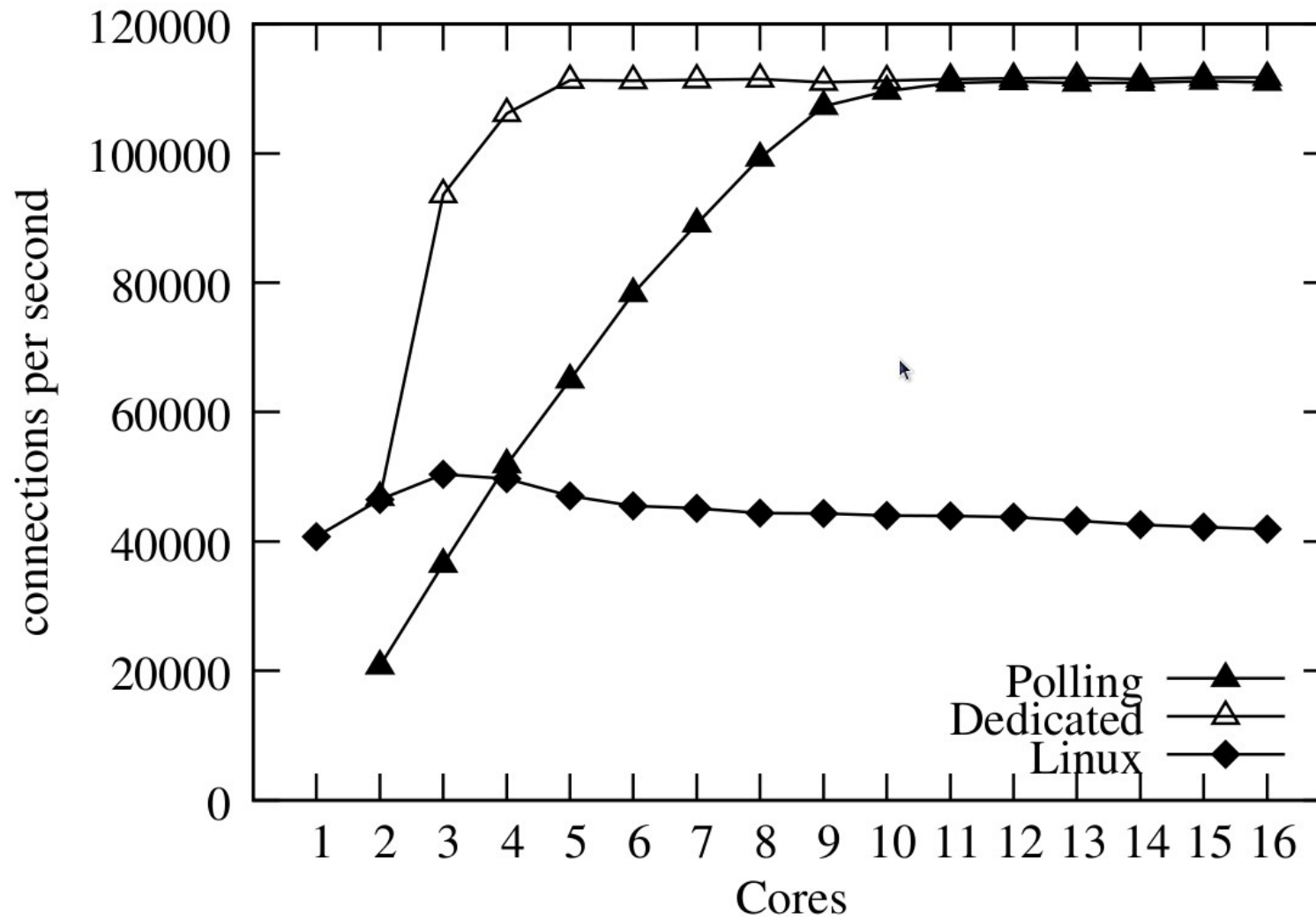
Without a Kernel Core



With a Kernel Core



Kernel Core Eval



Summary

- Earth shattering it is not
- But these are sensible optimisations
- And real progress
- Not a complete operating system
 - But we already have those