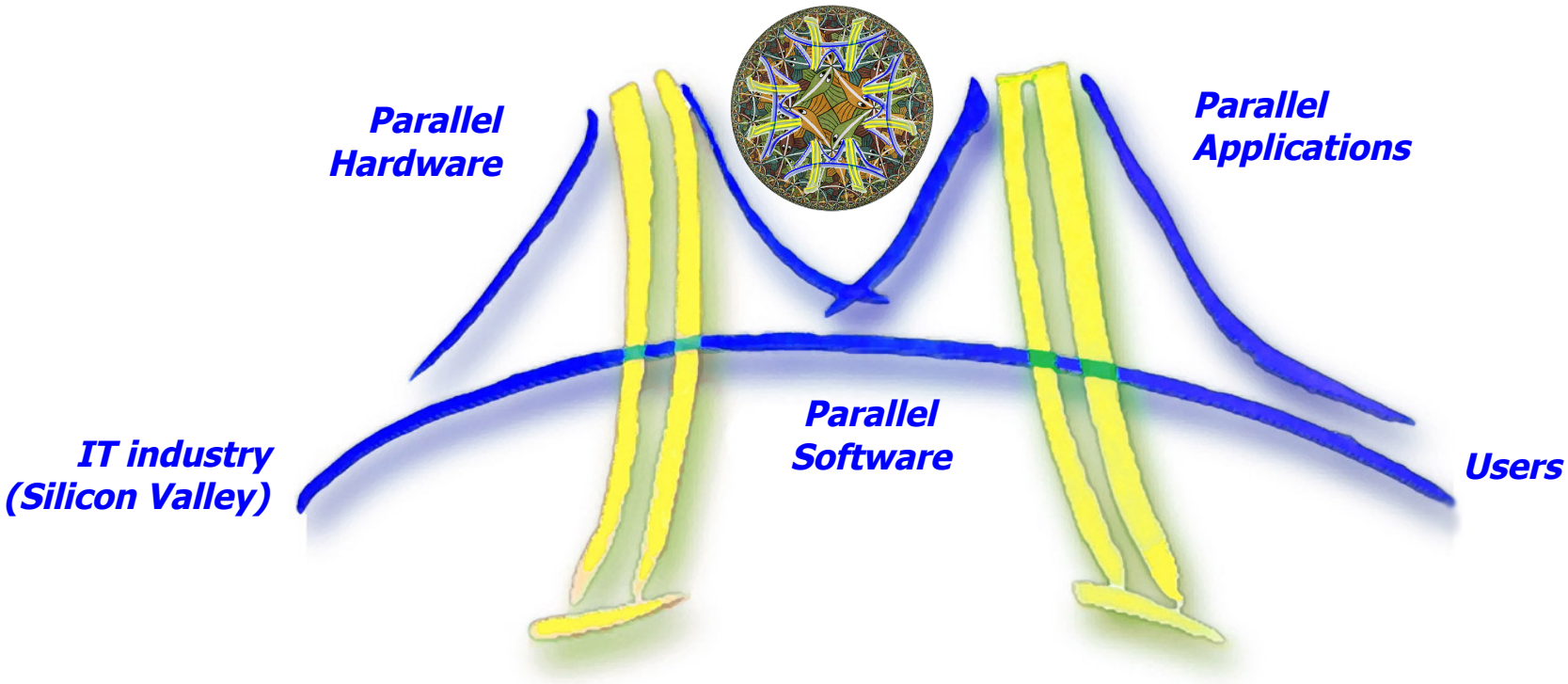
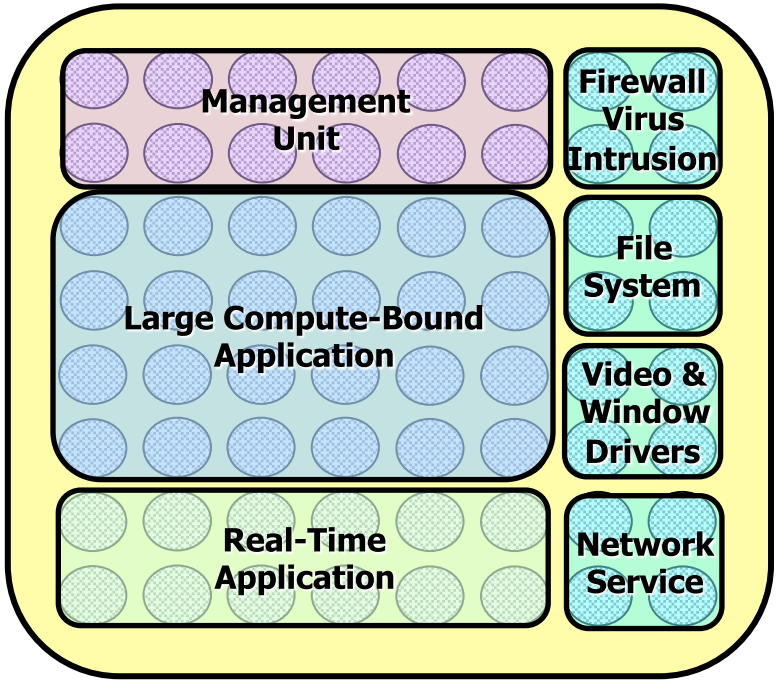


Abstractions for Scalable Operating Systems on Manycore Architectures

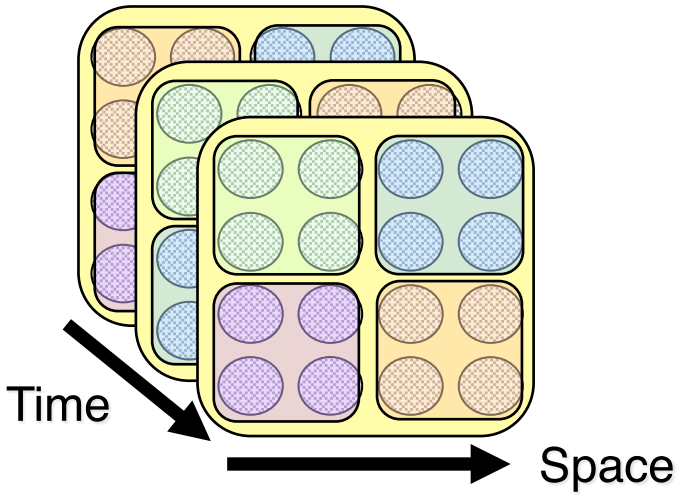
Kevin Klues, Barret Rhoden, David (Yu) Zhu, Paul Pearce, Eric Brewer, John Kubiatowicz

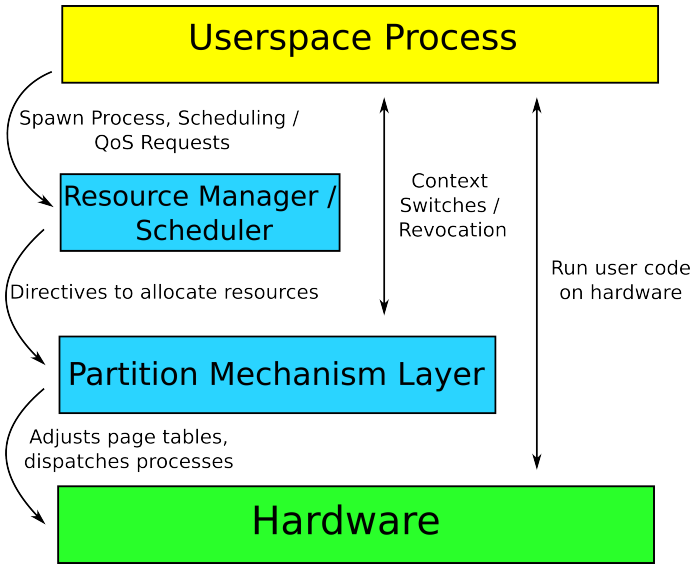
Asymmetrically Structured OS



- Management of the system limited to a small subset of cores
- Explicit separation of kernel services and user space applications onto different cores
- Communication done through dedicated channels
- Advantages:
  - Eliminates the need for per core run queues
  - Increases per core cache locality
  - Decreases cross core lock contention
  - Limits kernel interference with applications
  - Enables applications to perform remote asynchronous system calls

Guaranteed, Partitioned Resources

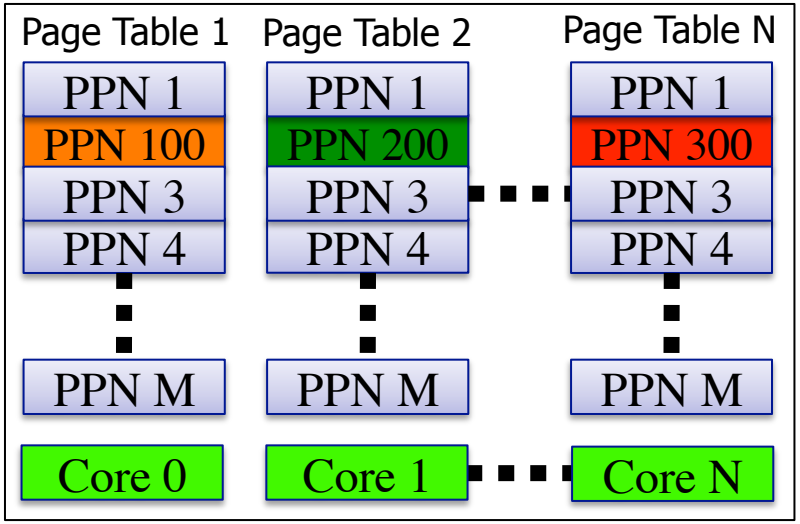




- Resources partitioned based on explicit requests from an application or system service
- Applications multiplexed in both time and space onto the hardware resources they have been granted
- Scheduling in the OS is coarse grained: based on enforcing resource guarantees (QoS)
- Scheduling in userspace is fine grained: based on mapping threads of control to the resources granted
- Resource guarantees enforced either in hardware or software depending on the architecture in use

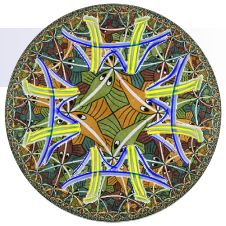
Private Memory Ranges

- Reserve range of addresses in an Address Space for processing per-context (or per-core) private data
- Most data is shared (e.g. file descriptors, security properties)
- But not everything (e.g. data to be processed in SIMD fashion)



- Advantages:
  - No locks needed to modify data in private memory ranges
  - Can do page re-mappings in private regions without requiring cross core TLB shootdowns
  - Eases development for data parallel applications
  - Allows for fast page remapping for streaming data applications

Acknowledgments:



From the UC Berkeley Parallel Computing Lab:
Ben Hindman, Juan Colmenares, Sarah Bird, Heidi Pan, Zach Anderson, Andrew Waterman, Krste Asanovic

From Lawrence Berkeley National Labs:
Eric Roman, Steven Hofmeyr, John Shalf, Costin Iancu, Kathy Yelick

Research supported by Microsoft (Award #024263) and Intel (Award #024894) funding and by matching funding by U.C. Discovery (Award #DIG07-10227).