Resource Management in Tessellation OS
Juan A. Colmenares, Sarah Bird, Henry Cook, Paul Pearce, John Shalf, Steven Hofmeyr, Krste Asanović, and John Kubiatowicz
1 ParLab, CS Division, UC Berkeley | 2 Lawrence Berkeley National Laboratory

1. Some Basic Goals
- Offer better support to a simultaneous mix of high-throughput parallel, interactive, and real-time applications
- Enable improved application performance by exploiting many-core hardware
- Allow applications to achieve high responsiveness
- Facilitate engineering applications with real-time guarantees
- Allow applications to consistently deliver the above performance improvements
- Enable rapid adaptation to changes in the application mix, resource availability, and other operational conditions

Other Important Goals
- Scalability
- Better handling of power-performance tradeoffs
- Additional protection, fault-containment, and security capabilities

2. Space-time Partitioning and Two-level Scheduling
- A Spatial Partition (or Cell) comprises a group of processors acting within a hardware boundary
  - Each cell receives a vector of basic resources
    - Some number of processors, a portion of physical memory, a portion of shared cache memory, and potentially a fraction of memory bandwidth
  - A cell may also receive
    - Exclusive access to other resources (e.g., certain hardware devices and raw storage partition)
    - Guaranteed fractional services (i.e., QoS guarantees) from other partitions (e.g., network service and file service)

- Spatial partitioning may vary over time
  - Partitioning adapts to needs of the system
  - Some cells persist while others change with time

- Scheduling at Level 1: Coarse-grained resource allocation and distribution at the cell level
- Scheduling at Level 2: Fine-grained application-specific scheduling within a cell

- A cell contains channel endpoints to other cells
  - Channels allow an application in a cell to access services and to interact with other applications residing in other cells
  - Communication between cells is controlled for security and QoS enforcement
  - Channels enable efficient and non-blocking message passing

3. Partition-management Layers

4. Resource-management Software Architecture

Research supported by Microsoft (Award #024263) and Intel (Award #024894) funding and by matching funding by U.C. Discovery (Award #DGO710227)