Tessellation OS: Partition Management and Two-level Scheduling
Juan A. Colmenares, Sarah Bird, Paul Pearce, and John Kubiatowicz
Par Lab, CS Division, University of California, Berkeley

1. Space-time Partitioning

- 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)

Goals of Space-time Partitioning
- Provide responsiveness and/or QoS guarantees and more predictable real-time behavior to (parts of) applications
- Achieve better handling of power-performance tradeoffs
- Offer additional protection, fault-containment, and security capabilities

2. Partition-management Software Layers

- Cell's Specification
  - Specification of physical resources
  - Specification of time-multiplexing policies
  - Specification of required services and QoSs

Policy Layer
- Basic time-mux'ing policies for cells
  - Pinning (i.e., null policy)
  - Time Triggering
  - Time Fraction
  - Round Robin

Mapping & Time-Multiplexing Layer

3. Two-level Scheduling

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

- 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
  - Message communication between cells is controlled for security and QoS enforcement
  - Channels enable efficient and non-blocking message passing

4. Future Demo Application

- Most of the engine's functionality
- The goal is to show that Tessellation can provide acceptable performance and time predictability

Research supported by Microsoft (Award #024263) and Intel (Award #024894) funding and by matching funding by U.C. Discovery (Award #DI00710227)