Rethinking Computer Architecture

Wen-mei Hwu
University of Illinois, Urbana-Champaign

Celebrating Yale@75
September 19, 2014
What Yale and I debate about in Samos and other places.
Application developers should not deal with variations in HW.
The HPS Vision - 1985

- One static program (algorithm)
- Many execution resource configurations
  - Types of Function Units
  - Number of Function Units
  - I-Fetch bandwidth
  - Memory Latencies
- Key enablers
  - Branch prediction
  - Resource mapping
  - Restricted data flow execution
  - Sequential retirement

Patt, Hwu, Shebanow, "HPS, A New Microarchitecture: Rationale and Initial Results
Some Lessons Learned

• Parallelism and communication costs motivate algorithm changes
  – Locality vs. parallelism tradeoffs in libraries

• Performance and efficiency pressure breaks abstraction
  – Java is great for abstraction portability but insufficient for performance and efficiency
  – MPI, OpenMP apps often explicitly handle hardware-centric details
Productivity and Performance

Triolet

\[ ys = \left[ \text{sum}(x \times \cos(r \cdot k)) \text{ for } (x, k) \in \text{zip}(xs, ks) \right] \text{ for } r \text{ in par(rs)} \]

- Library functions factor out data decomposition, parallelism, and communication

- CPUs/GPUs/Accelerators or entire nodes are the new function units
- Compute functions are the new instructions
- Distributed execution of functions to avoid data movement
  - Accelerators in/near Network I/O, Disk I/O, DRAM
  - Some come with own DRAM/SRAM for bandwidth
Example - Desirable Data Transfer and Compute Behavior

- Runtime/OS should map buffers and compute functions
  - I/O buffer to any major DRAM/SRAM
- Compute functions (decompression) to any CPU/GPU/accelerators
Example - Today’s Data Transfer and Compute Behavior

Each additional copy diminishes application-perceived bandwidth
A Call to Action

• Redefine system architecture
  – HAS/CUDA 6.0 a step in the right direction

• Redefine ISA binary standard
  – SPIR/HSAIL/PTX with finalizers a step in the right direction

• Redesign OS/Runtime for data and compute mapping
  – UNIX/Linux overdue for redesign

• Provide performance portable domain libraries to sustain abstraction
  – High-level mechanisms such as Triolet and Tangram to fuse and tune library code into apps
Congratulations, Yale!