Single Node Performance

Kunle Olukotun

Jarek Nieplocha

Multithreading and Latency Hiding

- Single thread performance critical because of Amdahl law
  - XMT limited application reach
- How many outstanding memory references?
  - Cell has 16 per SPE, but with DMA
  - Prefetching/streaming with DMA works for some apps
- Applications would like a lot of HW threads (like XMT)
  - Easier and cheaper to do latency hiding
  - More expensive to support

Heterogeneity

- SIMD vs vector designs
- Future systems will be heterogenous
  - Wide cores, narrow cores, vectors, app specific cores
  - Challenge to applications
  - Want compilers manage heterogenuity rather than leave to programmers

Programmability

- Mixed programming models hard to use
  - MPI+OpenMP
- Would like single programming model
  - XMT has this
- Make it easy to develop kleenex code
  - Lots of graph code run once

Load Balancing

- Future applications will have increasing degree of load imbalance
- Who should manage it?
  - Graph algs.

Locality and Caches

- Graph algorithms have little locality
- Most people believe that numerical algorithms have locality
  - Possibly we know only of those
- Caches
  - Should have the ability to turn off
  - No caching of global data
  - Software control of globally shared data

Performance Tools

- Hardware profiling
  - Hardware support to tell programmer how code is performing
- Provide programmer relevant feedback