The Memory Latency Problem

- Commodity DRAM is optimized mainly for capacity, not latency
  - 16x increased capacity vs. 1.3x reduced latency

Latency-Capacity Tradeoff

- 

Leveraging the TL-DRAM Substrate

- Fully transparent (no change to system)
  - Use near-segment as hardware-managed cache
    - Far segment: Main memory
    - Near segment: Caches an accessed row
    - Memory controller manages the near segment
  - Use near-segment as software-managed cache
    - OS/VMM manages the near segment
  - Multi-level main memory
    - Allocate from fast vs. slow DRAM
    - Application or system software decides where a page goes

Results

- Single-core
  - System: CPU: 5.3GHz/LLC: 512KB (per core)
  - Memory: DDR3-1066, Row-interleaved & Closed-row
  - Benchmark: TPC, Stream, SPEC CPU2006, random-access
  - Simulation: in-house x86 simulator with detailed memory model

- Multi-core
  - SC: 11.5%, WMC: 8.5%, BBC: 20.1%-17.9%
  - Normalized Energy Consumption

Summary and Ongoing Work

- TL-DRAM: A new memory architecture that introduces latency heterogeneity by keeping technology homogeneity
  - Same chip, same technology: fast and slow portions
- Exposing TL-DRAM to system software
  - System software management algorithms
- Exploring Tiered Latency in NVM
  - Could be easier to adopt
- Fitting TL-DRAM into DRAM/NVM/Flash/Disk cooperative page management and allocation mechanisms