A HyperTransport-Enabled Global Memory Model For Improved Memory Efficiency

Jeffrey Young, Sudhakar Yalamanchili

School of Electrical and Computer Engineering
Georgia Institute of Technology

Federico Silla, José Duato

Universidad Politecnica de Valencia

Talk Outline

- Memory Efficiency Challenges
- Dynamic Partitioned Global Address Space (DPGAS) model
  - Architectural Support
  - Memory Management
  - Performance Evaluation
- Lessons Learned and Conclusions

Memory Efficiency Challenges

- Modern data centers are presenting challenges in terms of cost and power efficiency
  - Statistics suggest that 1.5 % of all U.S. energy costs go to datacenters and this cost could double by 2011 [1]
  - Focus has been on CPU power and overall cooling of datacenter
  - Memory can be a dominant consumer of cost and power [2]

Photo from http://eetd.lbl.gov

1) EPA Report to Congress on Server and Data Center Efficiency, 2007

2) C. Lefurgy, et al., Energy Management for Commercial Servers, IEEE Computer 2003

Power Challenges

SUN MD S20: Water cooled containers 187.5KW

Google data center in Oregon

Power densities of 10-20 KW/m^2 → footprint → cost

From R. Katz, "Tech Titans Building Boom," IEEE Spectrum, February 2009,http://www.spectrum.ieee.org/feb09/7327

Improving Memory Efficiency

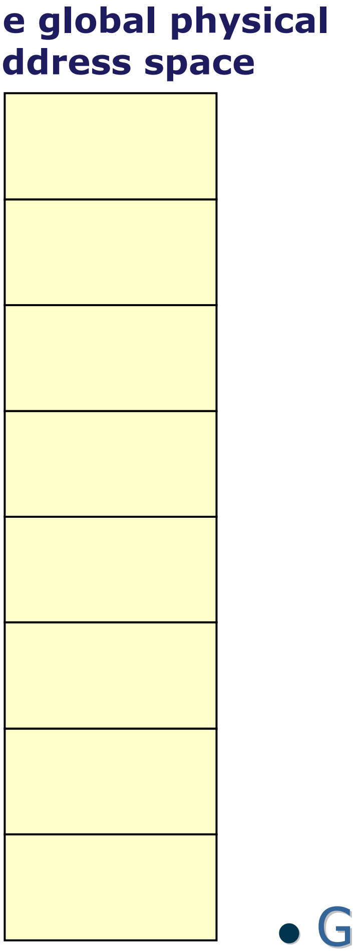
- Workloads are time-varying
  - Memory requirements can vary from 1 GB to 4 GB for average server applications [1]
  - System workloads have time varying footprints
- Provision memory/blade for the average case not peak
- On-Demand Sharing of physical memory
  - Virtualize the location of physical memory location
- Enabled by low latency HyperTransport

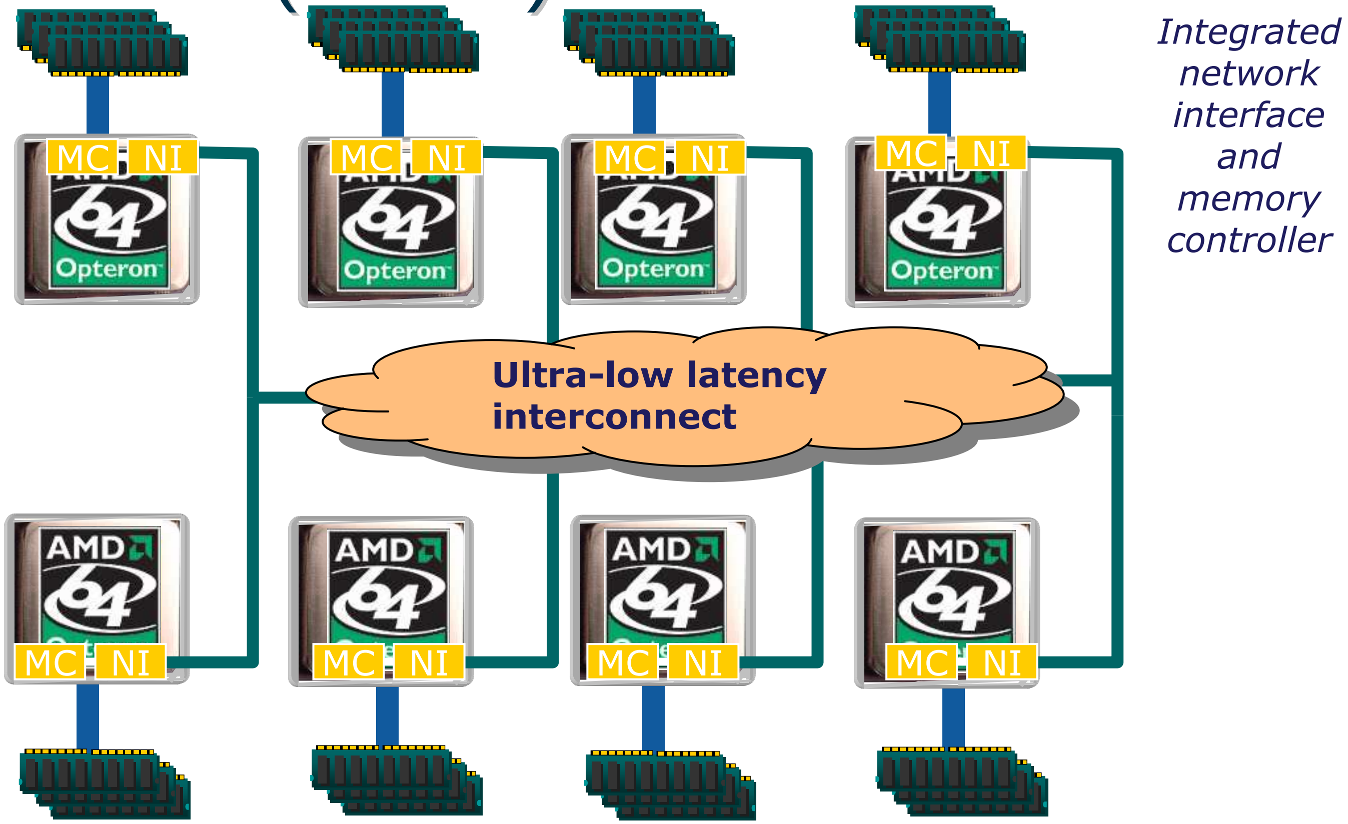
1) S. Chalal and T. Glasgow. Memory sizing for server virtualization. 2007. http://communities.intel.com/docs/

Proposed Approach

- A dynamically managed system-wide global address space
  - Physical address spaces dynamically mapped across memory controllers as needed
- Integrated network interface and memory mapping unit
  - Memory mapping integrated into the HyperTransport interface
  - Bridge to commodity or specialized interconnection networks
- Location-aware page allocators
  - Demand driven
  - Coordinated cross operating systems

Partitioned Global Address Space Systems (PGAS)

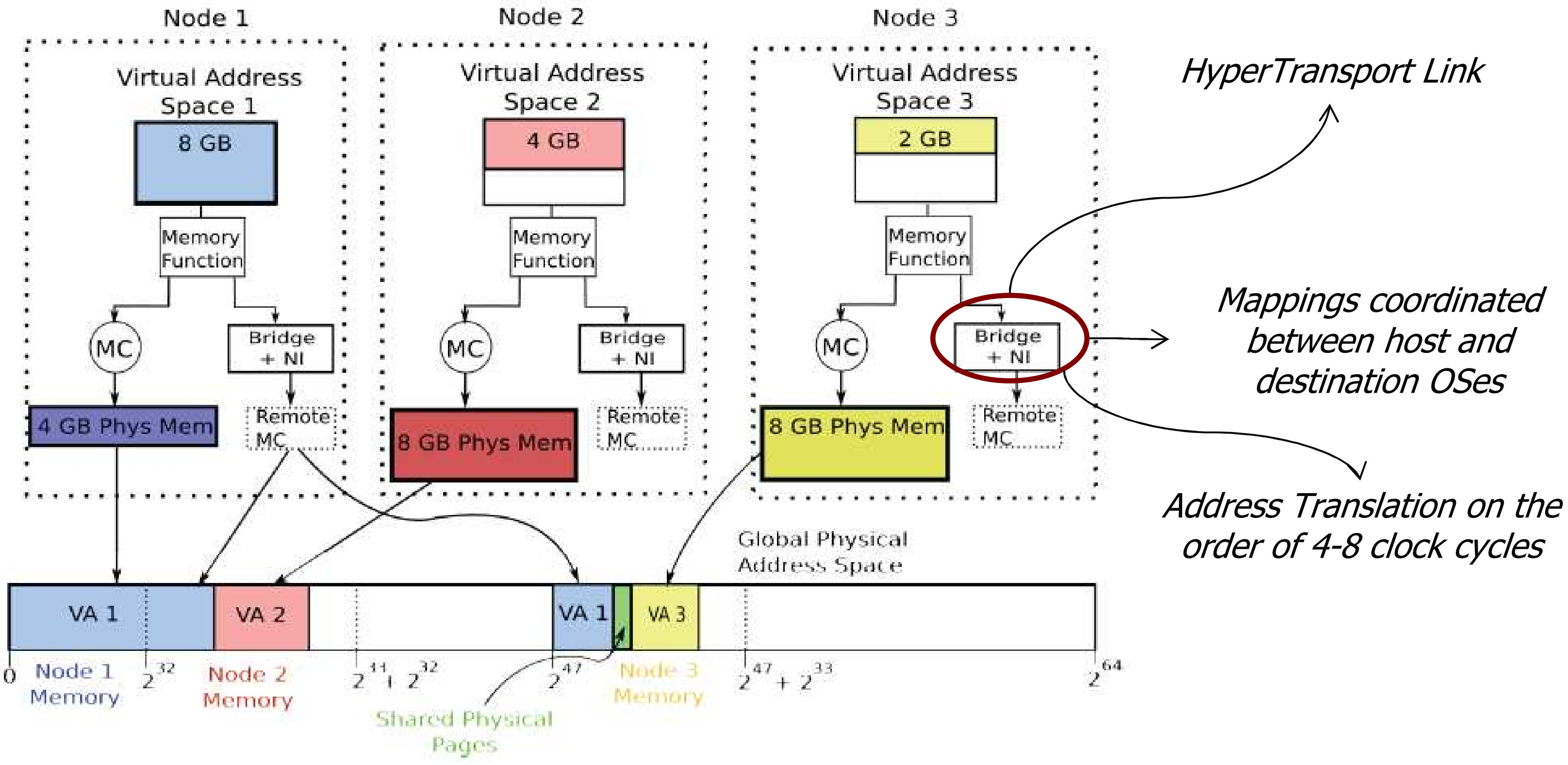



MC
NI
Opteron
AMD
64
Ultra-low latency interconnect
Integrated network interface and memory controller

- Global Address Space Partitioned Into Local and Remote Partitions Based on Access Latency

Low Latency, System-Wide Non-Uniform Memory Access

System View



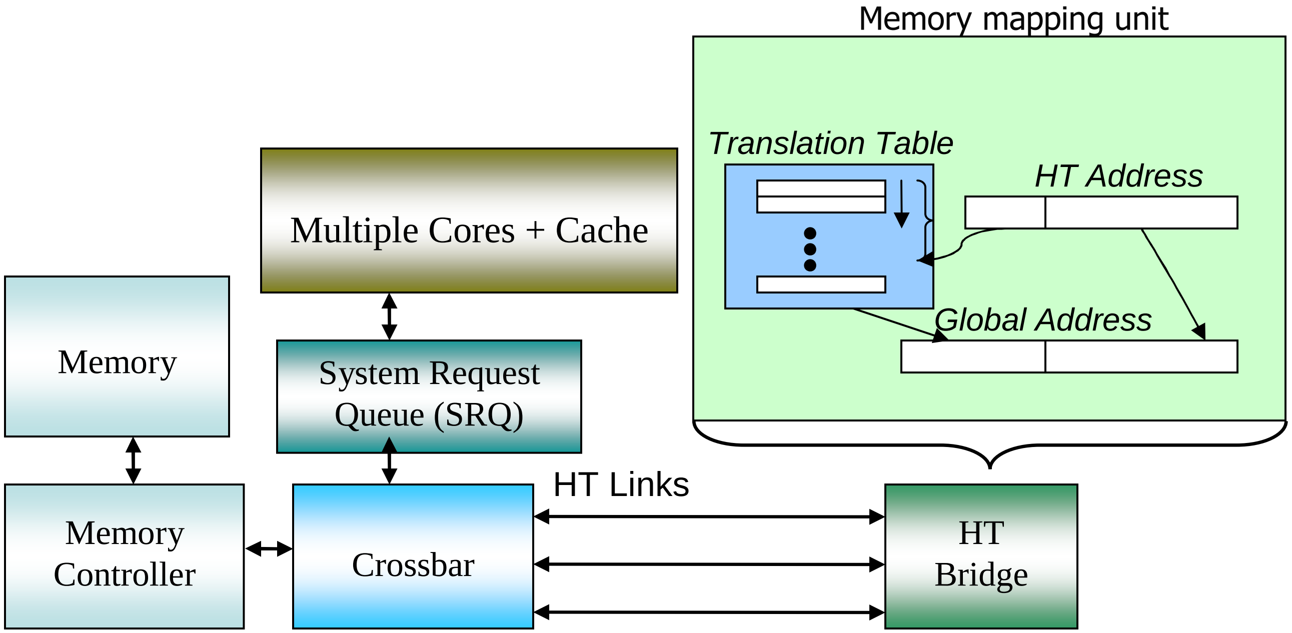
HyperTransport Link

Mappings coordinated between host and destination OSes

Address Translation on the order of 4-8 clock cycles

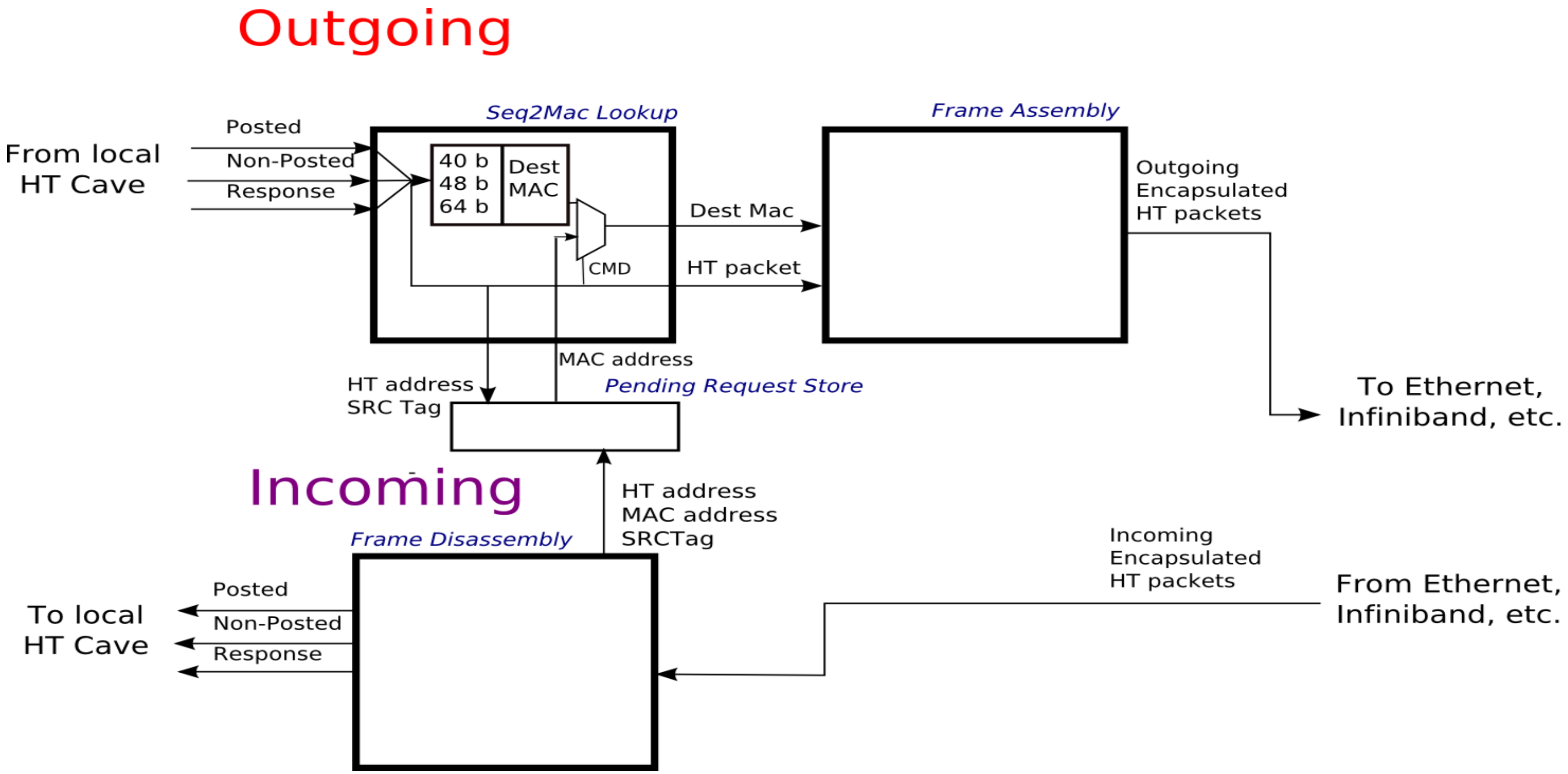
- Portion of the HT address space mapped to remote memory
- Protection issues handled by virtual memory system
  - Bridge mapping handled and coordinated by OS
  - System-wide RDMA analogy

Architectural Support



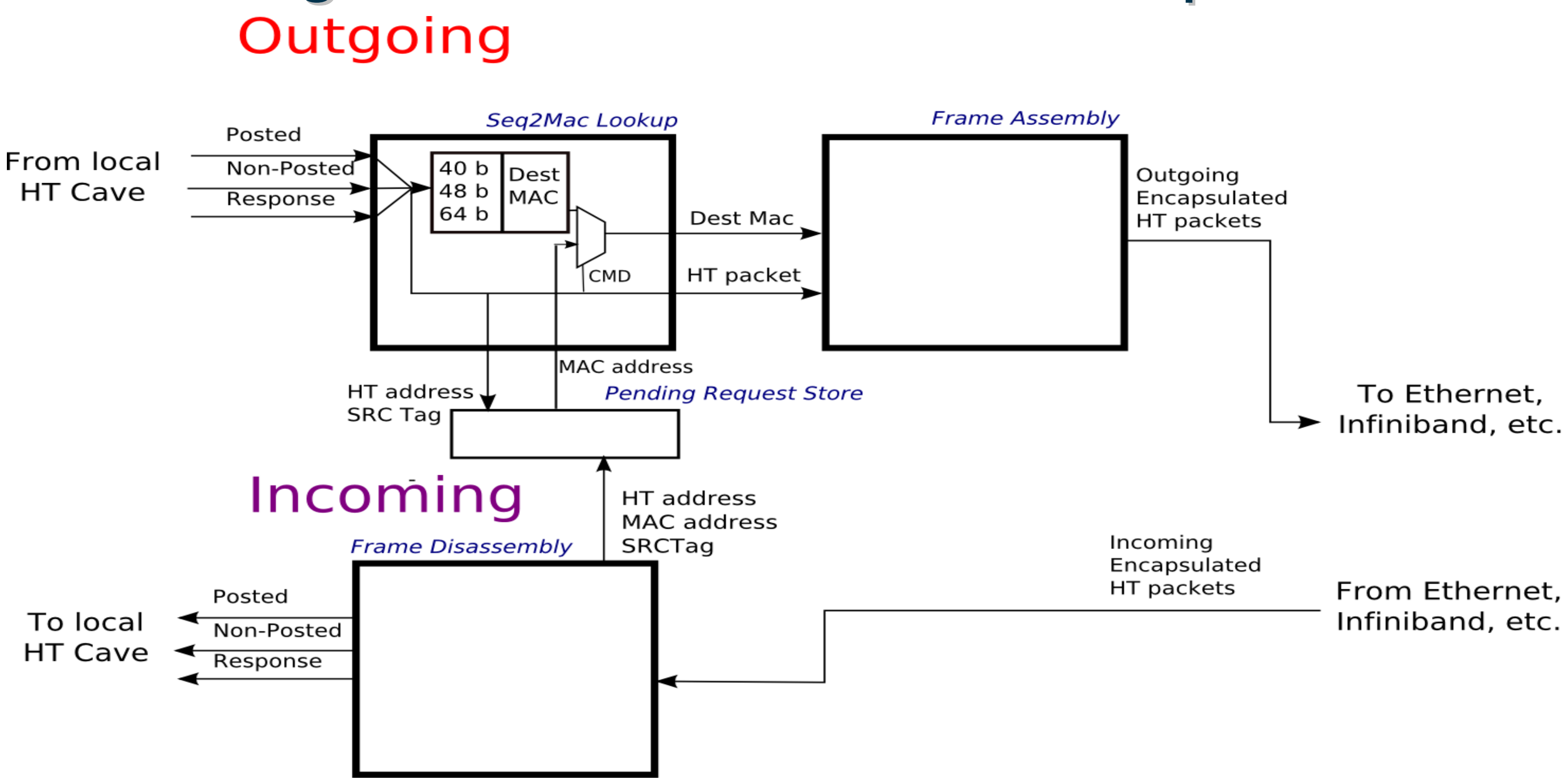
- Address translated into a node address and remote local memory address
- Low latency memory bridge: encapsulation takes 24 – 72 ns in current FPGA implementation
  - To Ethernet
- Bridge ➔ 1300-1500 FPGA slices (Virtex 4 FX140)

HT Bridge – Translation and Encapsulation



- Modular approach and encapsulation allows software to be portable as processor physical address space grows.
  - Extension from the 40-bit to 64-bit physical address
  - Creation of a HyperTransport packet which includes a 64-bit extended address
  - Map the most significant 24 bits of destination address to a 48-bit MAC address and encapsulation into an Ethernet frame.

HT Bridge – Translation and Encapsulation



- Addition of remote_flush transactions to support consistency models
- Limits on the number of outstanding transactions → back pressure
- Handling non-unique message IDs

Evaluation of Cost/Power Efficiency

- Trace based simulation used to generate profiles of page fault behavior vs. memory footprint
- Random workload mix generated from applications
- Static bin-packing algorithm for allocation memory
  - "Spill" model for generating remote allocations
- Analytical models used to evaluate power and cost of DRAM memory and reductions for HP server configurations
  - Cost statistics obtained from HP's server business website [2]
  - Power statistics obtained from HP power calculators [3]

[2] HP Proliant DL servers - cost specifications, http://h18004.www1.hp.com/products/servers/platforms/, 2008

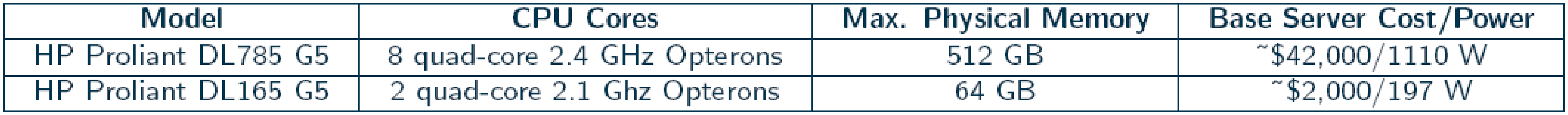
[3] HP Power Calculator Utility, http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00881066/c00881066.pdf, 2008

Experimental Setup – Trace Generation

- Sampled from high frequency and large footprint program regions
- 2.1 billion references were taken from these points using Simics 3.0.31 infrastructure [4]
  - Simics traces included background operating system traffic
  - 100 million references were used to warm the page table simulation
- C++ page table simulation evaluated 7 physical memory sizes (32 MB - 2 GB) for each application

4) Peter S. Magnusson, et al., Simics: A full system simulation platform. Computer, 35(2):50–58, 2002.

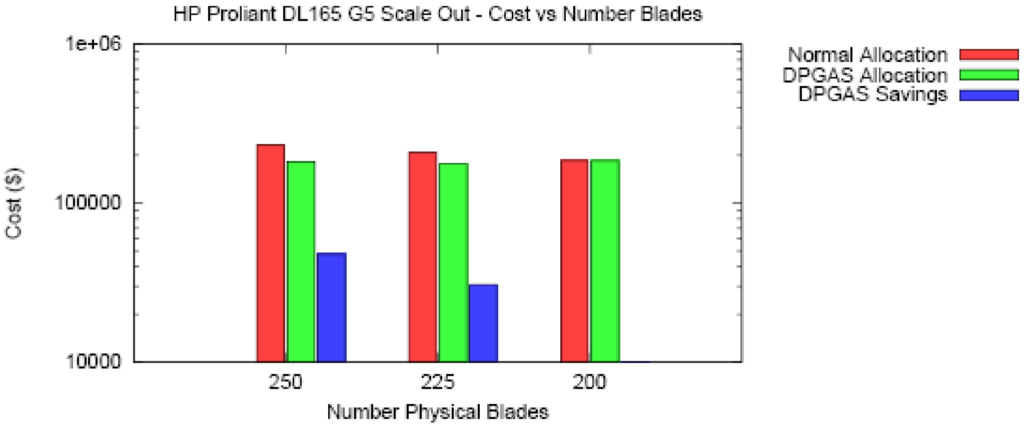
Experimental Setup – Server Configuration

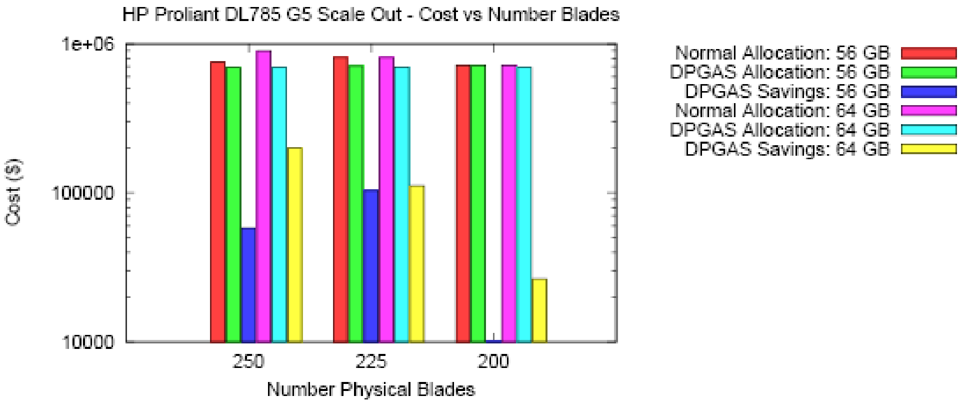


Model	CPU Cores	Max. Physical Memory	Base Server Cost/Power
HP Proliant DL785 G5	8 quad-core 2.4 GHz Opterons	512 GB	˜$42,000/1110 W
HP Proliant DL165 G5	2 quad-core 2.1 Ghz Opterons	64 GB	˜$2,000/197 W

- Two Server configurations: i) high-end (8 CPU server) and ii) low-end (2 CPU server)
  - 64 GB (high-end) or 16 GB (low-end of memory)
- Combinations of each benchmark were used to model independent workloads running in VMs
  - Nearest neighbor (2) spill model for allocation
- Benchmark applications
  - MCF, MILC, and LBM from Spec2006
  - HPCS SSCA graph benchmark
  - DIS Transitive closure benchmark

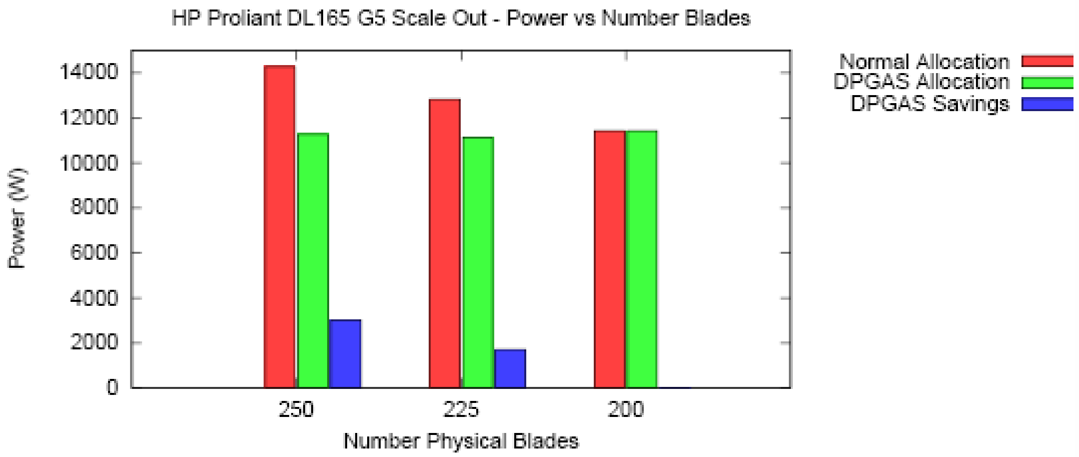
Server Cost – Scale Out

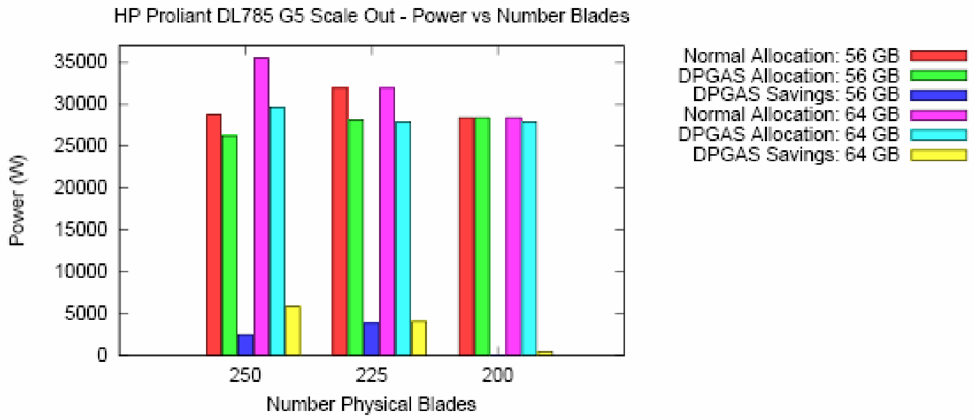




- Scale out tested constant number of workloads consolidated onto varying numbers of servers
- High-end server was given upper bound (64 GB) and lower bound (56 GB) of memory needs
- DPGAS savings in base case - 15% to 26% ($30,000 and $200,000)

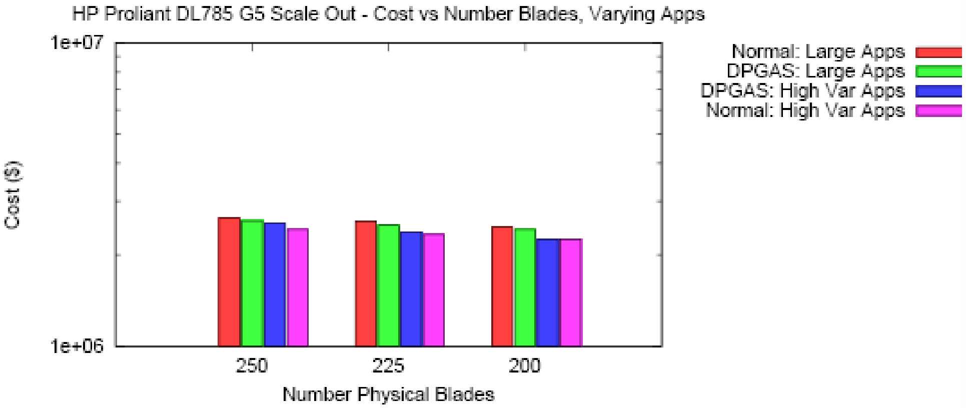
Server Power – Scale Out

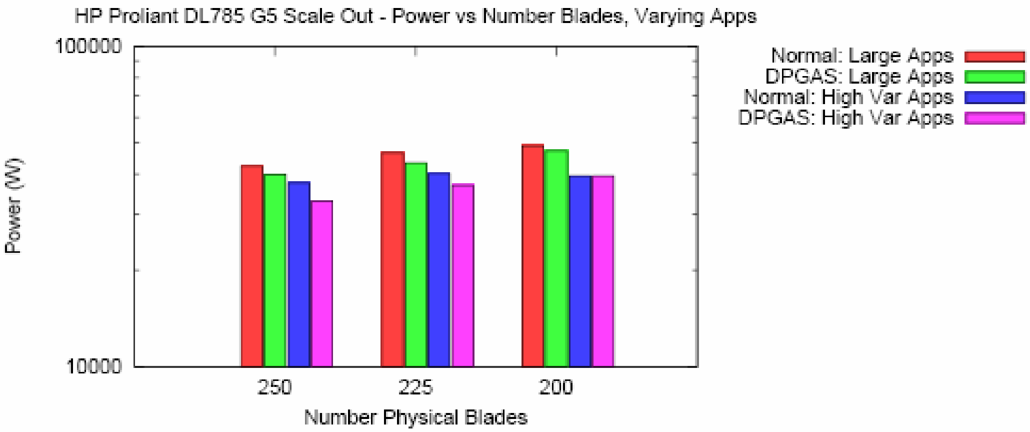

HP Proliant DL165 G5 Scale Out - Power vs Number Blades


HP Proliant DL785 G5 Scale Out - Power vs Number Blades

- DPGAS savings vary from 9% to 25%
  - Translates to a savings of 2,500 Watts to 3,375 Watts
- Low-end server has less fragmented memory, thus smaller savings

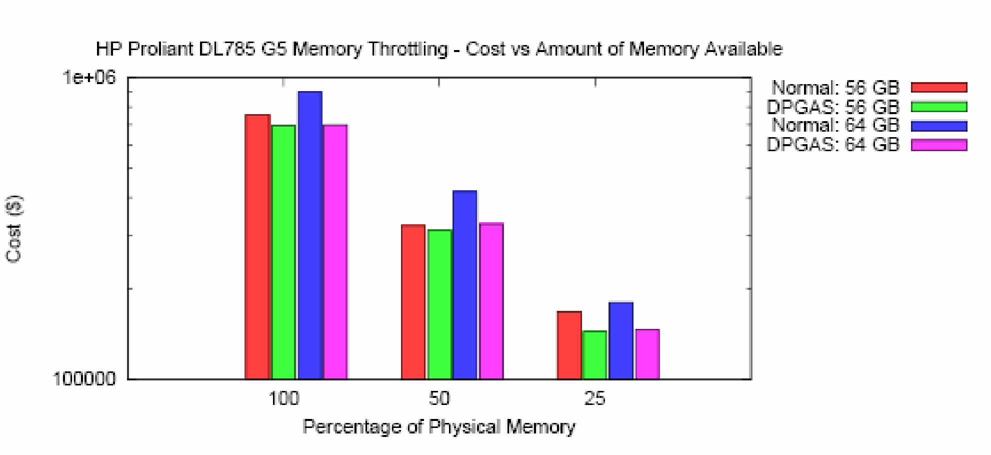
Server Savings – Varying Workloads

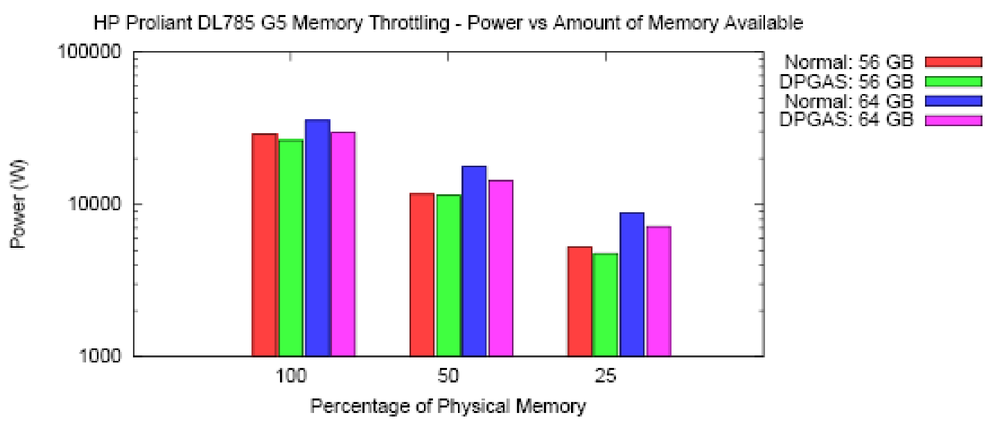




- Workload variations:
  - Only three large applications allocated
  - Allocate one large and one small footprint
- DPGAS savings are worse than normal case
  - 2% to 4% cost savings, 6% to 12% power savings
- Conservative bin packing allocation leads to little memory fragmentation

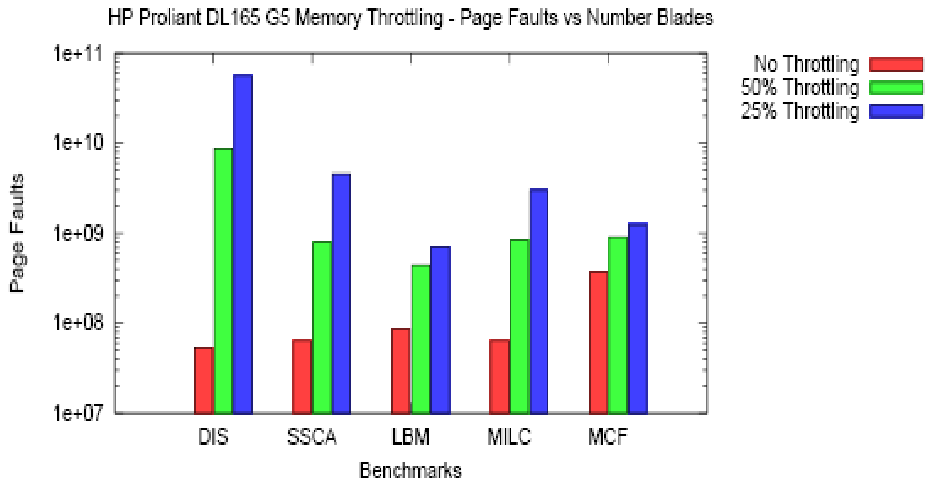
Memory Throttling Power and Cost

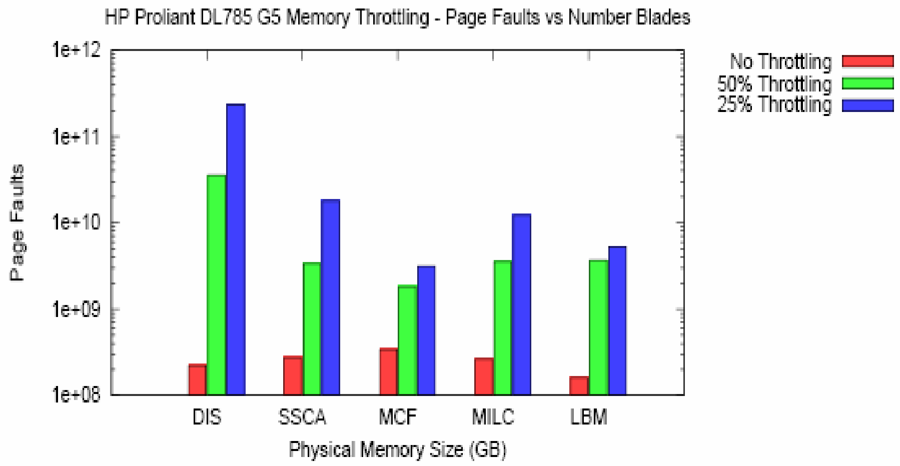




- Memory Throttling: to 50%, 25% of original physical memory footprint
  - Cost and power savings at expense of increased page fault rate
- Reducing memory from 64 GB to 32 GB in high-end server can reduce memory cost by $478,000 and memory power by 17,750 W with DPGAS
  - ~20% savings over normal allocation
- Lower-end savings (56 GB) is 4% to 14% for cost and 2% to 10% for power

Memory Throttling Performance





- Throttling increases page fault rate by order of magnitude
- Some applications suffer more due to small footprints, poor spatial reuse
- Use of throttling with DPGAS may make sense due to possible savings
  - DPGAS can also work similar to third-level cache to improve performance

Some Related Work

- Memory Efficiency
  - HP Labs, Virginia Tech
- PGAS
  - Software approaches - UPC, X10, Titanium, Gasnet
  - RDMA - Liang ’05 is the best example of low-level implementation
- Power and Cost Analysis
  - HP labs – memory blades
  - Feng - Green Destiny

Lessons Learned

- DPGAS savings vary according to server configurations, applications
  - Dependent on time-varying workload characteristics
  - Less memory fragmentation, smaller DPGAS savings
    - Applications with small footprints can be efficiently packed
    - Highly dependent on workload variance

- Memory allocation algorithm matters as well
  - Allocation algorithm searched all servers for space to allocate an application
    - May not be realistic in large server environment
    - Results may favor DPGAS-based sharing in this case

Conclusions

- DPGAS enables opportunities for dynamic sharing of a global physical address space
- HyperTransport provides the basis for low-latency remote accesses
  - Modest connectivity can be valuable
- Power and cost savings vary according to memory fragmentation, application workload, and server configuration
  - DPGAS best suited for high-end servers with application workloads with high variance (large memory fragmentation)