Beyond MPI Microbenchmarks: Beyond point-to-point benchmarks

OFA 2008 International Sonoma Workshop

Tom Elken
Manager, Performance Engineering
QLogic

Motivation

- **Why might we want to measure MPI Performance?**
  - Typically it might be to test a new component in an InfiniBand cluster
    - MPI version or OFED version
    - switch, HCAs or their firmware
    - compute nodes
- **Benchmarks are tools to measure performance**
- **… and quality**

What is the spectrum of MPI benchmarks?

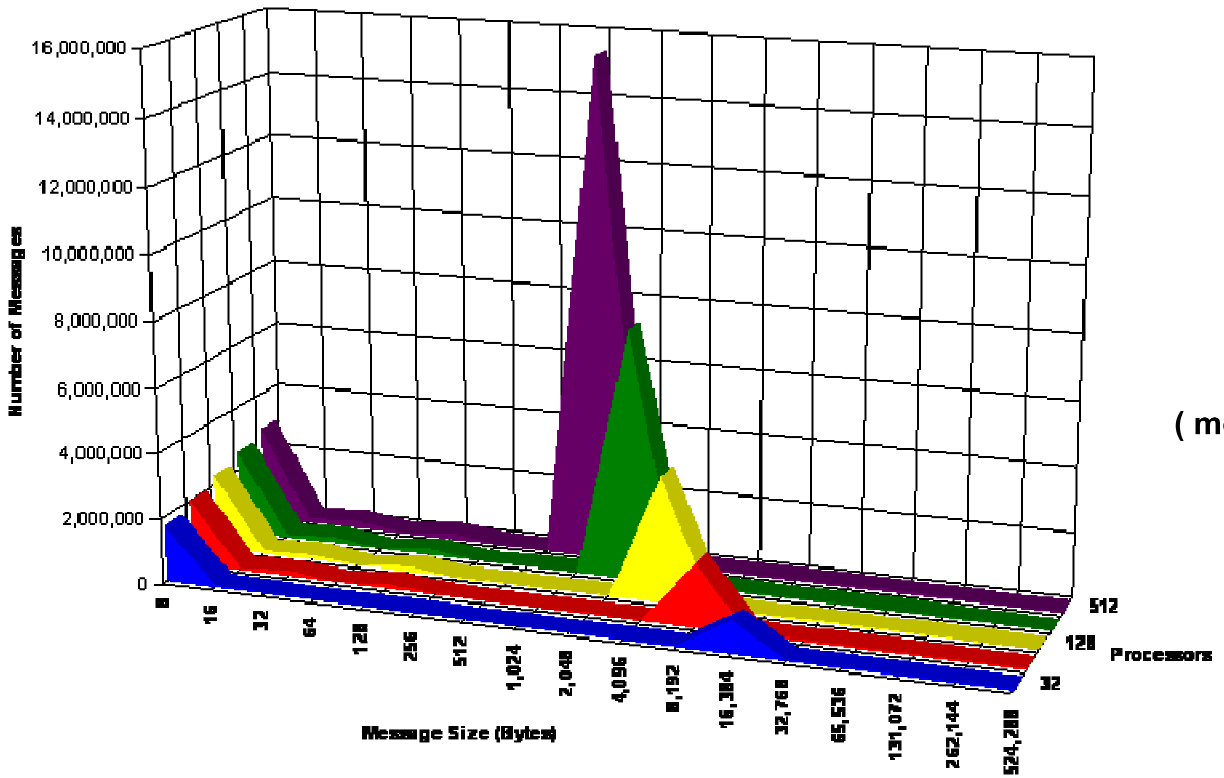
- **Microbenchmarks include (there are more):**
  - OSU MPI Benchmarks (OMB)
  - Intel MPI Benchmarks (IMB) formerly Pallas
- **Mid-level Benchmarks**
  - HPC Challenge
  - Linpack, e.g. HPL
  - NAS Parallel Benchmarks (NPB)
- **Application Benchmark Suites**
  - SPEC MPI2007
  - TI-0n (TI-06, TI-07, TI-08) DOD benchmarks
- **Your MPI application**

Popular MPI Microbenchmarks

- **MPI latency, bandwidth, message rate (point-to-point) tests:**
  - OMB: osu_latency, osu_bw, osu_bibw, osu_mbw_mr, osu_multi_lat
  - IMB (Pallas): PingPong, SendRecv
- **MPI collective tests**
  - IMB tests: AlltoAll, Bcast, Barrier, Reduce, AllReduce, Gather, Scatter, …
  - OMB: Bcast
- **MPI latency and bandwidth benchmarks are very useful IB cluster "health-checkers"**

Relationship of applications to micro-benchmarks

- **As the number of processors is increased:**
  - Message size goes down (→ small-message latency)
  - Number of messages goes up (→ message rate)



DL Poly

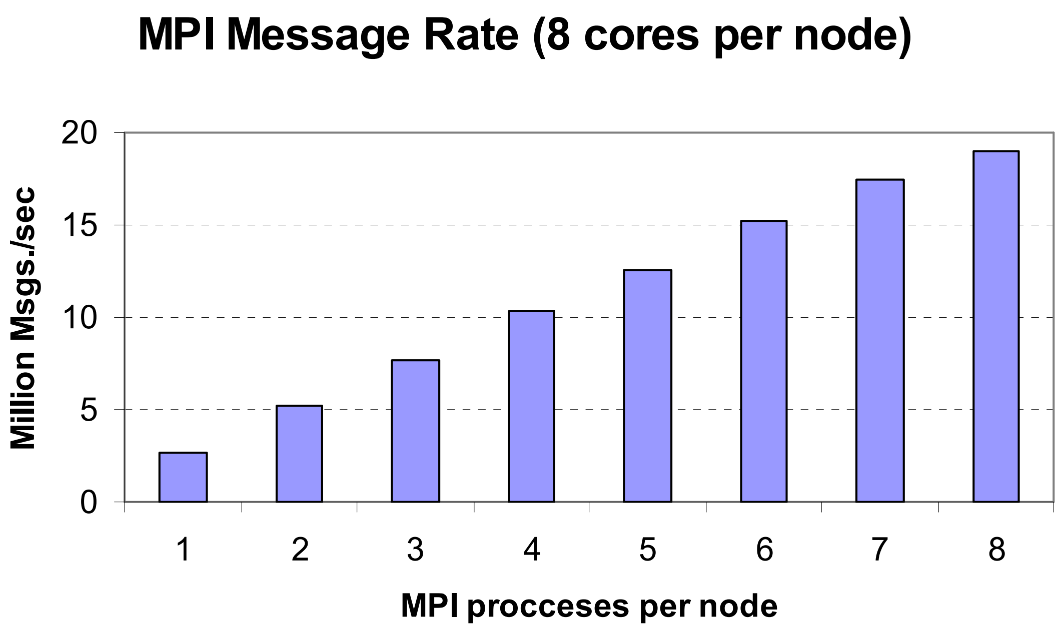
(molecular dynamics)

Small message latency

- **What can you learn from pt-to-pt latency tests**
  - The total time consumed by
    - Software stack
    - transit from CPU core, across memory bus(es), PCIe chipset, HCA, Switch chips, cables
    - If latency is out of expectations, you may be transiting more components than you thought
  - Looking at latency of different small message sizes may be useful for applications that have frequent use of message sizes that are small, but > 8 bytes.
- **Point-to-point tests are special cases. Can be optimized heavily.**

MPI Message Rate

- **Tips for using osu_mbw_mr to measure Message Rate:**
  - measure at several processes per node counts
  - Be careful to get 1st half of MPI processes running on 1st node
  - See if results scale with additional processes per node

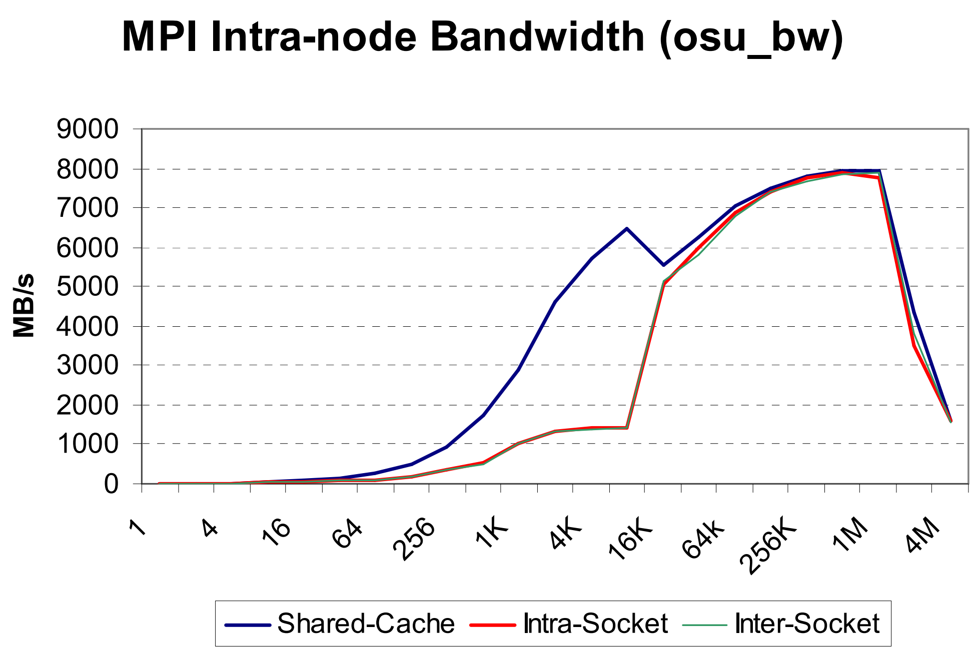


What's new in the OSU MPI Benchmarks?

- **OSU has started to publish results on one node**
  - Intra-node MPI performance importance growing as nodes grow their core-counts
  - Pure MPI applications under some competition from more complex hybrid OpenMP – MPI styles of development
- **OMB v3.1 has added a benchmark: Multiple Latency test (osu_multi_lat.c)**

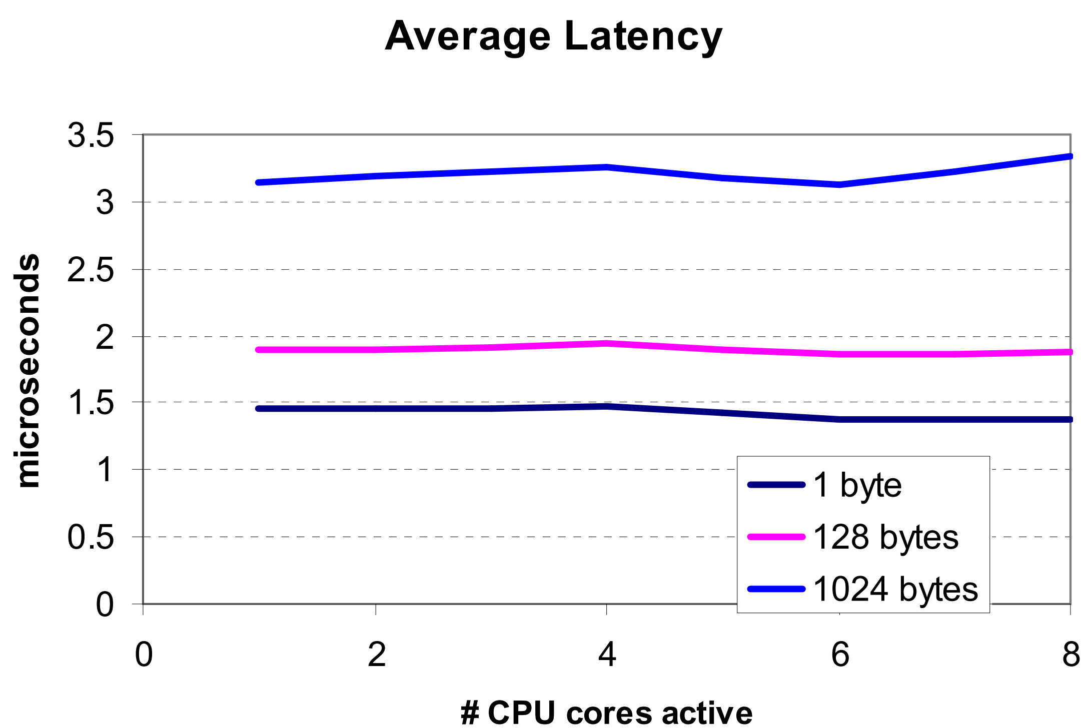
Intra-node MPI Bandwidth measurement

- Most current MPIs use shared-memory copies for intra-node communications – might expect that they all do equally well
- After an improvement in intra-node bandwidth was made, average performance of applications improved 2% (on 8x 4-core nodes)



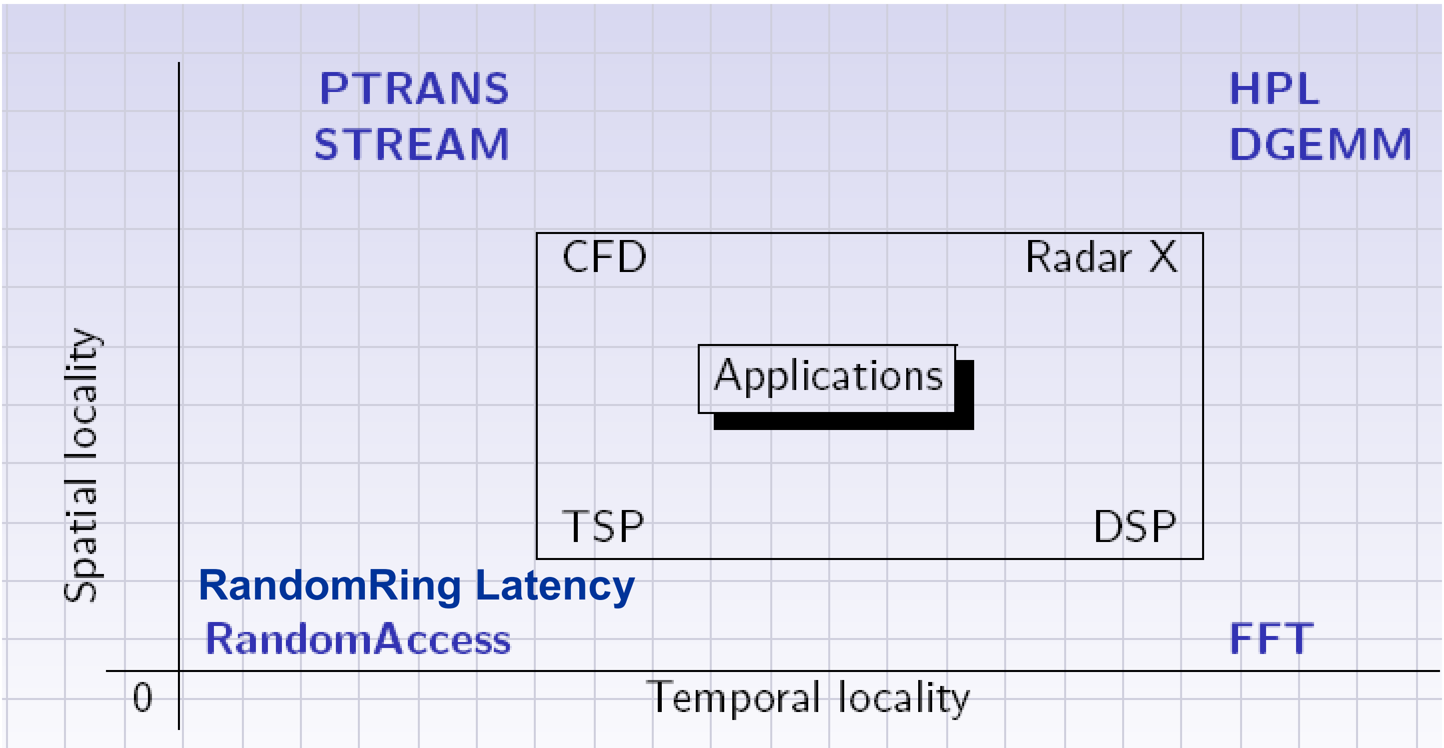
New OSU Multiple Latency Test

- **Measure avg. latency as you add active cores running the latency benchmark in parallel**
- **Interesting to measure on large core-count nodes, and at multiple message sizes …**



HPC Challenge Overview

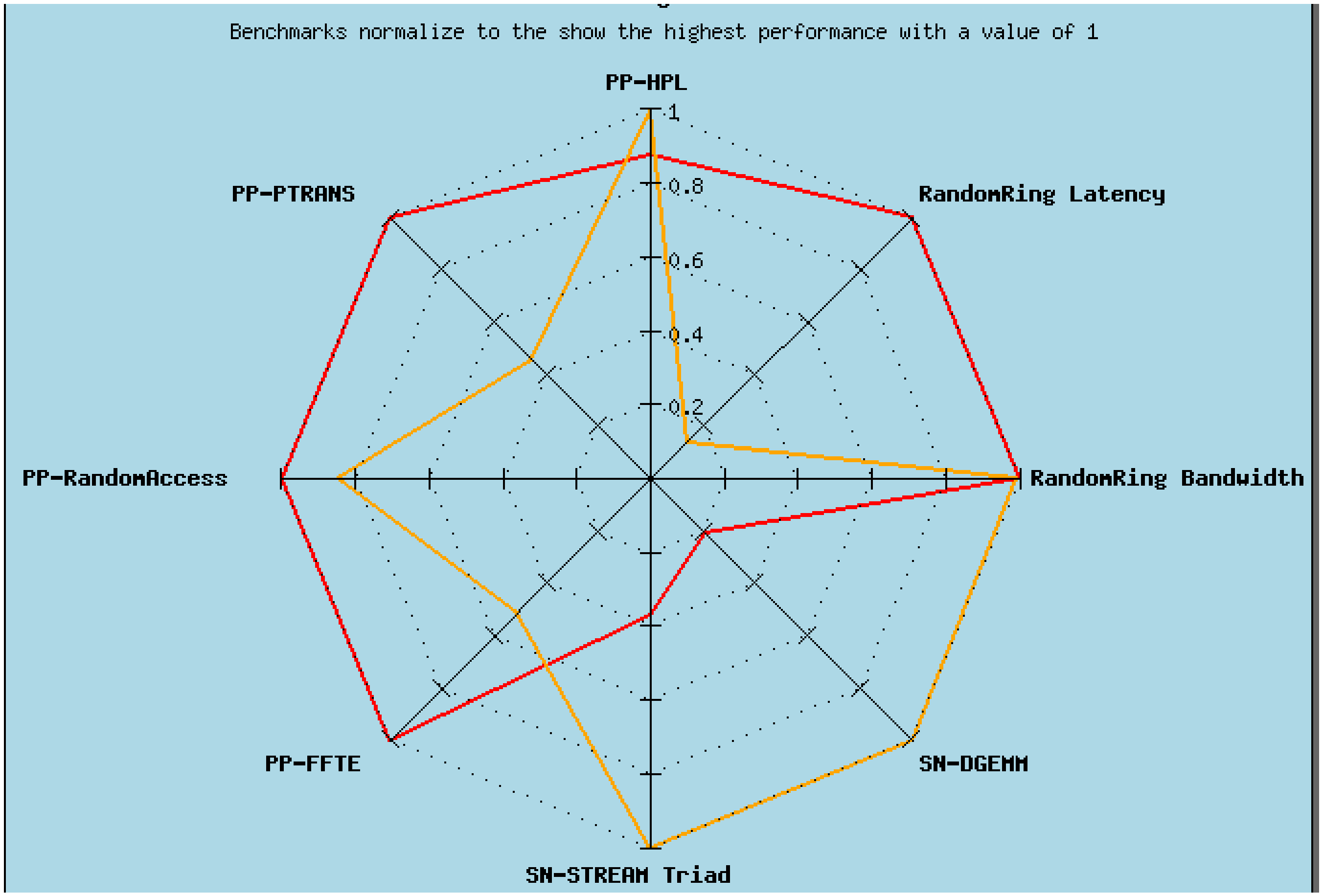
- **HPC Challenge component benchmarks are intended to test very different memory access patterns**



Source: “HPC Challenge Benchmark,” Piotr Luszczek, University of Tennessee Knoxville, SC2004, November 6-12, 2004, Pittsburgh, PA

How to Interpret HPC Challenge results

- **Results at:** http://icl.cs.utk.edu/hpcc/hpcc_results.cgi
- **There is no aggregate metric, but you can compare systems with Kiviat diagrams from** http://icl.cs.utk.edu/hpcc/hpcc_results_kiviat.cgi



Relationship of HPC Challenge to Point-to-Point benchmarks

- **Are there benchmarks in HPCC that focus on latency, bandwidth & message rate but involve more of the cluster than two cores on two nodes?**
  - Latency: Random Ring Latency
  - Bandwidth: PTRANS and
    Random Ring Bandwidth
  - Message Rate: MPI Random Access

SPEC MPI2007

- **An application benchmark suite that measures CPU, memory, interconnect, compiler, MPI, and file system performance.**
- **SPEC institutes discipline and fairness in benchmarking:**
  - Rigorous run rules
  - All use same source code, or performance-neutral alternate sources
  - Disclosure rules: system, adapter, switch, firmware, driver, compiler optimizations, etc.
  - Peer review of submissions before SPEC publication
  - Therefore, more difficult to game

SPEC MPI2007 Benchmarks 1- 6

Benchmark	Language	Application Area	Brief Description
104.milc	**C**	**Quantum Chromodynamics**	A gauge field generating program for lattice gauge theory programs with dynamical quarks
107.leslie3d	**Fortran**	**Computational Fluid Dynamics**	CFD using Large-Eddy Simulations with linear-eddy mixing model in 3D.
113.GemsFDTD	**Fortran**	**Computational Electromagnetics**	Solves the Maxwell equations in 3D using the finite-difference time-domain (FDTD) method
115.fds4	**Fortran**	**CFD: Fire dynamics simulator**	A CFD model of fire-driven fluid flow, with an emphasis on smoke and heat transport from fires
121.pop2	**Fortran/C**	**Climate Modeling**	The Parallel Ocean Program (POP) developed at LANL
122.tachyon	**C**	**Graphics: Ray Tracing**	A nearly E.P. parallel ray tracing program with low MPI usage

SPEC MPI2007 Benchmarks 7- 13

QLOGIC

Benchmark	Language	Application Area	Brief Description
126.lammps	**C++**	**Molecular Dynamics**	**a classical molecular dynamics simulation code designed for parallel computers**
127.wrf2	**C/Fortran**	**Weather Forecasting**	**Code is based on the Weather Research and Forecasting (WRF) Model**
128.GAPgeofem	**C/Fortran**	**Heat Transfer using FEM**	**A parallel finite element method (FEM) code for transient thermal conduction with gap radiation**
129.tera_tf	**Fortran**	**3D Eulerian Hydrodynamics**	**Code uses a 2^{nd} order Gudenov scheme and a 3^{rd} order remapping**
130.socorro	**C/Fortran**	**Molecular Dynamics**	**Molecular Dynamics using density-functional theory (DFT)**
132.zeusmp2	**Fortran**	**Computational Astrophysics**	**Performs various hydrodynamic simulations on 1, 2, and 3D grids**
137.lu	**Fortran**	**Implicit CFD**	**Solves a regular sparse block Lower- and Upper-triangular system using SSOR**

SPEC MPI2007 on the web

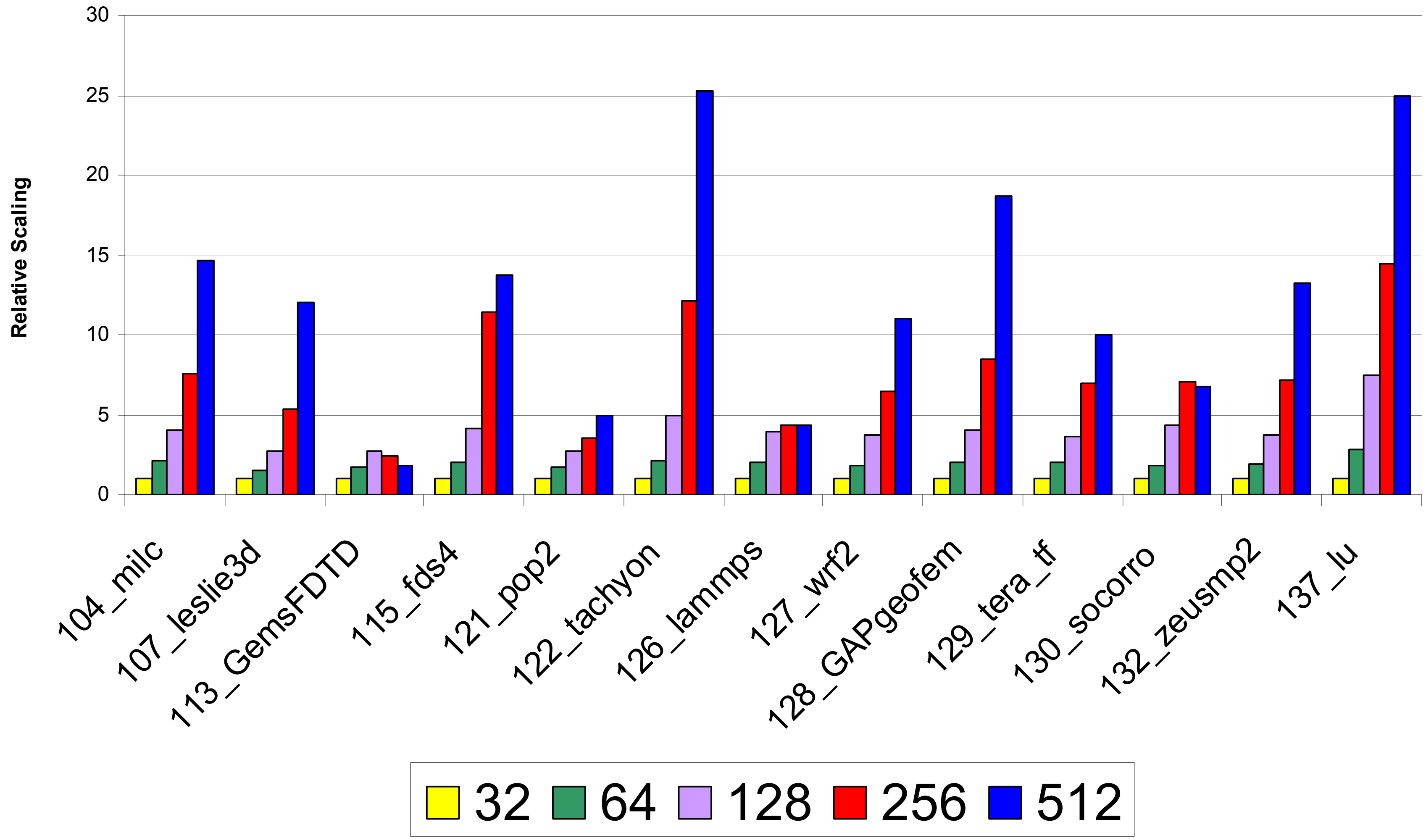
- Result score is an average of ratios for each of 13 codes: the ratio of the run time of a code on your system to the runtime on the reference platform (1st listed).

Test Sponsor	System Name	System Configuration				Results	
		MPI Ranks	Compute Threads Used	Compute Nodes Used	Compute Cores Enabled	Base	Peak
Advanced Micro Devices	A2210 ("Serenade") -- Reference Platform HTML | CSV | Text | PDF | PS | Config	16	16	8	16	0.999	0.999
Hewlett-Packard Company	HP Proliant BL460c blade Cluster Platform 3000BL HTML | CSV | Text | PDF | PS | Config	128	128	32	128	11.9	Not Run
Hewlett-Packard Company	HP Proliant BL460c blade Cluster Platform 3000BL HTML | CSV | Text | PDF | PS | Config	256	256	64	256	19.8	Not Run
Hewlett-Packard Company	HP Proliant BL460c blade Cluster Platform 3000BL HTML | CSV | Text | PDF | PS | Config	64	64	16	64	6.39	Not Run
Hewlett-Packard Company	HP Proliant BL460c blade Cluster Platform 3000BL HTML | CSV | Text | PDF | PS | Config	32	32	8	32	3.40	Not Run
Hewlett-Packard Company	HP Proliant BL460c blade Cluster Platform 3000BL HTML | CSV | Text | PDF | PS | Config	16	16	4	16	1.75	Not Run
Intel Corporation	Endeavor HTML | CSV | Text | PDF | PS | Config	256	256	32	256	18.5	Not Run
Intel Corporation	Endeavor HTML | CSV | Text | PDF | PS | Config	32	32	4	32	3.05	Not Run
Intel Corporation	Endeavor HTML | CSV | Text | PDF | PS | Config	64	64	8	64	6.21	Not Run
Intel Corporation	Endeavor HTML | CSV | Text | PDF | PS | Config	128	128	16	128	11.6	Not Run

Scaling with SPEC MPI2007

QLOGIC

Scaling by application to 512 Cores



MPI profiles of MPI2007

QLOGIC

- **Profiles of MPI function usage in the 13 applications are quite varied; implies usefulness as a QA test**

- **Profiles of interest:**
  - 121.pop2 (POP) has largest message rate: 128K / sec / core on average → need for message rate
  - 130.socorro sends the most data per second: 65 MB / sec / core→ need for bandwidth
  - 107.leslie3D sends largest messages, up to 283 MB → need for bandwidth
  - 128.GAPgeofem has small avg. message size (609 bytes) and 2nd highest message rate: 26K / sec / core → need for latency and message rate

In Summary

- **The best benchmark is "your application"**
- **There is a range of MPI benchmarks because they all have their place:**
  - microbenchmarks are easier, quicker to run and may focus on a component of the system you are interested in
  - application benchmarks are a bit more difficult to run, but are a better predictor of performance across a range of applications
- **Benchmarks are evolving to serve the needs of ever-expanding multi-core systems**