東京大学情報基盤センター
INFORMATION TECHNOLOGY CENTER, THE UNIVERSITY OF TOKYO

JCAHPC

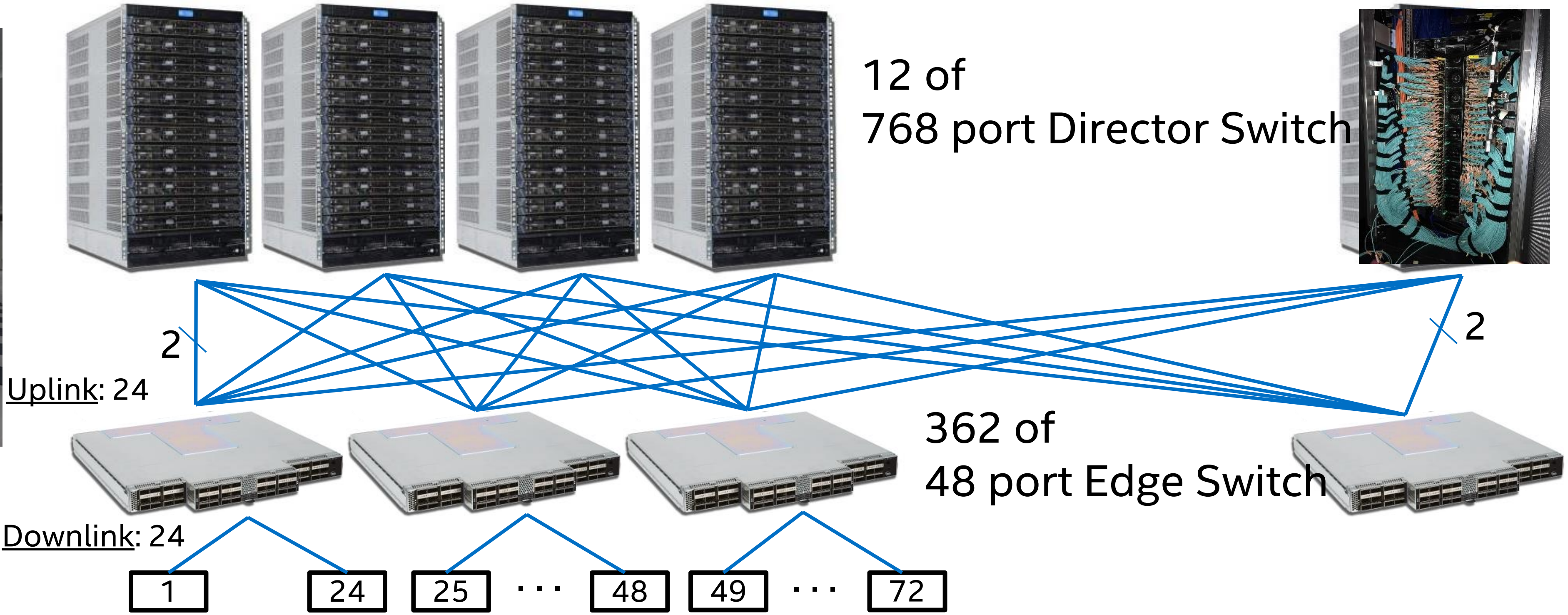
筑波大学
計算科学研究センター
Center for Computational Sciences

SCALING COLLECTIVES ON LARGE CLUSTERS USING INTEL(R) ARCHITECTURE PROCESSORS AND FABRIC

Masashi Horikoshi, Larry Meadows, Thomas Elken, Pradeep Sivakumar, Edward Mascarenhas, James Erwin, Dmitry Durnov, Alexander Sannikov, Alexey Malhanov (Intel), Toshihiro Hanawa (The University of Tokyo) and Taisuke Boku (University of Tsukuba)

January 31, 2018 (IXPUG Workshop at HPC Asia 2018)

System: One of World largest Intel® Xeon Phi™ + Intel® Omni-Path Architecture (Intel® OPA) system => OakForest-PACS



12 of
768 port Director Switch

Uplink: 24

2

2

362 of
48 port Edge Switch

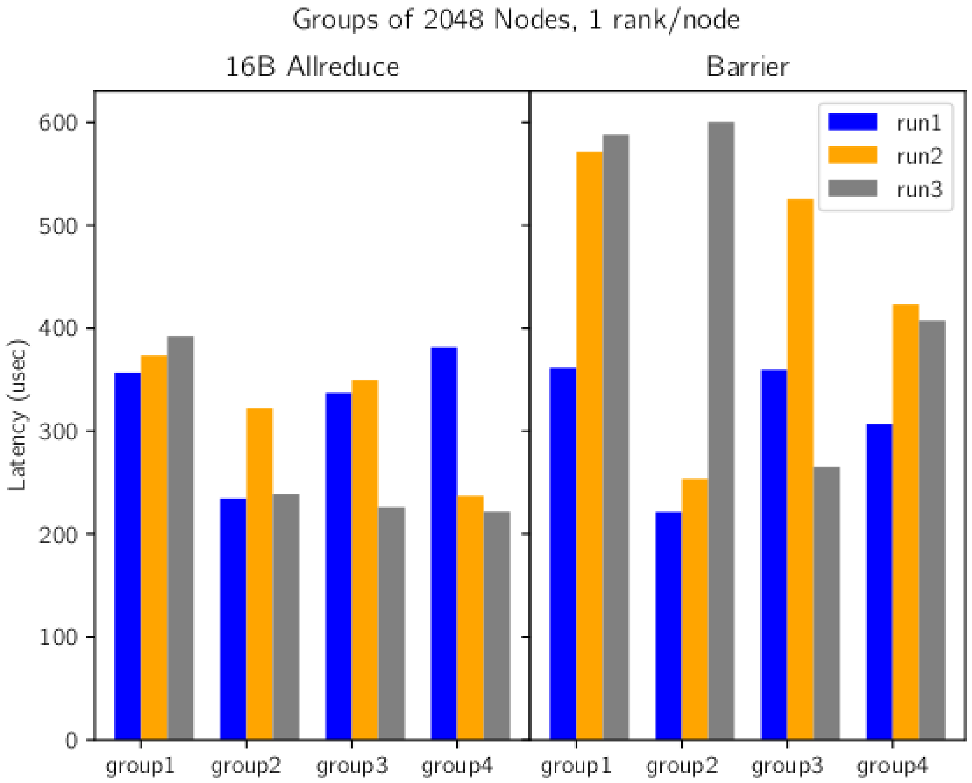
Downlink: 24

8208 node of Intel Xeon Phi (KNL) 7250 (68c, 1.4GHz) with full bi-section BW fat tree and 26PB Lustre by single rail Intel OPA interconnect. CentOS 7.2 on compute node.

25PFLOPS peak and #6 in Top 500 at launch

*Detail configuration in backup slides

Initial results: Run-to-run variability



Intel MPI Benchmark (IMB) Barrier and 16 Byte Allreduce. 1 rank per node.

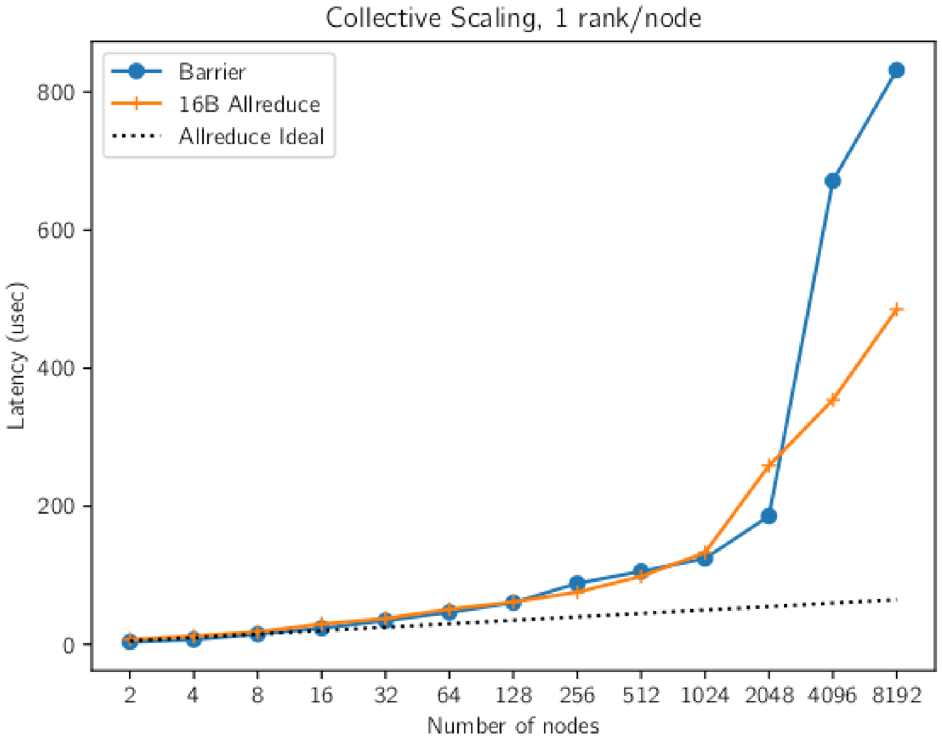
4 groups of 2048 node

Almost same latency in each group would be expected but…

Group to group variance would be less significant than inside group

Wide variances due to OS noise? (hypothesize)

Initial results: performance



IMB barrier and 16B allreduce results. 1 process per node.

Ideal $\log_2(N)$, N: rank

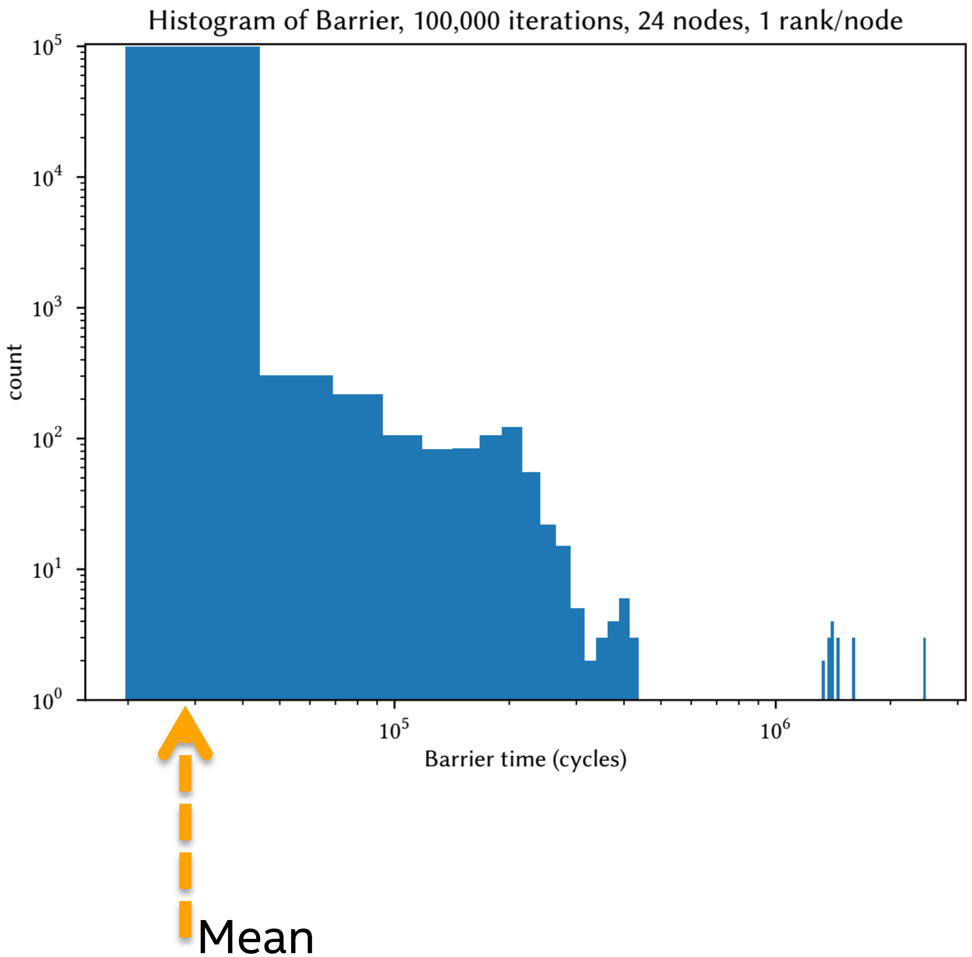
OS noise increases non-linearly

No good explanation for worse barrier scaling than allreduce at high count

Initial investigation

// Recording time for each iteration on each rank

```
MPI_Barrier(MPI_COMM_WORLD);
tscs[0] = _rdtsc();
for (int i = 0; i < ntimes; ++i) {
  MPI_Barrier(MPI_COMM_WORLD);
  tscs[i+1] = _rdtsc();
}
report (rank, nranks, ntimes, tscs,
benchmark, 0);
```



Mean

- Extreme excursions from mean are due to OS noise
- Kernel trace verified correlation with this

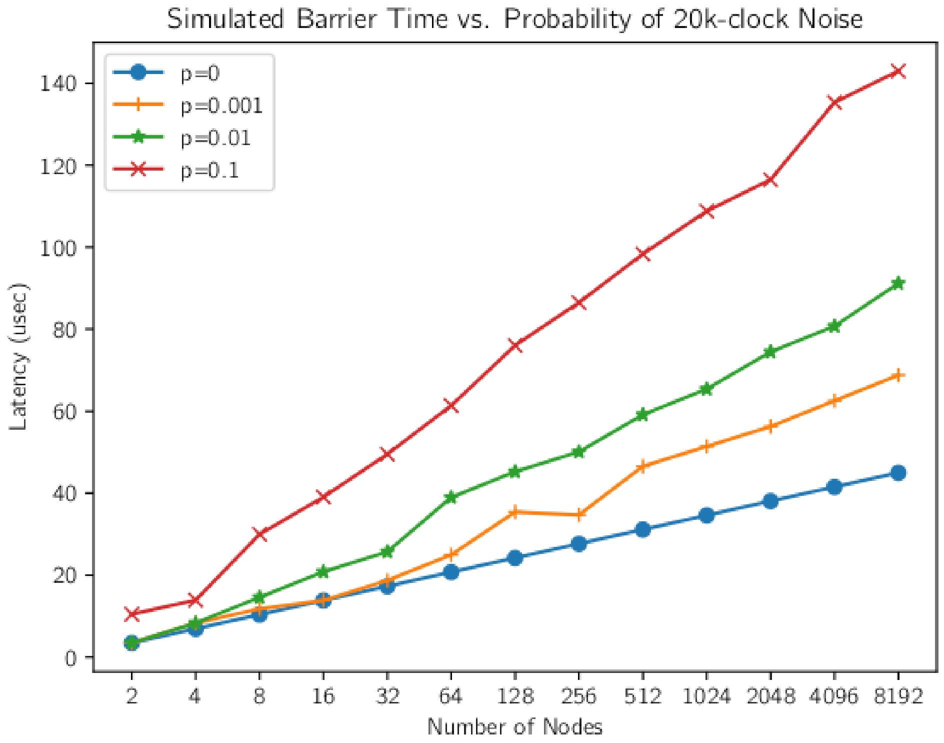
Cause of variance

ps, top, Vtune, kernel ftrace analysis used to find 3 major sources of variance

- **Frequency transition (Turbo):** 1.4GHz <-> 1.5GHz <-> 1.6GHz Transition stalls many microseconds.

- **Periodic MWAIT wake-up:** Linux system default is using idle=mwait. MONITOR and MWAIT instructions on idle hardware threads. KNL forces a periodic wake-up of hardware threads in an MWAIT state 10 times per second and additionally cause frequency transitions on the entire processor .

- **OS work:** Daemons, hardware interrupts, middleware (system monitoring, scheduling). idle thread on the same core or tile is awakened to perform OS work, the application thread will be delayed and additionally cause frequency transitions.



Simulated barrier results by recursive doubling.
Theoretical: log2(# of node) * Latency.
20K cycle injected with probability p at each step.
L=3.5usec, 20K cycle=14usec.

Remedies

Impact of effect

idel=halt: Stopping MONIOTR/MWAIT and single-tile turbo (No 1.6GHz)

Tickless mode (nohz_full=2-67,70-135,138-203,206-271): Decreasing OS timer interrupt from 1KHz to 1Hz except tile-0. And excluding tile-0 from application.

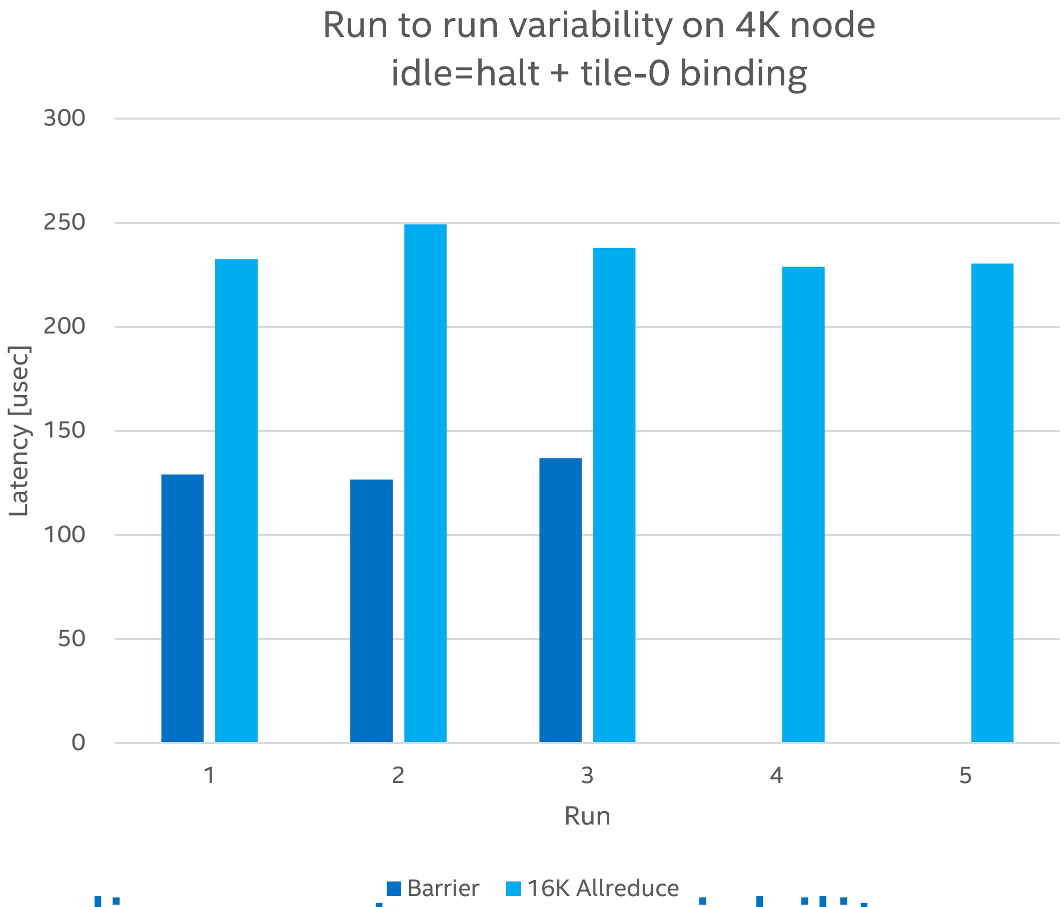
Binding Lustre daemon and system process to tile-0

Using acpi-cpufreq driver rather than intel_pstate

Tuning spinning: `PSM2_YIELD_SPIN_COUNT=10000` and `I_MPI_COLL_SHM_PROGRESS_SPIN_COUNT=100000`

* These remedies have cons side effects (effect depends on situation and application).

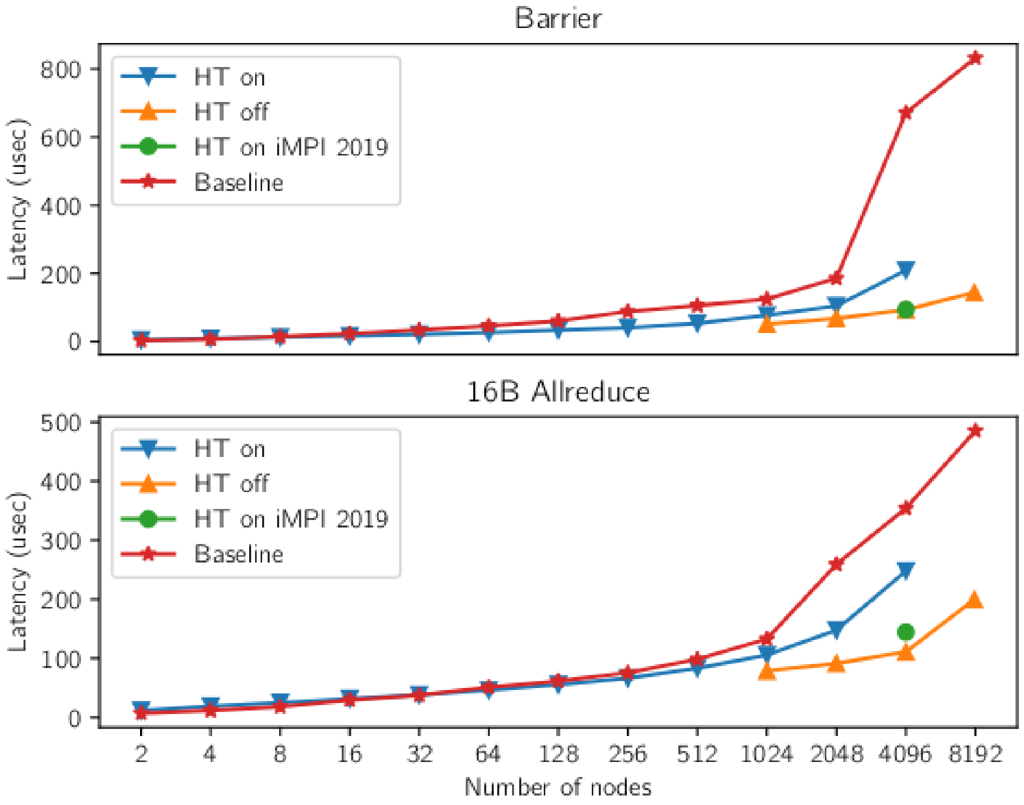
Run to run variability improvement on 4096 node



Applying remedies, run to run variability was largely improved

+-4% from median now

Performance Results



4K node collective	Target [usec]	Baseline [usec]	Optimized [usec]
Barrier	105	671	94
16B Allreduce	160	485	145

HT ON result with iMPI 2017U3 improved hugely (7.1x and 3.3x) vs. baseline.

HT OFF with iMPI 2017U3 better than HT ON.

8K node allreduce has still some noise even with HT OFF.

By MPI library tuning (reduced # of inst), HT ON with iMPI2019TP matched HT OFF (iMPI2017U3) result.

Future work: HT OFF with iMPI2019 and multi process per node

Conclusion and Call to Action

System and MPI library optimizations on large scale KNL+OPA cluster achieved 7.1x and 3.3x improvement for IMB barrier and 16B allreduce on 4K node.

Call to action:

- Read carefully “latest” Intel® Omni-Path Fabric Performance Tuning User Guide (now Rev. 10.0, Oct. 2017)
- Provide suggestions to masashi.horikoshi@intel.com and lawrence.f.meadows@intel.com

Legal Disclaimers

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer. No computer system can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.
Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Intel, the Intel logo, Atom, Xeon, Xeon Phi, 3D Xpoint, Iris Pro and others are trademarks of Intel Corporation in the U.S. and/or other countries. *Other names and brands may be claimed as the property of others.

Legal Disclaimers

Optimization Notice

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

[Pre-Patch Disclaimer]   Benchmark results were obtained prior to implementation of recent software patches and firmware updates intended to address exploits referred to as "Spectre" and "Meltdown". Implementation of these updates may make these results inapplicable to your device or system.

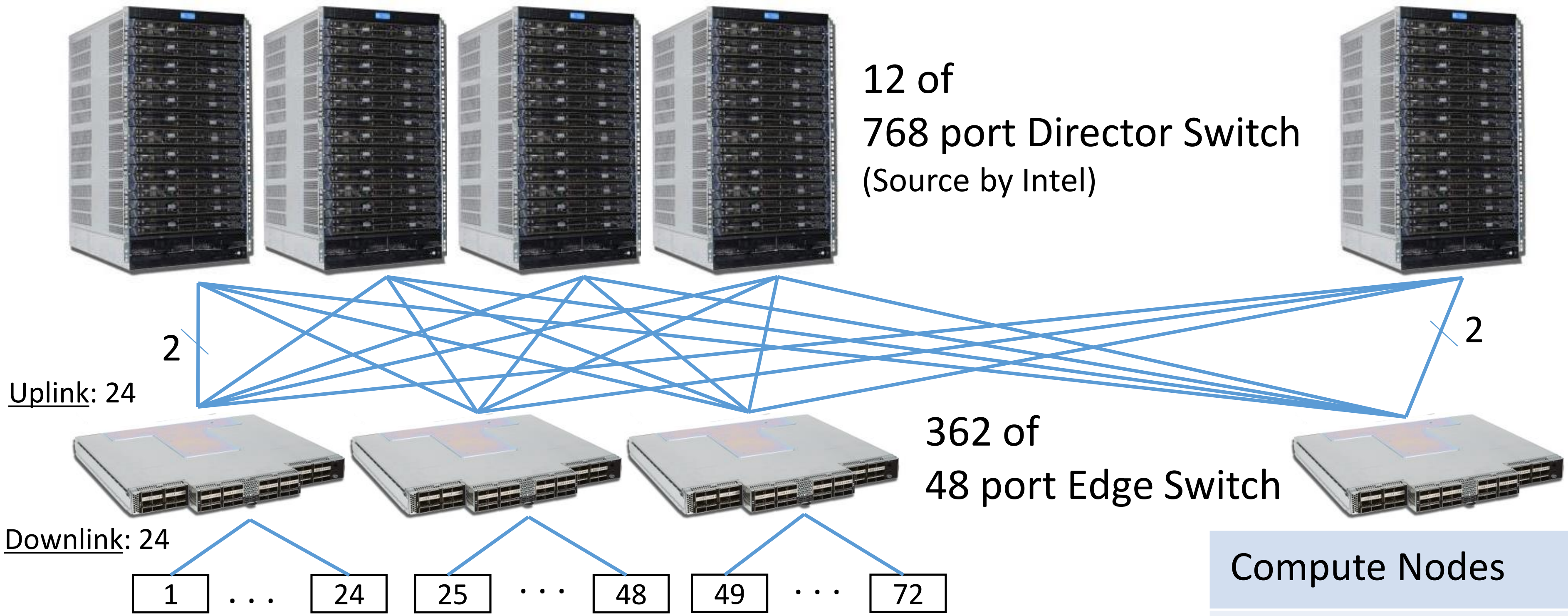
intel®

Specification of Oakforest-PACS system

Total peak performance			25 PFLOPS
Total number of compute nodes			8,208
Compute node	Product		Fujitsu PRIMERGY CX600 M1 (2U) + CX1640 M1 x 8node
	Processor		Intel® Xeon Phi™ 7250 (Code name: Knights Landing), 68 cores, 1.4 GHz
	Memory	High BW	16 GB, 490 GB/sec (MCDRAM, effective rate)
		Low BW	96 GB, 115.2 GB/sec (peak rate)
Interconnect	Product		Intel® Omni-Path Architecture
	Link speed		100 Gbps
	Topology		Fat-tree with (completely) full-bisection bandwidth

Slide courtesy of Prof. Hanawa and Prof. Boku

Full bisection bandwidth Fat-tree by Intel® Omni-Path Architecture



Firstly, to reduce switches&cables, we considered :

- All the nodes into subgroups are connected with FBB Fat-tree
- Subgroups are connected with each other with >20% of FBB

But, HW quantity is not so different from globally FBB, and globally FBB is preferred for flexible job management.

Compute Nodes	8208
Login Nodes	20
Parallel FS	64
IME	300
Mgmt, etc.	8
Total	**8600**

Slide courtesy of Prof. Hanawa and Prof. Boku

Specification of Oakforest-PACS system (Cont’d)

Parallel File System	Type	Lustre File System
	Total Capacity	26.2 PB
	Product	DataDirect Networks ES14K
	Aggregate BW	500 GB/sec (50 GB/sec x 10 OSS)
	Metadata	MDS x 12, MDT x 3, 3 DNE (Distributed Namespace)
File Cache System	Type	Burst Buffer, Infinite Memory Engine (by DDN)
	Total capacity	940 TB (NVMe SSD, including parity data by erasure coding)
	Product	DataDirect Networks IME14K
	Aggregate BW	1,560 GB/sec (with 25 x2 IME servers)
Power consumption		4.2 MW (including cooling)
# of racks		102

Slide courtesy of Prof. Hanawa and Prof. Boku

Acknowledgements

Part of the computational resource of the Oakforest-PACS was awarded by the "Large-scale HPC Challenge" Project, JCAHPC (Joint Center for Advanced High Performance Computing).

We thank the faculty and staff of JCAHPC, as well as the engineers at Fujitsu (Computational Science and Engineering Solution DIV., Technical Computing Solutions Unit) and Intel, particularly Professor Kengo Nakajima (The University of Tokyo), Yoshio Sakaguchi (Fujitsu), Kohta Nakashima (Fujitsu Laboratories), Alexey Malhanov (Intel) and John Pennycook (Intel).