SOLUTION BRIEF

HPC Converged Platform

Supporting Simulation and Modeling, AI, and Analytics on a Common Platform

Many organizations are expanding their definitions of high-performance computing (HPC) to include workloads such as artificial intelligence (AI) and analytics, in addition to simulation and modeling. Intel is working on a number of fronts to facilitate that transition, such as contributing optimizations for Intel® architecture to open source projects, co-engineering with other members of the ecosystem, and developing reference architecture solutions. This solution brief introduces capabilities for running TensorFlow* and Apache Spark* on existing HPC systems using standard batch schedulers.

HPC clusters are an ideal platform to provide large-scale compute resources to demanding workloads in areas such as AI (including machine and deep learning) and high-performance data analytics. All these workloads have similar demands, in terms of running best on robust compute cores connected with high-performance network fabrics, and to high-performance shared storage. Enterprises and academic institutions must support a growing number of increasingly complex AI and analytics workloads, which is generating demand to run those workloads using HPC infrastructure.

Several challenges exist in bringing multiple types of workloads together onto a single cluster infrastructure, in the areas of resource management, network usage, and storage usage. Fundamentally, these challenges arise from the fact that resource managers for each type of workload are not natively designed to interoperate with each other.

The requirement to support simulation and modeling, AI, and analytics workloads on a common platform is driven by a need to maximize the value of existing resources. In particular, unifying these infrastructures provides an alternative to running multiple clusters, as shown in Figure 1, which otherwise adds significantly to the capital expenditure (CAPEX) associated with purchasing and deploying the systems in the first place, as well as the operating expenditure (OPEX) to keep them running, including moving data among them. In addition, a unified cluster architecture optimizes the ability to manage multiple workflows holistically.

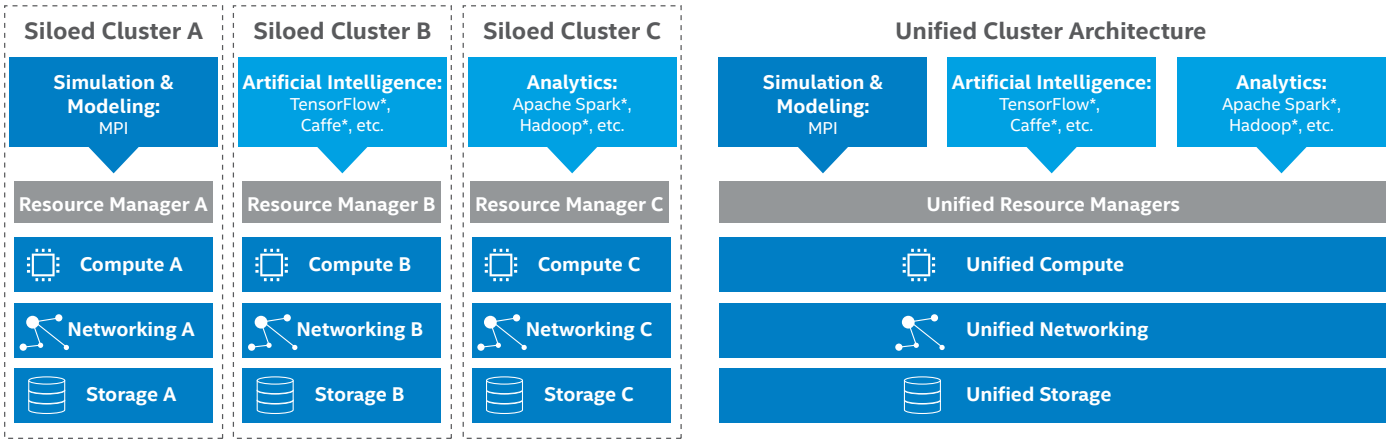

Figure 1. Siloed versus unified cluster architecture for simulation and modeling, AI, and analytics workloads.

Overall resource utilization tends to decrease in scenarios where resources are not pooled together, because of usage fluctuations in the separate domains. Most organizations that need infrastructure for deep learning networks in usages such as computer vision, for example, don't run these workloads on a 24x7 basis. The part-time nature of these workloads means that the special-purpose infrastructure often stands idle and may require rarefied skills to support, both of which can be costly to the organization. Rather than building dedicated environments for such operations using expensive, special-purpose hardware, it can be far more efficient to run these workloads using shared clusters built from general-purpose servers based on Intel architecture.

Multiple clusters within a single organization are often isolated from each other because of different infrastructure, support models, and software across the organization. Separate silos may be created for simulation and modeling, AI, and big data analytics. This creates inefficiencies, complicates efforts to bring new products and services to market, and drives up costs. Unifying resources in a converged environment is therefore a key contributor to driving up infrastructure and solution value. In particular, such unification can help reduce the costs associated with storage duplication and data movement, as compared to the siloed approach.

It is increasingly common for companies and institutions to run simulation and modeling, AI, and analytics workloads on converged clusters. Intel is developing capabilities and solution architectures that help streamline their efforts to do so, while being non-disruptive to production applications and workloads. This brief introduces recent work by Intel to simplify the path to convergence using optimized, open source and commercial solutions, with a multiphased approach to be enabled through industry partners. Topics that will be covered include storage abstraction, software optimization, and common resource managers.

Reducing Data Movement with Storage Abstraction

The complexity associated with multi-cluster environments is particularly acute in cases where different operations in a larger process must each be performed on different clusters. The costs associated with moving and staging data among these environments can be significant or even prohibitive. An alternative is to abstract the storage layer using data virtualization to create a shared data-access layer.

Data transfer burdens between multi-cluster environments have an increasing impact when the uses overlap or integrate. Examples of integrated uses include:

- **Discover**: Run the results of a simulation or model through a neural network to uncover additional patterns or insights.
- **Adjust**: Identify best inputs for the simulation or model by preprocessing data through a neural network.
- **Replace**: Replace key parts of an HPC or high-performance data analytics (HPDA) workflow with neural networks in order to reach a decision faster.
- **Reveal**: Use a simulation to better understand the nature of the decisions made inside a neural network.
- **Expand**: Simultaneously simulate and model data from both image and language formats from the deep learning model.

To address this need, the two solutions described in this paper each use Alluxio* for storage abstraction. Alluxio is open-source software originally developed at UC Berkeley that creates a single point of access to data from multiple object and file stores on various types of persistent storage. Using this approach, applications access data in place, without complex, time-consuming configuration requirements. Eliminating the need to move or duplicate data around the organization creates significant performance and efficiency gains.

The latest Intel® technology innovations for convergence of AI and HPC

Intel is dedicated to supporting increasingly complex workloads through ongoing technology innovations.

2nd Gen Intel® Xeon® Scalable processors have been designed to enable more robust computing clusters that drive results across an array of workloads, including:

- **Advanced HPC performance:** In recent testing, a platform based on 2nd Gen Intel Xeon Scalable processors delivered up to 3.7x average performance improvement in HPC CPU benchmarks compared to a three-year-old system.[1] It also achieved a world-class 5.8x performance improvement on LINPACK* CPU benchmarks compared to competing processors.[2]
- **Improved AI inference:** Testing shows that 2nd Gen Intel Xeon Scalable processors running Intel® Deep Learning Boost can improve inference performance on image classification by 25x compared to competing processors.[3]

Moreover, organizations can improve workload performance with Intel® Optane™ DC persistent memory. This innovative memory technology allows organizations to move and maintain larger amounts of data closer to the processor, meaning data-intensive HPC and AI workloads can be processed quickly and on a large scale.

Alluxio presents any collection of disparate data stores to applications as a single, coherent virtual data source that can be accessed using a single standard interface. This enables users to forge any-to-any connections between applications and data, without regard to where those resources are physically located or how they are formatted.

From a software development perspective, it is not necessary to write separate pieces of code that can access all of the multiple data stores involved. Instead, applications interface directly with Alluxio, which transparently handles the specific data-access requirements of each data store. In addition, this approach to data access enables changes from one persistent store to another to be done without application changes.

Contention for caching resources is a common challenge in distributed applications, which can degrade performance by causing applications to have to access data from disk rather than from cache. Alluxio provides distributed shared caching to help overcome these performance deficits by enabling applications to cache frequently accessed data from the shared access layer to system memory.

Because the data is held close to the processor and is accessible at memory speeds, this caching service can substantially accelerate data access, increasing overall workload throughput. Storage abstraction using Alluxio can also help organizations transition to a smaller set of common data stores that serve the converged platform.

Accelerating Machine Learning with Framework Optimizations

Intel makes open-source contributions and works elsewhere across the ecosystem to help optimize popular deep learning frameworks, including TensorFlow, MXNet*, and Caffe*, among others. Common applications for AI use cases include image recognition, language translation, recommender engines, generative design, and generative adversarial networks (AI algorithms designed for unsupervised machine learning). A combination of these uses is increasingly prevalent as users are looking for data insights for fraud detection, seismic correlation, safety, security, and predictive maintenance.

Key tuning efforts for these frameworks include implementation of the Intel® Math Kernel Library for Deep Neural Networks (Intel® MKL-DNN), which provides highly tuned math functions for resource-intensive operations associated with both training and inference on Intel architecture. In particular, these efforts contribute to these AI frameworks' abilities to scale out effectively across hardware, which helps optimize their suitability for use on HPC infrastructure.

These and other optimizations made by Intel to deep learning frameworks are similar to the types of optimizations that are common in the optimizing of performance-sensitive simulation and modeling applications. The key focus in both cases is to enable code to make better use of hardware resources, including improvements in the following areas:

- **Vectorization**. Taking advantage of single instruction multiple data (SIMD) technology with Intel® Advanced Vector Instructions 512 (Intel® AVX-512) allows functions for operations such as convolution, matrix multiplication, and batch normalization to operate on 512 bits of data per clock cycle (the equivalent of 32 double-precision floating-point numbers).
- **Acceleration**. Now with Intel® Deep Learning Boost built into the hardware, the performance acceleration extends to integer operations and handles dense computations characteristic of CNN and DNN workloads. It accelerates AI workloads, increasing Int16 and Int8 peak operations/second. Intel DL Boost was designed to accelerate performance of AI deep learning (inference) workloads (e.g., speech recognition, image recognition, object classification, machine translation, and others).
- **Parallelization**. Optimizations that help AI frameworks make the most efficient possible use of all logical processing cores available to them are another significant focus. Intel code contributions and co-engineering help frameworks divide workloads efficiently at the software thread level.
- **Data locality**. Intel draws on its platform expertise to design data structures that help applications maintain spatial and temporal locality of data. Together with approaches such as prefetching and cache-blocking techniques, these optimizations help ensure that data is available where and when it is needed.

In particular, Intel has an ongoing co-engineering relationship with Google for the optimization of TensorFlow. Initial optimizations for Intel® Xeon® processors using Intel MKL-DNN showed order-of-magnitude performance improvements[4]. Ongoing optimizations continue to build on those gains. For example, optimizations help to ensure that default TensorFlow operations can be replaced with the versions optimized for Intel architecture as much of the time as possible, while also reducing the associated overhead. Another focus of this effort is to fuse together multiple operations to use cache resources as efficiently as possible.

Notably, taking advantage of these optimizations does not require any software or configuration changes, beyond building TensorFlow using the build settings for Intel MKL-DNN, which is itself an open-source project. These optimizations are ongoing and made continually to the TensorFlow repository, with performance gains continuing to increase[5].

In addition to optimizing TensorFlow for per-node performance, cross-node operation is also vital, in terms of efficiency, performance, and scalability within multi-node training networks. Particularly for complex deep learning models, distributing training across multiple nodes is often desirable. Challenges tend to arise for users as they enable training models across clusters using native TensorFlow methods due to communication overhead and the need to make significant modifications to model-building code.

These dramatic capabilities for accelerating training of deep learning networks by executing that training across clusters of Intel Xeon processor-based nodes emphasize the suitability of existing clusters for these training activities. Particularly because training tends to occur sporadically, there is clear value to executing those jobs on converged infrastructure that can also be applied to other tasks.

Aside from intensive enablement efforts such as the optimization of TensorFlow, Intel is also engaged on a large number of open-source projects to provide expanded capabilities that contribute to converged workflows on HPC clusters. For example, Intel is currently developing capabilities for Apache Spark to take advantage of high-performance fabrics that provide messaging via OpenFabrics* interfaces. These efforts are part of development on many fronts that will improve outcomes from converged clusters in the future.

Enhanced Flexibility with a Multi-Solution Approach

Because there is no one-size-fits-all solution, Intel is engaged in a multi-solution approach which supports a retrofit strategy to existing environments. Resource managers are responsible for maintaining the job queue for a cluster environment, batch scheduling jobs, and allocating resources according to the priority and requirements of each job. To accommodate the existing HPC environments operated by various customers, Intel is developing multiple solutions in parallel, with planned development of additional solutions as time goes on.

Solution 1: Magpie* for HPC Batch Schedulers

For organizations that are already running batch schedulers in their HPC environments, Intel has developed an architecture to extend those environments to support AI and analytics workloads, as illustrated in Figure 2.

Intel is contributing to Lawrence Livermore National Laboratory's open source project called Magpie* to support running TensorFlow in HPC environments using Horovod* Magpie handles submitting jobs on behalf of AI or analytics workloads with negligible overhead, creating no discernable performance degradation. In addition, workloads benefit from optimizations to many of these environments that have been made by Intel. The abstraction offered by these scripts also dramatically simplifies implementation for customers.

In support of this approach, Intel has also developed best practices to ensure optimal performance and stability for various types of workloads. In this solution, Apache Spark runs in standalone mode as an HPC job. Simulation and modeling workloads continue to operate as usual, creating a unified environment from the standpoint of resource management. Intel is working with the ecosystem toward providing advanced support for this solution.

Solution 2: Univa* Grid Engine and Resource Broker

Univa* Grid Engine is a commercially licensed resource manager. The company recently made Univa Universal Resource Broker available as open source for managing and optimizing distributed applications, services, and big data frameworks. It abstracts distributed data center resources to create a single virtual pool, running across bare metal servers, virtual machines, hybrid clouds, and containers.

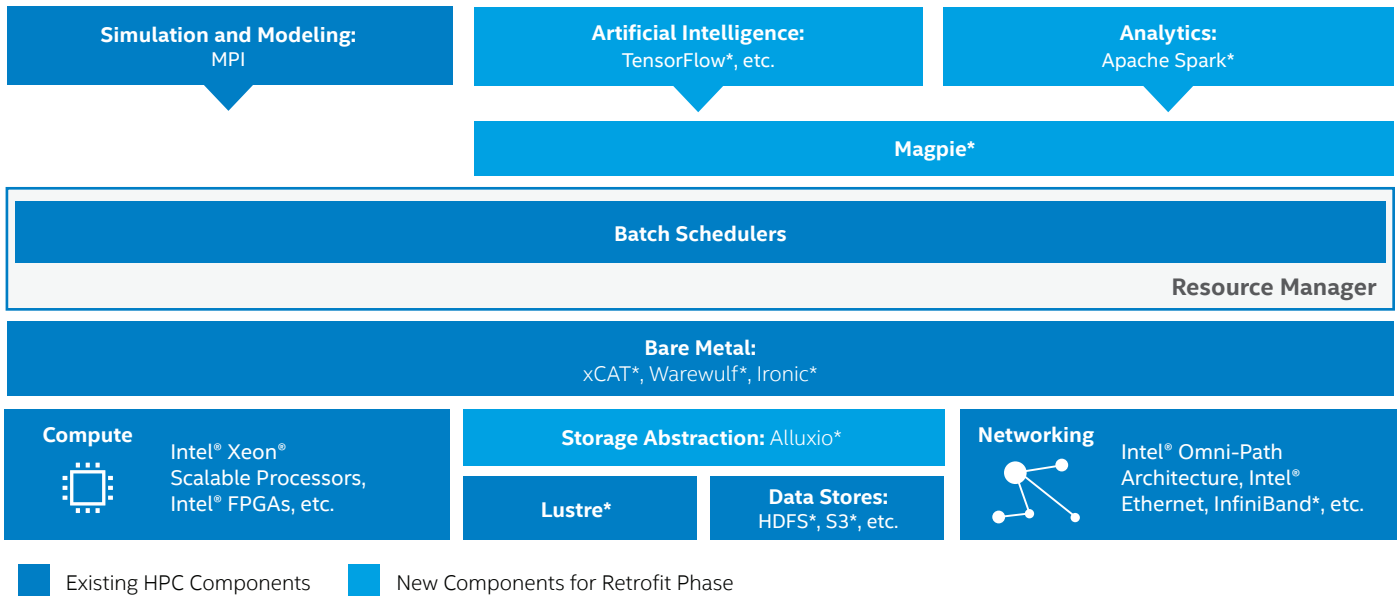

Figure 2. Solution 1: Extend HPC batch schedulers.

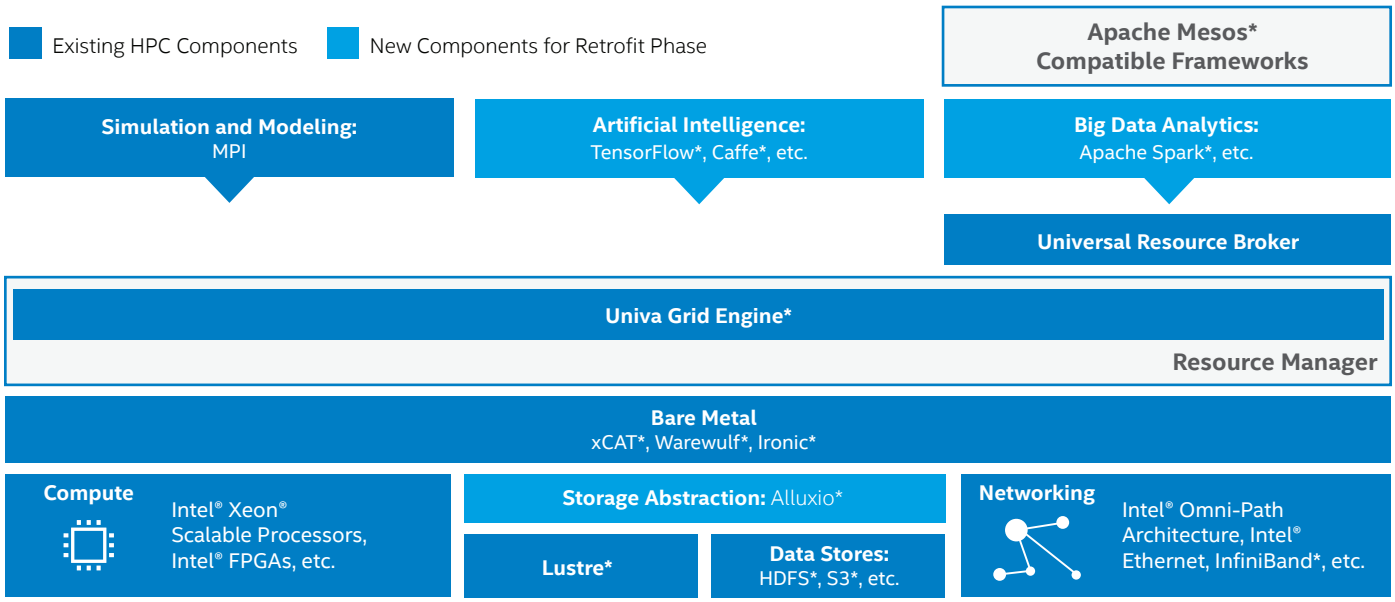

Figure 3. Solution 2: Univa* Grid Engine and Resource Broker.

This solution helps organizations that are already using Univa Grid Engine in production for scheduling and queueing to extend their clusters in support of workloads such as Apache Spark and TensorFlow, in addition to simulation and modeling jobs. In essence, it accomplishes this by using Universal Resource Broker as an adapter to integrate the various workloads with Univa Grid Engine, as shown in Figure 3. Specifically, Universal Resource Broker enables any Apache Mesos* compatible software to run on Univa Grid Engine. (Apache Mesos is an open-source cluster management project originally developed at UC Berkeley that supports a wide range of compatible frameworks.)

In this solution, simulation and modeling workloads continue to operate using existing APIs and command-line access to the Univa Grid Engine, requiring no change from existing implementations. Commercial support for the solution is available from Univa.

Conclusion

Enterprises and institutions that operate HPC clusters stand to gain significant efficiencies by running AI and analytics workloads on that existing infrastructure. While challenges exist due to fundamental differences in the programming approaches and technology stacks used among these workload types, advances by Intel and others in the ecosystem are making it viable for growing numbers of organizations to converge simulation and modeling, AI, and analytics workloads onto a single infrastructure.

By avoiding the need to build and operate multiple clusters, convergence dramatically reduces CAPEX requirements in the data center. Unifying cluster workloads also drives up efficiency by eliminating the need to move and stage data among clusters, reducing complexity and OPEX. Moreover, high-performance infrastructure improves the performance of AI and analytics workloads. Companies across industry verticals are adopting AI use cases, as shown in Table 1.

Table 1. AI use cases across industry verticals.

CONSUMER	HEALTH	FINANCE	RETAIL	GOVERNMENT	ENERGY	TRANSPORT	INDUSTRIAL
Smart Assistants	Enhanced Diagnostics	Algorithmic Trading	Support	Defense	Oil & Gas Exploration	Autonomous Cars, Ships	Factory Automation
Chatbots	Drug Discovery	Fraud Detection	Experience	Data Insights	Smart Grid	Automated Trucking	Predictive Maintenance
Search	Patient Care	Research	Marketing	Safety & Security	Operational Improvement	Aerospace	Precision Agriculture
Personalization	Research	Personal Finance	Merchandising	Resident Engagement	Conservation	Safety (incl. PPE)	Field Automation
Augmented Reality	Sensory Aids	Risk Mitigation	Loyalty	Smarter Cities	Seismic Analysis	Search & Rescue	LEAN waste removal
Robots	Security	Security	Supply Chain	Research	Reservoir Simulation	Security	Black Box analysis

As companies expand their use of AI and analytics in the coming years, a converged single infrastructure approach will pay dividends by adding new, innovative capabilities while reducing the infrastructure requirements to support them.

Accelerate HPC innovation: **www.intel.com/hpc**

[1] Average geomean of STREAM, HPCG, HPL, WRF, OpenFOAM, LS-Dyna, VASP, NAMD, LAMMPS, Black Scholes, and Monte Carlo., Individual workload may vary. Performance results are based on testing as of dates shown in configuration and may not reflect all publicly available security updates. See configuration disclosure for details. No product or component can be absolutely secure. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit www.intel.com/benchmarks. OpenFOAM Disclaimer: This offering is not approved or endorsed by OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com, and owner of the OPENFOAM® and OpenCFD® trademark. Configurations:

3.7x avg gain w/Intel® Xeon® Platinum 9242 processor vs 3-year old server: Average geomean of STREAM, HPCG, HPL, WRF, OpenFOAM, LS-Dyna, VASP, NAMD, LAMMPS, Black Scholes, and Monte Carlo. Individual workload may vary. Intel Xeon E5-2697 v4 Processor: Intel Reference Platform with 2S Intel Xeon E5-2697 v4 processors (2.3GHz, 18C), 8x16GB DDR4-2400, 1 SSD, Cluster File System: Panasas (124 TB storage) Firmware v6.3.3.a & OPA based IEEL Lustre, BIOS: SE5C610.86B.01.01.0027.071020182329, Microcode: 0xb00002e, Oracle Linux Server release 7.6 (compatible with RHEL 7.6) on a 7.5 kernel using ksplice for security fixes, Kernel: 3.10.0-862.14.4.el7.crt1.x86_64, OFED stack: OFED OPA 10.8 on RH7.5 with Lustre v2.10.4, HBA: 100Gbps Intel OPA 1 port PCIe x16, Switch: Intel OPA Edge Switch 100 Series 48 Port. STREAM OMP 5.10, Triad, HT=ON, Turbo=OFF, 1 thread per corescore: 128.36. HPCG, Binary included MKL 2019u1, HT=ON, Turbo=OFF, 1 thread per corescore: 23.78. HPL 2.1, HT=ON, Turbo=OFF, 2 threads per corescore: 1204.64. WRF 3.9.1.1, conus-2.5km, HT=ON, SMT=ON, 1 thread per corescore: 4.54. OpenFOAM 6.0, 42M_cell_motorbike, HT=ON, Turbo=OFF, 1 thread per corescore: 3500. LS-Dyna 9.3-Explicit AVX2 binary, 3car, HT=ON, SMT=ON, 1 thread per corescore: 2814. VASP 5.4.4, CuC, HT=ON, Turbo=OFF, 1 thread per corescore: 384.99. NAMD 2.13, apoa1, HT=ON, Turbo=OFF, 2 threads per corescore: 4.4. LAMMPS version 12 Dec 2018, Water, HT=ON, Turbo=ON, 2 threads per corescore: 54.72. Black Scholes, HT=ON, Turbo=ON, 2 threads per corescore: 2573.77. Monte Carlo, HT=ON, Turbo=ON, 2 threads per corescore: 43.2. Intel Xeon 9242 Processor: Intel Reference Platform with 2S Intel Xeon 9242 processors (2.2GHz, 48C), 16x16GB DDR4-2933, 1 SSD, Cluster File System: 2.12.0-1 (server) 2.11.0-14.1 (client), BIOS: PLYXCRB1.86B.0572.D02.1901180818, Microcode: 0x4000017, CentOS 7.6, Kernel: 3.10.0-957.5.1.el7.x86_64, OFED stack: OFED OPA 10.8 on RH7.5 with Lustre v2.10.4, HBA: 100Gbps Intel OPA 1 port PCIe x16, Switch: Intel OPA Edge Switch 100 Series 48 Port. STREAM OMP 5.10, Triad, HT=ON, Turbo=OFF, 1 thread per corescore: 407. HPCG, Binary included MKL 2019u1, HT=ON, Turbo=OFF, 1 thread per corescore: 81.91. HPL 2.1, HT=ON, Turbo=OFF, 2 threads per corescore: 5314. WRF 3.9.1.1, conus-2.5km, HT=ON, SMT=ON, 1 thread per corescore: 1.44. OpenFOAM 6.0, 42M_cell_motorbike, HT=ON, Turbo=OFF, 1 thread per corescore: 1106. LS-Dyna 9.3-Explicit AVX2 binary, 3car, HT=ON, SMT=ON, 1 thread per corescore: 768. VASP 5.4.4, CuC, HT=ON, Turbo=OFF, 1 thread per corescore: 133.96. NAMD 2.13, apoa1, HT=ON, Turbo=OFF, 2 threads per corescore: 19.9. LAMMPS version 12 Dec 2018, Water, HT=ON, Turbo=ON, 2 threads per corescore: 276.1. Black Scholes, HT=ON, Turbo=ON, 2 threads per corescore: 9044.32. Monte Carlo, HT=ON, Turbo=ON, 2 threads per corescore: 227.62. OpenFOAM Disclaimer: This offering is not approved or endorsed by OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com, and owner of the OPENFOAM® and OpenCFD® trademark. Date of testing March 15th, 2019

[2] OpenFOAM Disclaimer: This offering is not approved or endorsed by OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com, and owner of the OPENFOAM® and OpenCFD® trademark. Configurations: LINPACK: AMD EPYC 7601: Supermicro AS-2023US-TR4 with 2 AMD EPYC 7601 (2.2GHz, 32 core) processors, SMT OFF, Turbo ON, BIOS ver 1.1a, 4/26/2018, microcode: 0x8001227, 16x32GB DDR4-2666, 1 SSD, Ubuntu 18.04.1 LTS (4.17.0-041700-generic Retpoline), High Performance Linpack v2.2, compiled with Intel(R) Parallel Studio XE 2018 for Linux, Intel MPI version 18.0.0.128, AMD BLIS ver 0.4.0, Benchmark Config: Nb=232, N=168960, P=4, Q=4, Score =1095GFs, tested by Intel as of July 31, 2018. vs. 1-node, 2x Intel® Xeon® Platinum 9282 cpu on Walker Pass with 768 GB (24x 32GB 2933) total memory, ucode 0x400000A on RHEL7.6, 3.10.0-957.el7.x86_65, IC19u1, AVX512, HT off, Turbo on, score=6411, test by Intel on 2/16/2019.

[3] **1-node, 2x AMD EPYC™ 7601** on Supermicro with 128 GB (16 slots / 8 GB / 2666) total memory, ucode 0x8001227 on Linux-4.15.0-43-generic-x86_64-with-Ubuntu-18.04-bionic, TOSHIBA MG03ACA100, gcc (Ubuntu 7.3.0-27ubuntu1~18.04) 7.3.0, SMT on, Turbo on, AIXPRT, Tensorflow 1.12 (Stock) on ResNet 50 using fp32 throughput, BS 4, 32 instances, score=195.8, test by Intel on 2/8/2019. 1-node, 2x Intel® Xeon® Platinum 9242 cpu on Walker Pass with 288 GB (18 X 16GB 2933) total memory, ucode 0x400000A on Linux-4.15.0-45-generic-x86_64-with-Ubuntu-18.04-bionic, , gcc (Ubuntu 7.3.0-27ubuntu1~18.04) 7.3.0, HT on, Turbo on, AIXPRT, OpenVINO R5 (DLDTK Version :1.0.19154) on ResNet 50 using int8 throughput, BS 2, 96 instances, score=4930, test by Intel on 2/20/2019

[4] *TensorFlow* Optimizations on Modern Intel® Architecture*. Intel® Developer Zone, https://software.intel.com/en-us/articles/tensorflow-optimizations-on-modern-intel-architecture.

[5] *Using Intel® Xeon® processors for Multi-node Scaling of TensorFlow* with Horovod*.* Intel® Developer Zone, https://software.intel.com/en-us/articles/using-intel-xeon-processors-for-multi-node-scaling-of-tensorflow-with-horovod.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. Check with your system manufacturer or retailer or learn more at intel.com. No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

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.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

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 SSSE3 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.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit: http://www.intel.com/performance.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.