



# Boost Performance and Scalability of End-to-End Enterprise AI

with 4th Gen Intel® Xeon® Scalable Processors and Kubernetes/Kubeflow

**Business Challenge:** Most large enterprises have now completed one or two successful artificial intelligence (AI) projects — but to increase AI's value, enterprises must efficiently scale machine learning across a wide variety of use cases. Disparate systems, data silos and limited compute infrastructure are barriers to AI scalability.

 Up to  
**3.5X Higher**  
Transfer Learning Throughput  
Compared to 3rd Gen AMD  
EPYC processors

 Up to  
**3.9X Higher**  
Inference Throughput for NLP  
Compared to 3rd Gen AMD  
EPYC processors

## Solution Overview and Summary

**Solution:** To achieve end-to-end scalability for enterprise AI, organizations must invest in a unified platform that can integrate data across use cases, including predictive analytics, deep learning training and transfer learning and deep learning inference. Because AI workloads can be computationally demanding, they need a flexible infrastructure that can accommodate growing compute requirements today and into the future.

This pre-validated hardware-and-software solution integrates open-source software components with several hardware performance-enhancing features of 4th Generation Intel® Xeon® Scalable processors. These processors have the most built-in accelerators of any CPU on the market<sup>1</sup> to help improve performance efficiency for emerging workloads, especially those powered by AI. For example, Intel® Advanced Matrix Extensions (Intel® AMX) are specifically designed to provide massive speedup to the low-precision math operations that underpin AI inference for natural-language processing (NLP), recommendation systems and image recognition.

From data ingestion and data preparation to analysis, the solution provides tools optimized for Intel® architecture. Examples include NumPy, XGBoost, TensorFlow and the SigOpt model development platform, as well as optimized libraries like Intel oneAPI Data Analytics Library (oneDAL) and Intel® oneAPI Deep Neural Network Library (oneDNN). The solution's hardware-software stack is enabled with [Kubeflow](#), the machine-learning toolkit for [Kubernetes](#), which provides production-grade container orchestration.

**Results:** By deploying enterprise AI on this solution, organizations can unify their AI workloads onto a single platform and achieve higher AI workload performance. For example, this solution delivers up to 3.5x higher transfer learning throughput and up to 3.9x higher inference throughput for NLP compared to workloads running on 3rd Gen AMD EPYC processors.<sup>2,3</sup> Organizations can also potentially consolidate servers to decrease data center footprint. See the full discussion of test results on [page 2](#).

### Kubernetes Workload Pod Setup Details

- Topology, CPU and memory manager policies were configured in Kubernetes to tie specific workload pods to specific CPU NUMA nodes.
- Two pods (two workload processes) per node with affinity were launched.
- Four dedicated logical CPU cores were set aside for other Kubernetes resources.

## Test Methodology

Various use cases in enterprise segments like retail, healthcare and financial services benefit from NLP, one of the most prevalent forms of AI. People also experience NLP when they interact with applications like digital voice assistants. The widely known NLP model, BERT, is often used to pre-train unlabeled text by jointly conditioning both left and right context in all layers of the model. Pre-training, however, is expensive and usually a one-time approach for each language. Fine-tuning, on the other hand, is inexpensive, and it starts from an existing pre-trained model. Fine-tuning is achieved by adding one additional output layer and without significant modifications to the model architecture.

For this design testing, the BERT-Large uncased (whole word masking) pre-trained model was used as the model checkpoint. This model contains 340 million parameters. The Stanford Question Answering Dataset (SQuAD) v1.1 dataset was used for the fine-tuning analysis. The requirements for the use case were followed as documented in the Model Zoo for Intel Architecture online repository [documentation](#). To showcase the performance and scalability of BERT-Large fine-tuning using the scalable end-to-end enterprise AI stack solution, potential Intel optimizations like [oneDNN-optimized TensorFlow](#) and [Horovod](#) for distributed training were employed. Horovod was deployed with [Intel® MPI Library](#), which further takes advantages of Intel optimizations at the MPI layer. Container images and a set of [MPIJob spec](#) files were used to run the BERT-Large workload on top of Kubeflow's [Training Operator](#).

## Results

Figures 1, 2 and 3 illustrate the performance gains for the BERT-Large model resulting from upgrading from a system powered by 3rd Gen Intel Xeon Scalable processors or 3rd Gen AMD EPYC processors to a system with 4th Gen Intel Xeon Scalable processors that use numerical precisions accelerated by Intel AMX or Intel® Deep Learning Boost with Vector Neural Network Instructions (VNNI).<sup>2,3</sup>

- Up to 2.9x and 3.5x transfer learning (fine-tuning) throughput improvement for 4th Gen Intel Xeon Scalable processor with BF16 precision compared to 3rd Gen Intel Xeon Scalable processors and 3rd Gen AMD EPYC processors with FP32 precision.
- Up to 6.11x and 3.48x inference throughput improvement for 4th Gen Intel Xeon Scalable processor with BF16 and INT8 precisions, respectively, compared to 3rd Gen Intel Xeon Scalable processors with FP32 and INT8 precisions.
- Up to 3.9x inference throughput improvement for 4th Gen Intel Xeon Scalable processor with FP32 precision compared to a 3rd Gen AMD EPYC processor-based system; up to 2.1x inference throughput improvement with FP32 precision compared to a 4th Gen AMD EPYC processor-based system.

The table to the right provides comprehensive testing data for transfer learning throughput scalability across hardware configurations and number of server nodes.

Increased Transfer Learning Throughput  
Model: BERT-Large (Higher Is Better)

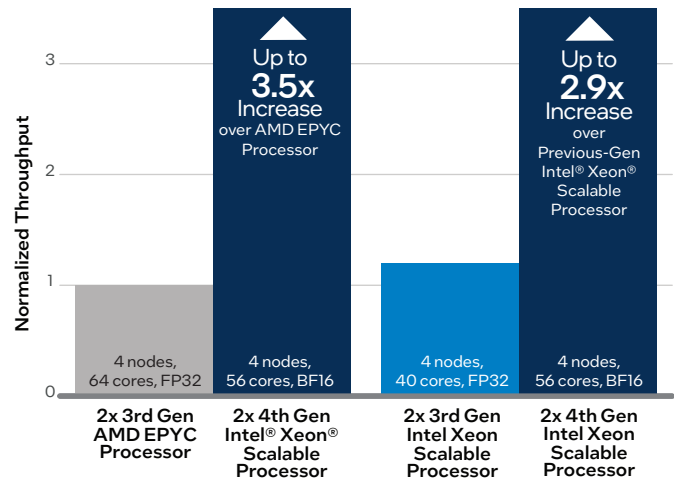


Figure 1. Transfer learning throughput improvement for 4th Gen Intel® Xeon® Scalable processors with BF16 precision compared to 3rd Gen AMD EPYC processors and 3rd Gen Intel Xeon Scalable processors.<sup>2</sup>

Increased NLP Inference Throughput  
Model: BERT-Large (Higher Is Better)

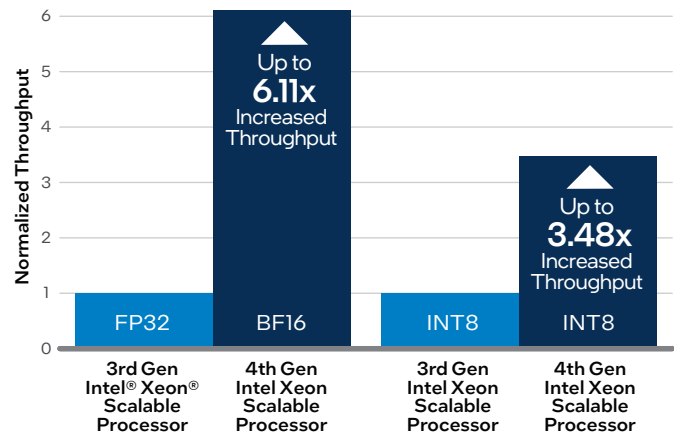


Figure 2. Inference performance improvement for 4th Gen Intel® Xeon® Scalable processor compared to the previous generation.<sup>3</sup>

| Throughputs (samples/second)  | Throughputs (samples/second) |         |         |
|---|------------------------------|---------|---------|
|   | 1 Node                       | 2 Nodes | 4 Nodes |
| 3rd Gen AMD EPYC 7773x processor with FP32 precision (batch size; 96)             | 5.06                         | 10.10   | 20.17   |
| 3rd Gen Intel® Xeon® Platinum 8380 processor with FP32 precision (batch size; 64) | 6.08                         | 12.37   | 24.58   |
| 4th Gen Intel Xeon Platinum 8480+ processor with FP32 precision (batch size; 96)  | 8.71                         | 17.47   | 34.70   |
| 4th Gen Intel Xeon Platinum 8480+ processor with BF16 precision (batch size; 136) | 17.92                        | 35.50   | 71.02   |

### Increased NLP Inference Throughput (FP32) Model: BERT-Large (Higher Is Better)

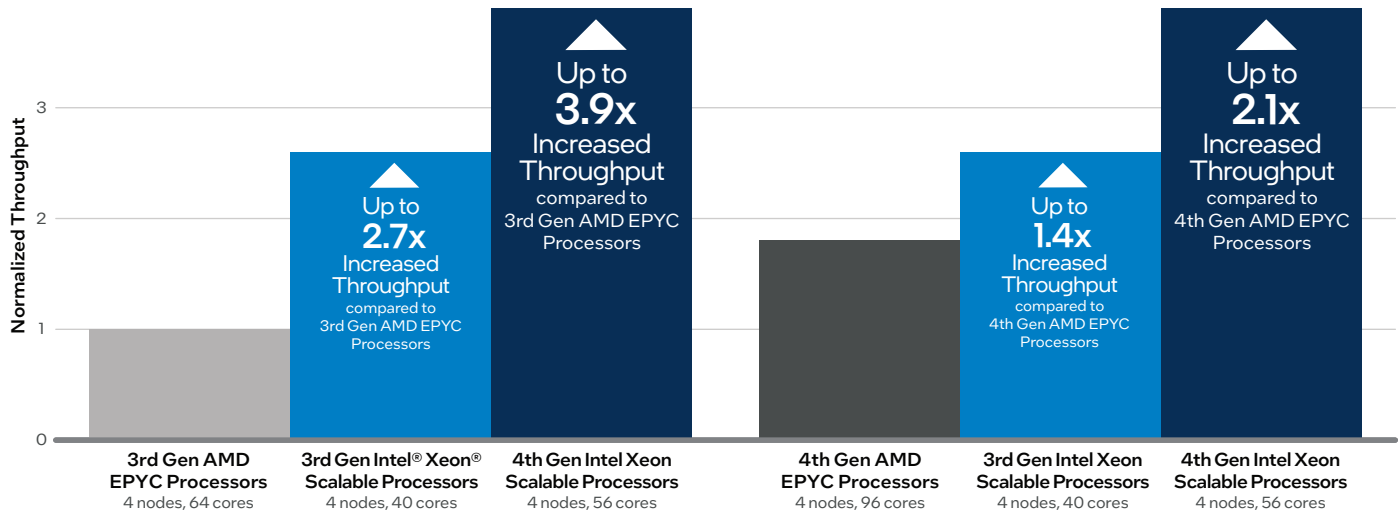


Figure 3. Inference performance improvement for 3rd and 4th Gen Intel Xeon Scalable processors (FP32) compared to 3rd and 4th Gen AMD EPYC processors.<sup>3</sup>

## Configuration Details

The following tables provide information about components and settings of the infrastructure used for performance analysis and characterization testing.

| Hardware Configurations |   |  |   |
|-------------------------|---|--|---|
|                         | 3rd Gen AMD EPYC processor-based worker nodes                           | 3rd Gen Intel Xeon Scalable processor-based worker nodes               | 4th Gen Intel Xeon Scalable processor-based worker nodes                |
| <b>Processor</b>        | 2x AMD EPYC 7773X processor (64 cores, 2.2 GHz)                         | 2x Intel Xeon Platinum 8380 processor (40 cores, 2.3 GHz)              | 2x Intel Xeon Platinum 8480+ processor (56 cores, 2.0 GHz)              |
| <b>Memory</b>           | 512 GB (16x 32 GB DDR4 3200 MT/s)                                       | 512 GB (16x 32 GB DDR4 3200 MT/s)                                      | 512 GB (16x 32 GB DDR5 4800 MT/s)                                       |
| <b>Network</b>          | Broadcom NetXtreme BCM5720 1 GbE, Broadcom NetXtreme BCM57414 10/25 GbE | Broadcom NetXtreme BCM5720 1 GbE, Intel Ethernet Adapter E810-C 10 GbE | Intel Ethernet Adapter X710 1 GbE, Intel Ethernet Adapter E810-C 10 GbE |

| Important System Settings               |  |
|---|--|
| <b>Number of Nodes</b>                  | 1, 2, 4  |
| <b>Hyper-Threading</b>                  | Enabled  |
| <b>Turbo Boost</b>                      | Enabled  |
| <b>Power &amp; Perf Policy</b>          | Performance  |
| <b>Accelerator Technologies Enabled</b> | <ul style="list-style-type: none"> <li>Intel Advanced Matrix Extensions (Intel AMX)</li> <li>Intel Advanced Vector Extensions 512 (Intel AVX-512)</li> <li>Intel Deep Learning Boost with Vector Neural Network Instructions (VNNI)</li> <li>Intel Hyper-Threading Technology</li> <li>Intel Turbo Boost Technology</li> </ul> |

| Software Versions                        |                         |            |
|--|-------------------------|------------|
| <b>OS</b>                                | Rocky Linux             | 8.6        |
| <b>Orchestration Layer</b>               | Kubernetes              | 1.24.6     |
| <b>Container Runtime Interface (CRI)</b> | containerd              | 1.6.8      |
| <b>Container Network Interface (CNI)</b> | Calico/Multus           | 3.23.3/3.8 |
| <b>Container Storage Interface (CSI)</b> | Ceph                    | 17.2.5     |
| <b>Object Storage</b>                    | MinIO                   | 4.5.2      |
| <b>MLOps</b>                             | Kubeflow                | 1.6.1      |
| <b>Container Image Repository</b>        | Docker Private Registry | 2.8.1      |

## Profiles and Workloads

The following table describes the workload used in testing.

|                                     | 3rd Gen AMD EPYC<br>7773X Processor   | 3rd Gen Intel® Xeon® Platinum<br>8380 Processor | 4th Gen Intel Xeon Platinum<br>8480+ Processor |
|-------------------------------------|---|---|--|
| Workload                            | BERT-Large machine-learning model fine-tuning (transfer learning) and inference |   |  |
| Kubernetes Topology Manager         | Enabled   |   |  |
| # of pods per node                  | 2   |   |  |
| System logical cores                | 256   | 160   | 224  |
| # of logical cores for Kubernetes   | 4   | 4   | 4  |
| # of logical cores for workload     | 252   | 156   | 220  |
| # of logical cores per workload pod | 126   | 78  | 110  |
| Memory assigned per workload pod    | 250 GB  | 250 GB  | 250 GB   |

## Conclusion

Configured with 4th Gen Intel Xeon Scalable processors with Intel AMX and DDR5 memory, the scalable end-to-end enterprise AI stack allows users to process up to 3.5x higher transfer learning throughput and up to 3.9x higher inference throughput than 3rd Gen AMD EPYC processors on a four-node cluster. Furthermore, a gen-over-gen comparison of Intel Xeon Scalable processors shows that a 4th Gen Xeon Scalable processor-based distributed platform can perform up to 2.9x better than a 3rd Gen Intel Xeon Scalable processor-based platform.

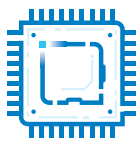
As enterprise leaders continue to adopt AI to improve their business processes and drive value, Intel's hardware and software enhancements will continue to provide advancements in the field of AI to deliver compelling performance in the transfer learning and inference space.

## Further information

- [Scalable End-to-End Enterprise AI on 4th Gen Intel® Xeon® Scalable Processors white paper](#)
- [4th Generation Intel® Xeon® Scalable processors](#)
- [Intel® Advanced Matrix Extensions \(Intel® AMX\)](#)
- [Intel® artificial intelligence and deep learning solutions](#)

### Authors

Abirami Prabhakaran, Francisco M. Casares, Mishali Naik, Marcin Hoffmann, Marcin Gajzler, Venkata Kranthi Kumar Dhanala, Vaishali Deshpande, Katarzyna Szymkow, Andy Morris, Ronak Shah



Learn more about  
4th Gen Intel®  
Xeon® Scalable  
processors



Learn more about  
Intel® Advanced  
Matrix Extensions  
(Intel® AMX)



Contact your Intel  
representative to  
learn more about  
this solution.

## Solution Provided By:



<sup>1</sup> <https://www.intel.com/content/www/us/en/products/docs/processors/xeon-accelerated/4th-gen-xeon-scalable-processors.htm>

<sup>2</sup> **Transfer Learning Results. 3rd Gen AMD EPYC processor-based system:** Test by Intel as of November 26, 2022. 4 nodes each with 2x AMD EPYC 7773X 64-core processor, 64 cores/socket, 2 sockets, Hyper-threading = ON, Turbo = ON, total memory 512 GB (16x 32 GB 3200 MHz [run @ 3200 MHz]), BIOS 2.8.4, microcode 0xa001229, Rocky Linux 8.6 (Green Obsidian), kernel 4.18.0-372.32.1.el8\_6.x86\_64, gcc 8.5.0, BERT-Large, TF 2.10.0, Horovod 0.25.0, OpenMPI 4.1.4, Python 3.8.12, openblas-devel-0.3.15, moreutils-0.63, gcc-8.5.0, gcc-c++-8.5.0, make-4.2.1, cmake-3.20.2, libffi-3.1, libffi-devel-3.1, git-2.31.1, curl-7.61.1, vim-enhanced-8.0.1763, wget-1.19.5, ca-certificates-2022.2.50, libjpeg-turbo-devel-1.5.3, libpng-devel-1.6.34, python38-3.8.12, python38-devel-3.8.12, python38-pip-19.3.1, python3-distutils-extra-2.39, rdma-core-37.2, perl-5.26.3, bc-1.07.1, expect-5.45.4, net-snmp-5.8, net-snmp-utils-5.8, tcl-8.6.8, libibmad-37.2, librdmacm-utils-37.2, libibverbs-utils-37.2, gcc-gfortran-8.5.0, libfabric-1.14.0, libpsm-11.3.0, mpi4py-3.1.3, Kubespray 2.20.0, Multus 3.8, Calico 3.23.3, containerd 1.6.8, Docker Registry 2.8.1, Kubernetes 1.24.6 (Topology Manager-enabled), Kubeflow 1.6.1, DirectPV 3.2.0, MinIO 4.5.2, Prometheus 2.39.1, BERT-Large model fine-tuning w/SQuAD 1.1 dataset, batch size = 96, learning rate 1e-5, SeqLength 384, throughput (sentences/sec) for 1, 2 and 4 nodes (FP32): 5.06, 10.10, 20.17. **3rd Gen Intel Xeon Scalable processor-based system:** Test by Intel as of November 4, 2022. 4 nodes each with 2x Intel® Xeon® Platinum 8380 processors @ 2.30 GHz, 40 cores/socket, 2 sockets, Intel® Hyper-Threading Technology = ON, Intel® Turbo Boost Technology = ON, total memory 512 GB (16x 32GB 3200 MHz [run @ 3200 MHz]), Dell PowerEdge R750, BIOS 1.6.5, microcode 0xd000375, Rocky Linux 8.6 (Green Obsidian), kernel 4.18.0-372.32.1.el8\_6.x86\_64, gcc 8.5.0, BERT-Large, Intel® Optimization of TensorFlow 2.11 (Intel internal), Horovod 0.26.1, Intel® MPI Library/oneCCL 2021.7, Python 3.8.12, openblas-devel-0.3.15, moreutils-0.63, gcc-8.5.0, gcc-c++-8.5.0, make-4.2.1, cmake-3.20.2, libffi-3.1, libffi-devel-3.1, git-2.31.1, curl-7.61.1, vim-enhanced-8.0.1763, wget-1.19.5, ca-certificates-2022.2.50, libjpeg-turbo-devel-1.5.3, libpng-devel-1.6.34, python38-3.8.12, python38-devel-3.8.12, python38-pip-19.3.1, python3-distutils-extra-2.39, rdma-core-37.2, perl-5.26.3, bc-1.07.1, expect-5.45.4, net-snmp-5.8, net-snmp-utils-5.8, tcl-8.6.8, libibmad-37.2, librdmacm-utils-37.2, libibverbs-utils-37.2, gcc-gfortran-8.5.0, libfabric-1.14.0, libpsm-11.3.0, mpi4py-3.1.3, Kubespray 2.20.0, Multus 3.8, Calico 3.23.3, containerd 1.6.8, Docker Registry 2.8.1, Kubernetes 1.24.6 (Topology Manager-enabled), Kubeflow 1.6.1, DirectPV 3.2.0, MinIO 4.5.2, Prometheus 2.39.1, BERT-Large model fine-tuning w/SQuAD 1.1 dataset, batch size = 64, learning rate 1e-5, SeqLength 384, throughput (sentences/sec) for 1, 2 and 4 nodes (FP32): 6.08, 12.37, 24.58. **4th Gen Intel Xeon Scalable processor-based system:** Test by Intel as of November 4, 2022. 4 nodes each with 2x Intel Xeon Platinum 8480+ processor @ 2.00 GHz, 56 cores/socket, 2 sockets, Intel Hyper-Threading Technology = ON, Intel Turbo Boost Technology = ON, total memory 512 GB (16x 32 GB 4800 MHz [run @ 4800 MHz]), Quanta Cloud Technology Inc., QuantaGrid D54Q-2U, BIOS 3A06, microcode 0x2b000081, Rocky Linux 8.6 (Green Obsidian), kernel 4.18.0-372.32.1.el8\_6.x86\_64, gcc 8.5.0, BERT-Large, Intel Optimization of TensorFlow 2.11 (Intel internal), Horovod 0.26.1, Intel MPI/oneCCL 2021.7, Python 3.8.12, openblas-devel-0.3.15, moreutils-0.63, gcc-8.5.0, gcc-c++-8.5.0, make-4.2.1, cmake-3.20.2, libffi-3.1, libffi-devel-3.1, git-2.31.1, curl-7.61.1, vim-enhanced-8.0.1763, wget-1.19.5, ca-certificates-2022.2.50, libjpeg-turbo-devel-1.5.3, libpng-devel-1.6.34, python38-3.8.12, python38-devel-3.8.12, python38-pip-19.3.1, python3-distutils-extra-2.39, rdma-core-37.2, perl-5.26.3, bc-1.07.1, expect-5.45.4, net-snmp-5.8, net-snmp-utils-5.8, tcl-8.6.8, libibmad-37.2, librdmacm-utils-37.2, libibverbs-utils-37.2, gcc-gfortran-8.5.0, libfabric-1.14.0, libpsm-11.3.0, mpi4py-3.1.3, Kubespray 2.20.0, Multus 3.8, Calico 3.23.3, containerd 1.6.8, Docker Registry 2.8.1, Kubernetes 1.24.6 (Topology Manager-enabled), Kubeflow 1.6.1, DirectPV 3.2.0, MinIO 4.5.2, Prometheus 2.39.1, BERT large model fine-tuning w/SQuAD 1.1 dataset, batch size = 136, learning rate 1e-5, SeqLength 384, throughput (sentences/sec) 1, 2 and 4 nodes (BFloat16): 17.92, 35.50, 71.02.

<sup>3</sup> **Inference Results. 3rd Gen Intel Xeon Scalable processor-based system:** Test by Intel as of March 18, 2023. 4 nodes each with 2x Intel Xeon Platinum 8380 processor @ 2.30 GHz, 40 cores/socket, 2 sockets, Intel Hyper-Threading Technology = ON, Intel Turbo Boost Technology = ON, total memory 512 GB (16x 32 GB DDR4 3200 MHz [run @ 3200 MHz]), Intel Corporation, reference server platform M50CYP2SB2U, BIOS SE5C6200.86B.0022.D64.2105220049, microcode 0xd000363, Rocky Linux 8.6 (Green Obsidian), kernel 4.18.0-372.32.1.el8\_6.x86\_64, Kubespray 2.20.0, Multus 3.8, Calico 3.23.3, containerd 1.6.8, Docker Registry 2.8.1, Kubernetes 1.24.6, Kubeflow 1.6.1, rook.io 1.10 (Ceph), MinIO 4.5.2, Prometheus 2.39.1, Container Images: intel/language-modeling:spr-bert-large-inference, dataset: SQuAD1.1, workload: BERT-Large inference, FP32, INT8, batch size = 16, 32, 64, 128, SeqLength 384. Throughput (sentences/sec) for 1 node: 27.88 (FP32; BS128), 66.15 (INT8; BS128). **4th Gen Xeon Scalable processor-based system:** Test by Intel as of March 18, 2023. 4-nodes each with 2x Intel Xeon Platinum 8480+ processor @ 2.00 GHz, 56 cores/socket, 2 sockets, Intel Hyper-Threading Technology = ON, Intel Turbo Boost Technology = ON, total memory 512 GB (16x 32 GB DDR5 4800 MHz [run @ 4800 MHz]), Quanta Cloud Technology Inc., QuantaGrid D54Q-2U, BIOS 3A11.uh, microcode 0x2b000181, Rocky Linux 8.6 (Green Obsidian), kernel 4.18.0-372.32.1.el8\_6.x86\_64, Kubespray 2.20.0, Multus 3.8, Calico 3.23.3, containerd 1.6.8, Docker Registry 2.8.1, Kubernetes 1.24.6, Kubeflow 1.6.1, rook.io 1.10 (Ceph), MinIO 4.5.2, Prometheus 2.39.1, Container Images: intel/language-modeling:spr-bert-large-inference, Dataset: SQuAD1.1, Workload: BERT-Large Inference, FP32, INT8, BFloat16, batch size = 16, 32, 64, 128, SeqLength 384. Throughput (sentences/sec) for 1 node: 40.95 (FP32, BS128), 170.38 (BFloat16; BS128), 230.55 (INT8; BS128). **3rd Gen AMD EPYC processor-based system and 4th Gen AMD EPYC processor-based system:** 7763 (64 cores) and 9654 (96 cores) FP32/BS128/SeqLength 384 <https://www.amd.com/system/files/documents/amd-epyc-9004-pb-aiml.pdf> Workload: BERT-Large Inference, FP32, batch size 128, SeqLength 384. Throughput (sentences/sec) for 1 node: 10.27 (FP32, BS128, 3rd Gen), 18.65 (FP32, BS128, 4th Gen).

Performance varies by use, configuration, and other factors. Learn more on the [Performance Index](#) site. Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. No product or component can be absolutely secure. Your costs and results may vary. Intel technologies may require enabled hardware, software, or service activation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others. © Intel Corporation 0823/JCAP/KC/PDF