

Choose the Best Accelerated Technology

AI on Intel Architecture

Dr. Séverine Habert– AI Engineering Manager

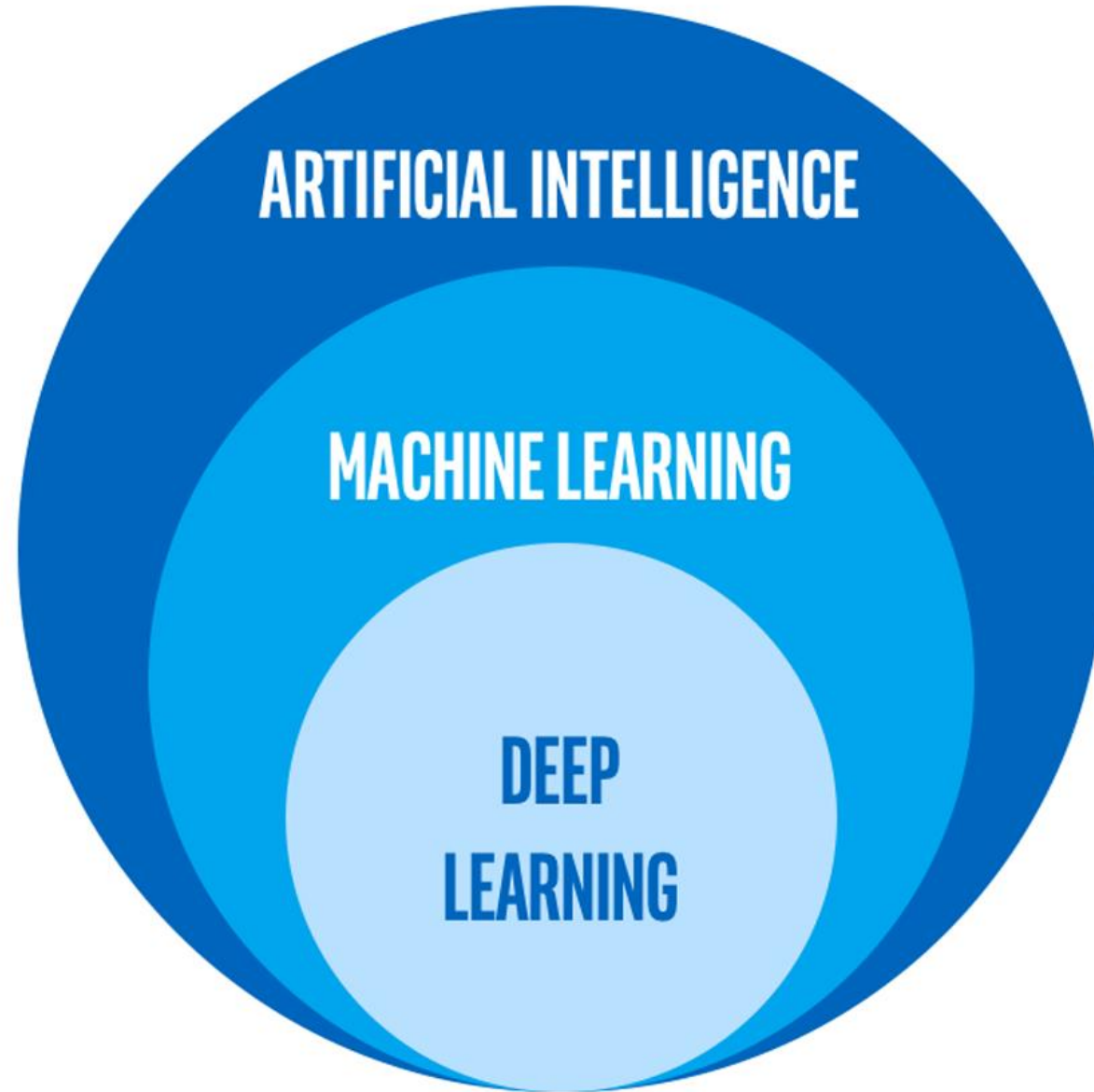
Severine.habert@intel.com

October 13th 2022



intel[®]

lrz

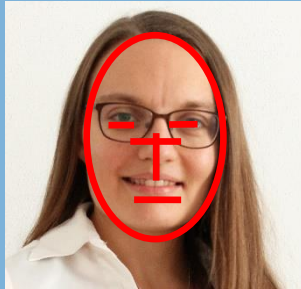


MACHINE LEARNING AND DEEP LEARNING

CLASSICAL MACHINE LEARNING

How do you engineer the best features?

$N \times N$



(f_1, f_2, \dots, f_K)

Roundness of face
Distance between eyes
Nose width
Eye socket depth
Cheek bone structure
Jaw line length
Etc.

CLASSIFIER ALGORITHM

SVM
Random Forest
Naïve Bayes
Decision Trees
Logistic Regression
Ensemble methods

Séverine

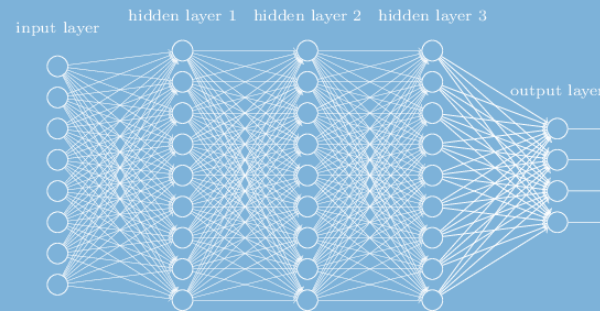
DEEP LEARNING

How do you guide the model to find the best features?

$N \times N$



NEURAL NETWORK



Séverine

Intel Hardware for AI

Flexible AI Acceleration

CPU *only*

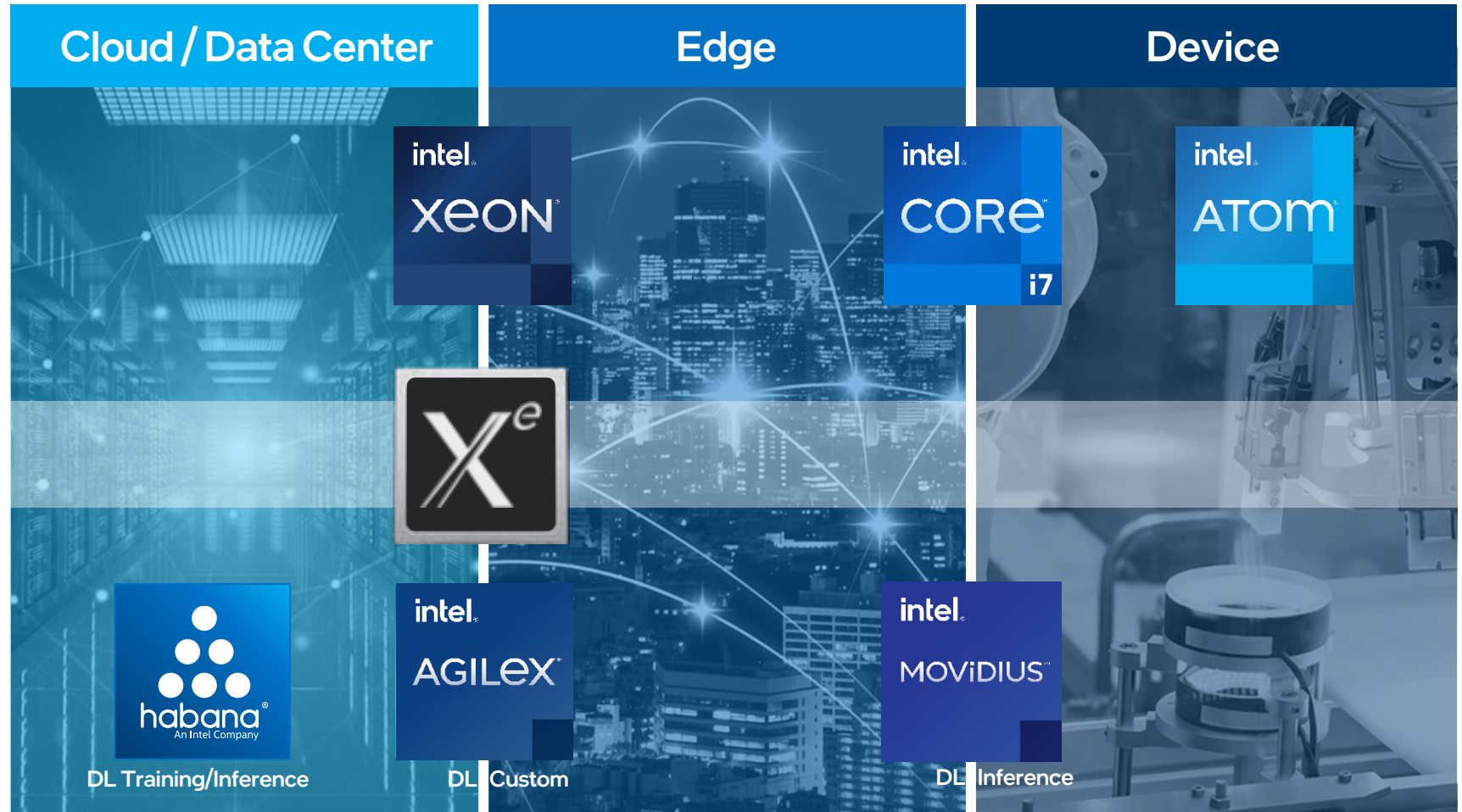
Built-in AI acceleration for mainstream AI use cases

CPU + GPU

When compute is dominated by AI, HPC, graphics, and/or real-time media

CPU + custom

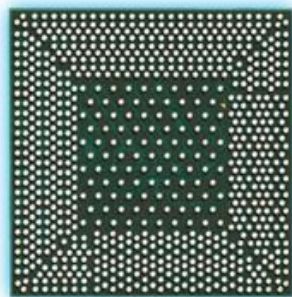
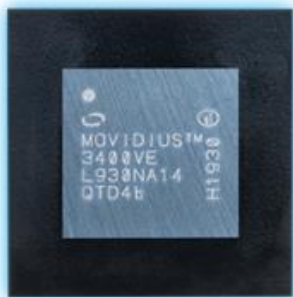
When compute is dominated by deep learning (DL)



Intel® Movidius™ VPU and Accelerator Cards

Built for Edge AI

- ✓ Deep learning inference + computer vision + media
- ✓ Faster memory bandwidth
- ✓ Groundbreaking high-efficiency architecture
- ✓ OpenVINO toolkit enabled



Flexible Form Factors



Edge Experiences



Intel® Xeon® Scalable Processors

The **Only** Data Center CPU with Built-in AI Acceleration

Intel Advanced Vector Extensions 512
Intel Deep Learning Boost (Intel DL Boost)
Intel Optane Persistent Memory

Shipping

Cascade Lake

New Intel DL Boost (VNNI)
New memory storage hierarchy

Cooper Lake

Intel DL Boost (BFLOAT16)

April 2021

Ice Lake

Intel DL Boost (VNNI) and new
Intel Software Guard Extensions
(Intel® SGX) that enable new
AI use cases like federated learning

2022

Sapphire Rapids

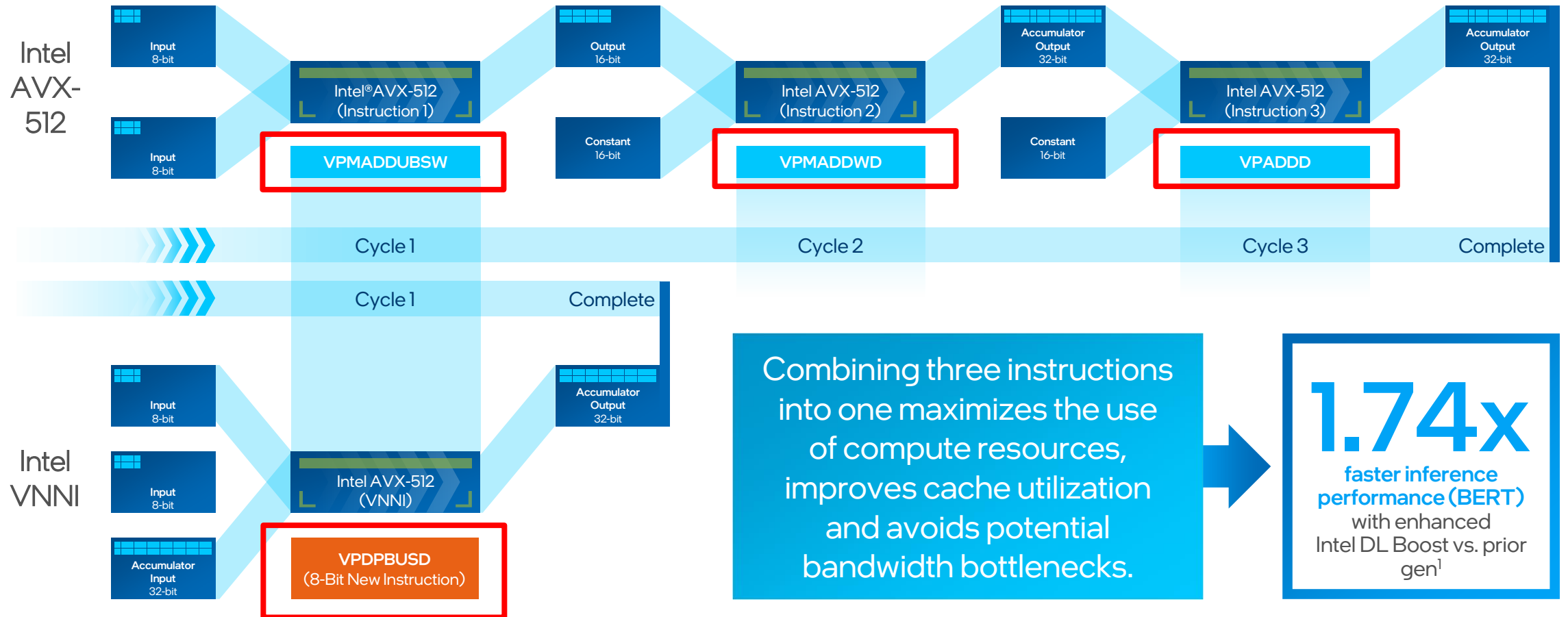
Intel Advanced Matrix Extensions (AMX)
extends built-in AI acceleration
capabilities on Xeon Scalable

Leadership performance

Intel Deep Learning Boost

A Vector Neural Network Instruction (VNNI)

Extends Intel AVX-512 to Accelerate AI/DL Inference

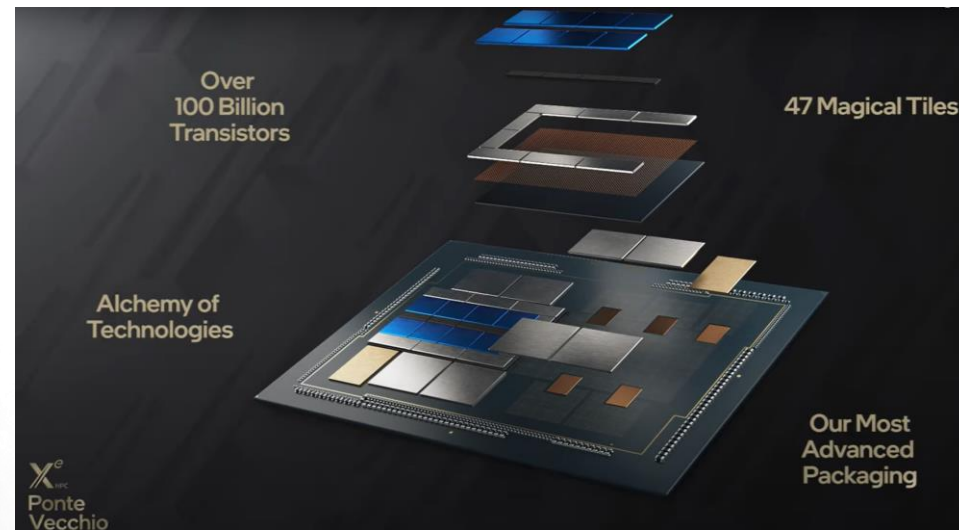


1. See [123] at www.intel.com/3gen-xeon-config. Results may vary.

X^e HPC (Ponte Vecchio)

Leadership Performance for Data-level Parallel AI Workloads

>40 active tiles, over 100 billion transistors integrated into a single package



Powering New Phase of SuperMUC-NG at Leibniz Supercomputing Centre (LRZ)



<https://www.youtube.com/watch?v=JzbNII0AcwY>

Habana – an Intel Company



Deep Learning ASIC for Training and Inference



Gaudi accelerators in AWS EC2 instances, leverages up to 8 Gaudi accelerators and deliver up to 40% better price performance than current GPU-based EC2 instances for training

All products, computer systems, dates, and figures are preliminary based on current expectations, and are subject to change without notice.

AI Software Stack for Intel XPU's

oneAPI

One Programming Model for Multiple Architectures & Vendors

Freedom to Make Your Best Choice

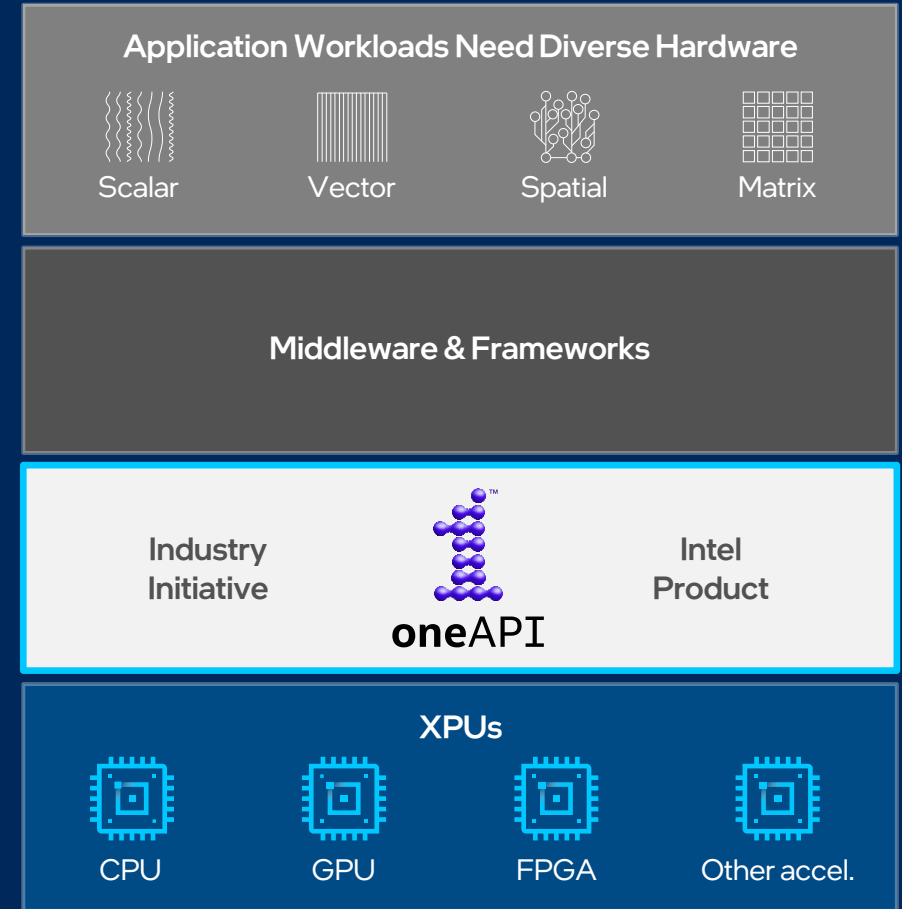
- Choose the best accelerated technology the software doesn't decide for you

Realize all the Hardware Value

- Performance across CPU, GPUs, FPGAs, and other accelerators

Develop & Deploy Software with Peace of Mind

- Open industry standards provide a safe, clear path to the future
- Compatible with existing languages and programming models including C++, Python, SYCL, OpenMP, Fortran, and MPI



Intel's oneAPI Ecosystem

Built on Intel's Rich Heritage of CPU Tools Expanded to XPU

oneAPI

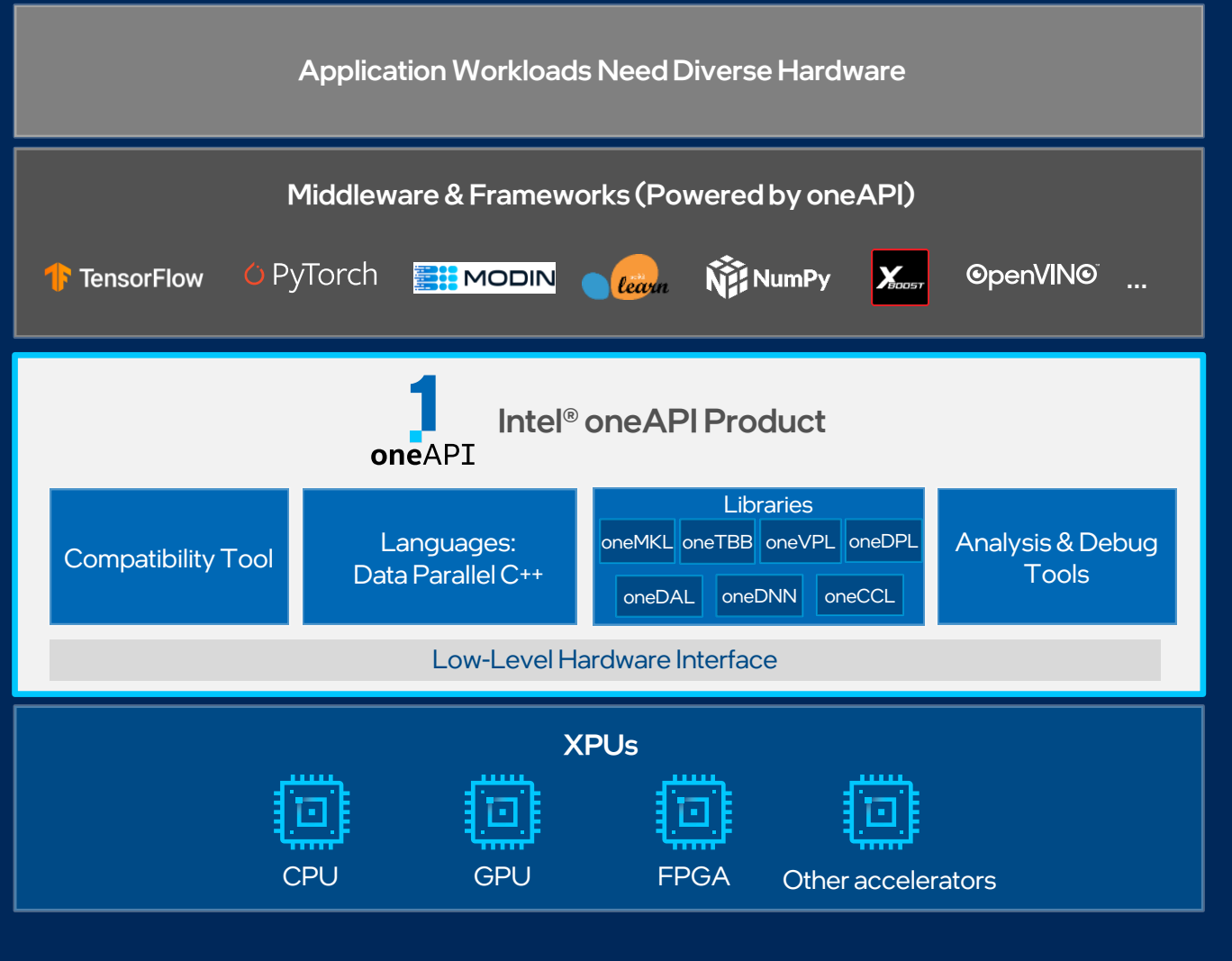
A cross-architecture language based on C++ and SYCL standards

Powerful libraries designed for acceleration of domain-specific functions

A complete set of advanced compilers, libraries, and porting, analysis and debugger tools

Powered by oneAPI

Frameworks and middleware that are built using one or more of the oneAPI industry specification elements, the DPC++ language, and libraries listed on oneapi.com.



[Available Now](#)

Intel oneAPI Software Tools for AI & Analytics

Intel® oneAPI Toolkits



Intel® oneAPI AI Analytics Toolkit

Accelerate machine learning & data science pipelines with optimized deep learning frameworks & high-performing Python libraries

Data Scientists, AI Researchers, DL/ML Developers



Intel® oneAPI Base Toolkit

Incl. Intel® oneAPI Deep Neural Network Library (oneDNN), Intel® oneAPI Collective Communications Library (oneCCL), & Intel® oneAPI Data Analytics Library (oneDAL)

Optimize primitives for algorithms and framework development

DL Framework Developers - Optimize algorithms for Machine Learning & Analytics

Toolkit Powered by oneAPI

Intel® Distribution of OpenVINO™ Toolkit

Deploy high performance inference & applications from edge to cloud

AI Application, Media, & Vision Developers

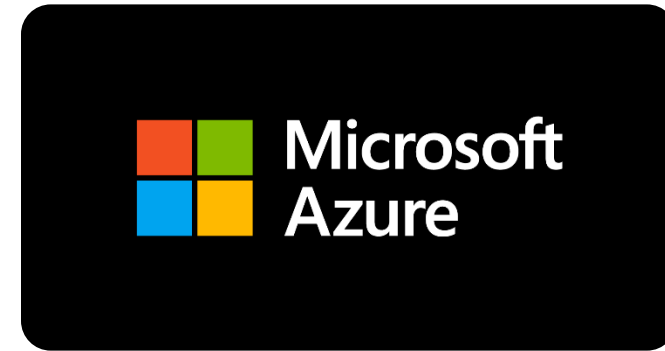


oneAPI Ecosystem Endorsements for AI domain

The industry needs a programming model where developers can take advantage of an array of innovative hardware architectures. The goal of oneAPI is to provide increased choice of hardware vendors, processor architectures, and faster support of next-generation accelerators. Microsoft has been using oneAPI elements across Intel hardware offerings as part of its initiatives and supports the open standards-based specification. We are excited to support our customers with choice and accelerate the growth of AI and machine learning.

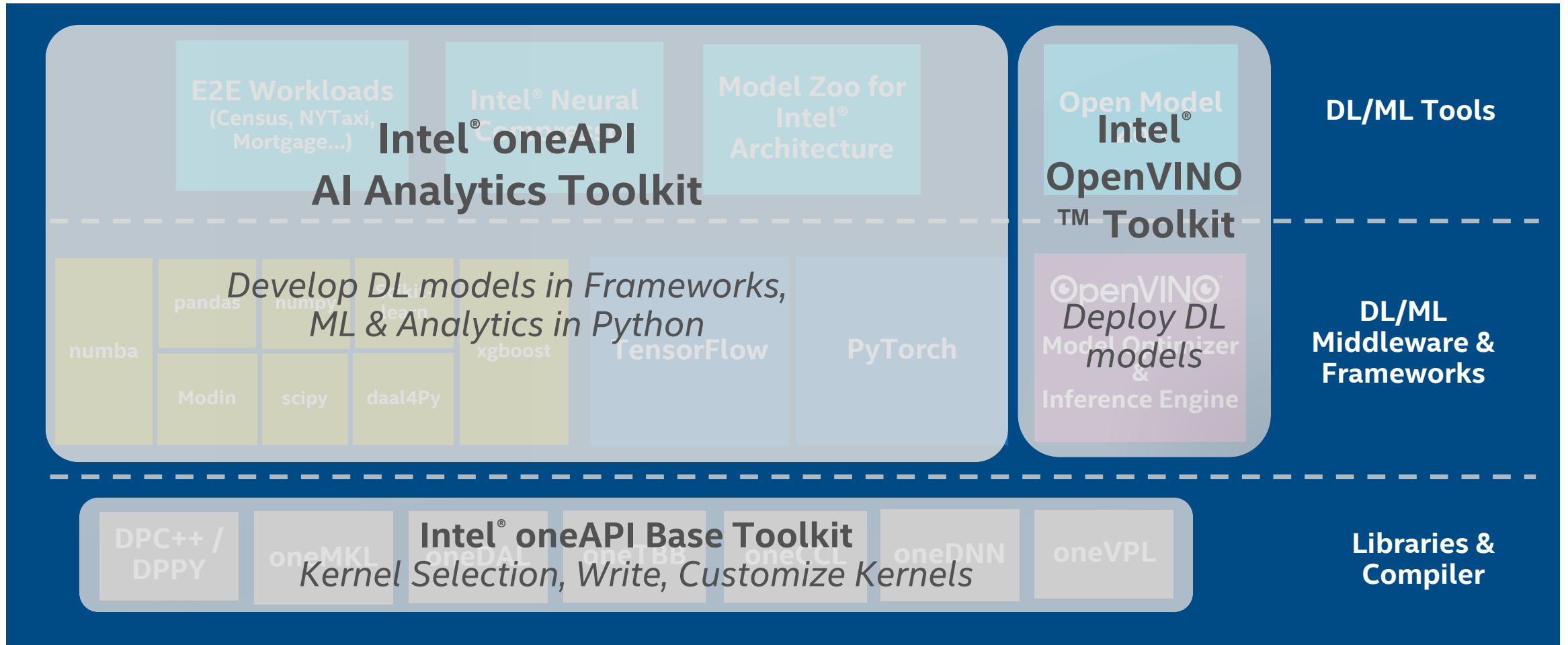
- Tim Harris, Principal Architect, Azure AI, Microsoft

With the growth of AI, machine learning, and data-centric applications, the industry needs a programming model that allows developers to take advantage of rapid innovation in processor architectures. TensorFlow supports the oneAPI industry initiative and its standards-based open specification. oneAPI complements TensorFlow's modular design and provides increased choice of hardware vendor and processor architecture, and faster support of next-generation accelerators. TensorFlow uses oneAPI today on Xeon processors and we look forward to using oneAPI to run on future Intel architectures.



AI Software Stack for Intel XPU

Intel offers a Robust Software Stack to Maximize Performance of Diverse Workloads



Full Set of Intel oneAPI cross-architecture AI ML & DL Software Solutions

Intel® AI Analytics Toolkit

Powered by oneAPI

Accelerate end-to-end AI and data analytics pipelines with libraries optimized for Intel® architectures

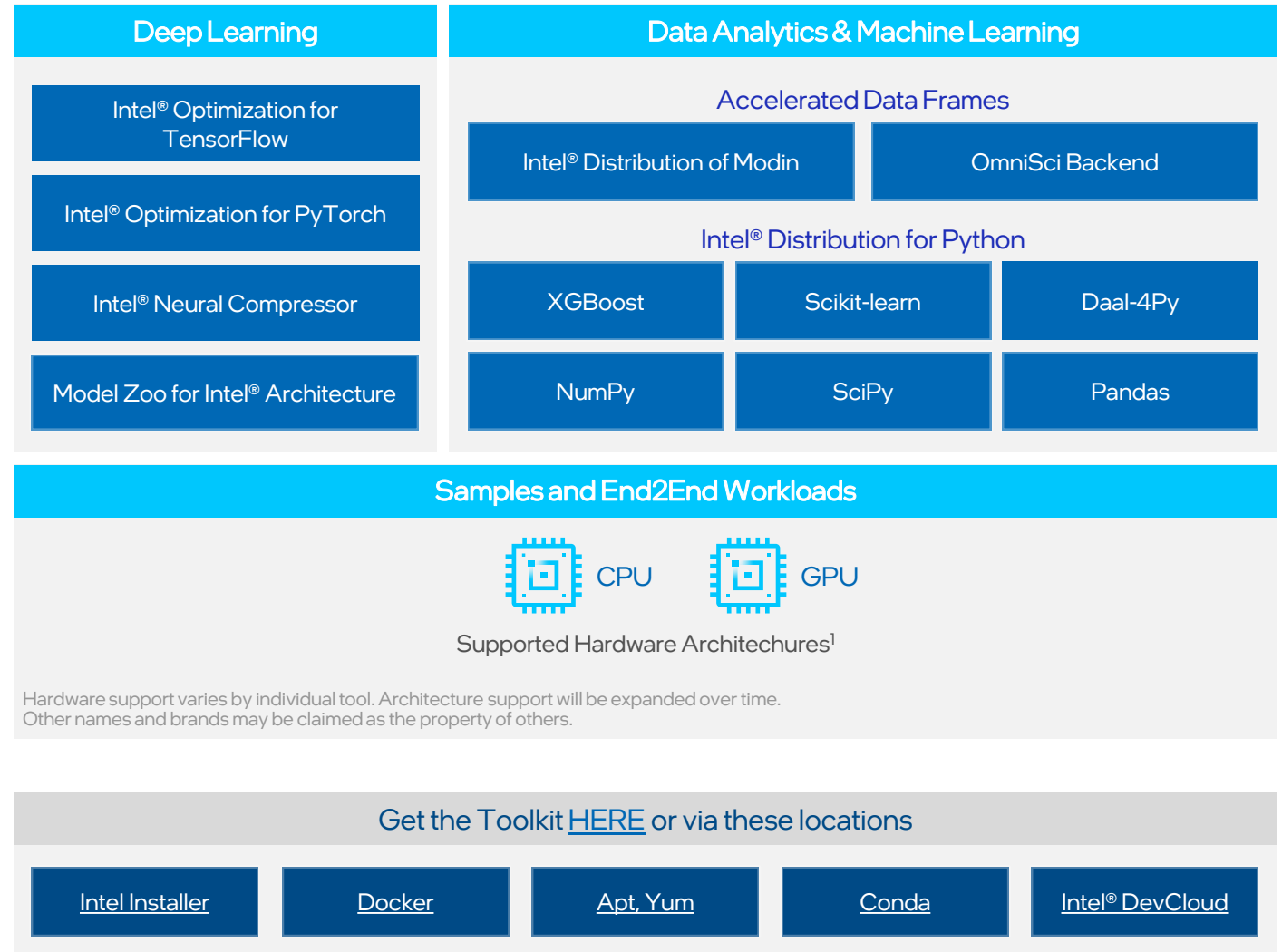
Who Uses It?

Data scientists, AI researchers, ML and DL developers, AI application developers

Top Features/Benefits

- Deep learning performance for training and inference with Intel optimized DL frameworks and tools
- Drop-in acceleration for data analytics and machine learning workflows with compute-intensive Python packages

Learn More: software.intel.com/oneapi/ai-kit



oneAPI Available on Intel® DevCloud for oneAPI

A development sandbox to develop, test and run workloads across a range of Intel CPUs, GPUs, and FPGAs using Intel's oneAPI software.

Get Up & Running In Seconds!

Sign up at:
software.intel.com/devcloud/oneapi

intel
DevCloud



1 Minute to Code

No Hardware Acquisition

No Download, Install or Configuration

Easy Access to Samples & Tutorials

Support for Jupyter Notebooks, Visual Studio Code

High-Performance Deep Learning Using Intel® Distribution of OpenVINO™ toolkit - Powered by oneAPI

A toolkit for fast, more accurate real-world results using high-performance AI and computer vision inference deployed into production on Intel XPU architectures (CPU, GPU, FPGA, VPU) from edge to cloud

Who needs this product?

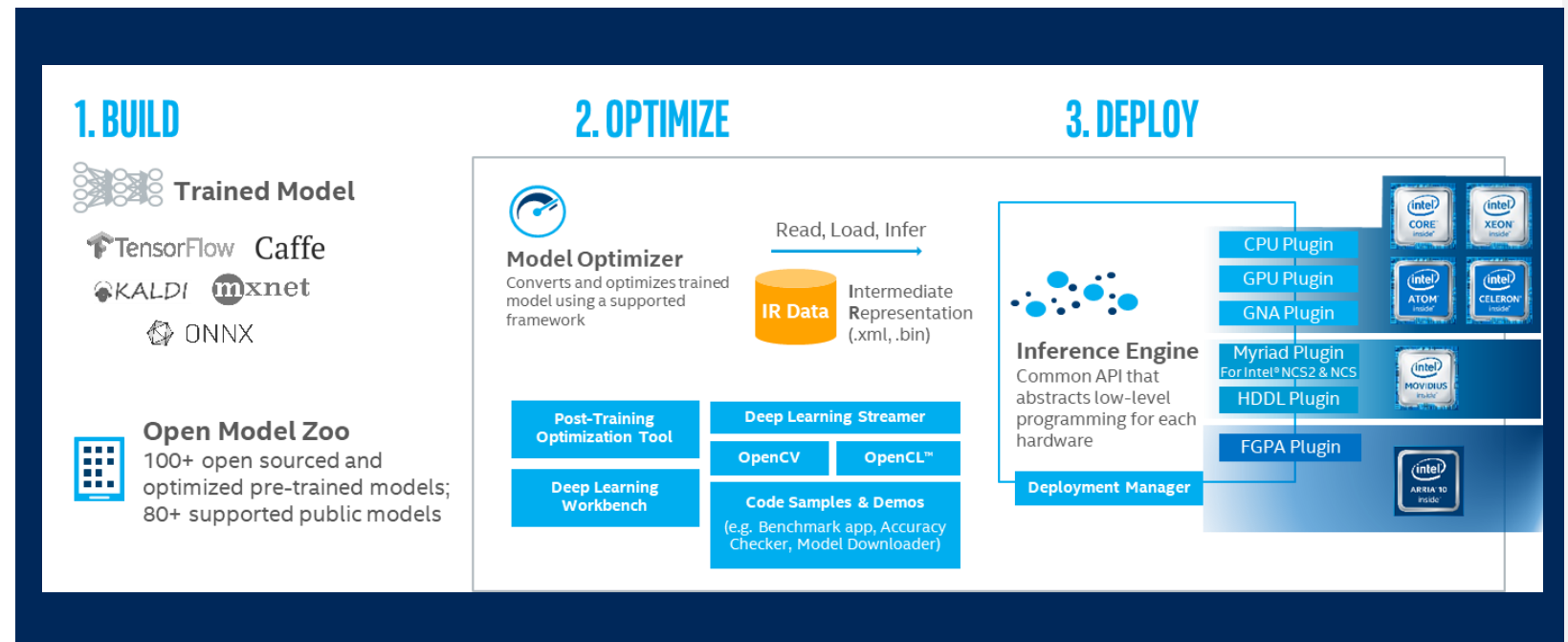
AI application developers, OEMs, ISVs, System Integrators, Vision and Media developers

Top Features/Benefits

High-performance, deep learning inference deployment

Streamlined development; ease of use

Write once, deploy anywhere

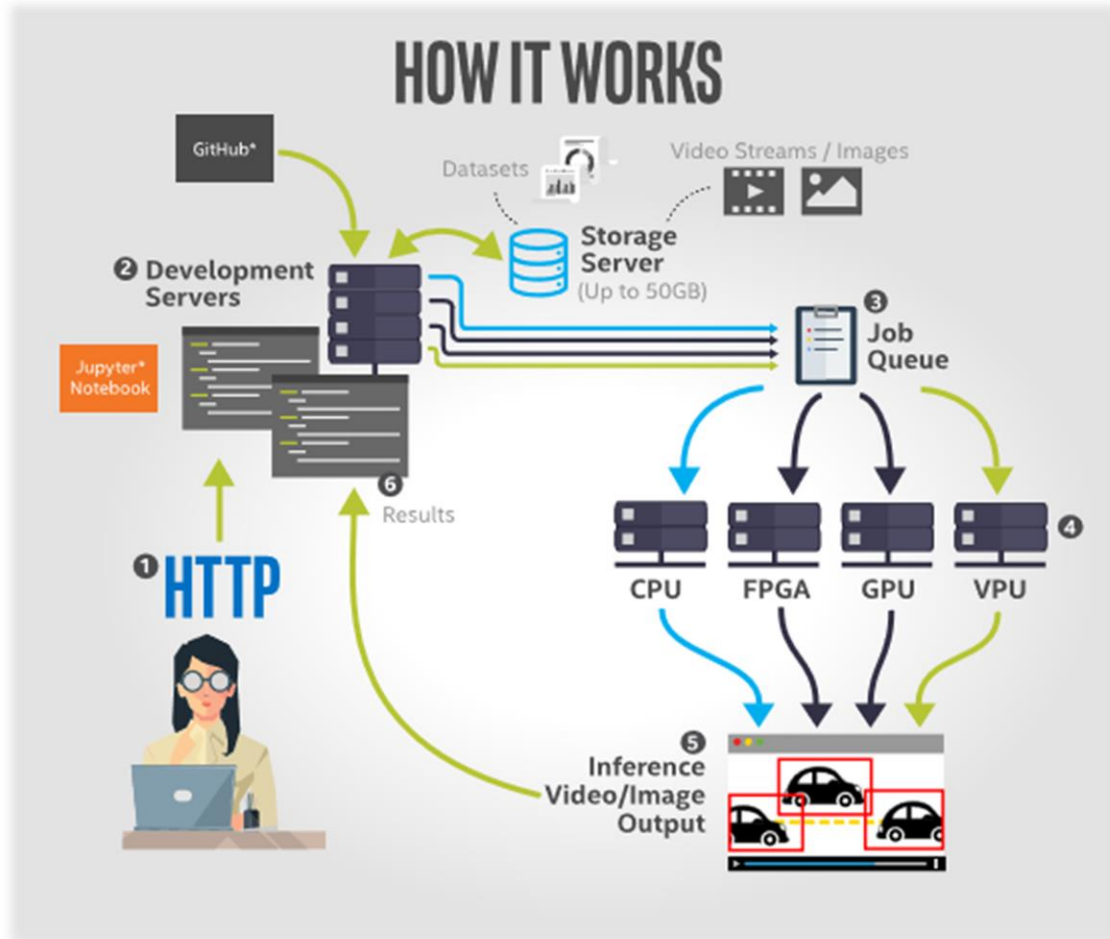


Proven, industry-leading accelerated technology

software.intel.com/opencvino-toolkit

Accelerate Time to Production with Intel® DevCloud for the Edge

See immediate AI Model performance across Intel's vast array of Edge Solutions



- **Instant, Global Access**
Run AI applications from anywhere in the world
- **Prototype on the Latest Hardware and Software**
Develop knowing you're using the latest Intel technology
- **Benchmark your Customized AI Application**
Immediate feedback - frames per second, performance
- **Reduce Development Time and Cost**
Quickly find the right compute for your edge solution

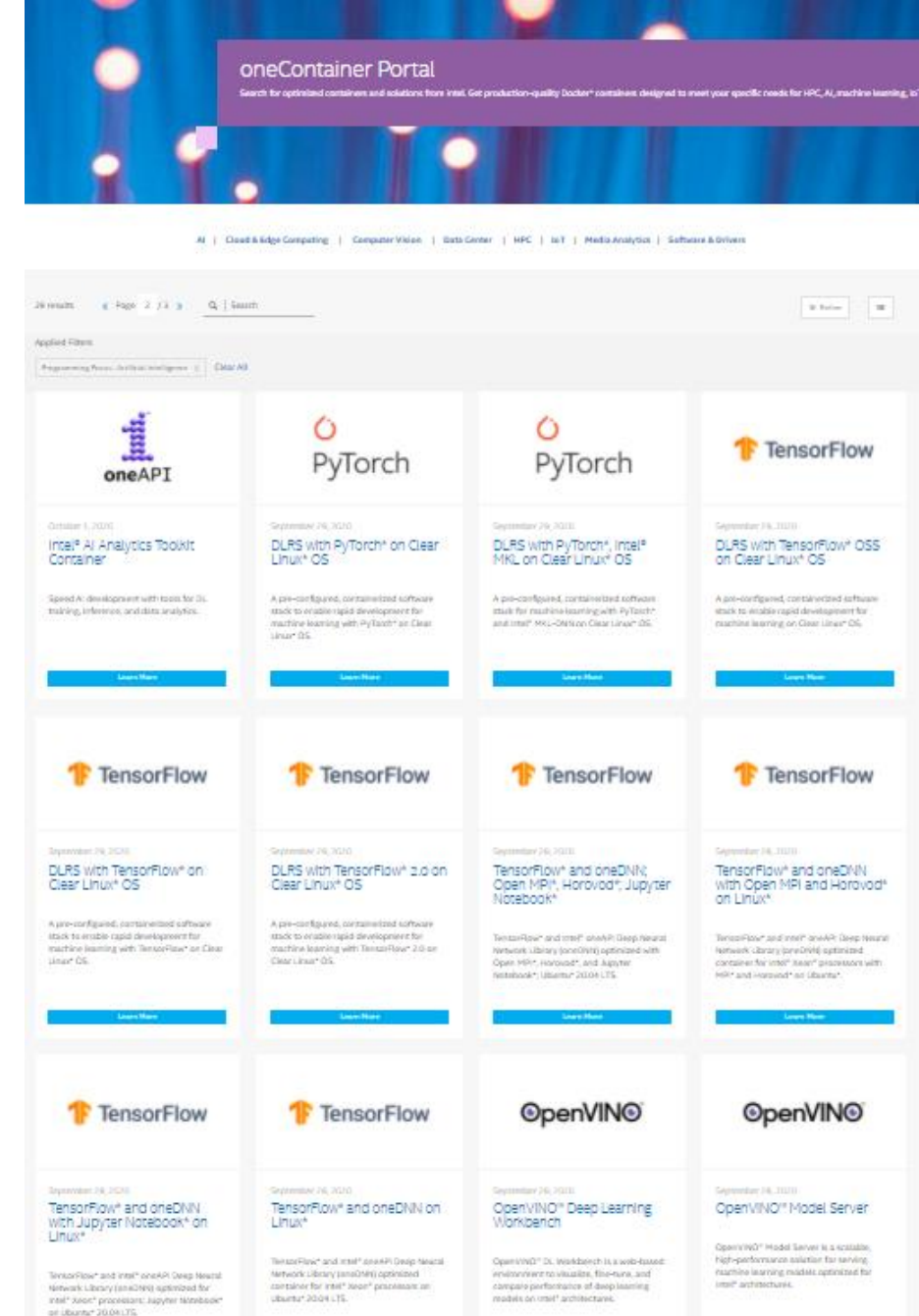
[Sign up now for access](#)

AI Containers for Flexibility

- Optimized, validated, deployable AI containers
- Available via Docker containers. Will expand to include Kubernetes orchestrations, Helm charts
- [Access from oneContainer Portal](#)
 - Include containers with ready-to-use AI software stacks
 - And containers with full AI workloads (including models)



Topology	Frameworks	Topology	Framework
DLRM	PYT	Mask R-CNN	PYT, TF, OV
ResNet50	PYT, TF, OV	RNN-T	PYT, TF, OV
BERT-large	PYT, TF, OV	3D-UNet	TF, OV
Transformer-LT	PYT, TF	DIEN	TF
MobileNet-v1	PYT, TF, OV	Wide & Deep	PYT, TF
SSD-Mobilenet-v1	PYT, TF, OV	RNX101	
SSD-Resnet34	PYT, TF, OV	Yolo-V3	PYT, TF, OV
WaveNet*	TF	NCF*	TF



Which Toolkit Should I Use

Use Both!

Intel® oneAPI Analytics Toolkit & Intel® Distribution of OpenVINO™ toolkit

Toolkits are complementary to each other and recommendation is to use them both based on your current phase of AI Journey

- I am **exploring and analyzing data**; I am **developing models**
- I want **performance and compatibility** with frameworks and libraries I use
- I would like to have **drop-in acceleration** with little to no additional code changes
- I prefer **not to learn any new tools or languages**



Data Scientist/ML Developer
Intel® oneAPI AI Analytics Toolkit



App Developer
Intel® Distribution of OpenVINO™ toolkit

- I am **deploying models**
- I want **leading performance and efficiency** across multiple target HW
- I'm concerned about **having lower memory footprint**, which is critical for deployment
- I am **comfortable with learning and adopting a new tool or API** to do so

If you prefer working on primitives and to optimize kernels and algorithms directly using oneAPI libraries (oneDNN, oneCCL & oneDAL), then use [Intel® oneAPI Base Toolkit](#)

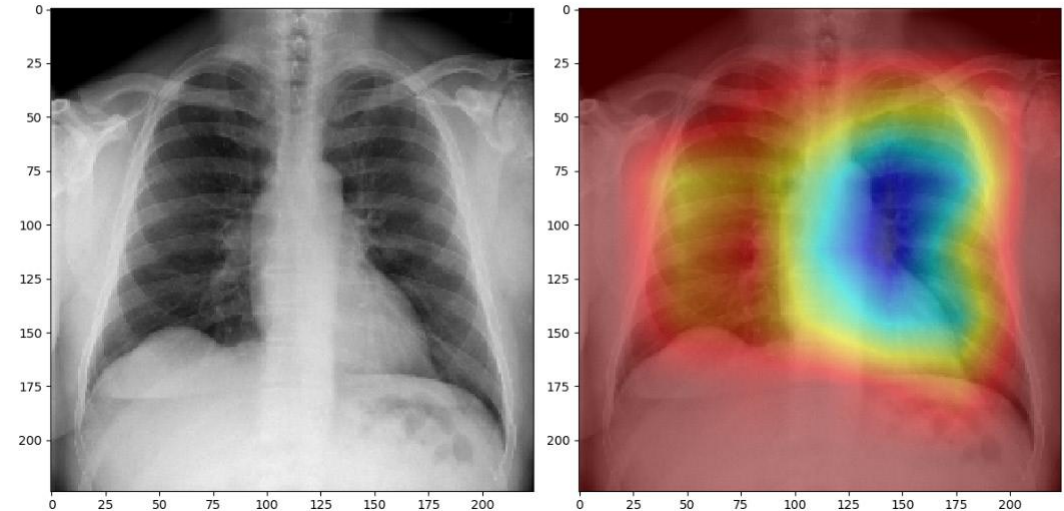
Accrad AI-based Solution Helps Accelerate COVID-19 Diagnosis

Optimized by Intel® oneAPI Analytics Toolkit & Intel® Distribution of OpenVINO™ toolkit

CheXRad helps radiologists and physicians identify COVID-19, viral pneumonia and other diseases on chest X-ray images, and predict the need for ventilators.

- *CheXRad* comes pre-configured with a COVID-19 and viral pneumonia classification neural network.
- To architect, train and validate the neural network, Accrad used **Intel Tensorflow from AI Analytics Toolkit** and the **Intel oneAPI DevCloud** to develop the model.
- To optimize its model for deployment, Accrad used **OpenVINO™ toolkit** and **Intel® DevCloud for Edge**.
- *CheXRad* could classify pathologies in 140 chest x-rays in just **90 seconds** —up to **160x faster** than radiologists, at comparable levels of accuracy, sensitivity and specificity.

Ground Truth Class: 0 (non-COVID-19)
Predicted Class: 0 (non-COVID-19)
Prediction probabilities: ['1.00', '0.00']



Learn more in this [solution brief](#)

Key Takeaways & Call to Action

- Intel toolkits are **FREE**, complementary & work seamlessly together
- They help achieve performance & efficiency across different stages of AI Journey
- Recommend the toolkits based on current phase of customer pipeline

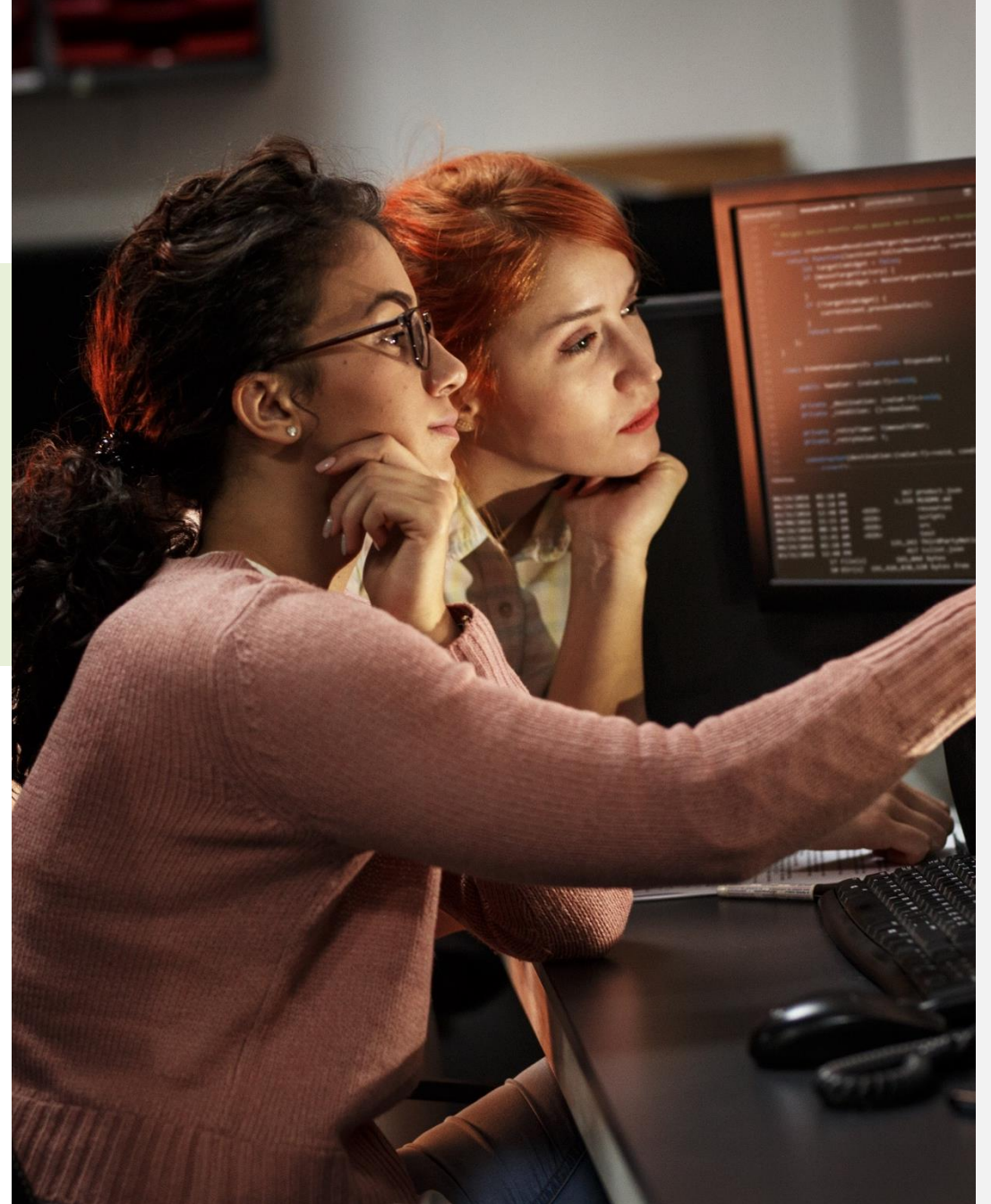
Download the toolkits

[Intel® oneAPI AI Analytics Toolkit](#)

[Intel® Distribution of OpenVINO™ toolkit](#)

[Intel® oneAPI Base Toolkit](#)

Learn more about [Intel® oneAPI Toolkits](#)
intel.com/oneAPI-AllToolkits



intel®

Which Toolkit to Use When ?

	Intel® oneAPI AI Analytics Toolkit	Intel® Distribution of OpenVINO™ toolkit
Key Value/ Benefits	<ul style="list-style-type: none"> Provides performance and easy integration across end-to-end data science pipeline for efficient AI model development Maximum compatibility with open source FWKs and Libs with drop-in acceleration that requires minimal to no code changes 	<ul style="list-style-type: none"> Provides high performance and efficiency for DL inference solutions to deploy across Intel XPU architectures (cloud to edge) Optimized package size for deployment based on memory requirements
Users	Data Scientists, AI Researchers, DL/ML Developers	AI Application Developers, Media and Vision Developers
Use Cases	<ul style="list-style-type: none"> Data Ingestion, data pre-processing, ETL operations Model training and inference Scaling to multi-core / multi-nodes / clusters 	<ul style="list-style-type: none"> Inference applications for vision, speech, text, NLP Media streaming / encode, decode Scale across hardware architectures – edge, cloud, datacenter, device
Hardware Support	<ul style="list-style-type: none"> CPUs – Data center, server, workstation segments – Intel® Xeon® and Core™ processors Future Intel Xe GPUs – Artix Sound/Ponte Vecchio 	<ul style="list-style-type: none"> CPU – Intel Xeon, Core and Atom processors GPU – Intel® Processor Graphics (integrated), Intel® Iris® Xe Max Graphics, Future Intel Xe architecture Artix Sound/Ponte Vecchio VPU - NCS & Intel® Vision Accelerator Design Products FPGA - Intel® Arria® 10 FPGA GNA - Intel® Gaussian & Neural Accelerator
Low Precision Support	<p>Use Intel® Low Precision Optimization Tool when using the Intel oneAPI AI Analytics Toolkit</p> <ul style="list-style-type: none"> Supports BF16 for training and FP16, Int8 and BF16 for Inference Seamlessly integrates with Intel optimized frameworks Available in the AI toolkit and independently 	<p>Use Post Training Optimization Tool when using OpenVINO</p> <ul style="list-style-type: none"> Supports FP16, Int8 and BF16 for inference Directly works with Intermediate Representation Format Available in the Intel Distribution of OpenVINO toolkit Provides Training extension via NNCF for PyTorch with FP16, Int8

Exception: If a model is not supported by OpenVINO™ toolkit for inference deployment, build custom layers; or fall back to use the Intel oneAPI AI Analytics Toolkit and use optimized DL frameworks for inference.

BackUp

Intel® oneAPI Base Toolkit

Accelerate Data-centric Workloads

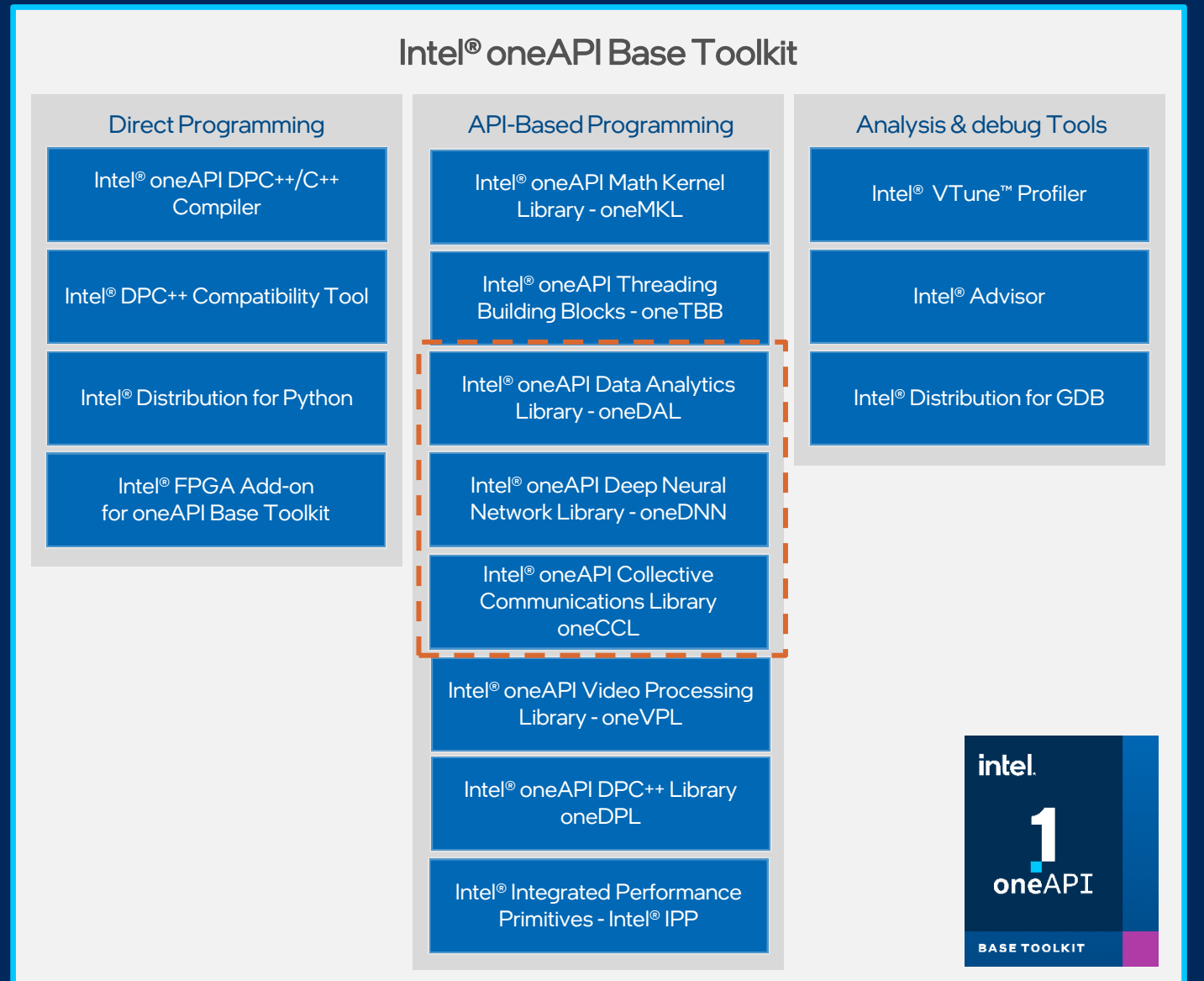
A set of core tools and libraries for developing high-performance applications on Intel® CPUs, GPUs, and FPGAs

Who Uses It?

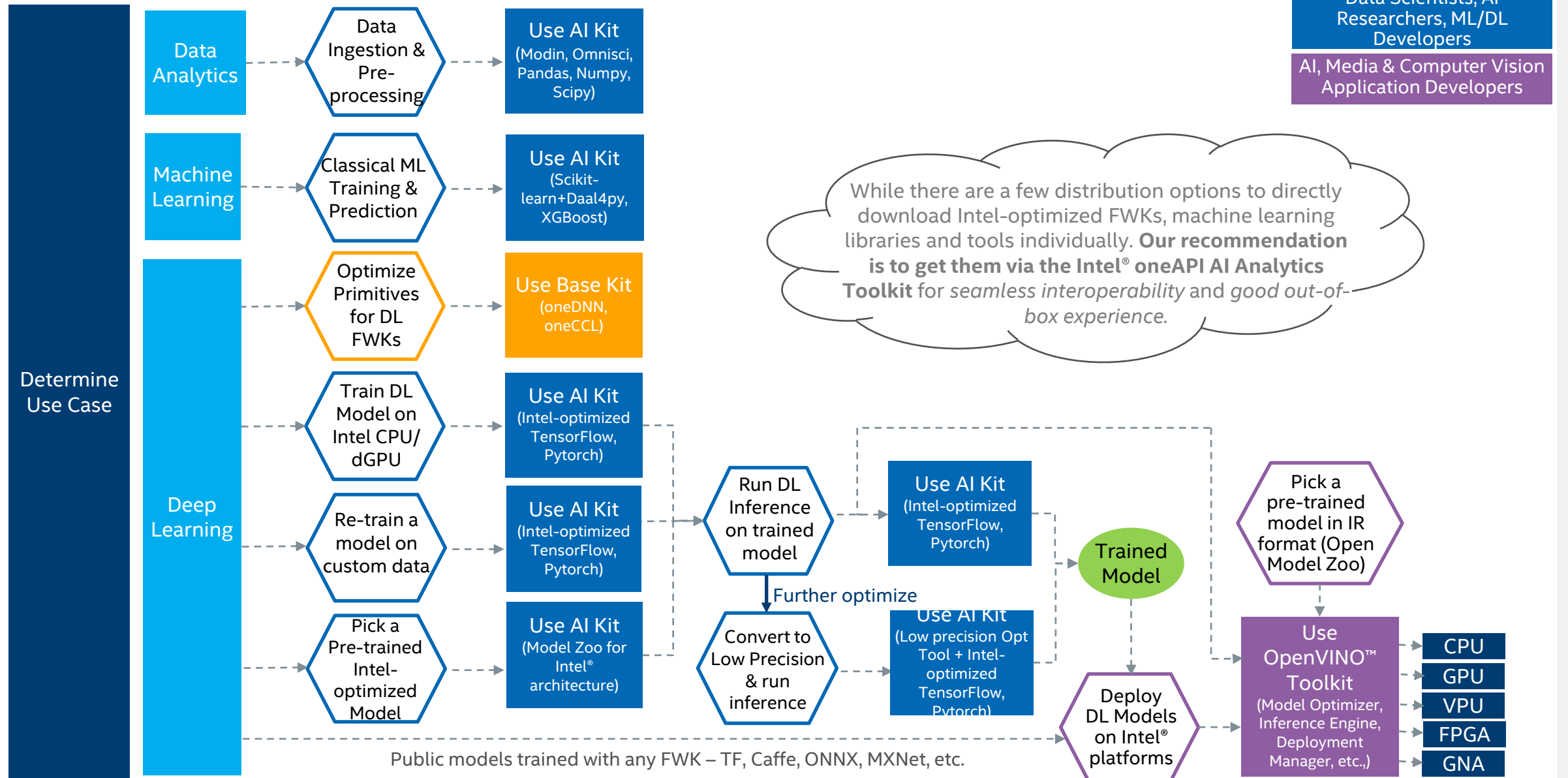
- A broad range of developers across industries
- Native Code Developers/Framework Developers

Top Features/Benefits

- Data Parallel C++ (DPC++) compiler, library and analysis tools; DPC++ Compatibility tool helps migrate existing code written in CUDA
- Optimized performance libraries for threading, math, data analytics, deep learning, and video/image/signal processing



AI Development Workflow

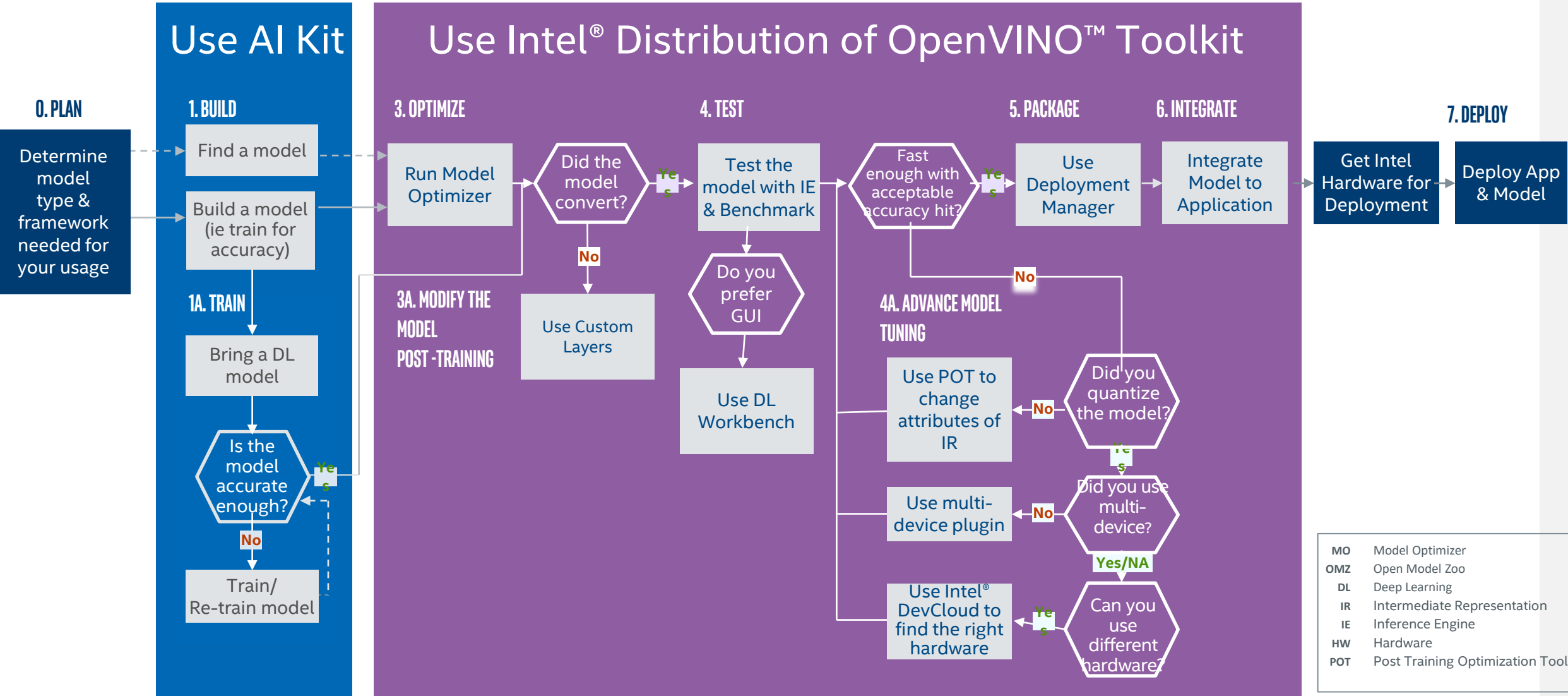


AI Kit = Intel® oneAPI AI Analytics Toolkit
 Base Kit = Intel® oneAPI Base Toolkit

AI Model Deployment Workflow

Data Scientists, AI Researchers,
ML/DL Developers

AI, Media & Computer Vision
Application Developers



A comprehensive workflow to optimize your DL model for the Intel Hardware that will be used for running inference

Notices & Disclaimers

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. 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 does not control or audit third-party data. You should consult other sources to evaluate accuracy.

© Intel Corporation. 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.

Configurations

Deep Learning Training and Inference Performance using Intel® Optimization for PyTorch with 3rd Gen Intel® Xeon® Scalable Processors

ResNet50/ResNext101 (FP32/BF16): batch size 128/instance, 4 instances.

ResNet50/ResNext101 dataset (FP32/BF16): [ImageNet Dataset](#)

DLRM batch size (FP32/BF16): 2K/instance, 1 instance

DLRM dataset (FP32/BF16): [Criteo Terabyte Dataset](#)

DLRM batch size (INT8): 16/instance, 28 instances, dummy data.

Tested by Intel as of 6/2/2020.

Intel® Xeon® Platinum 8380H Processor, 4 socket, 28 cores HT On Turbo ON Total Memory 768 GB (24 slots/ 32GB/ 3200 MHz), BIOS: WLYDCRB1.SYS.0015.P96.2005070242

(ucode: 0x700001b), Ubuntu 20.04 LTS, kernel 5.4.0-29-generic

PyTorch: <https://github.com/pytorch/pytorch.git>

Intel Extension for PyTorch: <https://github.com/intel/intel-extension-for-pytorch.git>

gcc: 8.4.0,

Intel® oneAPI Deep Neural Network Library (oneDNN) version: v1.4

ResNet50: <https://github.com/intel/optimized-models/tree/master/pytorch/ResNet50>

ResNext101 32x4d: https://github.com/intel/optimized-models/tree/master/pytorch/ResNext101_32x4d

DLRM: <https://github.com/intel/optimized-models/tree/master/pytorch/dlrm>

Inference Throughput FP32 vs Int8 optimized by Intel® Optimization for Tensorflow and Intel® Low Precision Optimization Tool (part of the Intel® oneAPI AI Analytics Toolkit)

Tested by Intel as of : 10/26/2020: TensorFlow v2.2 (<https://github.com/Intel-tensorflow/tensorflow/tree/v2.2.0>); Compiler: GCC 7.2.1; DNNL(<https://github.com/oneapi-src/oneDNN>) v1.2.0 75d0b1a7f3586c212e37acebbb8acd221cee7216; Dataset: ImageNet/Coco/Dummy, refer to each model README; Precision: FP32 and Int8

Platform: Intel® Xeon® Platinum 8280 CPU; #Nodes: 1; #Sockets: 2; Cores/socket: 28; Threads/socket: 56; HT: On; Turbo: On; BIOS version:

SE5C620.86B.02.01.0010.010620200716; System DDR Mem Config: 12 slots / 16GB / 2933; OS: CentOS Linux 7.8; Kernel: 4.4.240-1.el7.elrepo.x86_64

Stock scikit-learn vs Intel-optimized scikit-learn

Testing by Intel as of 10/23/2020. Intel® oneAPI Data Analytics Library 2021.1 (oneDAL), scikit-learn 0.23.1, Intel® Distribution for Python 3.8; Intel® Xeon® Platinum 8280LCPU @ 2.70GHz, 2Sockets, 28 cores per socket, 10M samples, 10 features, 100 clusters, 100 iterations, float32

XGBoost CPU vs GPU

Test configs: Tested by Intel as of 10/13/2020;

CPU: c5.18xlarge AWS Instance (2 x Intel® Xeon Platinum 8124M @ 18 cores, OS: Ubuntu 20.04.2 LTS, 193 GB RAM. GPU: p3.2xlarge AWS Instance (GPU: NVIDIA Tesla V100 16GB, 8 vCPUs), OS: Ubuntu 18.04.2 LTS, 61 GB RAM. SW: XGBoost 1.1:build from sources. compiler – G++ 7.4, nvcc 9.1. Intel® Data Analytics Acceleration Library (Intel® DAAL): 2019.4 version; Python env: Python 3.6, Numpy 1.16.4, Pandas 0.25, Scikit-learn 0.21.2.

XGBoost fit CPU acceleration

Test configs: Tested by Intel as of 10/13/2020; c5.24xlarge AWS Instance, CLX 8275 @ 3.0GHz, 2 sockets, 24 cores per socket, HT:on, DRAM (12 slots / 32GB / 2933 MHz); SW: XGBoost 0.81, 0.9, 1.0 and 1.1:build from sources. compiler – G++ 7.4, nvcc 9.1. Intel® DAAL: 2019.4 version; Python env: Python 3.6, Numpy 1.16.4, Pandas 0.25, Scikit-learn 0.21.2.

End-to-End Census Workload Performance

Tested by Intel as of 10/15/2020. 2x Intel® Xeon® Platinum 8280 @ 28cores, OS: Ubuntu 19.10.5.3.0-64-generic Mitigated, 384GB RAM. SW: Modin 0.8.1, scikit-learn 0.22.2, Pandas 1.0.1, Python 3.8.5, Daal4Py 2020.2 Census Data, (21721922, 45). Dataset is from IPUMS USA, University of Minnesota, www.ipums.org. Version 10.0.

Tiger Lake + Intel® Distribution of OpenVINO™ toolkit vs Coffee Lake CPU

System Board	Intel prototype, TGL U DDR4 SODIMM RVP	ASUSTeK COMPUTER INC. / PRIME Z370-A
CPU	11 th Gen Intel® Core™ -5-1145G7E @ 2.6 GHz.	8 th Gen Intel® Core™ i5-8500T @ 3.0 GHz.
Sockets / Physical cores	1 / 4	1 / 6
HyperThreading / Turbo Setting	Enabled / On	Na / On
Memory	2 x 8198 MB 3200 MT/s DDR4	2 x 16384 MB 2667 MT/s DDR4
OS	Ubuntu* 18.04 LTS	Ubuntu* 18.04 LTS
Kernel	5.8.0-050800-generic	5.3.0-24-generic
Software	Intel® Distribution of OpenVINO™ toolkit 2021.1.075	Intel® Distribution of OpenVINO™ toolkit 2021.1.075
BIOS	Intel TGLIFUI1.R00.3243.A04.2006302148	AMI, version 2401
BIOS release date	Release Date: 06/30/2020	7/12/2019
BIOS Setting	Load default settings	Load default settings, set XMP to 2667
Test Date	9/9/2020	9/9/2020
Precision and Batch Size	CPU: INT8, GPU: FP16-INT8, batch size: 1	CPU: INT8, GPU: FP16-INT8, batch size: 1
Number of Inference Requests	4	6
Number of Execution Streams	4	6
Power (TDP Link)	<u>28 W</u>	<u>35W</u>