Configure your Research Platform
Infrastructure Needs for Embedded and Heterogeneous Computing

Heterogeneous Computing Cluster for Deep Learning Workloads

April 16, 2018

Dr. Yassine Hariri, CMC Microsystems, Hariri@cmc.ca

Agenda

- A quick recap of Artificial Intelligence applications and Deep Learning
- Heterogeneous Computing for Deep Learning Workloads
- CMC planned heterogeneous computing infrastructure
- Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access

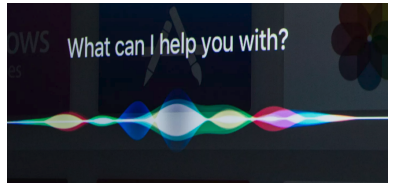
Agenda

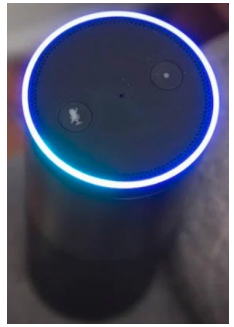
- ***A quick recap of Artificial Intelligence applications and Deep Learning***
- Heterogeneous Computing for Deep Learning Workloads
- CMC planned heterogeneous computing infrastructure
- Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access

AI: Area of Specialization

- Transforming almost every business
- Exploding ecosystem of tools, making it more accessible to even non-experts

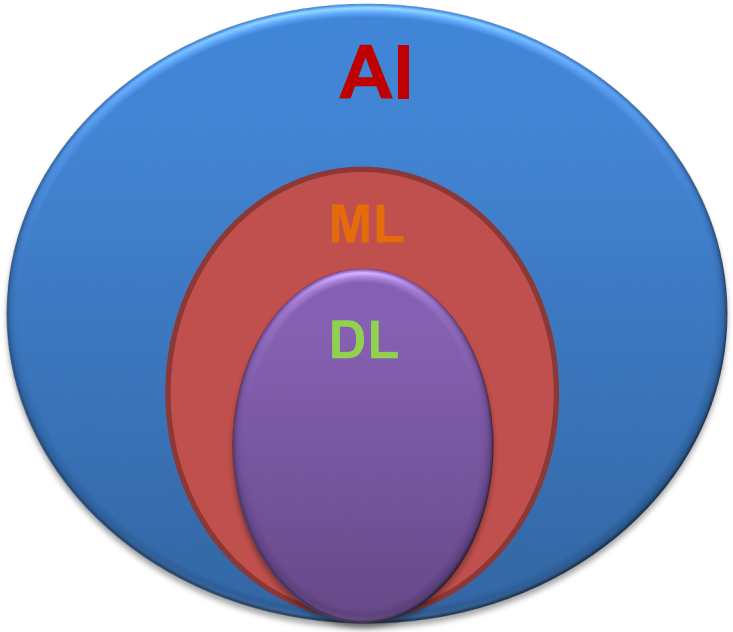
- **Area of Specialization**
  - **Gaming**
  - **Natural Langage Processing**
  - **Computer Vision**
    - Robotics
    - Autonomous Cars
  - **…**





AI and Machine Learning

- AI: The theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages. –*Source oxfordreference.com*



- **AI: Artificial Intelligence**
  - Sense, reason, act and adapt
- ML: Machine Learning
  - Algorithms that improve as they are exposed to data over time
- DL: Deep Learning
  - Multilayered neural networks learn from vast amounts of data
- **DL Training:**
  Using a set of training sample data to determine the optimal weights of the artificial neurons in a DNN.
- **DL Inference:**
  - Analyzing specific data using a previously trained DNN.

Source: What's the Difference Between Artificial Intelligence (AI), Machine Learning, and Deep Learning? by Glenn Evan Touger

- After a neural **network** is trained, it is deployed to run **inference**:
  - to classify, recognize, and process new inputs.

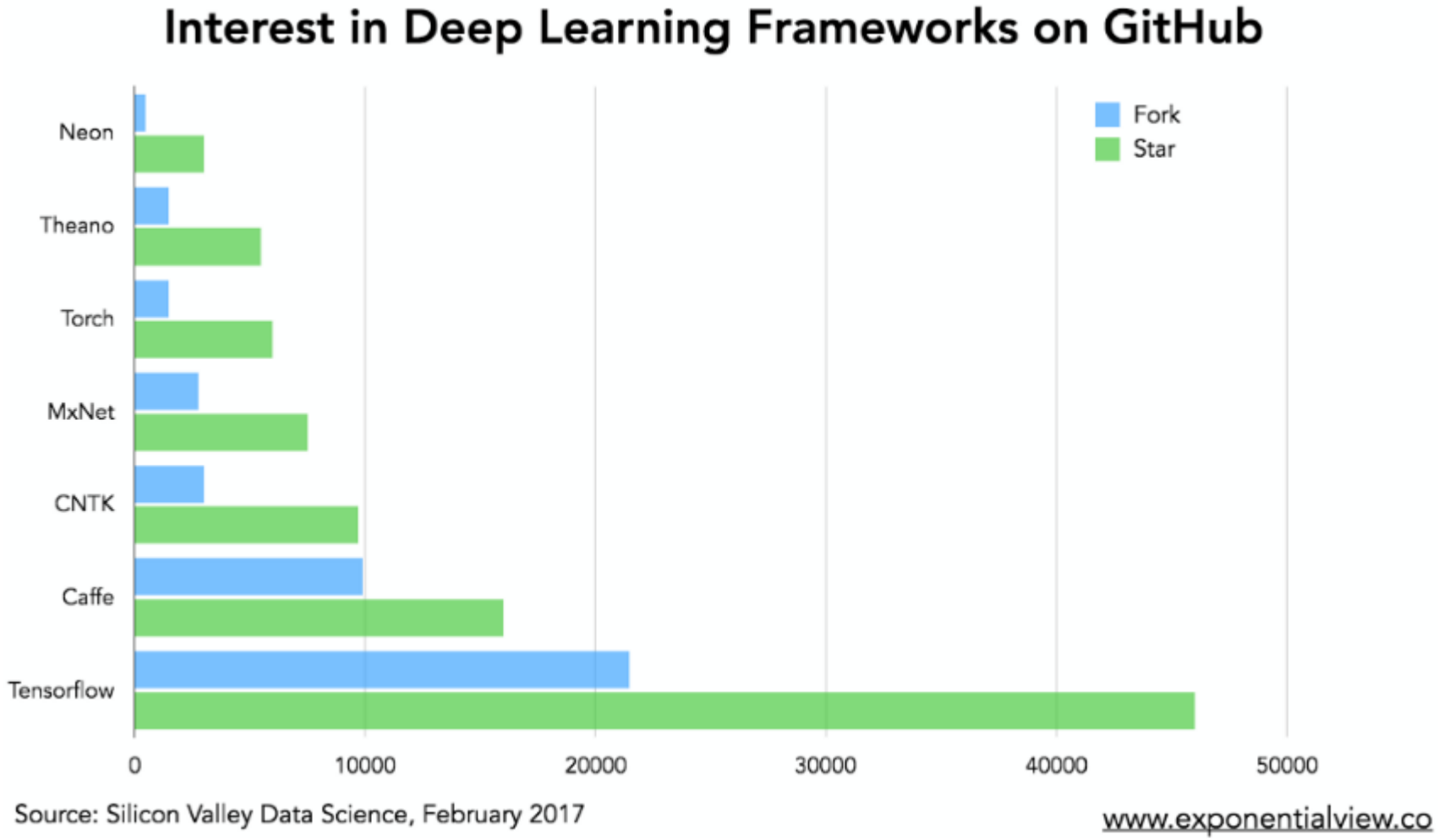
Rise in popularity of deep learning

- **Key enablers:**
  - **Greater availability of large data sets**, containing more training examples
  - **Efficient use of accelerators** such as GPUs, FPGAs and custom hardware such as Tensor Processor to train deep learning models
  - **Open source** machine learning flow, as well as libraries

Software for Deep Learning

- **DNN training is very computationally intensive:**
  - Basic Linear Algebra Subprograms (BLAS), GEMV and GEMM routines, Conv2D operations,
  - batch normalization, SGD, SVD, element wise matrix operations, and non-linear functions such as Softmax and ReLU.
- **Availability** of these building blocks along with **an efficient data transfer mechanism** between CPU, GPU and FPGAs is **essential** for the efficient implementation of these algorithms.
  - **Deep-Learning frameworks available as open source : Caffe, TensorFlow…**
  - **Middleware libraries from different vendors**:
    - AMD: MIOpen
    - NVIDIA: cuDNN
    - Xilinx: DNN (xfDNN) and BLAS (xfBLAS)
    - Intel FPGA: mxnet, caffee …

Interest in Deep Learning Framework on GitHub



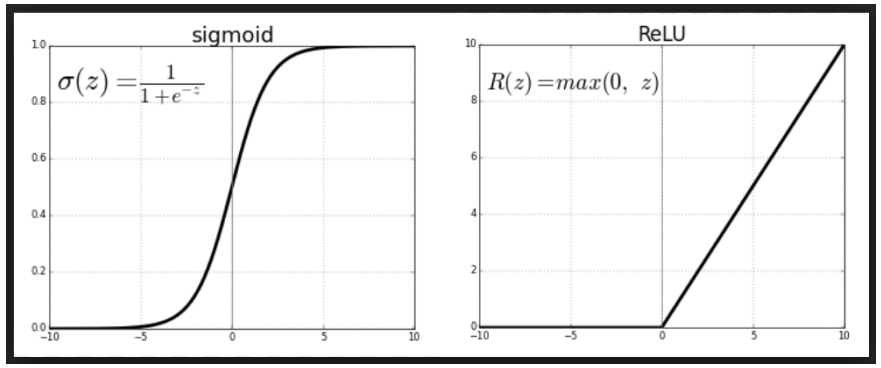
How it works?

- Computation is expressed as a dataflow graph

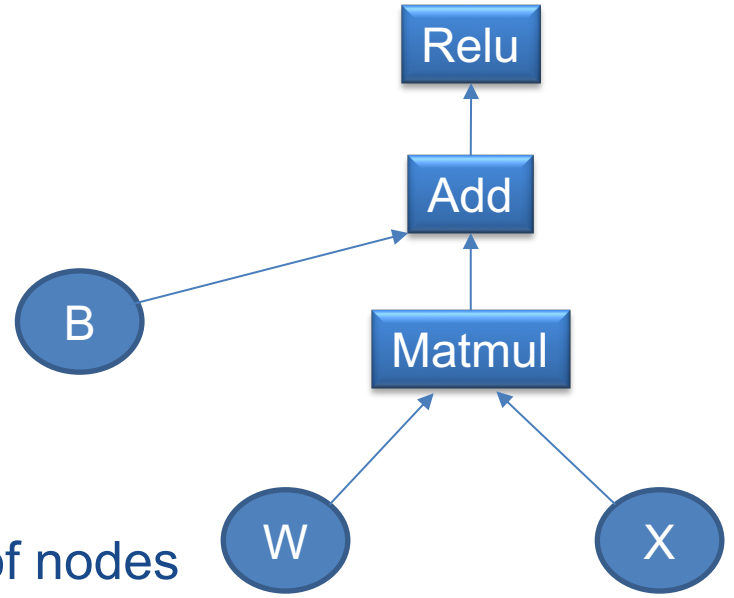
Tensor

5.0	2.2	-1.4	…
2.5	0.1	3	…
…	…	…	…
…	…	…	…

Rectified Linear Units
Output=0 (input<0)
Output=input (input>=0



Computational Graph



TensorFlow programs can be divided to two sections:

- The computational graph: series of operations arranged into a graph of nodes
- Session to run the Computational graph

Agenda

- A quick recap of Artificial Intelligence applications and Deep Learning
- ***Heterogeneous Computing for Deep Learning Workloads***
- CMC planned heterogeneous computing infrastructure
- Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access

Need for Heterogeneous Computing

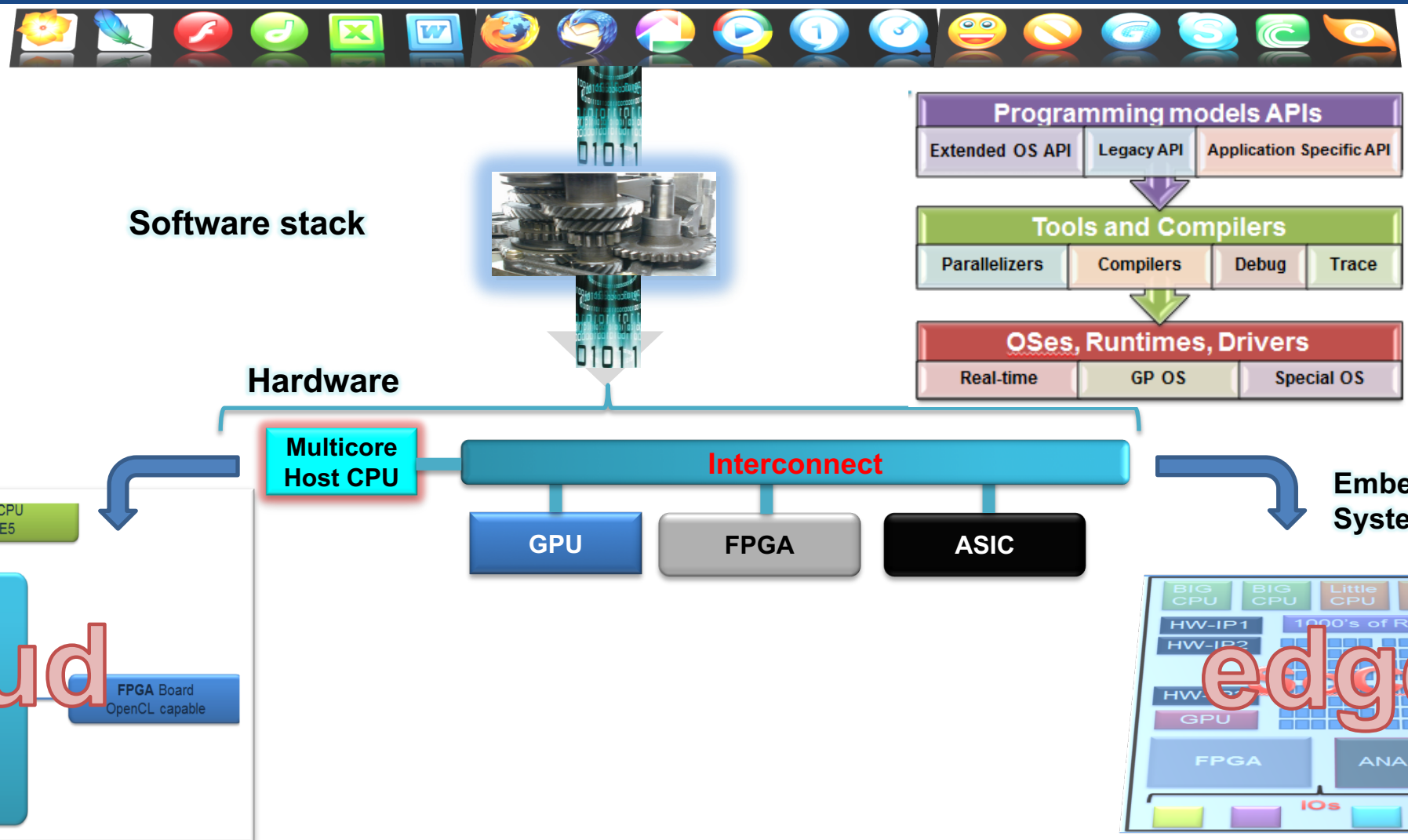
- **The end of frequency scaling as the dominant cause of processor performance gains has caused an industry-wide shift**
  - **Power Wall** (unable to scale power)
  - **Instruction-Level Parallelism Wall** (can't extract enough parallelism from a single instruction stream)
  - **Memory Wall** (cant keep up with the processor)
- **Solution:**
  - Multicore CPUs
  - Heterogeneous computing

Need for Heterogeneous Computing

- **Applications**
  - Control (searching, parsing, etc..)
  - Data intensive (ML, image/video processing, Computer vision, data mining, etc…)
  - Compute intensive (ML, iterative methods, quantum simulation, etc...)

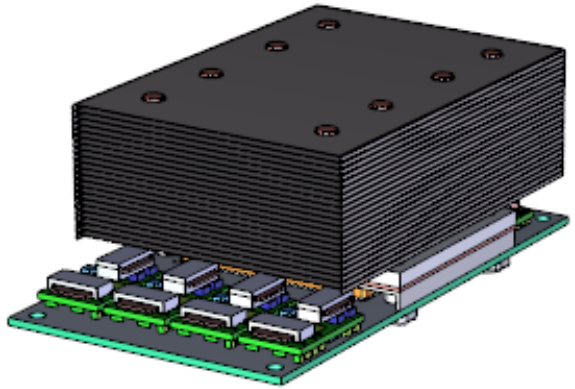
➢ **Gain performance by offloading the right workload to the right processor architecture.**

Heterogeneous Systems Architecture



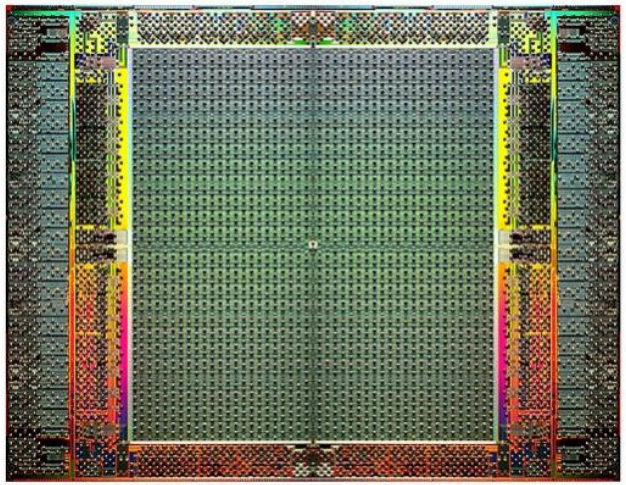
GPUs

- Good
  - GPGPU, Easy to Program, SIMD
  - Massive floating point computational power
  - Lots of APIs, Cheap for developers (gaming)
  - Available in Clouds
- Bad
  - Super power hungry
  - Expensive in Datacenters
- Ugly
  - CUDA is proprietary
  - Open languages lagging (OpenCL, OpenACC, Ope…… ,



FPGAs

- Good
  - Ultimate power efficiency, Massive parallelism
  - Multiple instructions on single data (MISD)
  - Great at leveraging/accelerating storage and networking
  - Great at replacing ASICs
  - OpenCL and High-Level Synthesis are helping to make them more programmable
- Bad
  - Difficult to find in Clouds
  - Lots of IP available, but rarely is it free
  - Cheaper dev kits exist, but they are weak in performance capability
- Ugly
  - Compile & synthesis time takes hours



Summary of CPU, GPU, and FPGA comparison

Feature	Analysis	Winner
DNN Training	GPU floating point capabilities are greater	GPU
DNN Inference	FPGA can be customized, and has lower latency	FPGA
Large data analysis	CPUs support largest memory and storage capacities. FPGAs are good for inline processing.	CPU/FPGA
Timing latency	Algorithms implemented on FPGAs provide deterministic timing, can be an order of magnitude faster than GPUs	FPGA
Processing/Watt	Customized designs can be optimal	FPGA
Processing/$$	GPUs win because of large processing capabilities. FPGA configurability enables use in a broader acceleration space.	GPU/FPGA
Interfaces	FPGA can implement many different interfaces	FPGA
Backward compatibility	CPUs have more stable architecture than GPUs. Migrating RTL to new FPGAs requires some work.	CPU
Ease of change	CPUs and GPUs provide an easier path to changes to application functionality.	GPU/CPU
Customization	FPGAs provide broader flexibility	FPGA
Size	CPU and FPGA's lower power consumptions leads to smaller volume solutions	CPU/FPGA
Development	CPUs are easier to program than GPUs, both easier than FPGA	CPU

Source: "*Unified Deep Learning with CPU, GPU, and FPGA Technologies*
Allen Rush1, Ashish Sirasao2, Mike Ignatowski1

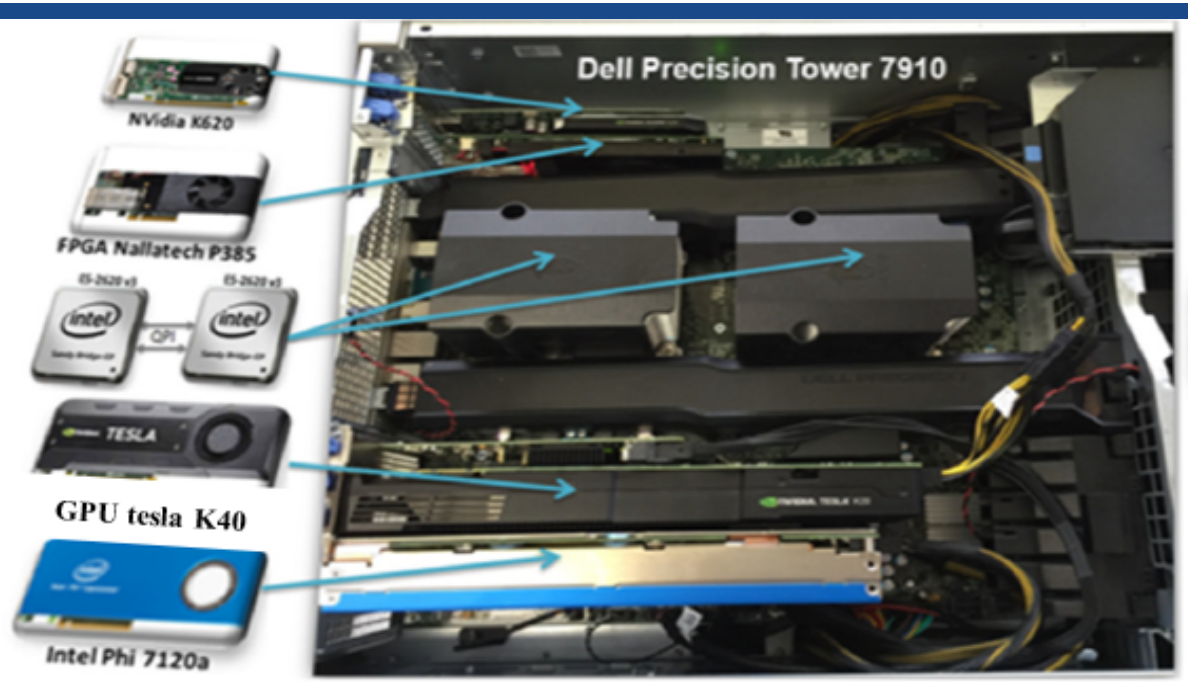
Unified Deep Learning Configurations and Emerging Applications

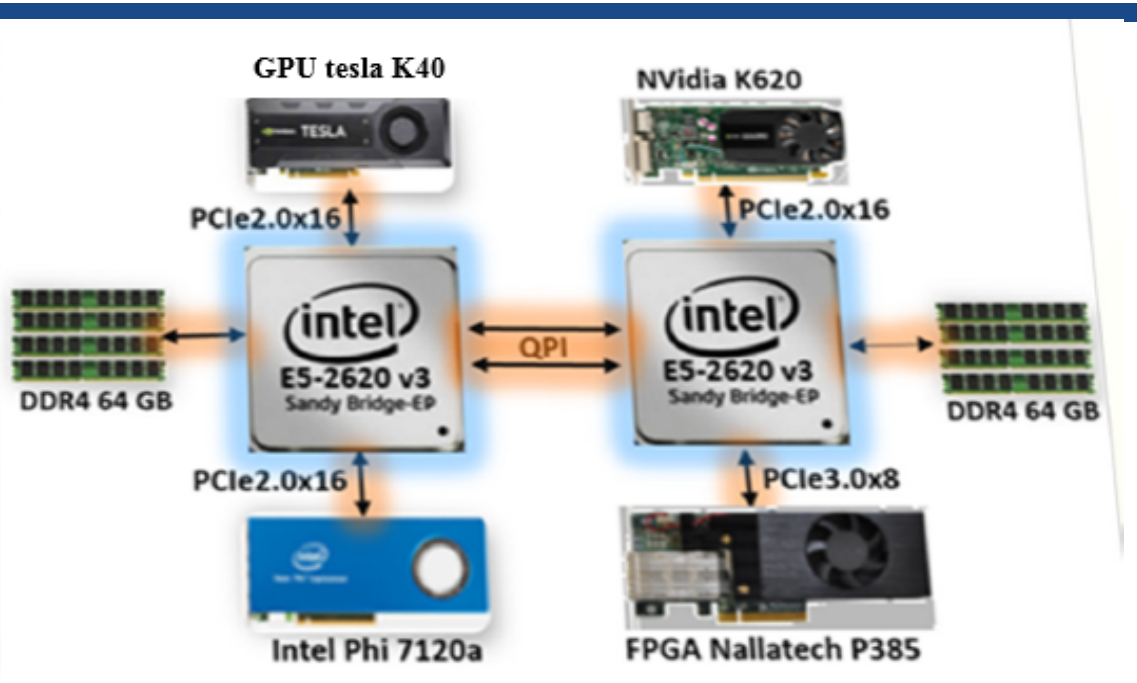
- **Complementary strengths** of CPUs, GPUs, and FPGAs for different types of deep learning operations.
- **A unified deep learning platform** that simultaneously employs GPUs for training and FPGAs for inference

Agenda

- A quick recap of Artificial Intelligence applications and Deep Learning
- Heterogeneous Computing for Deep Learning Workloads
- ***CMC planned heterogeneous computing infrastructure***
- Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access

CMC Heterogeneous Systems





Accelerator	Features	Host Interface	Compute Performance	Power
Nallatech 385	▪ Altera Stratix V ▪ Memory: 2 banks of 4 GB	PCIe 3.0 x 8	Unavailable	Typical application ≤ 25 W
TESLA K40	▪ 2880 CUDA cores ▪ Memory: 12 GB at 288 GB/s	PCIe 2.0 x 16	4.29 TFLOPS (single precision) 1.43 TFLOPs (double precision)	225 W
Xeon Phi 7120a	▪ 61 Cores, 1.33 GHz ▪ Memory: 16 GB at 352 Gb/s	PCIe 2.0 x 16	Peak Double Precision 1.003 TFLOPs	300 W

Supported platforms

- **SAP**: Simulation Acceleration platform; CPU + FPGA
- **MPA**: Multiprocessor Array Platform; CPU + GPU or Xeon Phi
- **HPP**: Heterogeneous Processing Platform; MPA + SAP

The HPP Resources

- **Deployment** (32 system, ~60 users)
  - Assembled, cloned, tested and shipped
- **Tutorials and reference designs**
  - Quick start guide : Heterogeneous Processing Platform (HPP) **Available online**
  - User Guide: Performance and Power profiling for the HPP **Available online**
  - Computer vision using OpenCV/OpenCL targeting the HPP- Heterogeneous Processing Platform **Available online**
  - Deep Learning TensorFlow targeting the HPP-Heterogeneous Processing Platform
- **Webinars series for the HPP**
  - Introduction to the HPP-Heterogeneous Parallel Platform: A combination of Multicores, GPUs, FPGAs and Many-cores accelerators (August 26th) **Available online**
  - Programming models, performance and power profiling for the HPP-Heterogeneous Processing Platform (December 2nd) Available online **Available online**
  - Computer vision using OpenCV/OpenCL targeting the HPP- Heterogeneous Processing Platform (January 13th) **Available online**
  - Deep Learning TensorFlow targeting the HPP-Heterogeneous Processing Platform

32 HPP Deployed, ~60 Users

Institutions	#	Dev. Sys. Coord.	HPP end users	Research
CMC	1	NA	Yassine and Remote access users	Installation, setup and Testing
Western	2	Robert Sobot	Abdallah Shami, Robert Sobot,	Parallel processing of optical FDTD and FEM design
U of Ottawa	1	Miodrag Bolic	Miodrag Bolic	Optimizing a Virtualized Base Station
McGill	1	Zeljko Zilic	Warren Gross, Gross and Mey[illegible]	Parallel algorithms and architectures for belief propagation
UQO	1	Ahmed Lakhssassi	Ahmed LAKHSSASSI	Embedded Distributed sensors array for thermal peak monitoring
York	1	Magierowski	Sebastian Ma[illegible]ski	Bioinformatics and DNA sequencing.
RMC	1	Rachid Beguenane	Mos[illegible]	Beamforming schemes for cognitive (LS MU-MIMO) systems
UQTR	2	Adel Omar Dahmane	Dahm[illegible], Adel-Omar	Acceleration platform for virtual reality applied to health issues
Windsor	1	Rashid Rashidzadeh	Mohammed khalid	Using OpenCL for accelerating computationally intensive tasks
Ryerson	4	Fei Yuan Fyuan	Fei Yuan, Andy G. Ye, Gul N. Khan	video processing applications targeting HPP
Uvic	1	Nikitas Dimopoulos	Nikitas Dimopoulos	Parallel architecture, neural networks, and power aware systems
Wilfrid Laurier	2	Hongbing Fan	Hongbing Fan	Reconfigurable computing with applications in grid/cloud computing.
Sasktachewan	1	Seok-Bum Ko	Seok-Bum Ko	Efficient Implementation of Advanced Encryption Standards

Gen. 1

Poly	**6**	**Yvon Savaria**	**Yvon Savaria**	**Cluster for AI**
poly	1	Yvon Savaria	Jérome Le-Ny & David Saussié & Guchuan Zhu	
poly	1	Yvon Savaria	Sylvain Martel	
poly	1	Yvon Savaria	Jean-Pierre David	
poly	1	Yvon Savaria	Mohamad [illegible]awc	
Concordia	1	Ted Obuchowicz	Ted Obucho[illegible]	
McMaster	1	Nicola Nicolici	Nicola Nicolici	
Memorial	1	Lihong Zhang	Lihong Zhang	
UofT	2	Paul Chow	Paul Chow	

Gen. 2

HPP Cluster at Polytechnique Montreal

Cluster of HPPs

- 1GB Ethernet to connect the hosts for management
- 10 Gb Ethernet switch to connect the hosts
- 40 Gb Ethernet switch to connect the FPGAs

SW stack

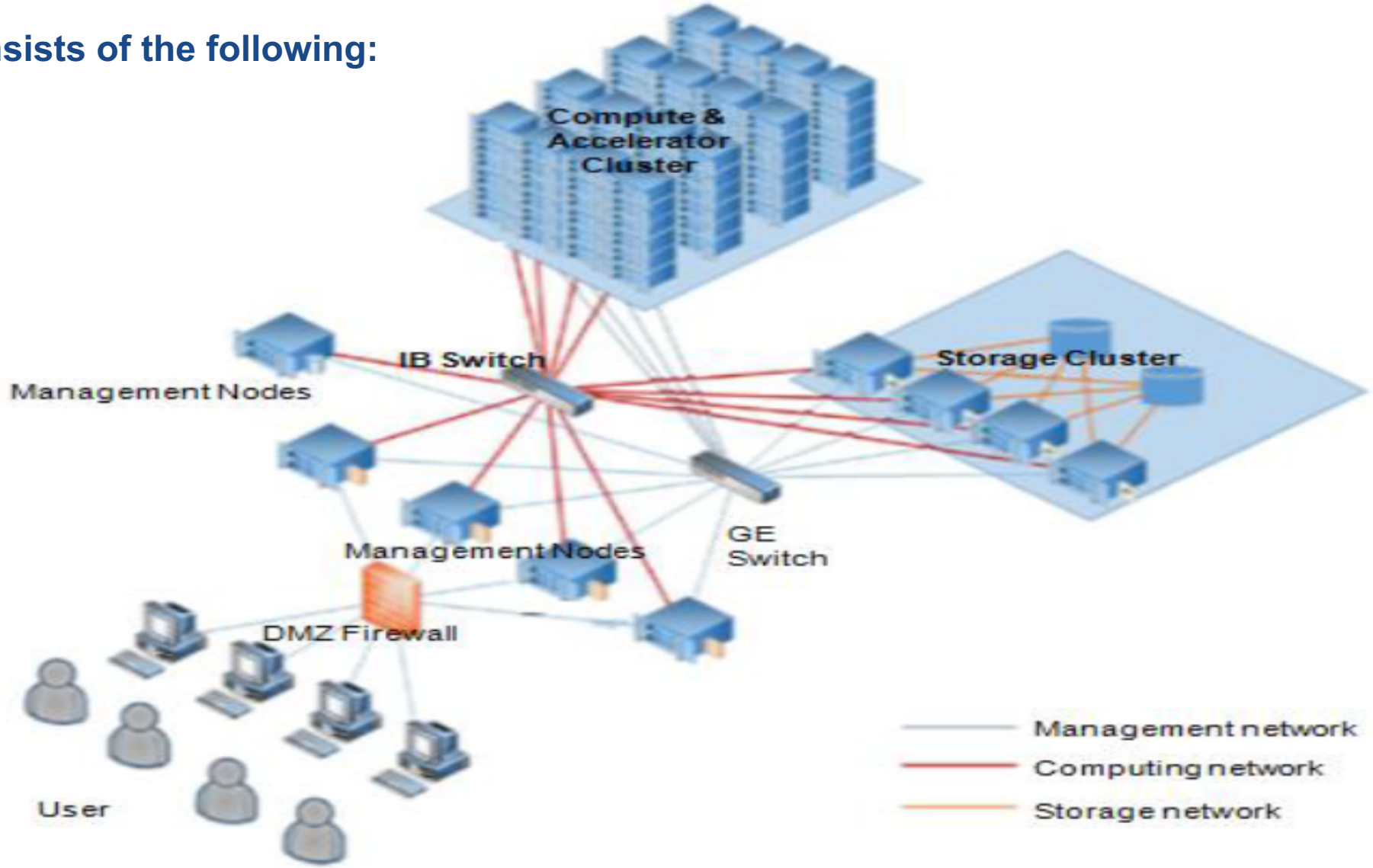
- MPI, OpenMP, OpenCL, CUDA

Applications

- Machine learning and AI
- 5G..

Proposed Compute, Accelerator and Storage Cluster Solution Overview

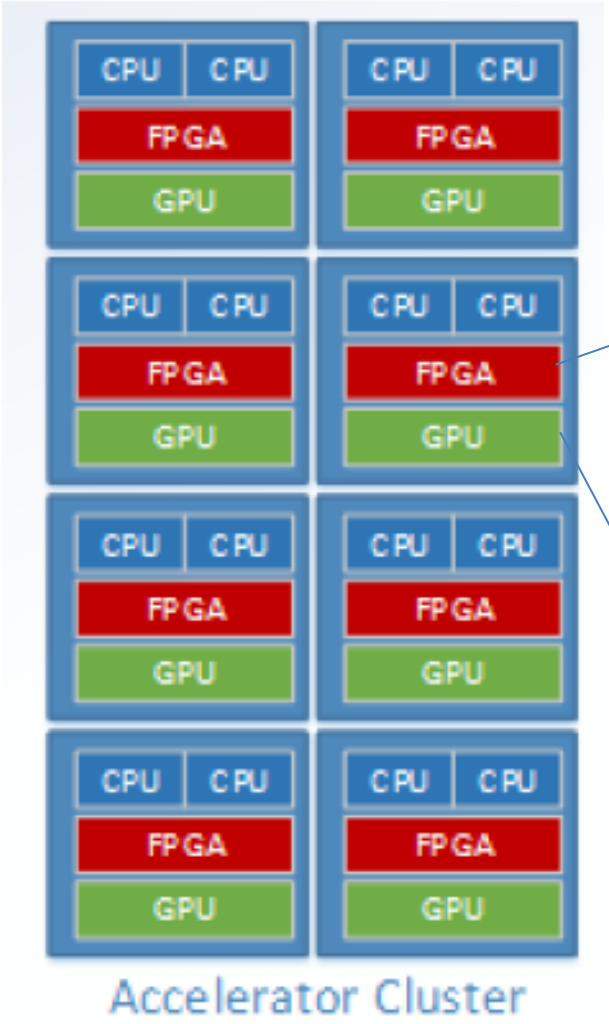
- Supercomputing Cluster consists of the following:
  - Compute Cluster
  - Accelerator Cluster
  - Storage



Heterogeneous Computing Cluster

An Accelerator Cluster

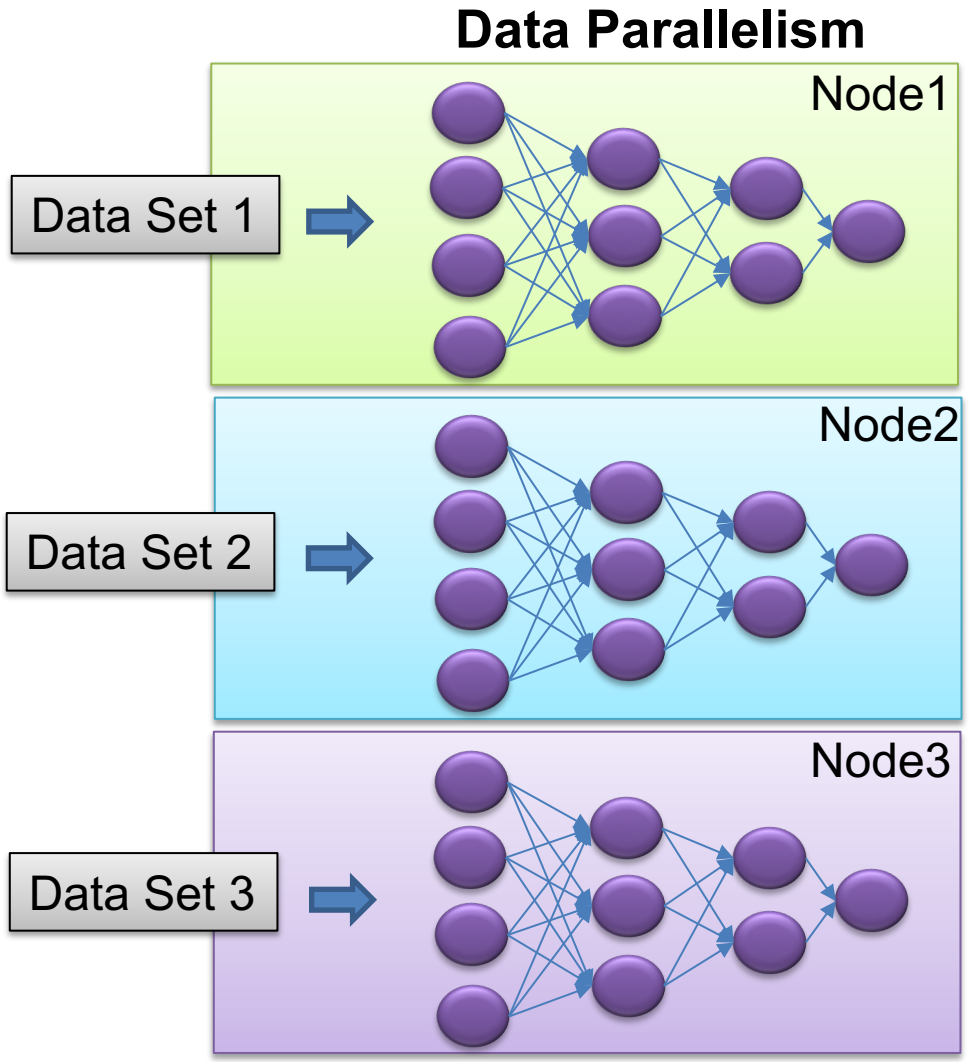
- ~8 Accelerator Nodes to Start
- Each Accelerator Node to have 1 x **P100 GPU** and 1 x Xilinx vcu1525 FPGA SDAccel board to support a more flexible workload
- CPU need not be of high core count, 10 core CPU like the E5-2630v4 will do
- Appropriate memory configuration to support the Accelerator Cluster

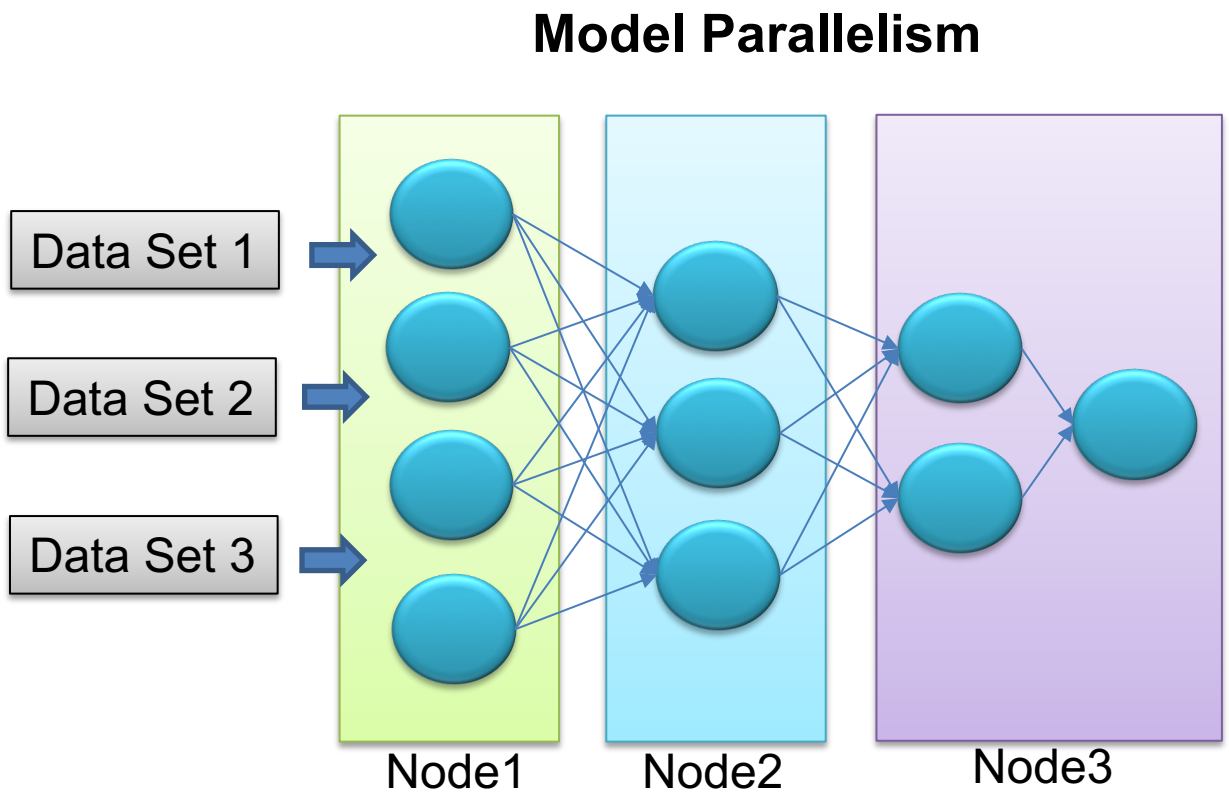


Xilinx vcu1525

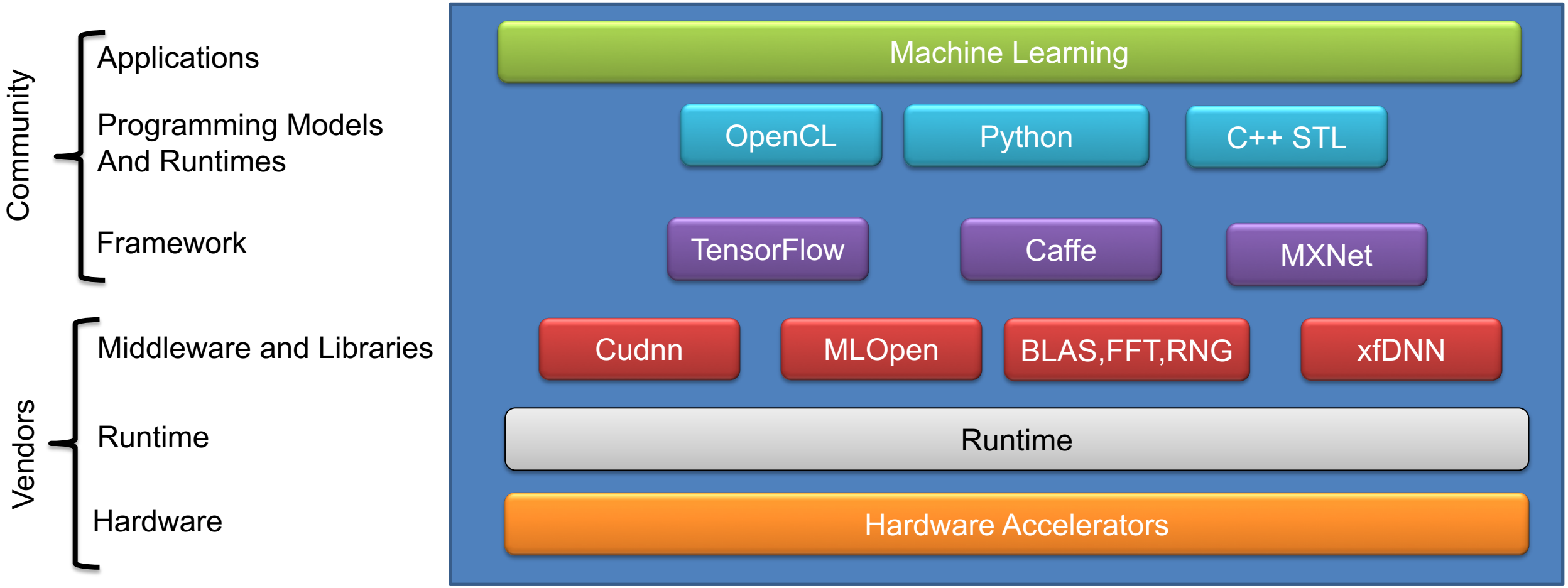
P100

Scale-out for Training and Inference





Software stack for the HCC



Agenda

- A quick recap of Artificial Intelligence applications and Deep Learning
- Heterogeneous Computing for Deep Learning Workloads
- CMC planned heterogeneous computing infrastructure
- ***Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access***

Open Discussion: Configure the Heterogeneous Processing Cluster for You to Access

We're Seeking Your Feedback on the Concept and Architecture:

- **What's the best configuration of FPGA and GPUs?**
- **What flexibility is needed?**
  - Hosted and managed by CMC as a cloud resource; accessible at your desktop
  - What architecture? How many nodes? Xilinx? Altera? NVIDIA? AMD? Tensor Processors?
  - Software stack for OpenCL + MPI heterogeneous cluster computing. Other software?
  - Data: what data bases should be available for ML?
- **What new research opportunities would it enable?**
  - Scale up opportunity for applications developed on single node, desktop systems?
  - Faster results: place and route engine to speed up the FPGA development cycle?
  - Tackling larger compute problems requiring machine learning, cloud computing?
  - Unified AI with Heterogeneous Computing Systems including: CPU, GPU, and FPGA Technologies