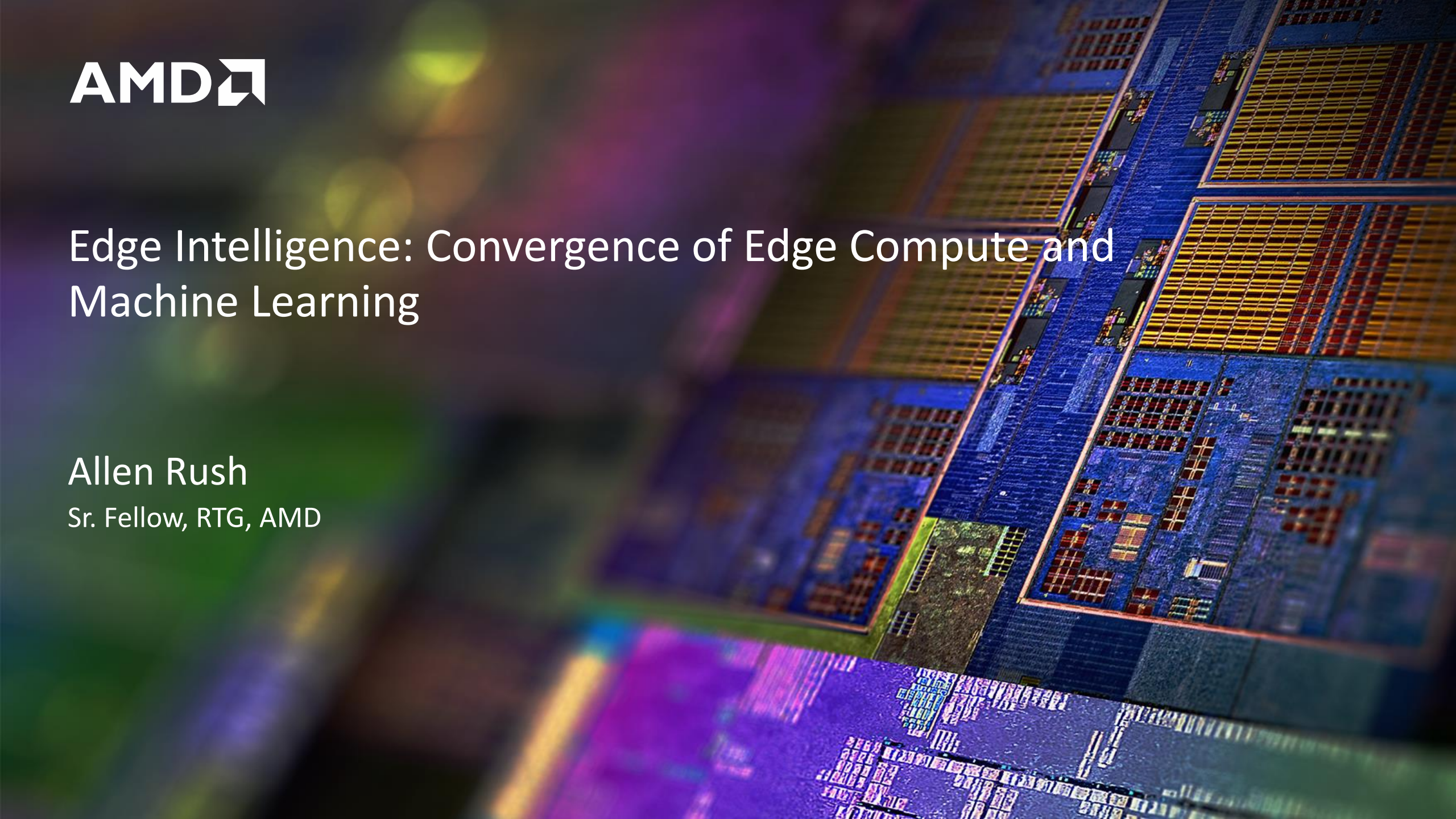


Edge Intelligence: Convergence of Edge Compute and Machine Learning

Allen Rush
Sr. Fellow, RTG, AMD

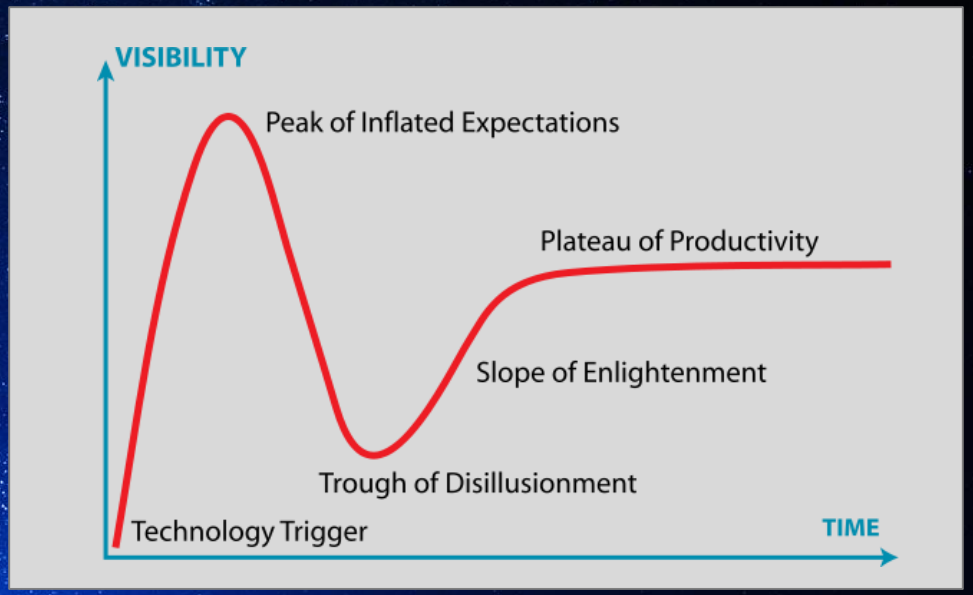
AGENDA

- Deep Learning Emerging as Major Application Focus
- Deep Learning Compute and Memory Drivers
- Training and Inference: Edge Intelligence and Machine Learning at the Edge
- SW, Acceleration, and Optimization
- AMD AI/ML Products and MIOpen

AMD

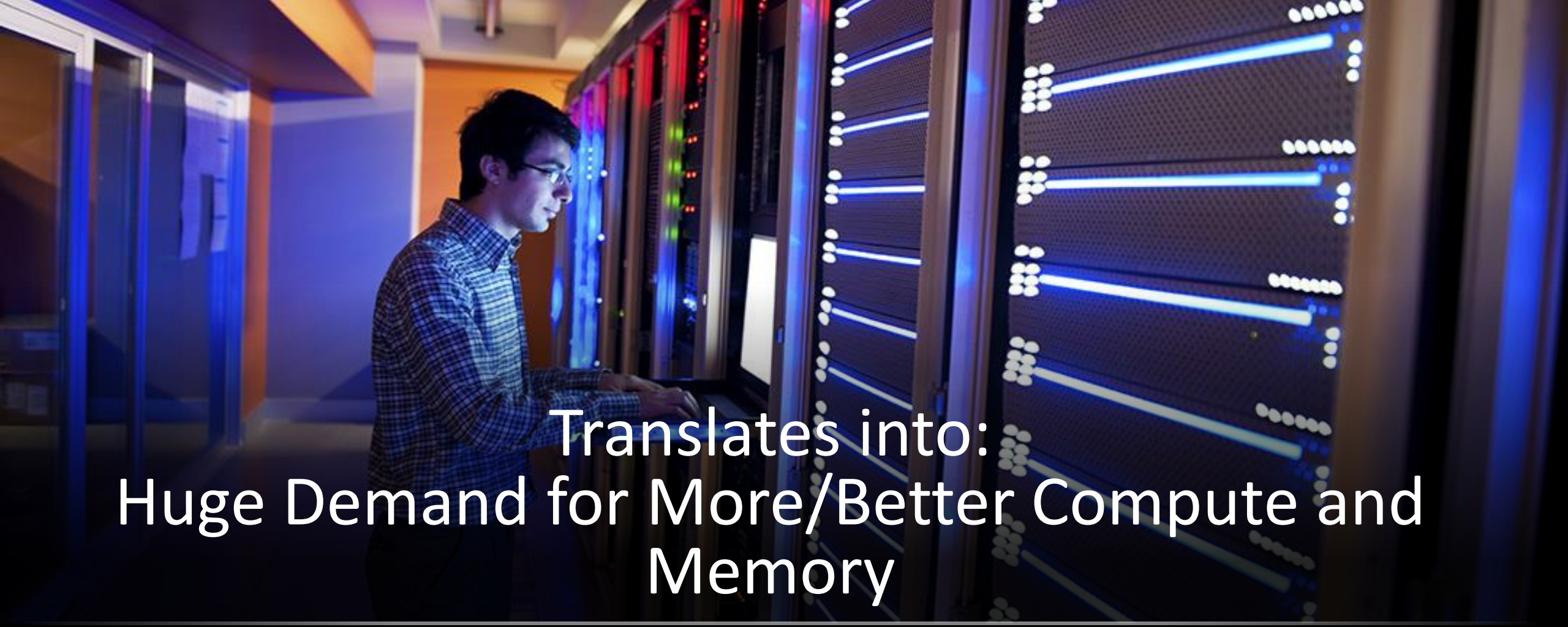
MACHINE LEARNING: REINVENTED TECHNOLOGY

- Factors supporting resurgence in ANN:
  - Big Data, large training set data bases
  - Much more powerful machines: GPU, CPU, FPGA…
  - Innovation in algorithms: DNN, CNN, DBN, RNN
- Modern NNs loosely based on neural network function of the brain
  - Also visual cortex, speech processing
- We are in year 5 of the enlightenment part of the Gartner Hype Cycle
  - Substantial investment in algorithms, HW, SW infrastructure, applications, and deployment



(Pent Up) Demand for Machine Learning Solutions

Internet and Cloud	Health Care	Media	Security and Defense	Automation
Image Classification	Cancer Cell Detection	Video Search	Face Detection	Factory Automation
Language Processing	Drug Development	Advertising	Video Surveillance	Autonomous Driving
Marketing and Advertising	Health Database Management	AR/VR	Object Detection, Classification	Investment Automation



Translates into: Huge Demand for More/Better Compute and Memory

Big Data Analytics

High Performance Computing

Machine Learning

Compute Infrastructure Today

Machine Intelligence Datacenter

Network Infrastructure

User

Homogenous processors

Open source software

Open interconnect

Some installations of

Proprietary accelerators

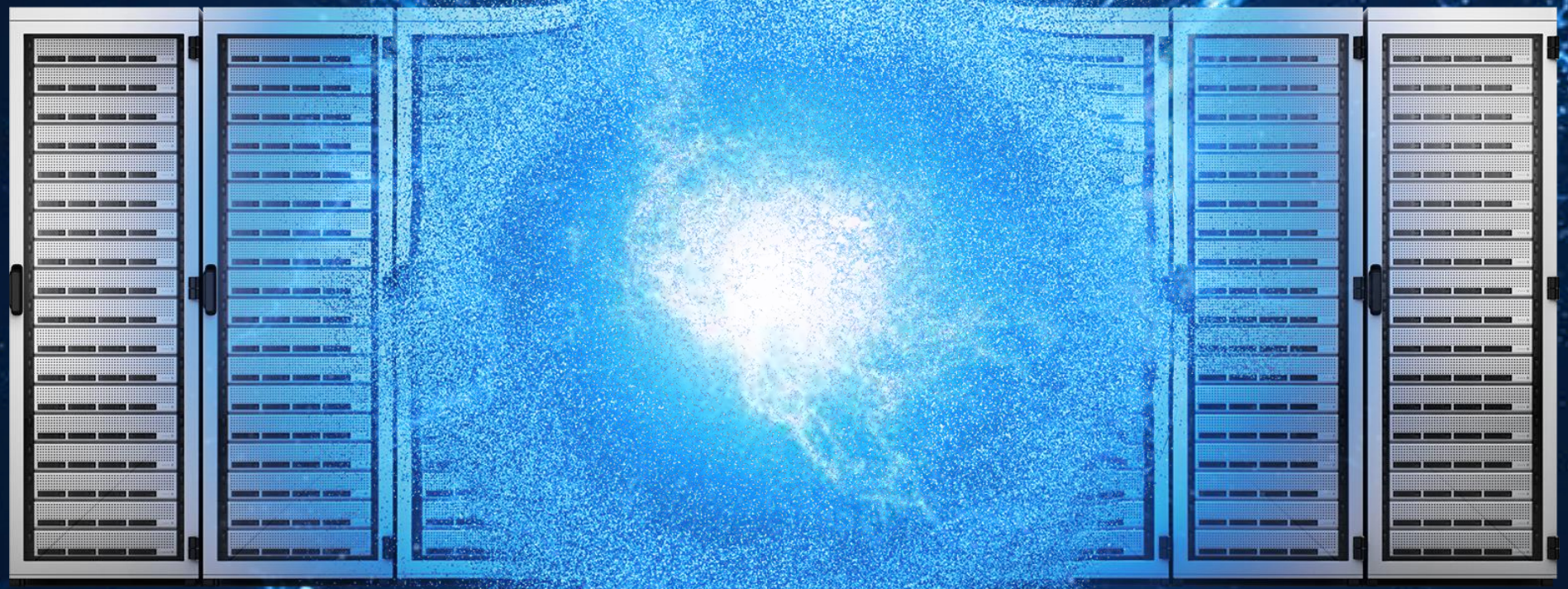
Proprietary accelerator software

Proprietary accelerator interconnect

Compute Infrastructure Tomorrow

- Heterogeneous processors
- Open source software
- Open interconnect
- Open accelerators

- Inference moving to the edge
- Optimized Computation for real time Inference
- Intelligent Updates and Hybrid Training



Big Datacenter Disruption

2.5 Quintillion Bytes of Data is Generated Every Day

500 million
Tweets

4 million
hours of content on YouTube

4.3 billion
Facebook entry

3.6 billion
Instagram

6 billion
Google searches

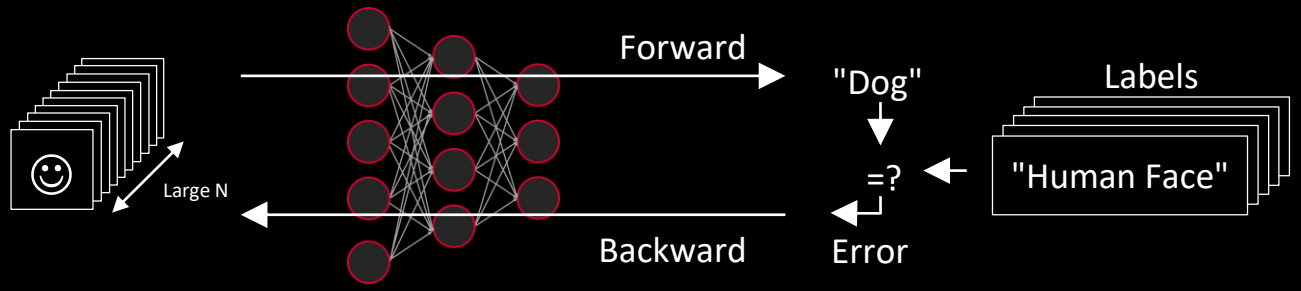
205 billion
emails

and many more…

Example Big Data Generators:

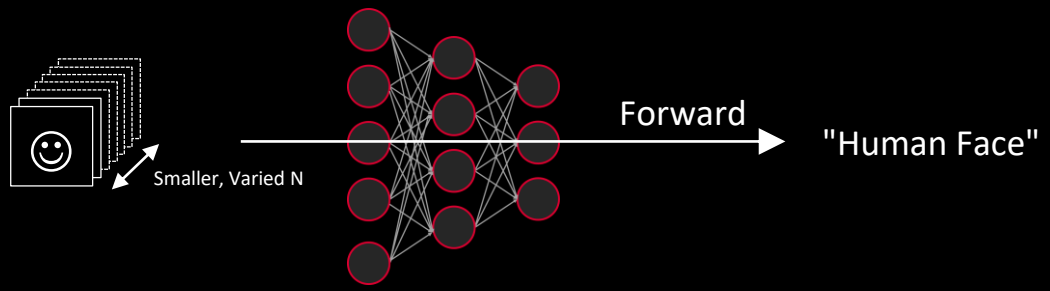
Platform	Data/day
Surveillance Camera	250Mb/sec->21.6 TB/day
Airplane (connected	5TB
Self Driving Car	4TB

Training vs Inference



Training

> Takes Hours, Days

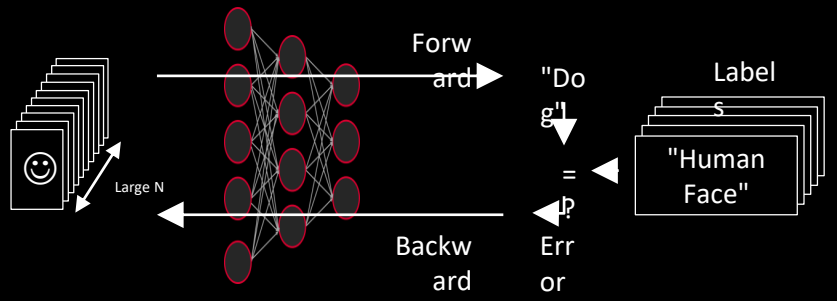


Inference

> Takes ms, seconds

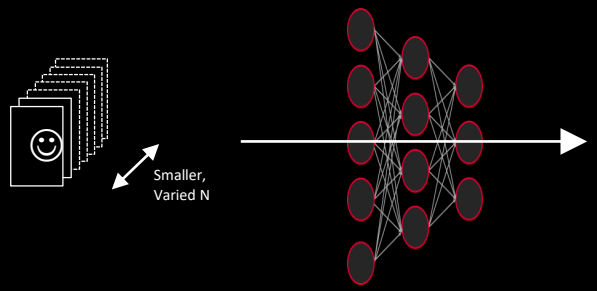
Single GPU Today	10-15 TFLOPS	500 GB/Sec

Training vs Inference – Transition to Edge



Training

> Takes Hours, Days

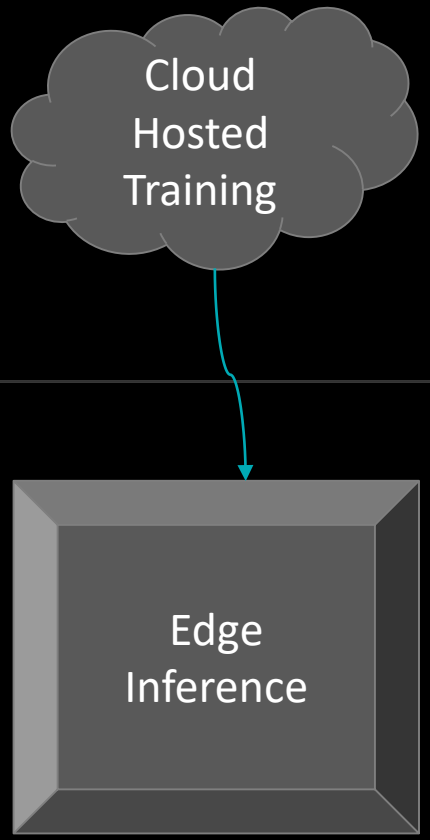


Forward

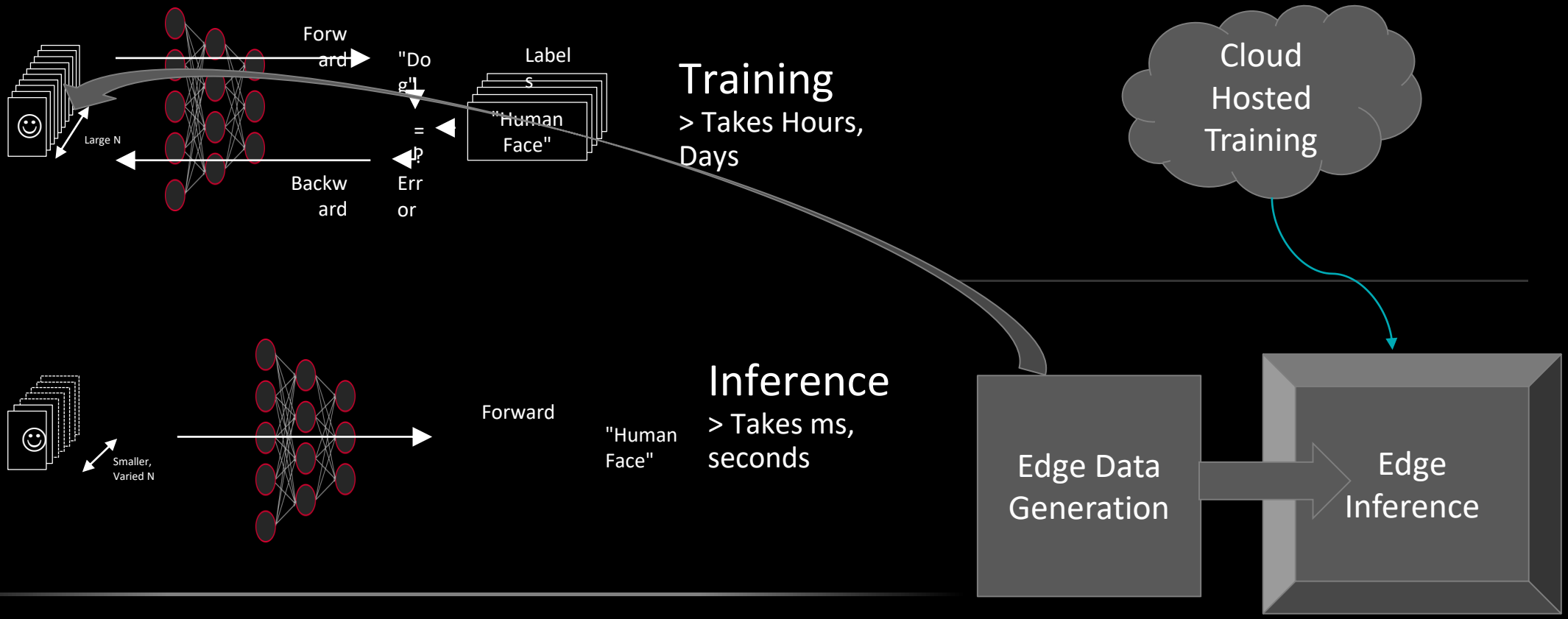
"Human Face"

Inference

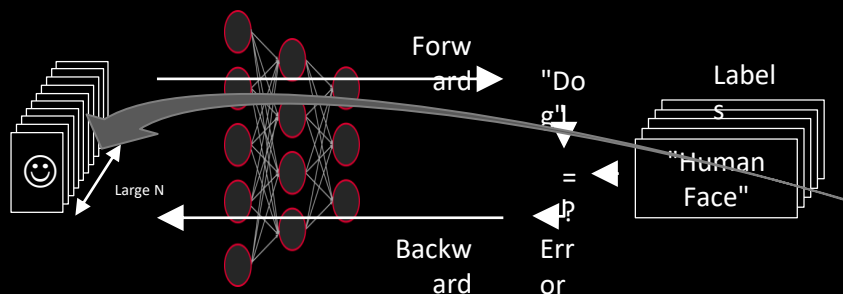
> Takes ms, seconds



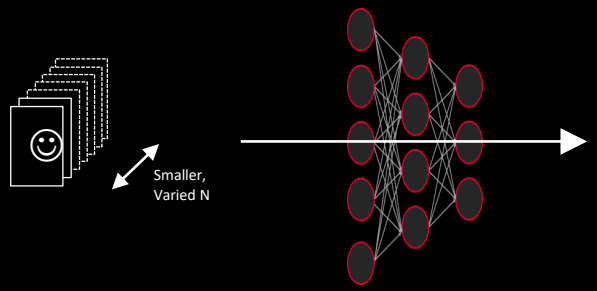
Edge Intelligence: Cloud Training



Edge Intelligence: Cloud Training, Edge Local Training



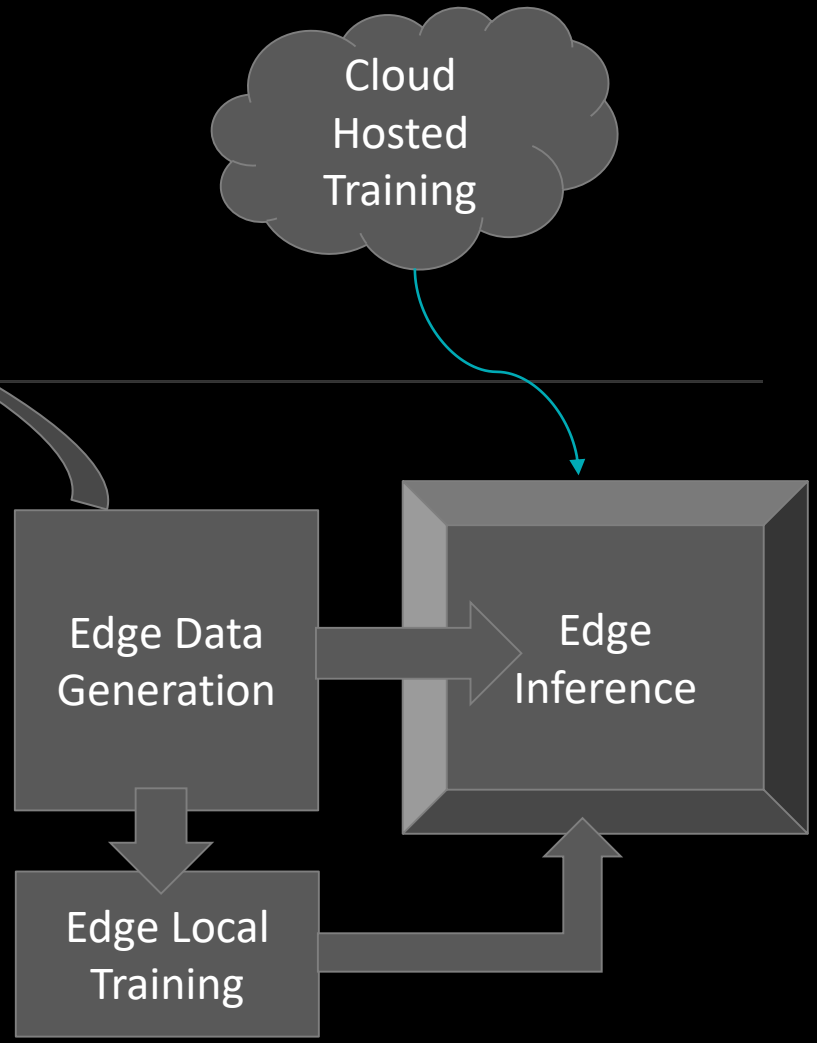
Training
> Takes Hours, Days



Forward

"Human Face"

Inference
> Takes ms, seconds



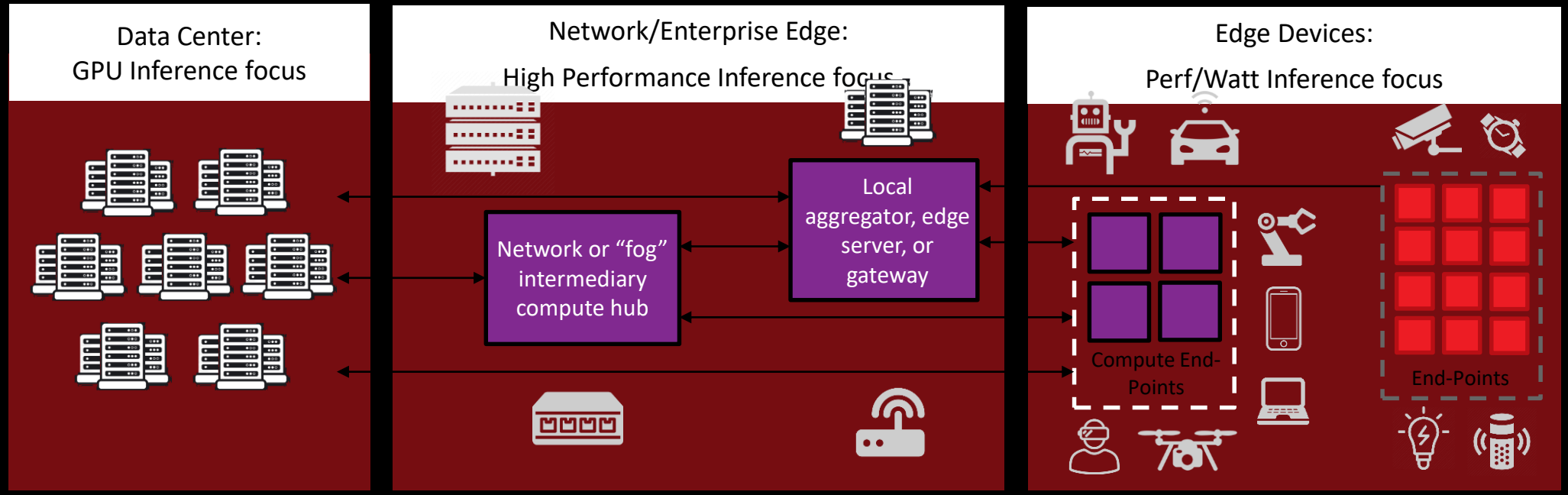
Edge Compute and Inference

Definitions differ, but we will define **Edge Compute** as anywhere that data is processed between the data center and “zero compute” IoT end-points.

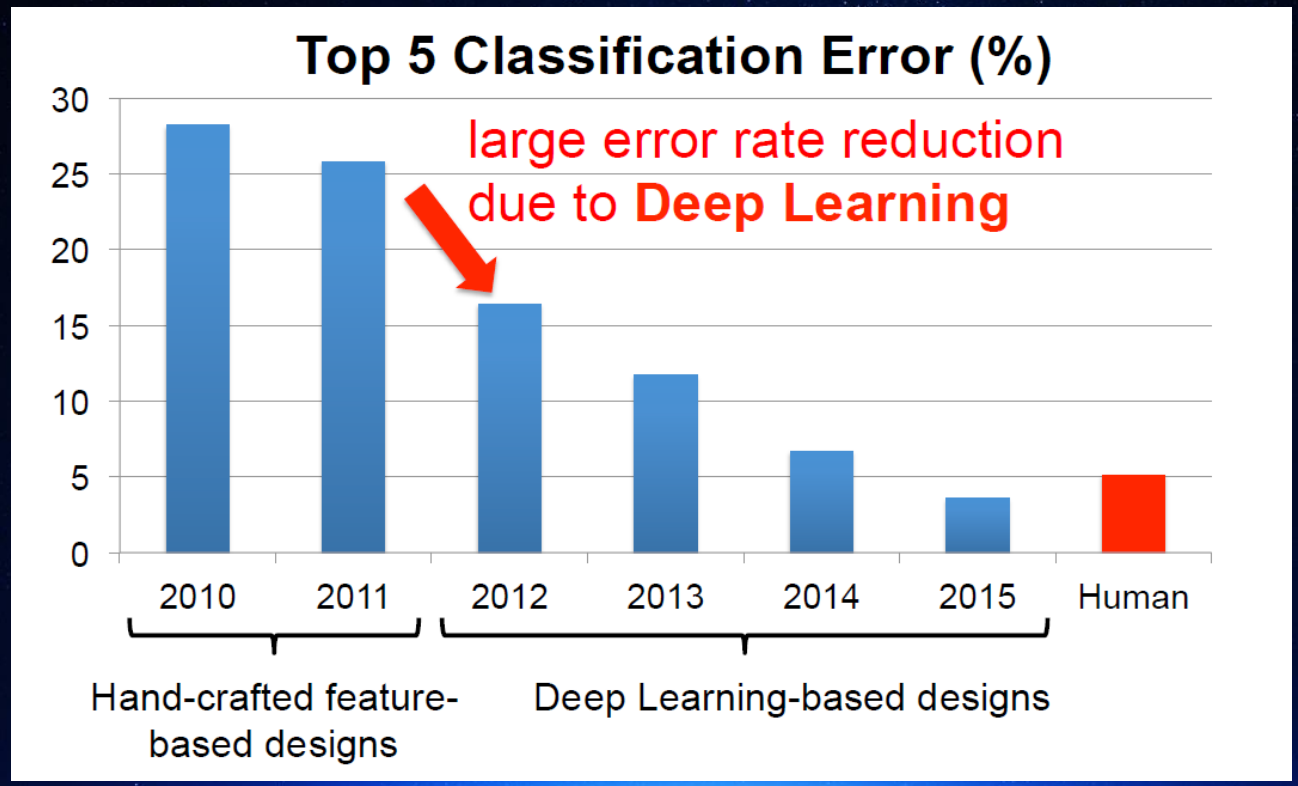
Network compute hubs and local edge server domains are sometimes referred to as **Fog Compute**.

The need for **Fog Compute** is driven by the explosion of data generated by connected devices on growing networks.

Moving the compute closer to the end-points mitigates bandwidth, latency and security issues



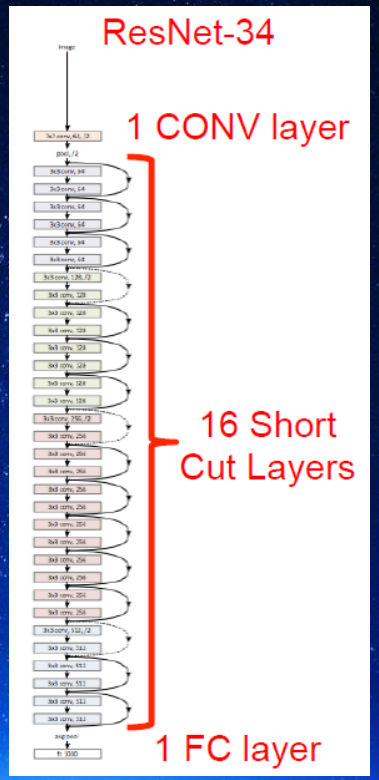
SIGNIFICANT IMPROVEMENT IN ACCURACY FOR IMAGE CLASSIFICATION



[Russakovsky et al., IJCV 2015]

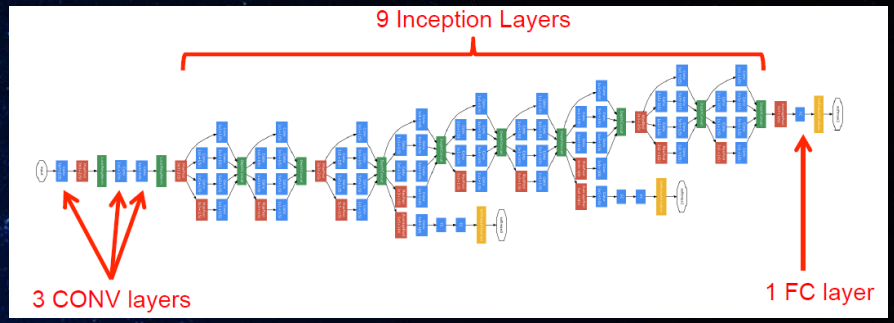
ADVANCED NETWORK ARCHITECTURES

- Compressed networks
- Residual networks
- Inception
- Fully Convolutional Networks (FCN)
- Sparse networks
- Reduced precision/binary networks
- Softmax approximation
- Adversarial NNs
- Spectral (FFT, Winograd) convolutions
- Recurrent NN + LSTM: persistent and temporal updates

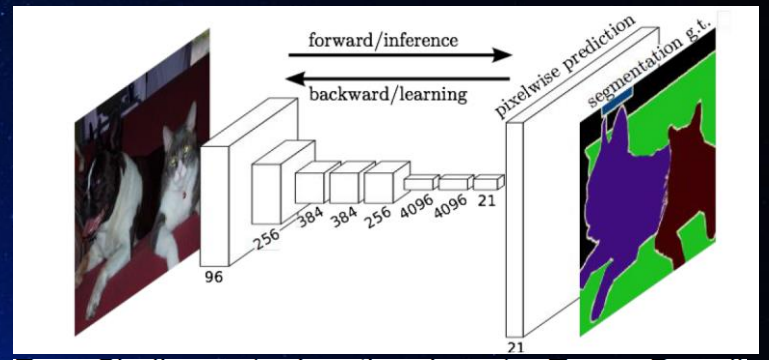


[He et al., arXiv 2015, CVPR 2016]

Googlenet



[Szegedy et al., arXiv 2014, CVPR 2015]



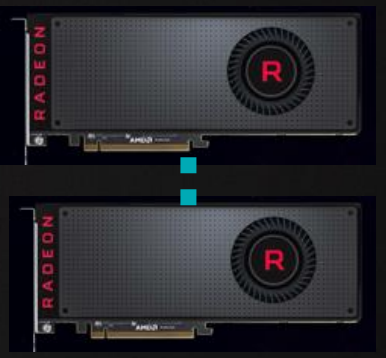
Evan Shelhamer* Jonathan Long* Trevor Darrell
UC Berkeley in CVPR'15, PAMI'16

AMD

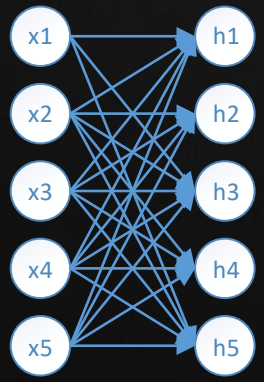
MACHINE LEARNING: END-TO-END STACK

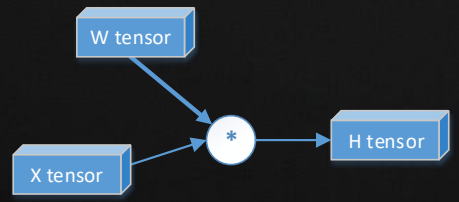
- Efficiency at every stage: GPU, Accelerator, SW stack, Cluster, Network

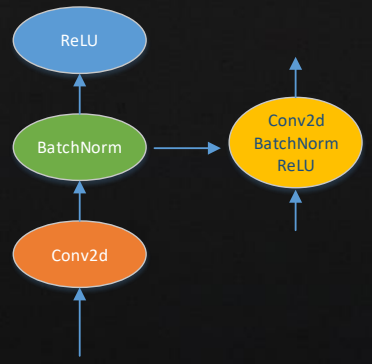
• Express networks in higher level languages such as Python, R, etc.	• Express data as tensors • Gradient computation	• Fuse operations • Dead code elimination • Memory planning and optimization	• Out-of-order dispatch • Work placement across multiple CPU/GPU nodes	• Tensors transformed to matrices • CPU to GPU commands
Application	API	Graph Optimizations	Parallel Scheduling	

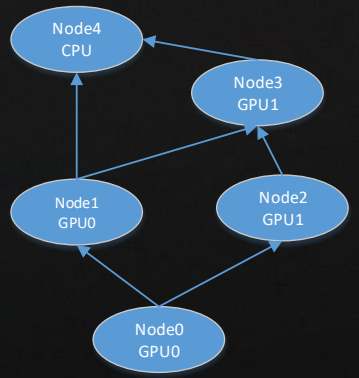


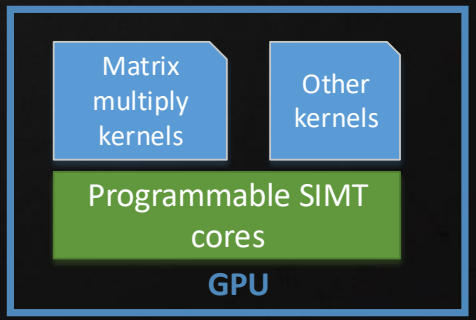
$h_k = w_k^T * x_i$











OPTIMIZATION TECHNIQUES

Many ways to simplify/optimize networks

-Pruning (static, dynamic)

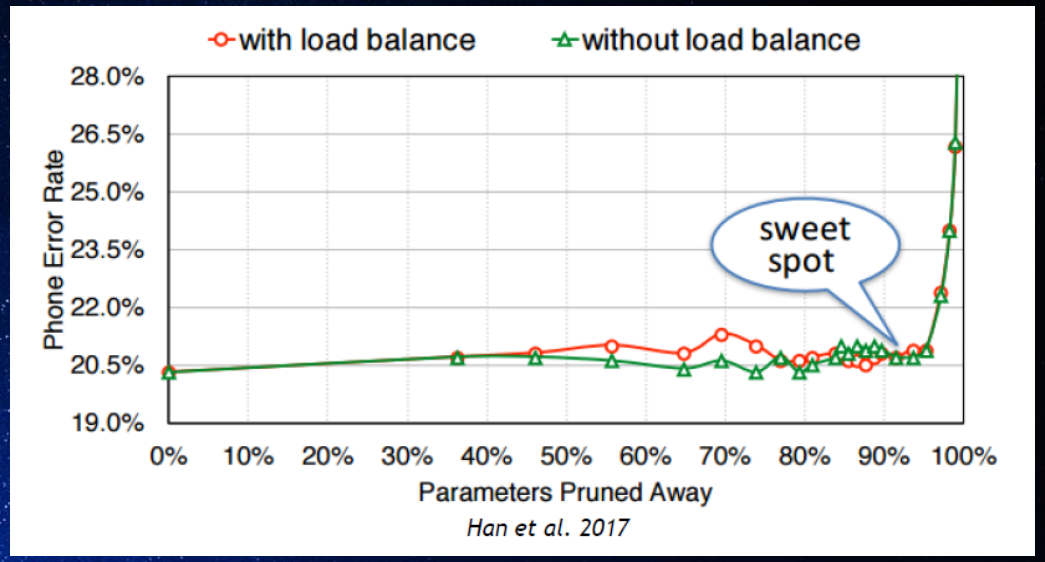
-Compression

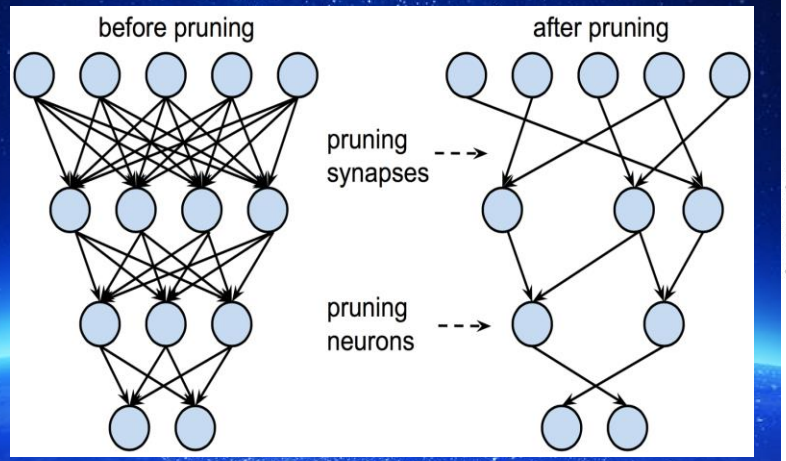
-Lower Precision

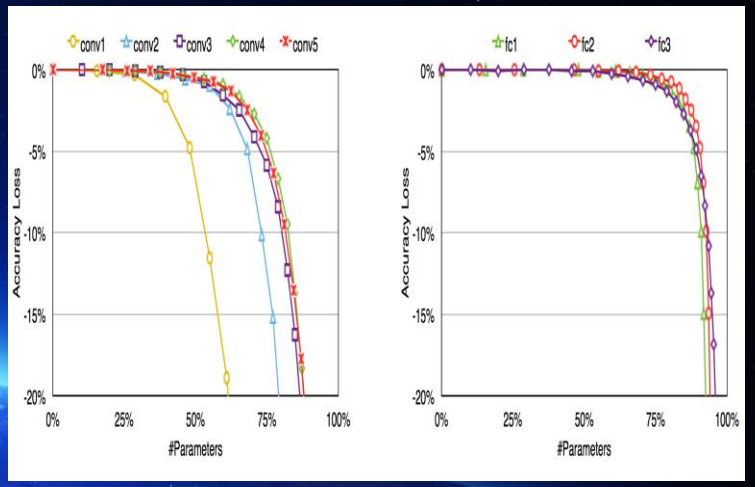
-Fusion, etc.

Leads to overall lower computational cost…

At the expense of some accuracy



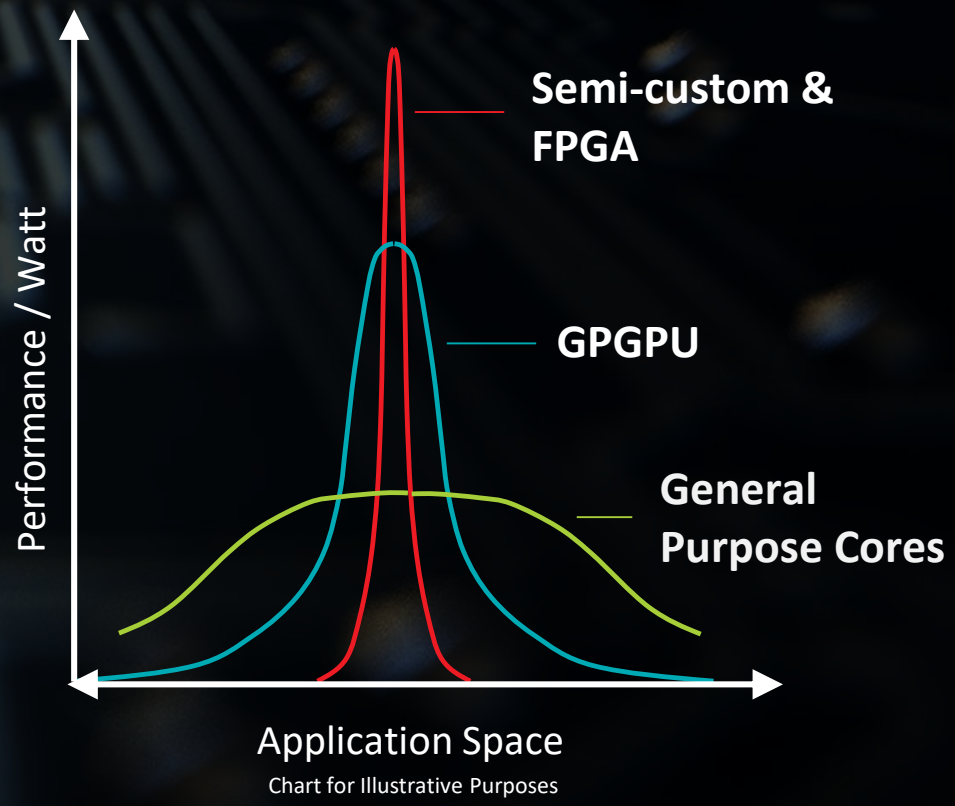




Source: Han, et al "Learning both weights and connections for efficient neural networks", NIPS 2015

AMD

Systems Design for Acceleration



CPU Cores

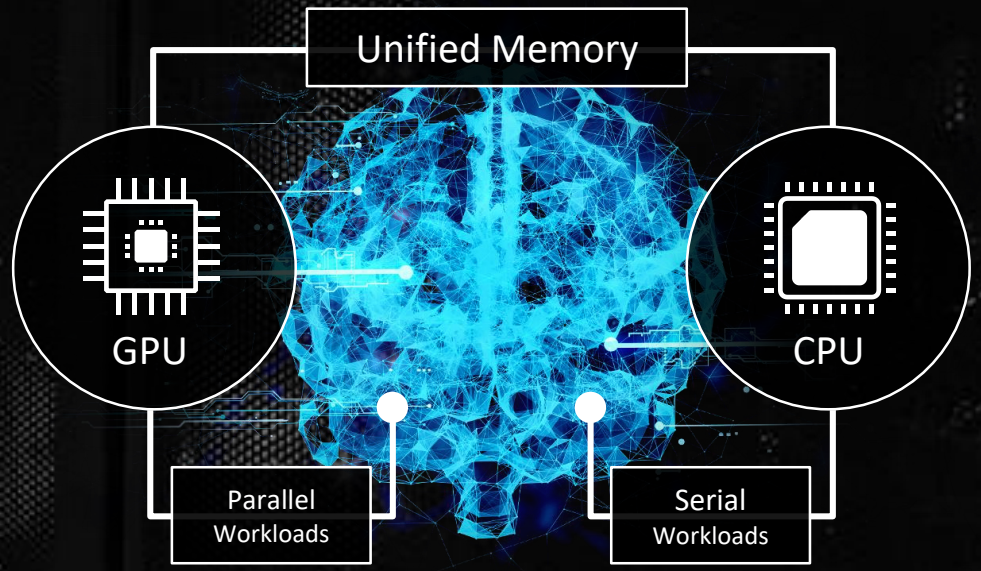
Graphics Processors

Semi-custom

FPGAs

AMD

Edge Computing – Leveraging Best Resources



Founding Member of Heterogeneous System Architecture Foundation

Radeon Open Compute (ROCm)

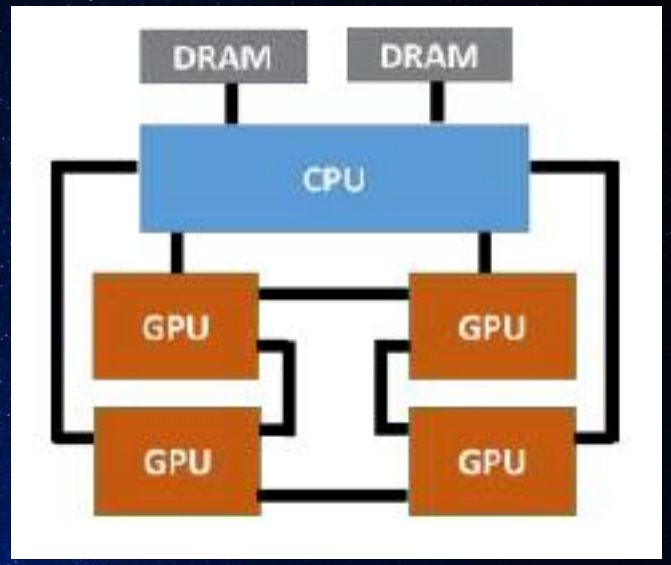
Support for x86, GPU compute platforms, deep learning frameworks

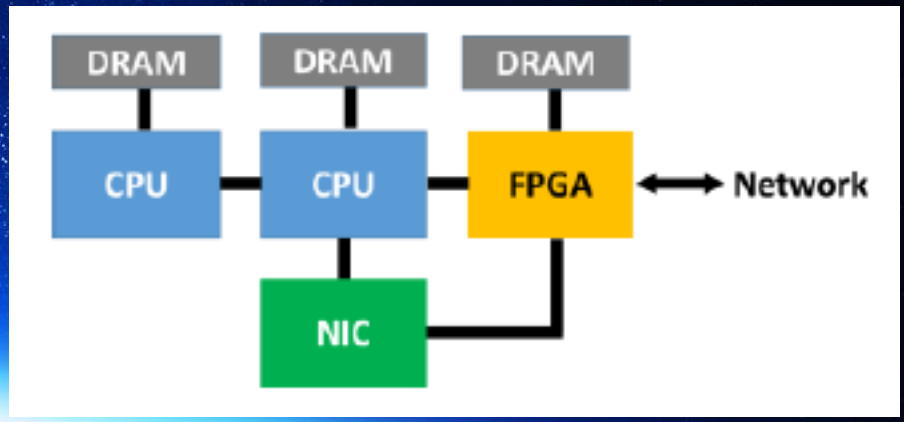
Open Interconnect Standards for Heterogeneous Accelerators

Founding member of CCIX, Gen-Z and OpenCAPI

EFFICIENT NETWORK CONFIGURATIONS TO SUPPORT ML/DL AT THE EDGE

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

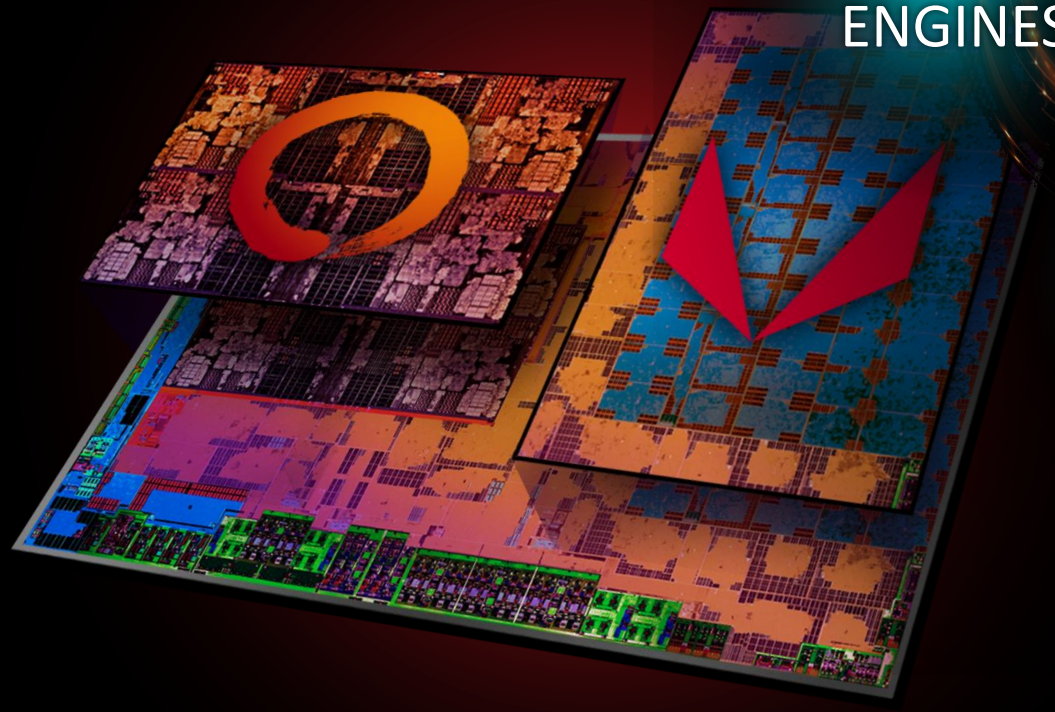




Source: Rush, Sirasao, Ignatowski, "Unified Deep Learning with CPU, GPU, and FPGA Technologies", 2017

INTEGRATION:
“ZEN” & “VEGA”

UNITED WITH INFINITY FABRIC



“ZEN” CORE COMPLEX

“VEGA” GRAPHICS

MULTIMEDIA ENGINES

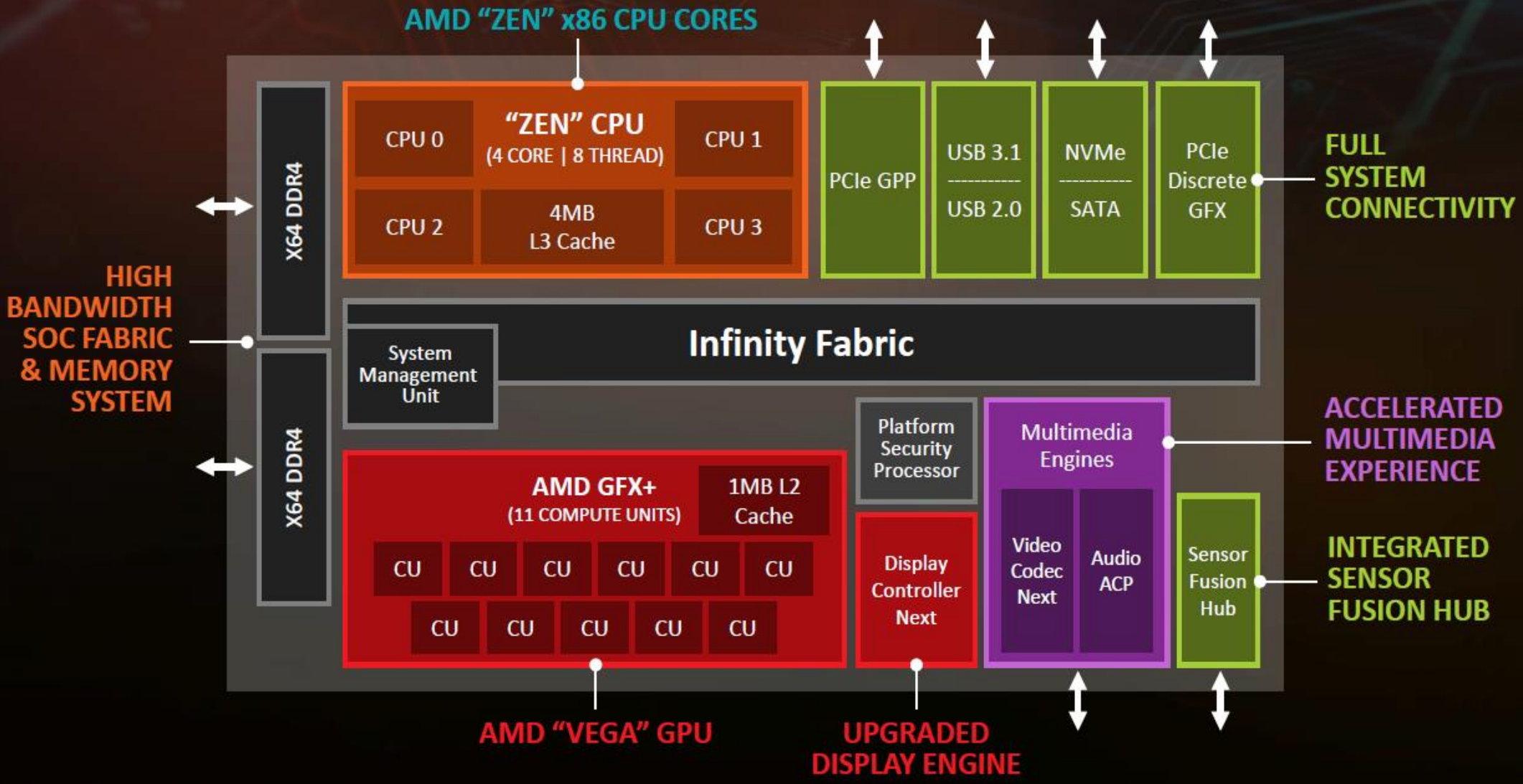
INFINITY FABRIC

I/O AND SYSTEM HUB

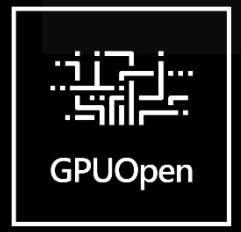
DISPLAY ENGINE

DDR4 MEMORY CONTROLLERS

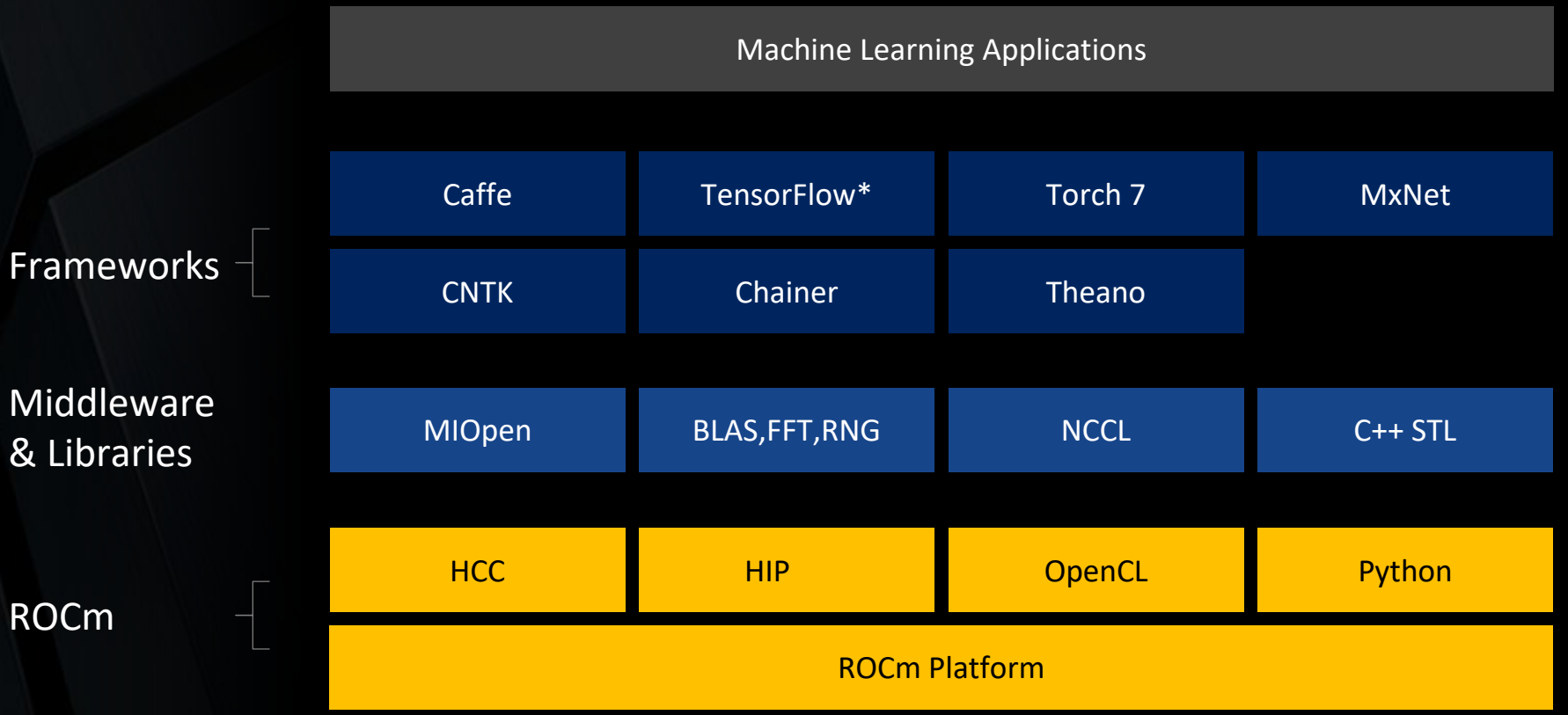
“RAVEN RIDGE” APU



Software:
Open Standard Open Source



Radeon Open Compute Software – Machine Learning



*Tensorflow support is expected to be available January 2017.

SOFTWARE STACK

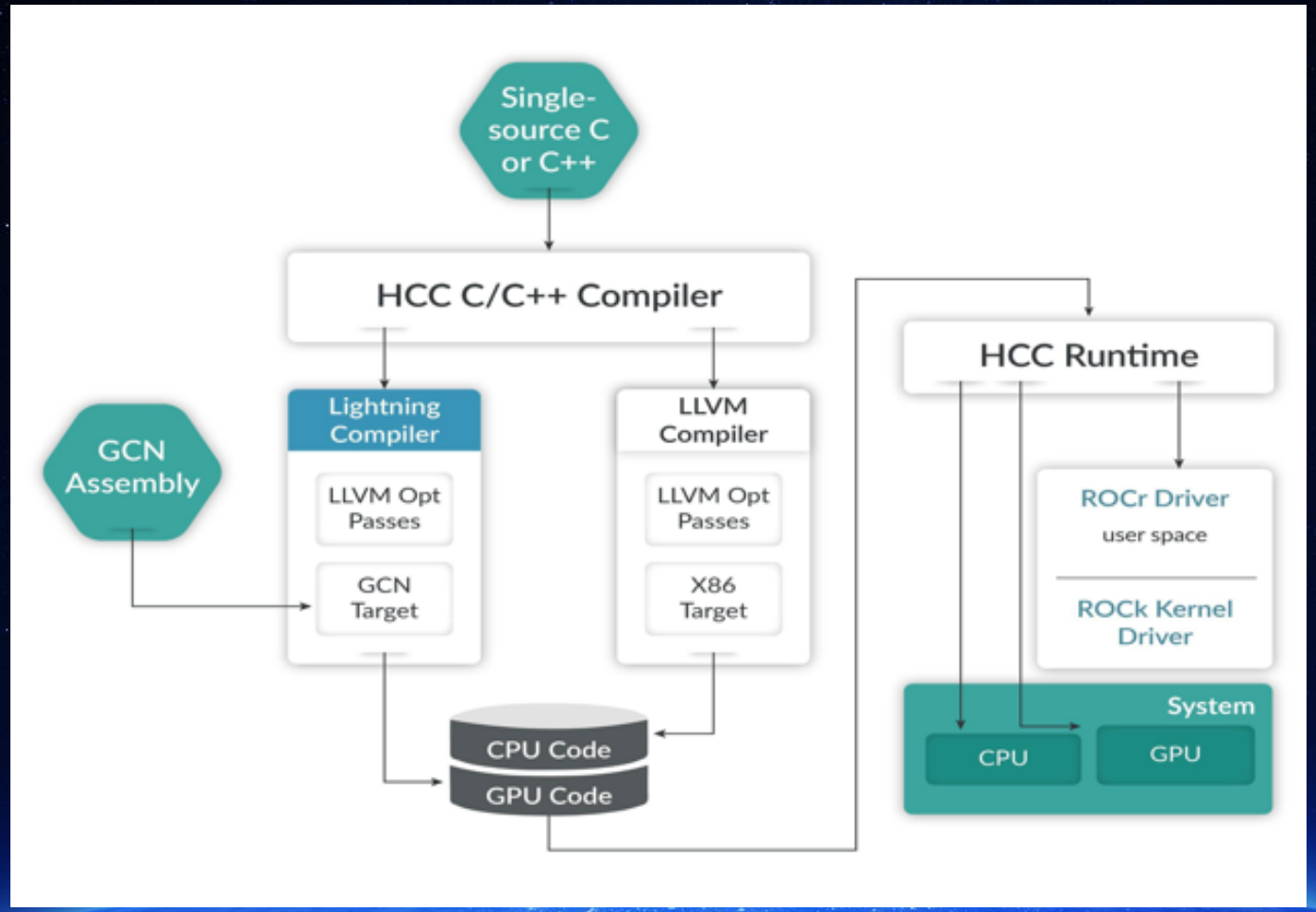
Applications	Machine Learning				
Frameworks	Caffe	TensorFlow	Keras		
	Caffe2	MxNet	Torch 7		
Middleware & Libraries	MIOpen	BLAS, FFT, RNG	NCCL	Eigen	C++ STL
ROCm	HCC	HIP	OpenCL™	Python	
	ROCm Platform				

LLVM

HSA

Open-source

Heterogeneous ML: Support for GPU, CPU targets



AMD

Machine Intelligence Applications – Rapid Development and Expanding Uses for Edge Compute

Autonomous Vehicles 	Autopilot Drone 	High Performance Computing 	Cloud Control 	Nano-Robots 	Medicine 	Personal Assistance 
Smart Home 	Personal Robots 	Security 	Financial Services 	Manufacturing & Engineering 	Energy 	


Thank You!
AMD

ATTRIBUTION & DISCLAIMER

DISCLAIMER

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

Use of third party marks / products is for informational purposes only and no endorsement of or by AMD is intended or implied.