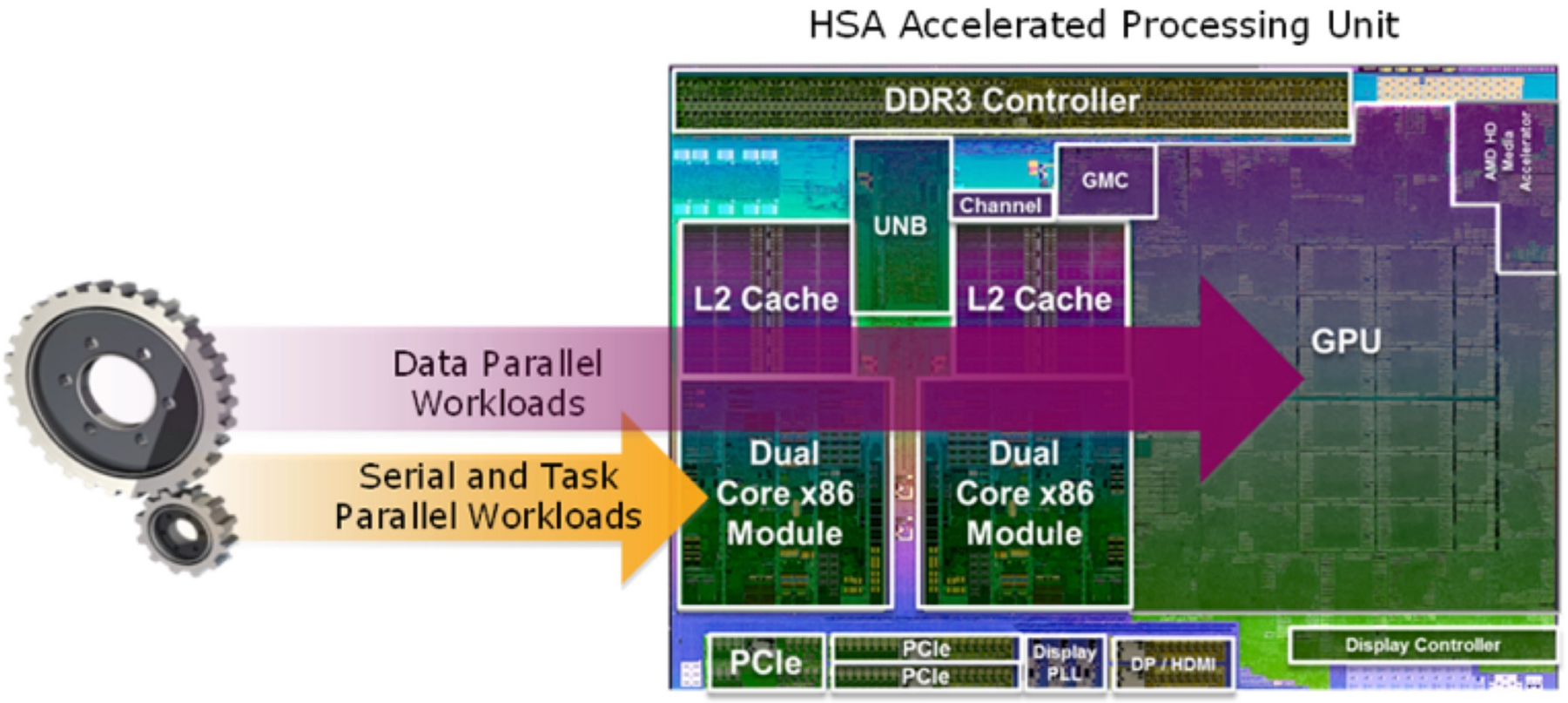
Lecture 19

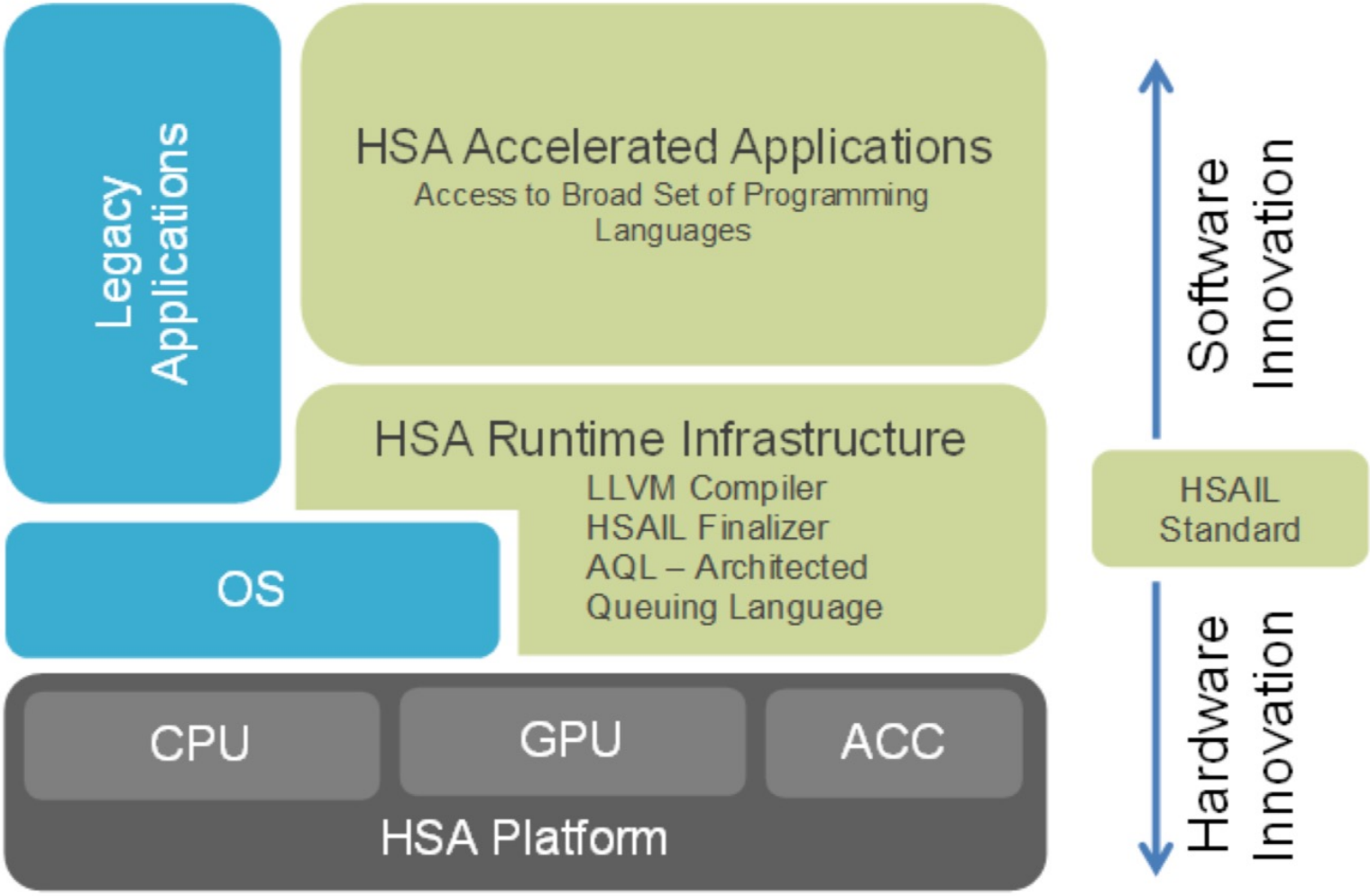
Heterogeneous System Architecture (HSA)

Heterogeneous System Architecture (HSA)

- **Provides a unified view of fundamental computing elements**
  - HSA allows a programmer to write applications that seamlessly integrate CPUs (called *latency compute units*) with GPUs (called *throughput compute units*), while benefiting from the best attributes of each



HSA Foundation



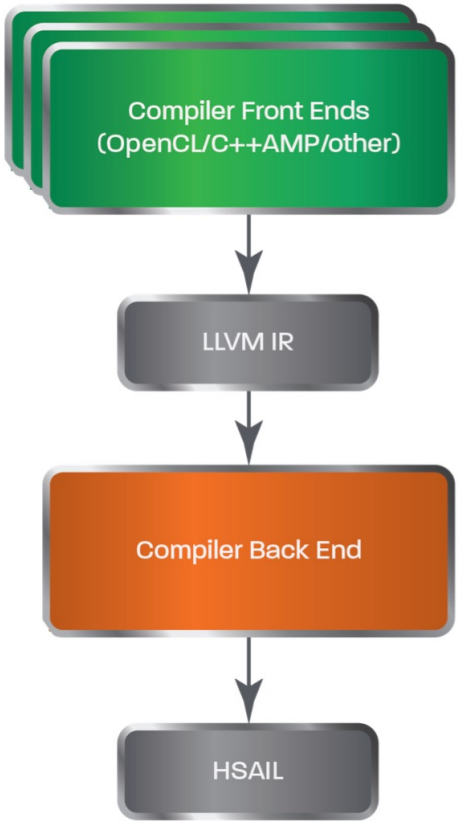
HSA Solution Stack

Goals of the HSA Foundation

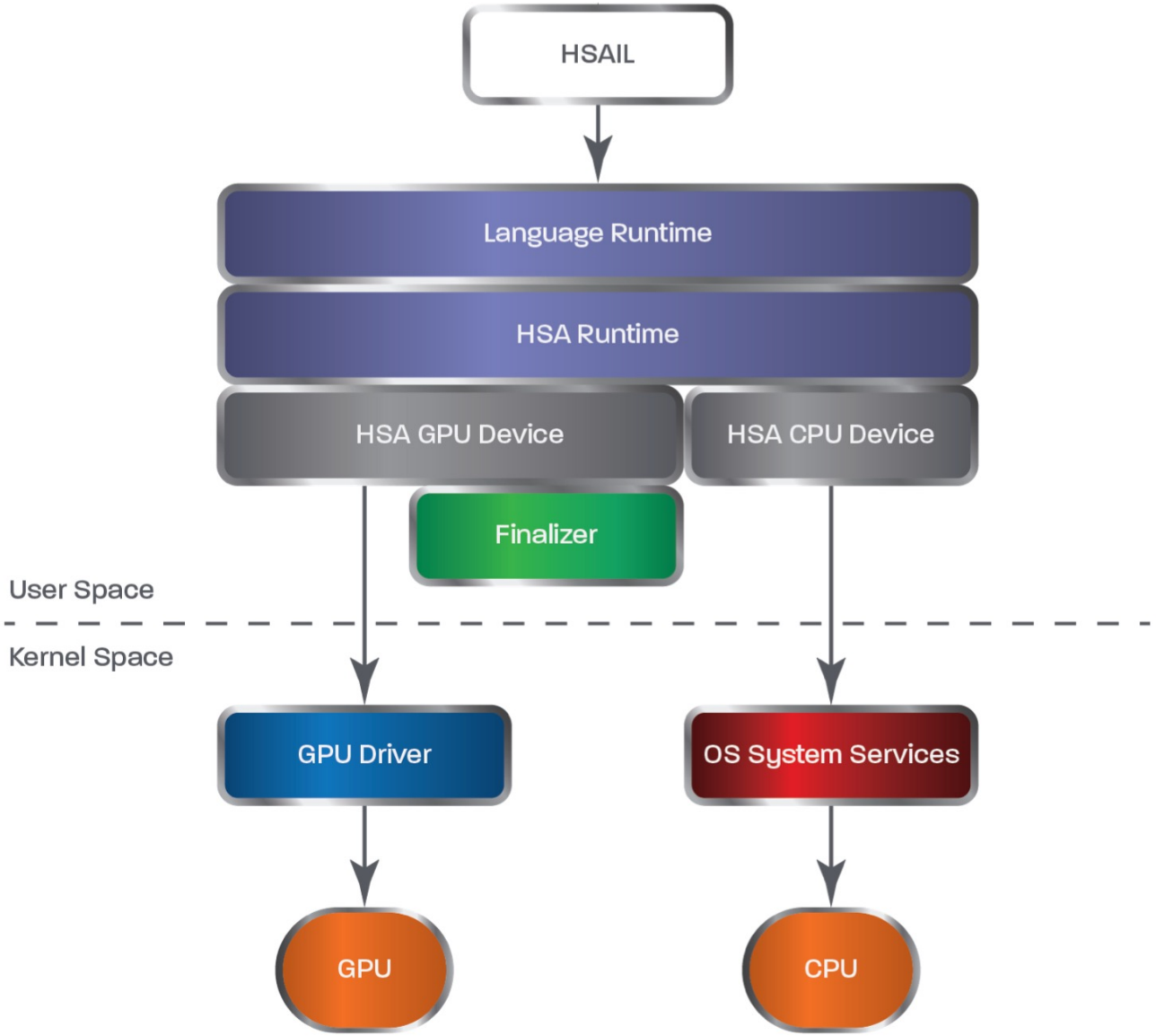
- To enable power-efficient performance
- To improve programmability of heterogeneous processors
- To increase the portability of code across processors and platforms
- To increase the pervasiveness of heterogeneous solutions throughout the industry

Implementation Components

- **A heterogeneous hardware platform that integrates both LCUs and TCUs, which operate coherently in shared memory**
- **A software compilation stack consisting of a compiler, linker and loader**
- **A user-space runtime system, which also includes debugging and profiling capabilities**
- **Kernel-space system components**
  - Device rivers



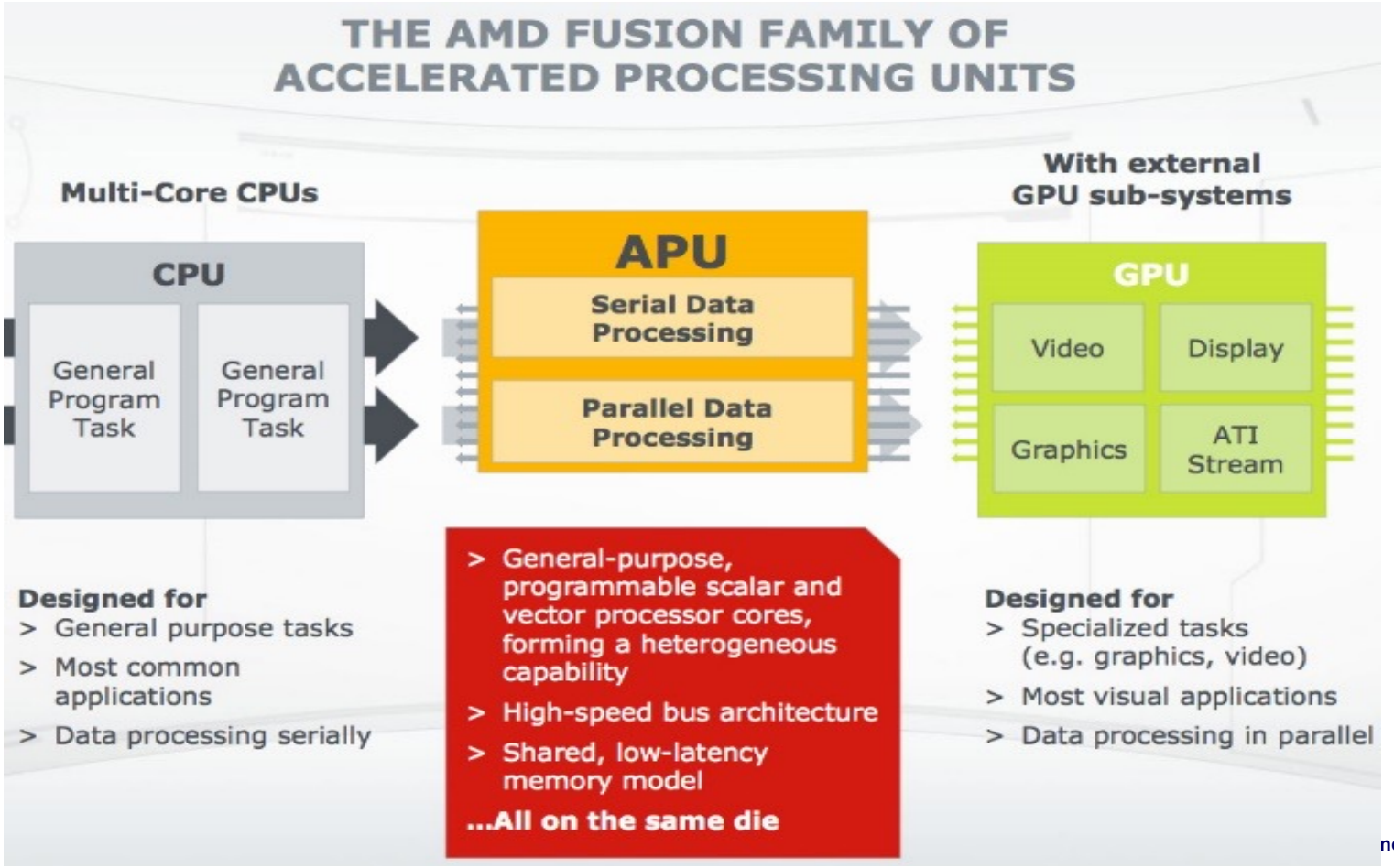
HSA Runtime Stack



Accelerated Processing Unit (APU)

APU

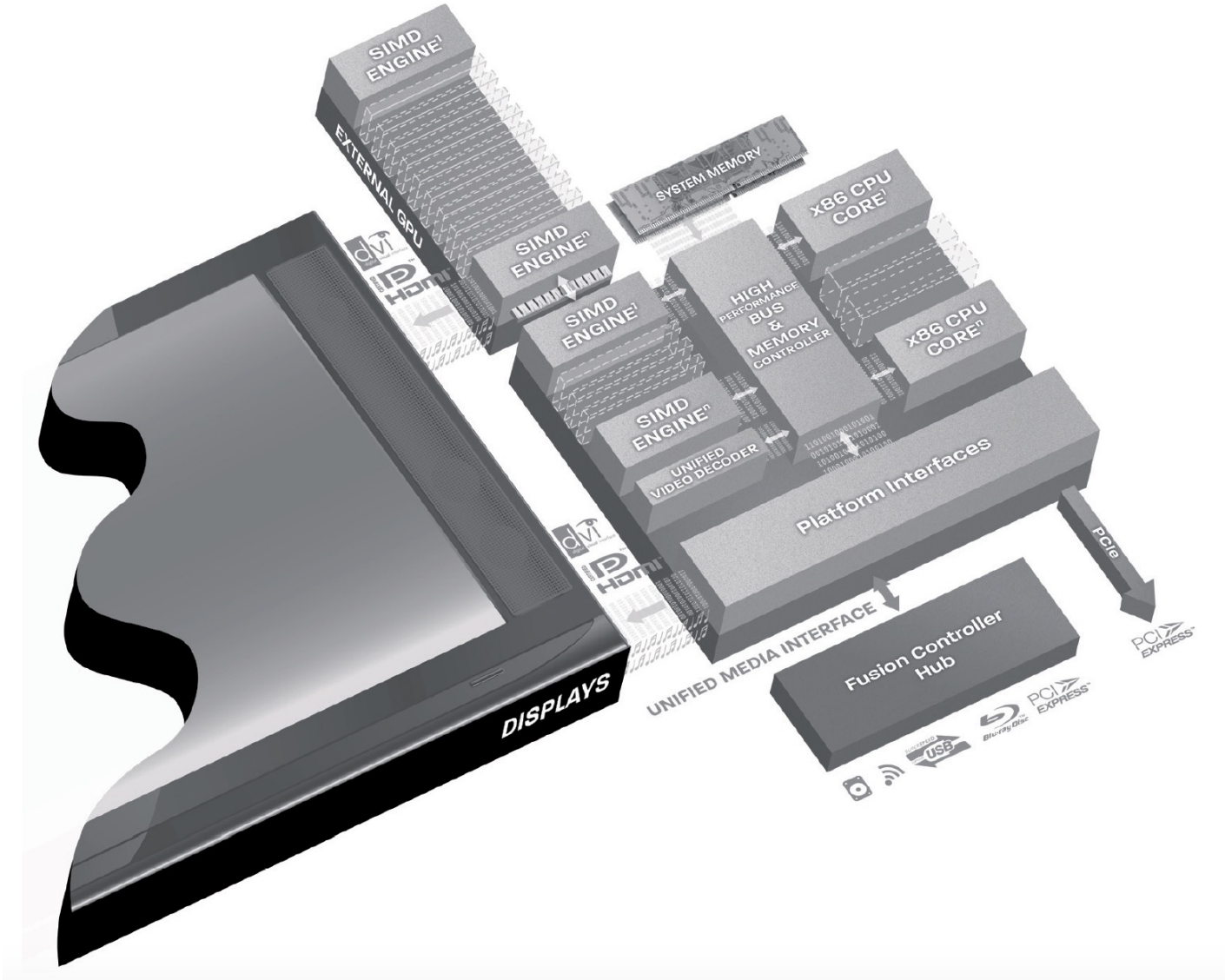
- **It is a processor that combines the CPU and the GPU elements into a single architecture (Appeared publicly in 2006)**



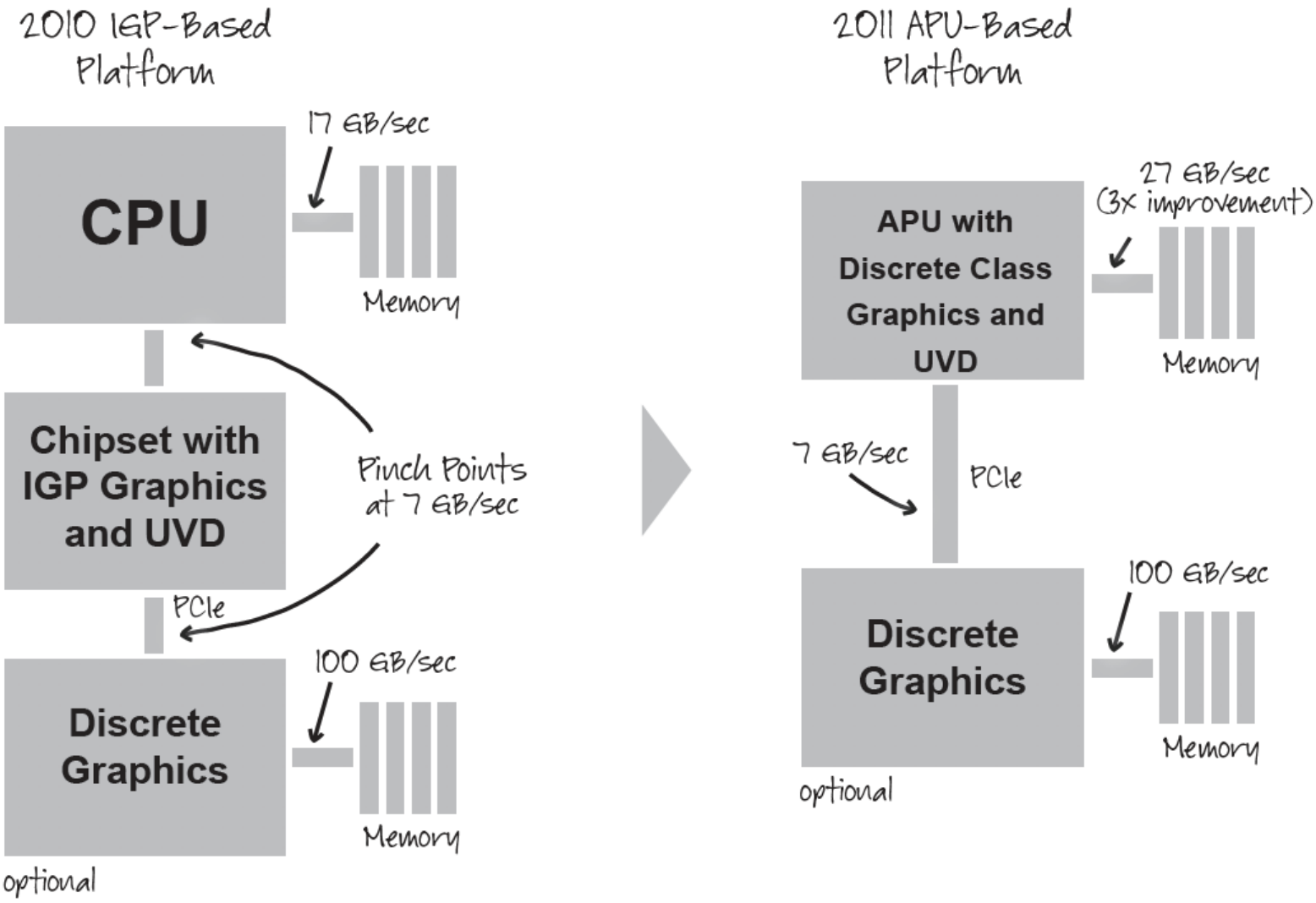
CPU vs. GPU

- **Different Design Goals**
  - CPUs are base on maximizing performance of a single thread
  - GPUs maximize throughput at cost of individual thread performance
- **CPU**
  - Dedicated to reduce latency to memory
- **GPU**
  - Focus on ALU and registers
  - Focus on covering latency

Balanced Computing



Bandwidth


2010 IGP-Based Platform
17 GB/sec
CPU
Memory
Chipset with IGP Graphics and UVD
Pinch Points at 7 GB/sec
PCIe
100 GB/sec
Discrete Graphics
Memory
optional
2011 APU-Based Platform
27 GB/sec (3x improvement)
APU with Discrete Class Graphics and UVD
Memory
7 GB/sec
PCIe
100 GB/sec
Discrete Graphics
Memory
optional

When to use an APU

- **Mobile Computing**
  - Low Power-usage = Longer Battery Life
  - Affordable Casual Gaming solution, or Media workstation.
- **Computations that can be done by GPU**
  - Allows both GPU and CPU to work together on one chip, without the need for a discrete graphics card.
  - Bitcoin, etc.
- **Budget Desktops (exception of higher model Intel APU's)**

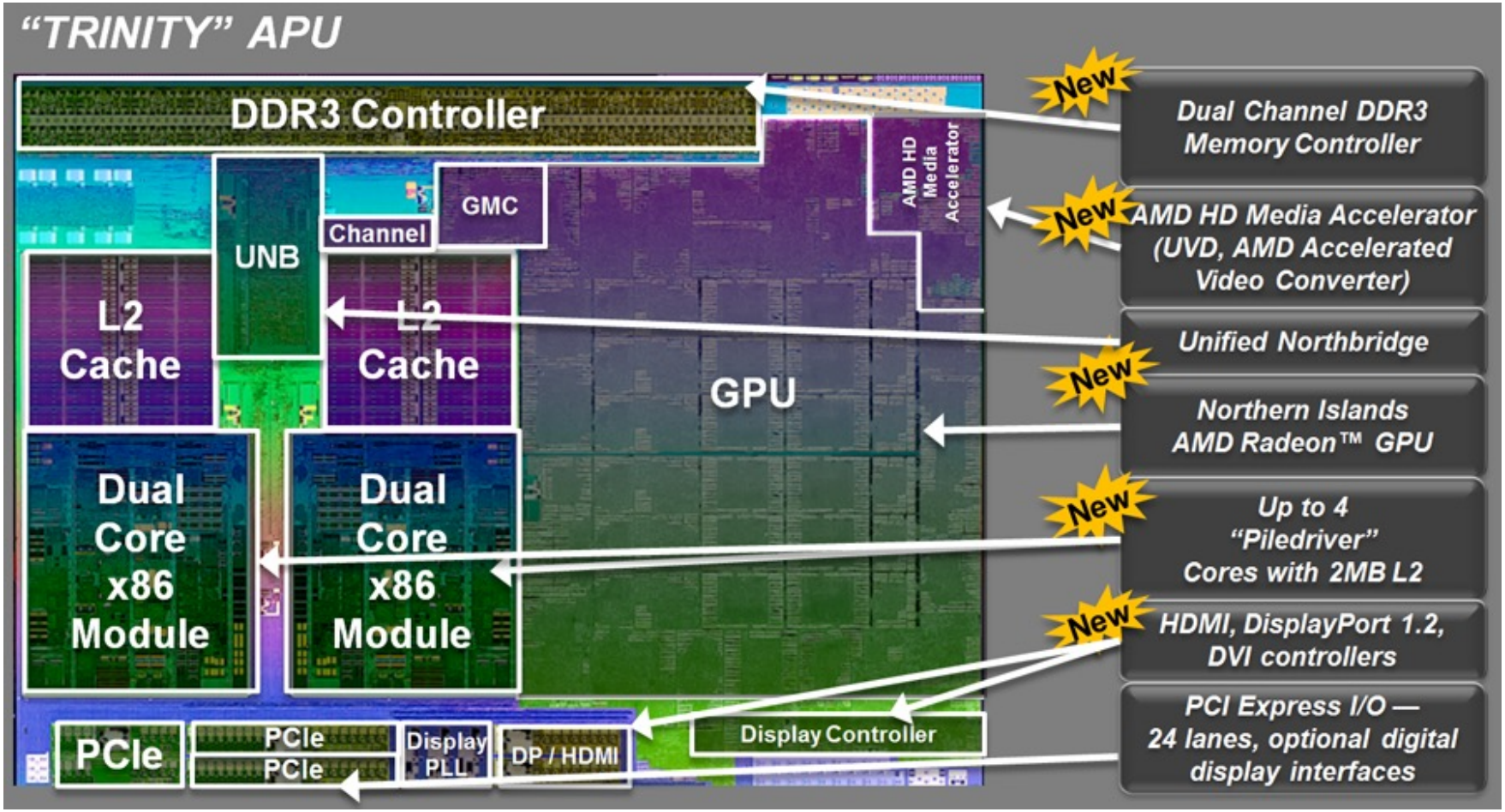
Who's Interested?

- **AMD**
  - Fusion Platform
- **Intel**
  - Sandy/Ivy Bridge series
- **IBM/Sony**
  - Cell (PS3)
- **NVidia**
  - Project Denver (ARM-based)

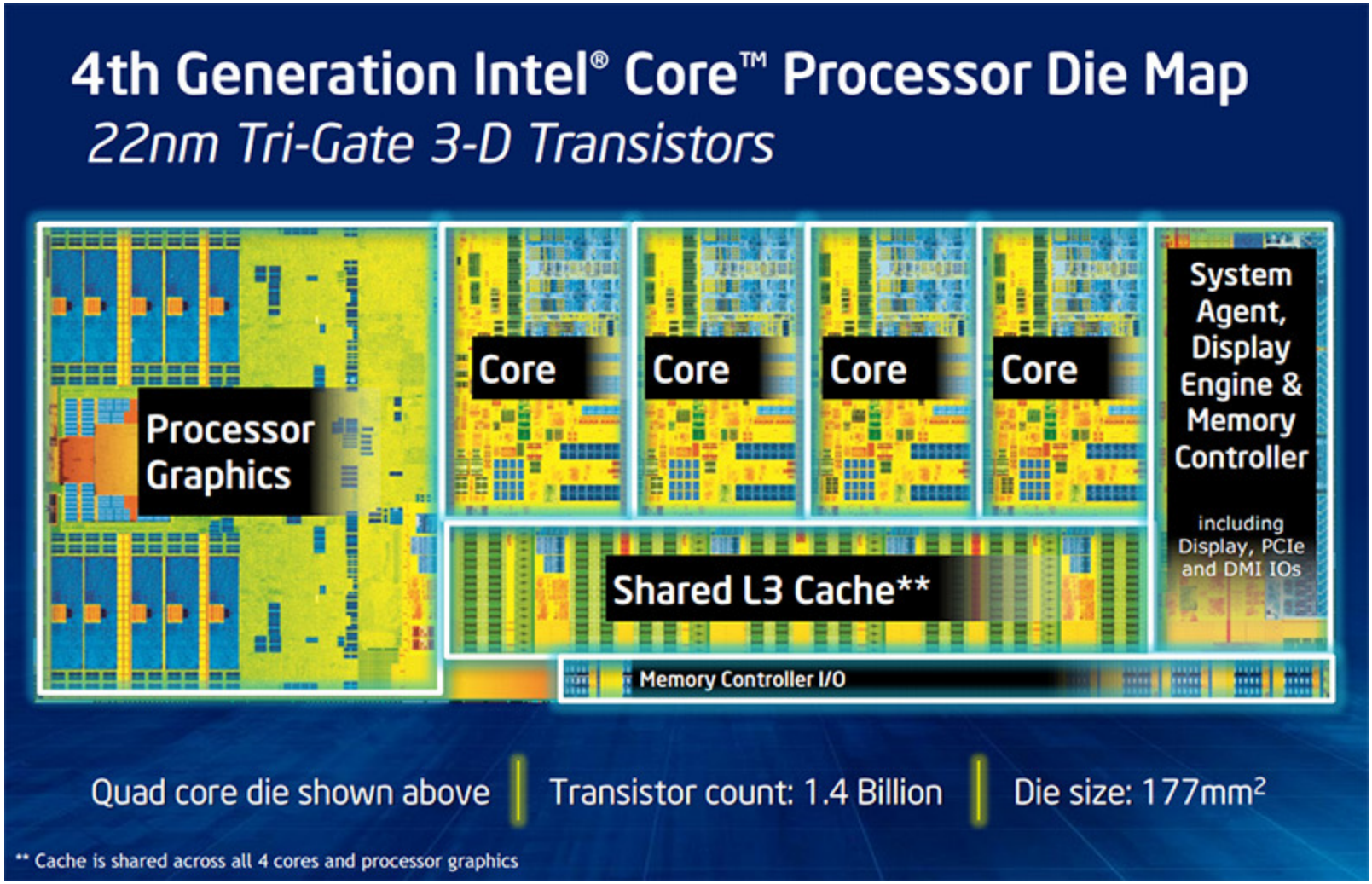
AMD Accelerated Processing Unit

- **AMD APU generations:**
  1. AMD announced the first generation APUs, *Llano* for high-performance and *Brazos* for low-power devices in January 2011
  2. The second generation *Trinity* for high-performance and *Brazos-2* for low-power devices were announced in June 2012
  3. The third generation *Kaveri* for high performance devices was launched in January 2014, while *Kabini* and *Temash* for low-power devices were announced in the summer of 2013
  4. In November 2017, HP released the Envy x360, featuring the Ryzen 5 2500U APU, the first 4th generation APU, based on the Zen CPU architecture and the Vega graphics architecture

AMD Trinity APU



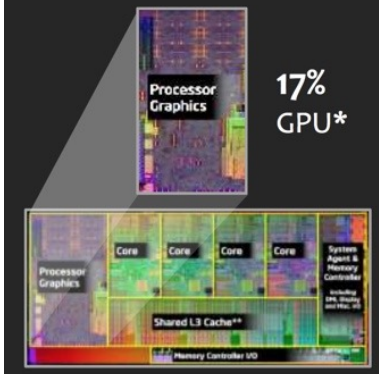
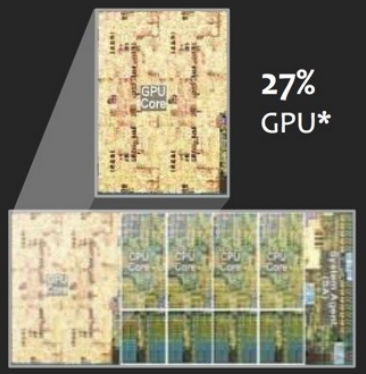
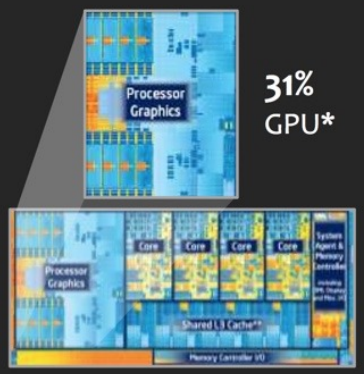
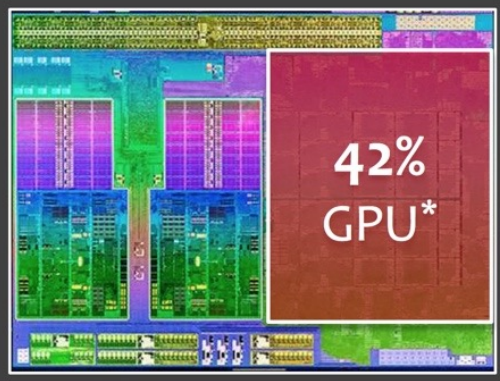
Intel Haswell



AMD APU

AND NOW THE APU IS EVERYWHERE

AMD

"SANDY BRIDGE"	"IVY BRIDGE"	"HASWELL"	ELITE AMD A-SERIES / CODENAMED "RICHLAND"
 17% GPU*	 27% GPU*	 31% GPU*	

- Balanced architectures are the future of computing
- Open CL™ is the future of parallel computing

A strong GPU is about;

- Gaming (DirectX®, OpenGL)
- Compute (Open CL™)
- Imaging (Adobe)
- Video (.mkv, Transcode)

*Percentage estimated by AMD based on relative portion of the GPU to the entire die

APU Advantages

- Eliminates the need for discrete GPU
- Good energy efficiency as it consumes very less power than the discrete GPU
- Improved heat dissipation
- Low latency access to the GPU
- Shared unified virtual memory for both the CPU and GPU
- Discrete graphics card can be installed with the APU to improve the performance to a greater extent
- Cheaper when compared to the discrete GPU's

APU Disadvantages

- Supports only entry level games on PC and not high resolution graphics like 4K
- Cannot upgrade the graphics card separately
- They have no internal memory and they take a chunk out of your RAM
- Not as powerful and flexible as the discrete GPU's
- Do not contain the number of cores as the discrete GPU's and so the performance will be no better than theirs

Intel Xeon Phi

Xeon Phi — MIC

- **Intel decided to enter the GPU market in the mid 2000s**
- **Xeon Phi = first product of Intel's Many Integrated Core (MIC) architecture**
- **Co-processor**
  - PCI Express card
  - Stripped down Linux operating system (busybox)
- **Dense, simplified processor**
  - Many power-hungry operations removed
  - Wider vector unit
  - Higher hardware thread count
- **Lots of names**
  - Many Integrated Core architecture, aka MIC
  - Knights Corner, aka KNC (code name)
  - Intel Xeon Phi Co-processor SE10P (product name)

Xeon Phi — MIC

- **Leverage x86 architecture (CPU with many cores)**
  - x86 cores that are simpler, but allow for more compute throughput
- **Leverage existing x86 programming models**
- **Dedicate much of the silicon to floating point ops**
- **Cache coherent**
- **Increase floating-point throughput**
- **Strip expensive features**
  - out-of-order execution
  - branch prediction
- **Widen SIMD registers for more throughput**
- **Fast (GDDR5) memory on card**

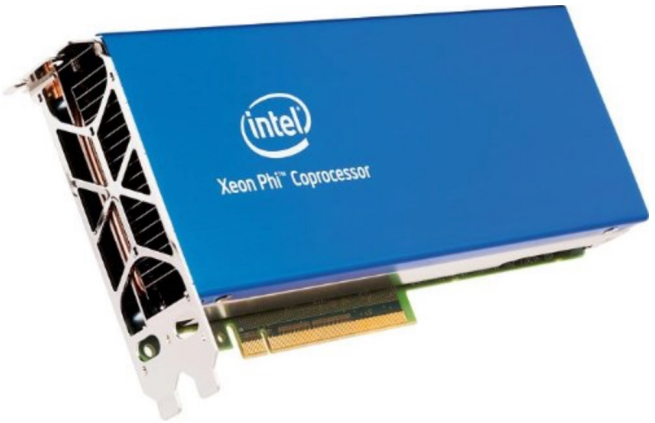
Intel Xeon Phi

- **Xeon Phi are a series of x86 manycore processor designed and made entirely by Intel**
- **Intended for use in supercomputers, servers, and high-end workstations**
- **Its architecture allows use of standard programming languages and APIs such as OpenMP**
- **Initially in the form of PCIe-based add-on cards, a second generation product, codenamed *Knights Landing* was announced in June 2013**

Code Name	Technology	Comments
Knights Ferry	45 nm	offered as PCIe card; derived from Larrabee project
Knights Corner	22 nm	derived from P54C; vector processing unit; first device to be announced as *Xeon Phi*
Knights Landing	14 nm	Abbr.: KNL[6]; derived from Silvermont/Airmont (Intel Atom)[7]; AVX-512
Knights Hill	10 nm	canceled
Knights Mill	14 nm	nearly identical to Knights Landing but optimized for deep learning

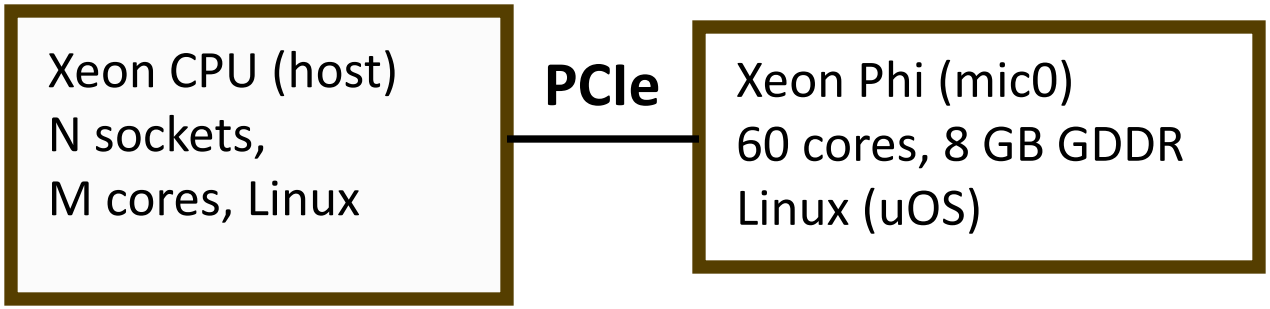
Intel Many Integrated Core (MIC)

- **Intel Xeon Phi (2012)**
- **PCI Express card, a ”PC in a PC”**
- **10s of x86-based cores**
  - Hardware multithreading
  - Instruction set extensions for HPC
- **Very high-bandwidth local GDDR5 memory**



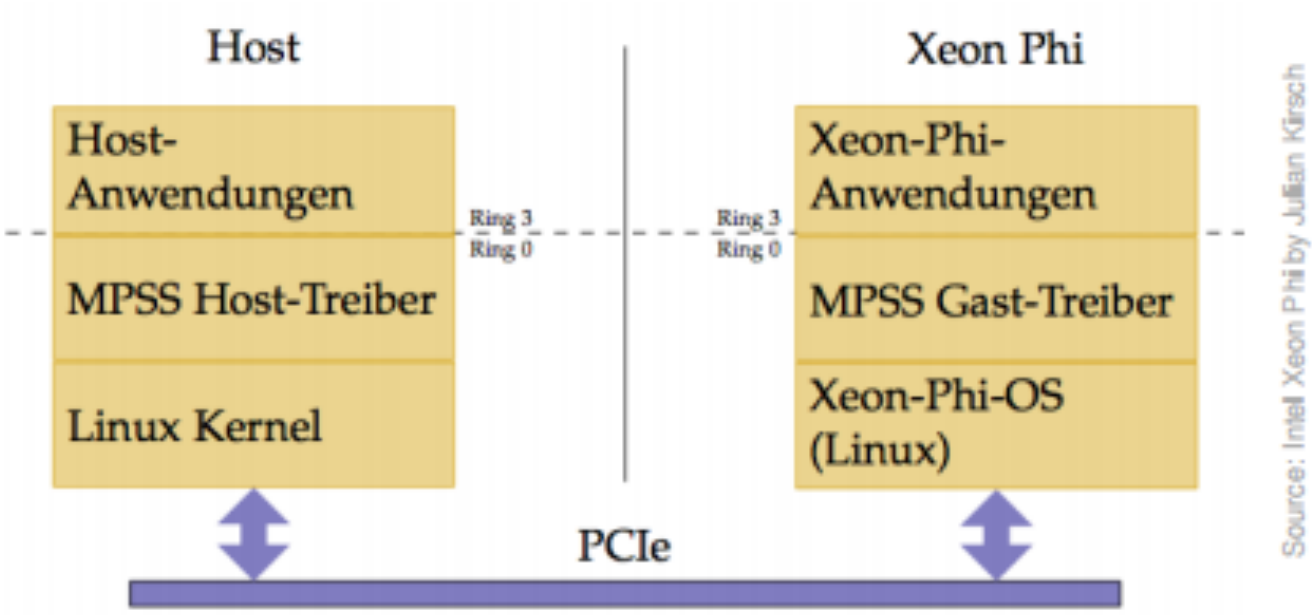
Software Environment on the Xeon Phi

- **The card runs a modified embedded Linux**
  - Called Micro OS (uOS) by Intel
  - The card boots from an image located on the host
  - The card does not have persistent memory
  - Provides a TCP/IP stack emulation over PCIe
  - Card appears as a network device to the host
  - **Busybox** is included for a variety of utilites (ls, top, …)



MPSS and uOS

- **Runs a stripped-down version of Linux**



MPSS: Manycore Platform Software Stack by Intel®

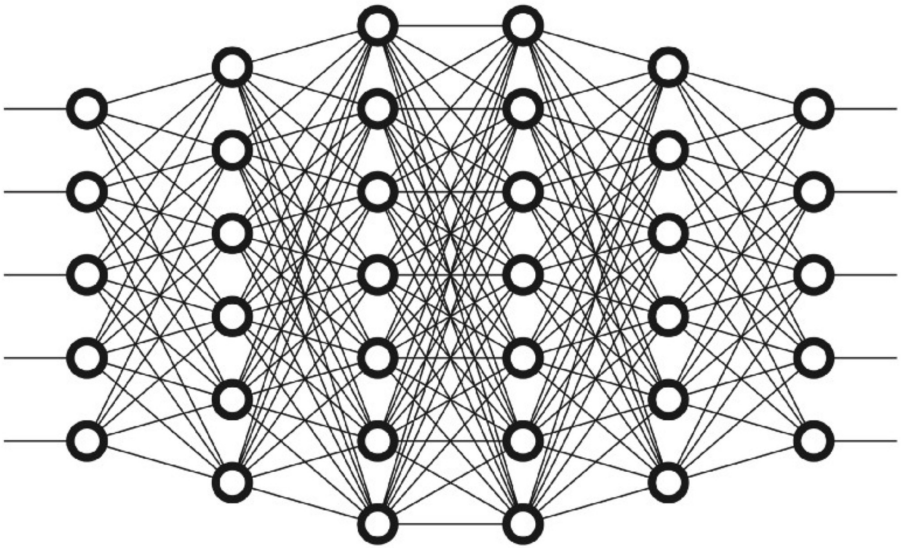
Advantages of Xeon Phi

- **Intel's MIC is based on x86 technology**
  - X86 cores w/ caches and cache coherency
  - SIMD instruction set
- **Programming for MIC is similar to programming for CPUs**
  - Familiar languages: C/C++ and Fortran
  - Familiar parallel programming models: OpenMP & MPI
  - MPI on host and on the coprocessor
  - Any code can run on MIC, not just kernels
- **Optimizing for MIC is similar to optimizing for CPUs**
  - **"Optimize once, run anywhere"**

Deep Learning: Deep Neural Netwokrs (DNN)

Deep Neural Networks

- **The neurons are grouped into three different types of layers:**
  1. Input Layer
  2. Hidden Layer(s)
  3. Output Layer
- **Each connection between neurons is associated with a weight**
  - This weight dictates the importance of the input value
  - The initial weights are set randomly
- **Each neuron has an Activation Function**



Training the Deep Neural Network

- **Requirements:**
  - A large data set
  - A large amount of computational power
- **Cost Function**
  - Define how wrong the outputs were from the real outputs
    - Turn to 0 when the outputs are the same as the data set outputs
  - Minimize the cost function and update the weights using gradient decent automatically
    - Converge and achieve a desired accuracy

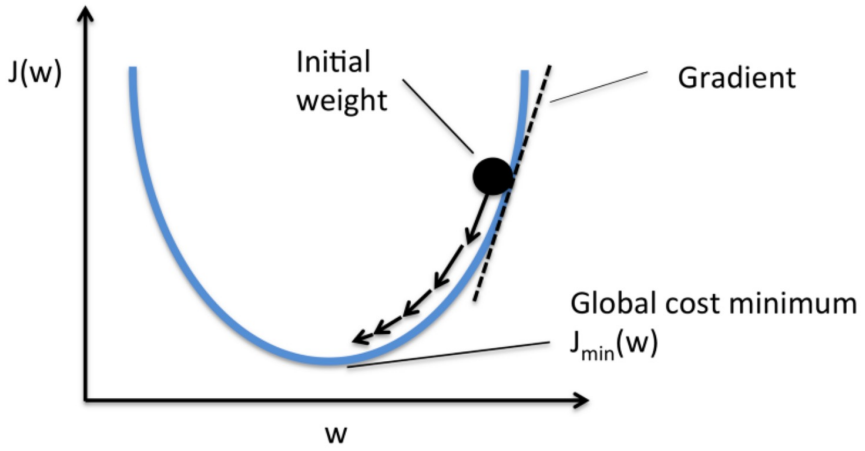


Image credit: Sebastian Raschka

Why to Talk about Deep Learning Here?

- **The whole deep neural network could be treated as a special "virtual processor"**
  - Training: guide the DNN model to converge and achieve a desired accuracy
  - Inference: use a trained DNN model to make predictions against previously unseen data

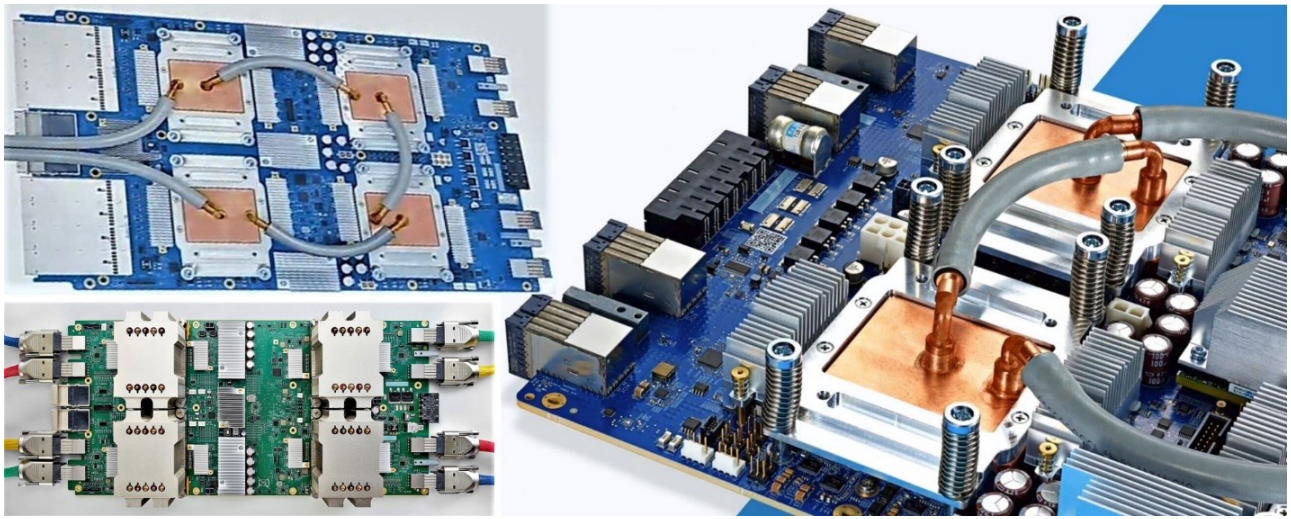
- **Turing Award winner Geoffrey Hinton (Godfather of Deep Learning): "… not only program computers, but also show [data to] computers …"**
  - Show Data = Train DNN

The DNN training datasets, not any program, define the behavior of DNN "virtual processor"

Tensor Processing Unit（TPU）

Tensor Processing Unit (TPU)

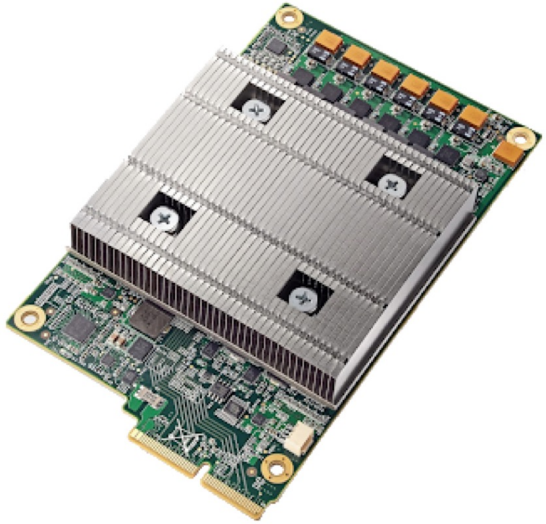
- **A custom ASIC for the phase of Neural Networks (AI accelerator)**
  - Google announced its TPU in May 2016 (4 generations + Edge TPU)
  - Improve the cost performance when compared to GPUs
  - Use Google's own TensorFlow software
    - TensorFlow is a symbolic math library, written in Python, C++, CUDA, and worked on CPU, GPU, and TPU

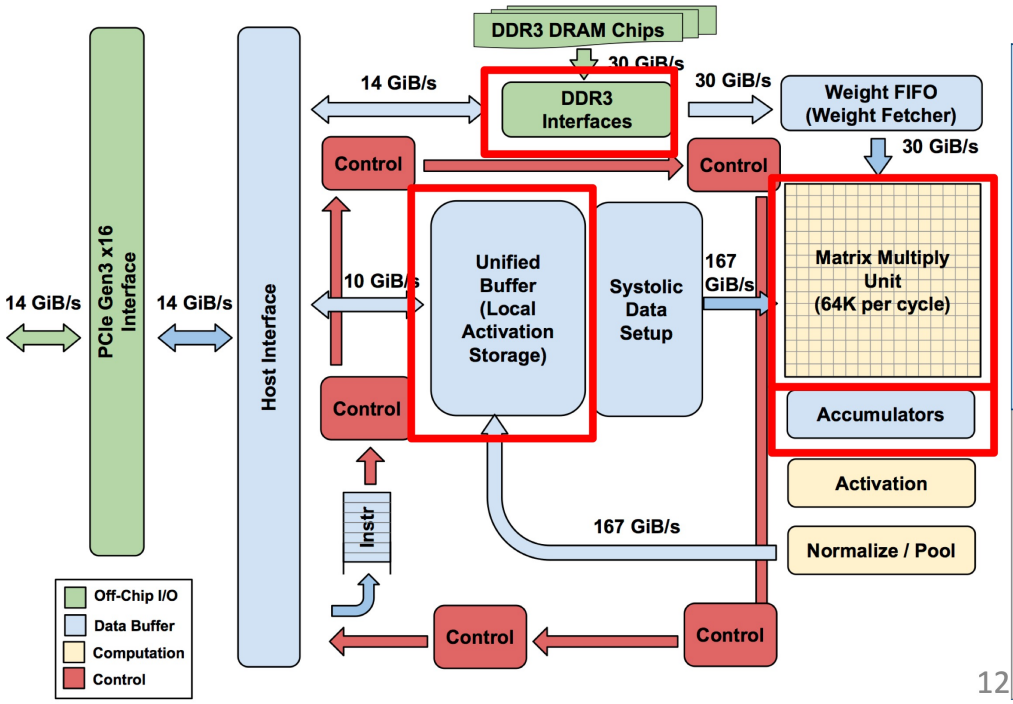


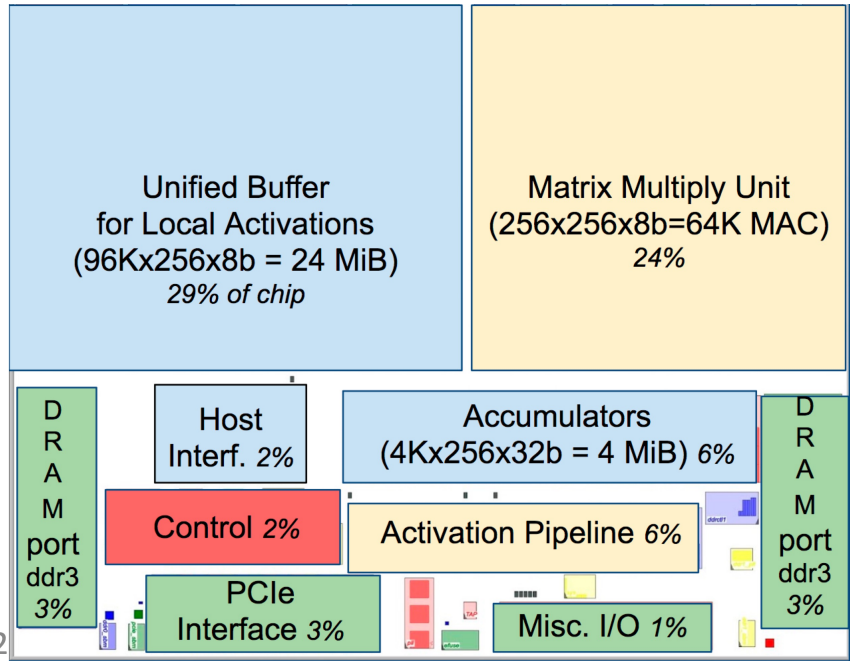
TPUv3 board (top left), TPUv2 board (bottom left), and TPUv3 board close-up (right)

TPUv1 Architecture

- **With dedicated Matrix Multiply Unit**
  - Most to memory and computation
  - Less to control
  - Half the size of CPUs and GPUs



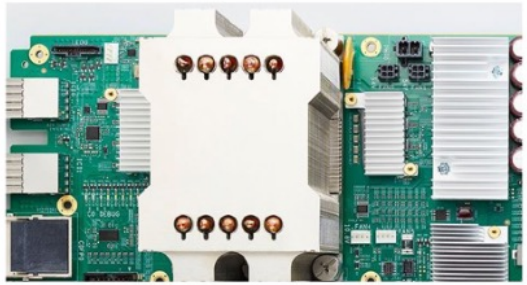




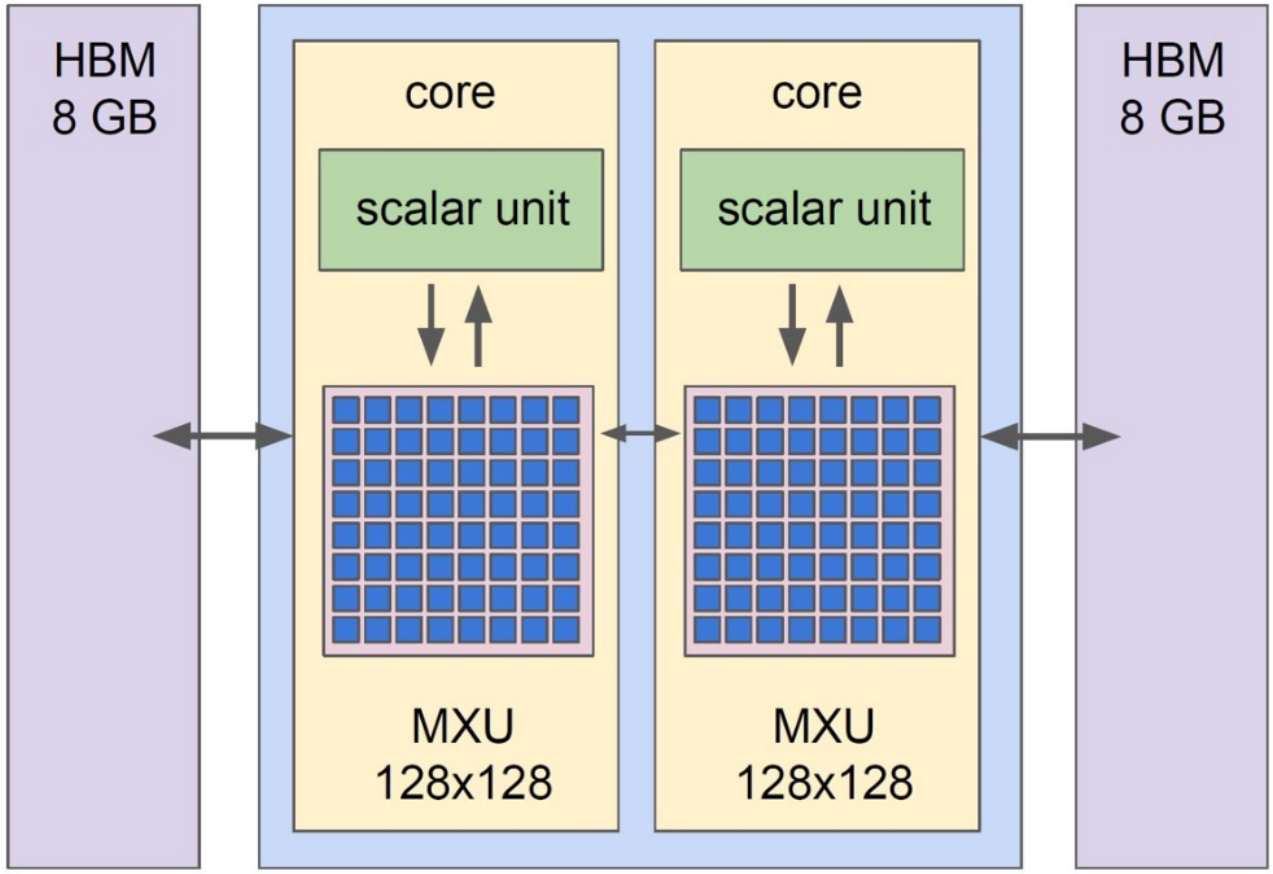
12

TPUv2 Architecture

TPUv2 Chip



- 16 GB of HBM
- 600 GB/s mem BW
- Scalar unit: 32b float
- MXU: 32b float accumulation but reduced precision for multipliers
- 45 TFLOPS



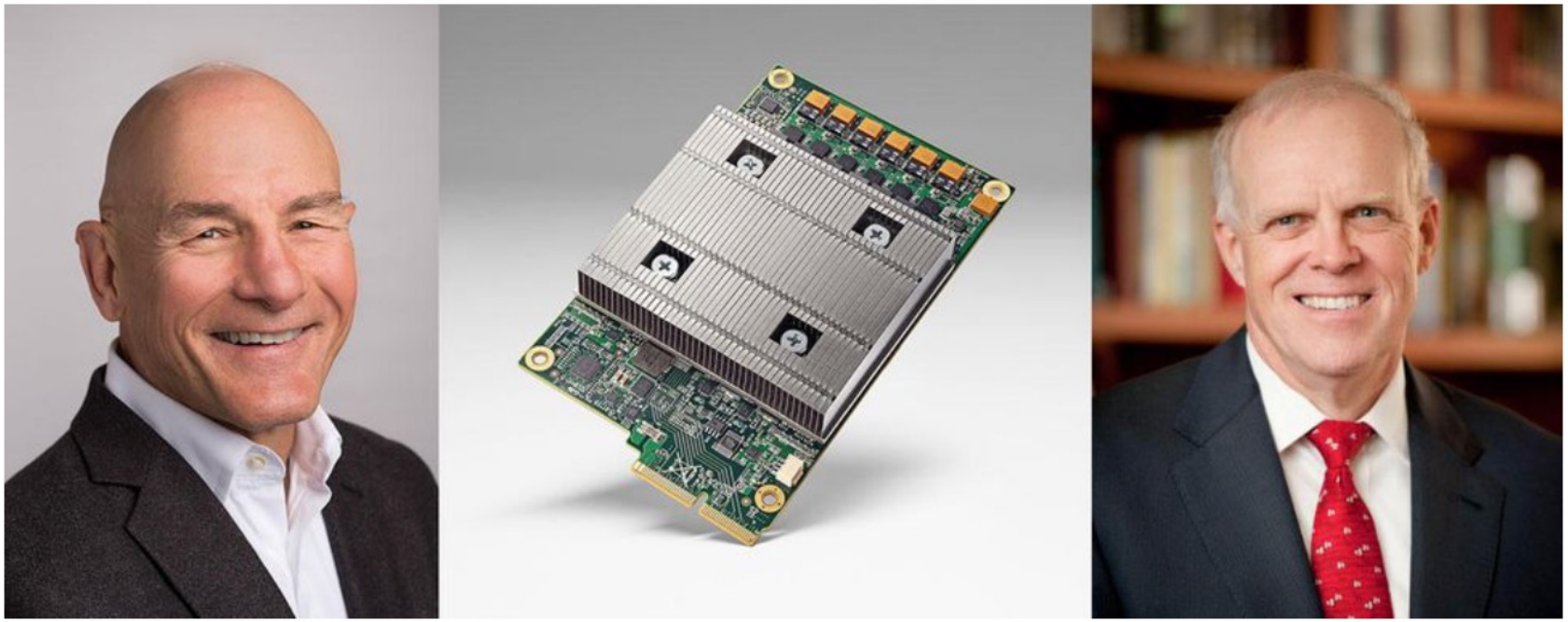
TPUv2 Architecture

Cloud TPU

- **All model parameters are kept in on-chip high bandwidth memory**
- **The cost of launching computations on cloud TPU is amortized by executing many training steps in a loop**
- **Input training data is streamed to an "infeed" queue on the cloud TPU**
- **A program running on cloud TPU retrieves batches from these queues during each training step**
- **The TensorFlow server running on the host machine (the CPU attached to the cloud TPU device) fetches data and pre-processes it before "infeeding" to the cloud TPU hardware**
- **Data parallelism:**
  - Cores on a cloud TPU execute an identical program residing in their own respective HBM in a synchronous manner
  - A reduction operation is performed at the end of each neural network step across all the cores

Who's in TPU Team?

- **Professor David Patterson retired from U.C. Berkeley in 2016 after a 40-year academic career in computer architecture**



David Patterson, left, and John Hennessy won the 2017 ACM Turing Award for inventing RISC processors. They're now pushing special-purpose chips such as Google's Tensor Processing Unit, center, for speeding up AI.

University of California, Google, Stanford University

Advantages of TPU

- **TPUs allows us to make predictions very quickly and respond within fraction of a second.**
- **First instance of a computer defeating a world champion in the ancient game of Go.**
- **Accelerate performance of linear computation, key of machine learning applications.**
- **Minimize the time to accuracy when you train large, complex network models**

Disadvantages of TPU

- Linear algebra that require heavy branching or are not computed on the basis of element wise algebra
- Non-dominated matrix multiplication is not likely to perform well on TPUs
- Workloads that access memory using sparse technique
- Workloads that use highly precise arithmetic operations