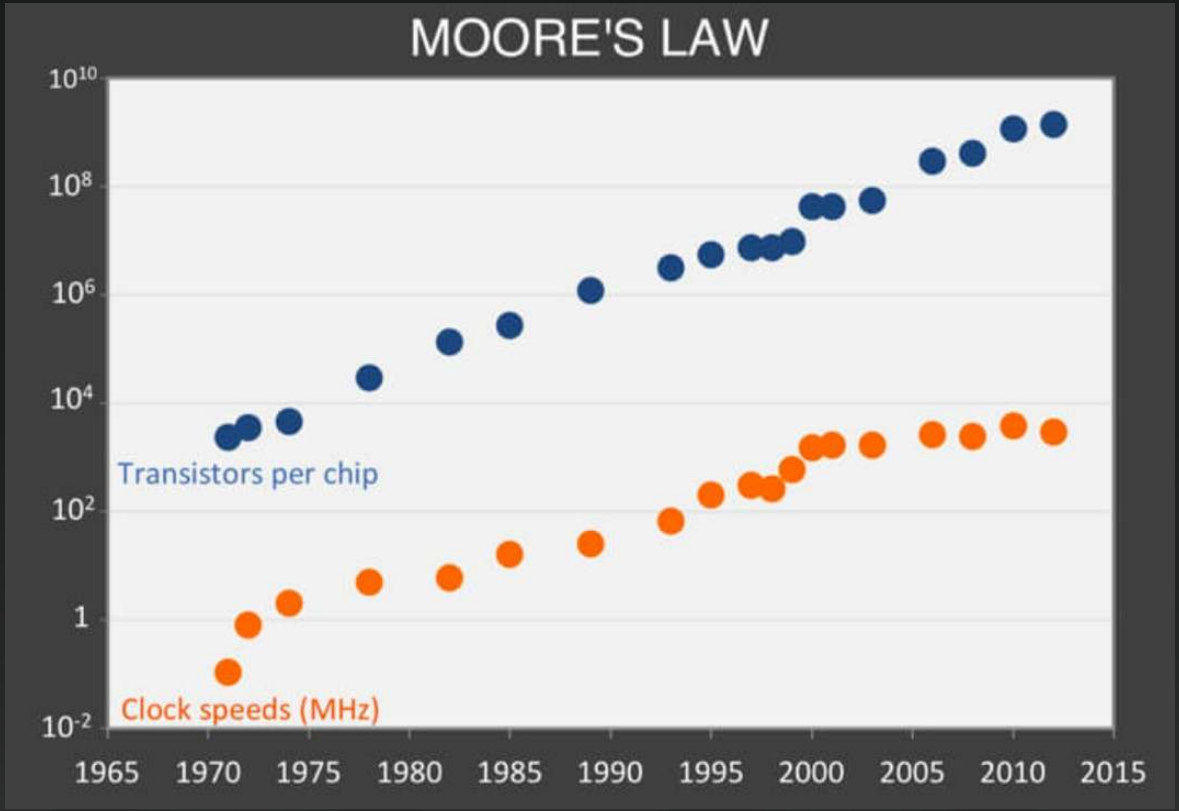
# Hardware acceleration with FPGAs on AWS

Julien Simon, Principal Evangelist AI/ML, AWS
Ramine Roane, Senior Director, Product Management, Xilinx
Sébastien Delerse, Co-founder, Mipsology

# Agenda

- The case for accelerating computing
- What is an FPGA?
- Using FPGAs on AWS
- Customer case studies
- Resources

# The case for accelerated computing




Source:
Intel

# Moore’s winter is (probably) coming

- « *I guess I see Moore’s Law dying here in the next decade or so, but that’s not surprising* », Gordon Moore, 2015
- Technology limits: a Skylake transistor is around 100 atoms across.
- New workloads require higher parallelism to achieve good performance.

http://www.economist.com/technology-quarterly/2016-03-12/after-moores-law
https://spectrum.ieee.org/computing/hardware/gordon-moore-the-man-whose-name-means-progress

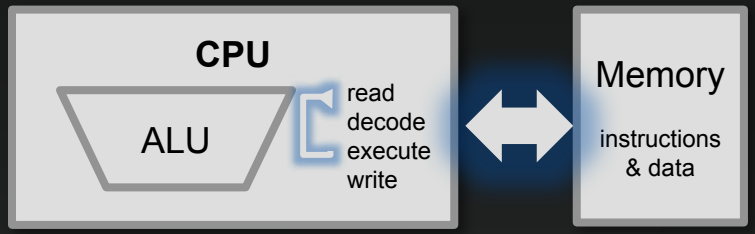
# GPUs are not optimal for some applications

- Power consumption and efficiency (TOPS/Watt)
- Strict latency requirements
- Other requirements
  - Custom data types, irregular parallelism, divergence

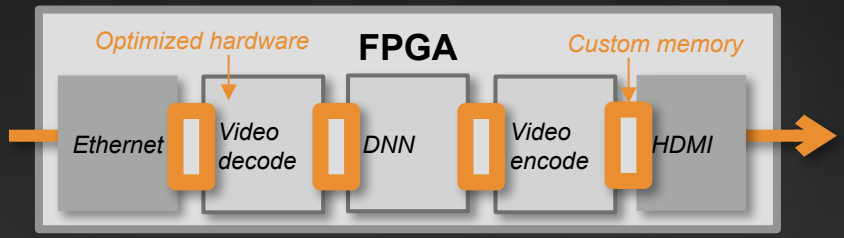
- Building your own ASIC may solve this, but:
  - It's a huge, costly and risky effort
  - ASICs can't be reconfigured
- What about FPGA?

# Why FPGA: Application Specific HW & Memory


- Given an algorithm to implement…




- CPU/GPU implementation: Von Neumann
  - Rigid sequential execution (SIMD for GPU)
  - Memory access bottleneck
  - Not optimal for custom width or decision handling



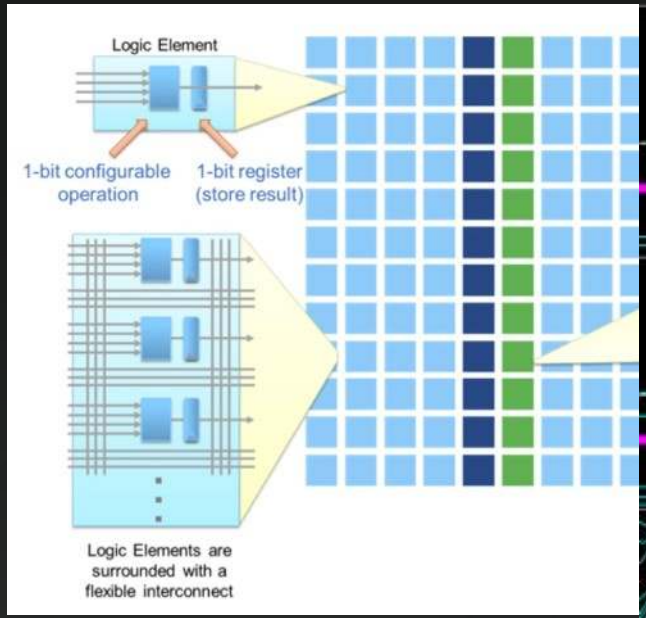

- FPGA implementation: WYSIWYG
  - Custom dataflow, width, decision handling
  - Custom memory hierarchy: keeps data inside
  - Custom IOs: high throughput & low latency

# What’s an FPGA?

# The FPGA

- First commercial product by Xilinx in 1985
- Field Programmable Gate Array
- Not a CPU (although you could build one with it)
- « Lego » hardware: logic cells, lookup tables, DSP, I/O
- Small amount of very fast on-chip memory
- Build custom logic to accelerate your SW application

# FPGA architecture




Sources:
https://www.embedded-vision.com/industry-analy
http://www.bober-optosensorik.de/fpga-entwicklu

# Where are FPGAs Traditionally Used?

## Communications
- Wired networking
- Wireless infrastructure

## Automotive
- Infotainment
- Driver assistance

## Datacenter
- High performance computing
- Solid state drives

## Aerospace and Defense
- Avionics, Communications
- Space

## Test & Measurement
- Communications instruments
- Semiconductor test equipment

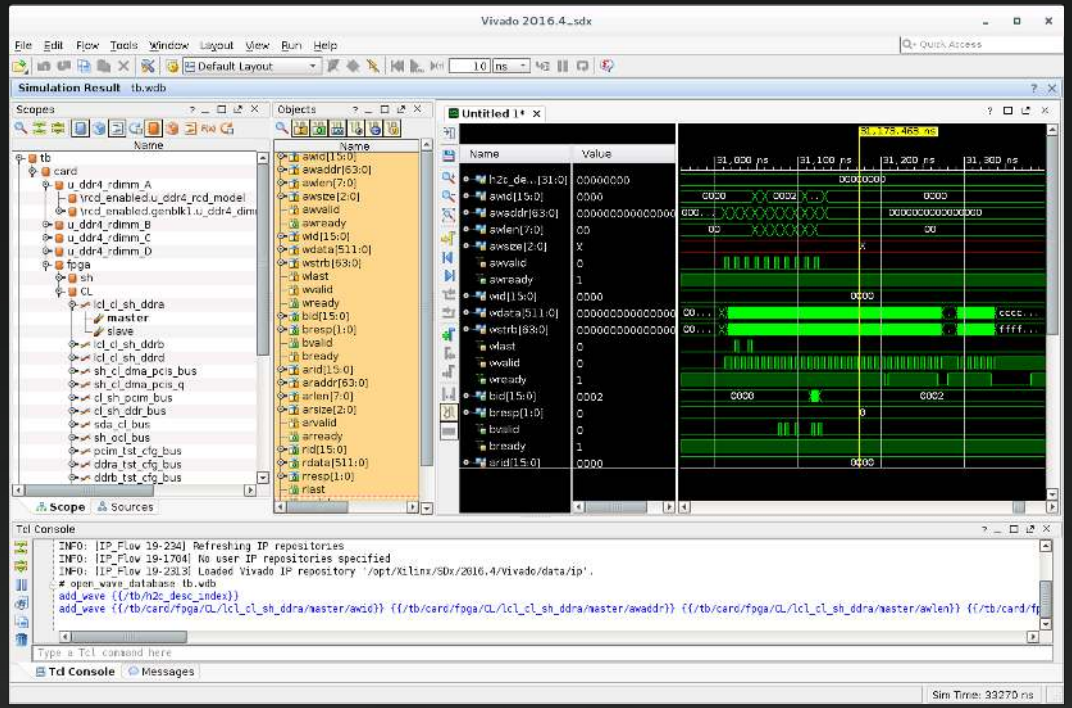
## Audio, Video, Broadcast
- 3D cameras
- Video transport

## Industrial, Scientific, Medical
- Ultrasound systems
- Motor controllers

## Consumer
- 3D television
- eReaders

# Developing FPGA applications

- Languages
  - VHDL, Verilog
  - OpenCL (C++)
- Software tools
  - Design
  - Simulation
  - Synthesis
  - Routing
- Hardware tools
  - Evaluation boards
  - Prototypes




Expensive and hard to scale

# Using FPGAs on AWS

# Amazon EC2 F1 Instances

- Up to 8 Xilinx UltraScale Plus VU9P FPGAs
- Each FPGA includes
  - Local 64 GB DDR4 ECC protected memory
  - Dedicated PCIe x16 connections
  - Up to 400Gbps bidirectional ring connection for high-speed streaming
  - Approximately 2.5 million logic elements, and approximately 6,800 DSP engines

| Model | FPGAs | vCPU | Mem (GiB) | SSD Storage (GB) | Networking Performance |
|---|---|---|---|---|---|
| f1.2xlarge | 1 | 8 | 122 | 470 | Up to 10 Gigabit |
| f1.16xlarge | 8 | 64 | 976 | 4 x 940 | 20 Gigabit |

# Deploy faster wherever you like

**18** Regions – **54** Availability Zones


aws

# The FPGA Developer Amazon Machine Image (AMI)

- AWS FPGA SDK
  - Amazon FPGA Image (AFI) Management Tools
  - Linux drivers
  - Command line
- AWS FPGA HDK
  - Xilinx SDAccel 2017.1
  - Free license for F1 FPGA development
  - Supports VHDL, Verilog, OpenCL
  - Design files and scripts required to build an AFI
  - Shell: platform logic to handle external peripherals, PCIe, DRAM, and interrupts
- Run simulation and design on a c4 to save money!

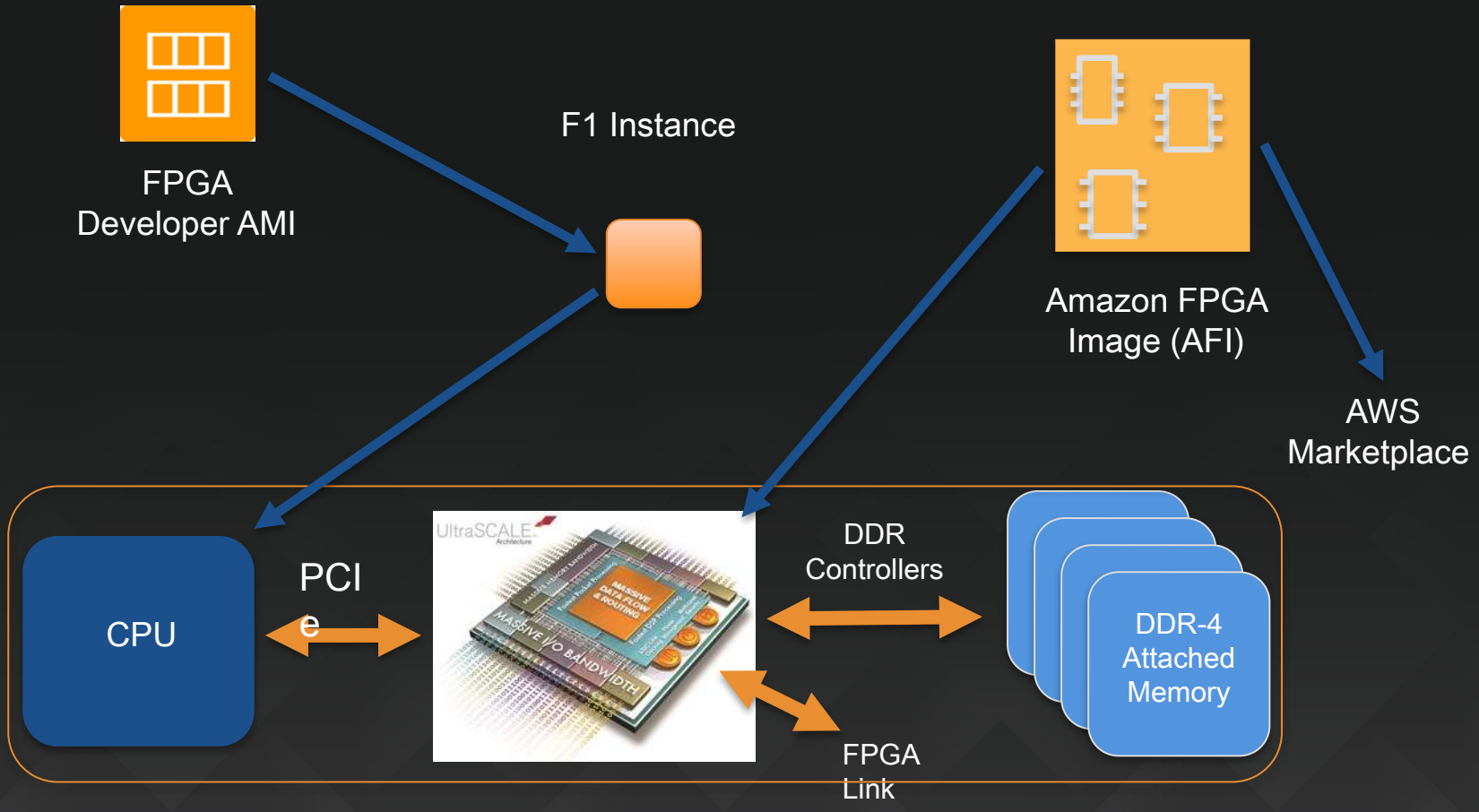
https://aws.amazon.com/marketplace/pp/B06VVYBLZZ
https://github.com/aws/aws-fpga

# Benefits of Amazon EC2 F1

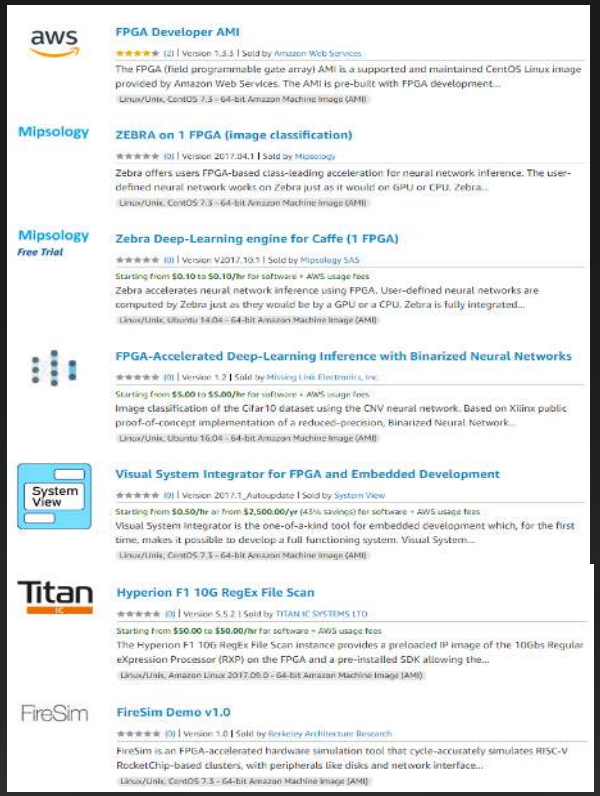
- Business model
  - Turns expensive HW appliance to cost effective, pay-per-use SaaS / API
  - Takes sales cycles / evaluations from months to hours
  - Scales to millions of AWS users

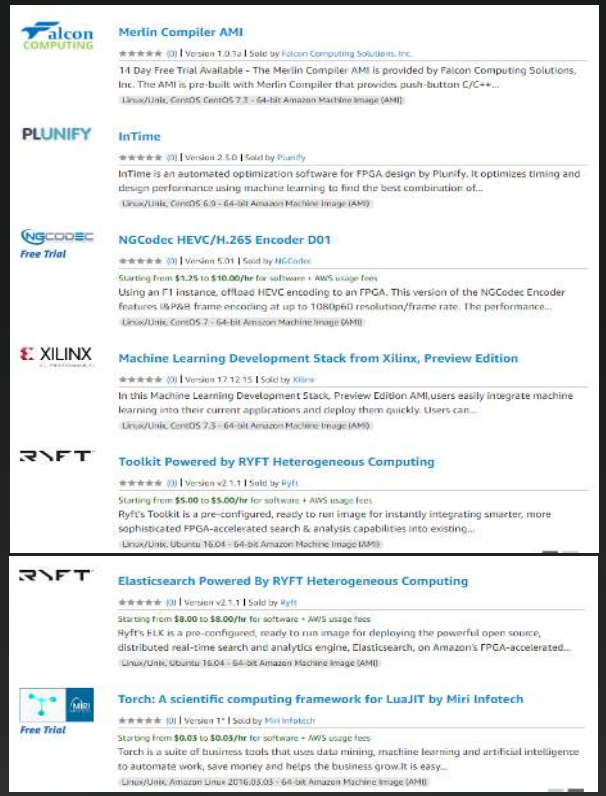
- Development model
  - Access to Cloud-based SW development tools
  - Access to fastest / best FPGA from anywhere
  - Applications can elastically scale to as many accelerators as needed

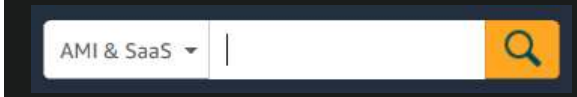
# FPGA Acceleration Using F1 instances

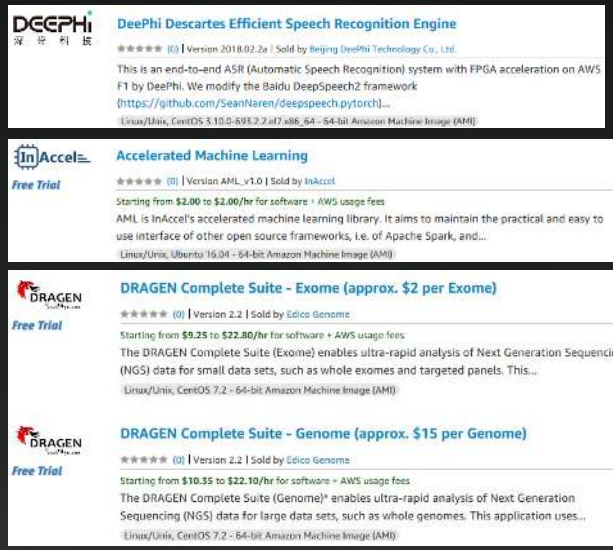


F1 Applications



# Customer use cases

# Compelling Use Applications on Amazon EC2 F1

**Machine Learning Inference**
Speech recognition

**40x**

DEEPHI 深鉴科技

**Video Streaming**
Frame rate for HEVC encoding

**10x**

NGCODEC NEXT GENERATION VIDEO COMPRESSION

**Genomics**
20 min vs. 33 hours for whole genome analysis

**100x**

edico genome

**Big Data Analytics**
40 min vs. 60 hours for logfile query

**90x**

RYFT™ ACTIONABLE INTELLIGENCE FROM COMPLEX DATA

aws

# Case study: Edico Genome

edico genome


- Genome diagnosis to treat critically ill newborns
- Analytics reduced from 1+ day to 20 minutes
- Dynamically reconfigures for patient-specific genomics acceleration

DRAGEN Genome Exome | DRAGEN Methylation Epigenome | DRAGEN Microbiome Metagenome | DRAGEN Joint Genotyping | DRAGEN Transcriptome RNAseq | DRAGEN Cancer Toolkit | DRAGEN Advanced Variant Caller | DRAGEN Structural Variant /CNVs | gatk GATK Best Practices

aws

# Case study: NGCodec


- Provider of UHD video compression technology
- Up to 50x faster vs. software H.265
- Higher quality video than *x265* 'veryslow' preset
  - Same bit rate
  - 60+ frames per second
- Lower latency between live stream and end viewing
- Optimized cost

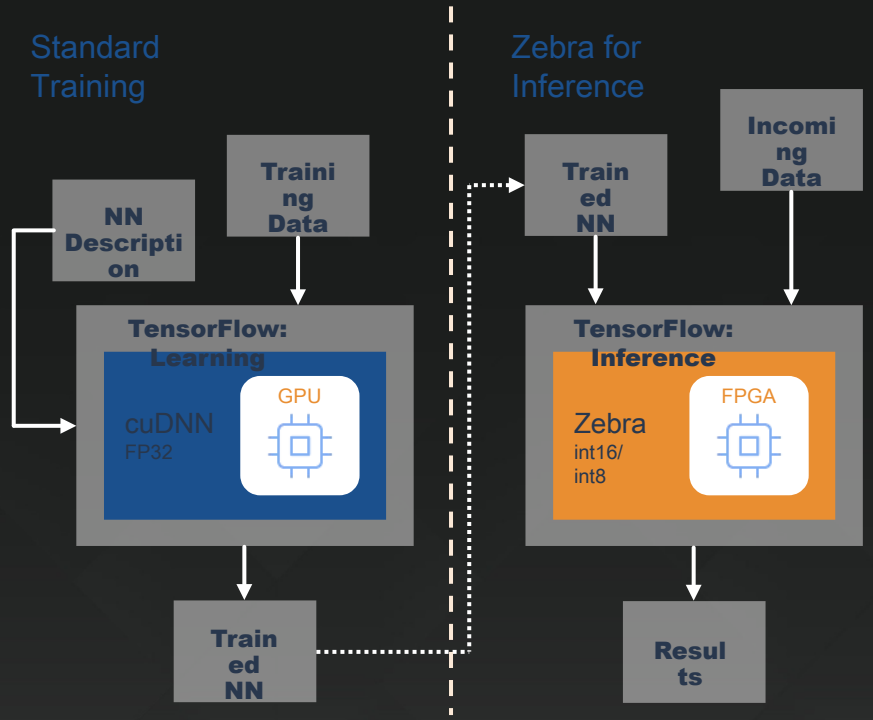
https://ngcodec.com/markets-cloud-transcoding/
https://aws.amazon.com/marketplace/pp/B074W1FPKR

aws

# Zebra from Mipsology

- Zebra is a Deep Learning Computing Engine

- Zebra allows to replace GPU/CPU by FPGA seamlessly so users can compute their neural networks faster, with lower power, at lower cost.



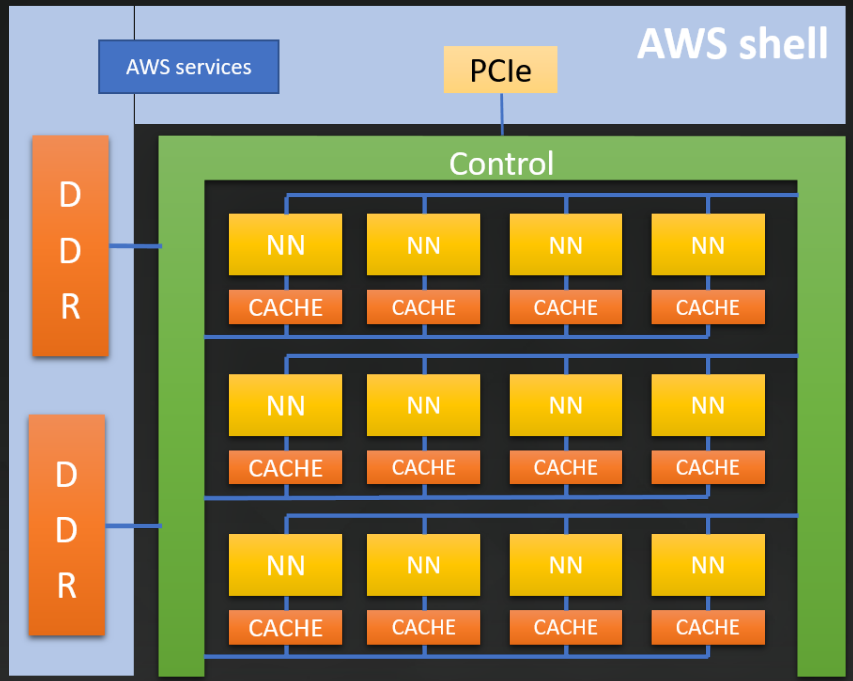
# Integration of Zebra in AWS EC2 F1 FPGA

- Zebra was designed to require only a PCIe and DDRs
- So, it was easy to port on AWS F1

R&D work
- Created a PCI-AXI bridge
- Created a AXI-2-DDR bridge
- Input clocks are provided by the shell

Challenge
- Delivering the same performances level as with a shell-free FPGA

Verification
- Used HDK BFMs for the shell and the external DDR memories in simulation



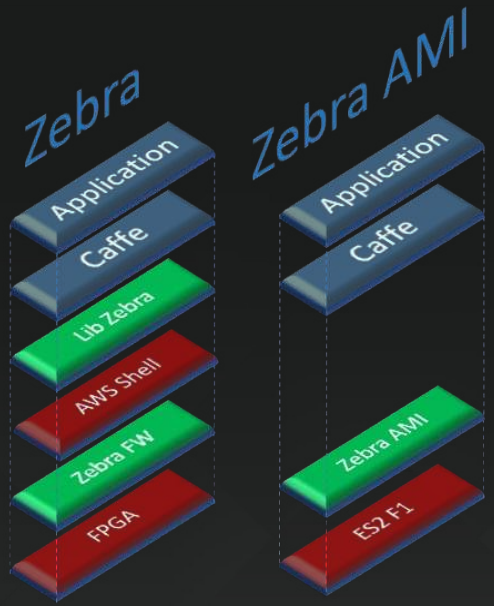
# Integration of Zebra in AWS EC2 F1 API

- Zebra is fully integrated in AI Frameworks, like TensorFlow, MXNet or Caffe
- From a user point of view, the Zebra AMI just makes the neural network runs on EC2 F1

R&D Work
- Used Deep Learning AMI from AWS as base
- Integrated SDK library to manage FPGA loading

Validation
- Scripts to download the ILSVRC images data base and run a CNN quality check (i.e. top1/top5)

Delivery on AWS
- ZEBRA on 1 FPGA (image classification)
- Zebra Deep Learning engine for Caffe (1 FPGA)



# Why Did We Provide Zebra on AWS EC2 F1

Porting Zebra on AWS is a small extra R&D effort

Mipsology does not need to deliver hardware for AWS users

Distributing Zebra on the Marketplace is fast and easy

Zebra can reach out all AWS users

Hardware is consistent and uniform

AWS users can try Zebra without upfront hardware cost

Access to the Zebra technology at a low cost

Scaling up and down is easy without upfront NRE

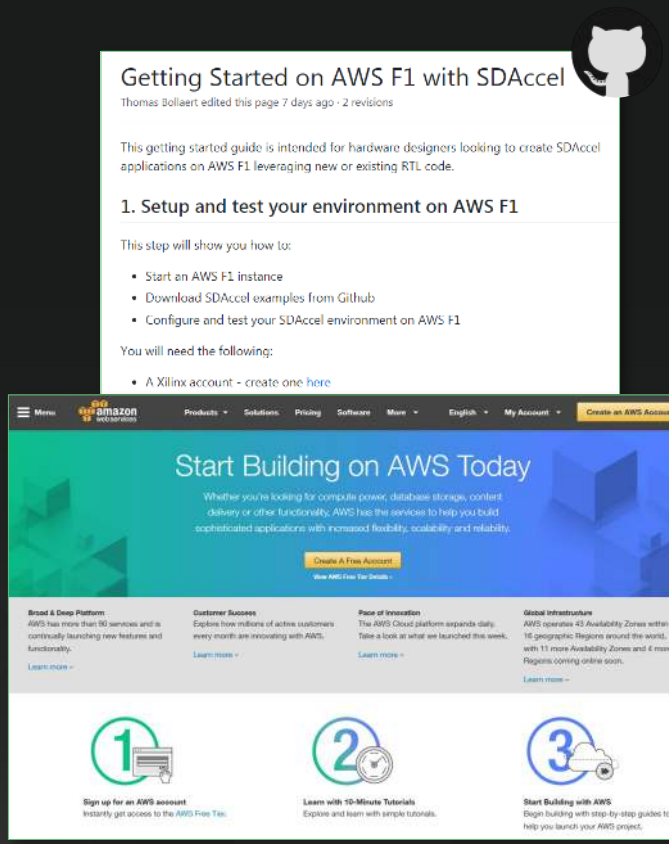
High quality of hardware without maintenance burden

FPGA Partially used by the shell

Compiling FPGA is harder

# Get Started on AWS F1 in 5 Simple Steps

- Onboarding program with step-by-step instructions, online training and Github examples
- Setup and test your AWS environment
- Run your first SDAccel hello world!
- Build your SDAccel knowledge
- Practice and Experiment
- Install and run SDAccel on your own machine



# Resources

https://aws.amazon.com/ec2/instance-types/f1
https://aws.amazon.com/ec2/instance-types/f1/partners/

https://github.com/aws/aws-fpga
https://github.com/aws/aws-fpga/blob/master/SDAccel/README.md

https://www.xilinx.com/
https://github.com/Xilinx/SDAccel_Examples/wiki/Getting-Started-on-AWS-F1-with-SDAccel-and-RTL-Kernels

http://www.mipsology.com/
https://aws.amazon.com/marketplace/seller-profile?id=904a5b3b-2c57-476f-95c2-3c4aeaa3ab8c

# Thank you!

Julien Simon, Principal Evangelist AI/ML, AWS
Ramine Roane, Senior Director, Product Management, Xilinx
Sébastien Delerse, Co-founder, Mipsology

aws XILINX ALL PROGRAMMABLE™ Mipsology