The Future of FPGAs Needs Open Middleware Now

Paul Chow and Great Team

High-Performance Reconfigurable Computing Group
Department of Electrical and Computer Engineering
University of Toronto

May 6, 2020

CONTEXT

Cloud vs HPC

- Somewhat different in how they are used
- For this discussion, assume they are the same

It's about Computing

- Using FPGAs for computing in the data center
- Not infrastructure in the data center

OUTLINE

- Observations about getting into the data center
- What is success for FPGAs?
- What do we need to become successful?
- UofT Galapagos middleware
- UofT AIgean application layer
- Where should we go from here?

IN THE BEGINNING

- Many, many papers showing FPGAs can accelerate applications and require less power
- No proof this works at scale for a data center
  - Most researchers don't have a data center!

Catapult at ISCA 2014

A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services

Andrew Putnam Adrian M. Caulfield Eric S. Chung Derek Chiou[1]
Kypros Constantinides[2] John Demme[3] Hadi Esmaeilzadeh[4] Jeremy Fowers
Gopi Prashanth Gopal Jan Gray Michael Haselman Scott Hauck[5] Stephen Heil
Amir Hormati[6] Joo-Young Kim Sitaram Lanka James Larus[7] Eric Peterson
Simon Pope Aaron Smith Jason Thong Phillip Yi Xiao Doug Burger

Microsoft

- FPGAs almost double performance
  - 10% power
  - 30% cost
  - 0% volume – card in existing server

It gets better!

- December 2015 – Intel closes acquisition of Altera
  - FPGAs are a legitimate computing technology!

- Early 2017 – Amazon announces F1
  - FPGAs have made it to the public cloud

Fantastic!!

HAVE FPGAS REALLY MADE IT?

- What are the killer apps and commercial successes?
- Are FPGAs just for video, bioinformatics, finance and ML?

What does the rest of the world think?

Jensen Huang says…

The really exciting thing right now is not to build yet another server. The exciting thing for the world is the server is not the computing unit anymore. The datacenter is the computing unit. You are going to program a datacenter, not a server.

The onion, celery, and carrots – you know, the holy trinity of computing soup – is the CPU, the GPU, and the DPU (SmartNIC). These three processors are fundamental to computing.

Nvidia CEO, TheNextPlatform, April 27, 2020

From the SIGARCH blog

Hardware researchers are proposing a large number of specialized chip architectures and the corresponding scheduling schemes, while software developers are optimizing deep learning frameworks to maximize the utilization of both existing CPU/GPU platforms and new hardware like TPU.

Mingyu Gao, Computer Architecture Today, May 4, 2020

Anybody else think about FPGAs but us?

WHAT IS SUCCESS FOR FPGAS?

Be part of Jensen Huang's soup mix!

- Be in the conversation, just one of the crowd
- Write your code and pick the best device for execution

Make application development easier

Okay, the tools still suck…

But, we can build a better environment

Get more people using FPGAs

In the beginning GPUs were a bit odd, but massive effort building infrastructure has made them successful

What Makes Software Successful

- Easy to build applications that can be platform agnostic
- Easy to deploy an application on any platform
- Scalable – we are talking about data centers
- Open source platforms

Challenges

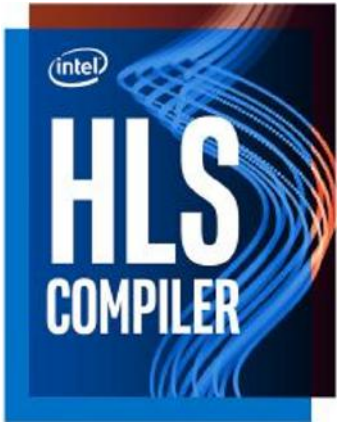
- We're mostly hardware designers
  - It's a culture thing
- Happy with one-off designs
  - Build it, ship it, start over
- Reusable IP is really a myth
  - Okay, if you have a lot of technical support ($$$$)
- Open source reusable free IP

What do we need? Putting it simply

Again, software is the model

WHERE DO WE STAND?

Programmability



Great start ☺

Necessary but not sufficient

Easy to Build Applications

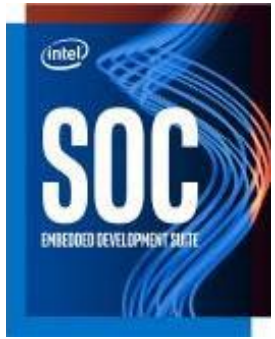
MAXELER Technologies
MAXIMUM PERFORMANCE COMPUTING

SDAccel™ Environment

SDSoC™ Environment

These are computing environments ☺

- Hardware abstraction
- Runtimes

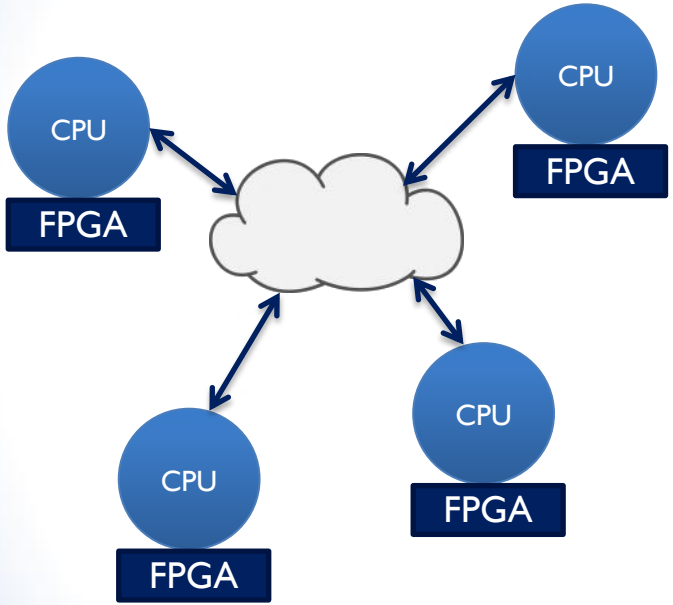


Portability

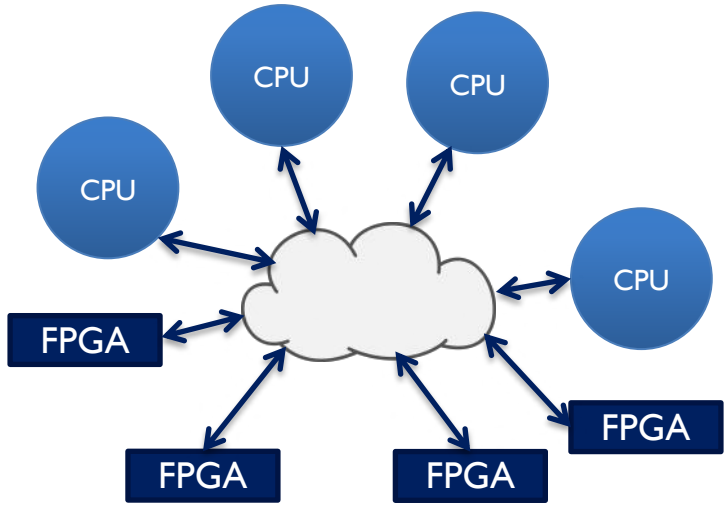
Nice try! ☹

Also, vendor-specific HLS, and vendor-specific computing environments not helping

Scalability: It's about Architecture



Accelerator - Main thinking today

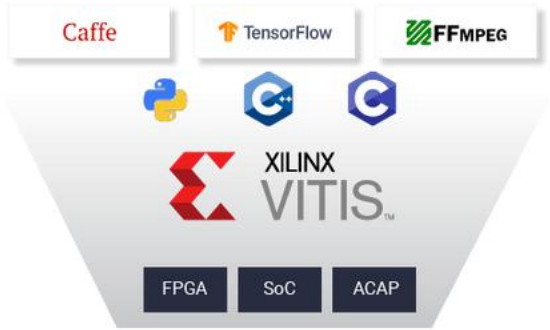


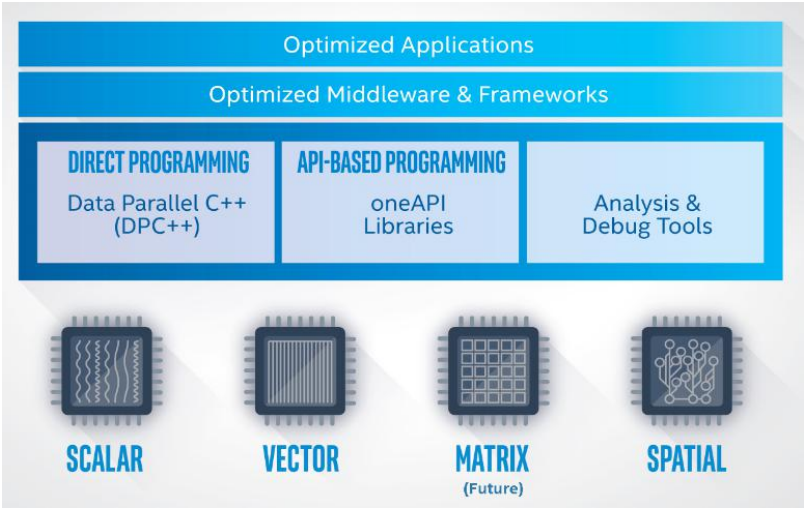
Peers Microsoft, cloudFPGA, UofT Galapagos

And the tools

- Vendor computing environments do not support scaling
- OpenCL, by definition, assumes a host and accelerators

Open Source

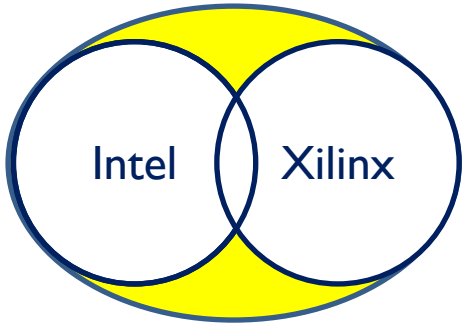




Sort of

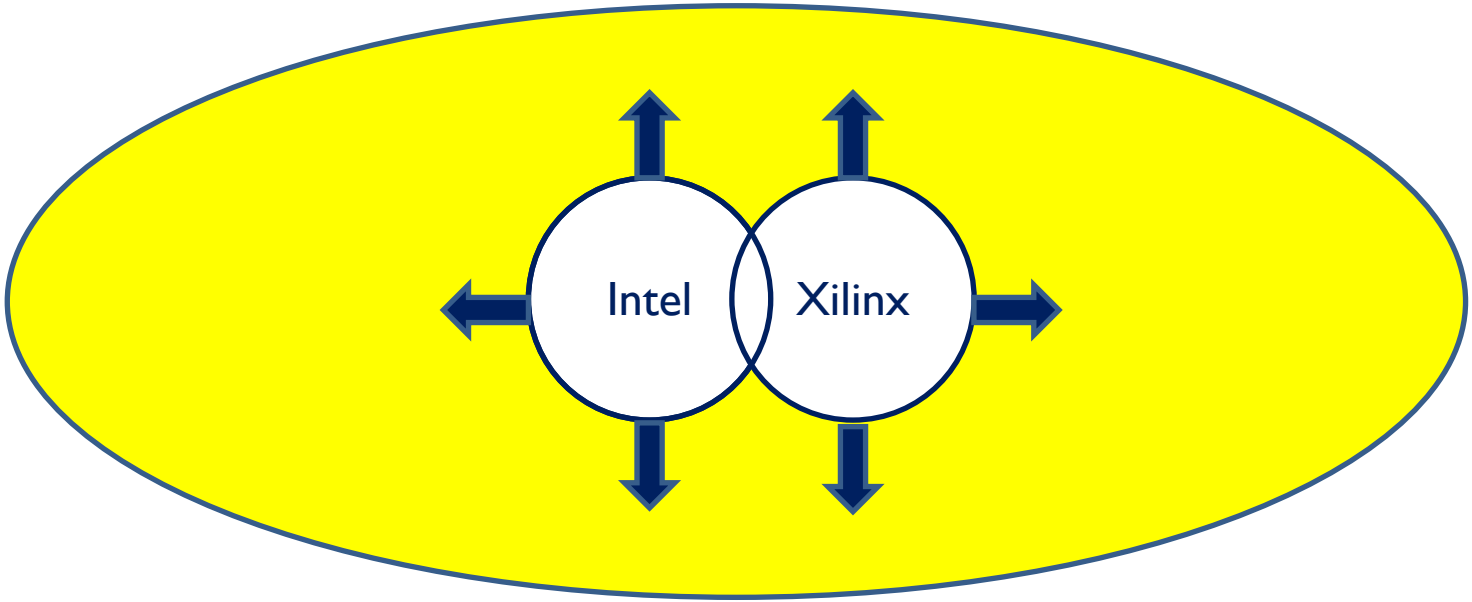
WHAT ARE WE TRYING TO ACCOMPLISH?

FPGA Market



Now

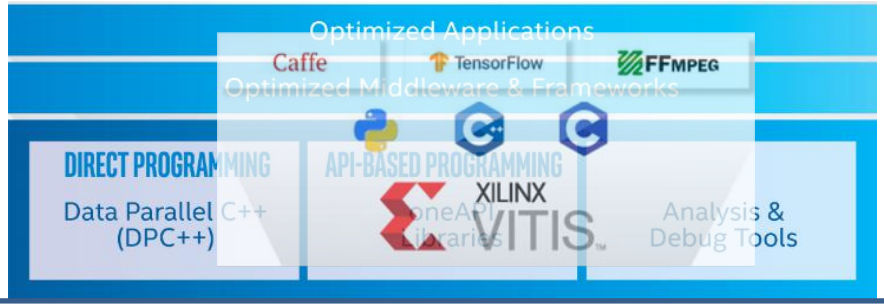
FPGA Market



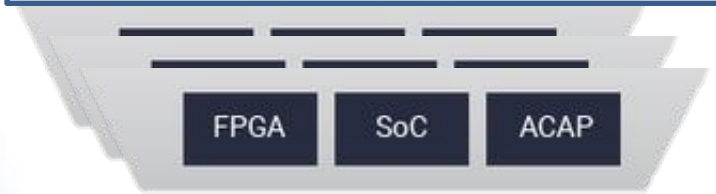
Success!

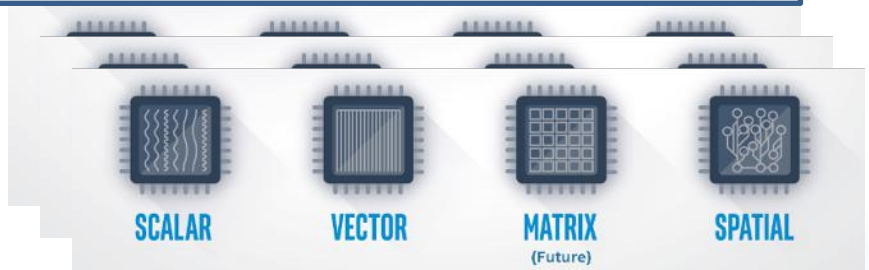
WHAT'S NEEDED TO ACHIEVE SUCCESS

- Work together on the common environment that everyone needs
  - HLS, libraries, toolkits
    - Vendors don't make money on these anyways
  - Vendors + users → Open source development
    - Vendors must support this
  - Linux and gcc are key drivers of software today
- Compete on the devices


Optimized Applications
Caffe
TensorFlow
FFMPEG
Optimized Middleware & Frameworks
DIRECT PROGRAMMING
Data Parallel C++
(DPC++)
API-BASED PROGRAMMING
XILINX
VITIS
Analysis &
Debug Tools

Middleware


FPGA
SoC
ACAP


SCALAR
VECTOR
MATRIX
(Future)
SPATIAL

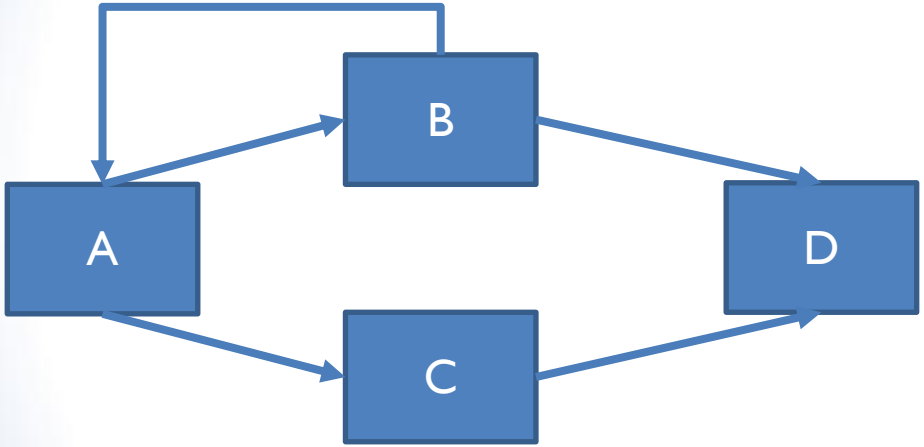
Middleware

- Vendor-neutral abstraction layer for applications, tools and other platforms
- Let's work together on this middleware
  - openRole is an example of an important component
- Support multi-FPGA
  - Better if assume heterogeneity in general
  - Scalability – Current vendor abstractions do not scale

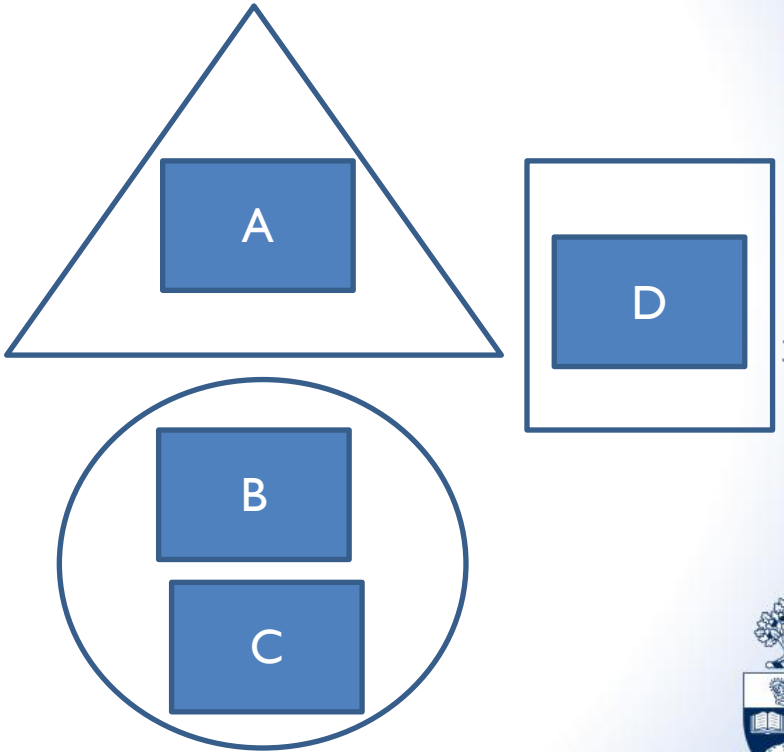
THE GALAPAGOS MIDDLEWARE

The Underlying Model

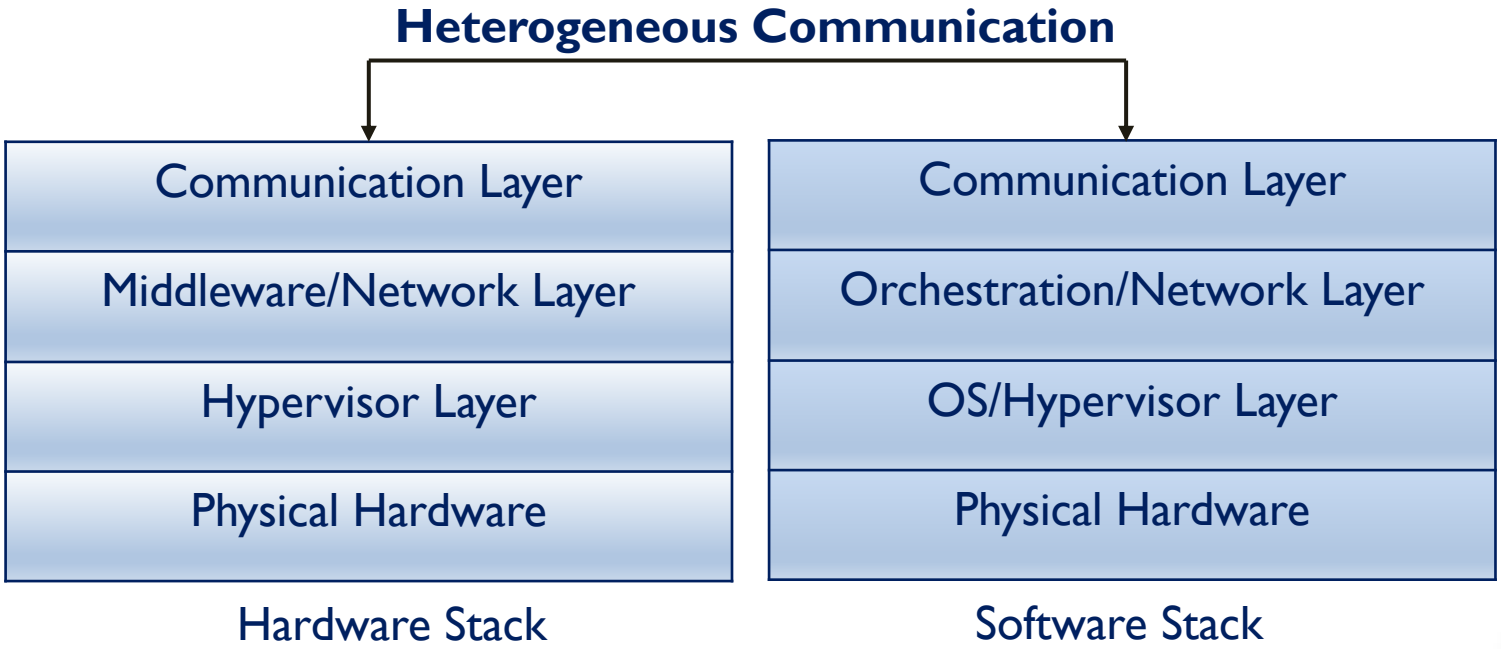
Distributed Flow Diagram



Heterogeneous Data Center

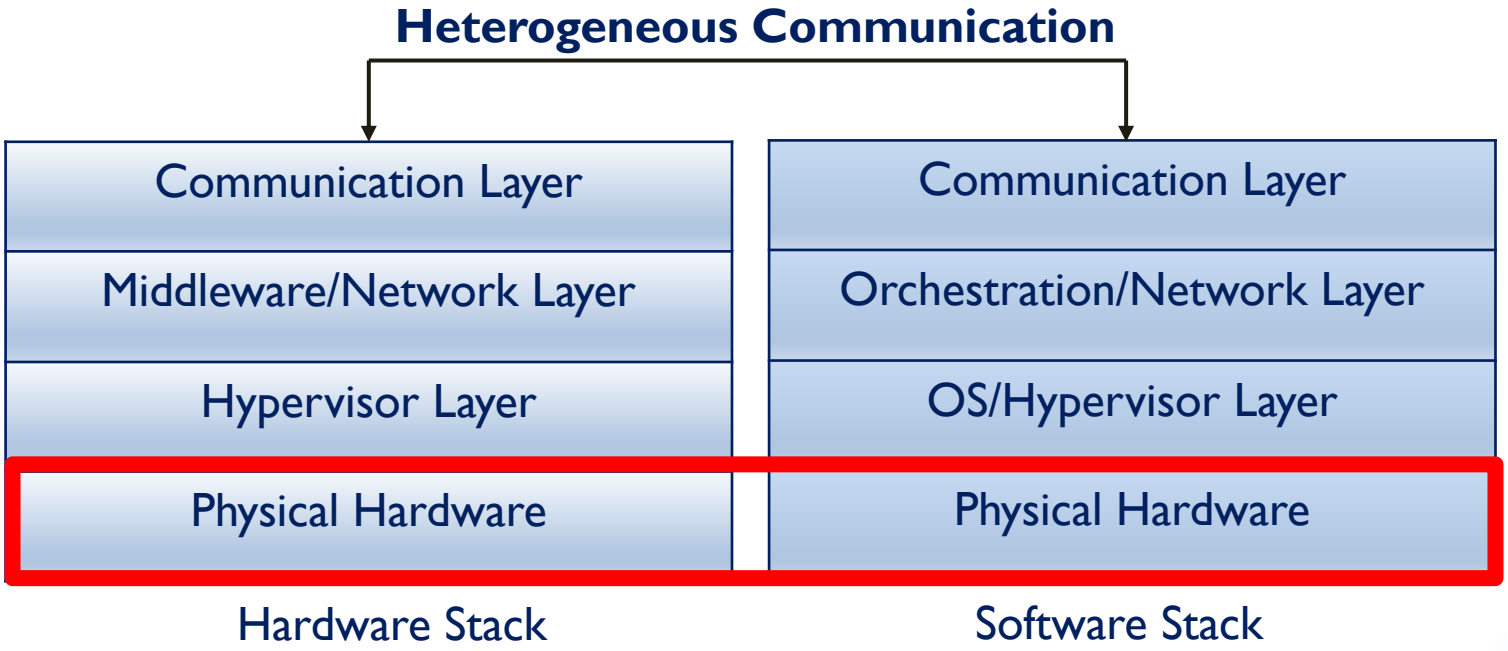


Heterogeneous Abstraction Stack

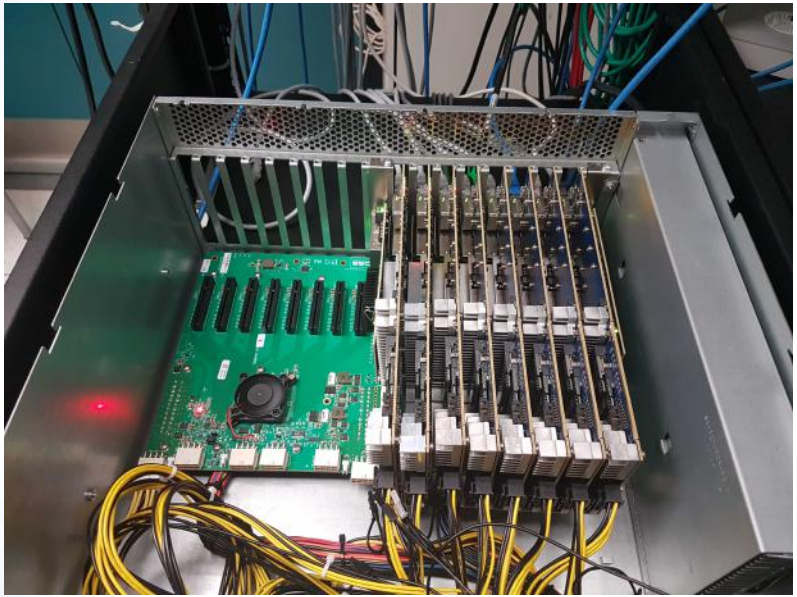


Galapagos

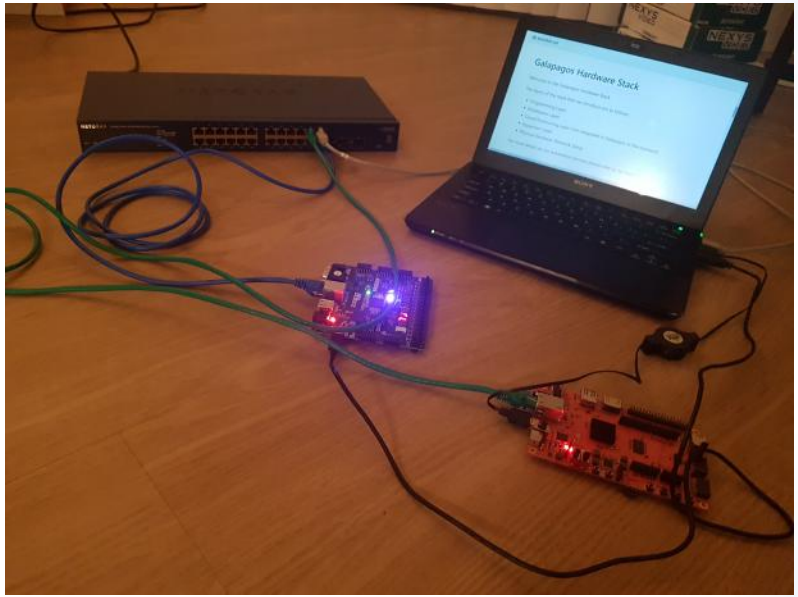
Heterogeneous Abstraction Stack



Our Datacenters

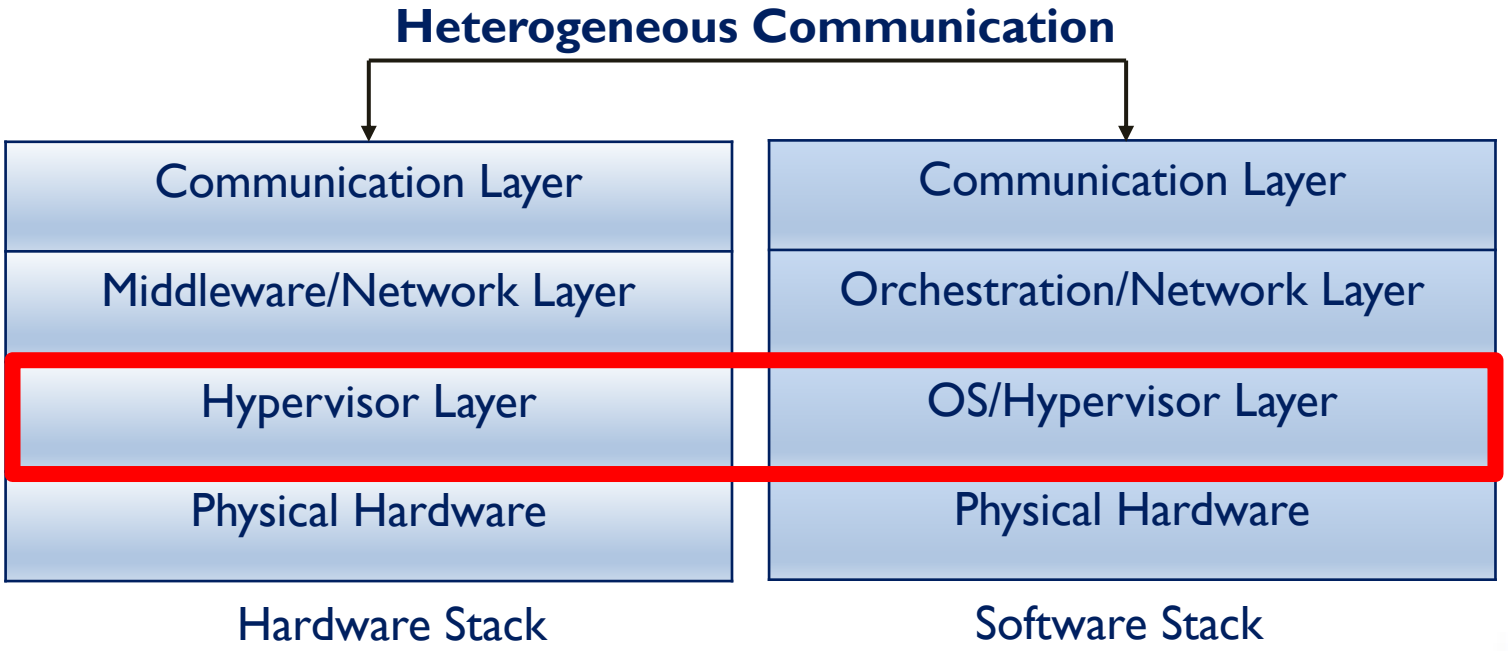


MPSoC Boards Connected via 100 GB/s Cables
~$100 000



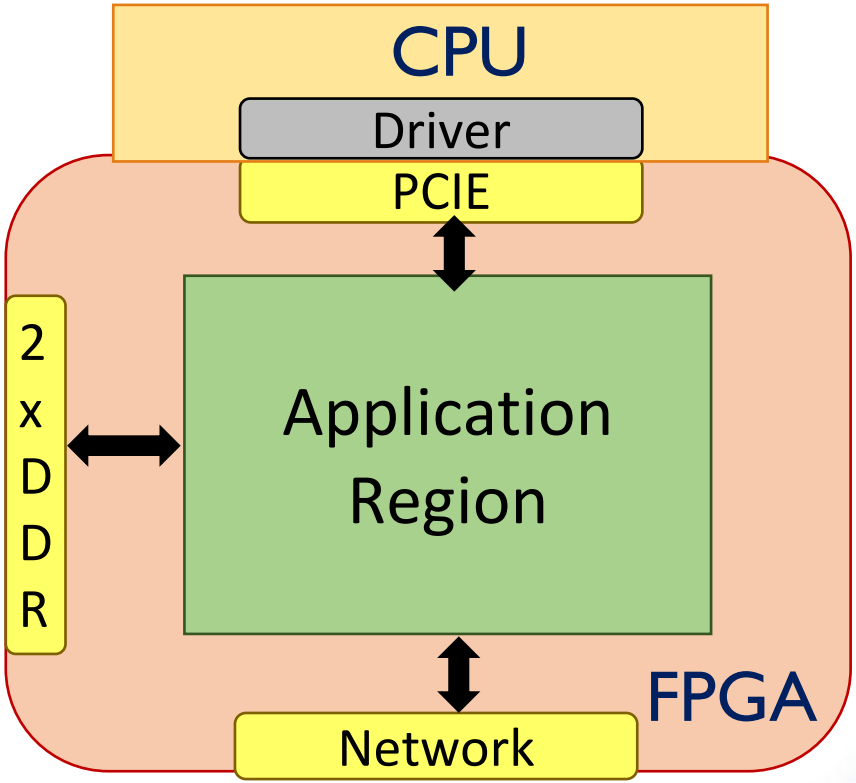
Nexys, Pynq, Laptop, 1G switch
~$2000

Heterogeneous Abstraction Stack

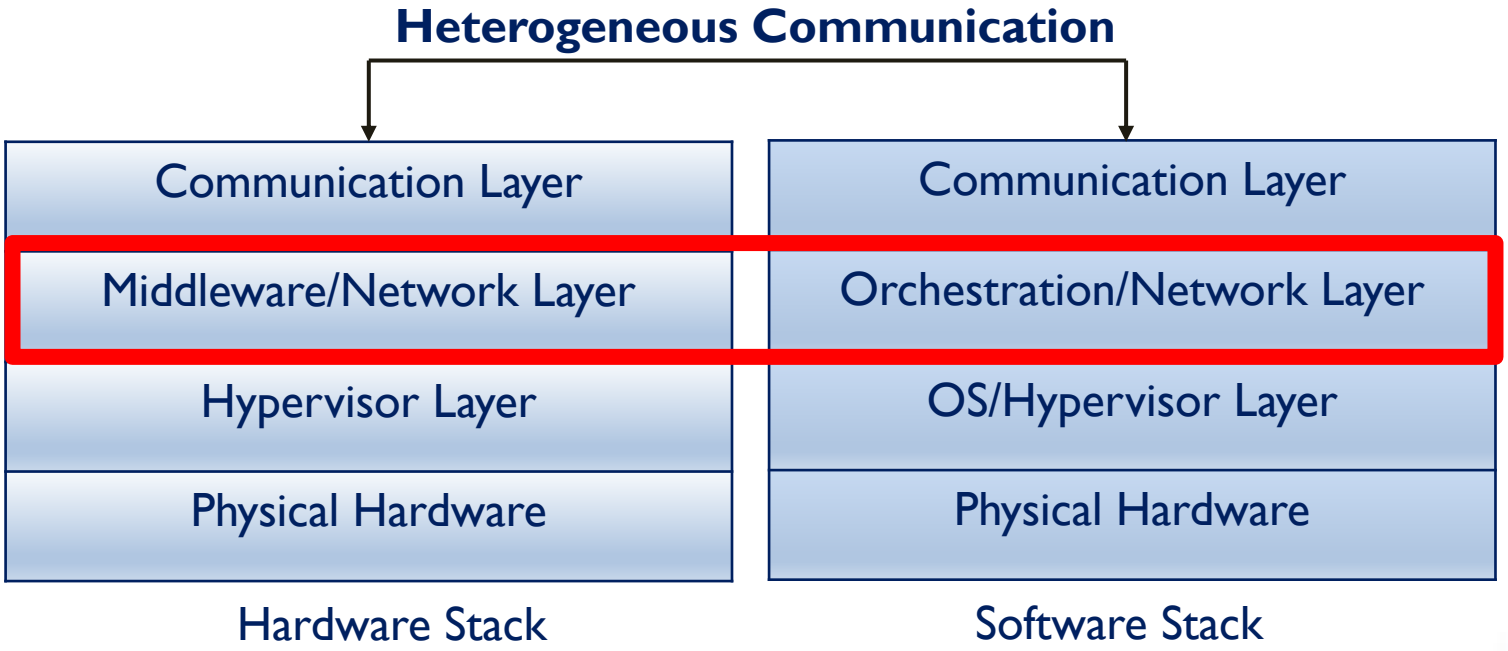


Galapagos Hypervisor

- The "shell"
- Abstracts all the I/O interfaces
- One for each board type: 8K5, Sidewinder, U200, U250



Heterogeneous Abstraction Stack

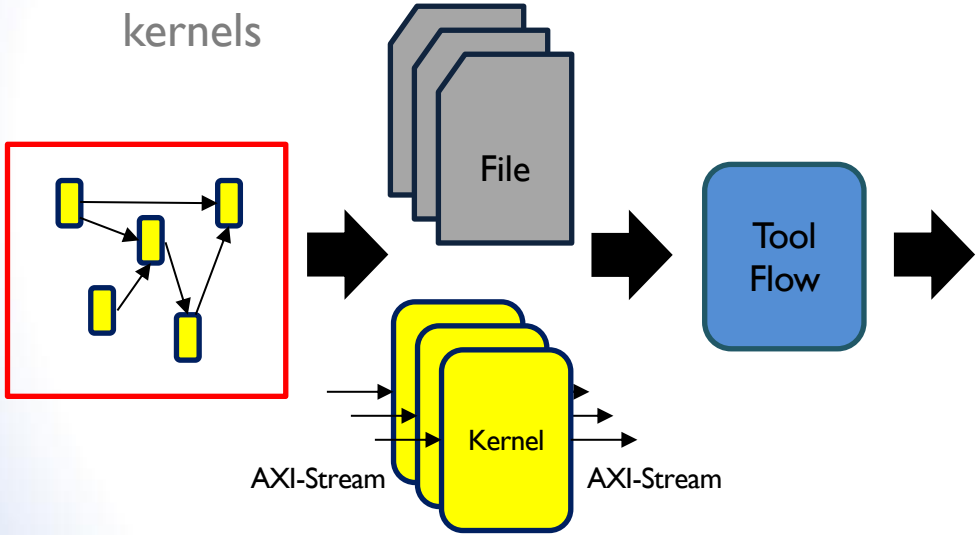


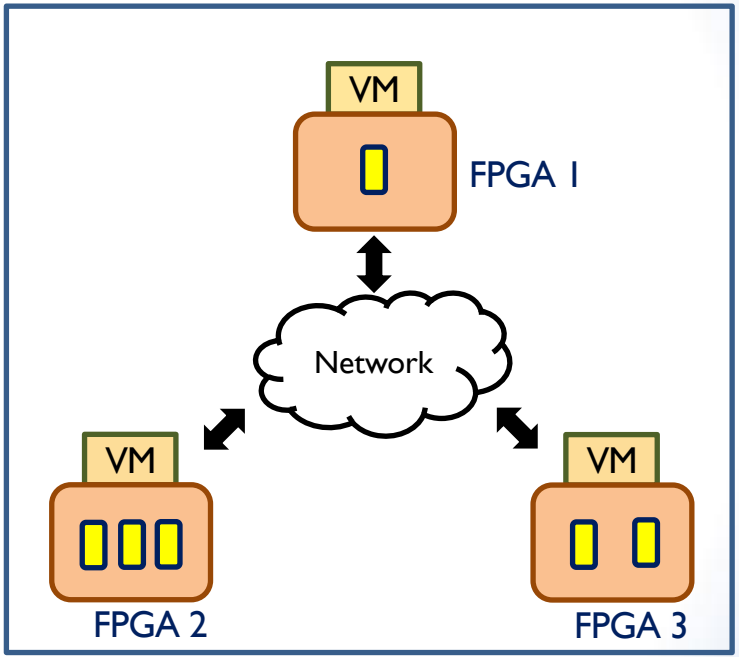
Middleware

- This layer refers to how we orchestrate clusters of resources
  - Includes FPGAs and CPUs
- Orchestration includes automating the connections between resources and providing handle to entire cluster

Galapagos: Middleware Layer

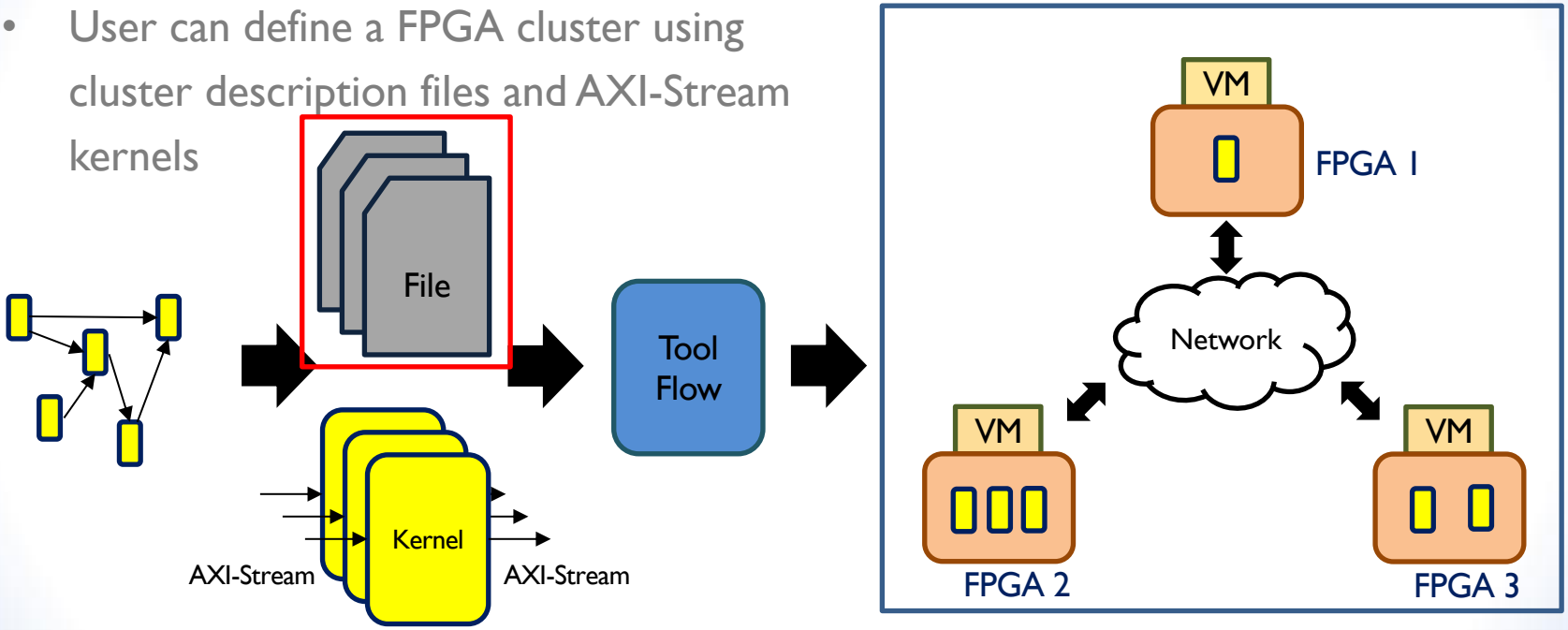
- User can define a FPGA cluster using cluster description files and AXI-Stream kernels





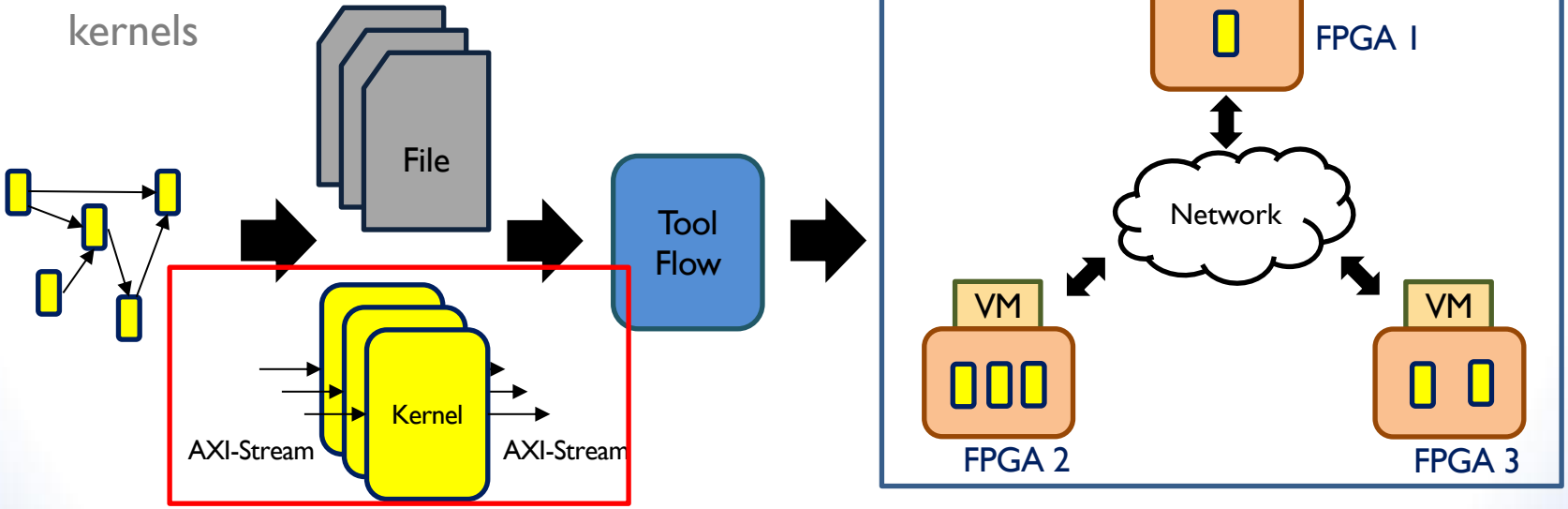
Galapagos: Middleware Layer

- User can define a FPGA cluster using cluster description files and AXI-Stream kernels



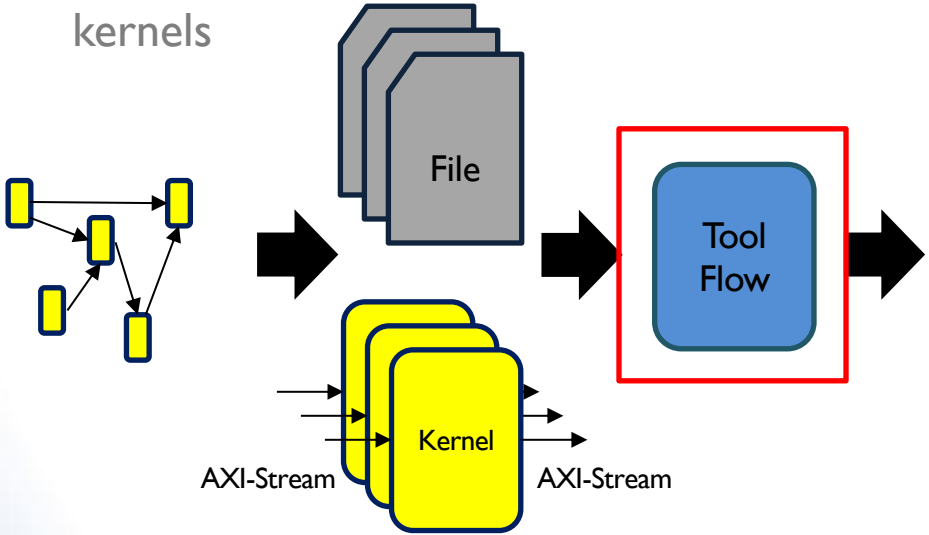
Galapagos: Middleware Layer

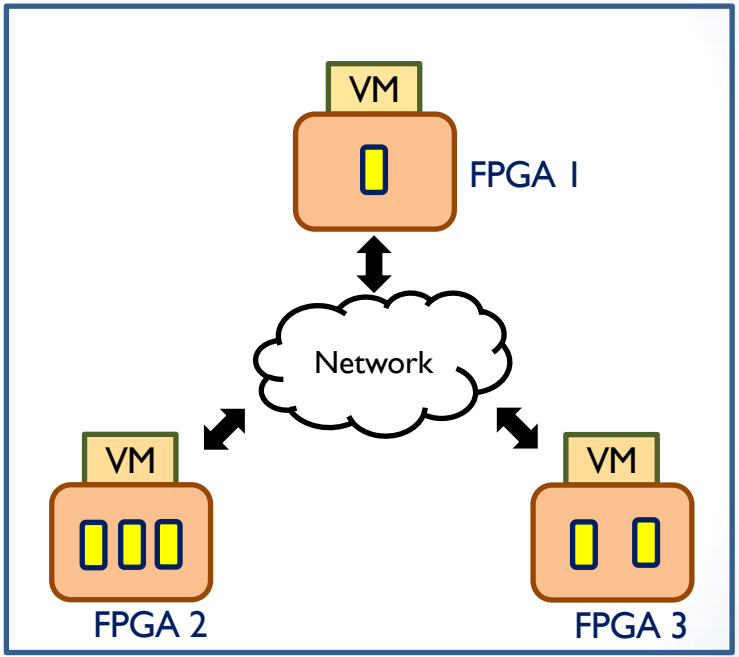
- User can define a FPGA cluster using cluster description files and AXI-Stream kernels



Galapagos: Middleware Layer

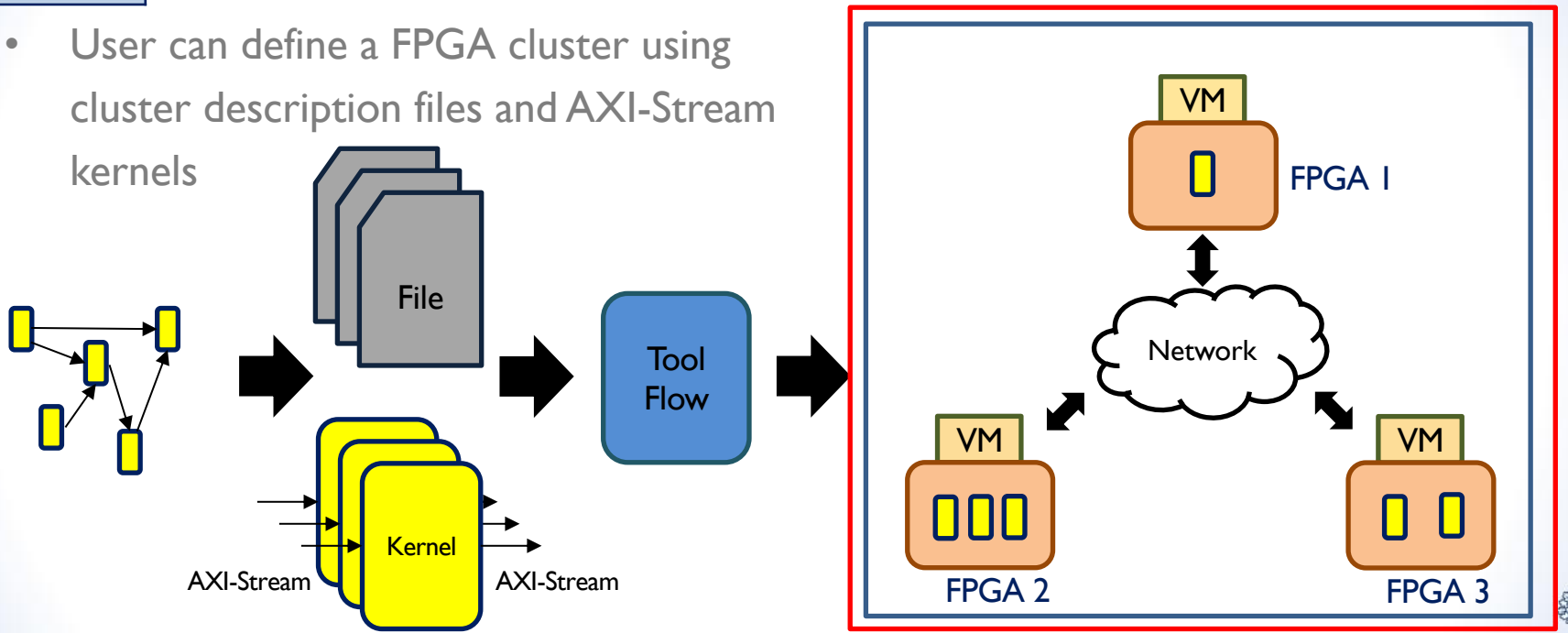
- User can define a FPGA cluster using cluster description files and AXI-Stream kernels





Galapagos: Middleware Layer

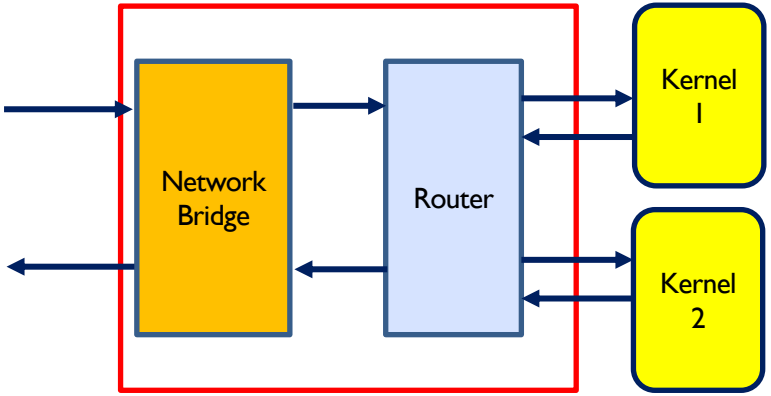
- User can define a FPGA cluster using cluster description files and AXI-Stream kernels



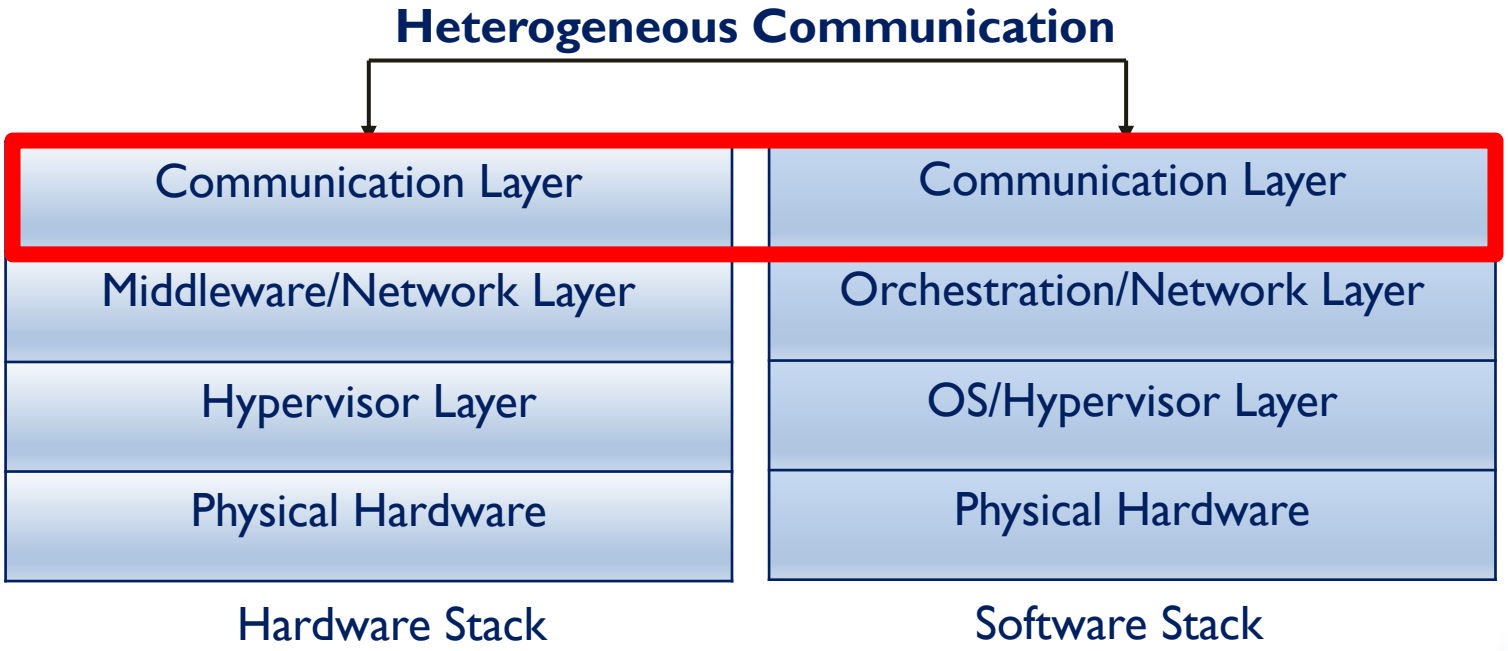
Galapagos: Middleware IP Blocks

Middleware generates additional IP blocks

- Can specify network protocols
- TCP, UDP, L2 eth, L1
- 10G, 100G

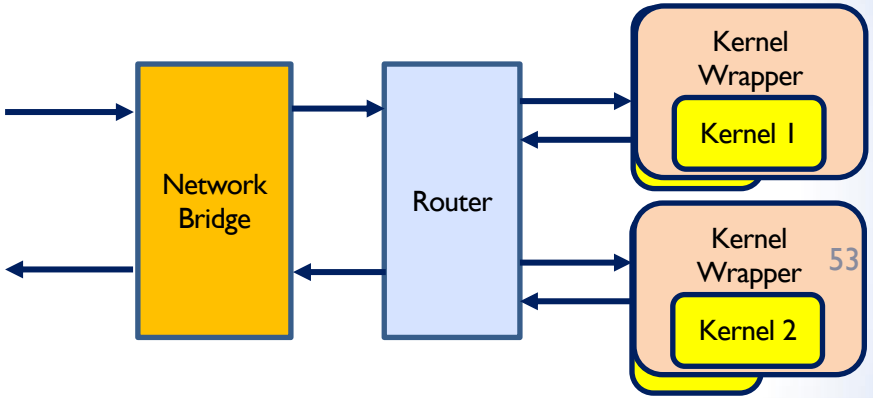


Heterogeneous Abstraction Stack



Example Communication Layer: libGalapagos

- Create software model of each component
- Galapagos software kernel object wrapper for HLS module
  - Functionally portable, uses same HLS code for software



BUILDING ON GALAPAGOS

Algean: An Open Framework for Machine Learning on a Heterogeneous Cluster

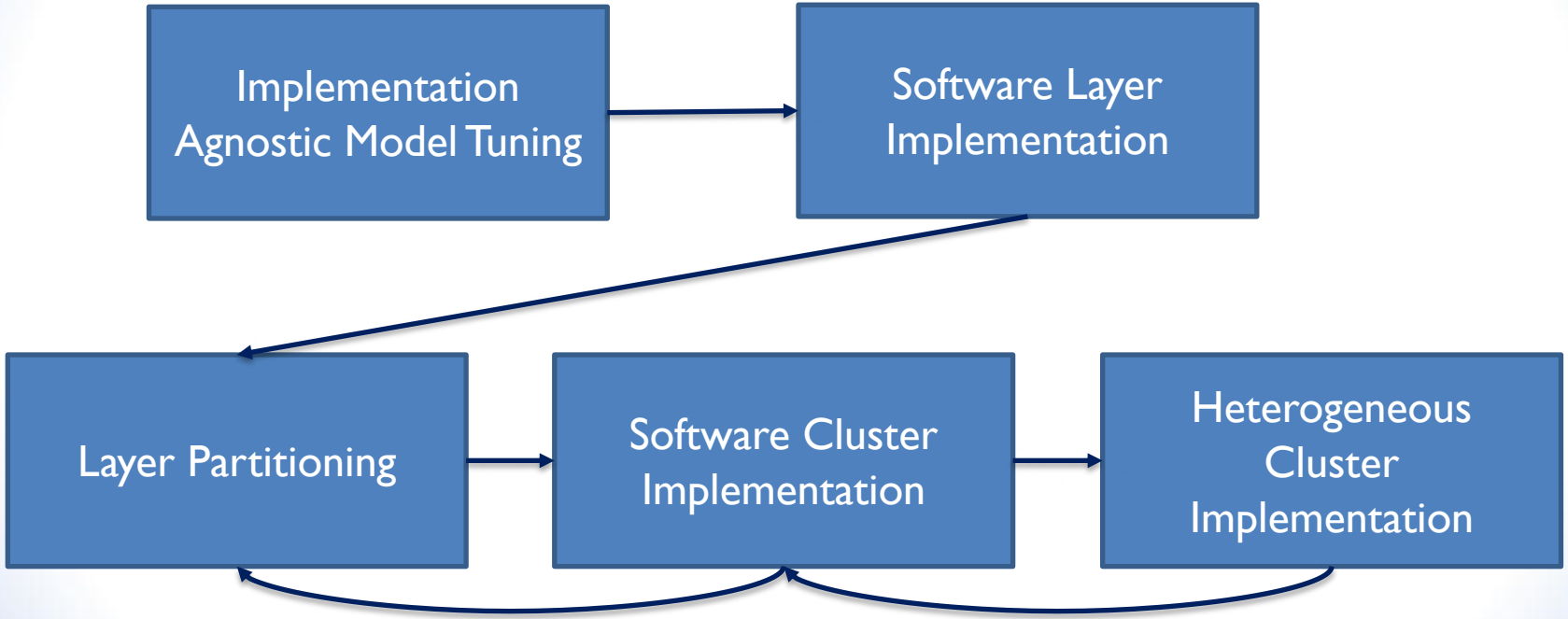
Naif Tarafdar[1], Giuseppe Di Guglielmo[2], Philip C Harris[3], Jeffrey D Krupa[3], Vladimir Loncar[4], Dylan S Rankin[3], Nhan Tran[5], Zhenbin Wu[6], Qianfeng Shen[1] and Paul Chow[1]

University of Toronto[1]
Columbia University[2]
Massachusetts Institute of Technology[3]
CERN[4]
Fermilab[5]
University of Illinois[6]

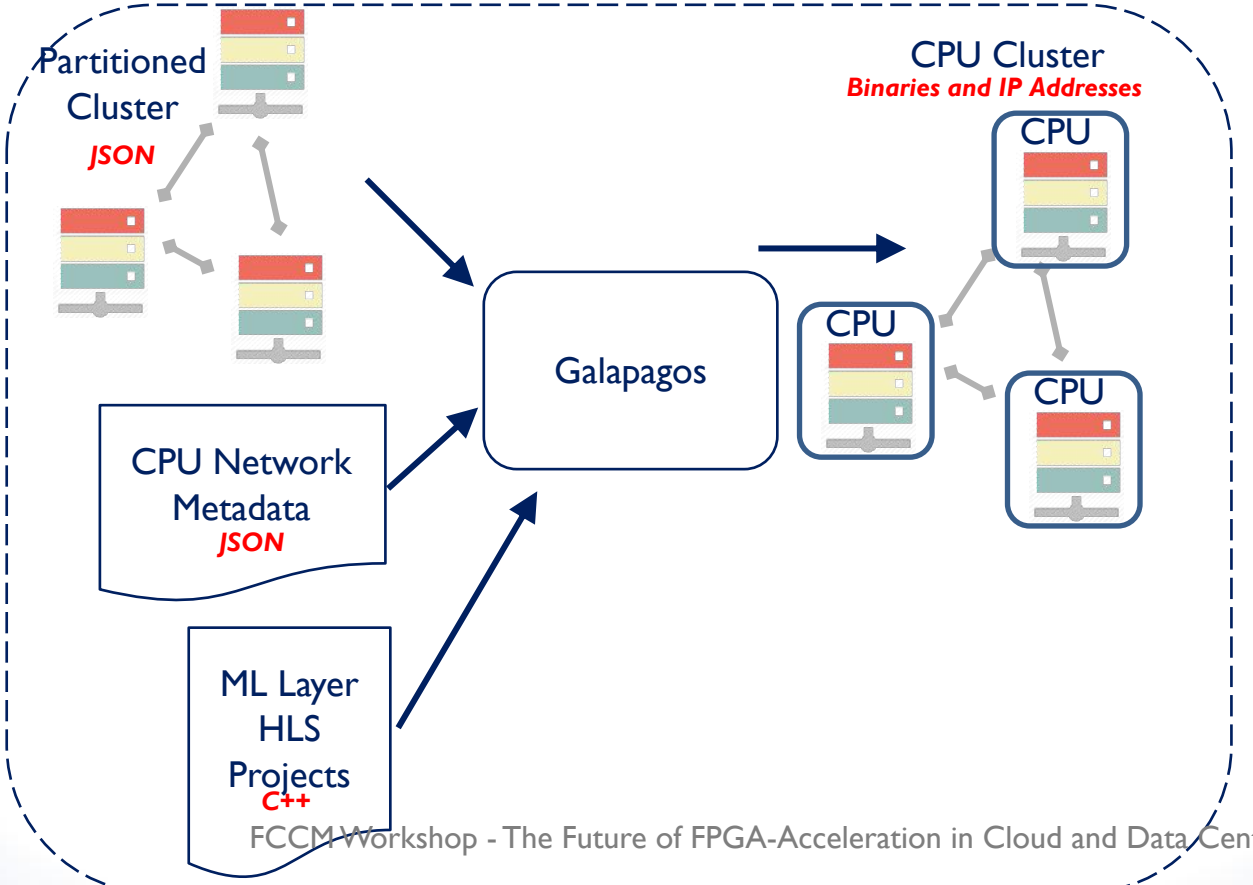
- Physicists doing particle detection at CERN

- Multi-FPGA/CPU neural net framework by leveraging and combining HLS4ML and Galapagos frameworks
- Tunable IP cores, flexible communication
- ML HLS IP cores deployed onto cluster of network connected FPGAs and CPUs
- Communication abstracted away from user

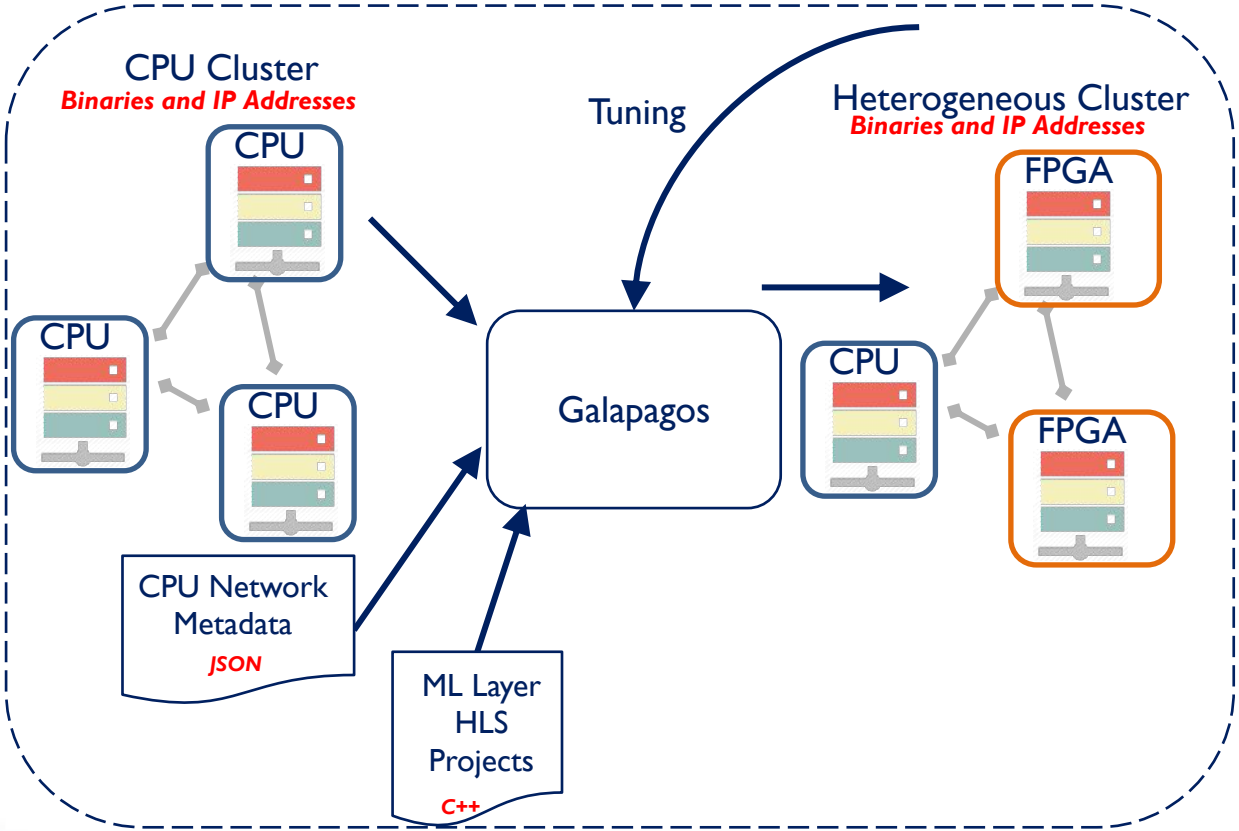
Algean Tool Flow



Software Cluster Implementation



Heterogeneous Cluster Implementation



Autoencoder: Results

- Split over 3 FPGAs in Algean, one FPGA in SDAccel

Device	Latency (ms)
CPU	3.3
GPU	2.5
SDAccel	0.24
Algean	0.08

- Algean through multi-device fabric can allow user to implement larger, higher performance cicruit

Observations

- Layered approach provides lots of flexibility to change things
  - Experimenting with communication protocols
  - In AIgean only added one small hardware core to middleware library
- Finding many other customers
  - Video data centers, NFV/VNF in telecom and 5G

WHERE NEXT?

Do you believe this story?

If you do, then we need to figure out how to collaborate on moving it forward so that FPGAs will have a future in the cloud and data center

Thanks for listening

More reading:

An Open Ecosystem for Software Programmers to Compute on FPGAs, FSP 2016; Third International Workshop on FPGAs for Software Programmers

pc@eecg.toronto.edu