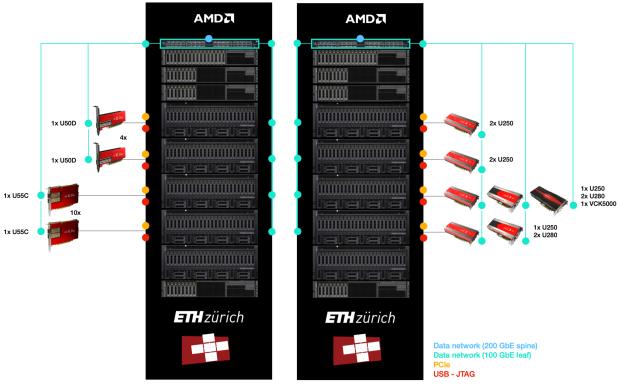


Open Source Resources and Infrastructure for FPGAs

Dario Korolija, Zhenhao He, Wenqi Jiang, Gustavo Alonso

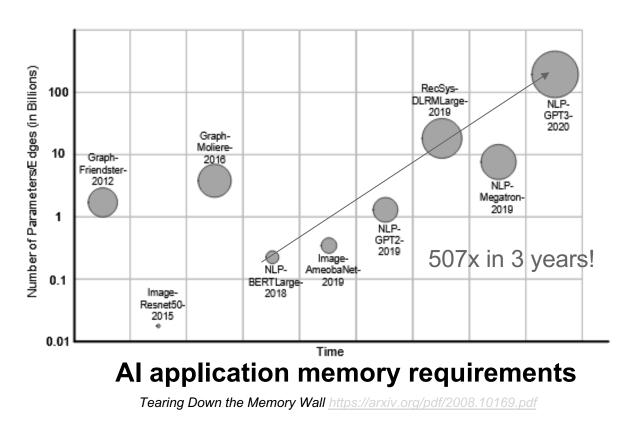


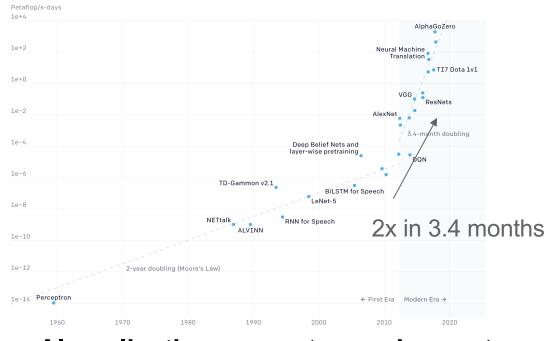
Copyright ETH Zürich - 2022



Trends in data centers

 Evolution of computation requirements in modern HPC and datacenter AI applications





Two Distinct Eras of Compute Usage in Training AI Systems

Al application compute requirements

Al and Compute https://openai.com/blog/ai-and-compute

- Dynamic workload;
- Heterogeneous devices;
- Distributed computing



Efficient data processing across data center is important...

Database example

SELECT * FROM T WHERE id=3

- A database will read the table from cloud storage
- Bring it all the way to the local memory, then to the CPU registers
- Just to throw away all tuples but 1
- Creates bottlenecks in storage, network, memory access, data buses, pollutes the caches, CPU cycles, etc.

As the amount of data to process keeps growing, its movement throughout the system has become one of the biggest bottlenecks and source of inefficiencies

\rightarrow Smart processing on distributed resources



Large deployment of FPGAs in the cloud



As consequences of growing compute/storage demands and requirements of efficiency...

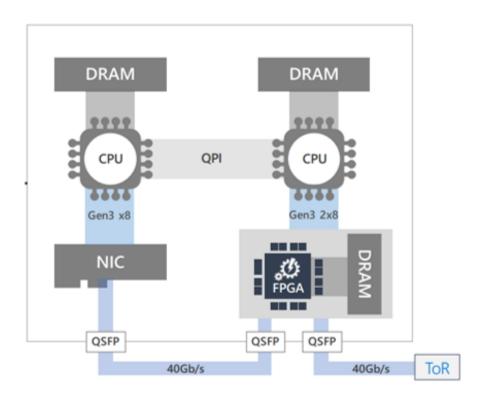
For specialization while keeping flexible for dynamic workloads...





Large deployment of FPGAs in the cloud – examples

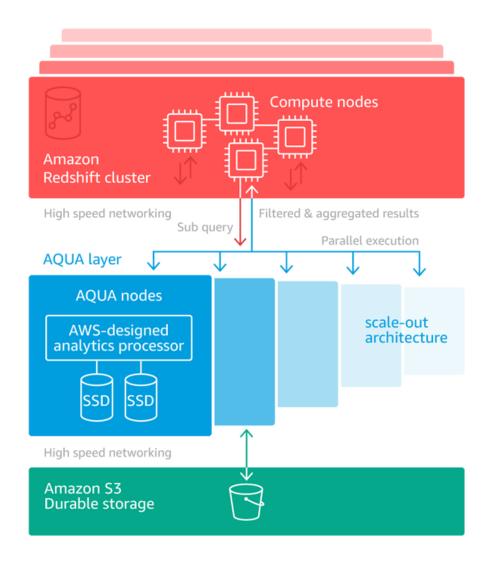
- FPGAs as in-network accelerator to meet growing compute demands
- Microsoft Catapult Project
 - https://www.microsoft.com/en-us/research/project/project-catapult/
 - Bump-in-the-wire architecture
 - Accelerate machine learning workload, e.g., CNN
 - Accelerate page rank
 - Also for QoS, storage acceleration ...
 - Production deployed in Bing





Large deployment of FPGAs in the cloud – examples

- FPGAs as smart accelerator for disaggregated resources
- Amazon AQUA
 - https://aws.amazon.com/blogs/aws/new-aqua-advanced-queryaccelerator-for-amazon-redshift/
 - Analytic engine with FPGAs
 - Pushing computation closer to data
 - Reduce CPU compute requirement
 - Reduce network traffic



Efforts from academia



Similar trends in academia to study how to place FPGAs in a larger system, for performance and efficiency. Just list a few...

Near-data processing:

KV-Direct (SOSP'17), Caribou (VLDB'17)

In-network processing

PANIC (OSDI'20), Corundum (FCCM'20) Strom (EuroSys'20), Farview (CIDR'22)

Distributed computing

FleetRec (KDD'21), FPDeep (TC'19)

| KV-Direct: | High-Performance | In-Memory | Kev-Value |
|------------|------------------|-----------|-----------|
| | | | |

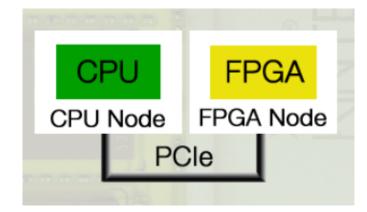
| Bc | | | | | ANTIFACT VALUARED Commun. | | | | |
|---|---|---|--|---|--|--|--|--|--|
| Yonş | PANIC: A Hi | gh-Performance Programmable | NIC for Mult | i-tenant Net | works | | | | |
| | | | - | | | | | | |
| | University e | | | | | | | | |
| ABSTRACT | | | | | | | | | |
| Performance of it to be of great im traditional object frastructure to sup in data centers. R of network bandw of most KVS fror NIC partly allevia | Abstract Programmable NIC a NIC platform tha co-resident applicat including hardware | FleetRec: Large-Scale Recommendation Inference on Hybrid GPU-FPGA Clusters | | | | | | | |
| by RDMA abstra | bedded processor c | Wenqi Jiang' Systems Group, ETH Zurich | Zhenhao Systems Group, | | Shuai Zhang Systems Group, ETH Zurich | | | | |
| grammable NICs | NIC design that can | Systems Group, ETH Zurich | Systems Group, | ETH Zurich | Systems Group, ETH Zurich | | | | |
| in-network proces | diverse offloads wh | Kai Zeng | Liang F | eng | Jiansong Zhang | | | | |
| high performance | multi-tenant isolatis | Alibaba Group | Alibaba O | Froup | Alibaba Group | | | | |
| extend RDMA pr | for offloads with va This paper press | Tongxuan Liu | Yong | 11 | Jingren Zhou | | | | |
| access to the main We develop sev | There are two new 1 | Alibaba Group | Alibaba C | | Alibaba Group | | | | |
| put and hide the | enable it to overcor | Thibaba Group | 7 110 10 1 | noup | rinousa ciroup | | | | |
| the NIC and the l | high-performance s | Ce Zł | Gustavo | | | | | | |
| tleneck. Combin | nects independent e | Systems Group | Systems Group | o, ETH Zurich | | | | | |
| KV-Direct to ach | hybrid push/pull pa performance isolati | ABSTRACT | | | Click-Through Rate | | | | |
| second, equivaler | parallel offload eng | We present FleetRec, a high-performance and s | | | | | | | |
| Compared with C | 100 Gbps FPGA-b | dation inference system within tight latency constraints. FleetRec Multi-Laver Persentron | | | | | | | |
| improves power e below 10 µs. Mo | overcomes the limi | takes advantage of heterogeneous hardware including GPUs and the latest FPGAs equipped with high-bandwidth memory. By disag- | | | | | | | |
| scalability with n | NICs. | gregating computation and memory to different | | Pertu | ve Interaction (e.g., Concetenation) | | | | |
| cards in a comm | 1. The second | and bridging their connections by high-speed | | 1 | | | | | |
| operations per sec | 1 Introduction | gains the best of both worlds, and can naturally a nodes to the cluster. Experiments on three pro- | | | Embedding Embedding | | | | |
| improvement ove | The gap between n | to 114 GB show that FleetRec outperforms optim | Dense Feature | Table 0 Table 1 Table N | | | | | |
| for a general-purg | CPU can produce a 66]. Emerging prog | by more than one order of magnitude in terms of | | | | | | | |
| | come this problem | achieving significantly lower latency. | | Input | > Lookup Indexee | | | | |
| Permission to make c | offloads that can be | ACM Reference Format: Wengi Jiang, Zhenhao He, Shuai Zhang, Kai Zeng, I | iong Europ Lioncomg | (Men | | | | | |
| personal or classroom made or distributed fo | These offloads, wh | Zhang, Tongxuan Liu, Yong Li, Jingren Zhou, Ce Zhang | | | | | | | |
| this notice and the full | the different layers | 2021. FleetRec: Large-Scale Recommendation Inferen | resentative deep recommendation m | | | | | | |
| of this work owned by credit is permitted. To | on the general pun throughput [32, 48, | FPGA Clusters. In KDD '21: ACM SIGKDD Conference ery and Data Mining, August 14–18, 2021, Virtual Evo | that we target to a | ccelerate. | | | | | |
| redistribute to lists, re | Many different c | NY, USA, 9 pages. https://doi.org/10.1145/nnnnnn.m | onnana | | e input feature vector consists of dense fea | | | | |
| permissions from perr | cases have been sho | 1 INTRODUCTION | | | e features (e.g., advertisement category) h sparse feature into a dense embedding v | | | | |
| SOSP '17, Shanghai, © 2017 ACM, 978-1- | to programmable ? | DNN-based recommendation inference compri | rer a home portion | | in sparse feature into a dense embedding v an embedding table. These vectors are | | | | |
| DOI: 10.1145/313274 | 47, 30, 36, 70, 69, | of the workload in data centers. Thus, it is crue | | combined with den | se features and fed to several fully-conne | | | | |
| | "silver bullet" offlo cases. Instead, we a | performance to serve these models efficiently. S | | (FC) layers before th | te model outputs the predicted CTR. Alth | | | | |
| | specify their own ch | can lead to instant economic benefits through mendation quality since more candidate items of | | | e architecture designs [1, 5, 6, 17, 18], mos ms are built around two major building b | | | | |
| | then merge these cl | same time frame; and (b) reduced energy consum | | | tables and the DNN classifier, thus sharin | | | | |
| | and run them on h | the improved inference efficiency. Figure 1 show | ws the architecture | inference challenges | | | | | |
| | vision, this paper p | of a classical deep recommendation model for the | Click-Through Rate | | b) Due to the embedding table architecture are recommendations, three challenges are in the second secon | | | | |
| | performance progra | Both authors contributed equally to this research. | | to build efficient in | ference systems for recommendations. | | | | |
| | | Permission to make digital or hard copies of all or part of th | is work for personal or | | e architecture becomes a performance b | | | | |
| | USENIX Associatio | characters to its inguited without for provided that copies are for profit or commercial advantage and that copies have this in on the first page. Capyrights for components of this work see must be however, advantage and works in provided. To copy to post on servers or to redistribute to lake, requires prior pro- pose XDD 22, Appendix 1-14, 2023. Unsue XDD 21, Appendix 1-14, 2023. Unsue XDD 22, Appendix 1-14, 2023. Unsue XDD 23, Appendix 1-14, 2023. Unsue XD 23, 2023. Unsue XD 24, 2023. Uns | not made or distributed stice and the full citation ned by others than ACM otherwise, or republish, | 64 dimensions) and to hundreds), the ei- because they induce low memory bandw performance. Even overhead if one reso | y size of each embedding vector (usually the large number of embedding tables mbedding table lookup operations are c e massive random DRAM accesses, leadi ridth utilization and significantly downg worse, these lookup operations result in rts to state-of-the-art machine learning ff | | | | |
| | | PROFESSION 1978-1-12222-2222-22121/00/01.0225.00 | | marks such as Tone | arElow and Dr.Tarah. For anomala man | | | | |

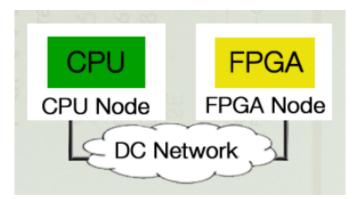


FPGAs in the context of cloud deployment is different...

From the previous examples, we see:

- FPGAs no longer viewed as slave accelerators
 - Connected to CPU host via PCIe
 - For traditional offload workload
- But as first-class citizen for data processing
 - Connected through DC network
 - Low latency
 - High throughput
 - Running DC communication protocol







FPGAs ideal for networking + computation

- Direct processing of network data
 - In contrast to GPUs (or other accelerators)
- Pipelined architecture fits network stream processing
 - Finite state machines
- Overlapping networking and computation
 - Store-and-forward not required
 - Batching not required
- Example: Microsoft Brainwave
 - https://www.microsoft.com/en-us/research/project/project-brainwave/
 - Real Time DNN
 - Low latency, high efficiency
 - Served in real production







Challenges to build applications on FPGAs

- Lack of open-source infrastructure for FPGAs
 - · Cloud infrastructure close-source, lacking testbed
 - Comparison to CPUs...
 - Rich APIs
 - Portability across platforms
 - Backward compatibility



• Lack of support from commonly available development framework

- Abstracting data movement through PCIe
- But not with network...
- Lack of traditional operating system abstractions
 - We are used to processes, threads ...
- Sometimes must first build the whole infrastructure before exploring in-network processing/distributed applications
 - Think about previous research...



Summary of Challenges

- Lack of open-source infrastructure
- Lack of traditional high-level abstractions and interfaces

Tutorial: Resources for Distributed Applications on FPGA Clusters

- Target: Facilitate practitioner and researcher exploring distributed applications on FPGA clusters
 - FPGA clusters (HACC)
 - Data center standard infrastructure

Frameworks and abstractions

- Shell support and abstractions for in-network processing, disaggregated computation, distributed applications ...
- Systems and applications built on top

Two working flows:





Infrastructure – HACC cluster





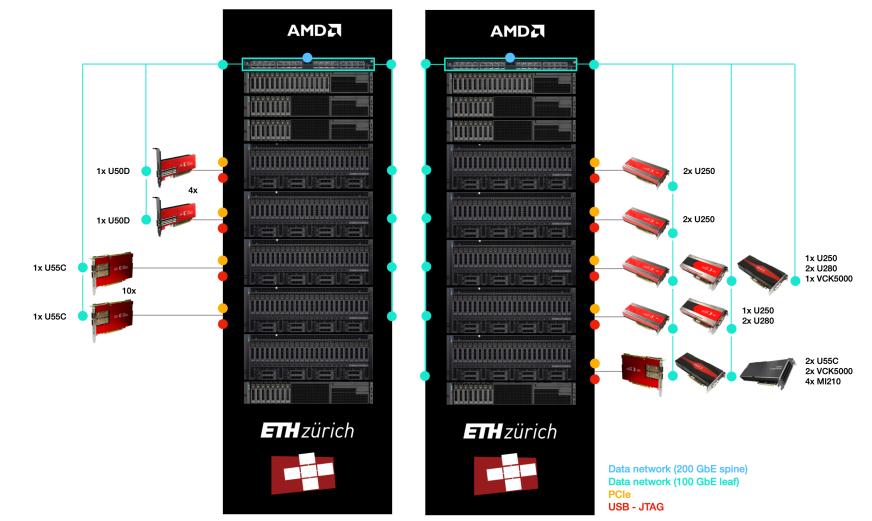
 The Heterogeneous Accelerated Compute Clusters (HACC) program is a unique initiative to support novel research in adaptive compute acceleration for data center settings and highperformance computing (HPC).

• ETH Zurich HACC

https://systems.ethz.ch/research/data-processing-onmodern-hardware/hacc.html



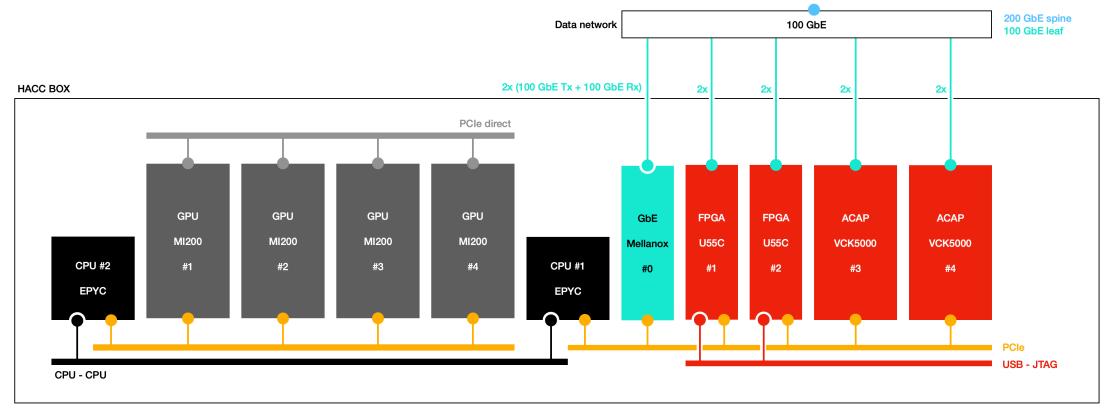
Introduction to HACC cluster



Copyright ETH Zürich - 2023



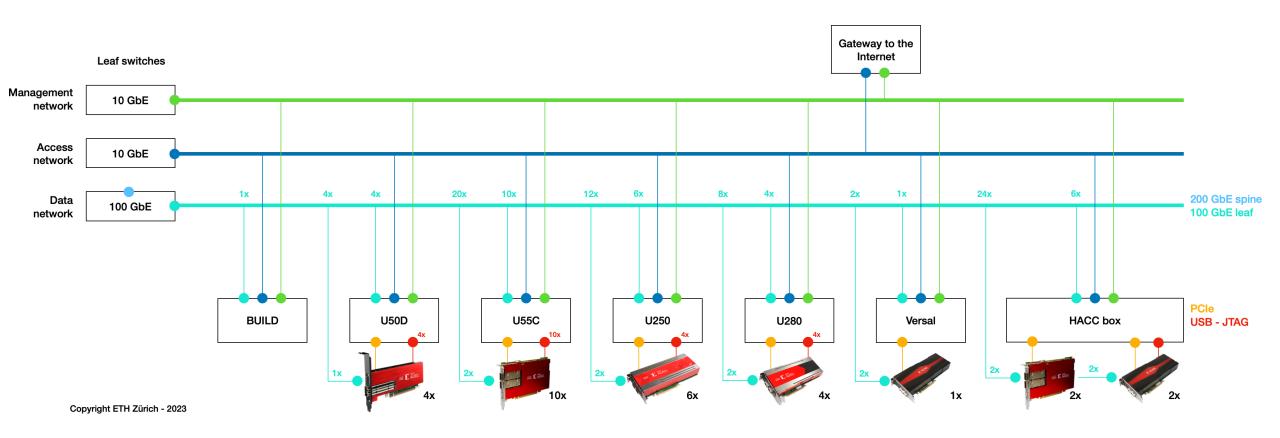
Overview (HACC boxes)



Copyright ETH Zürich - 2023

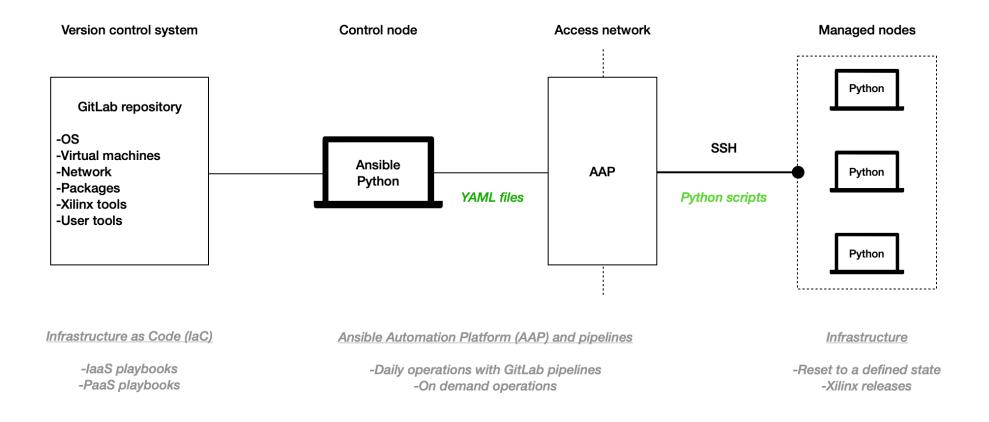


Overview





Operating the cluster





Booking system

- Reserving a specific VM/device for a specific period
 - Maximum 5 hours per reservation
- During a reservation, only the selected user can connect to the VM/device
- User can choose different workflows when login
 - Vitis workflow
 - Coyote workflow

| AMDZ XLLNX Heterogeneous Accelerated Compute Clusters University Research Partners University Research Partners University Research Partners | | | | | | | | | | | | | | | | |
|---|--|----|----|----------------|---------|----|----|------|----|----|---------|---------|----|----|--|--|
| | New Booking all times are CET (Zurich, Switzerland) Maximum booking time is 5 hours Time range * 2023-02-06 13:29 - 2023-02-06 14:25 | | | | | | | | | | | | | | | |
| | | | | Use Feb 202 | | | | | | | lar 202 | | | > | | |
| | Su | Мо | Tu | We | з Th | Fr | Sa | Su I | Mo | Tu | we | э Th | Fr | Sa | | |
| | 29 | | | | | | 4 | | 27 | 28 | | | | 4 | | |
| | | 6 | | 8 | 9 | 10 | | | 6 | | 8 | 9 | 10 | | | |
| | 12 | 13 | 14 | 15 | 16 | | 18 | 12 | | 14 | 15 | 16 | 17 | 18 | | |
| | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | | |
| | 26 | 27 | 28 | | | | 4 | 26 | 27 | 28 | 29 | 30 | 31 | | | |
| | | 6 | | 8 | 9 | 10 | | | | 4 | | 6 | | 8 | | |
| | 13 🗸 : 29 🗸 14 🗸 : 29 🗸 | | | | | | | | | | | | | | | |



User access

- Access requires registration
 - ETH users contact Gustavo Alonso
 - All others through AMD Xilinx (HACC program)
 - Users get guest account at ETH (renewable)



Build server (no FPGA)

Dell Power Edge R740

2 x Intel Xeon Gold 6248 2,5 GHz, 20C/40T

12 x 32 GB DDR4

6 x 960 GB SSD

Mellanox Connect X-5, single port (100Gb)

Intel 10 Gbs card

Large server for compilation, project development, and support of cluster activities

Large enough to support many concurrent users



Nodes with Multiple FPGAs (Hypervisor)

Nodes with 2 FPGAs

Dell Power Edge R740

2 x Alveo U250

2 x Intel Xeon Gold 6234 3,3 GHz, 8C/16T

12 x 32 GB DDR4

2 x 96GB SSD

2 x Mellanox Connect X-5, single port (100Gb)

Intel 10 Gbs card

Nodes with 3 FPGAs Dell Power Edge R940 **1 x Alveo U250 + 2 x Alveo U280** 2 x 2 x 2 x Intel Xeon Gold 6234 3,3 GHz, 8C/16T 24 x 16 GB DDR4 2 x 96GB SSD **2 x Mellanox Connect X-5, single port (100Gb)** Intel 10 Gbs card



Nodes with single FPGA (Bare-metal)

AMD EPYC

32 CPU cores

64 GB DDR4

Mellanox Connect X-5, single port (100Gb)

Intel 10 Gbs card

1 x Alveo U55C/U50D



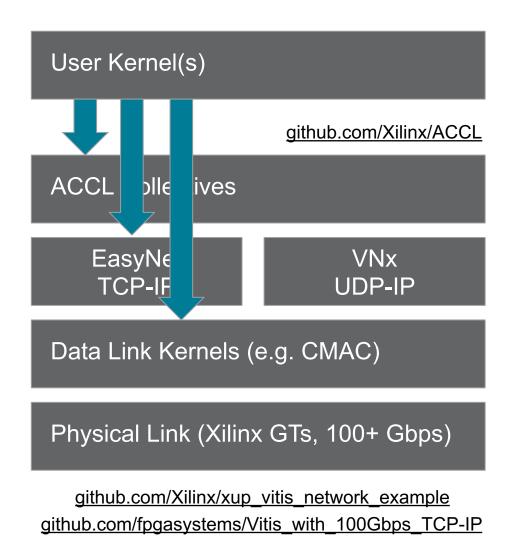
Fast transitioning between Vitis and Coyote workflows

- Hot-plug boot
 - Use Linux capabilities to re-enumerate PCI devices on the fly
 - without the need for cold or warm rebooting of the system
- Cold boot
 - Powering off and on the machine
 - Resets all the hardware peripherals (including all PCI devices)
 - Xilinx accelerator cards: causes the base shell be flashed from PROM
 - This operation is required to revert a server to the Vitis workflow
- Warm boot
 - Restarts the system without interrupting the power
 - Xilinx accelerator cards: re-enumerates the number of PCI functions
 - This operation is required to bring a server to the bare-metal workflow

24

Vitis flow

- Data Link Kernels
 - Ethernet or Aurora
 - Wraps MRMAC (Versal) or CMAC (UltraScale) and provides AXI Stream interface to upper layers
 - Limited reliability support e.g. FEC
- Network (IP) and Transport Kernels
 - UDP with VNx low footprint, unreliable
 - TCP with EasyNet higher footprint, reliable
- Collective Offload Kernel(s)
 - ACCL
- All kernels are Vitis-compatible, portable across Alveo range
- Full freedom to construct application on top of any layer





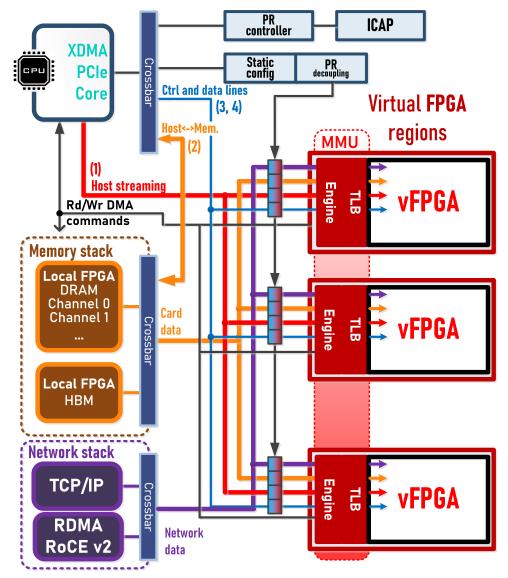




Coyote flow

• Microkernel for FPGAs

- Multiple isolated untrusted vFPGAs
- Spatial and temporal sharing (PR)
- Dynamic reconfiguration (scheduler)
- Virtualized memory
- Unified memory (HMM)
- HBM and DRAM striping service
- Shared TCP/IP service
- Shared RDMA service
- Unified logic interface -> portability
- RTL and HLS support
- Runs on u50, u200, u250, u280, u55c, vcu118, Enzian





Coyote flow

- User space abstractions
 - cSched- Coyote scheduler, reconfiguration controller
 - **cProcess** Coyote process, multiple can run within a single vFPGA
 - cThread Coyote thread, multiple can run within a single cProc. Task level parallelism
 - cTask Coyote task, arbitrary user variadic function executed by cThreads
 - **cService** Coyote library daemon, background service, UDS for IPC

