# WORKLOAD ACCELERATION IN DATACENTERS AND (EDGE) CLOUD

## Industriekolloquium TU Darmstadt 2020

- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up

## Company: Xelera Technologies GmbH

- Founded 2017 under EXIST program at TU Darmstadt
- GmbH founded 04/2018
- Exit from Exist 06/2019

- Current staff: 10 people, mostly technical

- Location: Rheinstraße 40-42, Darmstadt

## Speaker: Julian Käuser

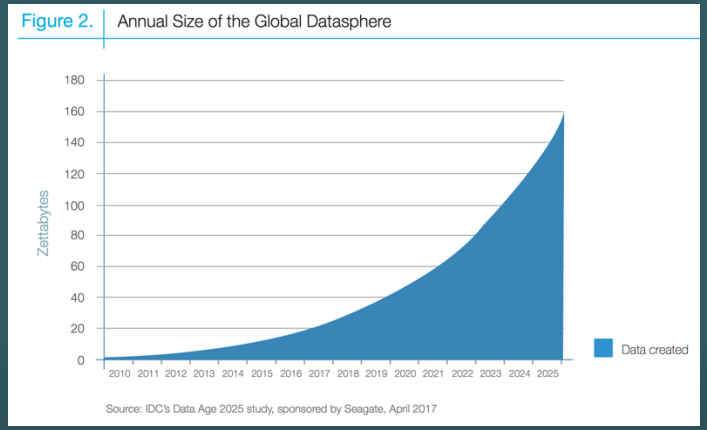
- SW - and FPGA developer
- Emphasis: porting algorithms to FPGA, framework integration

- Started as working student 09/2018
- Full-time developer since 12/2019

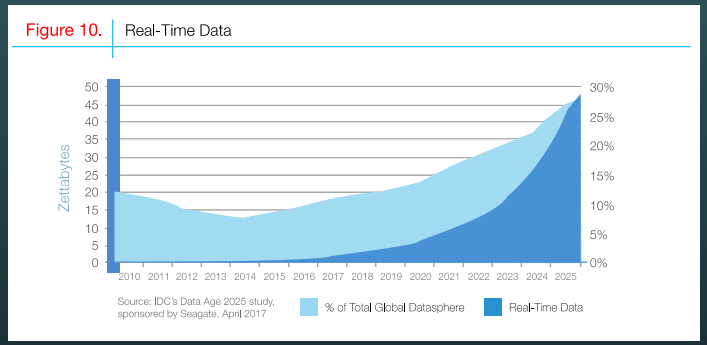
- Studies: ETiT TU Darmstadt
- Curriculum scheme: Computer Engineering ("Datentechnik")
- Emphasis: logic design, computer science classes

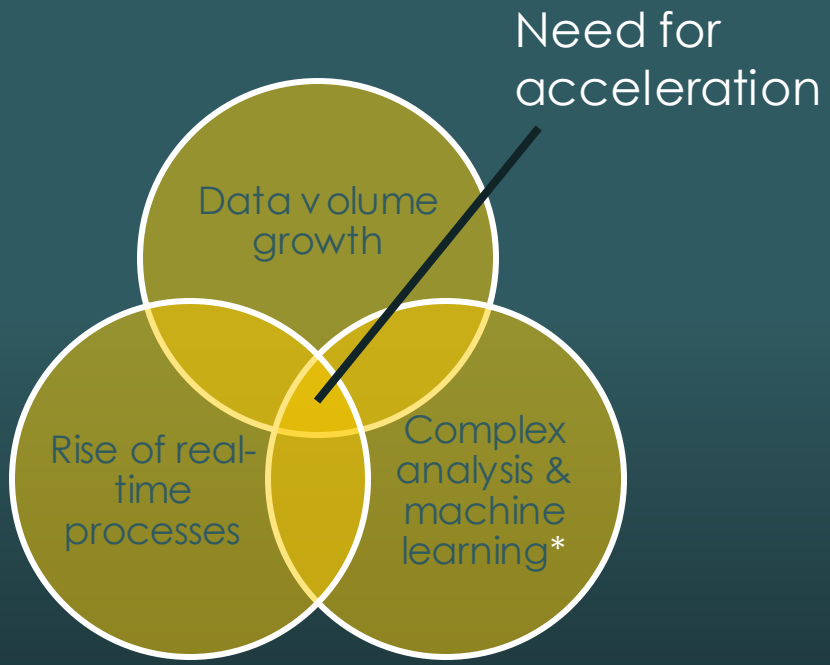
- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up

*




*




* https://www.import.io/wp-content/uploads/2017/04/Seagate-WP-DataAge2025-March-2017.pdf

Customer software application

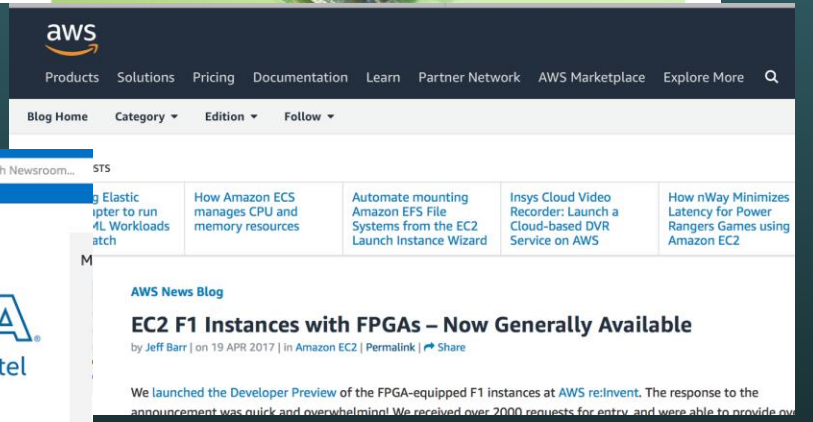
XELERA SUITE

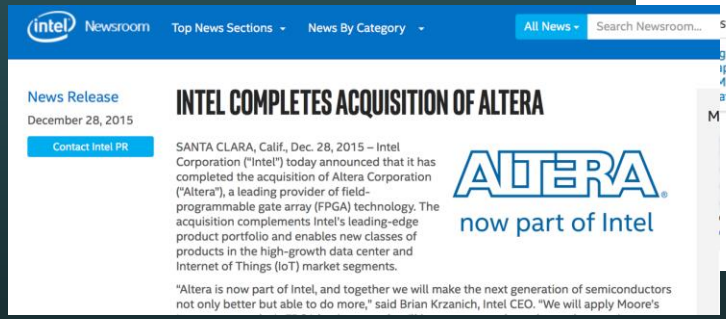
PLUG IN

PLUG IN

Conventional CPUs

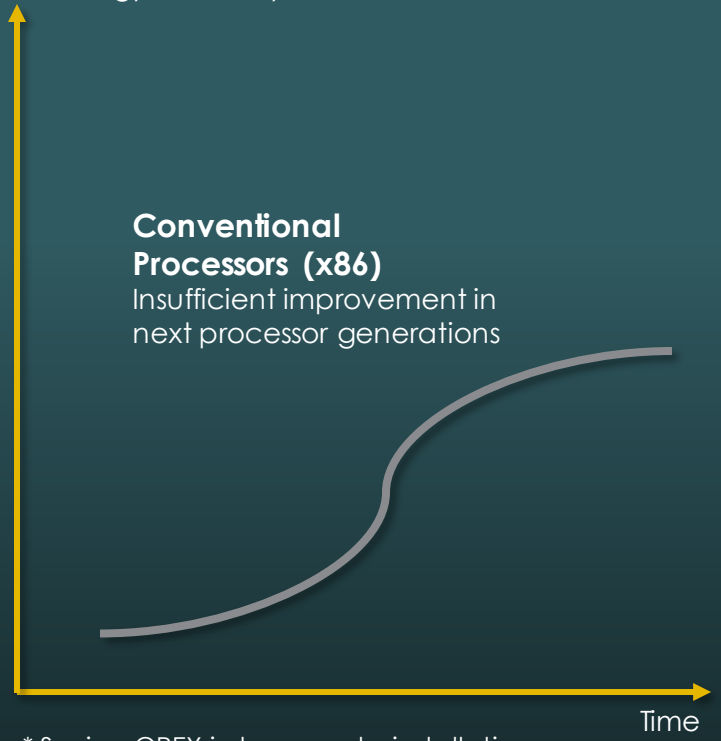
Accelerator platform

## Performance

- Speed
- Energy Efficiency



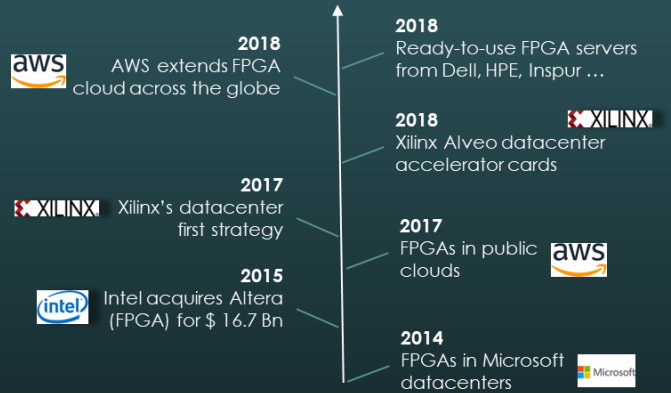

driven by
large OEMs

## Hardware accelerators (FPGAs/GPUs)

- Complementing CPUs in data centers and the cloud
- Acceleration beyond conventional CPUs (10x – 100x)
- FPGAs: better energy efficiency*

GPU

FPGA




* Saving OPEX in large-scale installations

- Pricing

| Name | FPGAs | vCPUs | Instance-Arbeitsspeicher (GIB) | SSD-Speicherung (GB) | Enhanced Networking | EBS-optimiert | On-Demand-Preis/Std.* |
|---|---|---|---|---|---|---|---|
| f1.2xlarge | 1 | 8 | 122 | 470 | Ja | Ja | 1,65 USD |
| f1.4xlarge | 2 | 16 | 244 | 940 | Ja | Ja | 3,30 USD |
| f1.16xlarge | 8 | 64 | 976 | 4 x 940 | Ja | Ja | 13,20 USD |

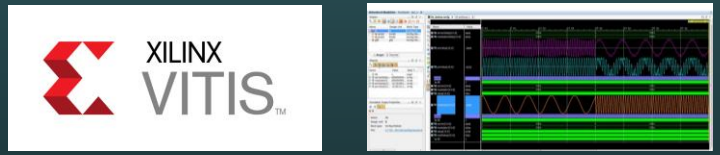
https://aws.amazon.com/de/ec2/instance-types/f1/

- Device

| Device Name | VU3P | VU5P | VU7P | VU9P |
|---|---|---|---|---|
| System Logic Cells (K) | 862 | 1,314 | 1,724 | 2,586 |
| CLB Flip-Flops (K) | 788 | 1,201 | 1,576 | 2,364 |
| CLB LUTs (K) | 394 | 601 | 788 | 1,182 |
| Max. Dist. RAM (Mb) | 12.0 | 18.3 | 24.1 | 36.1 |
| Total Block RAM (Mb) | 25.3 | 36.0 | 50.6 | 75.9 |
| UltraRAM (Mb) | 90.0 | 132.2 | 180.0 | 270.0 |
| DSP Slices | 2,280 | 3,474 | 4,560 | 6,840 |
| Peak INT8 DSP (TOP/s) | 7.1 | 10.8 | 14.2 | 21.3 |
| PCIe® Gen3 x16 | 2 | 4 | 4 | 6 |

https://www.xilinx.com/support/documentation/selection-guides/ultrascale-plus-fpga-product-selection-guide.pdf

- Development Kit




xilinx.com


| | Feature | Alveo U200 | Alveo U250 | Alveo U280 | Alveo U50 |
|---|---|---|---|---|---|
| Dimensions | Width | Dual Slot | Dual Slot | Dual Slot | Single Slot |
| | Form Factor, Passive<br>Form Factor, Active | Full Height, ¾ Length<br>Full Height, Full Length | Full Height, ¾ Length<br>Full Height, Full Length | Full Height, ¾ Length<br>Full Height, Full Length | Half Height, ½ Length |
| Logic Resources [1] | Look-Up Tables | 1,182K | 1,728K | 1,304K | 872K |
| | Registers | 2,364K | 3,456K | 2,607K | 1,743K |
| | DSP Slices | 6,840 | 12,288 | 9,024 | 5,952 |
| DRAM Memory | DDR Format | 4x 16GB 72b DIMM DDR4 | 4x 16GB 72b DIMM DDR4 | 2x 16GB 72b DIMM DDR4 | – |
| | DDR Total Capacity | 64GB | 64GB | 32GB | – |
| | DDR Max Data Rate | 2400MT/s | 2400MT/s | 2400MT/s | – |
| | DDR Total Bandwidth | 77GB/s | 77GB/s | 38GB/s | – |
| | HBM2 Total Capacity | – | – | 8GB | 8GB |
| | HBM2 Total Bandwidth | – | – | 460GB/s | 316GB/s [4] |
| Internal SRAM | Total Capacity | 43MB | 57MB | 43MB | 28MB |
| | Total Bandwidth | 37TB/s | 47TB/s | 35TB/s | 24TB/s |
| Interfaces | PCI Express® | Gen3 x16 | Gen3 x16 | Gen3 x16, 2xGen4 x8, CCIX | Gen3 x16, 2xGen4 x8, CCIX |
| | Network Interface | 2x QSFP28 | 2x QSFP28 | 2x QSFP28 | U50 [2] - 1x QSFP28<br>U50DD [3] - 2x SFP-DD |
| Power and Thermal | Thermal Cooling | Passive, Active | Passive, Active | Passive, Active | Passive |
| | Typical Power | 100W | 110W | 100W | 50W |
| | Maximum Power | 225W | 225W | 225W | 75W |
| Time Stamp | Clock Precision | – | – | – | IEEE Std 1588 |
| Tool Support | Vitis™ Developer Environment | Yes | Yes | Yes | Yes |

Alveo™ Data Center Accelerator Cards

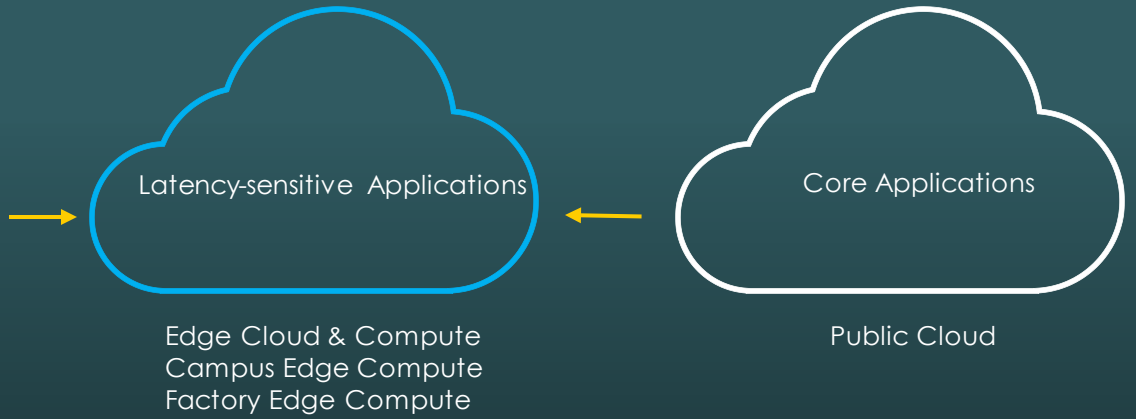
NVIDIA Tesla V100:
250 - 300W

https://www.xilinx.com/support/documentation/selection-guides/alveo-product-selection-guide.pdf 21.06.2020

- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up

# EDGE COMPUTING TECHNOLOGY VIEW

| | |
|---|---|
| Next-gen Digital Assistants | 60-100ms |
| Factory Automation | 0.25-10ms |
| AR / VR | 15-20ms |
| Robotics and Telepresence | 1-20ms |
| Healthcare | 1-10ms |
| Intelligent Transportation Systems | 10-100ms |
| Remote Gaming | 30-50ms |
| Smart Grid | 20-100ms |

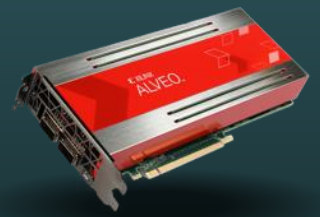



* Parvez et al., "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions", arXiv, 2018
** MGP - Mobil Gesteuerte Produktion/5G fuer Digitale Fabriken, 2018, https://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/mobil-gesteuerte-produktion.pdf?__blob=publicationFile&v=7, 2018

| | |
|---|---|
| Next-gen Digital Assistants | 60-100ms |
| Factory Automation | 0.25-50ms |
| AR / VR | 15-20ms |
| Robotics and Telepresence | 1-20ms |
| Healthcare | 1-10ms |
| Intelligent Transportation Systems | 10-100ms |
| Remote Gaming | 30-50ms |
| Smart Grid | 20-100ms |

Requirements

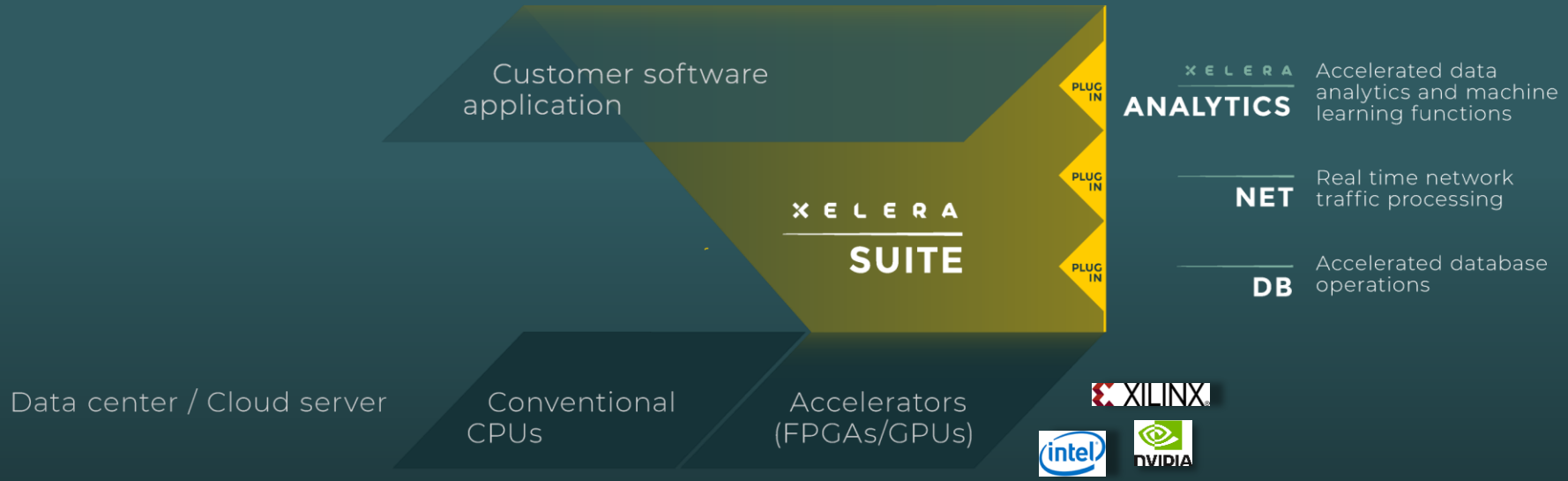
- Low network and compute latency
- Guaranteed network and compute latency
- High processing bandwidth
- Concurrent requests
- Small footprint (space & power)





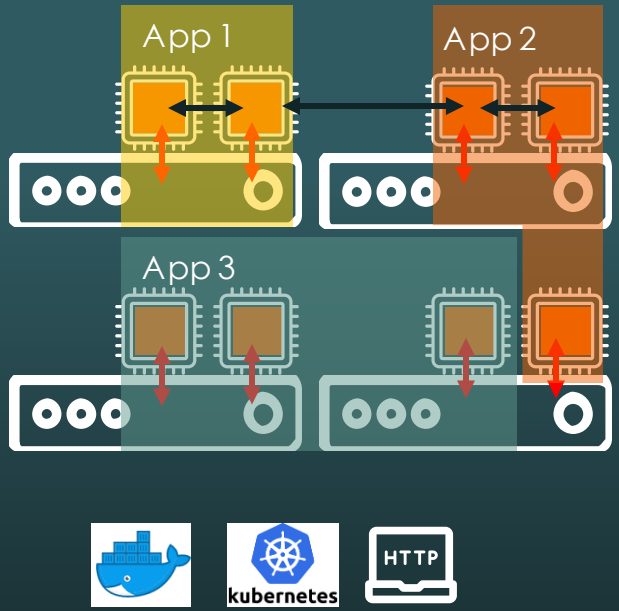
* Parvez et al., "A Survey on Low Latency Towards 5G: RAN, Core Network and Caching Solutions", arXiv, 2018
** MGP - Mobil Gesteuerte Produktion/5G fuer Digitale Fabriken, 2018, https://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/mobil-gesteuerte-produktion.pdf?__blob=publicationFile&v=7, 2018

- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up


Customer software application
XELERA SUITE
PLUG IN
PLUG IN
PLUG IN
XELERA ANALYTICS
Accelerated data analytics and machine learning functions
NET
Real time network traffic processing
DB
Accelerated database operations
Data center / Cloud server
Conventional CPUs
Accelerators (FPGAs/GPUs)
XILINX
intel
NVIDIA

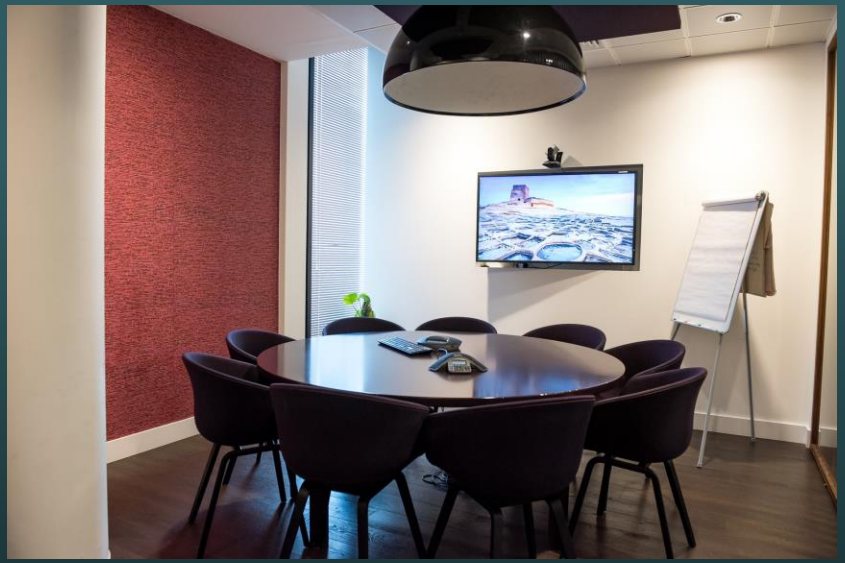
How to …

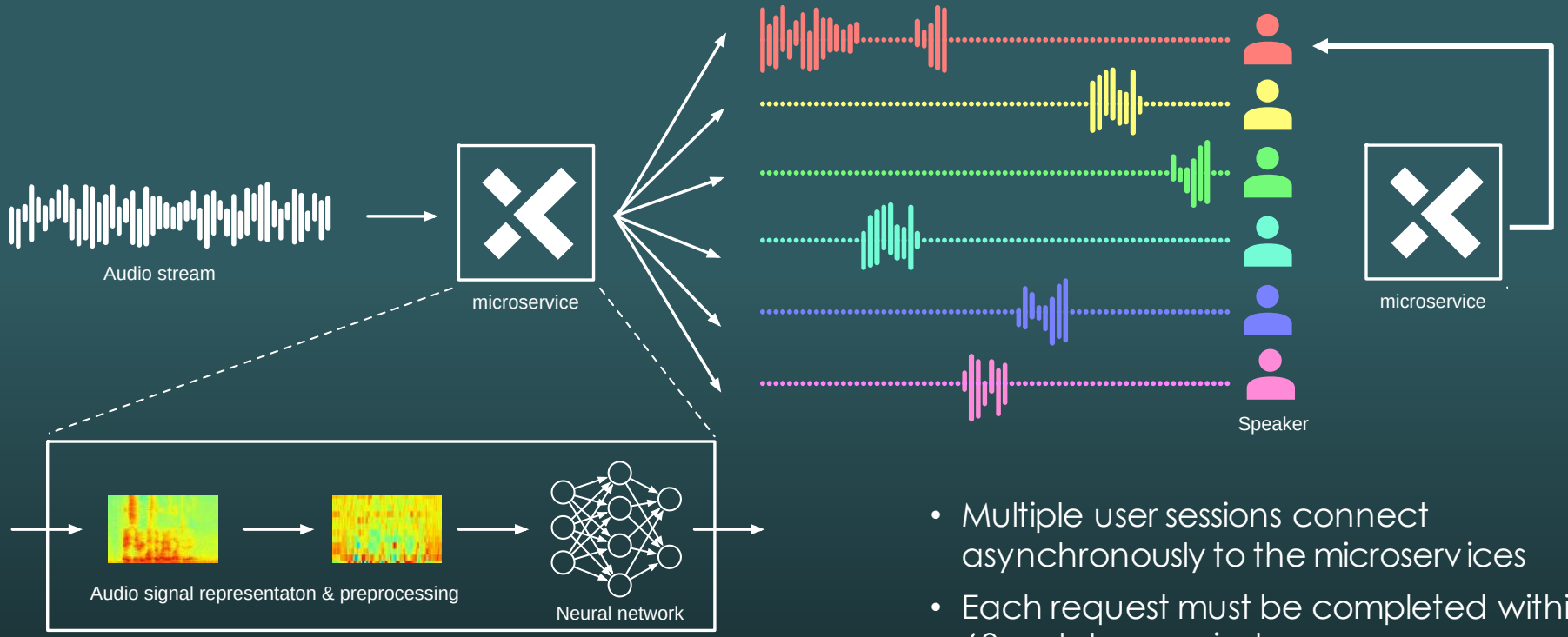
- build the accelerator kernels
- integrate with software
- distribute a workload across accelerators
- scale across a cluster
- communicate between (different) accelerators
- guarantee latency as concurrency increases
- deploy through containers & Kubernetes
- deploy as a network service



- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up

# Use Case: Low-latency, high-throughput Speaker Diarization

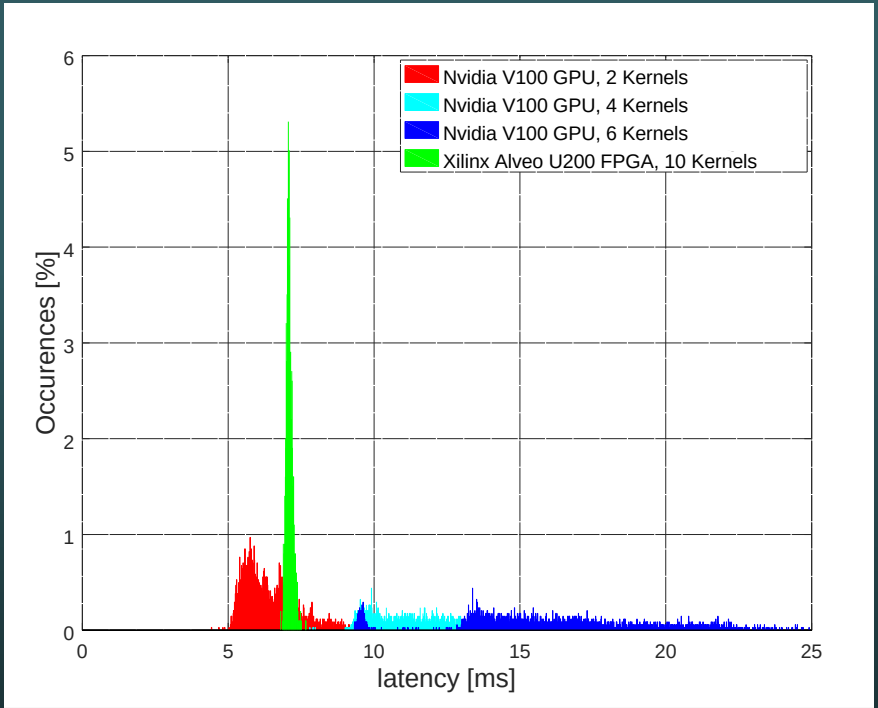





- Multiple user sessions connect asynchronously to the microservices
- Each request must be completed within a 60 ms latency window

# Maximize number of concurrent user sessions per accelerator card

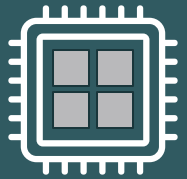
Determine
tail latency




(*) Benchmark obtained with Xilinx Alveo U200 FPGA on Dell R740 server vs. NVIDIA Tesla V100 SXM2 on AWS EC2 p3.2xlarge instance.

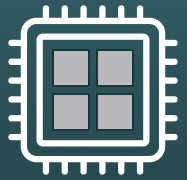
## Maximize number of concurrent user sessions per accelerator card

Alveo U250 FPGA, no batching



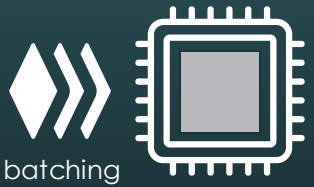
84 concurrent sessions

Tesla V100 GPU, no batching



12 concurrent sessions

Tesla V100 GPU, batching



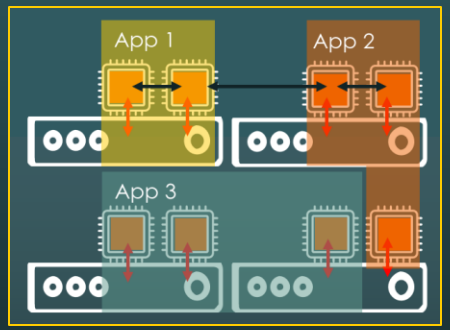

100 concurrent sessions

(*) Benchmark obtained with Xilinx Alveo U250 FPGA on Dell R740 servers vs. NVIDIA Tesla V100 SXM2 on AWS EC2 p3.2xlarge instance.

## Work Areas

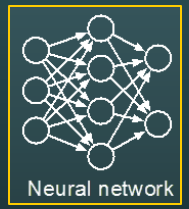
Software:

- Load balancing
- Frontend
- Integration of FPGA kernels




Device:

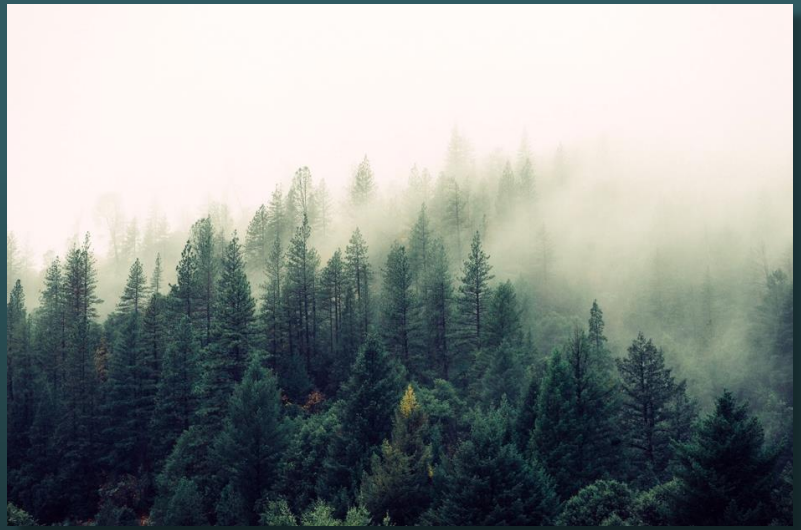
- Neural Network on FPGA
- Quantization
- handling large number of weights



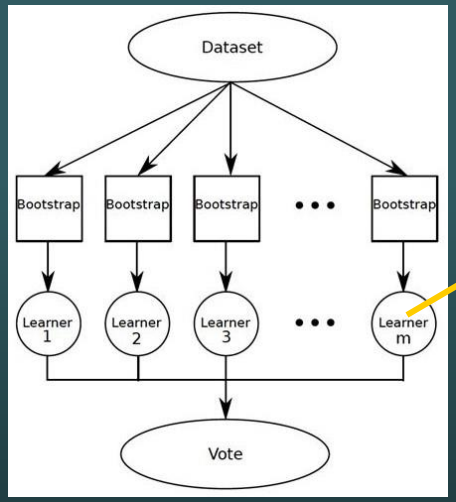

Algorithmic:

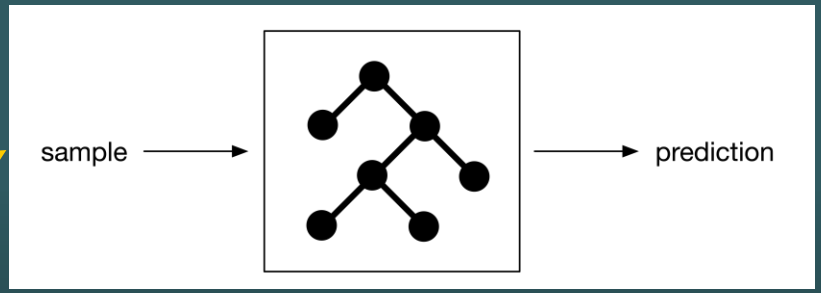
- Tweak neural network
- Classic audio preprocessing
- Minimize algorithmic delay

## Use Case: High-speed random forest



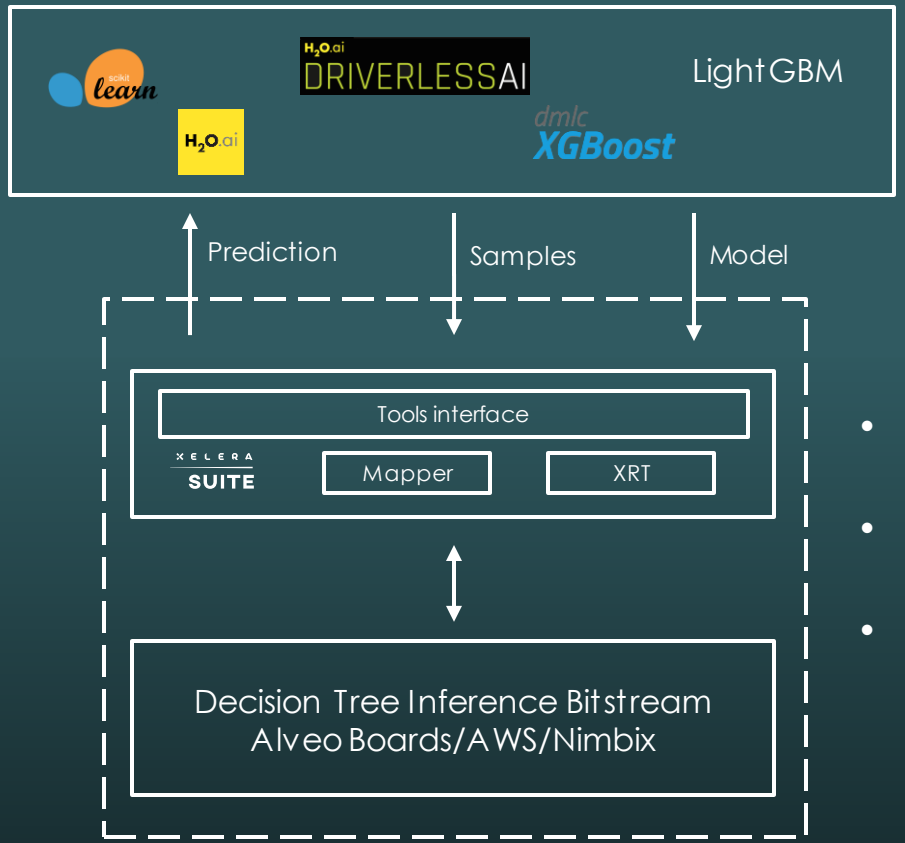
Principle: Randomize Decision Trees, apply reduction




avg, majority, sum, log. regression etc.

- Iterate many decisions trees → memory bound




- Python Integration
- Bring your Own Model
- Inference Server (real-time deployments)

# RF / END-TO-END LATENCY [S]

Xilinx Alveo U250

| Number of sample trees / batch size | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | 0.000361 | 0.000431 | 0.000874 | 0.00446539 |
| 10 | 0.000356 | 0.00052 | 0.000906 | 0.00509215 |
| 100 | 0.000723 | 0.00086 | 0.001895 | 0.01166138 |
| 1000 | 0.001236 | 0.002672 | 0.008208 | 0.05321404 |
| 10000 | 0.006452 | 0.020096 | 0.062077 | 0.4049033 |
| 100000 | 0.060914 | 0.17201 | 0.511658 | 3.91938754 |

Nvidia Tesla T4

| Number of sample trees / batch size | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | 0.01507 | 0.15316 | 2.02396 | 17.36702 |
| 10 | 0.01560 | 0.15130 | 2.07955 | 17.48893 |
| 100 | 0.01632 | 0.15840 | 2.06146 | 17.40507 |
| 1000 | 0.01621 | 0.15625 | 2.08708 | 17.48609 |
| 10000 | 0.01561 | 0.15516 | 2.12261 | 17.27794 |
| 100000 | 0.02500 | 0.17333 | 2.19849 | 18.52639 |

Intel Xeon 5118 (32 cores)

| Number of sample trees / batch size | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | 0.113754 | 0.558969 | 5.682151 | 60.4001916 |
| 10 | 0.112719 | 0.615108 | 5.674495 | 65.8061304 |
| 100 | 0.113577 | 0.717954 | 6.465377 | 65.3164281 |
| 1000 | 0.110944 | 0.818687 | 7.682359 | 77.2385877 |
| 10000 | 0.215964 | 1.42355 | 13.90265 | 136.317463 |
| 100000 | 1.354374 | 11.20106 | 112.6743 | 1129.38227 |

Xilinx Alveo U250 vs Nvidia Tesla T4

| Number of sample trees / batch size | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | x41.73 | x354.99 | x2317.01 | x3889.25 |
| 10 | x43.84 | x290.75 | x2296.40 | x3434.49 |
| 100 | x22.56 | x184.22 | x1087.98 | x1492.54 |
| 1000 | x13.12 | x58.47 | x254.28 | x328.60 |
| 10000 | x2.42 | x7.72 | x34.19 | x42.67 |
| 100000 | x0.41 | x1.01 | x4.30 | x4.73 |

Xilinx Alveo U250 vs Intel Xeon 5118 (32 cores)

| Number of sample trees / batch size | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | x217 | x971 | x4859 | x16175 |
| 10 | x183 | x1055 | x4108 | x15475 |
| 100 | x116 | x758 | x2720 | x5959 |
| 1000 | x24 | x176 | x671 | x1393 |
| 10000 | x5 | x35 | x154 | x321 |
| 100000 | x4 | x34 | x140 | x271 |

## Work Areas

Software:

- Distribute trees and samples across kernels
- Frontend for frameworks
- As fast as possible preprocessing


Device:

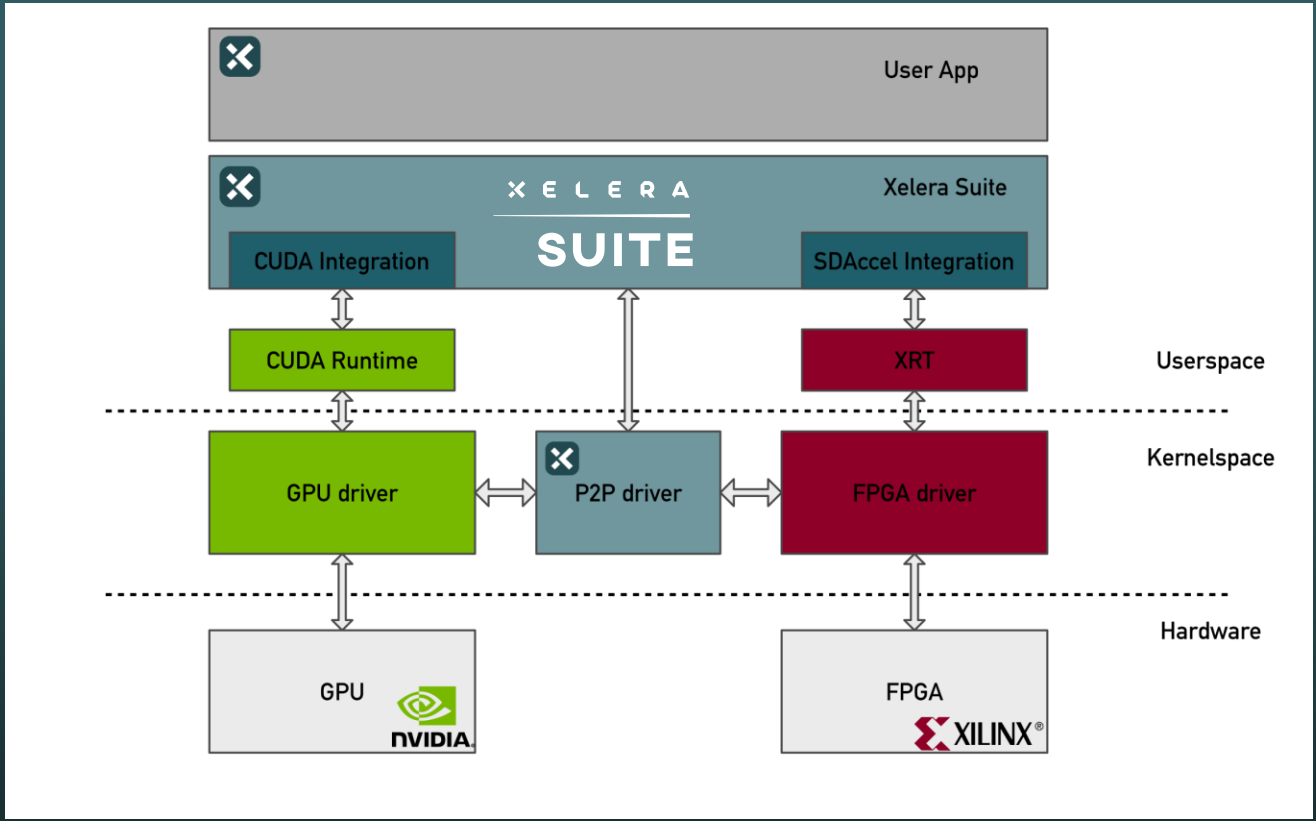
- Utilize on-chip memory bandwidth
- Optimal data formats
- Architecture that fits many frontend frameworks

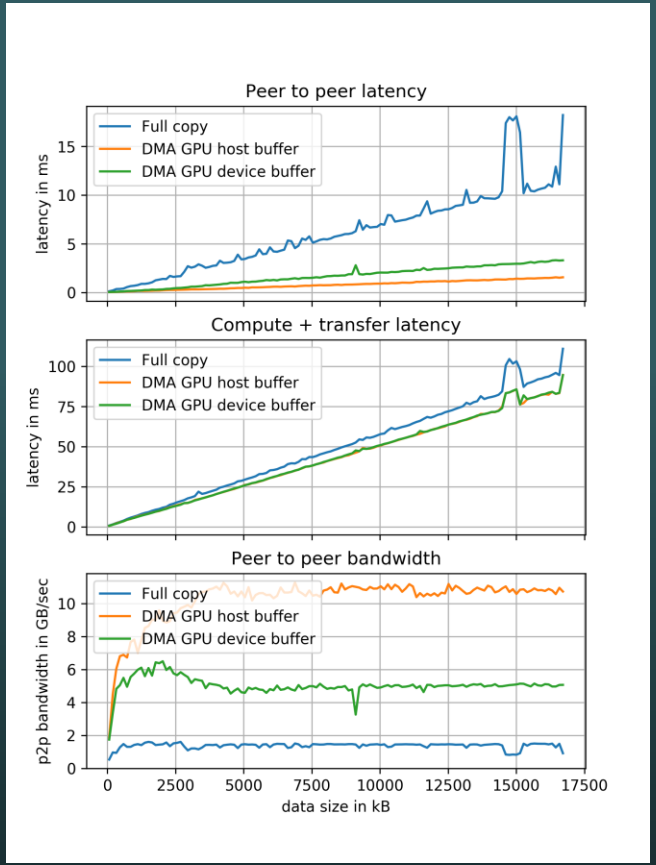
| | | |
|---|---|---|
| DRAM Memory | DDR Format | 4x 16GB 72b DIMM DDR4 |
| | DDR Total Capacity | 64GB |
| | DDR Max Data Rate | 2400MT/s |
| | DDR Total Bandwidth | 77GB/s |
| | HBM2 Total Capacity | – |
| | HBM2 Total Bandwidth | – |
| Internal SRAM | Total Capacity | 43MB |
| | Total Bandwidth | 37TB/s |

Algorithmic:

- Identify typical problem dimensions

| sample batch size \ Number of trees | 100 | 1000 | 10000 | 100000 |
|---|---|---|---|---|
| 1 | x217 | x971 | x4859 | x16175 |
| 10 | x183 | x1055 | x4108 | x15475 |
| 100 | x116 | x758 | x2720 | x5959 |
| 1000 | x24 | x176 | x671 | x1393 |
| 10000 | x5 | x35 | x154 | x321 |
| 100000 | x4 | x34 | x140 | x271 |

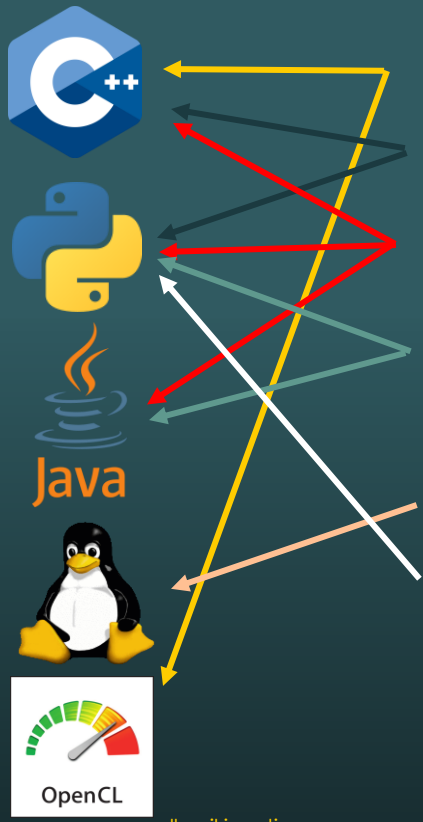

User App
XELERA
SUITE
Xelera Suite
CUDA Integration
SDAccel Integration
CUDA Runtime
XRT
Userspace
GPU driver
P2P driver
FPGA driver
Kernelspace
Hardware
GPU
NVIDIA
FPGA
XILINX


Peer to peer latency
Full copy
DMA GPU host buffer
DMA GPU device buffer
latency in ms
0
5
10
15
Compute + transfer latency
Full copy
DMA GPU host buffer
DMA GPU device buffer
latency in ms
0
25
50
75
100
Peer to peer bandwidth
Full copy
DMA GPU host buffer
DMA GPU device buffer
p2p bandwidth in GB/sec
0
2
4
6
8
10
0
2500
5000
7500
10000
12500
15000
17500
data size in kB

- Introduction
- FPGAs in Data Centers
- About the "Edge Cloud"
- XeleraSuite
- Application Examples
- Workflow and used Technologies
- Wrap-up

## Languages

- C++14
- Python
- Java
- REST
- C (Linux Driver)
- Unix
- OpenCL




all: wikipedia.org

## Skills

- Writing fast code
- Prepare data for device
- Cross-Language interface
- 3rd Party Framework examination
- Docker, Virtualization
- Data Science

## Languages

- C++/C (Vivado HLS)
- VHDL (seldomly...)
- CUDA (GPU)

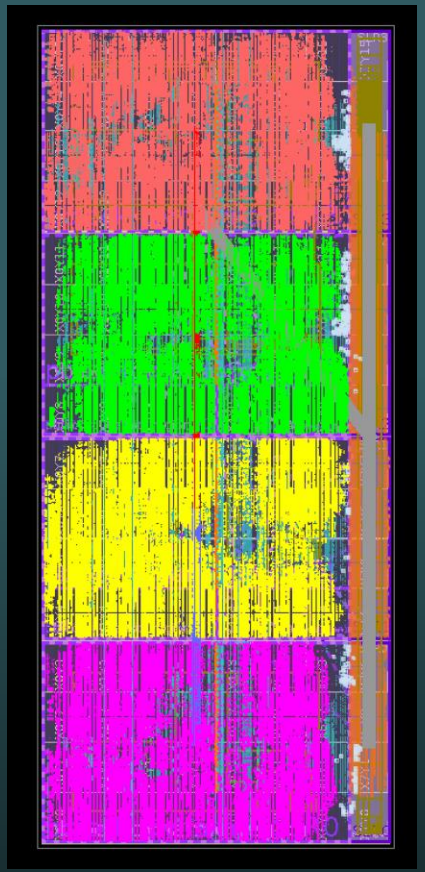
XILINX VITIS™

Vivado™ HLS

xilinx.com

## FPGA Skills

- Implement algorithms w.r.t. FPGA characteristics
- Difference latency/throughput optimization
- Usually, make full use of all HW resources



## Work Distribution: SW/HW ~ 50:50 (time + staff)

HLS for rapid FPGA development

- Standard data types
- Automatic interface generation
- Control result with pragmas
- Debugging in software (emulation – it's C/C++)

```
#pragma HLS INTERFACE m_axi offset = slave latency = 64 \
    num_write_outstanding = 16 num_read_outstanding = 16 \
    max_write_burst_length = 64 max_read_burst_length = 64 \
    bundle = gmem0_1 port = outputData


#pragma HLS INTERFACE s_axilite port = config bundle = control
#pragma HLS INTERFACE s_axilite port = status bundle = control
#pragma HLS INTERFACE s_axilite port = inputData bundle = control
#pragma HLS INTERFACE s_axilite port = outputData bundle = control
#pragma HLS INTERFACE s_axilite port = return bundle = control


// memory for key
ap_uint<_word_size> keyMem[_channelNumber][_key_mem_size];
#pragma HLS array_partition variable=keyMem dim=1

// memory for salt
ap_uint<_word_size> saltMem[_channelNumber][_salt_mem_size];
#pragma HLS array_partition variable=saltMem dim=1

#pragma HLS inline recursive
```

## at Xelera Technologies, we...

- ... accelerate software stacks on servers
- ... using off-the-shelf hardware
- ... enable acceleration technology for everyone

**If interested, apply for...**

- Internship
- Student Assistant
- Bachelor's / Master's Thesis
- Full Time


joinus@xelera.io

Personal
- julian.kaeuser@xelera.io

General:
- info@xelera.io
- www.xelera.io
- linkedin.com/company/xelera-technologies/