AMD
XILINX

FPGA/ACAP関連技術のHPC/AIに向けた取り組みと展望

ザイリンクス株式会社
Adaptive and Embedded Computing Group (AECG)
Data Center and Communications Group (DCCG)
堀江義弘
2022年 5月

アジェンダ

- 市場ニーズ
- HPCをターゲットとする製品
- 最近の取り組み
- AIをターゲットとする製品
- ユースケース事例の紹介
- まとめ

コンピューティングにおける課題

データの爆発的増加

- ビデオおよび画像コンテンツ
- 90% 非構造化
- より高いスループットとリアルタイムの演算能力が求められる

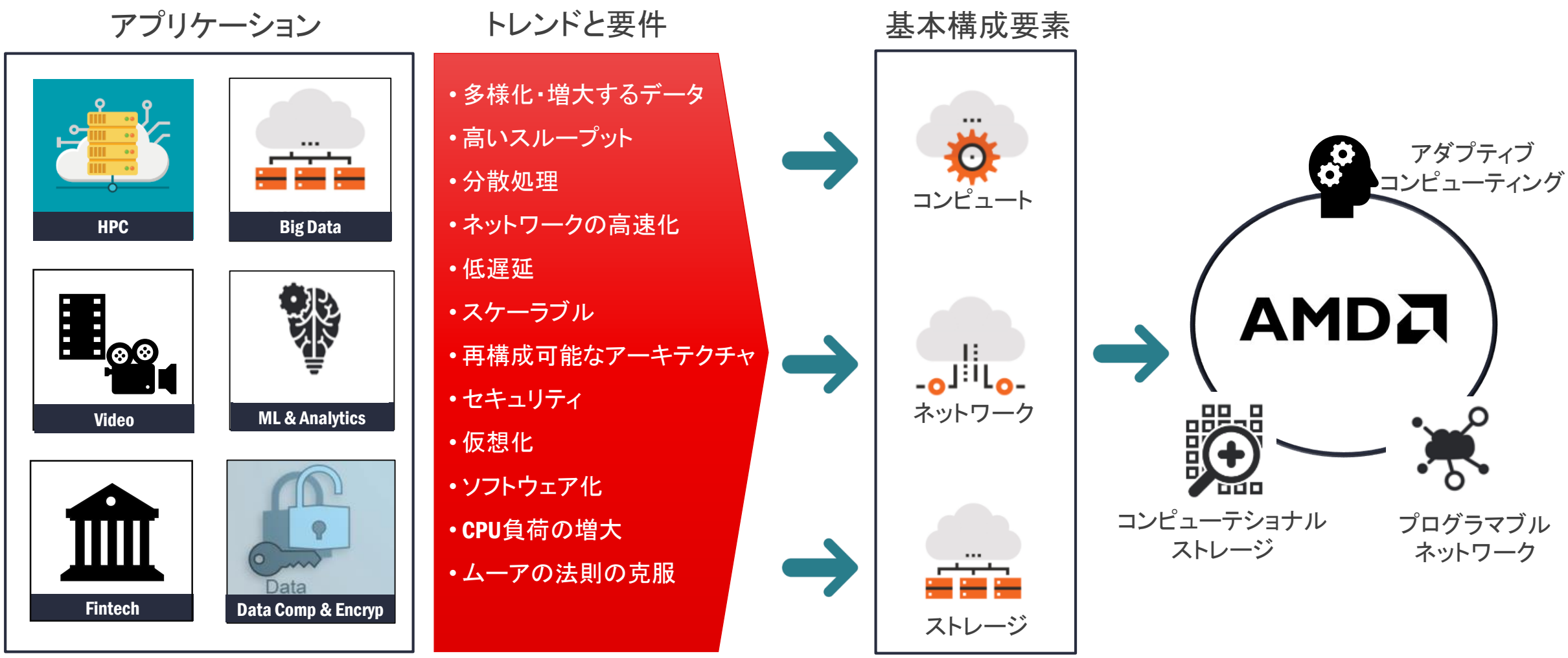
AI 時代の到来

- アプリケーションに最新のインテリジェンスが追加される
- エンドポイントからエッジ、クラウドに至るまで、あらゆる業界に浸透
- すべての場所の AI 処理を高速化する必要がある

「ムーアの法則」後のコンピューティング

- 設計サイクルがイノベーションのスピードに追いつかない
- 多くのアプリケーションが異なるアーキテクチャを必要としている
- アクセラレータを使用したヘテロジニアス コンピューティングの必要性が高まる

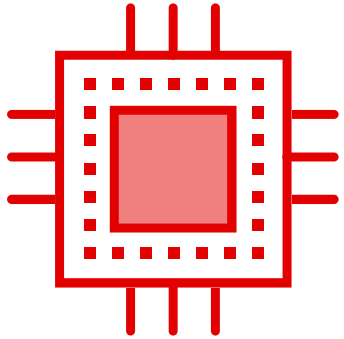
データセンターのトレンドと要件



AMD Provides Platforms Enabling Scalable Architecture

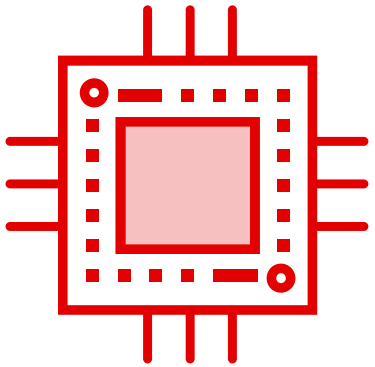
AMD
XILINX

適応型プラットフォームの
ドメイン特化アーキテクチャ (DSA) の必要性



CPU

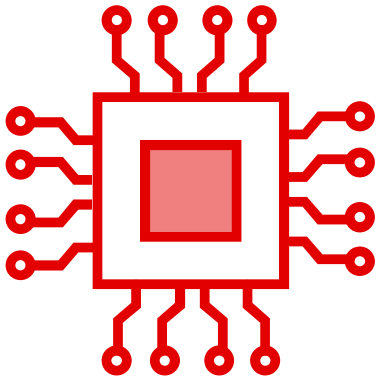
- SW プログラマブル、広く利用されている
- 大半のワークロードには非効率的



固定機能アクセラレータ

ASIC/ASSP/GPU

- 柔軟性に欠ける
- NRE が高く、シリコン サイクルが長い (ASIC)
- 消費電力に課題 (GPU)



適応性に優れた ハードウェア ソリューション

FPGA / ACAP (Adaptive SoC)

- 用途に合わせてハードウェアを最適化
- 進化する要件に迅速に対応

Xilinx Acquisition Creates Industry's High-Performance and Adaptive Computing Leader

Industry-Leading Products	Diversified and Growing Markets	Data Center Momentum	Non-GAAP Margin Expansion	Non-GAAP EPS and Free Cash Flow Accretive in 1st year

ALVEO™

優れたスループット性能

高いスループットと低レイテンシー
コンピュート、ネットワーク、ストレージ

適応性

用途に最適化したアーキテクチャ
ニーズの変化に柔軟・迅速に対応

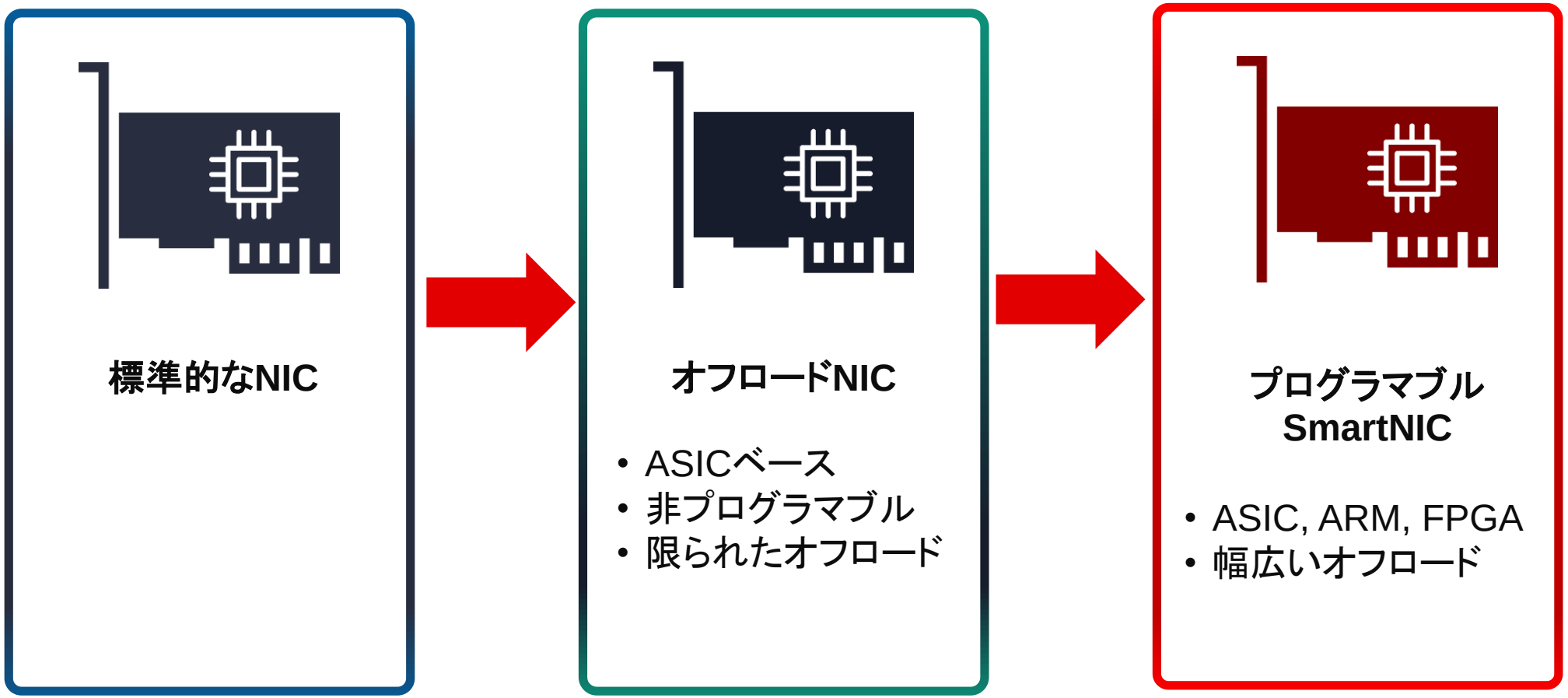
導入が容易

クラウドやオンプレミスで運用
豊富なアプリケーションライブラリ

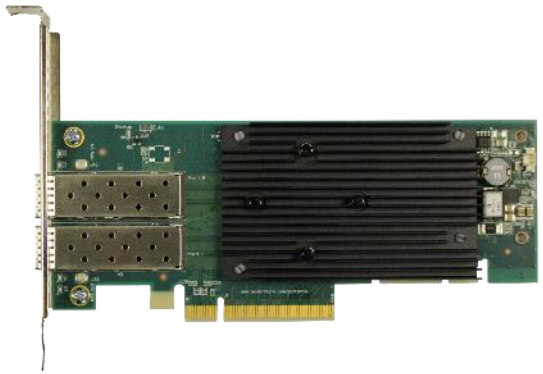
* Versal AI 開発カード

AMD
XILINX

Smart NIC の進化

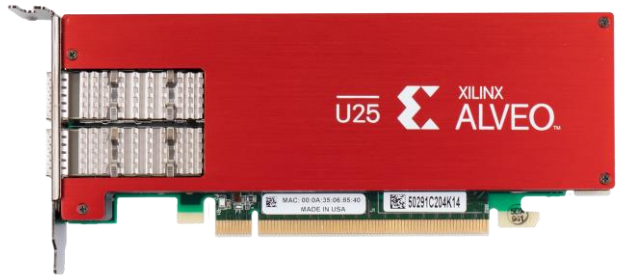


AMD NIC ファミリー



X2

10/25/100Gb オフロードNIC
ASICベース
HHHL PCIe



U25N

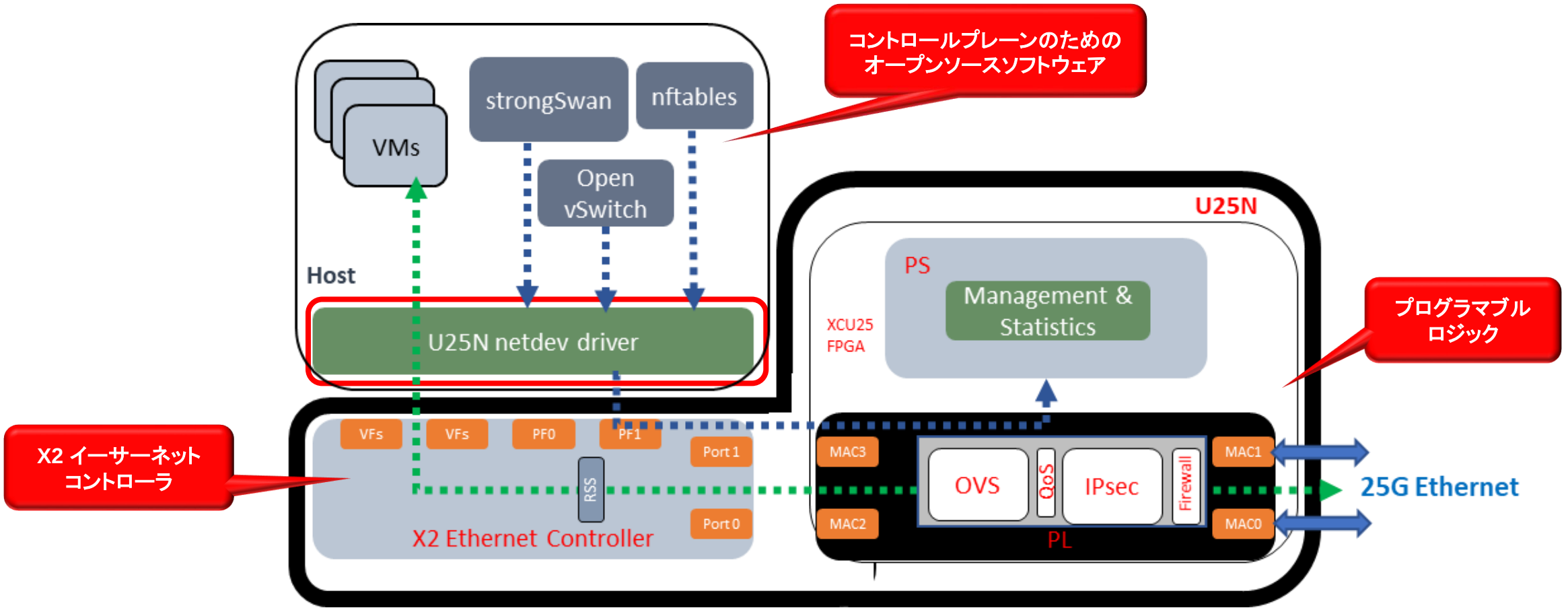
25Gb SmartNIC
ターンキー
HHHL PCIe

SN1022

100Gb SmartNIC
ネットワーク、ストレージ
FHHL PCIe

AMD
XILINX

U25N アーキテクチャ

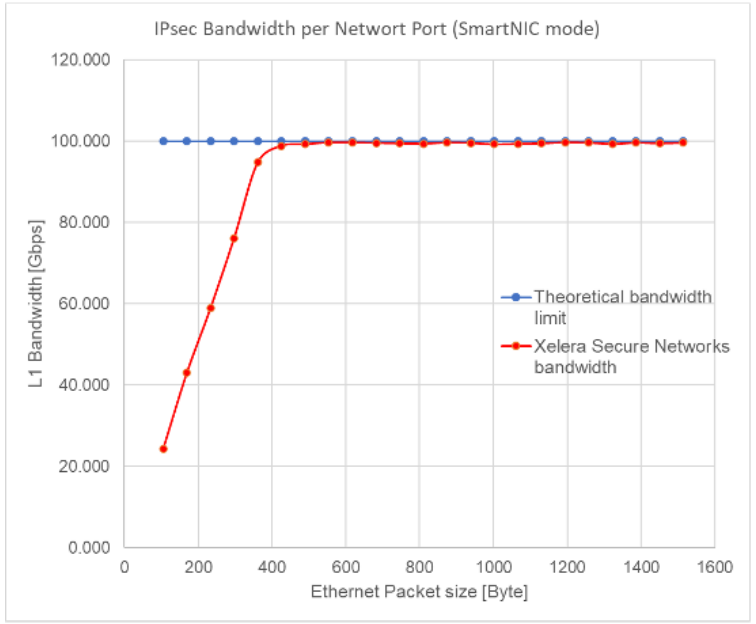


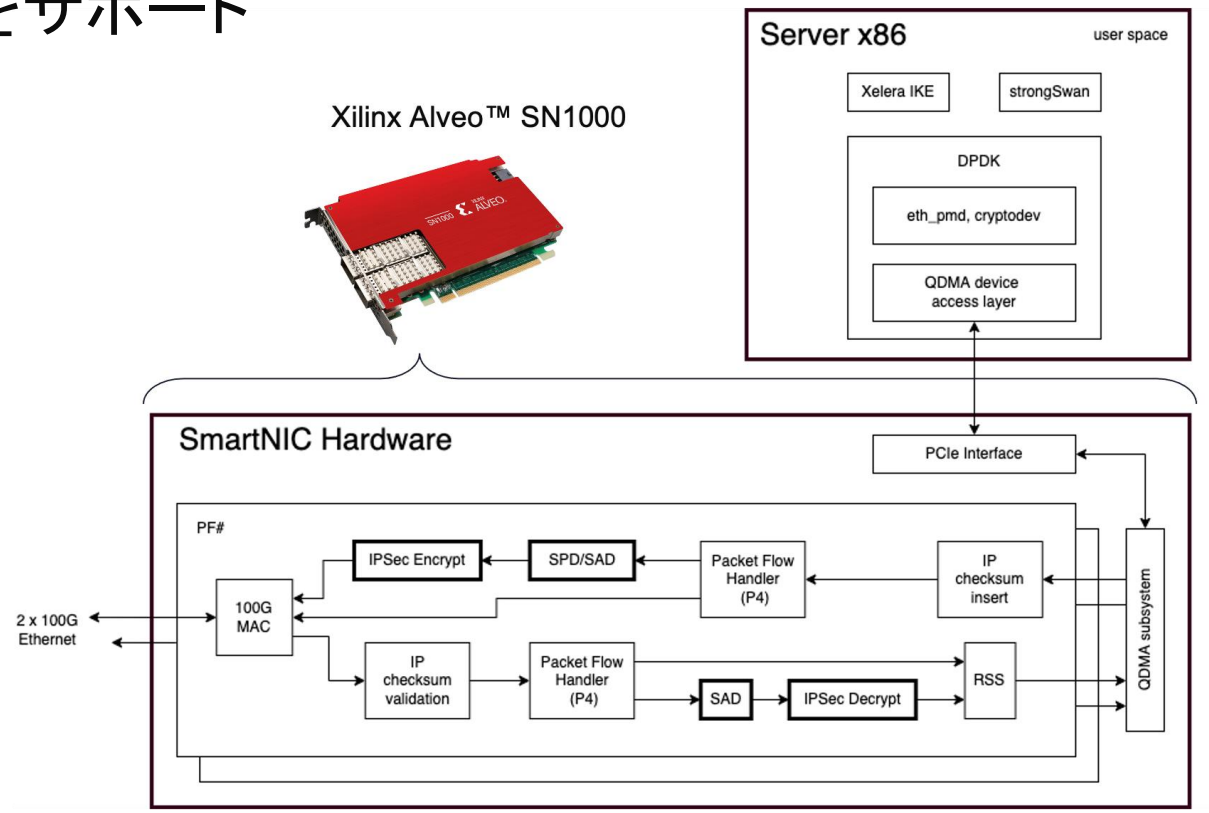
AMD
XILINX

SN1022 活用例 - IPsec 100Gbps x2 -

XELERA

- Xelera社 IPsec を実装、2x100Gbps ラインレートを実現
- トランスポートモード、トンネルモードをサポート

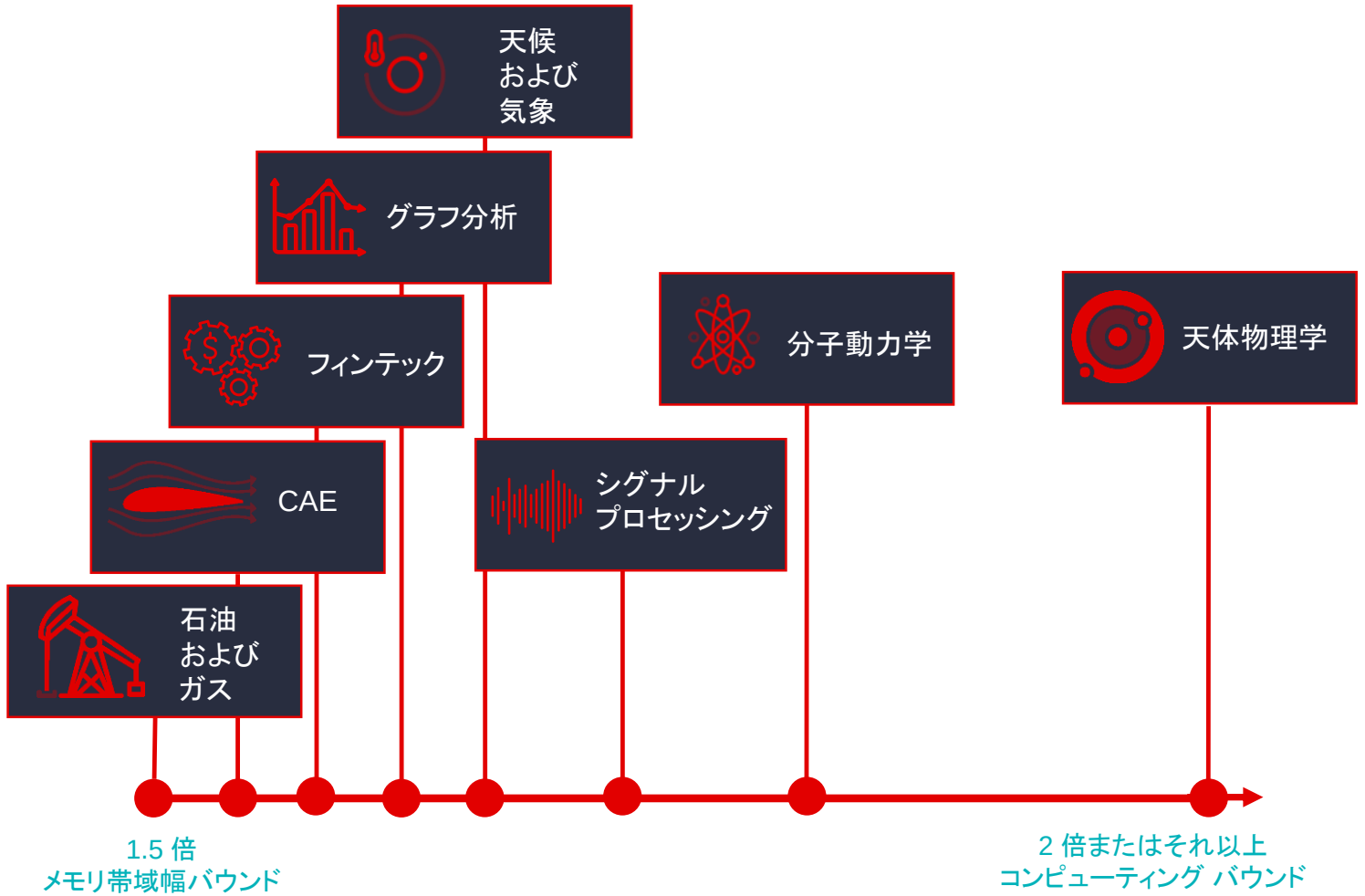




AMD XILINX

U55C: HPC およびビッグデータ ワークロードに特化した設計

- 多くの HPC ワークロードは、コンピューティング バウンドまたはメモリ帯域幅バウンドである…
- I/O 要件は時間の経過と共に指数関数的に増加していく…
- 消費電力はデータセンターで大きな問題になっている
- HPCには膨大な計算能力と高帯域メモリの双方が要求される
- これに対応して、ザイリンクス史上最もパワフルなアクセラレータを開発し、容易に拡張可能にした



AMD
XILINX

AMDザイリンクス最高性能のアクセラレータ

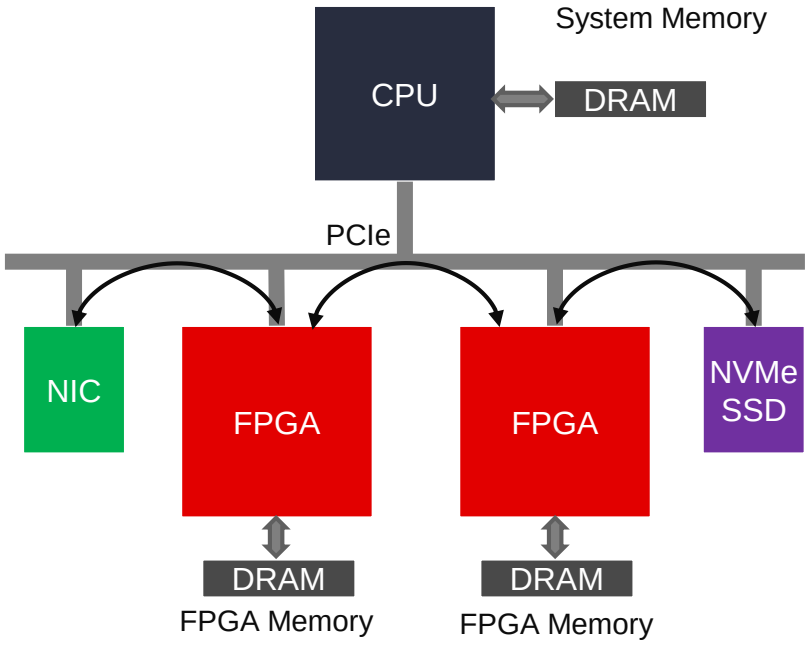
		Alveo U280	Alveo U55C
寸法	幅	デュアル スロット	シングル スロット
	フォーム ファクター、パッシブ フォーム ファクター、アクティブ	フルハイト、¾ レングス フル ハイト、フルレングス	フル ハイト、ハーフ レングス
ロジック	ルックアップ テーブル	1,304K	1,304K
	レジスタ	2,607K	2,607K
	DSP スライス	9,024	9,024
DRAM メモリ	DDR フォーマット	2x 16GB 72b DIMM DDR4	—
	DDR 総容量	32GB	—
	DDR 最大データ レート	2400MT/s	—
	DDR 総帯域幅	38GB/s	—
	HBM2 総容量	8GB	16GB
	HBM2 総帯域幅	460GB/s	460GB/s
内部 SRAM	総容量	43MB	43MB
	総帯域幅	35TB/s	35TB/s
インターフェイス	PCI Express®	Gen3 x16	Gen3 x16、2x Gen4 x8
	ネットワーク インターフェイス	2x QSFP28	2x QSFP28
電源/熱管理	冷却	パッシブ、アクティブ	パッシブ
	標準電力	100W	115W
	最大電力	225W	150W

AMD
XILINX

PCIe Peer-to-Peer (P2P)

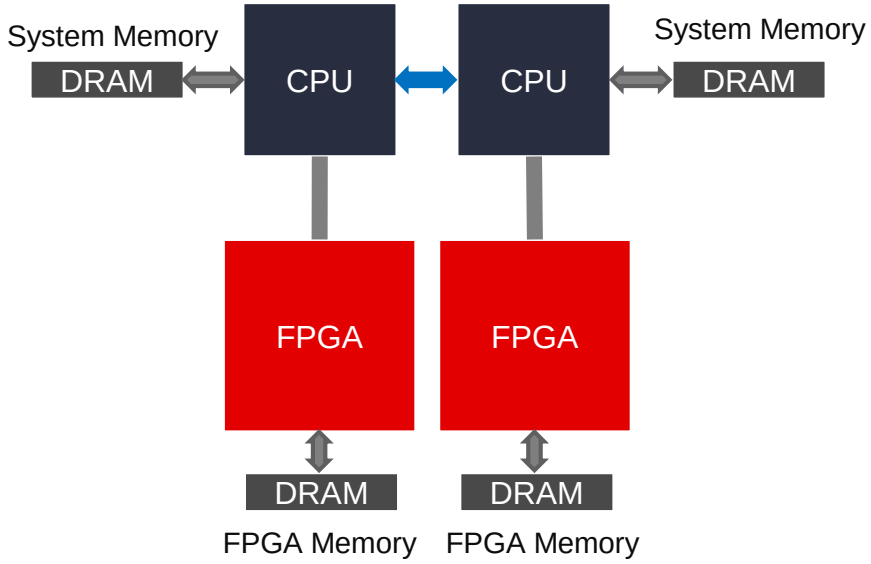
https://xilinx.github.io/XRT/master/html/p2p.html

- Direct data transfer between FPGAs or other PCIe devices without using the host CPU memory



Not Supported

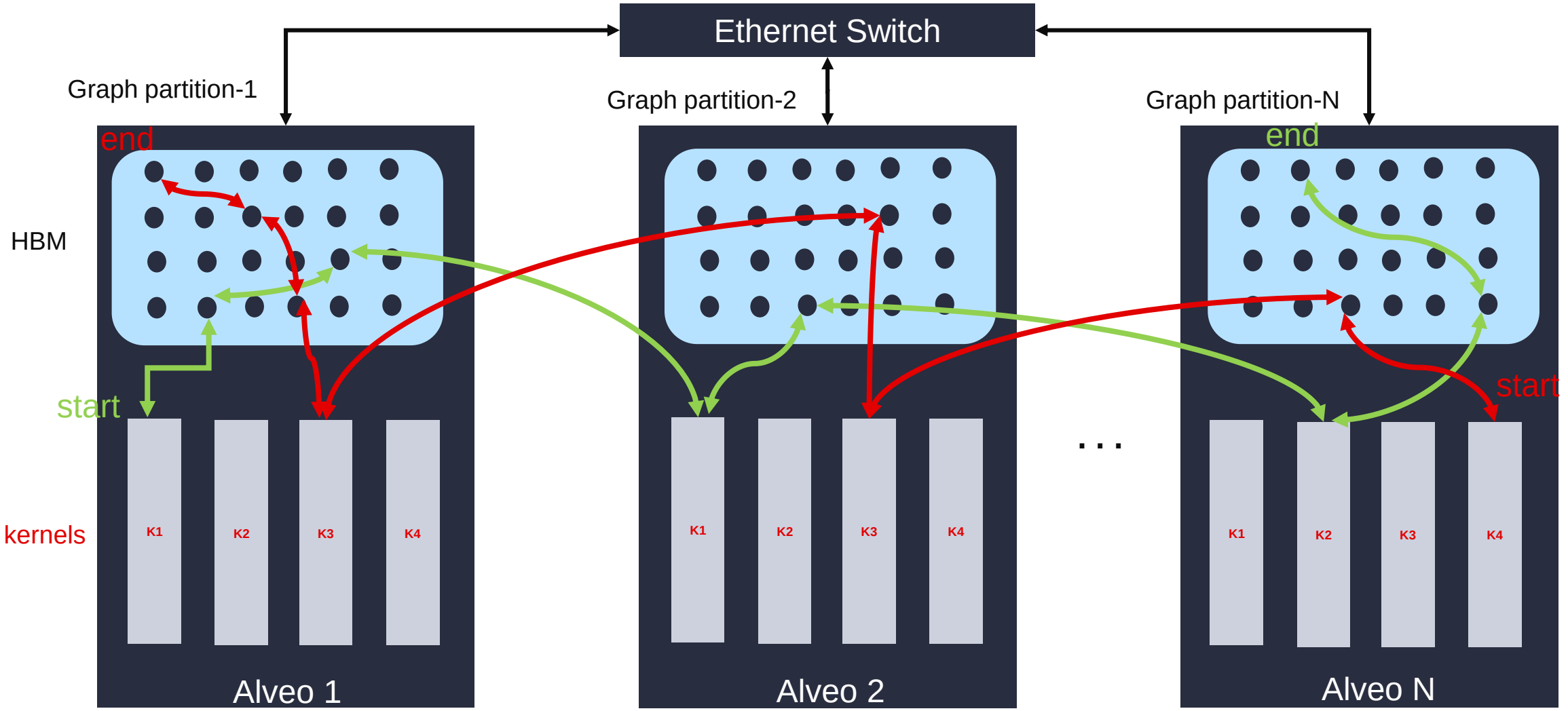
(Crossing different IO Hub)



- P2P Example between FPGA and NVMeSSD
  - https://github.com/Xilinx/Vitis_Accel_Examples/tree/master/host/p2p_simple
  - https://github.com/Xilinx/Vitis_Accel_Examples/tree/master/host/p2p_bandwidth

AMD XILINX

Example : N-hop query for ML Feature Extraction



Large number of U55C networked using Ethernet for big graph

AMD XILINX

Heterogeneous Accelerated Compute Clusters (HACC) program

- A special initiative to support novel research in adaptive compute acceleration for high performance computing (HPC).
- The scope of the program is broad and encompasses systems, architecture, tools and applications.
- This program was previously known as the XACC program - Xilinx Adaptive Compute Clusters.

(ご参考)

紹介サイト; https://japan.xilinx.com/support/university/XUP-XACC.html

紹介サイト (動画); https://japan.xilinx.com/program/developer-program/adapt-2021-session-videos/xacc-researc-initiative.html

https://xilinx.github.io/xacc/

Priority research areas for the HACCs include:

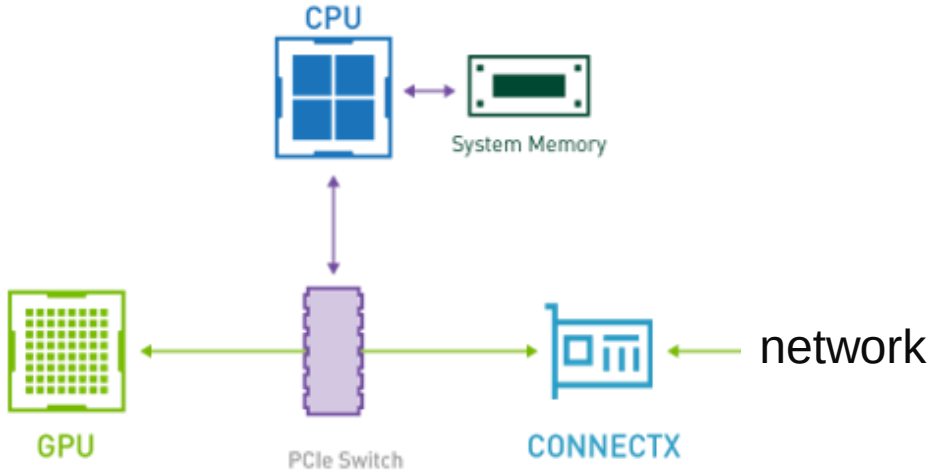
Adaptive compute acceleration
High performance computing (HPC)
Machine Learning
Database acceleration
Energy efficiency
Compilers
IOT
Computer architecture

AMD
XILINX

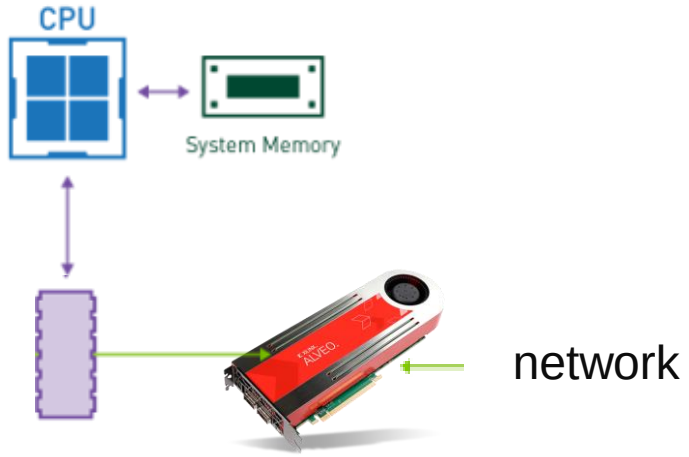
ACCL: FPGA-Accelerated Collectives over 100 Gbps TCP-IP - Architecture -

ACCL is a Vitis kernel and associated Pynq and XRT drivers which together provide MPI-like collectives for Xilinx FPGAs. ACCL is designed to enable compute kernels resident in FPGA fabric to communicate directly under host supervision but without requiring data movement between the FPGA and host. Instead, ACCL uses Vitis-compatible TCP and UDP stacks to connect FPGAs directly over Ethernet at up to 100 Gbps on Alveo cards.

- MPI for CPUs
  - OpenMPI
- MPI-like lib for GPUs
  - NCCL and RCCL



- MPI-like lib for FPGAs
  - Direct access to network
  - Process on network data



https://xilinx.github.io/xacc/publications.html Paper GitHub

FPGA Collective - State of the art

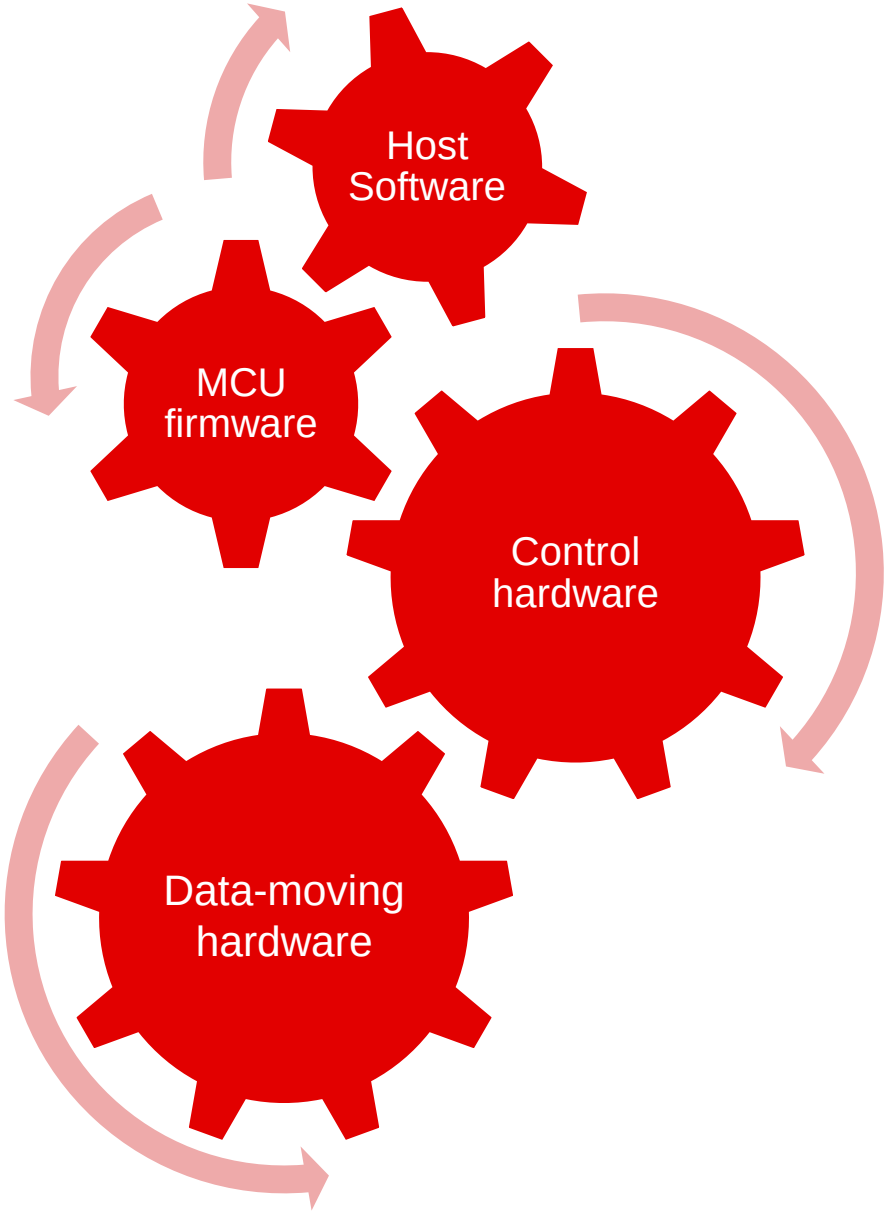
- Existing FPGA-based collective solutions fall short in certain aspects

Solution	Performance	Flexibility	Portability
Easynet	High (~90 Gbps)	Low	High
SMI	Medium (~40 Gpbs)	Low	Low
Galapagos	Low (< 10Gbps)	Low	High
ZRLMPI	Low (< 10Gbps)	Low	High
TMD-MPI	Low (< 10Gbps)	High	Low
ACCL(OUR)	High (~80 Gbps)	High	High

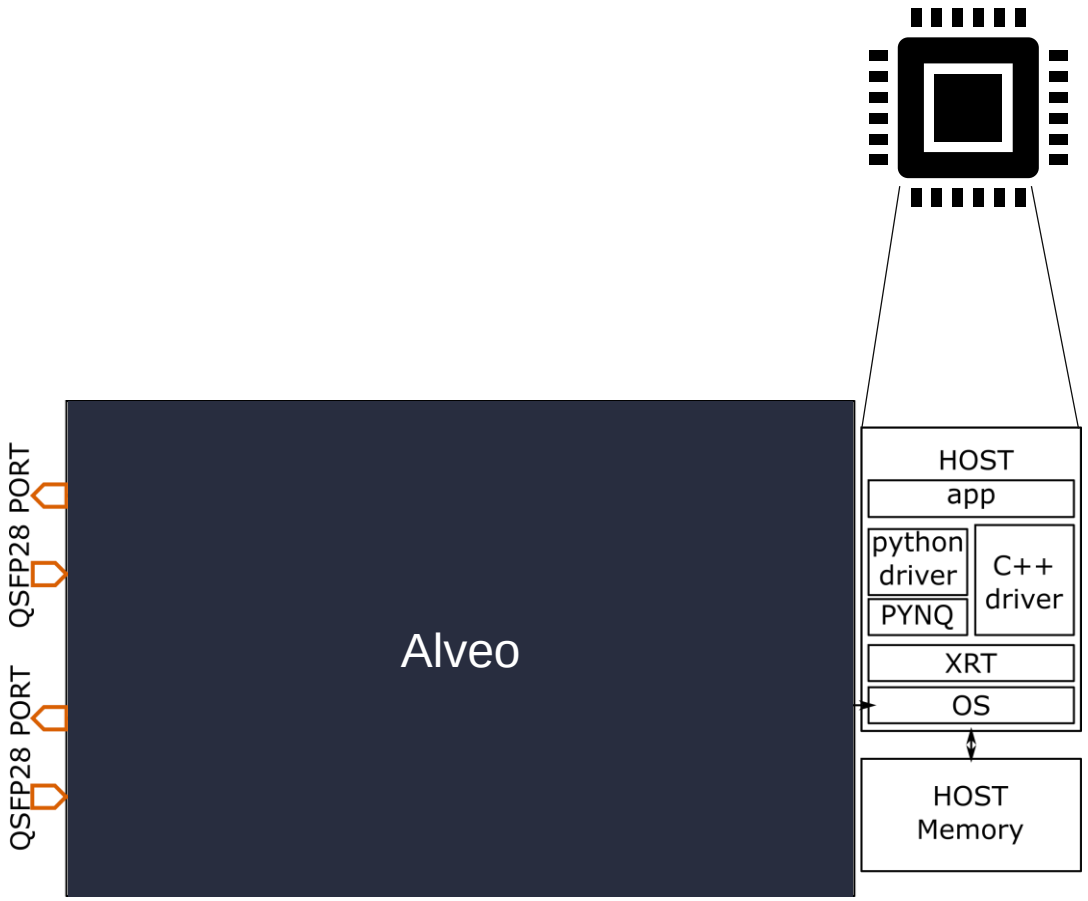
ACCL Overview

ACCL ; Accelerated Collective Communication Library

https://github.com/Xilinx/ACCL



ACCL Host Overview



Main modules:

- Software Stack
  - User application
  - ACCL drivers (Python or C++)
  - Xilinx Run-Time (XRT)

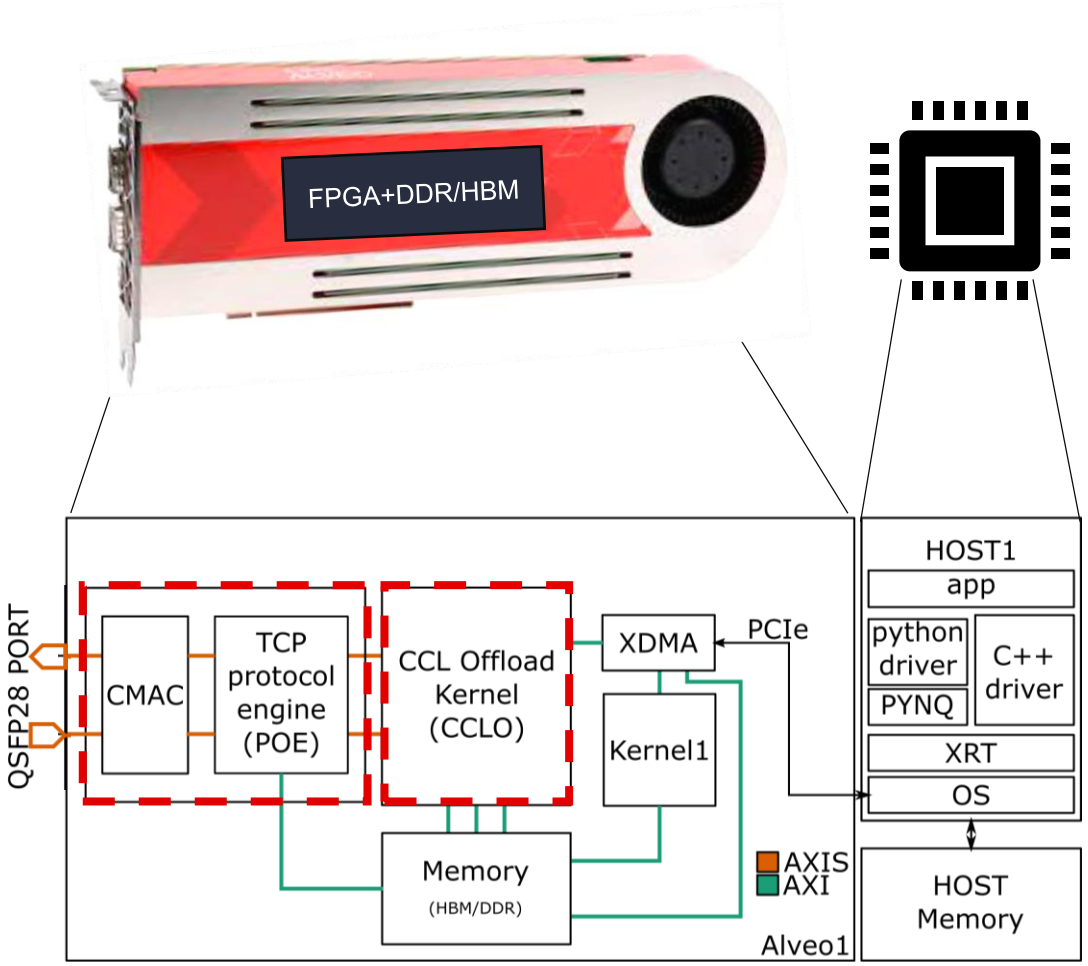
ACCL Host Software Example (Distributed NN Inference)

```
from pynq import Overlay, allocate
from mpi4py import MPI
#receive binfile, ranks_dict as inputs
ol = Overlay(binfile)
accl = ol.cclo
rank = MPI.COMM_WORLD.Get_rank()
bs = 16384

accl.setup_rx_buffers(nbufs=16, bufsize=bs,
devicemem=ol.bank0)
accl.configure_communicator(ranks_dict, rank)
accl.open_port(); accl.open_con()

txb=allocate((bs,), target=ol.bank0)
rxb=allocate((bs,), target=ol.bank0)


ch = accl.scatter(root=0, txb,
rxb, from_fpga=True, async=True)

ch = nn_accelerator.call(rx_buffer, waitfor=[ch])

accl.gather(root=0, txb, rxb, to_fpga=True, async=False,
waitfor=[ch])

accl.deinit() #releases FPGA memory, resets CCLO
```

ACCL collectives:

- Initialize communicator
- Allocate memory
- Invoke collectives:
  - Pass buffer pointers
  - Move data to FPGA (optional)
- Non-blocking collective calls in host
  - Asynch calls to pipeline collectives
  - Chaining execution

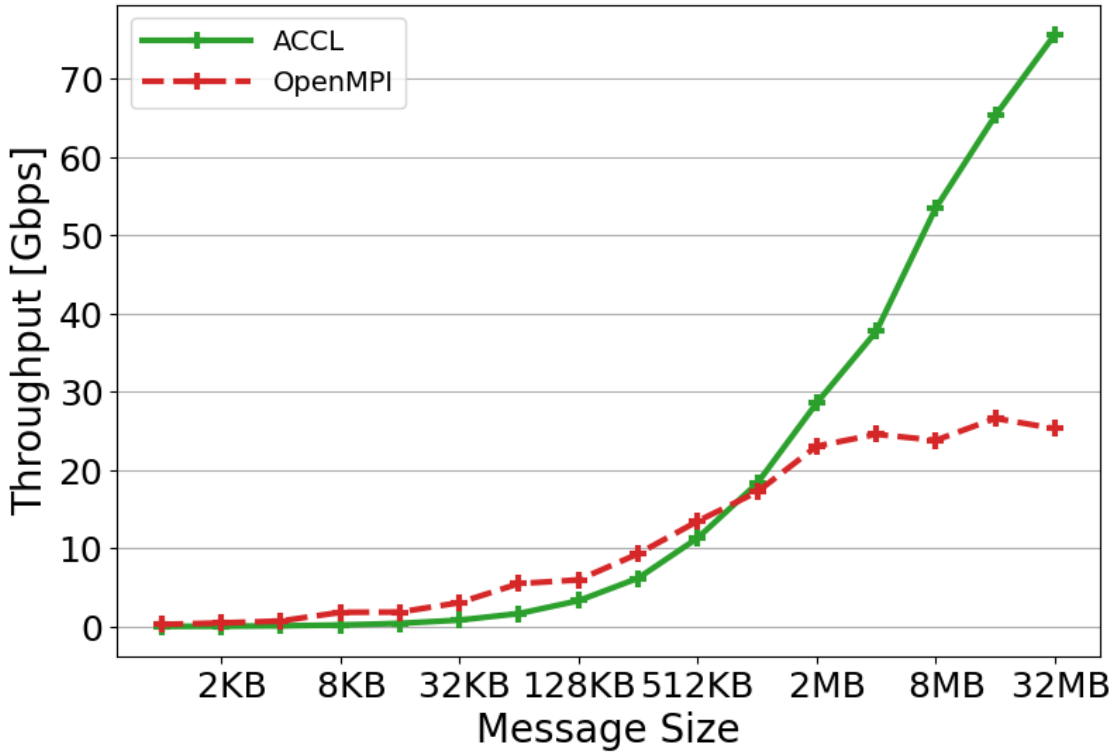
ACCL FPGA Overview



- Collective Offload engine (CCLO)
- Network Protocol Offload Engine (TCP/UDP POE)
- Interfaces to external Vitis kernels that access ACCL via:
  - Streaming interfaces
  - Off-chip memory (DDR/HBM)

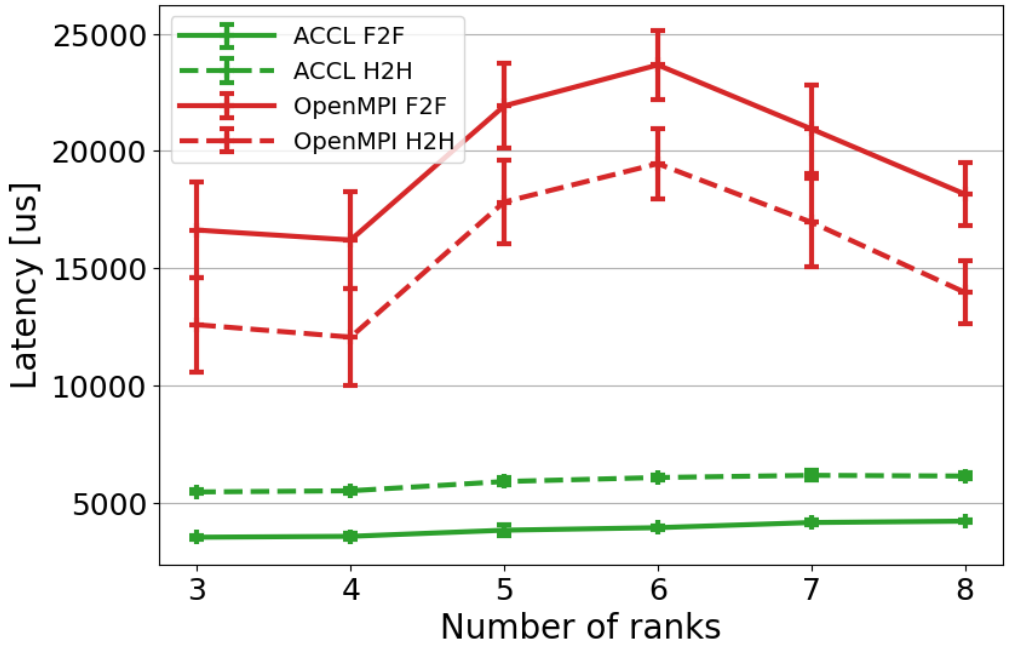
Send&Recv throughput

- Performance on U280 and U250 is similar, showing that our design is **portable**
- ACCL achieves **higher throughput** than OpenMPI for messages larger than 1MB
- Large message size to compensate initialization overhead



Scalability and Resource Consumption

- All-reduce
  - from 3 to 8 ranks
  - message size 8MB
  - 100 runs, average & variance
- Compared to OpenMPI, ACCL has
  - Lower increase in execution time
  - Lower jitter
- Resource consumption
  - Major dedicated to TCP POE
  - ACCL takes 15% LUTs on a U250
  - Enough space for computation kernel



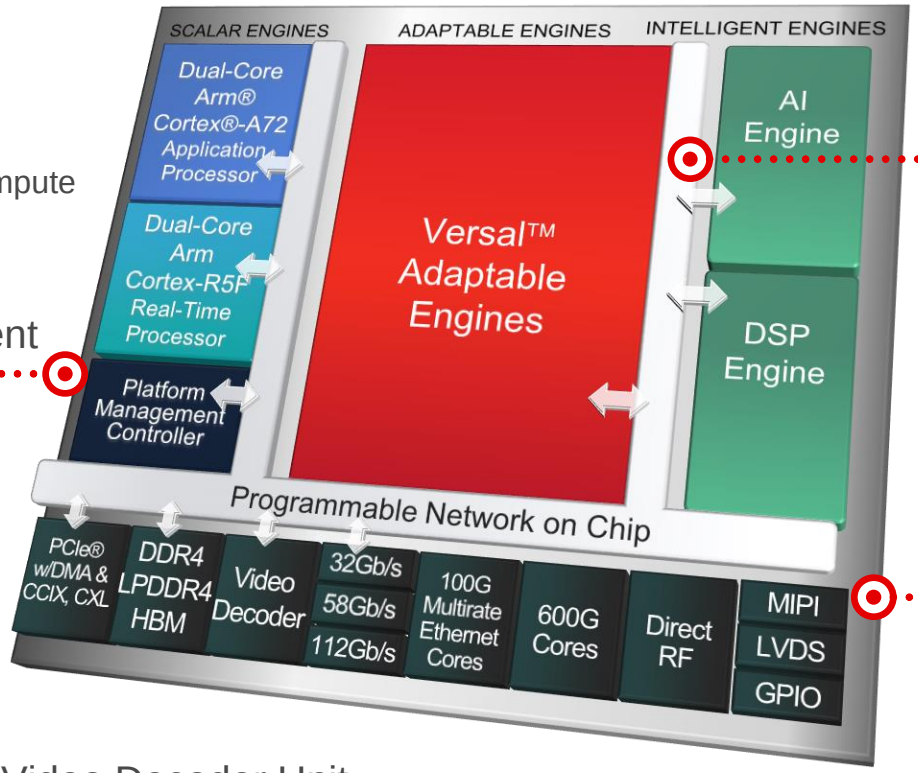
Component	kLUT	DSP	BRAM18	URAM
CCLO	78	56	169	0
TCP POE	111	0	813	1
UDP POE	23	0	115	0
CMAC	12	0	34	9

Versal™ Architecture - Comprehensive Overview

Adaptable Engines
- Re-architected for faster timing closure
- Tune for power vs. performance
- Adaptable to any workload

Intelligent Engines
- AI Compute
- Diverse DSP Workloads

Scalar Engines
- Platform Control
- Embedded Edge Compute



Programmable NoC
- Guaranteed Bandwidth
- Enables SW Programmability

SW-Controlled Platform Management

Programmable I/O
- Any interface or sensor
- Includes 3.2Gb/s MIPI

PCIe Gen5, CCIX, CXL
- 2X PCIe® & DMA bandwidth
- Cache-coherency

Protocol Engines
- 400G/600G cores
- Power-Optimized

DDR4 Memory
- 3200-DDR4, 4266-LPDDR4
- 2X bandwidth/pin

Dedicated Interfaces

RF
- Direct-RFADC and RF-DAC
- SD-FEC

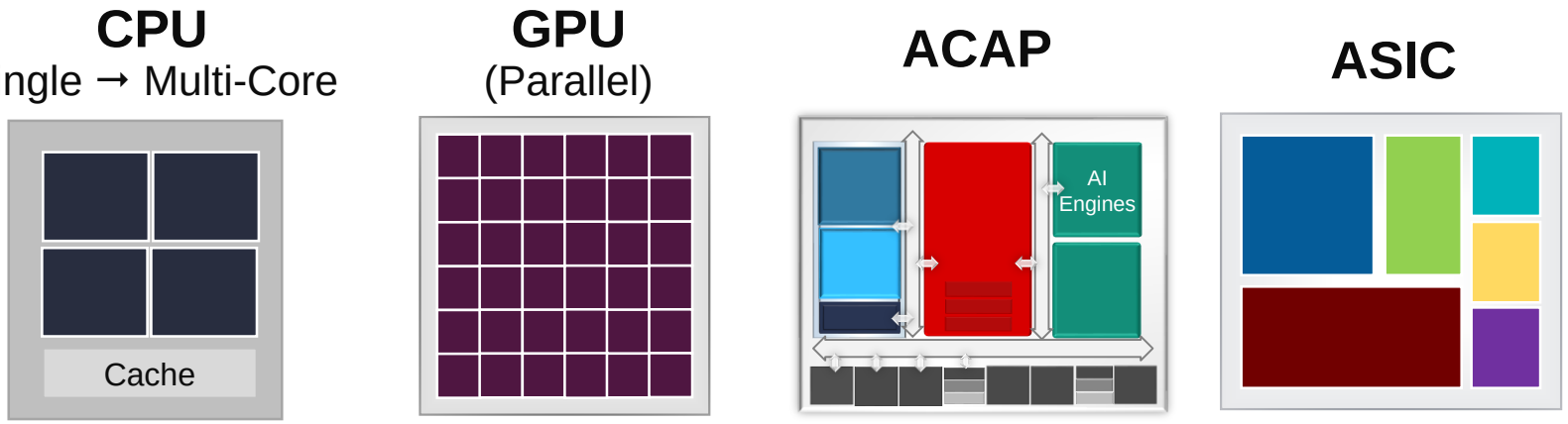
HBM
- 820GB/s Bandwidth
- 32GB Capacity

Video Decoder Unit
- H.264/H.265 decoding
- Up to 4x 4K UHD (or 32x 1080p)

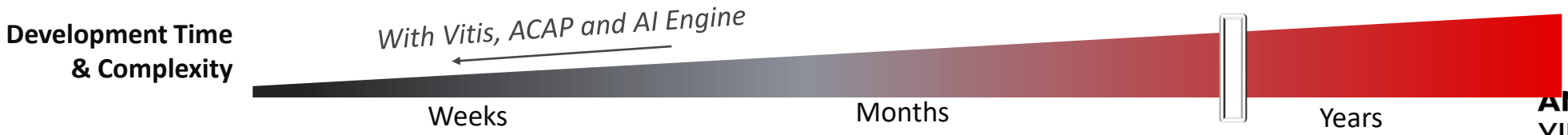
Transceiver Leadership
- Broad range, 1G →112G
- 58G in mainstream devices

AMD XILINX

Delivering Adaptable Compute Acceleration

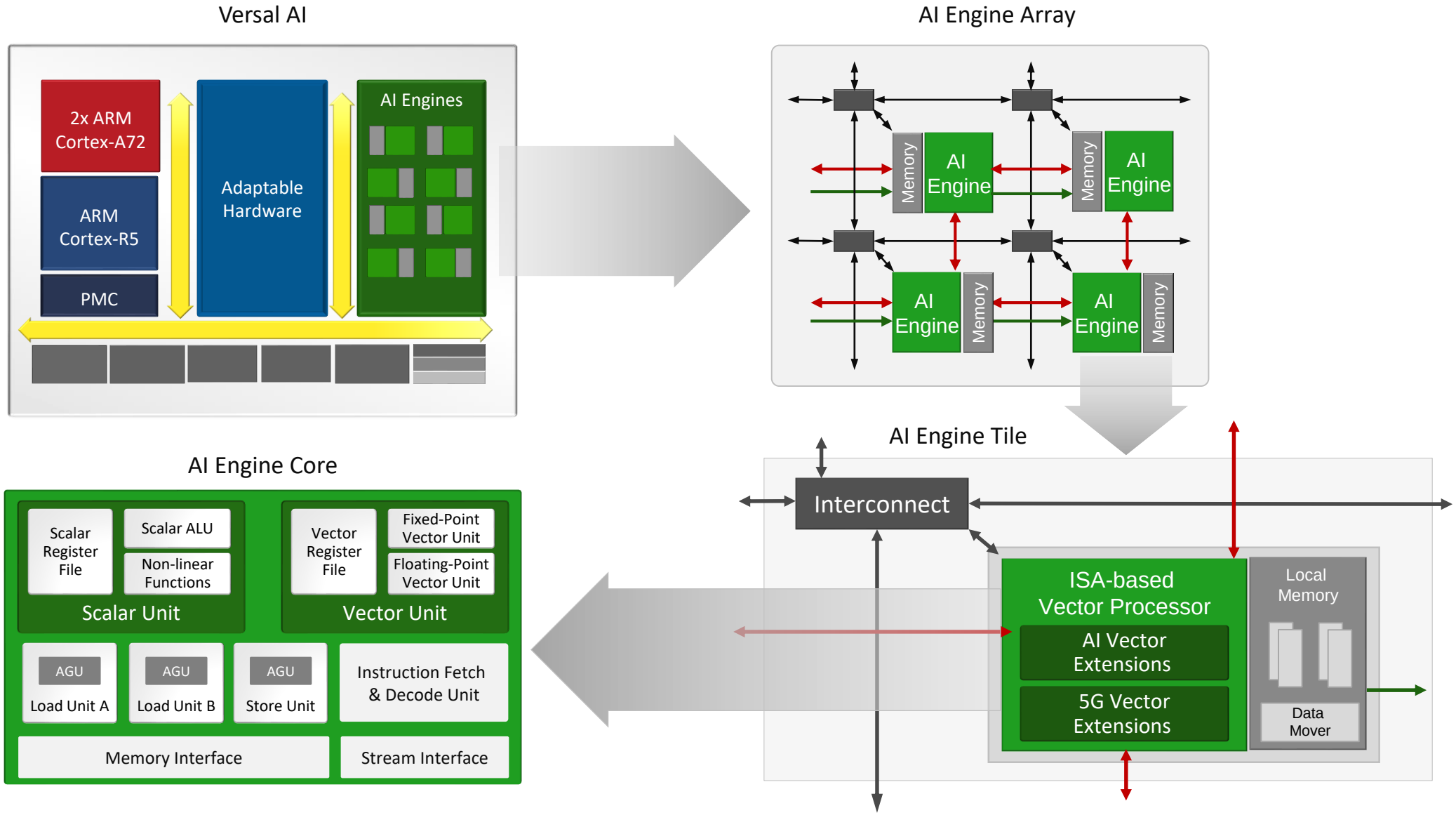


SW Programmable	✓	✓	✓	✓
HW Adaptable	–	–	✓	–
Workload Flexibility	✓	✓	✓	–
Throughput vs. Latency	–	–	✓	✓
Power Efficiency	–	–	✓	✓



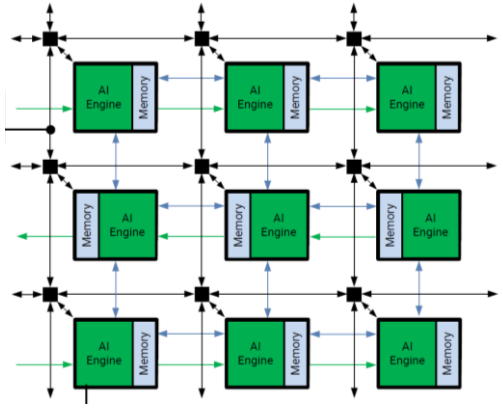
AMD
XILINX

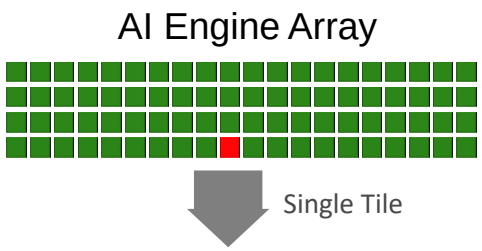
AI Engine: Terminology

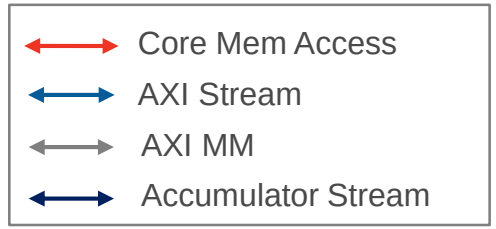


AMD XILINX

AI Engine Tile







AI Engine

- 1+ GHz VLIW / SIMD AI Engine
- 32-bit Scalar RISC processor
- Fixed and floating point vector units

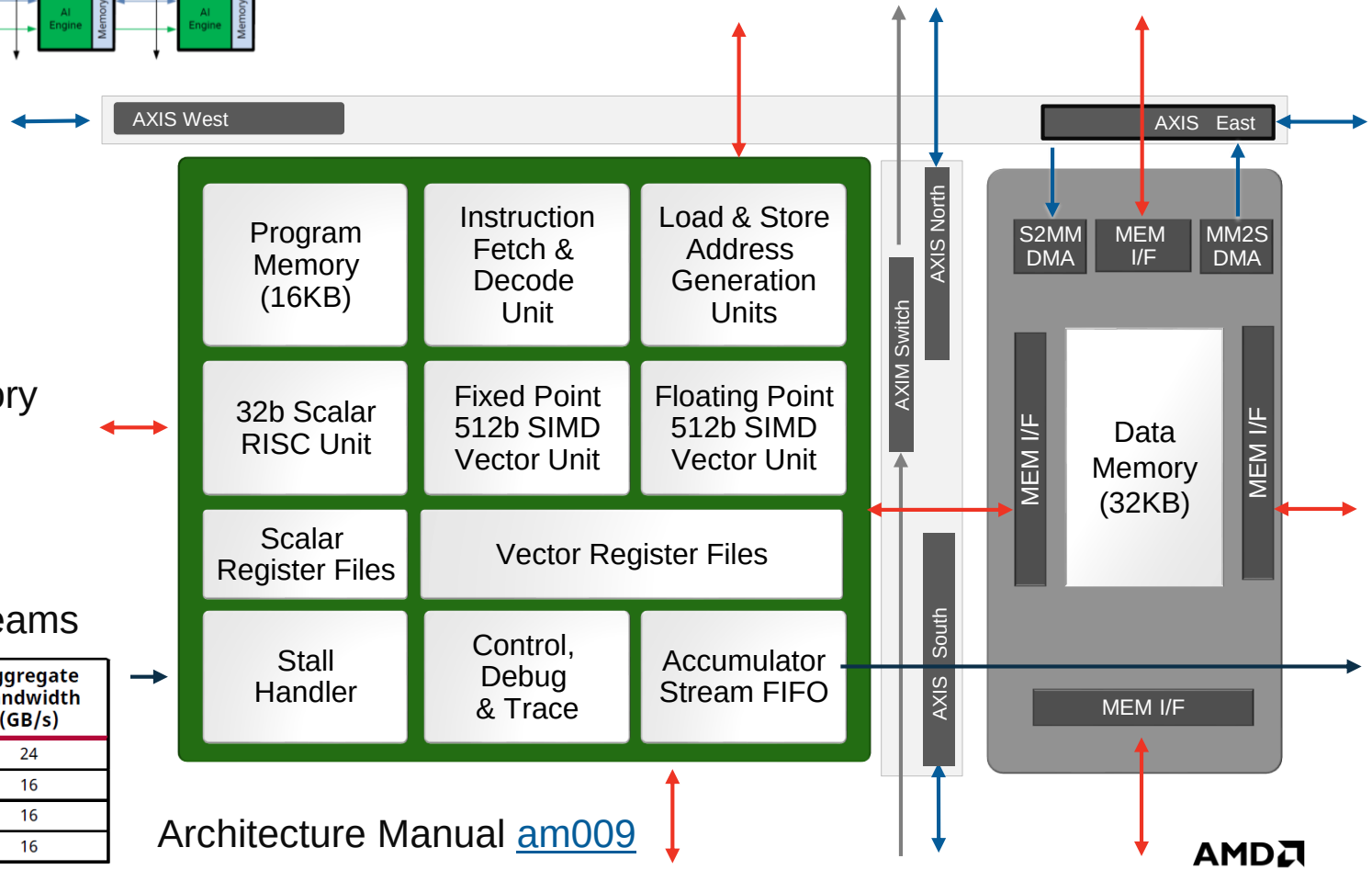
Data Memory

- Each AI Engine can access 4 Memory Modules (N,E,S,W) as one contiguous memory

Interconnect

- AXI-MM switch for config, control, and debug
- AXI-Stream crossbar for routing N/S/E/W streams

Connection Type	Number of Connections	Data Width (bits)	Clock Domain	Bandwidth per Connection (GB/s)	Aggregate Bandwidth (GB/s)
To North/From South	6	32	AI Engine (1 GHz)	4	24
To South/From North	4	32	AI Engine (1 GHz)	4	16
To West/From East	4	32	AI Engine (1 GHz)	4	16
To East/From West	4	32	AI Engine (1 GHz)	4	16



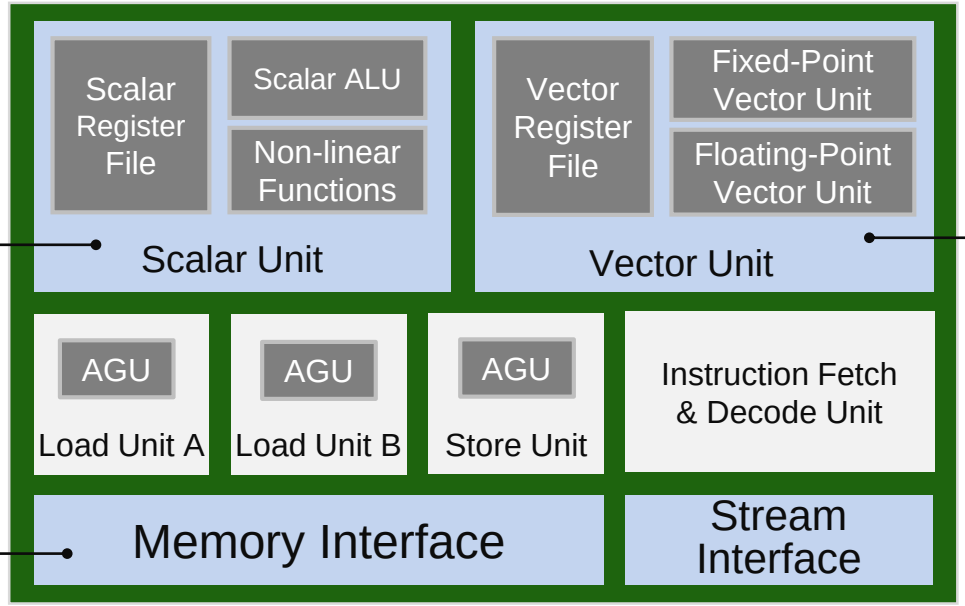
Architecture Manual [am009]

AMD XILINX

AI Engine: Highly Parallel Processor Core

32-bit Scalar RISC Processor

Local, Shareable Memory
- 32KB Local, 128KB Addressable



Vector Processor
512-bit SIMD Datapath

Instruction Parallelism: VLIW

7+ operations / clock cycle
- 2 Vector Loads / 1 Mult / 1 Store
- 2 Scalar Ops / Stream Access

Highly Parallel

Data Parallelism: SIMD

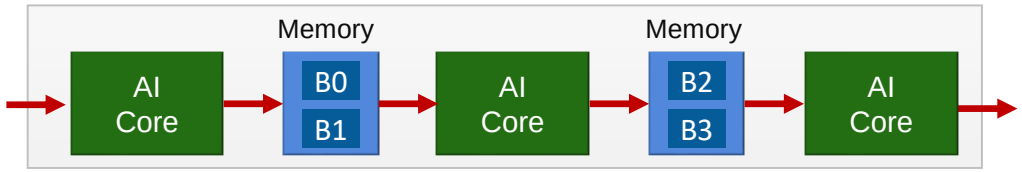
Multiple vector lanes
- Vector Datapath
- 8 / 16 / 32-bit & SPFP operands

Up to 128 MACs / Clock Cycle per Core (INT 8)

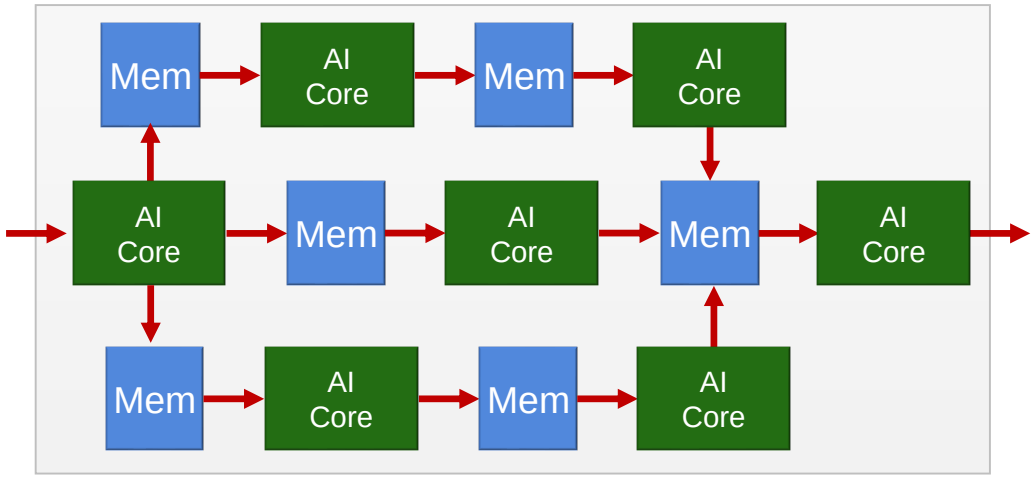
Data Movement Architecture

Memory Communication

Dataflow Pipeline



Dataflow Graph



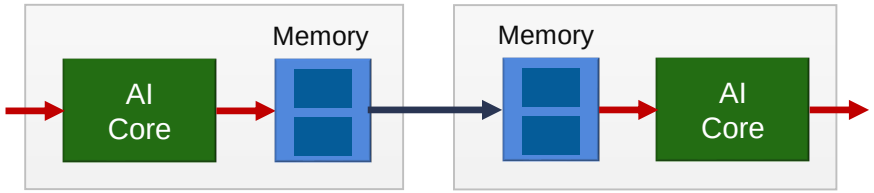
Memory Interface

Stream Interface

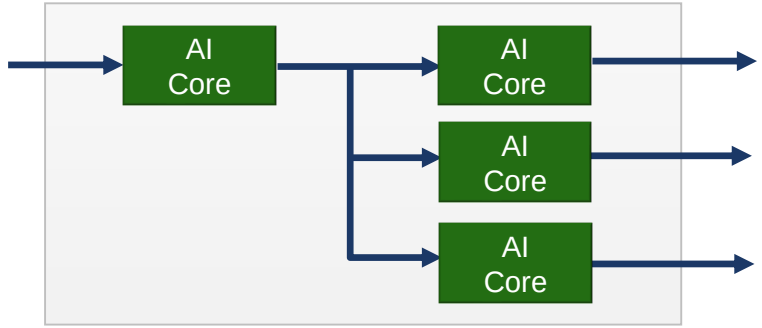
Cascade Interface

Streaming Communication

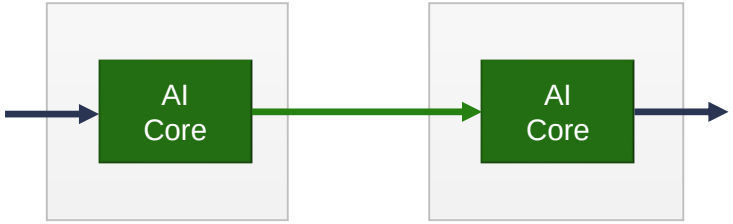
Non-Neighbor



Streaming Multicast

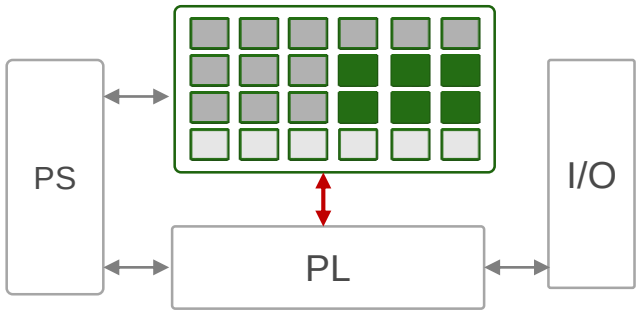


Cascade Streaming

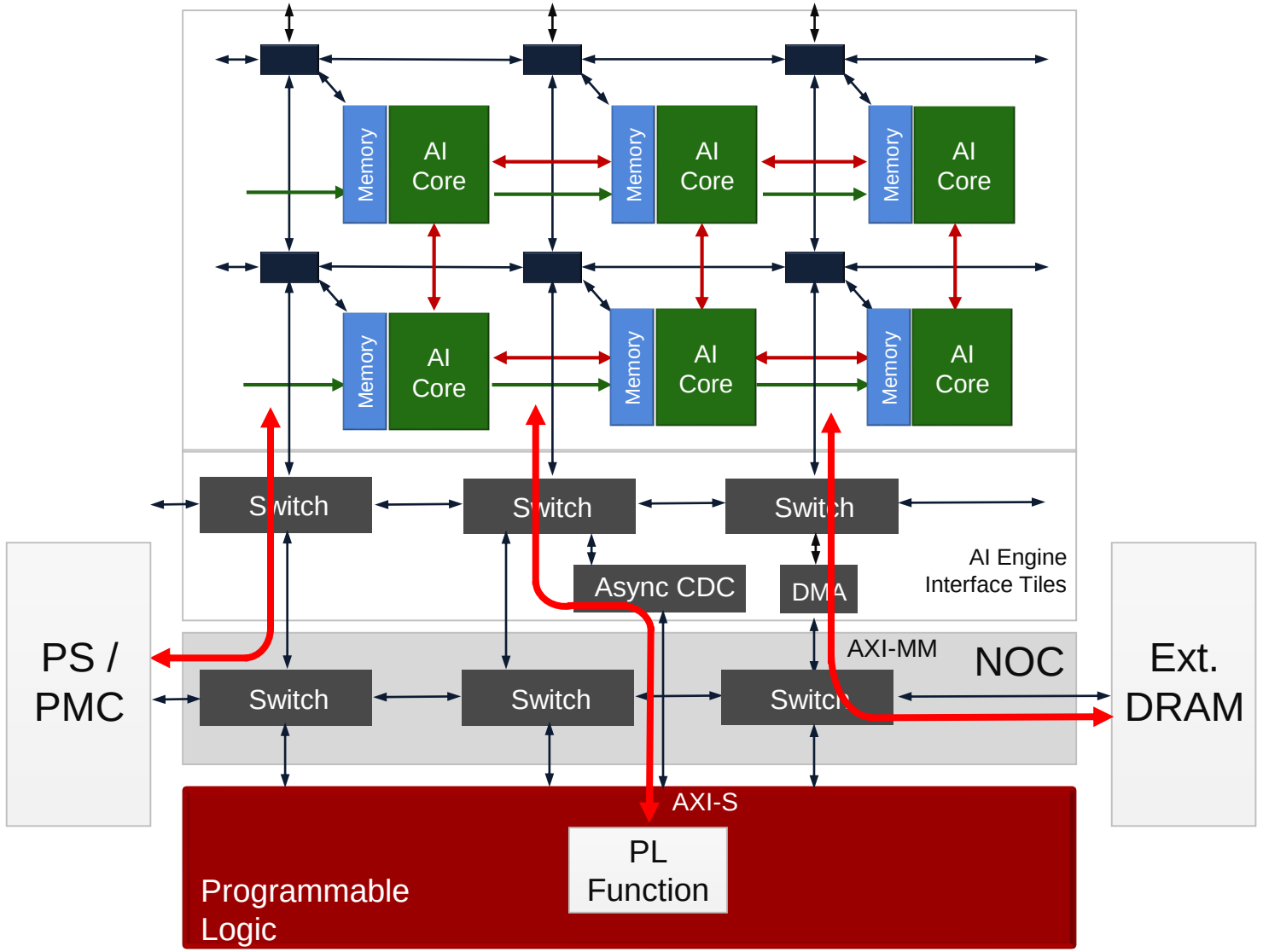


AMD XILINX

AI Engine Integration with Versal



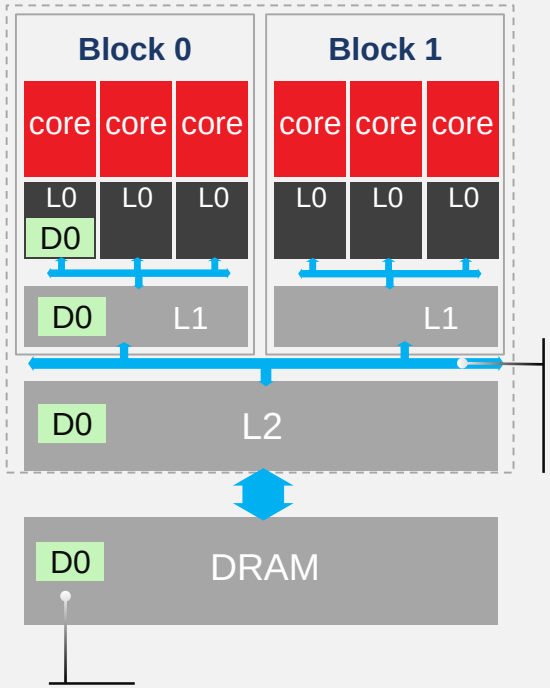
- **TB/s of Interface Bandwidth**
  - AI Engine to Programmable Logic
  - AI Engine to NOC
- **Leveraging NOC connectivity**
  - PS manages Config / Debug / Trace
  - AI Engine to DRAM (no PL req'd)



AMD XILINX

AI エンジン: マルチコア コンピュート に革新をもたらす

従来のマルチコア
(キャッシュ アーキテクチャ)

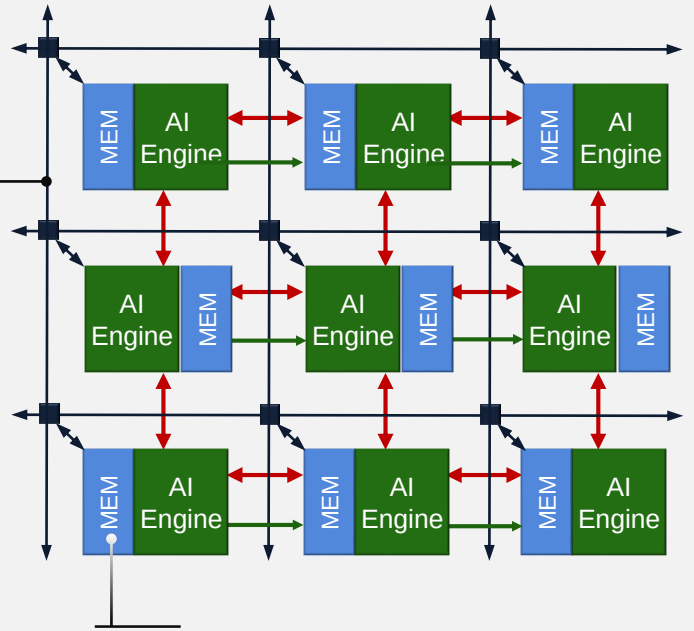


固定した共有接続
- システム性能を制約
- レイテンシーの大幅なばらつき

データの複製
- レイテンシーの大幅な増大とばらつき
- 帯域不足による性能制約
- 消費電力の大幅な増大

AI エンジン アレイ
(インテリジェント エンジン)

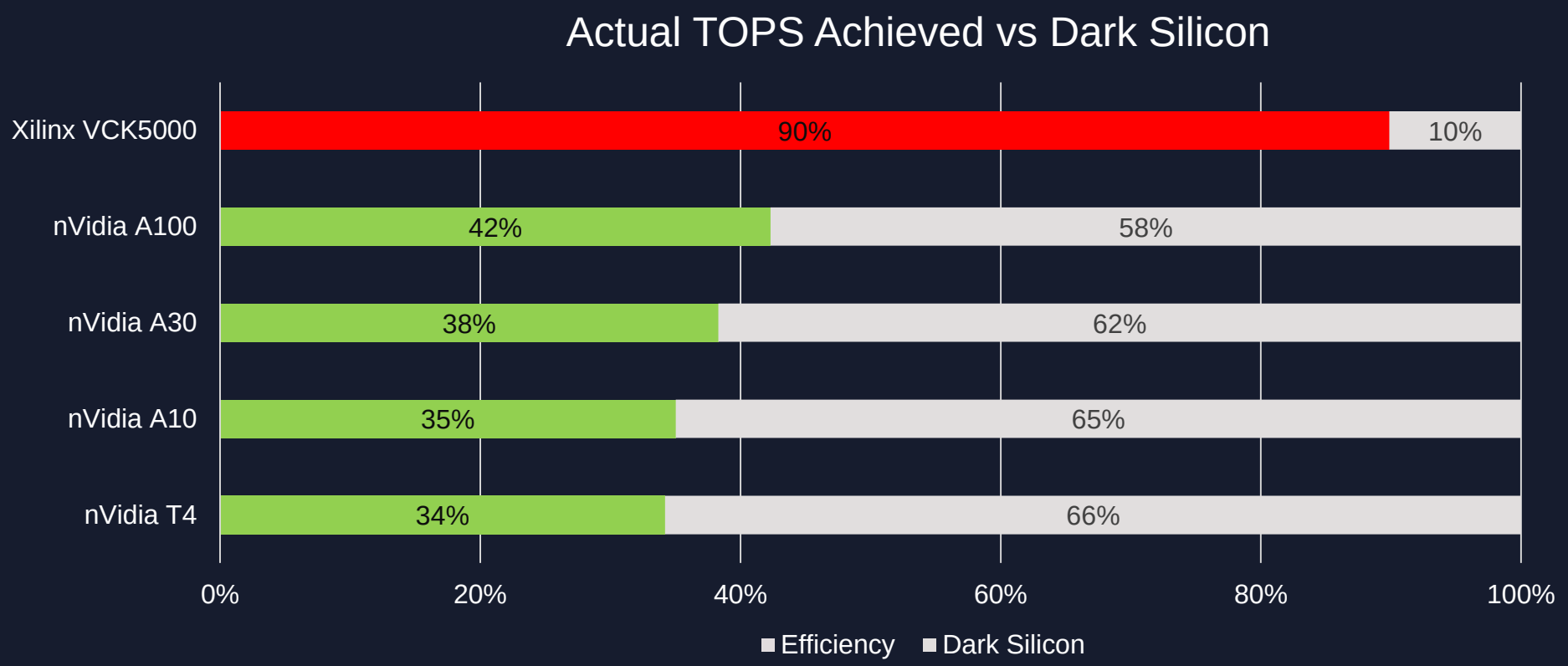
専用の接続
- システム性能の制約とならない
- レイテンシーは短く、かつ確定的



密結合したメモリを分散
- キャッシュミスは無し
- レイテンシーは短く、かつ確定的
- システム性能の制約とならない高帯域
- 全体のメモリサイズを節約
- 消費電力を大幅に低減

XILINX

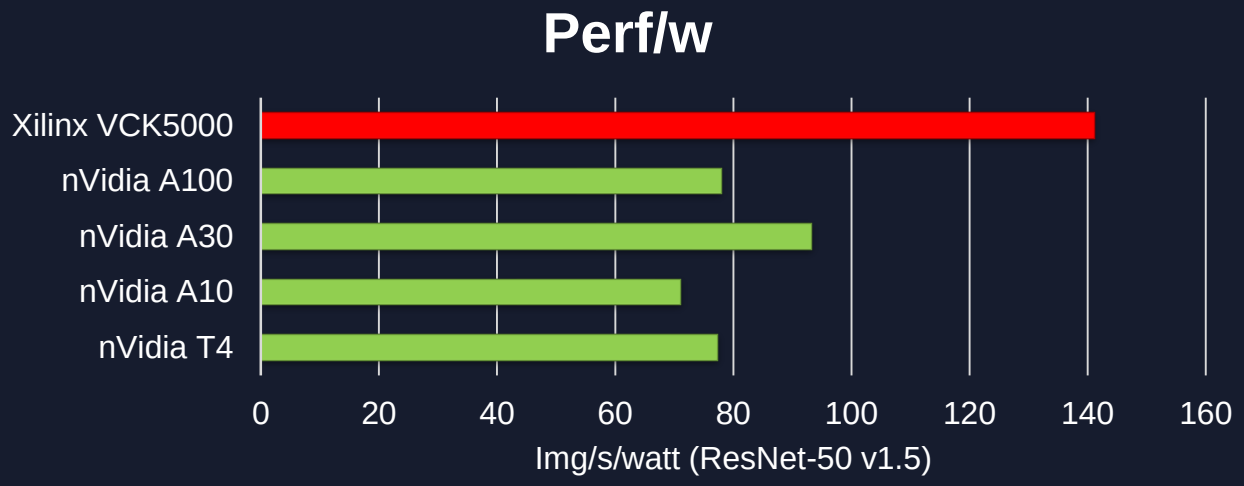
The World's First "0 Dark Silicon" AI Accelerator

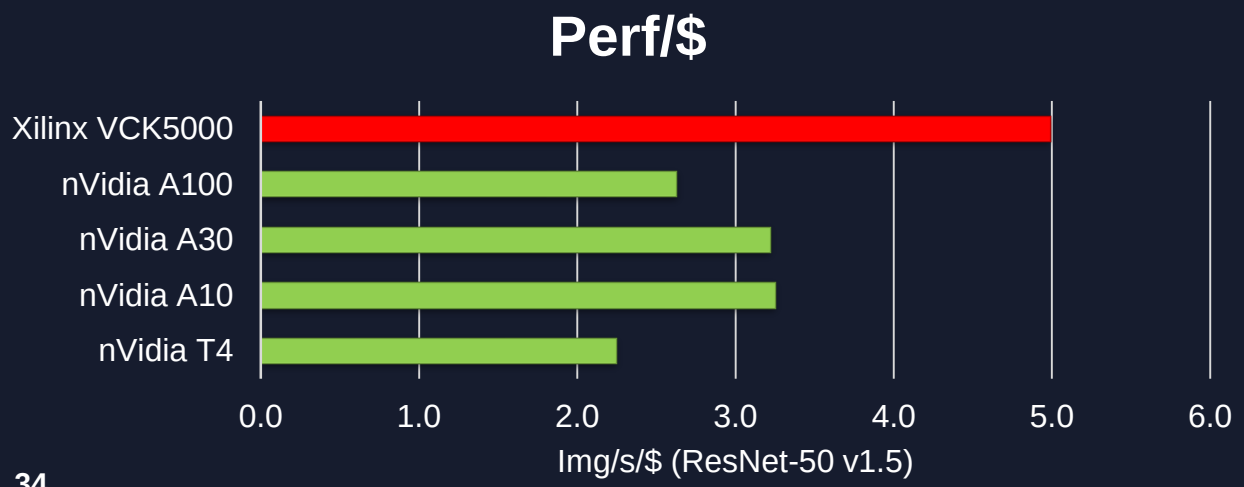


Near 100% efficiency: Achieving True Peak TOPS at Real AI Model Workloads

AMD
XILINX

Double Performance / Watt / $ vs Nvidia Flagship AI Cards





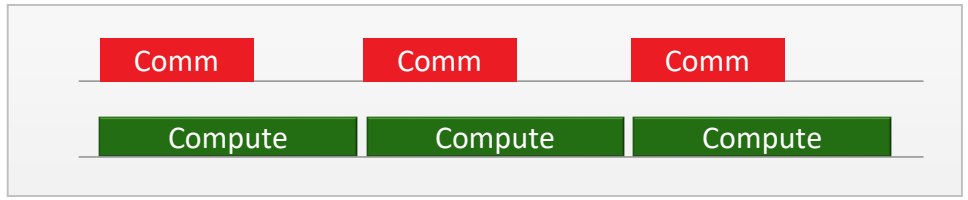
	ResNet-50 (img/s)	Power	SRP**
VCK5000	13,700	97W	$2,745
A100 SXM	32,204	413W	$12,235*
A30	15,411	165W	$4,787
A10	10,676	150W	$3,283
T4	5,423	75W	$2,410

* A100 SXM pricing not available, using A100 PCIe 80GB pricing instead. SXM price is typically more expensive than PCIe
** SRP captured from acmemicro.com as of Feb 22, 2022

AMD
XILINX

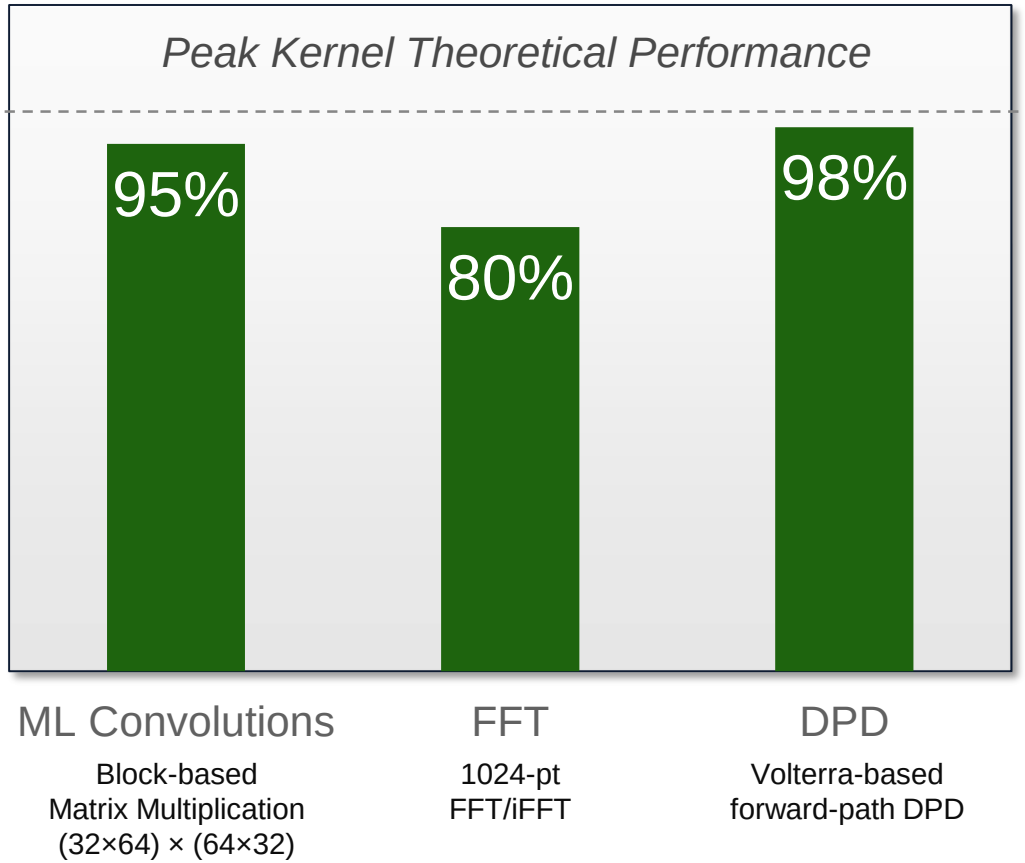
AI Engine Delivers High Compute Efficiency

- **Adaptable, 'non-blocking' interconnect**
  - Flexible data movement architecture
  - Avoids interconnect "bottlenecks"
- **Adaptable memory hierarchy**
  - Local, distributed, shareable = extreme bandwidth
  - No cache misses or data replication
  - Extend to PL memory (BRAM, URAM)
- **Transfer data while AI Engine Computes**



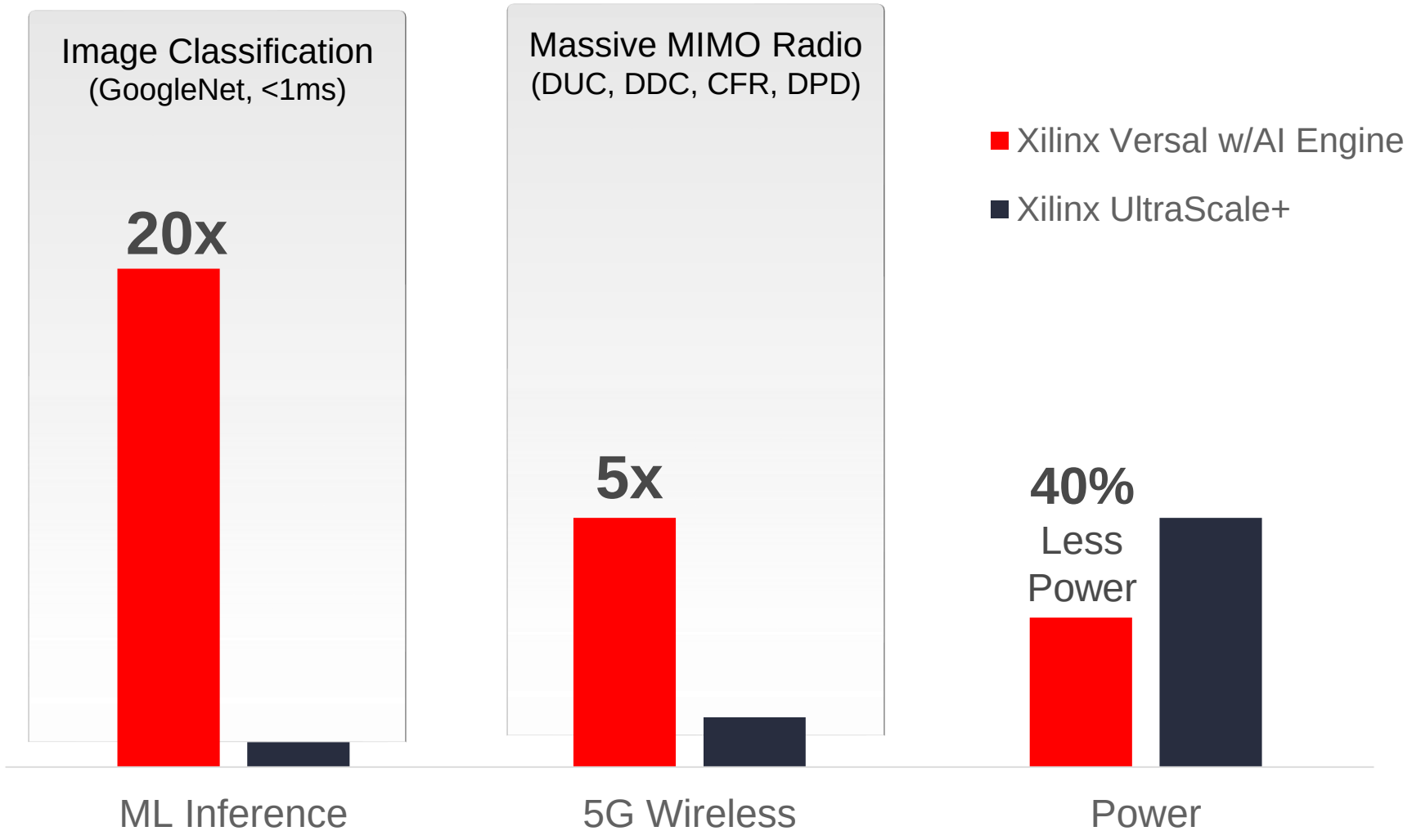
Overlap Compute and Communication

Vector Processor Efficiency



AMD XILINX

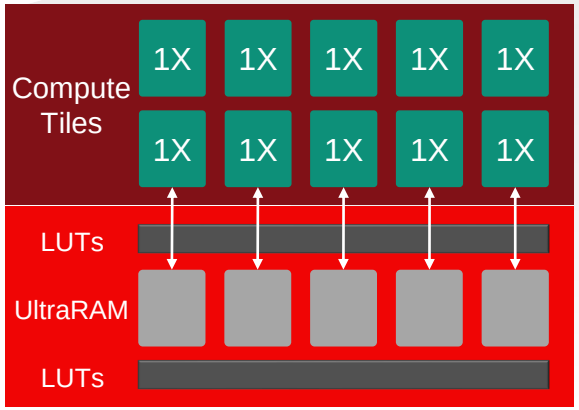
AI Engine Application Performance & Power Efficiency



AMD
XILINX

Intelligent Engines Optimized for Any AI Application

AI Engine Architecture



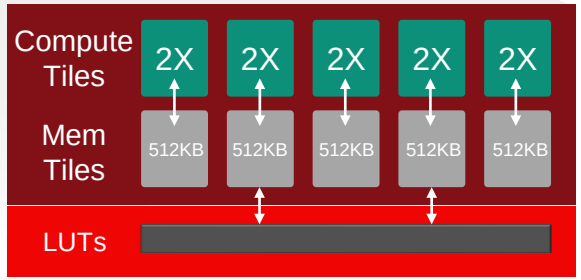
- Optimized for signal processing AND ML
- Flexibility for high performance DSP applications
- Native support for INT8, INT16, FP32

AIE	OPS / Tile	AIE-ML
256	INT4	1024
256	INT8	512
64	INT16	128
16	BFLOAT16	256
16	INT32	
16	FP32	42*

*Via software emulation

KB / Tile		
32	Data Memory	64
16	Program Memory	16

AIE-ML Architecture



- Optimized for ML Inference Applications
- Maximum AI/ML compute with reduced footprint
- Native support for INT4, INT8, INT16, bfloat16
- Fine grained sparsity HW optimization
- Enhanced FFT & complex math support

AMD XILINX

Intelligent Engines Optimized for Any AI Application

Signal Processing

← optimized —

— optimized →

AI Inference

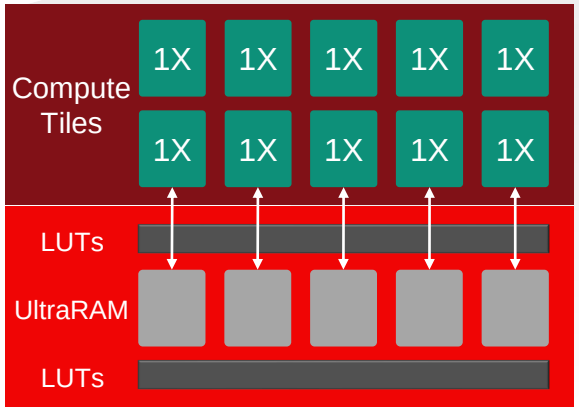
2X INT8/16 OPs/Tile
4X INT4 OPs/Tile

Reduced data movement

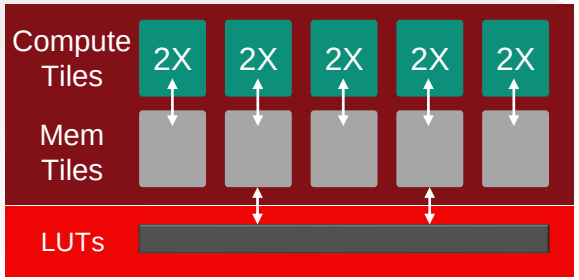
Reduced AI PL Footprint

2X AI Perf/W

AI Engine Architecture



AIE-ML Architecture



KB / Tile	AIE	AIE-ML
Data Memory	32	64
Program Memory	16	16
INT32	16	
FP32	16	42*

*Via software emulation

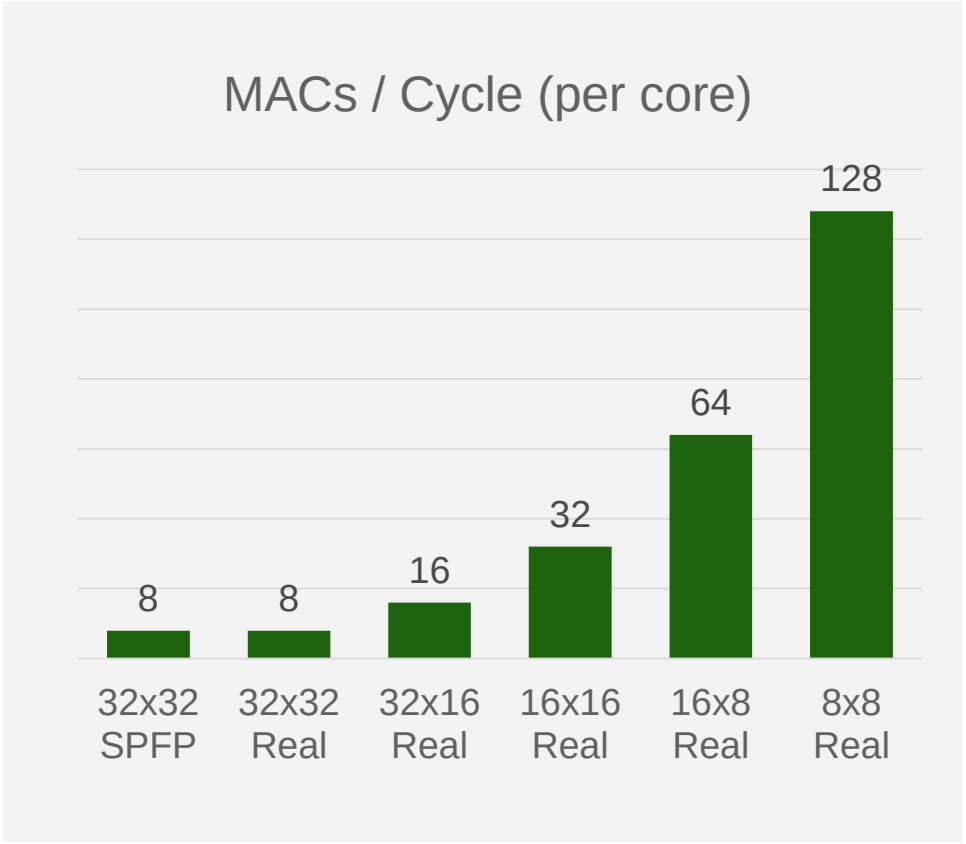
- Optimized for signal processing AND ML
- Flexibility for high performance DSP applications
- Native support for INT8, INT16, FP32

- Optimized for ML Inference Applications
- Maximum AI/ML compute with reduced footprint
- Native support for INT4, INT8, INT16, bfloat16
- Fine grained sparsity HW optimization
- Enhanced FFT & complex math support

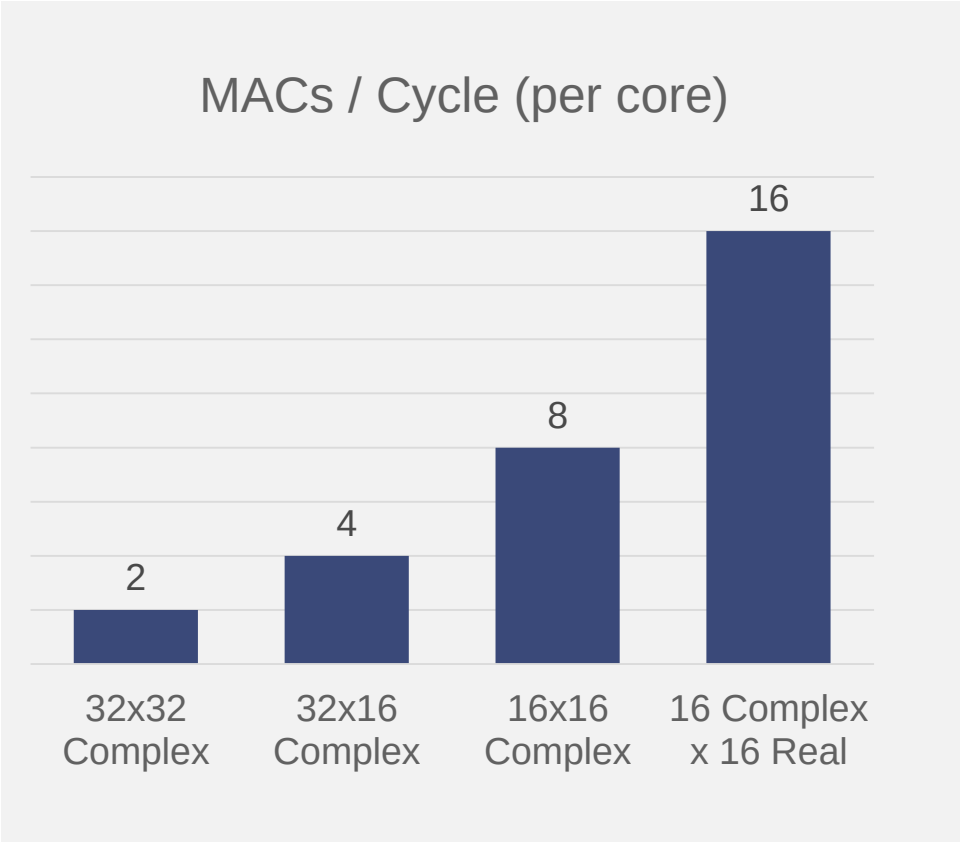
AMD XILINX

Multi-Precision Support (AIE)

AI Data Types



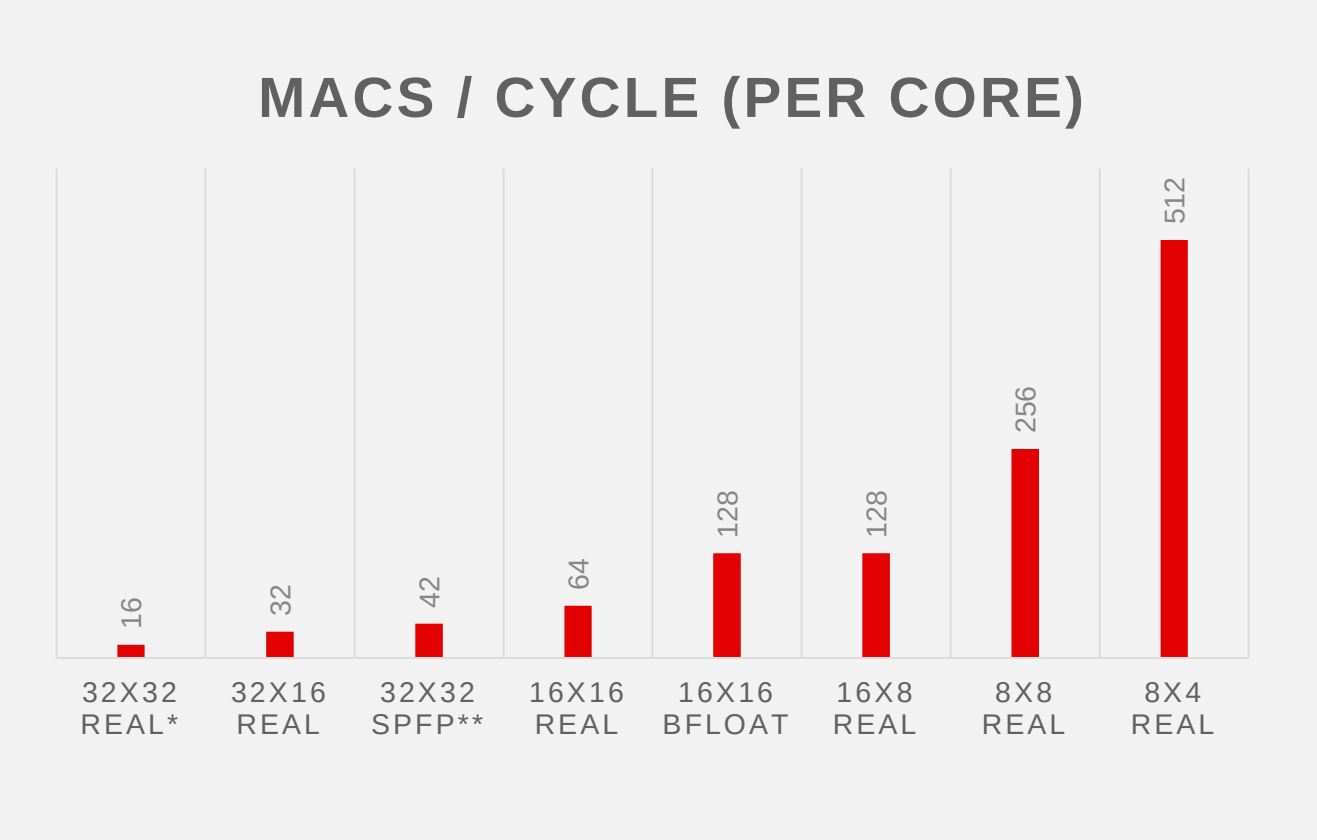
Signal Processing Data Types



AMD XILINX

Multi-Precision Support (AIE-ML)

Real Data Types



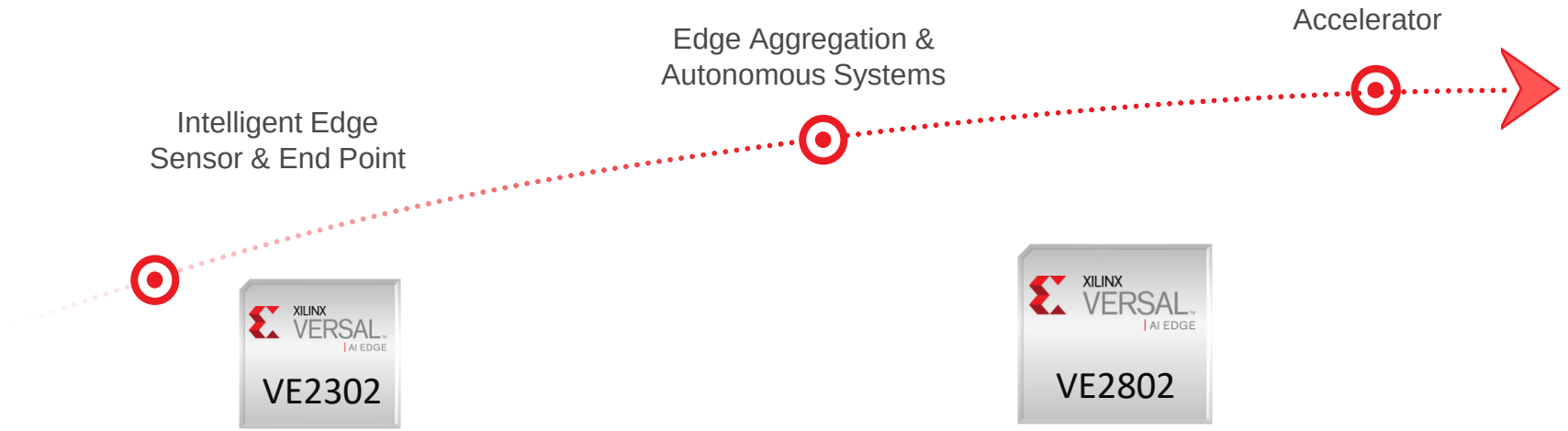
Complex Data Types

*32 Real X 32 Real is emulated

** 32 SPFP x 32 SPFP emulated using BFLOAT16 (following this paper). Not IEEE-754 compliant. Performances TBC

AMD XILINX

AI/MLに向けた幅広い製品群



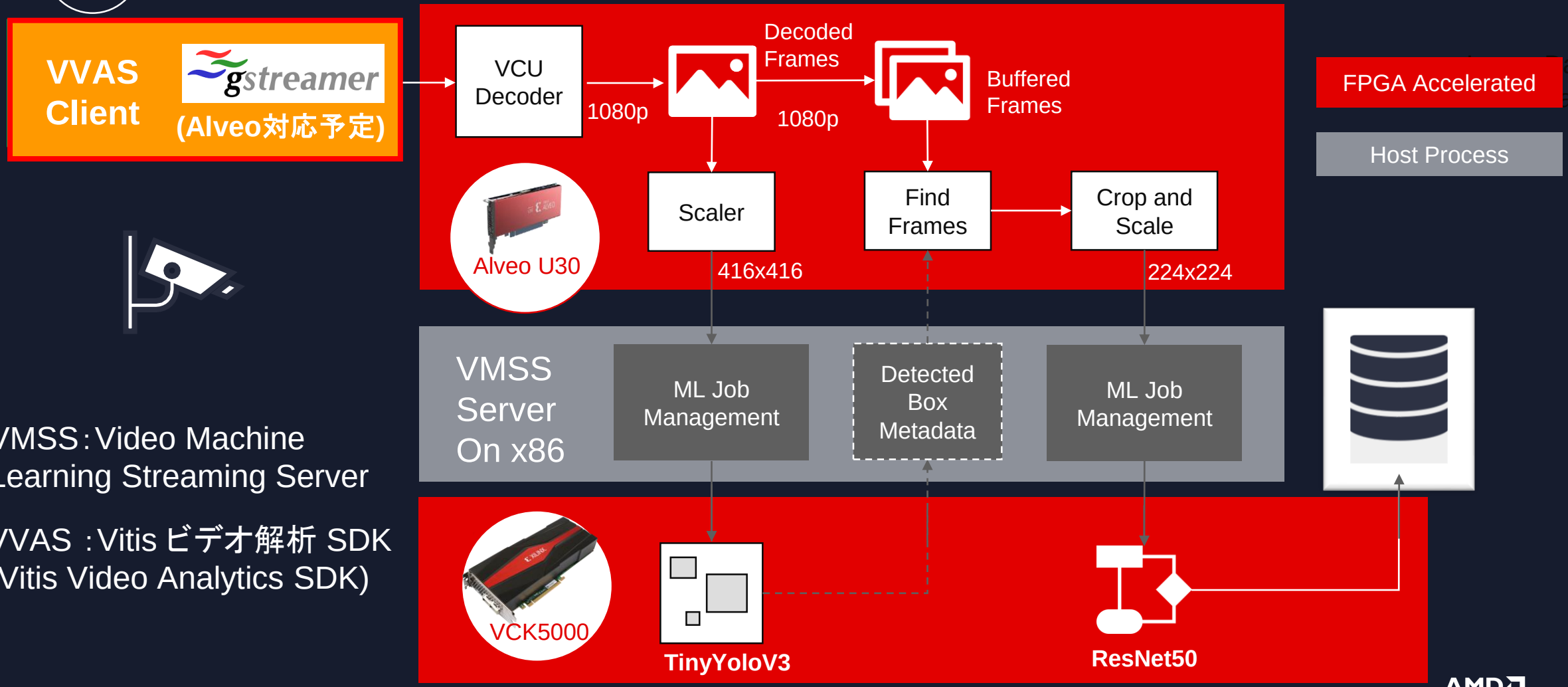
Engines	AI Compute (INT8x4)[1]	**67 TOPS**	**479 TOPS**
	AI Compute (INT8)[1]	**31 TOPS**	**228 TOPS**
	AIE-ML Tiles	34	304
	Adaptable Engines	150K LUTs	521K LUTs
	Processing Subsystem	Dual-Core Arm® Cortex®-A72 Application Processing Unit / Dual-Core Arm Cortex-R5F Real-Time Processing Unit	
RAM	Accelerator RAM (4MB)	✓	-
	Total Memory	172Mb	575Mb
	32G Transceivers	8	32
	PCIe®	✓	✓ (PCIe gen5 w/ DMA)
	Video Decode Unit (VDU)	-	4
	Power [2]	**15-20W**	**75W**

1: Total AI compute includes AI Engines, DSP Engines, and Adaptable Engines
2: Power Projections

AMD
XILINX

フル ソリューションスタック を提供 – 構築のステップ (2)

2 アプリケーション全体の流れをコンフィグレーション



VMSS：Video Machine Learning Streaming Server

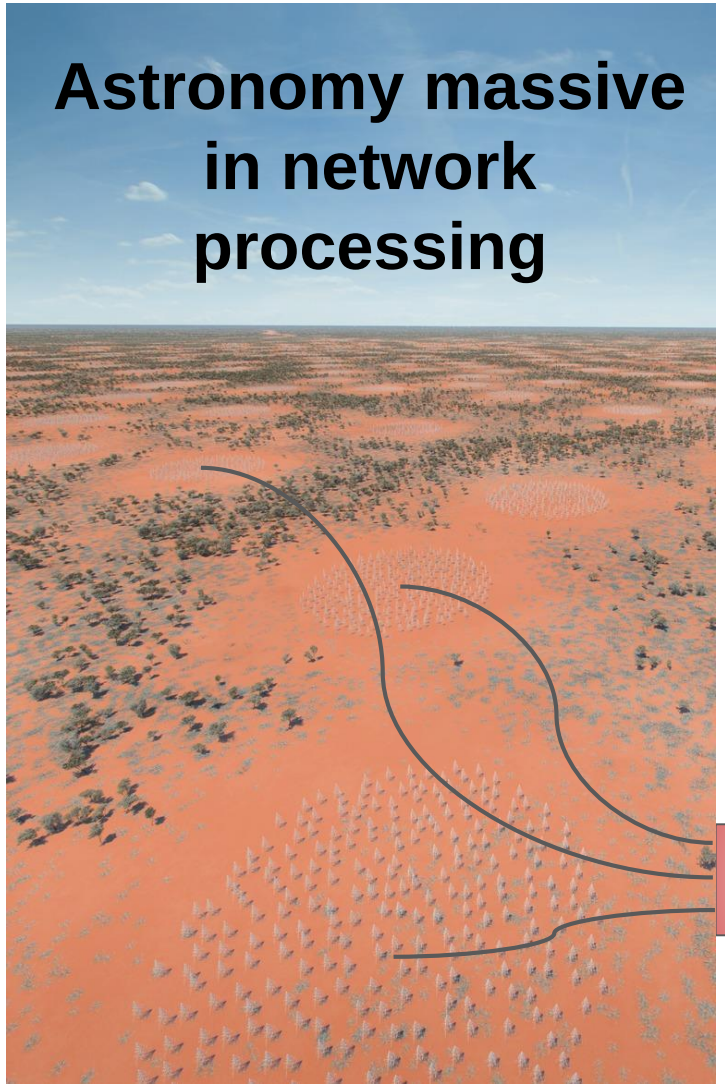
VVAS ：Vitis ビデオ解析 SDK (Vitis Video Analytics SDK)

AMD XILINX

ユースケース事例

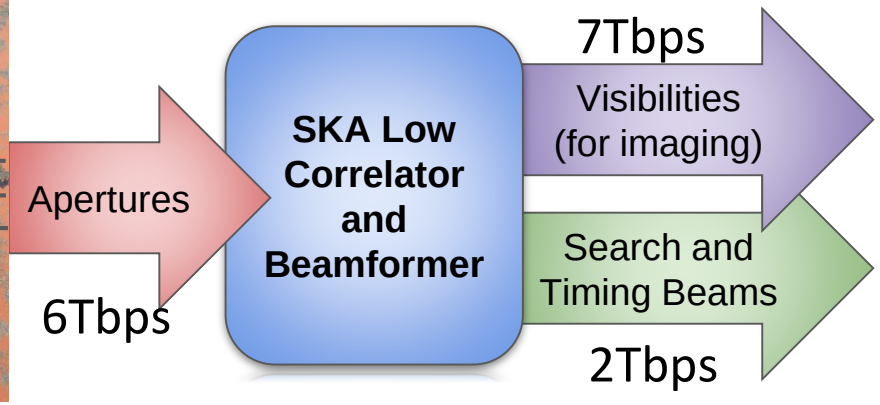
電波望遠鏡 分析システム

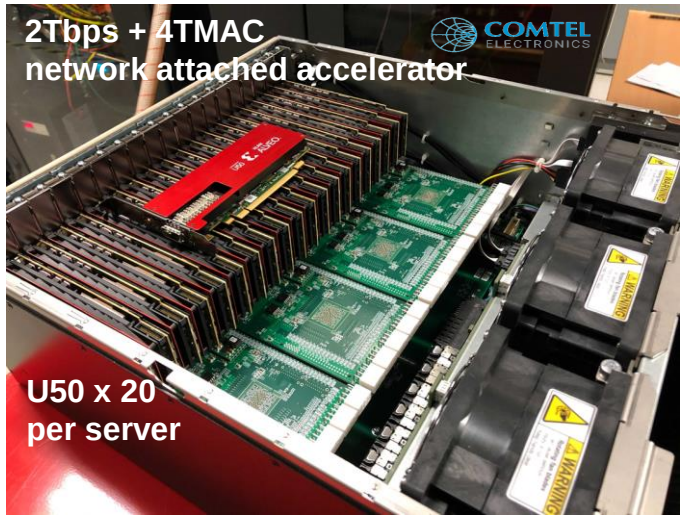
Square Kilometre Array Observatory (SKAO) Telescope



- ~15Tbps の継続的なデータフロー
- ~2 Peta オペレーション/秒
- 膨大なネットワークに接続されたアクセラレータ
  - ネットワークはデータ処理と同じくらい重要
  - 現状はネットワークスイッチの容量を上回る処理能力を有する
- 電力効率の高いコンピュートリソースとしてFPGAを選択
- データ移動の過程でデータをリアルタイムに処理

https://comtel-online.com/





AMD
XILINX

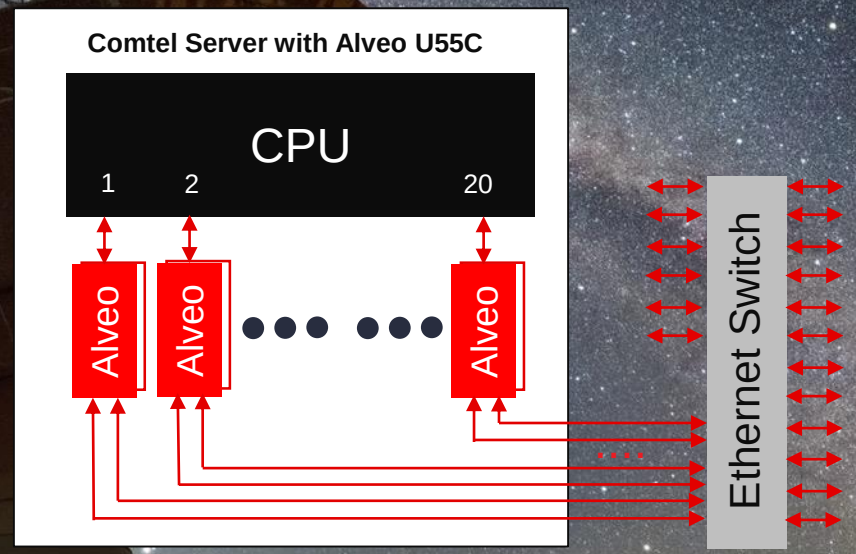
HPC: 信号処理

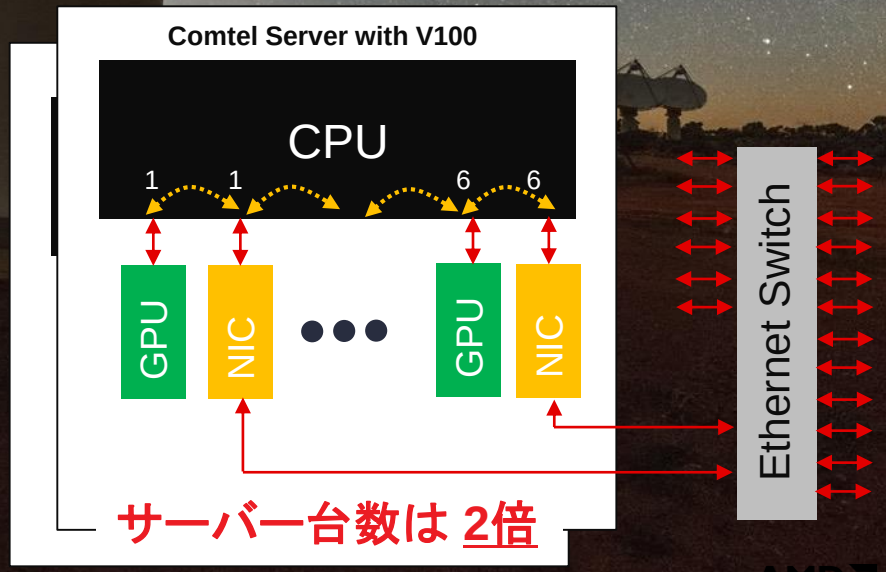
CSIRO

- 世界最大の電波天文用アンテナ アレイ
- 宇宙の起源に関する情報を収めるために設計
- テラビット/秒のセンサーデータをリアルタイム処理
- 420枚の Alveo を使用してリアルタイム分散処理
- リファレンス デザインを作成中

重要な要素

- **大規模なスケールアウト:** 21 ノード、420 カード
- **高性能、低遅延:** 15 テラビット/秒の処理
- **消費電力効率:** ソーラー電源、90W/カード
- **高信頼性**

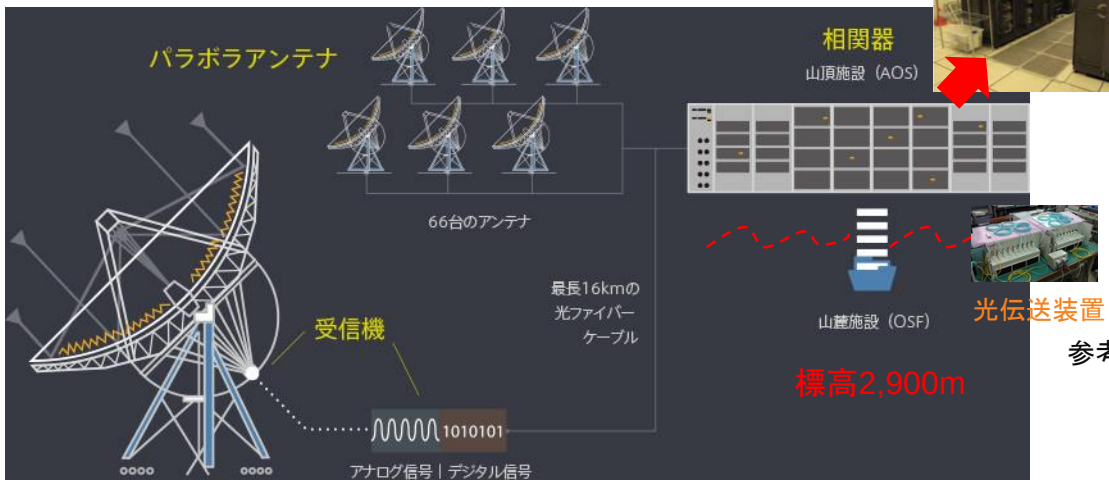




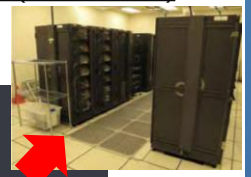
AMD
XILINX

図 1.1：アンテナが立ち並ぶアルマ望遠鏡山頂施設
Credit: Clem & Adri Bacri-Normier（wingsforscience.com）/ESO

参考) ALMA2 Project- アルマ望遠鏡が切り拓く2020年代の科学のフロンティア (国立天文台)



標高5,000m



光伝送装置

参考) アルマ望遠鏡とは https://alma-telescope.jp/about
国立天文台が中心となって開発したACA相関器 https://news.mynavi.jp/techplus/article/alma_project-9/

（Credit: NRAO/AUI/NSF）

ブラックホールの撮影に成功 (2019年4月10日)

参考) M87ブラックホール撮影への道のり https://www.elecs.co.jp/news/blackhole.html
史上初、ブラックホールの撮影に成功 https://www.nao.ac.jp/news/science/2019/20190410-eht.html
https://iopscience.iop.org/article/10.3847/2041-8213/ab0c96
https://github.com/casper-astro/casper-hardware#casper-hardware

AMD XILINX

YADDLE-MD

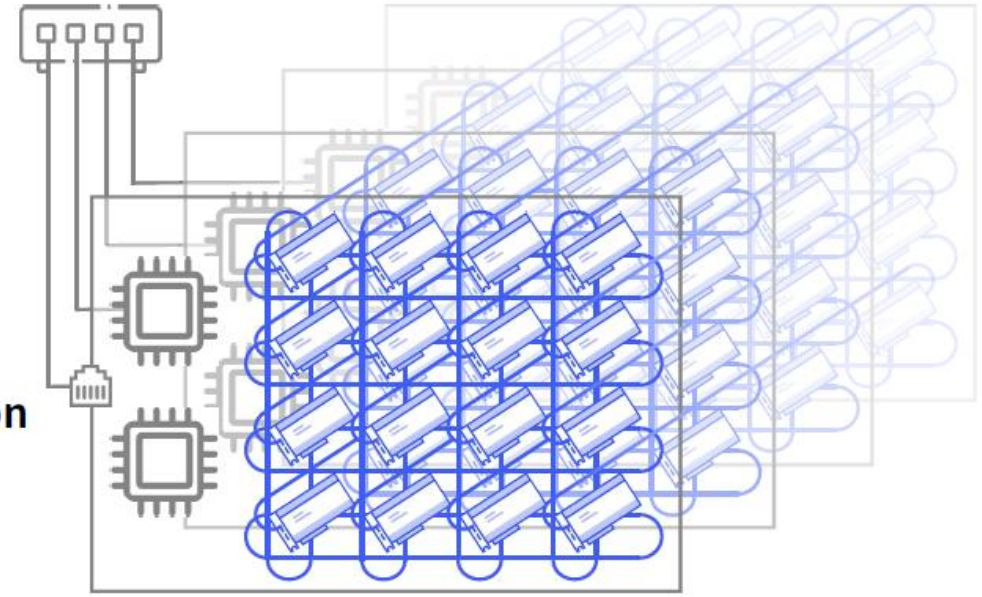
FPGA-based Molecular Dynamics Engine

YADDLE-MD is Snowlake's groundbreaking molecular dynamics engine. Depolyed on Xilinx's powerful FPGA-based Alveo accelerator cards, the YADDLE-MD delivers extraordinary performance unmatched by competing architectures. Moreover, YADDLE-MD has outstanding efficiency and scalability.

- **10 Gbps Ethernet** between nodes

On CPU side

- **No MD computing and communication**
- Dispatch
- Monitoring
- Collection



16 Alveo cards per node

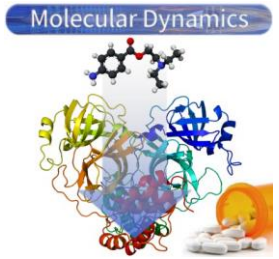
- **200 Gbps 3D-Torus** cross-node interconnection between cards

On FPGA side

- **Full MD computing and communication**
- Pair search
- Bonded/Nonbonded
- Long-range electrostatics
- Integration
- Constraints
- Communication

雪湖科技 SNOWLAKE http://www.snowlake-tech.com/

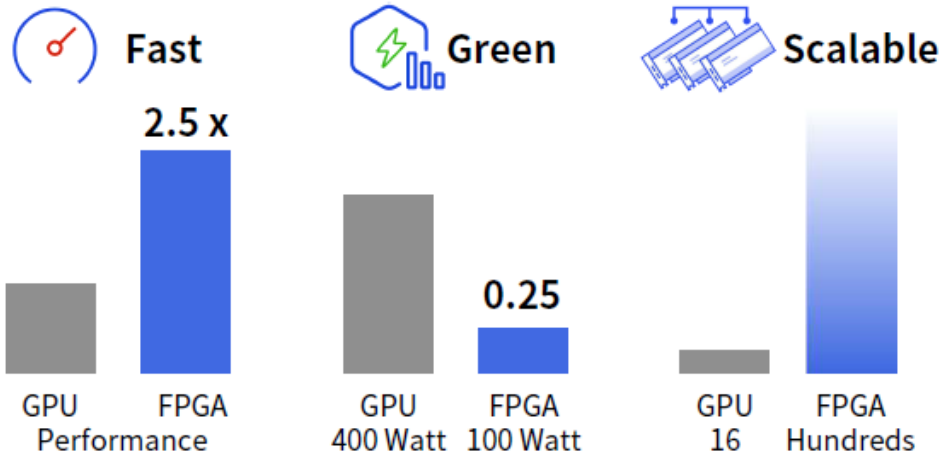
YADDLE-MD
FPGA-based Molecular Dynamics Engine



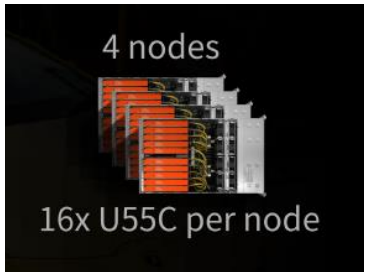
Molecular dynamics modeling is used in drug research and discovery

雪湖科技
SNOWLAKE

HIGHLIGHTS



When clustered



http://www.ks.uiuc.edu/Research/namd/benchmarks/

AMD
XILINX

Cascaded FPGA Accelerator with PTU
(Protocol Termination Unit ; TCP/IP Offload Engine)

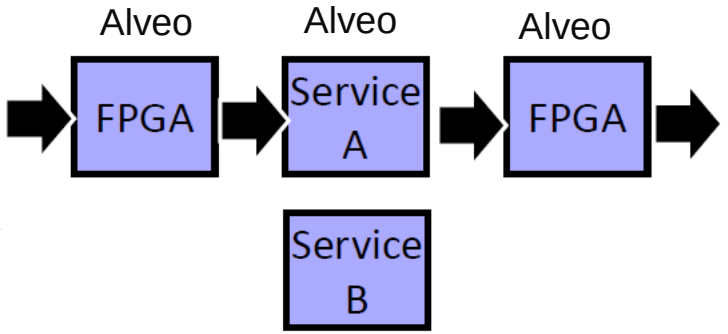
Intellectual Highway
(Intellectual Highway合同会社)
https://www.i-highway.com/

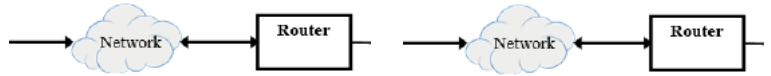
FPGA 処理の高速なディスアグリゲーションを容易に実現する技術

Multiple FPGAs are connected via Ethernet (25G/100G) without any interaction with its host PCs.
=> **CPUに負荷無くノード間で高速なネットワーク伝送**

Whole HW Implemented TCP/IP stack with security features
=> **安定した低レイテンシーの実現 かつ パケット損失なし**

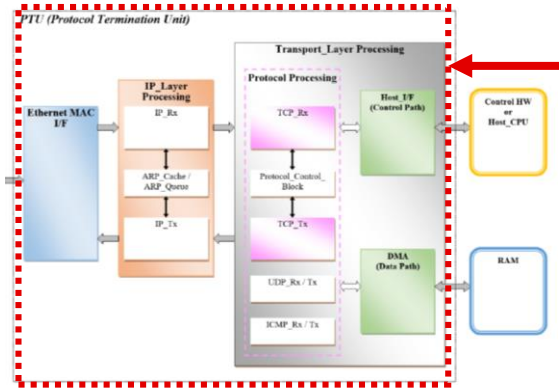
Multiple TCP Sessions (over 10,000) which allows each Data Flow to be conveyed in individual TCP Session
=> **高度なスケーラビリティ**



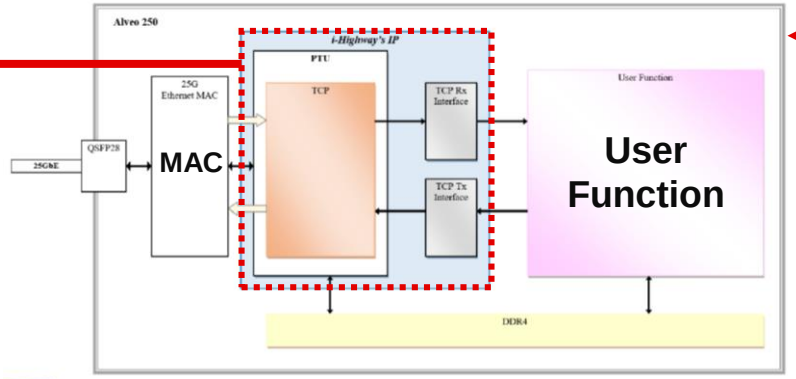


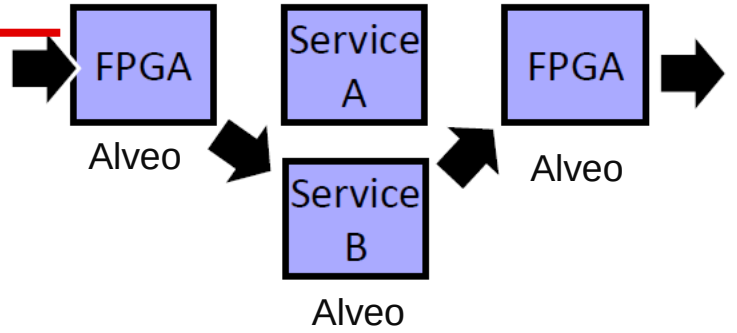
Dynamic Service Switch

PTU



Alveo U250



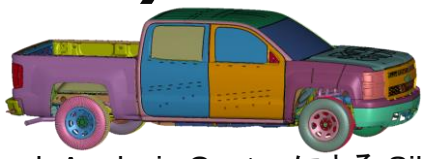


AMD XILINX

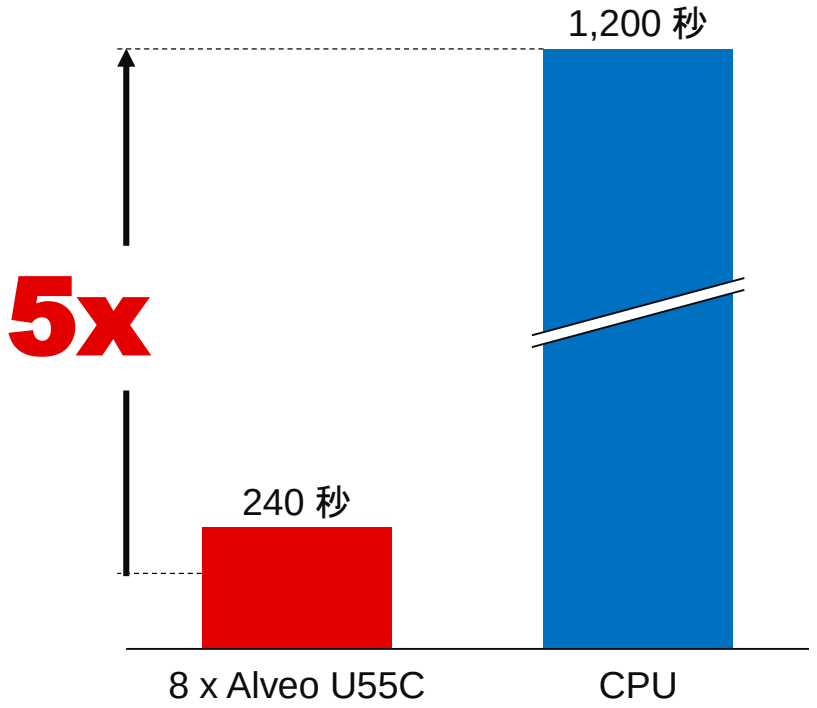
HPC: コンピューター支援エンジニアリング (CAE)
Ansys LS-DYNA

- LS-DYNA: 有限要素プログラム (FEM)
  - FEM を使用して現実世界の製品性能をシミュレーション
  - 無限の複雑性をもつシミュレーションを作成
- 大規模シミュレーションは CPU では数週間必要

  フォンノイマン アーキテクチャの限界
  - データ幅が固定 => データアクセス効率の低下、データ待ち時間はクロック、コンピュートリソースを浪費
  - オンチップ キャッシュのサイズは有限 => 大きなデータは外部のメモリアクセスが必要 = 性能劣化、消費電力増大
- **Alveo U55C はフォンノイマンの限界を超える**
  - データは機能ブロック間をパイプライン状にストリーミング
  - データーバス幅、データムーバ、メモリ階層構造をカスタマイズし最適なデータパイプラインを構築
  - 16GB HBM2 メモリ (32 HBM channels@ 460GB/s) をシングルデバイスに内蔵 = 大きなデータも外部メモリアクセスは不要
  - LS-DYNAのユースケースではCPUの**5倍**の性能を実現



- National Crash Analysis Center による Silverado モデル
- **700k の要素** (大部分はシェル + ソリッドとビーム)
- 膨大なコンピュート負荷



行列の**大きさ** -> 12M
nnzs: 非ゼロ要素の数 -900M
時間 (秒): JPCG ソルバーの実行時間
CPU モデル: インテル Xeon Platinum 8260L @2.4GHz、1.5TB メモリ

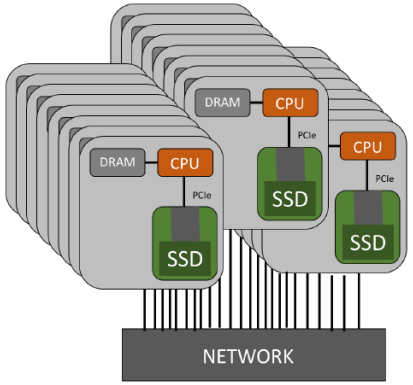
AMD
XILINX

LEWIS RHODES LABS –NPUsearch™ Integrated Search-in-Storage on Samsung SmartSSD® CSDs powered by Xilinx FPGAs

Neuromorphic Processing Unit (NPU) is the core of NPUsearch. Designed with the fine grain parallelism, hierarchical structure, it is highly efficient pattern matcher to accurately and rapidly scan data. Full content search of unindexed data is regex accessible via JupyterNotebook or other Python-based interface.

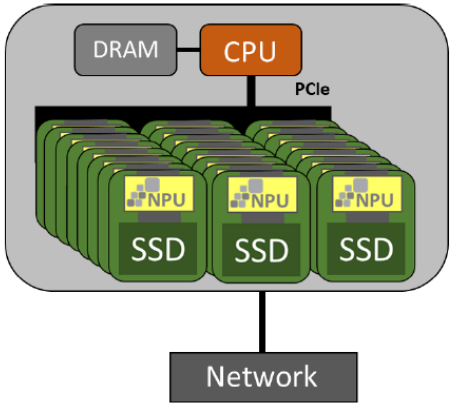
96TB NPU*search* storage appliance

CPU based architecture

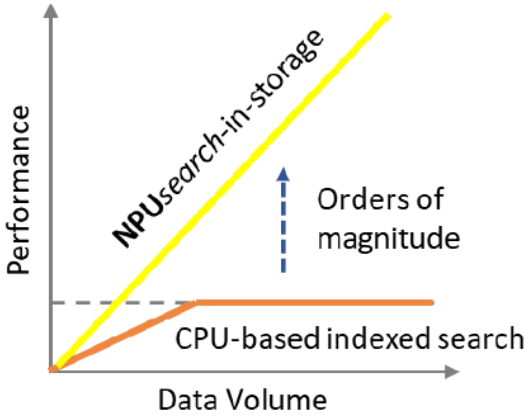


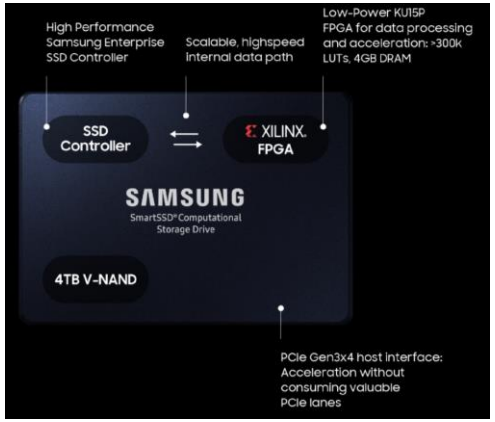
> Inconsistent search capacity
> CPU-intensive cost profile
> Move all data to CPU to search
> Heavy network requirements
> High volume of data flow
> Unable to scale performance

NPU search-in-storage architecture



> Rapid, deterministic search
> Reduce CPU costs
> Move only data of interest
> Light network demands
> Minimize data flow
> Scalable search capacity
> Lower power consumption



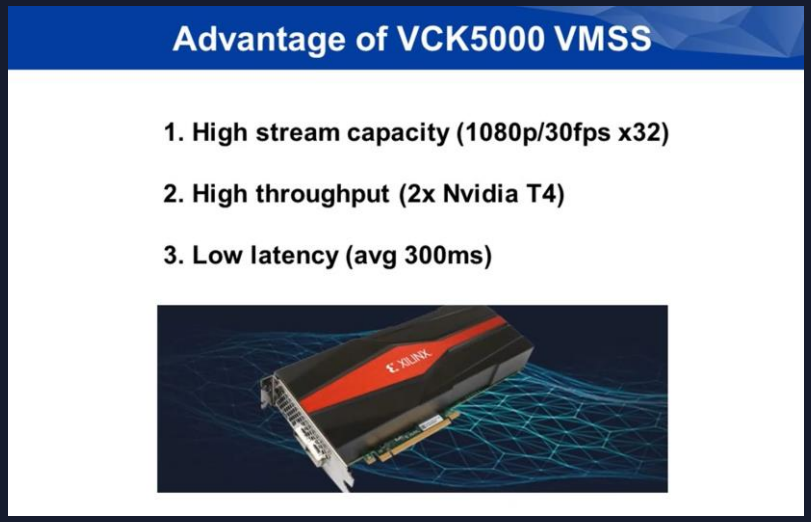


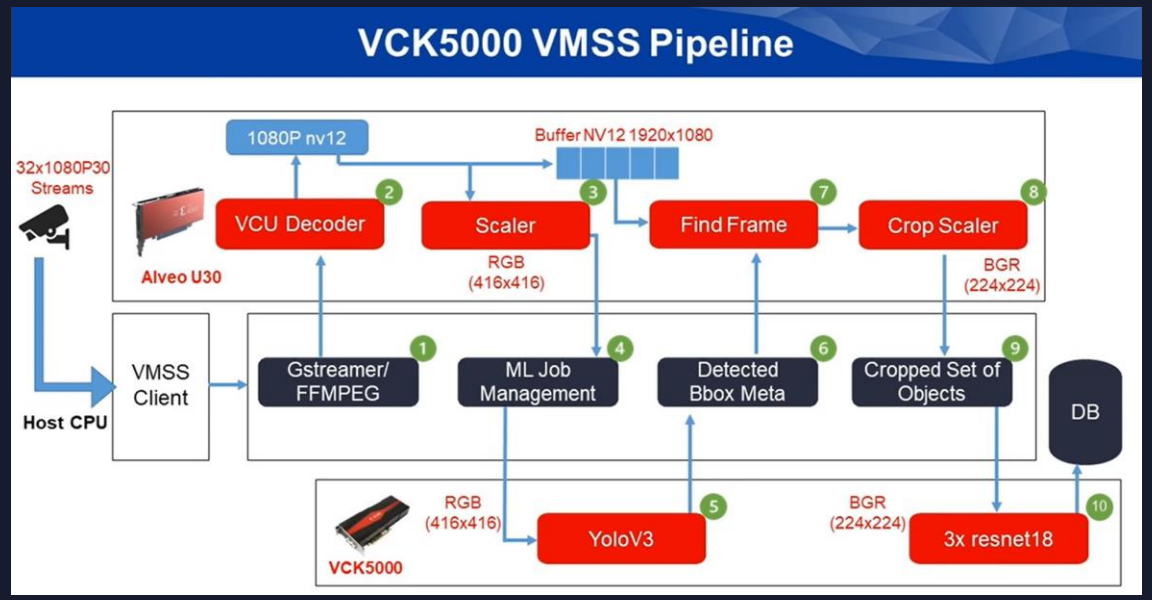
AMD XILINX

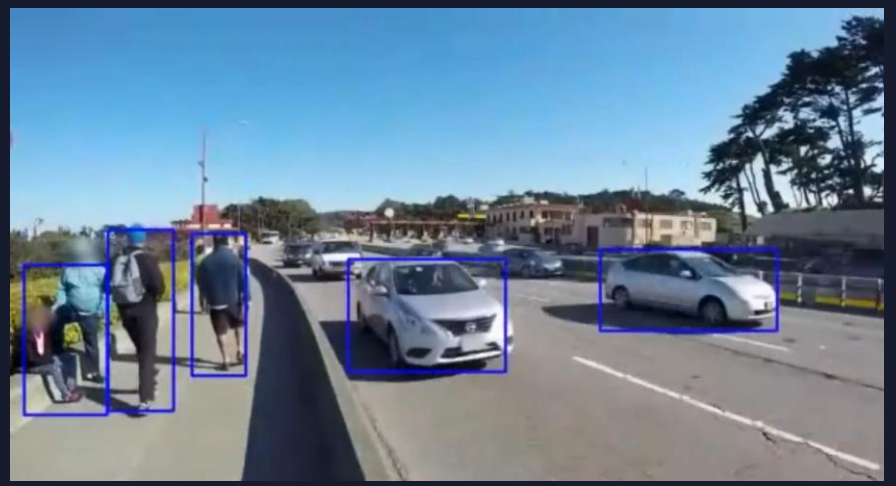
Aupera社 インテリジェント ビデオ解析ソリューション

https://japan.xilinx.com/products/boards-and-kits/vck5000.html

Aupera VMSS (Video Machine Learning Streaming Server) ソリューションは複数のフル HD カメラからの高密度映像ソースをサポートし、オブジェクトの識別と分類を実行します。
複数の推論モデルを同時に実行可能で、確定的かつ低レイテンシで精度の高い結果を出力します。業界最小のコスト (TCO) を実現できることが特徴です。





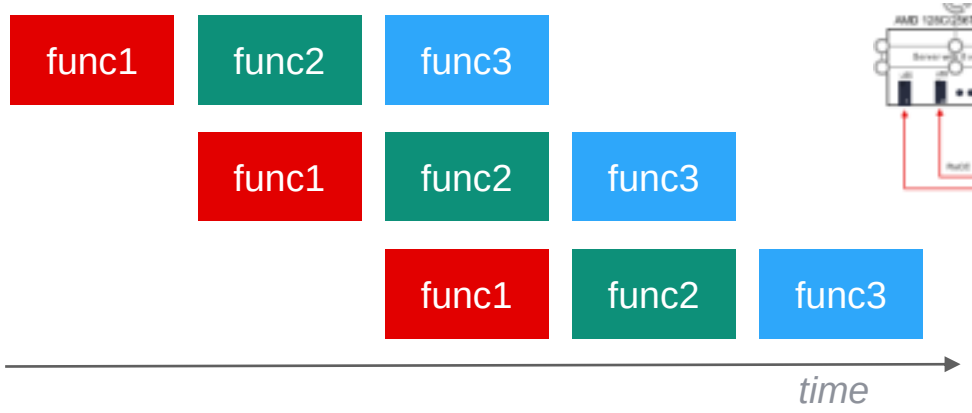


AMD XILINX

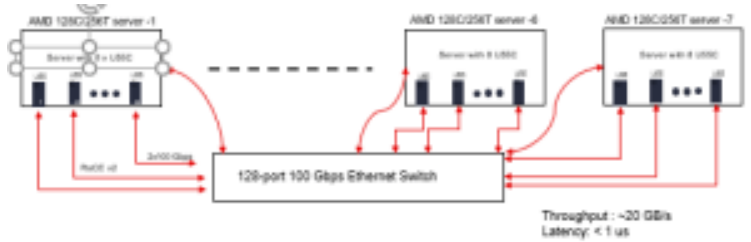
まとめ

AMD Alveo Parallelism

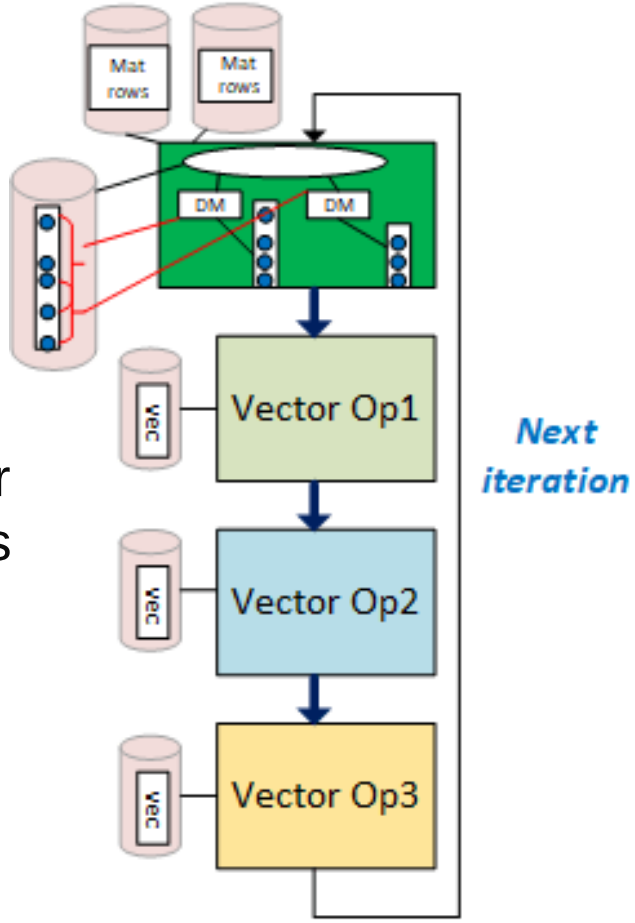
Function level Pipelining



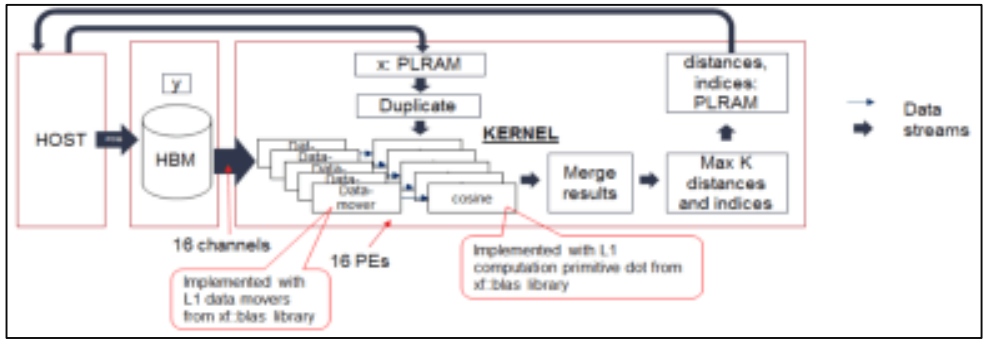
Natively networkable for scale-out



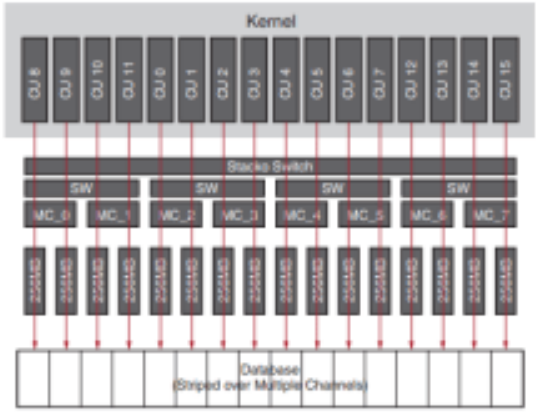
Custom Data Width and Movement



Large number of Compute elements and Custom super efficient circuits



32 memory channels for parallel stall-free access



AMD XILINX

まとめ

- FPGAとそのクラスタによる効果といくつかのユースケースについて触れた
- 以下の特徴を生かすことでパフォーマンス、消費電力、コストに有効
  - Memory vs Cache
  - Heterogeneous device architecture (Versal AIE/AIE-ML)
  - Flexible Data width/type, data type optimization, data movement optimization
  - Pipeline
  - Massive parallelism
  - Direct network
  - Compute near data
- 設計の容易さを推進
  - Alveo, Versal, Vitis, HLS, Silexica, ライブラリ, Model Composer, ACCL (MPI-Like), ...

Thank You

Disclaimer and Attribution

AMD
XILINX