# Pervasive and Sustainable AI with Adaptive Computing Architectures

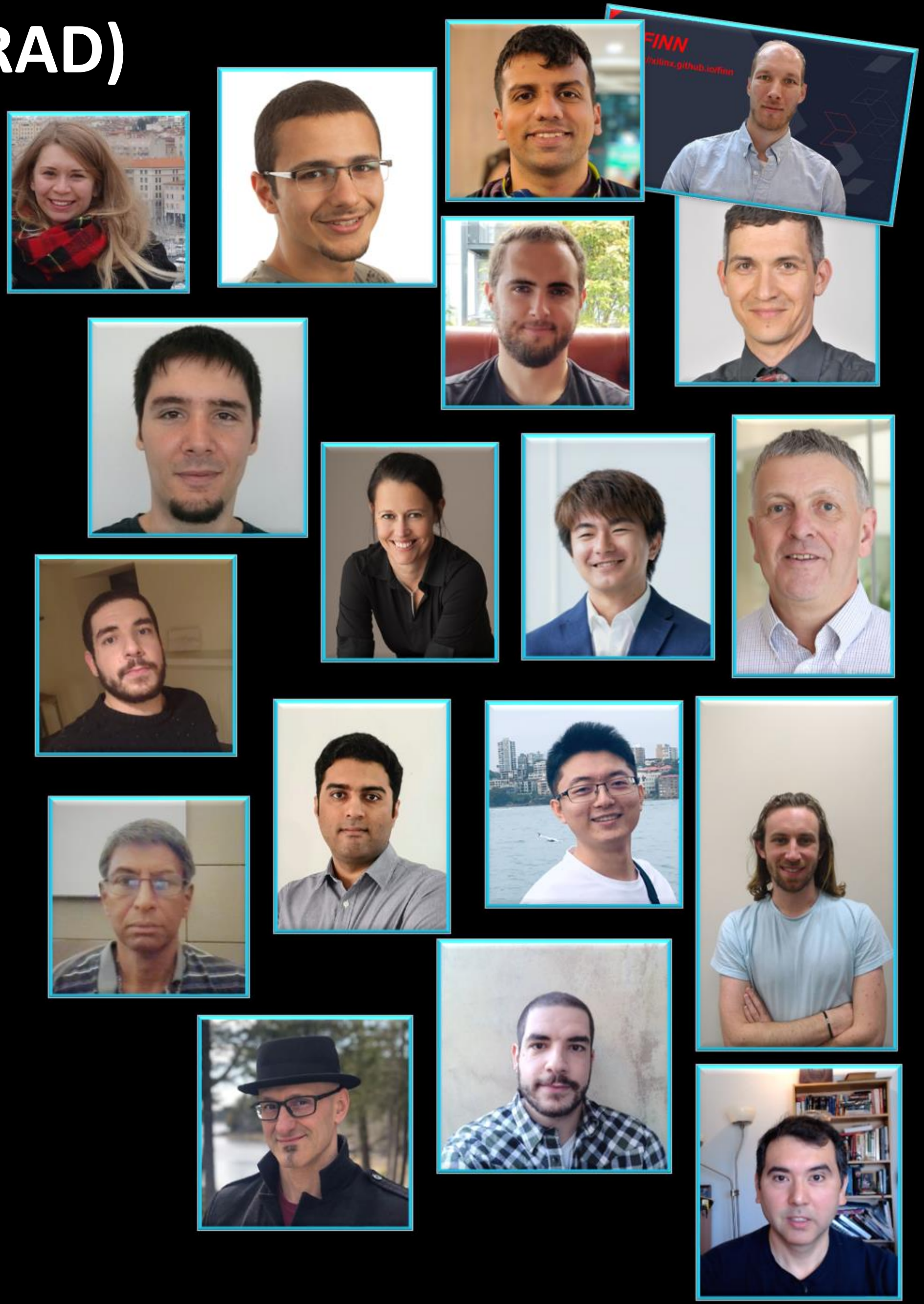
**Michaela Blott**
Senior Fellow
AMD Research & Advanced Development

AMD
together we advance_

# AMD Research and Advanced Development (RAD)

- **Integrated Comms and AI Lab (RADICAL)**
  - Established 18 years ago
  - ~20 researchers plus university program
    - 5 different locations

- **Focus: AI and Communications**
  - Building systems, architectural exploration, algorithmic optimizations, benchmarking
  - In collaboration with partners, customers, and universities
    - ETH Zuerich, Paderborn University, Imperial College, KIT, NTNU, Politecnico di Milano, NUS, University of Sydney



# Active Internship Program

- On average 8-10 interns at any given time
  - From top universities all over the world

- Overall
  - 100+ interns since 2007
  - Many collaborations have come from this
  - Recognize anyone?
  - Many found employment

NTNU
Norwegian University of Science and Technology

KIT
Karlsruher Institut für Technologie

POLITECNICO
MILANO 1863

UNIVERSITY OF CAMBRIDGE

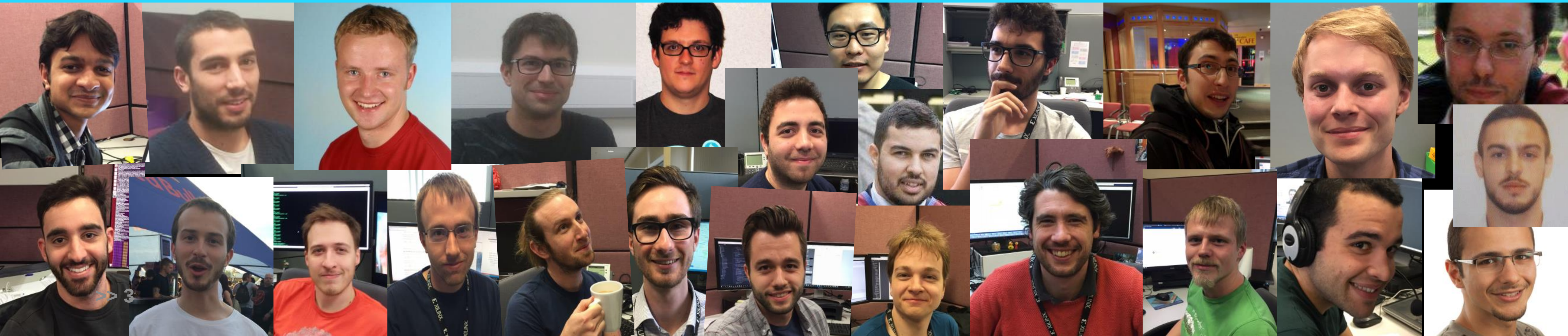
Technische Universität Kaiserslautern

ETH Zürich

TECHNISCHE UNIVERSITÄT DRESDEN

THE UNIVERSITY OF SYDNEY

Imperial College London



# CONTEXT

# DNNs and their Potential

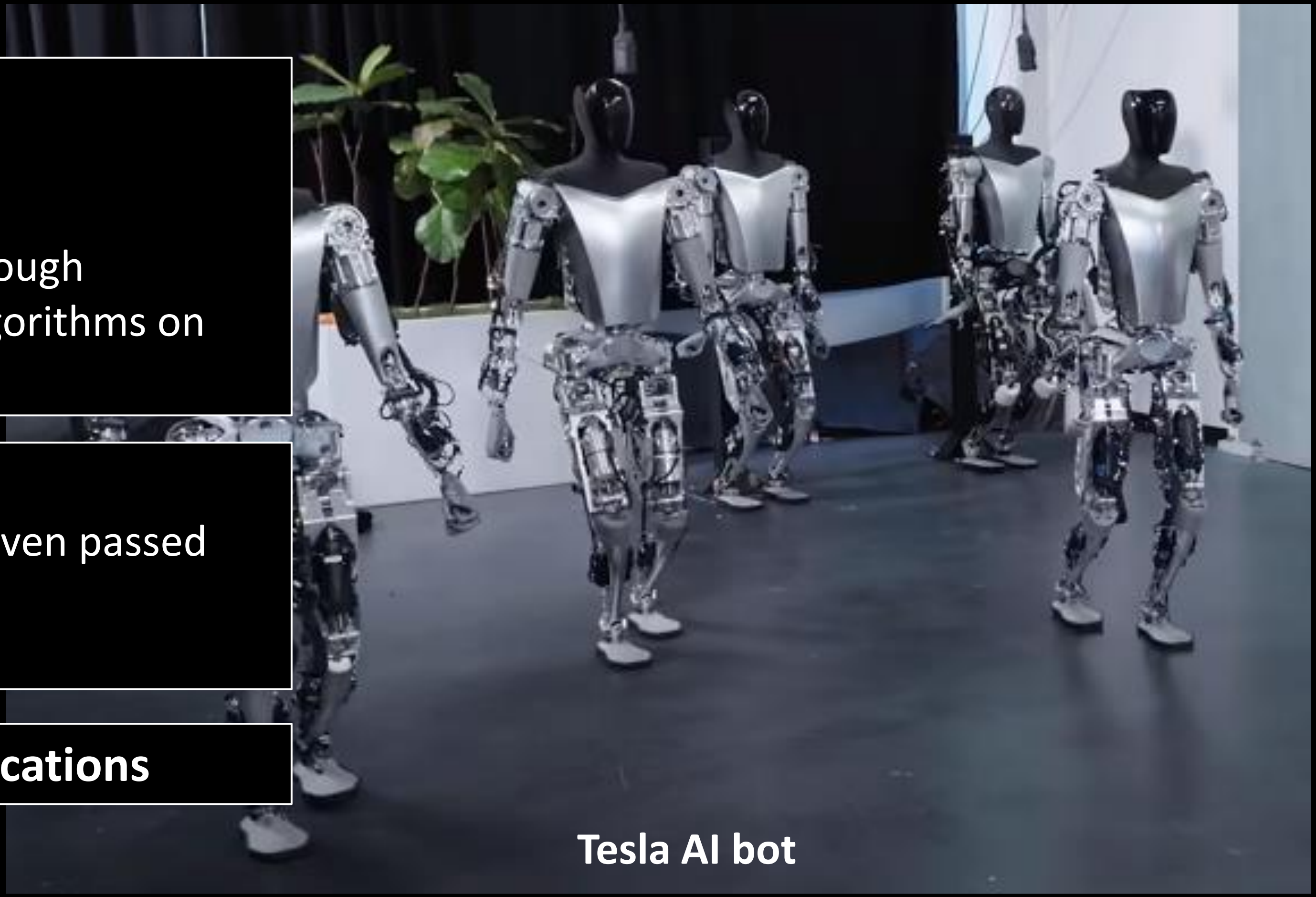
**Huge potential**

- Requires little domain expertise
- NNs are a “universal approximation function”
- If you make it big enough and train it long enough
  - Can outperform humans and existing algorithms on specific tasks

**Solves previously unsolved problems**

- Code, text and image generation, and GPT-4 even passed the bar exam in the 90th percentile
- Protein folding

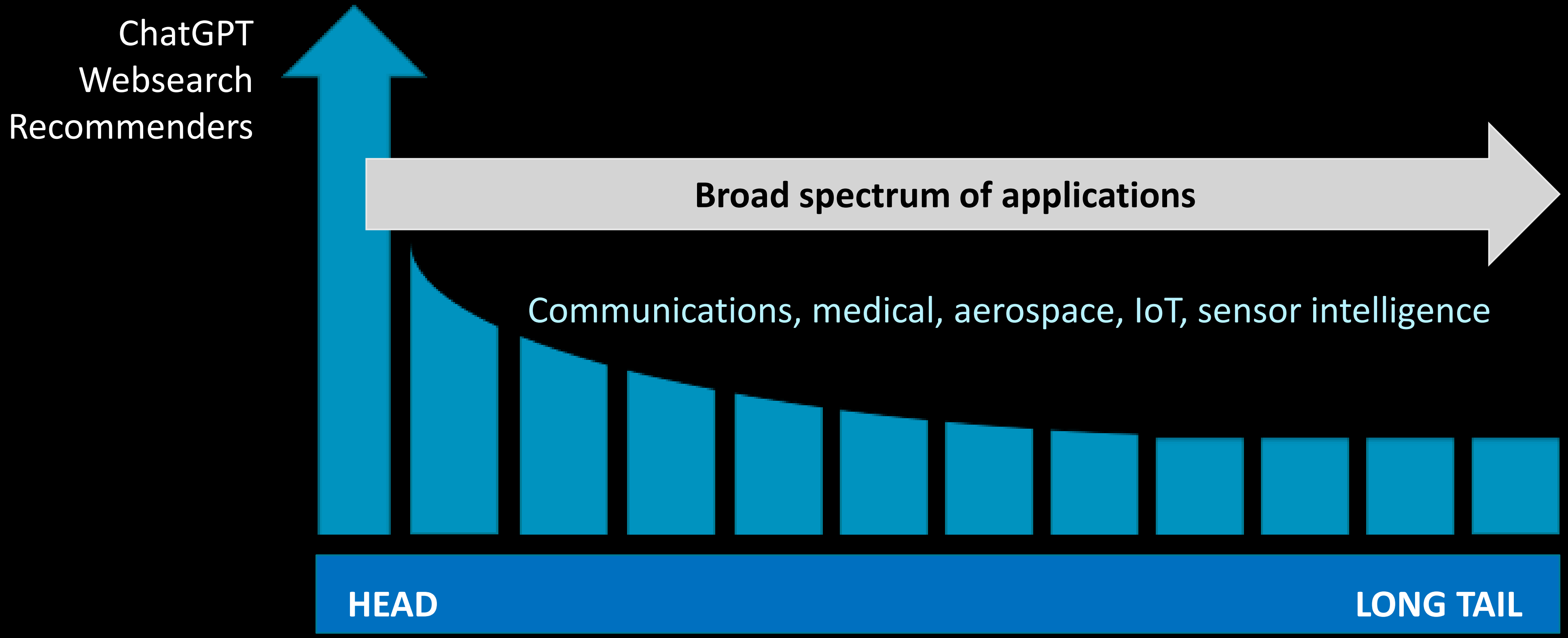
**Increasing adoption in many different applications**



Tesla AI bot

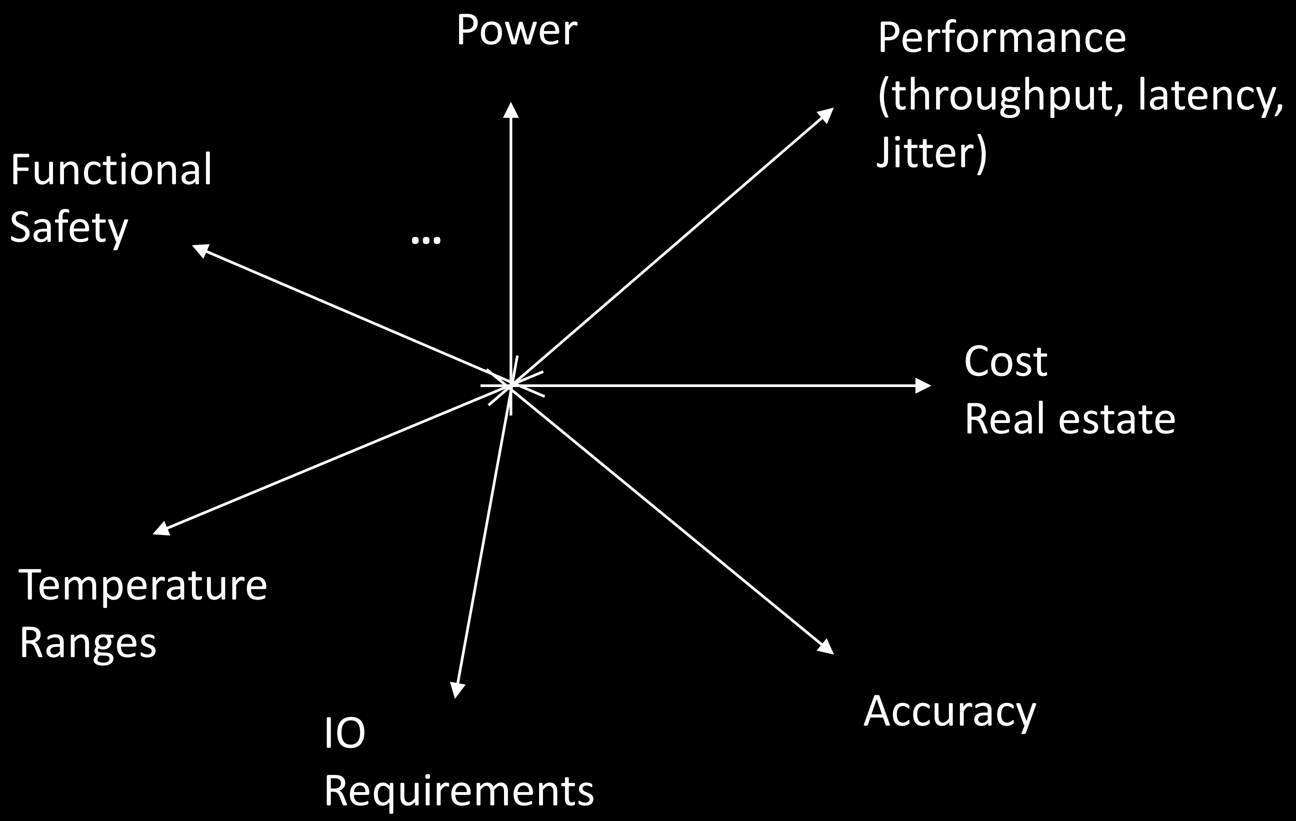
https://youtu.be/XiQkeWOFwmk?t=6

# Pervasive AI


ChatGPT
Websearch
Recommenders
Broad spectrum of applications
Communications, medical, aerospace, IoT, sensor intelligence
HEAD
LONG TAIL

Adapted from TED Talk: Andrew Ng "How AI could empower any business"

# Pervasive AI Comes with Diverse Requirements



# Examples of Diverse Requirements

- **IoT/Embedded**
  - Small resource footprint, low power (<10W), low latency (msec) and 0 jitter
  - Dynamic input sizes

- **High Frequency Trading**
  - High frequency trading (HFT) is an arms race of acquiring data and executing trading decisions fastest
  - Multi-million dollar advantages through nanosecond differences
  - Extreme low latency requirements (nsec) as DNNs are being adopted for better trading decisions

- **High Energy Particle Physics**
  - CERN CMS Experiment needs nsec latency for setting recording trigger
  - Incoming data needs to be processed at 7Tbps
  - Extreme latency requirements (nsec)

# Examples of Diverse Requirements

- **ML in Communications Requirements**
  - No run-time, streaming integration
  - Extreme throughput (100s Minferences/sec)
    - Line-rate processing for n*100G Ethernet
  - Low latency (<msec) for ML-based firewalls
    - Rule-based approaches are being replaced or augment with DNN-based classifiers
    - Protect proactively against new forms of attack and reduce zero-day vulnerabilities
  - Custom datatypes, tensor dimensions and kernel shapes
  - Combine DNNs with signal processing*

- DNNs will increasingly penetrate both wireless and wired telecommunications (monitoring, prediction, optimizing, learned physical interfaces,…)
- New range of requirements

*Korpi, Dani, et al. "DeepRx MIMO: Convolutional MIMO detection with learned multiplicative transformations." *ICC 2021*

# Dynamic Workloads

## *AI is a highly active research area*

- Algorithms are still changing, science is not mature yet
  - Next datatype? FP32->INT8->BF16  -> FP8 => Logarithmic?
  - Next operator that changes the compute paradigm? Transformers have arrived in 2017 and are now everywhere->?
  - Next generative paradigm? VAE – >  GAN –> Denoising Diffusion

- Fundamentally disruptive ideas
  - Hinton's NeurIPS 2022 keynote speech on Forward-Forward learning – backpropagation not be needed in the future?*

- Customer workloads are changing during the development cycle
  - Models are in flux (optimization)
  - First 3GPP 6G specification expected in 2028

Discover neural connectivity

*https://syncedreview.com/2022/12/08/geoffrey-hintons-forward-forward-algorithm-charts-a-new-path-for-neural-networks/

**Audibert, Rafael & Lemos, Henrique & Avelar, Pedro & Tavares, Anderson & Lamb, Luís. (2022). On the Evolution of A.I. and Machine Learning: Towards Measuring and Understanding Impact, Influence, and Leadership at Premier A.I. Conferences. 10.48550/arXiv.2205.13131.

# Sustainability & Energy Consumption

- Energy footprint on par with whole industrial nations

ChatGPT
4.3 GWh*

Meta AI cluster
53-561 TWh*

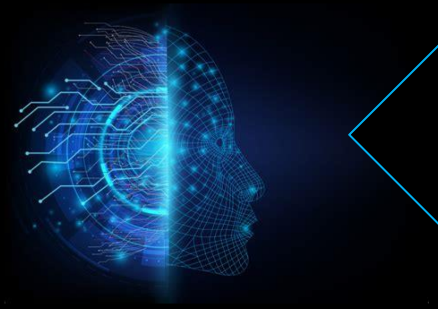
=

Ireland
26 TWh*

Germany
537 TWh*

- Current DNN algorithms represent a **sledgehammer approach**
  - Extremely inefficient

**100s kilo Watts**
**matrix multiply**



**Scope for Improvement:**
**Estimated 10^5**

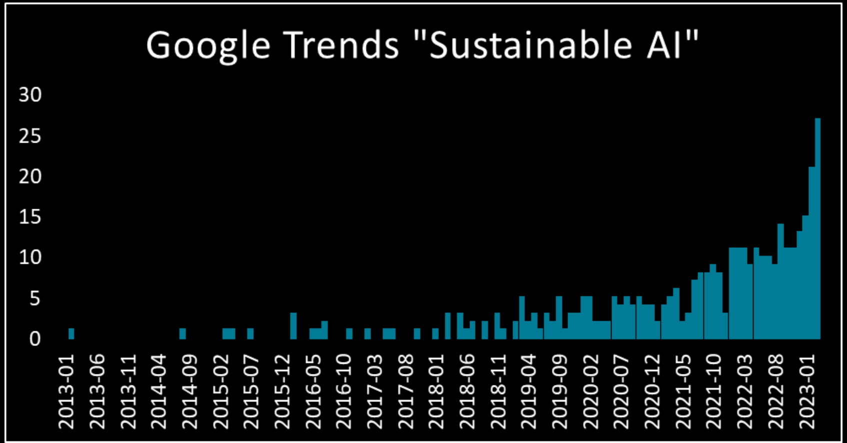
**20Watts**

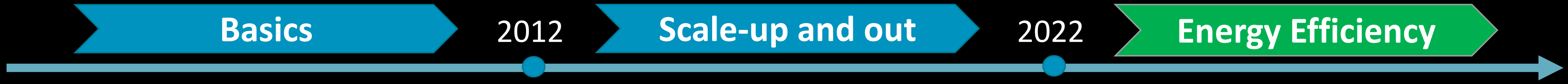
The carbon footprint of ChatGPT. An estimate of the carbon emissions… | by Chris Pointon | Dec, 2022 | Medium
https://www.semianalysis.com/p/meta-discusses-ai-hardware-and-co
Germany - Energy consumption in Germany (worlddata.info)
Ireland - Energy consumption in Ireland (worlddata.info)
**Yu Wang, Tsinghua University, Feb 2016 https://www.numenta.com/blog/2022/05/24/ai-is-harming-our-planet/

*TWh = Tera Watt hours

# The Paradigm Will Shift towards Energy Efficient AI

- Energy will become the limiting factor to scaling NNs




Basics

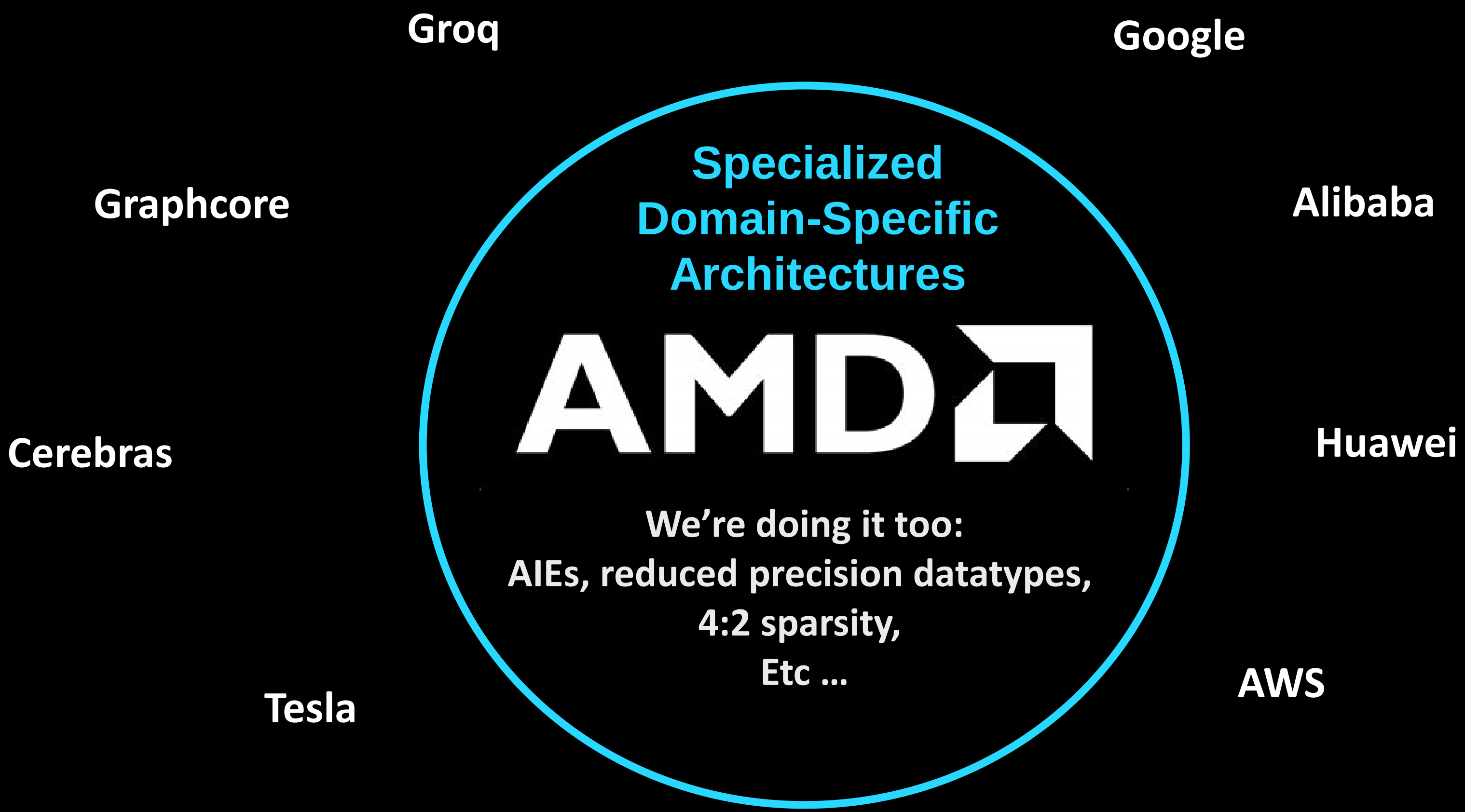
2012

Scale-up and out

2022

Energy Efficiency

# Specialization Is #1 Industry Approach to Energy Efficiency



Groq

Google

Graphcore

Alibaba

Cerebras

Huawei

Tesla

AWS

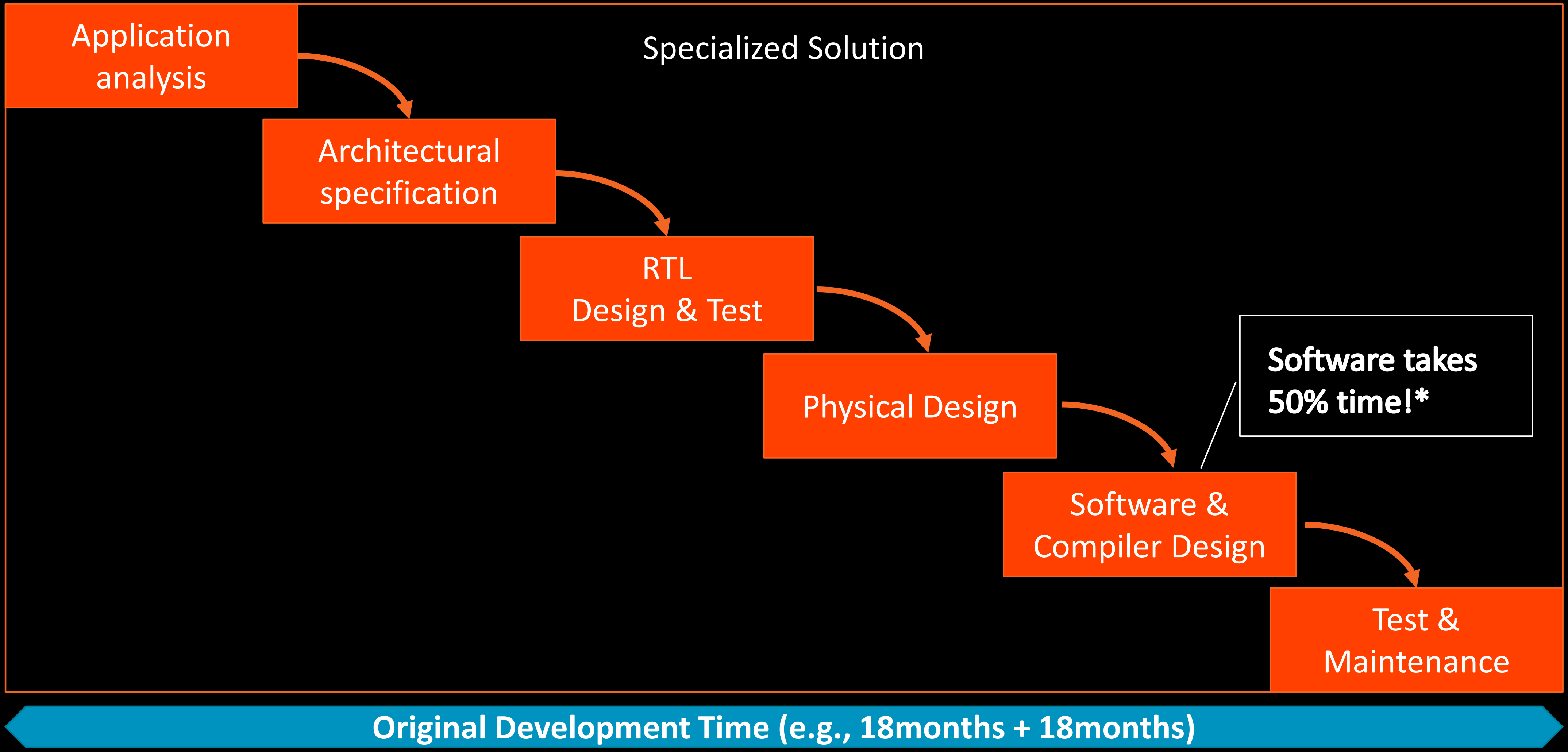
Specialized Domain-Specific Architectures

AMD

We're doing it too:
AIEs, reduced precision datatypes,
4:2 sparsity,
Etc …

# Solution Specialization

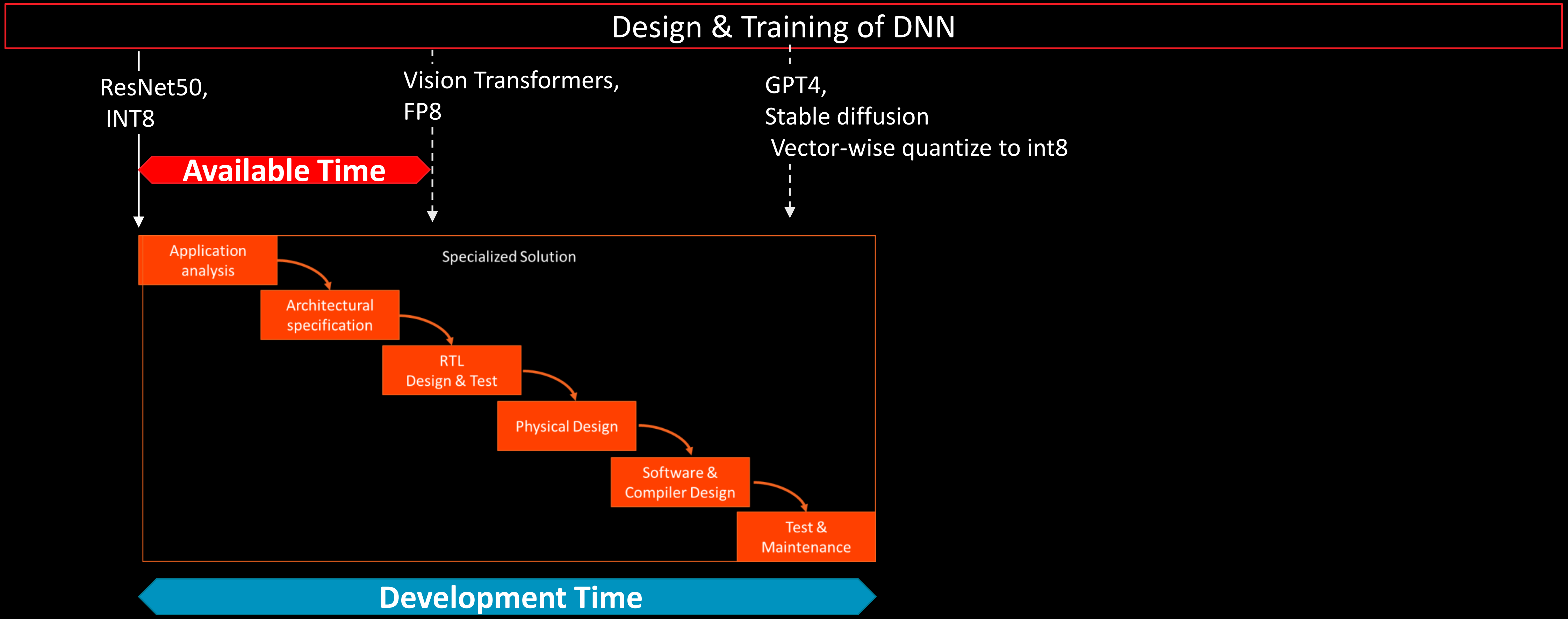
## *Classical Hardware Accelerator Design Process (Waterfall)*


Specialized Solution
Application analysis
Architectural specification
RTL Design & Test
Physical Design
Software & Compiler Design
Test & Maintenance
Original Development Time (e.g., 18months + 18months)

**Software takes 50% time!***

*Source: Chip Design and Manufacturing Cost under Different Process Nodes: Data... | Download Scientific Diagram (researchgate.net)

# Dynamic and Diverse Workloads vs Solution Specialization



# Challenges in a Nutshell

## *Dynamic, Diverse & Highly Customized*

**Dynamic & diverse**
Agility and
Fast turn-around times

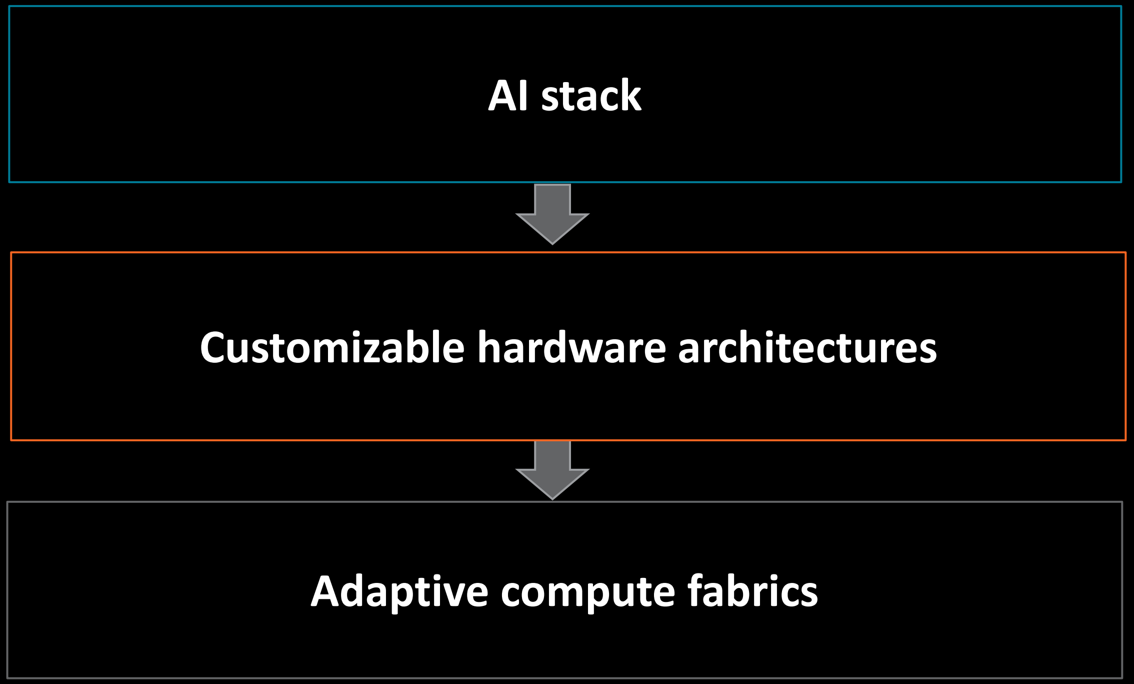
→ ⚡ ←

**Customization**
Hardware specialization
with
long development cycles

Agility in Customization is King.

# Our approach: Enabling Rapid Specialization with Adaptive Compute Fabrics and Agile AI Stacks

# Enabling Rapid Specialization with Adaptive Compute Fabrics and AI Stacks



AI stack

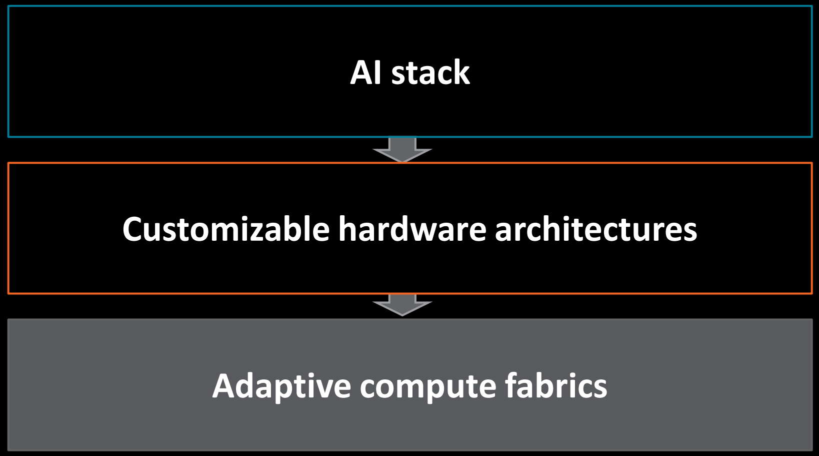
Customizable hardware architectures

Adaptive compute fabrics

Brevitas

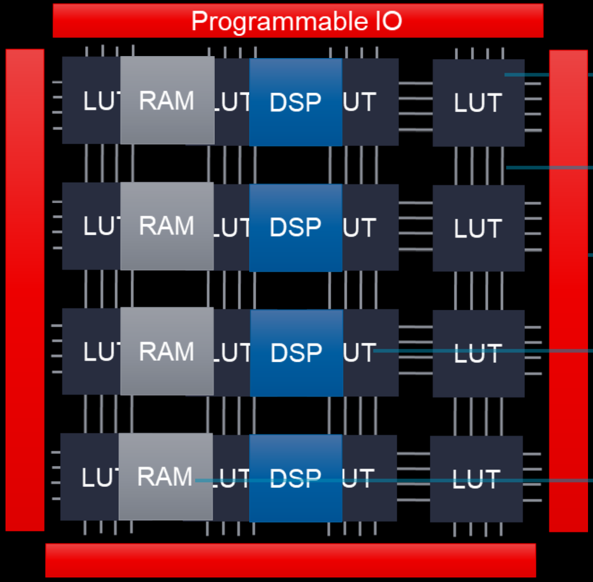
FINN

# What are adaptive compute fabrics?
# FPGAs and AIEs



# Primer: Adaptive Computing – FPGAs

- FPGAs are the **chameleon** amongst the semiconductors:  flexible, adaptive mostly homogeneous hardware architectures that enable **post-production customization at the architectural level**
- Customize
  - IO interfaces
  - **Functionality post-silicon** (compression, encryption, NN accelerator, key value store,…)
  - **Compute architectures** & **memory subsystems** to meet specific use case's performance or energy targets



Sea of programmable Lookup Tables (LUTs) ~millions

Programmable Interconnect

Programmable IO

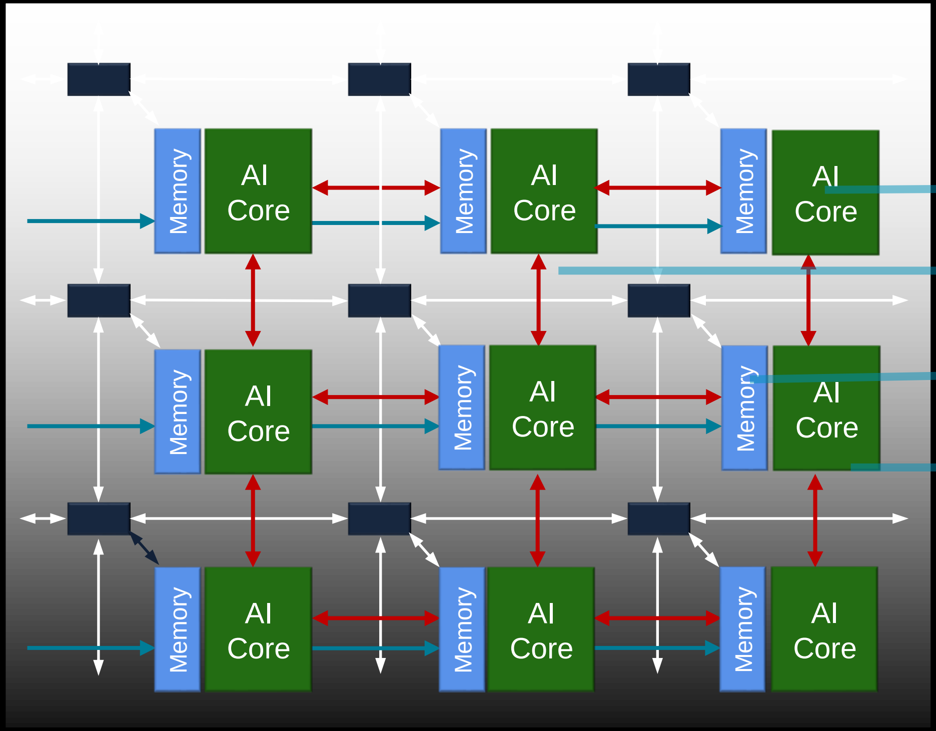
DSPs: n-bit MAC

Embedded SRAM ~high bandwidth

# Primer: Adaptive Computing – AIEs

- AI Engines (AIEs): new form of higher performant, adaptive compute fabric
  - Higher performance through hardened vector processing in VLIW cores, just word-based (instead of bit-based) with native support for ML-optimized data types (e.g., INT8, block float,…)
  - Great flexibility because of interconnectivity and separate control flow

    => **adapt the execution architecture to different workloads**



Matrix of VLIW/SIMD vector processors (10s...100x)

Flexible interconnect

Tightly coupled, embedded memory (1..10sMB)

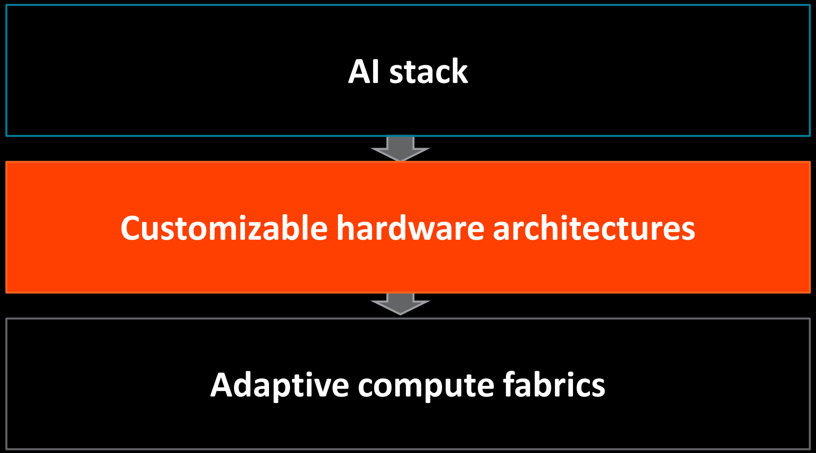
AIE are software compiled and don't require synthesis

# FPGAs Are Diverse and Widely Deployed

- ~100 Product Families
  - Spanning Si nodes from 350nm to 7nm
  - Devices ship for +20s year (for example Coolrunner XPLA3)
- 500+ Base Parts
  - Different fabric sizes and mixtures between DSPs and LUTs
  - Combinations with high-speed serial IO, ADC, HBM, ARM cores and other hard IP
- 3 basic temperature grades & 3 speed grades
- Other variants
  - Radiation hardened and custom variants

**FPGAs available in a broad spectrum of parts to cater to the diverse requirements in pervasive AI**

# Extreme Specialization of the Hardware Architecture (post silicon)



AI stack

Customizable hardware architectures

Adaptive compute fabrics

# Key Concepts

Custom Dataflow
Quantization
Sparsity

Customized for Specific Topologies

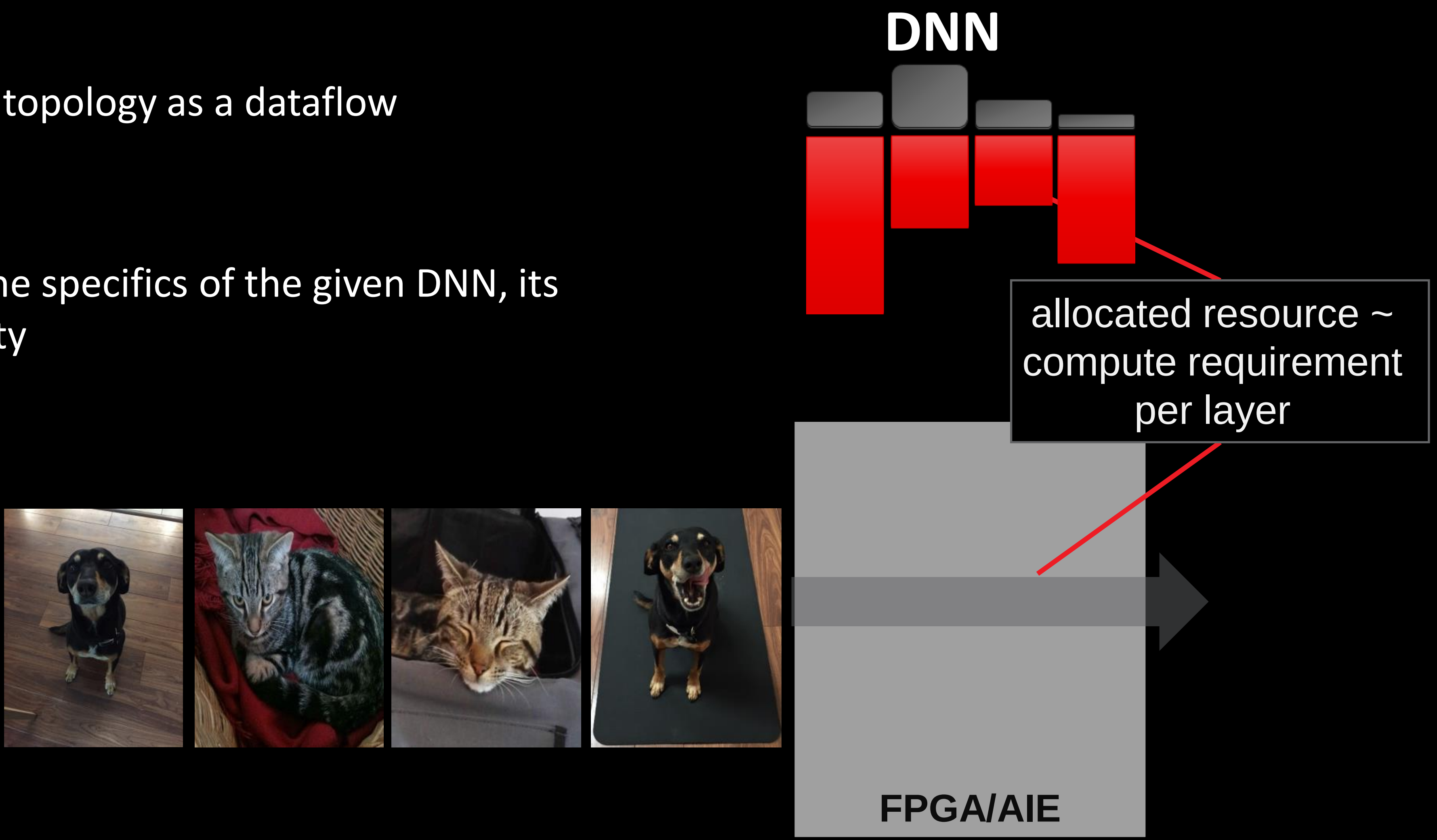
Customized in datatypes

Sparse neural circuits

# Dataflow - Specializing for Individual Topologies

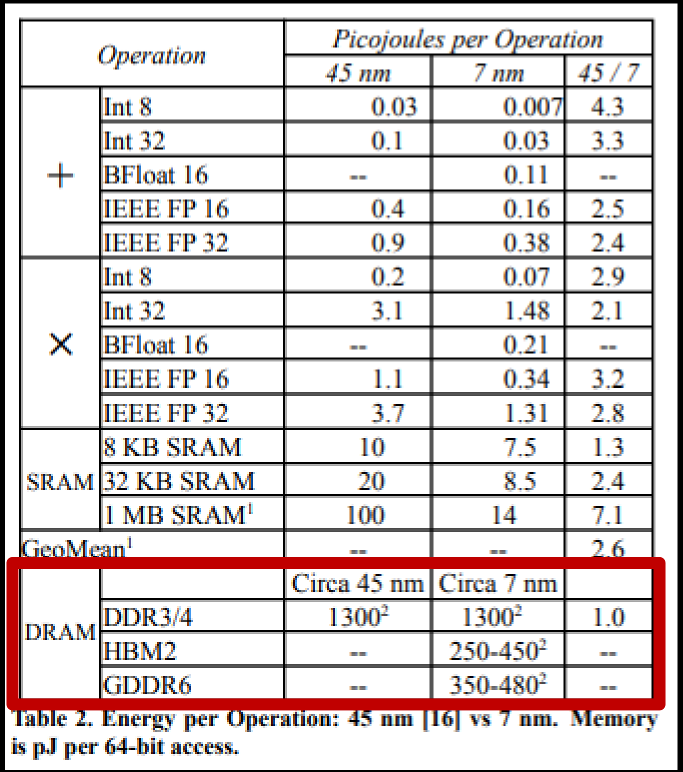
- Hardware instantiates the topology as a dataflow architecture

- Customize everything to the specifics of the given DNN, its operations and connectivity



# Energy Efficiency through Dataflow

- Architecture only computes and stores what's needed in the specific use case
  - Customized memory and compute subsystem
- Minimizes movement & storing of data
  - Activations are not buffered externally; they are in SRAM and registers moved directly from one layer to next
- High efficiency through concurrent communication and compute
  - Each layer starts computing as soon as first inputs are available

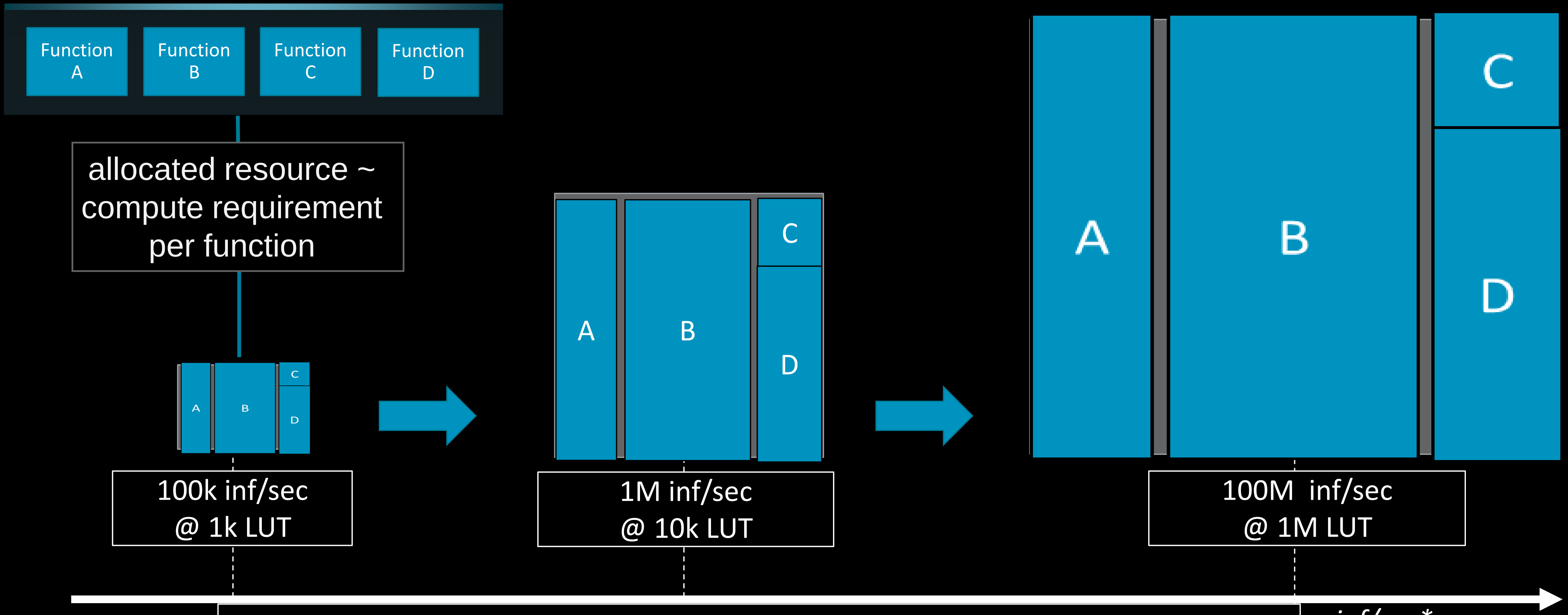


| *Operation* | | *Picojoules per Operation* | | |
|---|---|---|---|---|
| | | *45 nm* | *7 nm* | *45 / 7* |
| + | Int 8 | 0.03 | 0.007 | 4.3 |
| | Int 32 | 0.1 | 0.03 | 3.3 |
| | BFloat 16 | -- | 0.11 | -- |
| | IEEE FP 16 | 0.4 | 0.16 | 2.5 |
| | IEEE FP 32 | 0.9 | 0.38 | 2.4 |
| × | Int 8 | 0.2 | 0.07 | 2.9 |
| | Int 32 | 3.1 | 1.48 | 2.1 |
| | BFloat 16 | -- | 0.21 | -- |
| | IEEE FP 16 | 1.1 | 0.34 | 3.2 |
| | IEEE FP 32 | 3.7 | 1.31 | 2.8 |
| SRAM | 8 KB SRAM | 10 | 7.5 | 1.3 |
| | 32 KB SRAM | 20 | 8.5 | 2.4 |
| | 1 MB SRAM[1] | 100 | 14 | 7.1 |
| GeoMean[1] | | -- | -- | 2.6 |
| DRAM | | Circa 45 nm | Circa 7 nm | |
| | DDR3/4 | 1300[2] | 1300[2] | 1.0 |
| | HBM2 | -- | 250-450[2] | -- |
| | GDDR6 | -- | 350-480[2] | -- |

**Table 2. Energy per Operation: 45 nm [16] vs 7 nm. Memory is pJ per 64-bit access.**

Jouppi, Norman P., et al. "Ten lessons from three generations shaped Google's TPUv4i: *ISCA'*2021.

# Dataflow Can Adapt and Scale to Diverse Workloads



Function A

Function B

Function C

Function D

allocated resource ~ compute requirement per function

100k inf/sec
@ 1k LUT

1M inf/sec
@ 10k LUT

100M inf/sec
@ 1M LUT

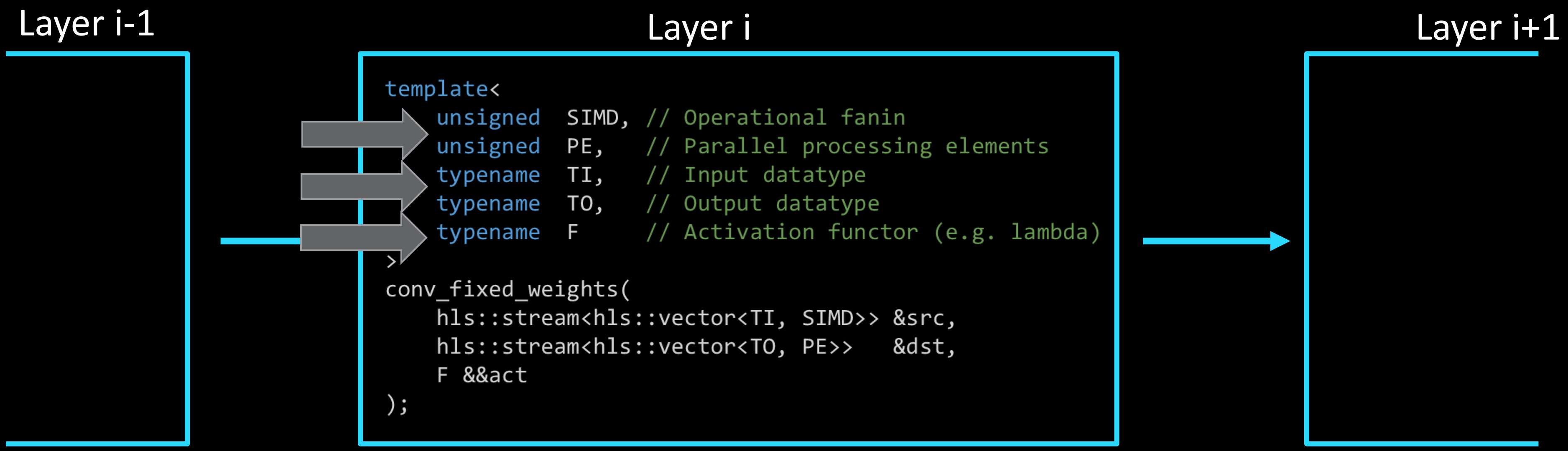
Dataflow can scale performance & resources to meet diverse requirements
Without batching!

*inf/sec**

*Inf/sec: inferences/second

# Parameterizable Kernel Library

- Kernels representing the individual layers, which can be parameterized
  - Degree of parallelism (output channels, input channels, kernel dimensions …) for different performance/resource trade-offs
  - Datatypes (INT8, ternary, INT2, …)
  - Behaviour (activation function)
- Composable through streaming I/O
- Programmed in synthesizable C++ (Vitis HLS)



```
template<
    unsigned  SIMD, // Operational fanin
    unsigned  PE,   // Parallel processing elements
    typename  TI,   // Input datatype
    typename  TO,   // Output datatype
    typename  F     // Activation functor (e.g. lambda)
>
conv_fixed_weights(
    hls::stream<hls::vector<TI, SIMD>> &src,
    hls::stream<hls::vector<TO, PE>>   &dst,
    F &&act
);
```

https://github.com/Xilinx/finn

# Key Concepts

Custom Dataflow
Quantization
Sparsity

Customized for Specific Topologies

Customized in datatypes

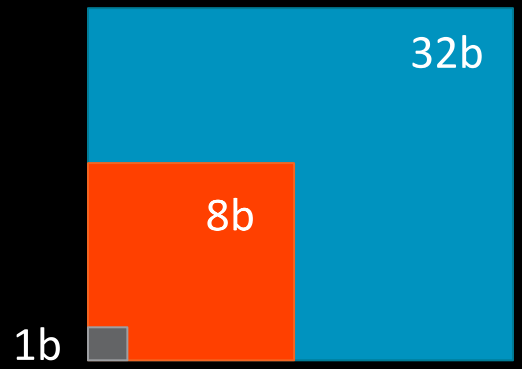
Sparse neural circuits

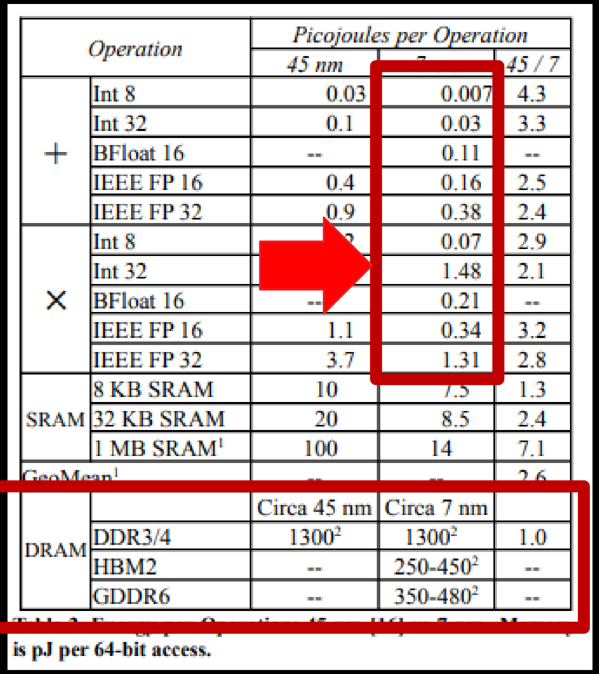
# Customizing Arithmetic to Minimum Precision - FINN

- Popular approach which reduces bits in the data representation of weights and activations while preserving accuracy

- Reducing precision shrinks hardware cost/scales performance
  - For integer datatypes, LUT cost proportional to bitwidths in weight and activations (for ex. INT1 : INT8: 70x)
  - Instantiate n-times more compute within the same fabric, thereby scale performance n-times

- Energy
  - Faster execution => less energy ($E = P * time$)
  - Using reduced precision operators saves energy
  - Reduces memory footprint
    - ResNet50 @ 32b: 102.5MB, ResNet50 @ 2: 6.4MB
    - NN model can stay on-chip => no external memory access => saves energy





| Operation | | Picojoules per Operation | | |
|---|---|---|---|---|
| | | 45 nm | 7 [illegible] | 45 / 7 |
| + | Int 8 | 0.03 | 0.007 | 4.3 |
| | Int 32 | 0.1 | 0.03 | 3.3 |
| | BFloat 16 | -- | 0.11 | -- |
| | IEEE FP 16 | 0.4 | 0.16 | 2.5 |
| | IEEE FP 32 | 0.9 | 0.38 | 2.4 |
| × | Int 8 | [illegible] | 0.07 | 2.9 |
| | Int 32 | [illegible] | 1.48 | 2.1 |
| | BFloat 16 | -- | 0.21 | -- |
| | IEEE FP 16 | 1.1 | 0.34 | 3.2 |
| | IEEE FP 32 | 3.7 | 1.31 | 2.8 |
| SRAM | 8 KB SRAM | 10 | 7.5 | 1.3 |
| | 32 KB SRAM | 20 | 8.5 | 2.4 |
| | 1 MB SRAM[1] | 100 | 14 | 7.1 |
| GeoMean[1] | | -- | -- | 2.6 |
| DRAM | | Circa 45 nm | Circa 7 nm | |
| | DDR3/4 | 1300[2] | 1300[2] | 1.0 |
| | HBM2 | -- | 250-450[2] | -- |
| | GDDR6 | -- | 350-480[2] | -- |

[illegible] is pJ per 64-bit access.

Jouppi, Norman P., et al. "Ten lessons from three generations shaped google's tpuv4i: *ISCA'*2021.

# Key Concepts

Custom Dataflow

Quantization

Sparsity

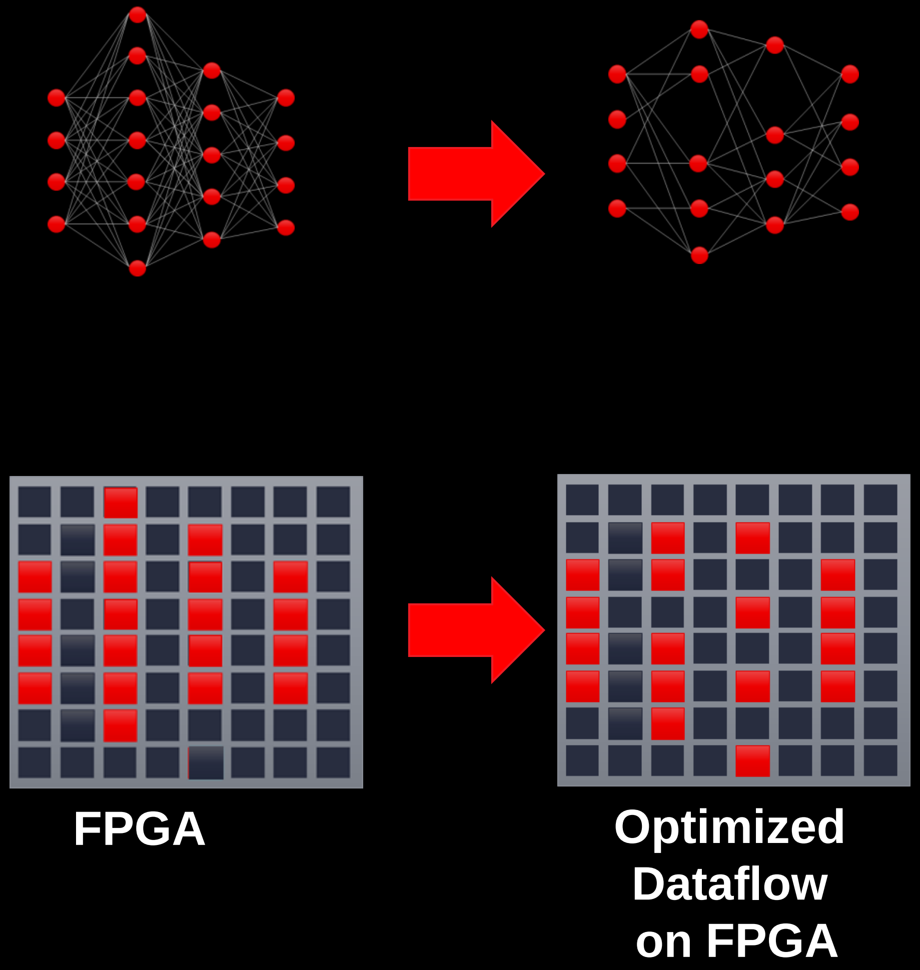
Customized for Specific Topologies

Customized in datatypes

Sparse neural circuits

# Energy Savings through Fine-Granular Sparsity

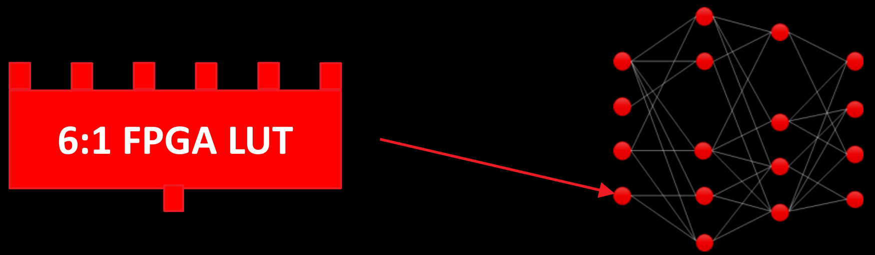
- DNNs are naturally sparse
- Sparse topologies result in irregular compute patterns which are difficult to accelerate on vector- or matrix-based execution units
  - Poor efficiency
- With streaming dataflow architectures, where every neuron and synapse is represented in the hardware, we can maximize efficiency



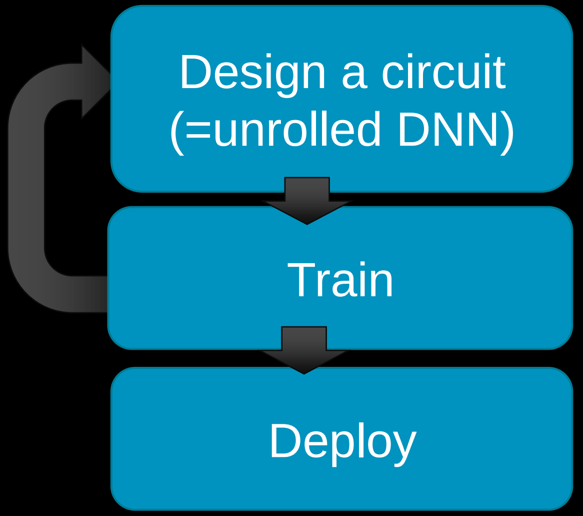
# Sparse Neural Circuits - LogicNets

- Massive scope to improve ML efficiency through sparsity
  - The human brain is highly sparse (98%) & operates on the power of a light bulb (~20W)*

- Idea
  - A LUT in an FPGA can represent a neuron
  - Design a highly sparse circuit in an FPGA
  - Represent this as a DNN to the training framework
  - Learn the look-up table contents



**High efficiency and maximum performance by design (classification at clock rate)**

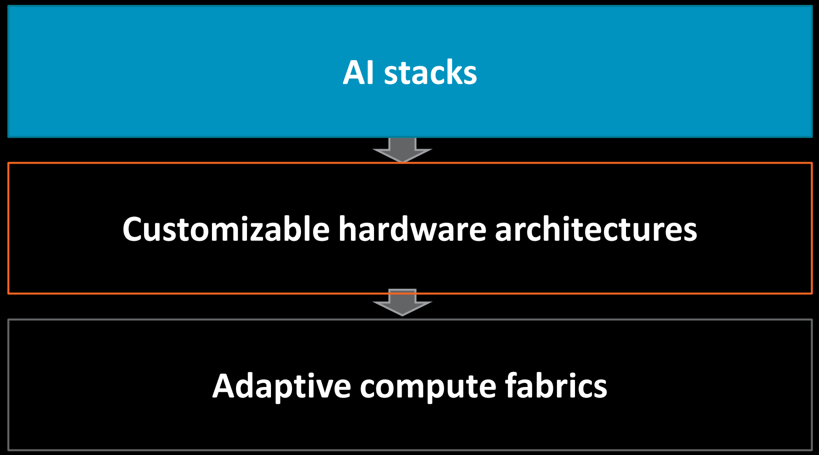


Adjust the parameters of DNN (=LUT contents) while iterating on training dataset until accuracy

*https://www.numenta.com/blog/2022/05/24/ai-is-harming-our-planet/

Umuroglu, Yaman, et al. "LogicNets: co-designed neural networks and circuits for extreme-throughput applications." FPL’2020.

# How can we support this specialization through agile AI stacks? (FINN with Brevitas)

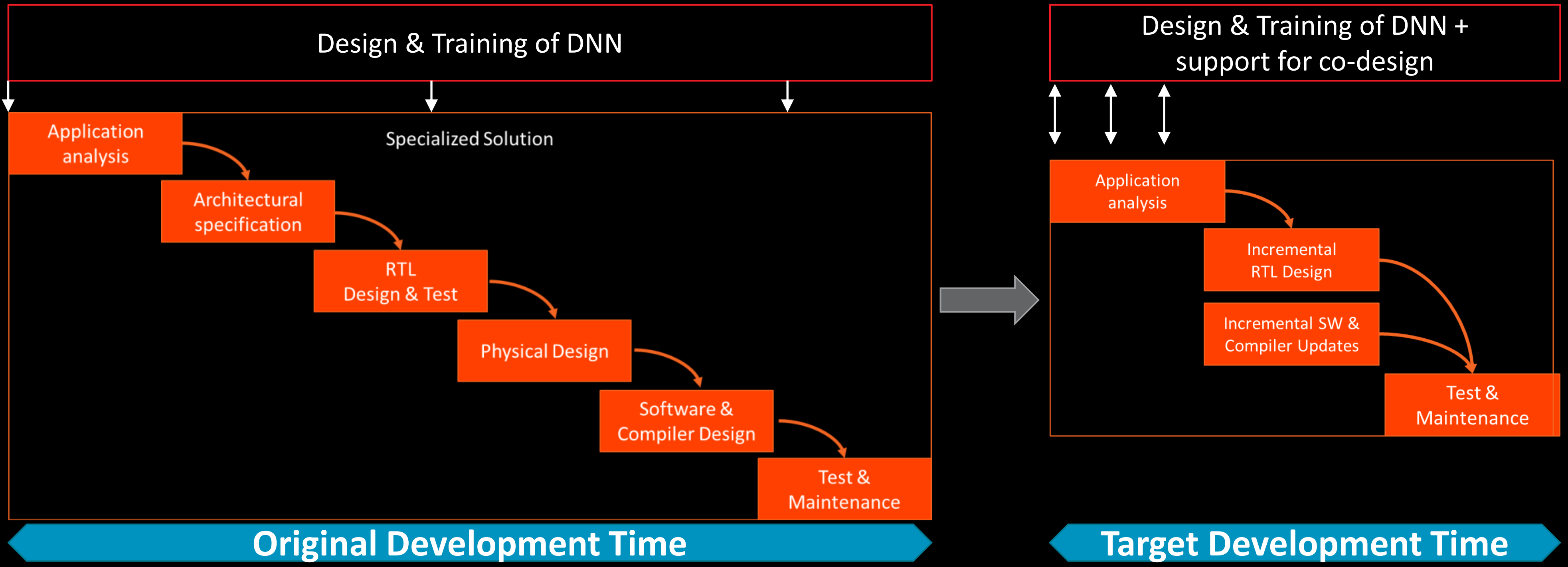


AI stacks

Customizable hardware architectures

Adaptive compute fabrics

# Faster Iterations with Shortened Development Cycles



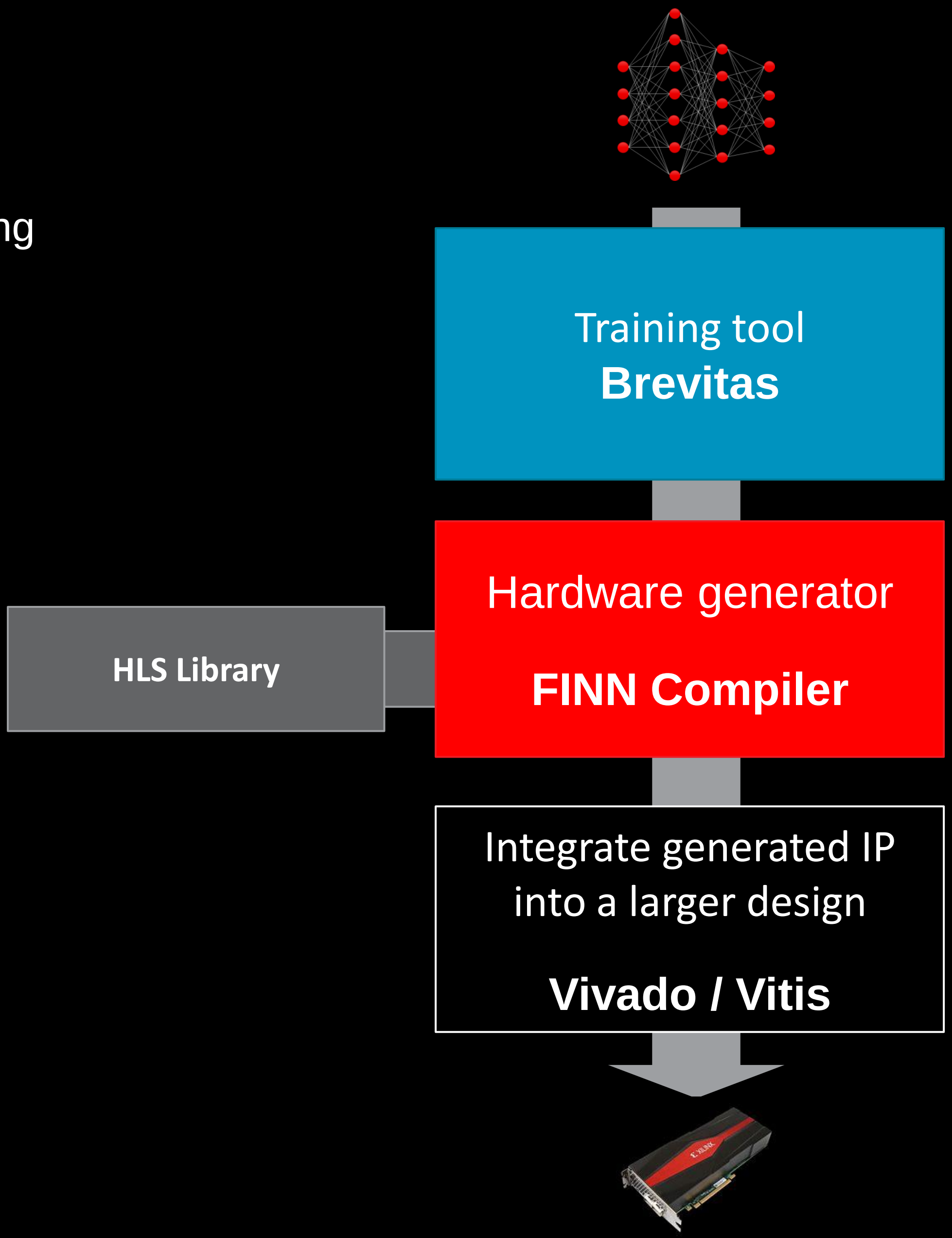
- Adaptive Computing eliminates the need for physical design
- Generalizable architectures which can incrementally adapt to new requirements
- Paired with graph compiler which automates the specialization
- Agile quantization support in training library

# Example: FINN & Brevitas

- End-to-end flow – from DNN to bitstream
  - Enables generation of highly customized hardware architectures using **quantization** and **dataflow** and **fine-granular sparsity**

- Components
  - Training tool: Brevitas
  - Hardware generator (FINN)
    - Kernel library (HLS)

- Open source
  - Enable customization for new layers, datatypes, etc.
  - Flexibility to adapt to fast-moving application space
  - Third party contributions



Training tool
**Brevitas**

Hardware generator
**FINN Compiler**

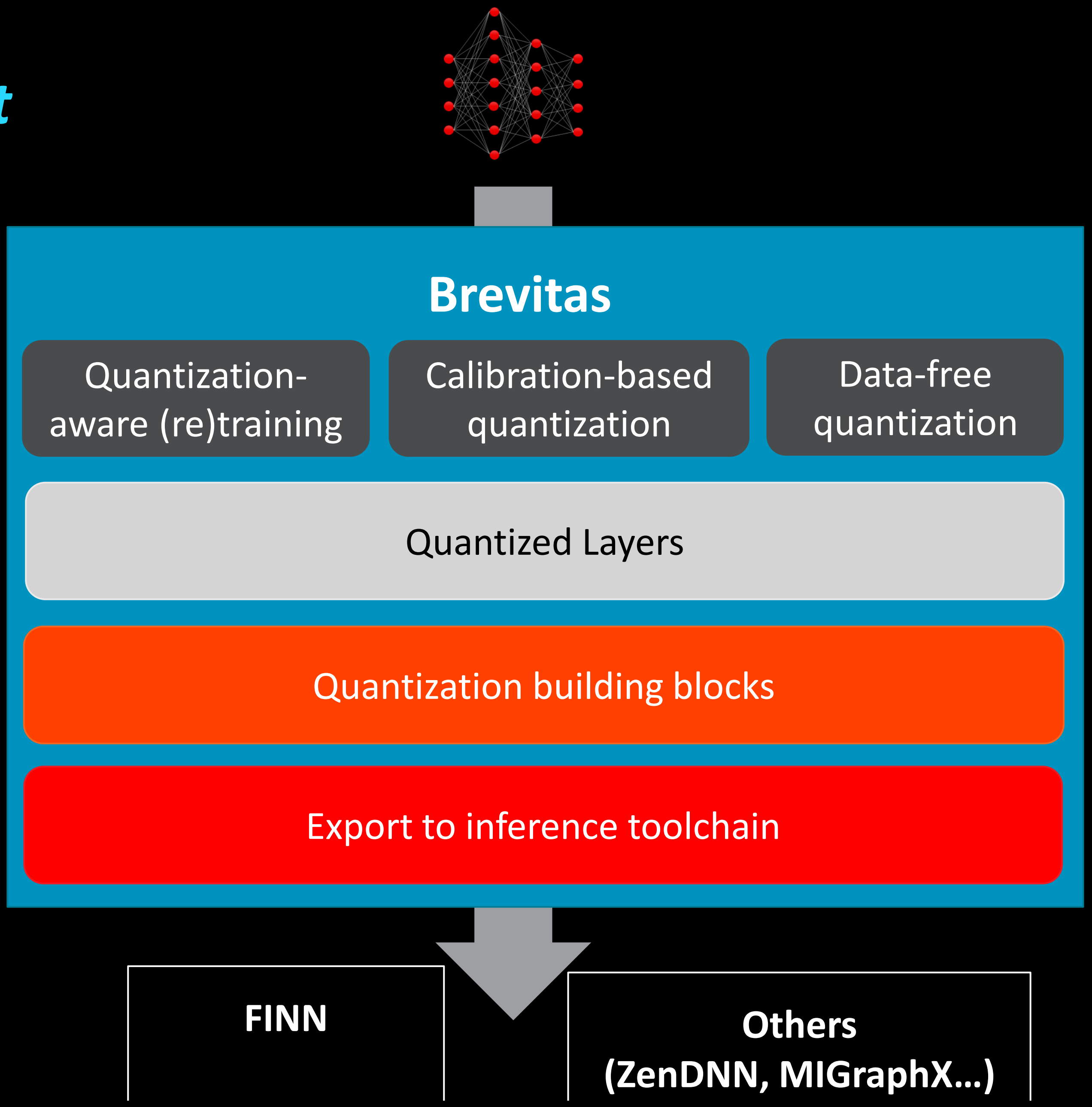
**HLS Library**

Integrate generated IP into a larger design
**Vivado / Vitis**

# Brevitas - PyTorch Library

## *Offering Agile Quantization Support*

- First class support for custom datatypes and operators at ML framework level
  - Arbitrary precision integer, float, block-style quantization
  - Extendible to user-defined datatypes and operators and support for any hardware-specific datatype at training
- Composable building blocks at multiple abstraction levels that can be arbitrarily combined
- Integration with different compiler stacks
  - Exports commonly used representation format (for example ONNX)



Brevitas

Quantization-aware (re)training

Calibration-based quantization

Data-free quantization

Quantized Layers

Quantization building blocks

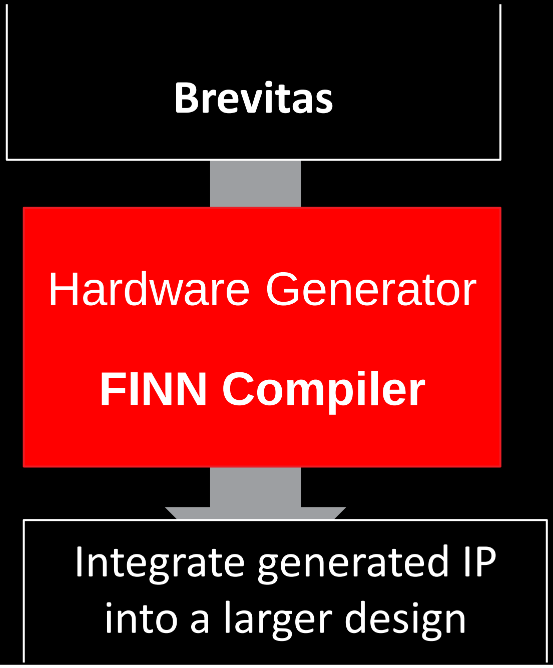
Export to inference toolchain

FINN

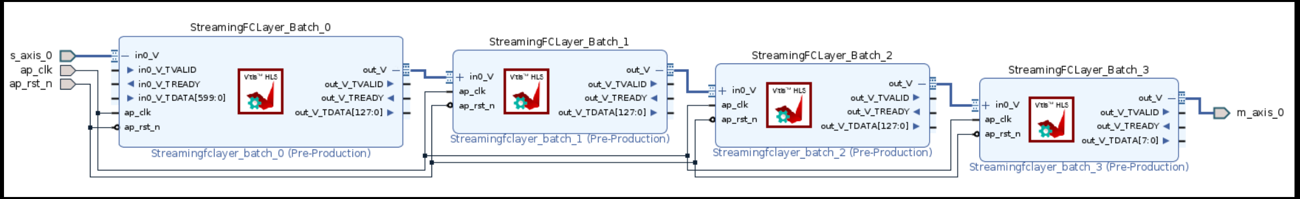
Others (ZenDNN, MIGraphX…)

# The FINN Compiler

- Modular **graph compiler** with well-defined abstraction levels
- Incrementally lowers ONNX graph to a hardware description through **transformations**
- Performs **optimizations**
  - Layer fusion
- Explores the **design space**
  - Calculates the degrees of  parallelism for each kernel using resource cost and performance models
- **Code-generate**s a dataflow C++ description using the parameterizable **kernel library**
- Creates **DNN hardware IP**



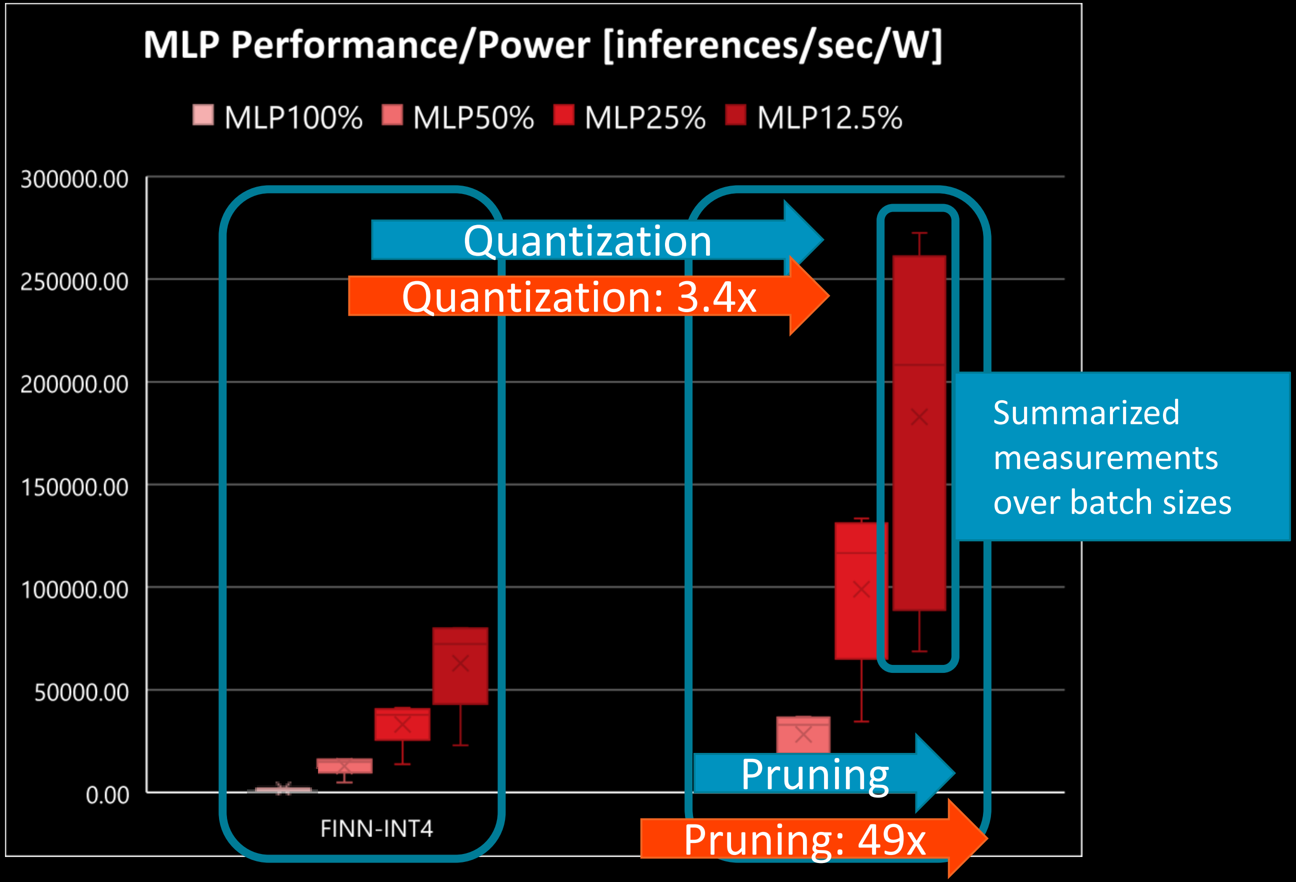
```
hls::stream<ap_int<185>> in
hls::stream<ap_int<100>> inter0, inter1, ...
...
StreamingFCLayer<BINARY, BINARY, ..>(in, inter0, ...)
StreamingFCLayer<BINARY, BINARY, ..>(inter0, inter1, ..)
...
```



# Some Example Results

# Energy Efficiency through Quantization and Sparsity

- Benchmarking activity* across topologies, devices and optimization schemes
- Example representing typical behaviour: one MLP and one CNV, using quantization & pruning on an FPGA (FINN)

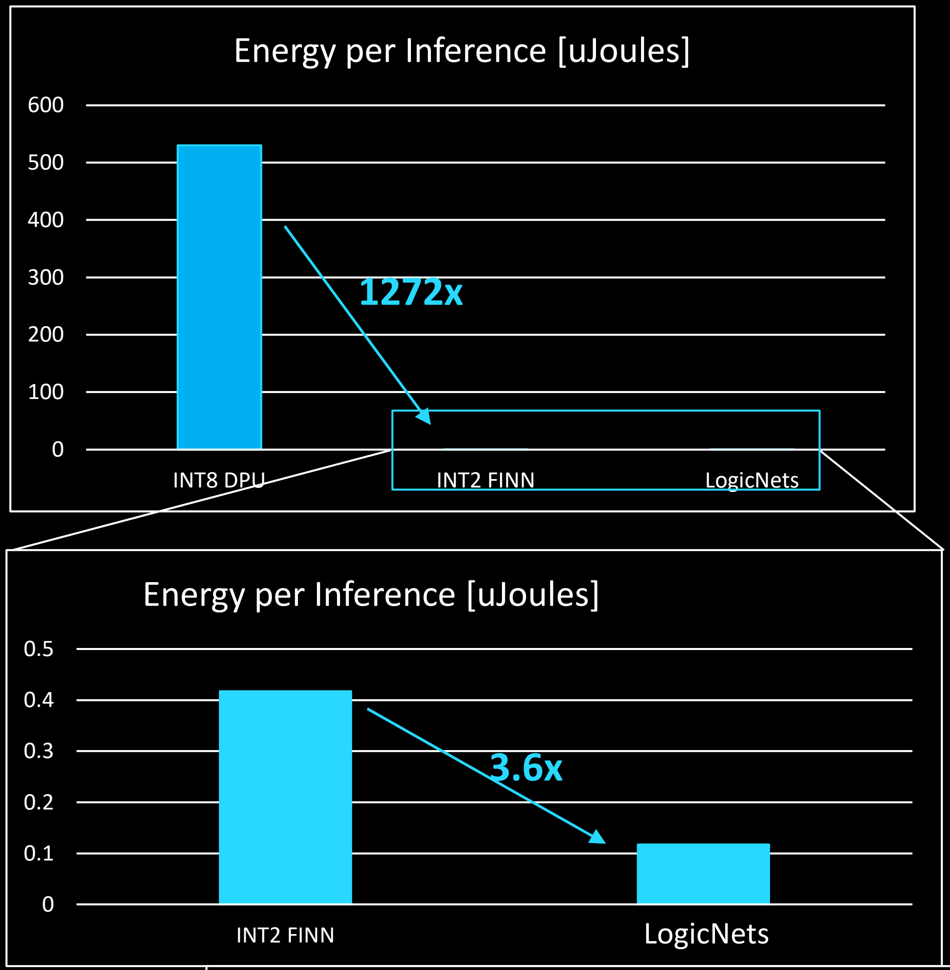


**Significant energy efficiency through pruning and quantization on FPGAs possible**

*QuTiBench (rcl-lab.github.io)
http://www.tara.tcd.ie/handle/2262/96391

# Energy Efficiency: FINN & LogicNets

## *Results Demonstrate the Potential*



Reducing precision & Dataflow => 1272 improvement

LogicNets: 3.6x over FINN

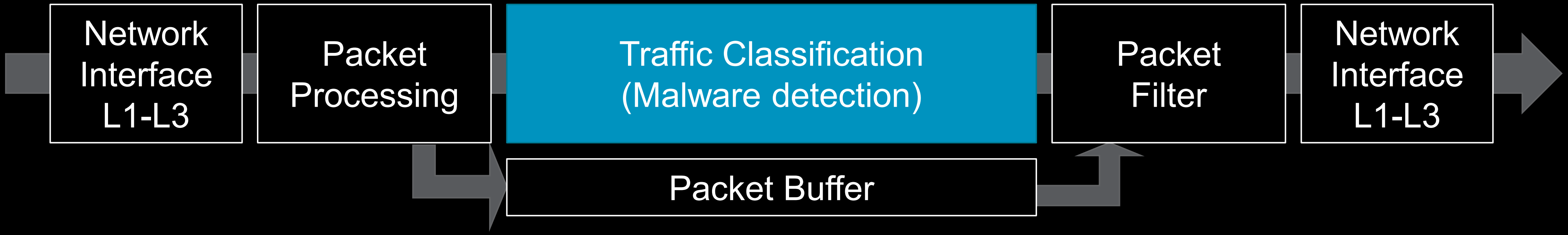
Details:
Network Security Application
Malware Classifier
UNSW dataset
MLP 92k Ops/inference
INT8 with VitisAI,
INT2 with Brevitas and FINN
Board power ZCU104

Energy calculated
LogicNets assume

**Total: ~4500x Energy Improvement through Post-Silicon Hardware Specialization**
**Much more work coming...**

# Diversity

## *Cyber Security – 300M inferences/sec and 18nsec latency*



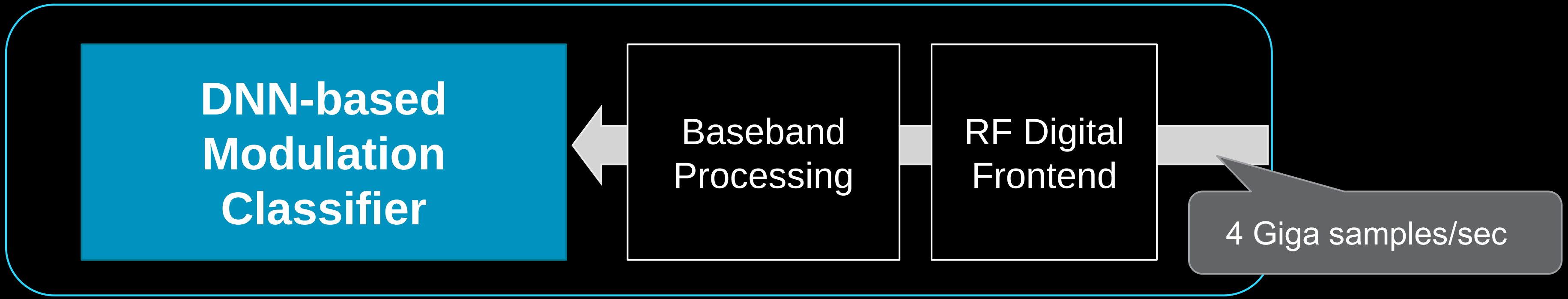
- **FINN implementation of UNSW-NB15 malware classifier**
  - 300M inferences/sec with 18nsec latency
  - 8k LUT

- **Customer engagement**
  - Leveraging quantization to 4b and learned sparsity – no accuracy loss!
  - 2000x speed-up over initial CPU implementation (processing 15M char/sec)

# Diversity

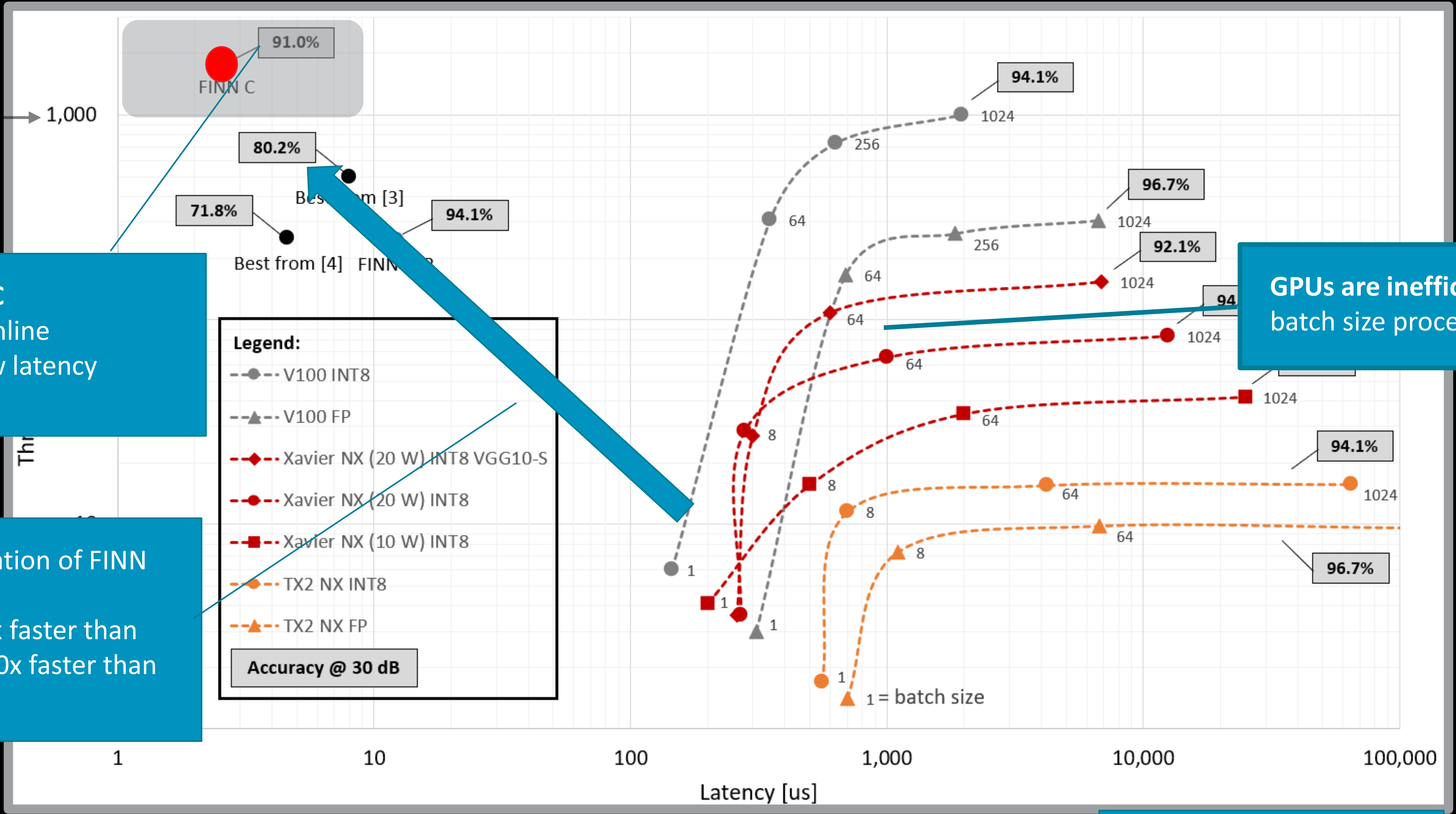
## *Modulation Classification: GHz sampling rate & usec latency*

- Rapidly label + understand RF spectrum
  - What modulations are used?
- Key enabler for many applications and key component of an AI-enabled (cognitive) software-defined radio
  - e.g., spectrum interference monitoring, dynamic spectrum access
- DNNs promising for modulation classification



**Challenge**: at GHz sampling, we need Minfps inference throughput

# DNN-Based Modulation Classification (RadioML)



> GHz

**Dataflow on RFSOC**
Enables real-time inline processing with low latency

Customer Evaluation of FINN **confirms**:
measured >100x faster than Xeon CPU and 10x faster than GPU

**GPUs are inefficient** at low batch size processing

**Low latency**

DF on a ZCU111: 1.75GSamples/sec, 2.6usec latency

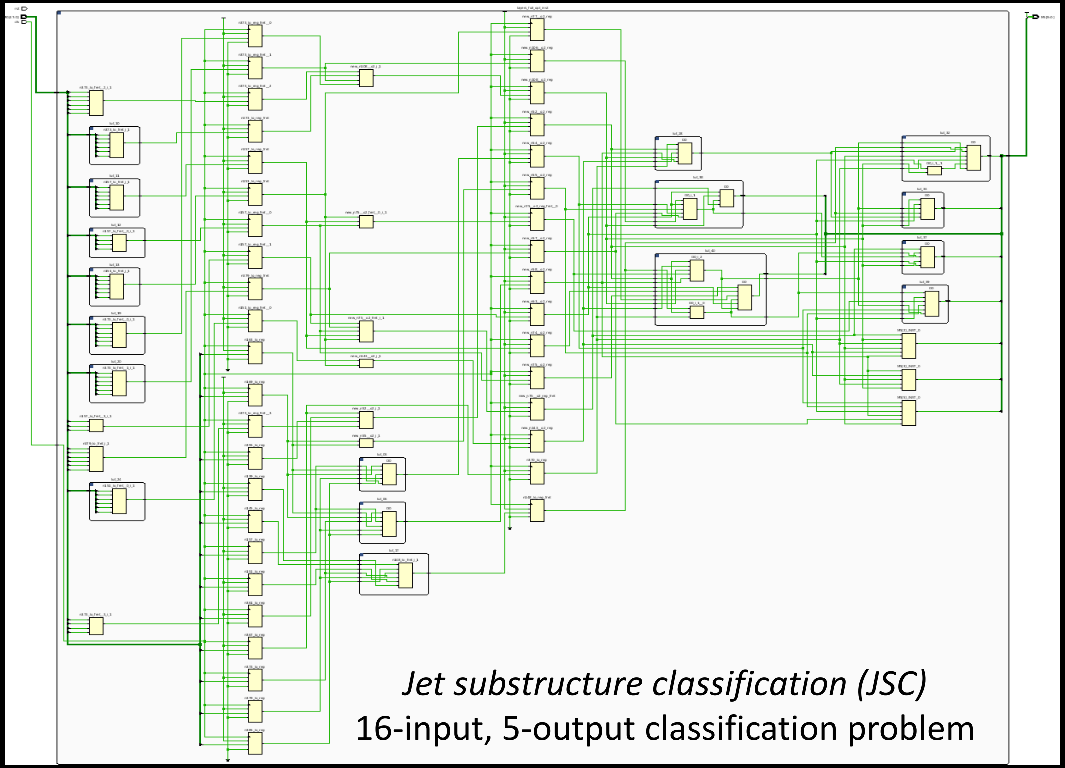
# Diversity

## *LogicNets Results – Tiny (!!!) and Fast*

- **DNN in similar area compared to an FPGA 32b adder**

- **High energy particle physics CERN L1 trigger experiment**
  - Inference rate: 666 Minferences/sec*
  - Latency: 3 nsec
  - Resources: 30 LUTs

**A Complete Neural Network @ 70% Accuracy!**



*Jet substructure classification (JSC)*

16-input, 5-output classification problem

Synthesized with Vivado 2019.2; $F_{Max}$ equals inference rate

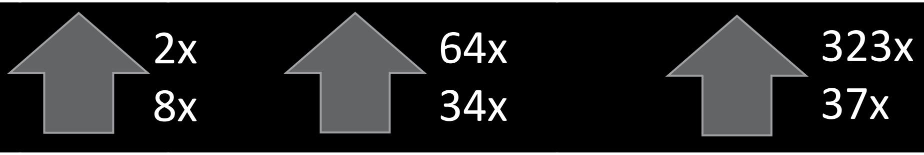
*max device frequency

# Diversity

## *LogicNets Results*

- *Quotation from Petersen et al., Dec 2022 @ NeurIPS:*
  - *“FINN […] the **fastest method** for classifying MNIST at an accuracy of 98.4%,”**

| | Acc. [%] | LUT | Latency [nsec] | Inferences/sec |
|---|---|---|---|---|
| FINN | 98.4 | 83k | 2,440 | 1.6M |
| | 95.8 | 91k | 310 | 12.4M |



2x
8x

64x
34x

323x
37x

| | Acc. [%] | LUT | Latency [nsec] | Inferences/sec |
|---|---|---|---|---|
| **LogicNets-M** | **97.7** | **45k** | **38** | **517M** |
| **LogicNets-S** | **95.8** | **12k** | **9** | **458M** |

**“World’s fastest MNIST classifier”* - now even faster**

Synthesized with Vivado 2019.2; $F_{Max}$ equals inference rate

*Petersen et al. "Deep Differentiable Logic Gate Networks." NeurIPS, 2022.

# FINN: Diverse Engagements and Open-Source Adoption

- Communications
- Medical
- Sensor Intelligence
- Automotive
- High-energy particle physics
- Aerospace & Defense
- High frequency Trading

- Open Source Adoption
  - ~2000 stars, 230k+ Brevitas downloads, 17k+ FINN compiler downloads

- 3 best paper awards
- ~1000 citations

https://xilinx.github.io/finn

https://github.com/Xilinx/brevitas

# Summary

**Pervasive AI: dynamic and diverse long tail of AI applications**

**Paradigm shift towards energy efficiency**

**Enabling Rapid Specialization with Adaptive Compute Fabrics and Agile AI Stacks**

**Adaptive computing available in great diversity** and can help by **customization of hardware execution architectures**

- Dataflow, shrinking precision, fine granular sparsity

  **Customization => energy savings**
  **Dataflow => scalable diversity**
  **LogicNets => tiniest and fastest***

**Speed-up and automated specialization through graph compilers such as FINN and training libraries Brevitas**

Proof points from FINN, Brevitas and LogicNets
demonstrate the potential for energy savings, and addressing truly divers requirements

*Petersen et al. "Deep Differentiable Logic Gate Networks." NeurIPS, 2022.

# HACCs: Heterogeneous Accelerated Compute Clusters

## Focus on heterogenous and adaptive computing

- Free usage for research with easy access
- Supporting high end compute research
- Bare metal access to adaptive compute hardware
- HACC community
- Growing community of over 100 institutions

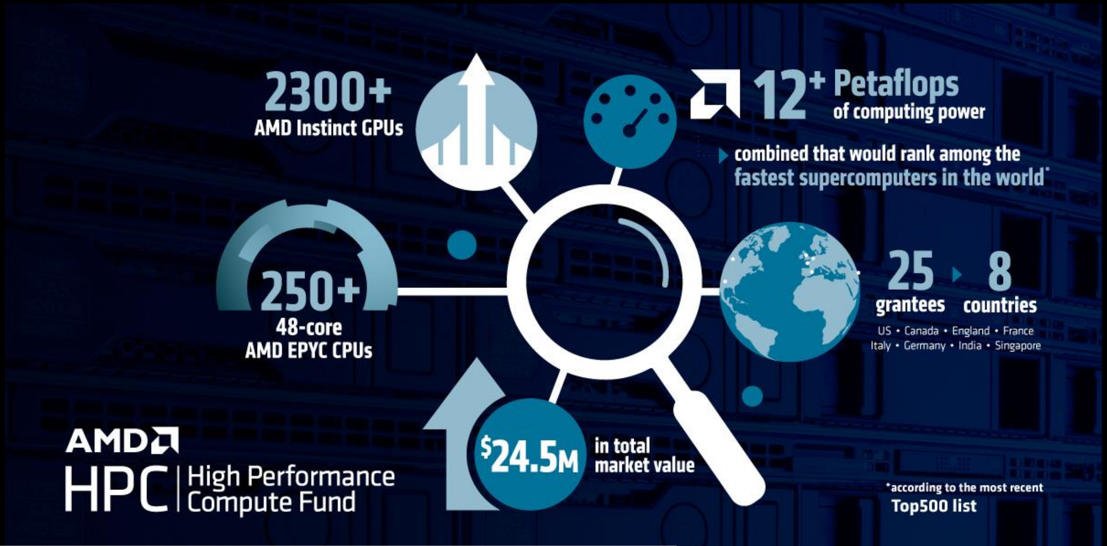
ILLINOIS
UCLA
ETH zürich
PADERBORN UNIVERSITY
NUS National University of Singapore

www.amd-haccs.io

# AMD HPC Fund

## Accelerating Science in the Public Interest

- Cloud access to AMD HPC CPU GPU technologies
- Customized technical training
- E-learning sessions
- Networking opportunities with peers around the world


2300+
AMD Instinct GPUs
12+ Petaflops
of computing power
combined that would rank among the fastest supercomputers in the world*
250+
48-core
AMD EPYC CPUs
25 grantees
8 countries
US • Canada • England • France
Italy • Germany • India • Singapore
$24.5M in total market value
AMD
HPC | High Performance Compute Fund
*according to the most recent Top500 list

Get involved!

https://www.amd.com/en/corporate/hpc-fund

https://www.amd-haccs.io/

AMD EPYC

INSTINCT

ALVEO™

ILINX

VERSAL™

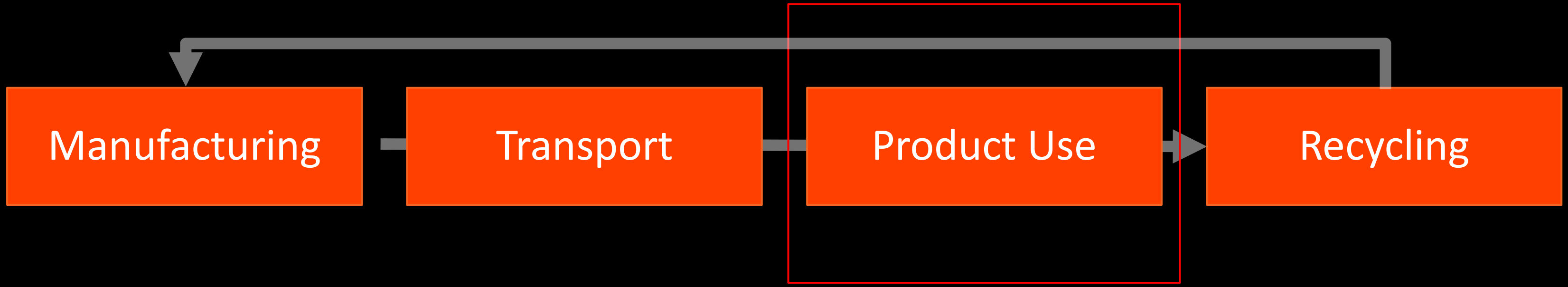
AMD

# Abstract

- In the context of AI, we face a plethora of challenges that extend beyond the widely discussed performance scalability required to meet the growing demands of compute and storage in the latest models. These challenges encompass sustainability, pervasiveness, agility, and diversity, which is needed to cater to a constantly evolving range of applications and algorithms from edge to cloud. In this talk, we explore how adaptive devices and agile compiler stacks can provide solutions by delivering post-production hardware specialization and co-designed algorithms. This results in highly optimized AI systems which not only provide the necessary performance scalability but also bring a reduction in carbon footprint while addressing the needs of a broad range of diverse applications and with the necessary agility.

# Sustainability

- **Definition**: Avoidance of the depletion of natural resources in order to maintain an ecological balance
- **Metric** : tons of $CO_2$
- $CO_2$ contributions in data centers [1]:



**Operational $CO_2$**
**= f(energy consumption)**

Source: adapted from Carole Wu, Meta, “Scaling AI Computing Sustainably” ISSCC’2023