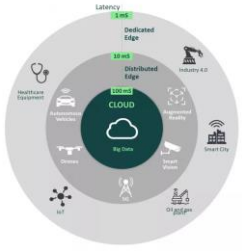
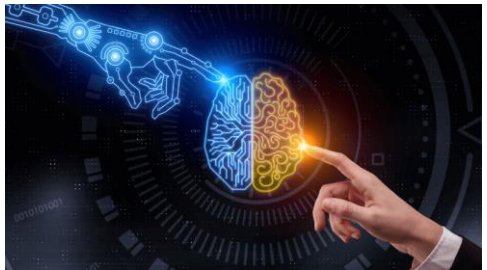
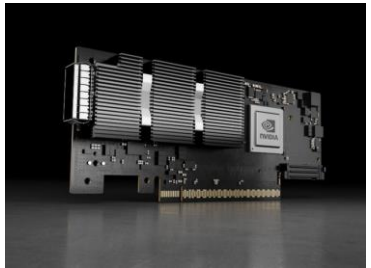
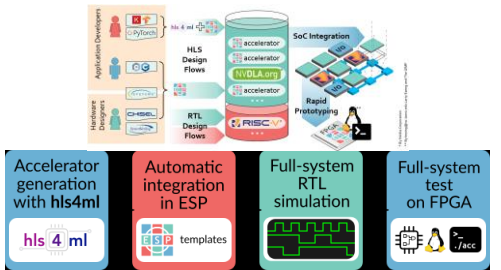
Advanced data réduction techniques with ML
– Methodology – Software – Hardware – Firmware –

- ➔ Dominant Design and Challenges
- ➔ Stakeholders and Technologies
- ➔ Methodology and optimized instruments
- ➔ Futur to prepare

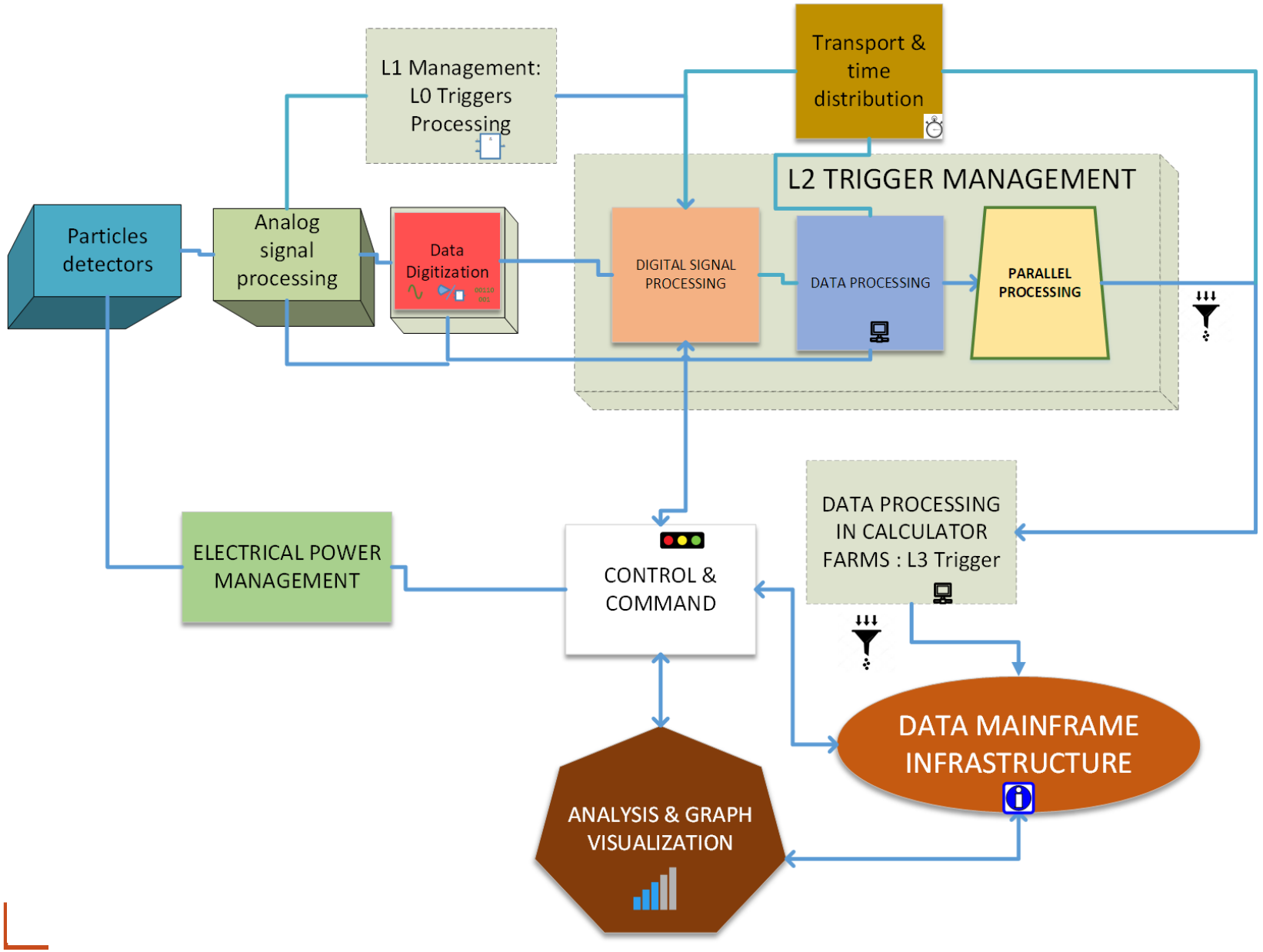




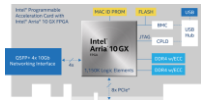


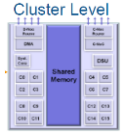


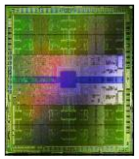
Dominant Design in Instruments for research in fundamental physics



Intelligents algorithms







- **Reduced Data and Selection management:**
  - *L1: FPGA, ASIC, SNN*
  - *L2: FPGA, GPU, SNN*
  - *L3: GPU, MPPA, Accelerated Card*

Challenges in the field

LHCb – 2032 ~2000 Exabytes/year
ATLAS+CMS 2027 ~ 260 Exabytes/year
Square Kilometers Array – 2030 ~ 30000 EB/year

2021 global Ethernet Dataflow ~2800 EB/year

DataStream before storage
LHCb – 2032 ~500TB/s
ATLAS+CMS 2027 ~ 20-40 TB/s

Forecast cost of storing data to disk (Annual)
LHCb – 2032 ~2,5 Billions of €
ATLAS+CMS 2027 ~ 325 Millions of €

Challenge: Real-time data reduction to avoid disk storage (very expensive):

➔ Embedded algorithms in decision nodes

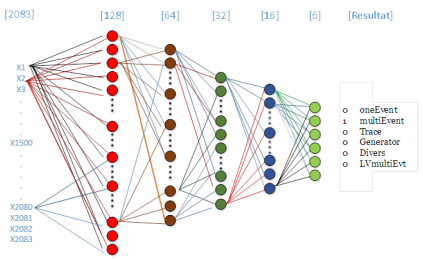
➔ Optimize processing (classifications, Prediction, Selection)

➔ Use a mixed GPU, MPPA, FPGA

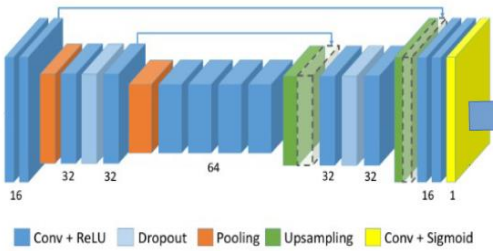
- **Use powerfull hardware component to compute ML Model**
- **And deploy them in ours instruments**

WHAT WE COULD DO WITH ML

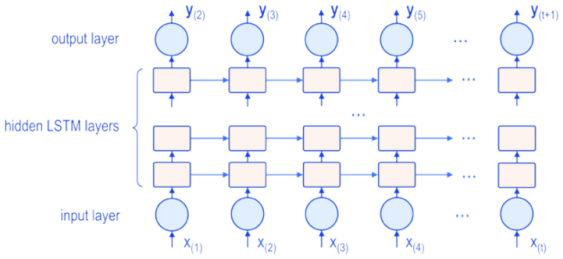
DEEP NEURAL NETWORK



CONVOLUTIONAL NEURAL NETWORK



RECURRENT NEURAL NETWORK



DECISION TREE

RANDOM FOREST

Off-line

- Signal generation
- Design Optimisation

On-line

- Instrument Optimization
  - Signal recognition
  - Pile-Up recovery
  - Signal deconvolution
- Selection/ Classification/ Decision
  - Data selection
  - Data parameters prediction
  - Denoizing
- Data Compression
  - Reduced data format

L1

L2

L3

Three mains technics

DATA DRIVEN

COMPUTING DRIVEN

Supervised Learning

- Makes machine Learn explicitly
- Data with clearly defined output is given
- Direct feedback is given
- Predicts outcome/future
- Resolves classification and regression problems

Unsupervised Learning

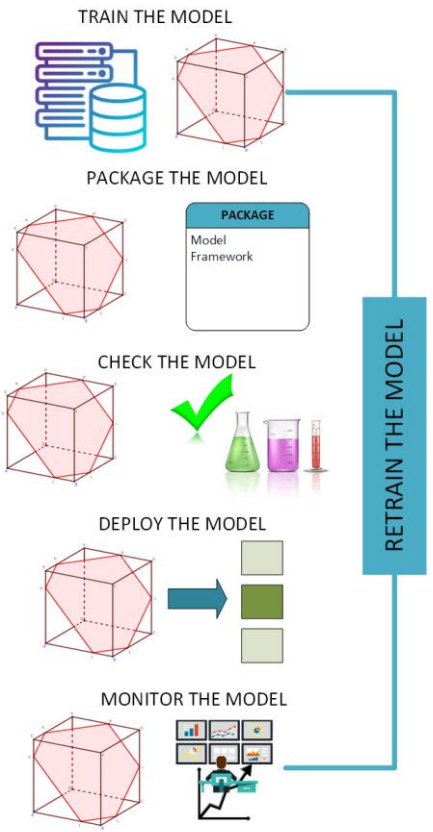
- Machines understands the data (Identifies patterns/ structures)
- Evolution is qualitative or indirect
- Does not predict/find anything specific

Reinforcement Learning

- An approach to AI
- Reward based learning
- Learning from +ve & +ve reinforcement
- Machine Learns how to act in a certain environment
- To maximize rewards

Embedded System

Stakesholders ➔ Responsive AI on the Edge



AI-Centric cloud

Big Data firms

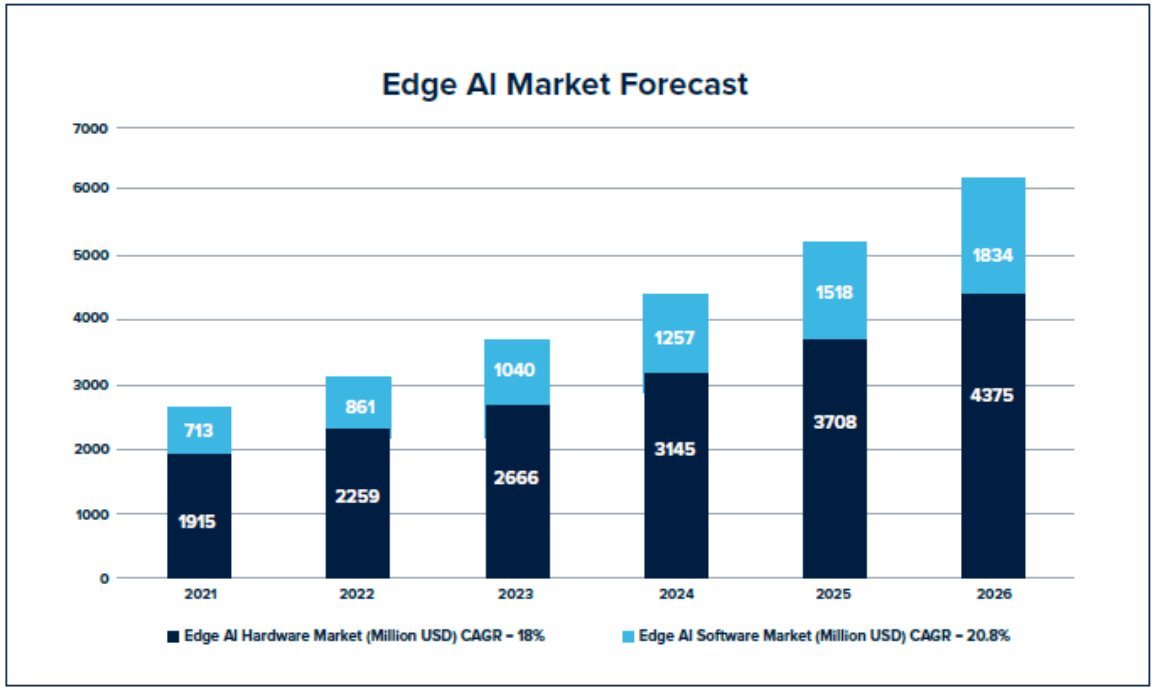
- Hungry in Electrical power
- Hungry in bandwidth
- Hungry in data quality

Migration

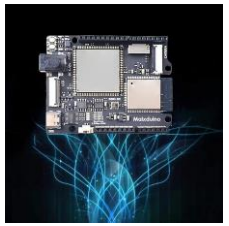
Embedded components network for edge AI

New Paradigme

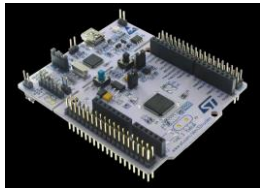
STMicroelectronics 2022



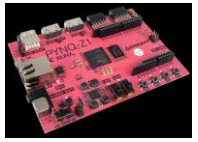
Kendryte K210



STM32 Cube AI



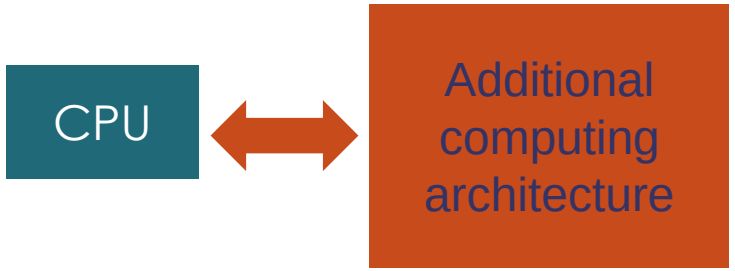
Digilent



AMD-Xilinx

Intel

Embedded AI: 2 technologies



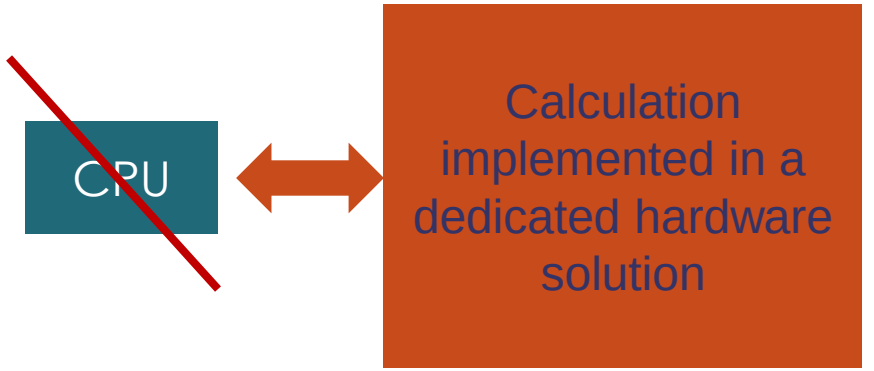
Edge AI Responsive

(~100µs to several second)

-- Based on software programmation

MMPA
GPU
FPGA – SOM-
PU designed for AI(TPU, KPU…)

1 Pilot current industrial developments



Spatial Accelerators = Fully Firmware

(~10ns to several 10µs)

-- based on hardware functions dedicated to calculations without software (matrix calculation)

ASIC
Neuromorphic Circuit
FPGA

2 In Progress

challenges of ML

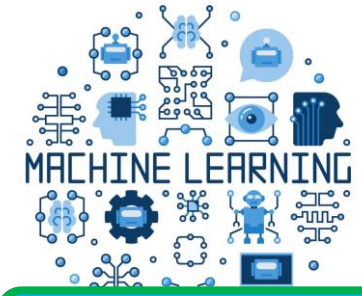
Roles & competencies

- *Data Physicist*
- *System Engineering team*
- *ML Engineer*
- *Software Engineer*
- *Hardware Engineer*
- *Infra & Security teams*

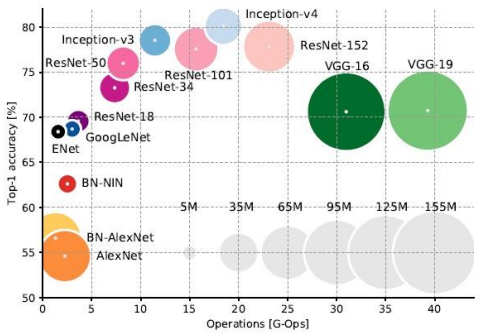
Tools

- *ML Tools:*
  - *TF-KERAS, PyTorch …*
- *HLS4ML (Xilinx…)*
- *HLS*
- *Brevitas & FiNN(Xilinx)*
- *CONIFER (LLR)*
- *N2D2 (CEA)*
- *VHDL*

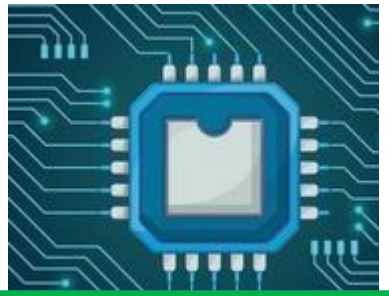
…



Artefacs & ML zoology



- *Model*
- *Code source…*



Digital hardware technologies

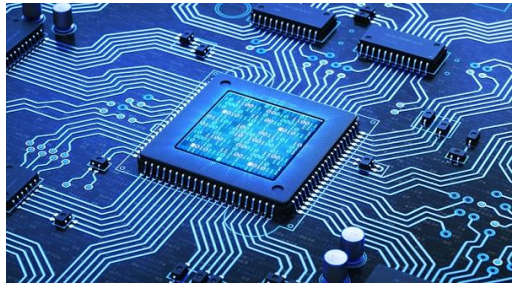
- *CPU*
- *FPGA SOM*
- *SNN*
- *MPPA*
- *GPU*
- …

Deployement & Operational AI

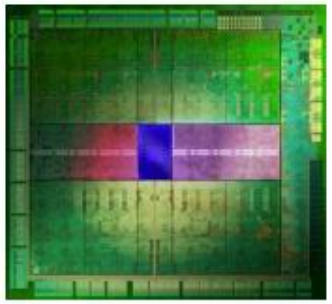
- *GitLab/Git*
- *Training Service skew*
- *Model Monitoring*
- *Responsible AI*
- …

Hardware architectures vs digital hardware engineer

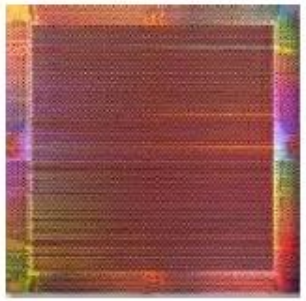
MPPA



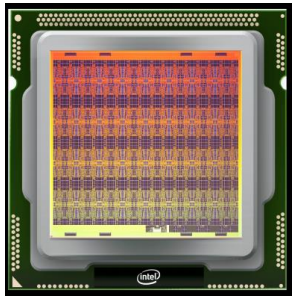
ASICs



GPUs



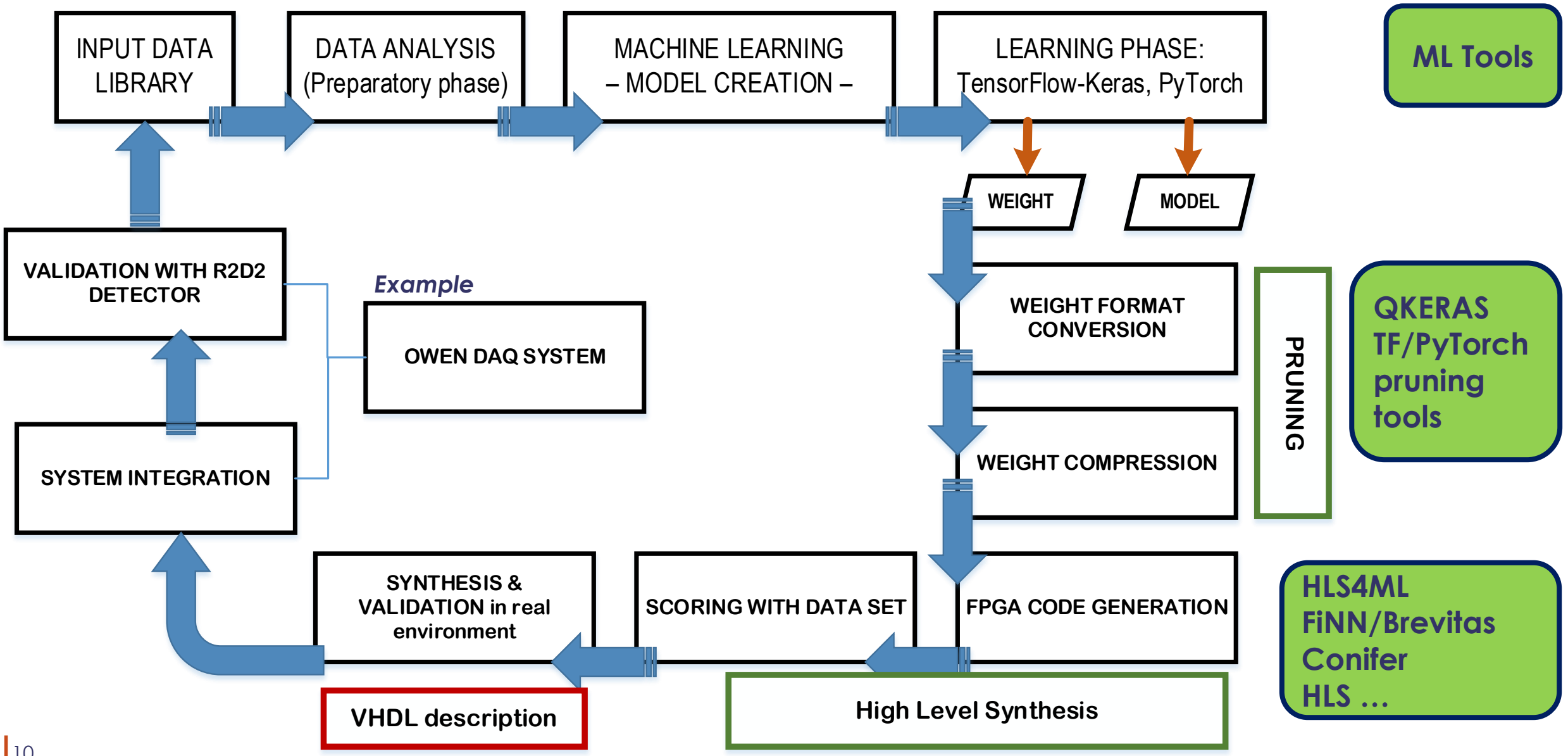
FPGAs



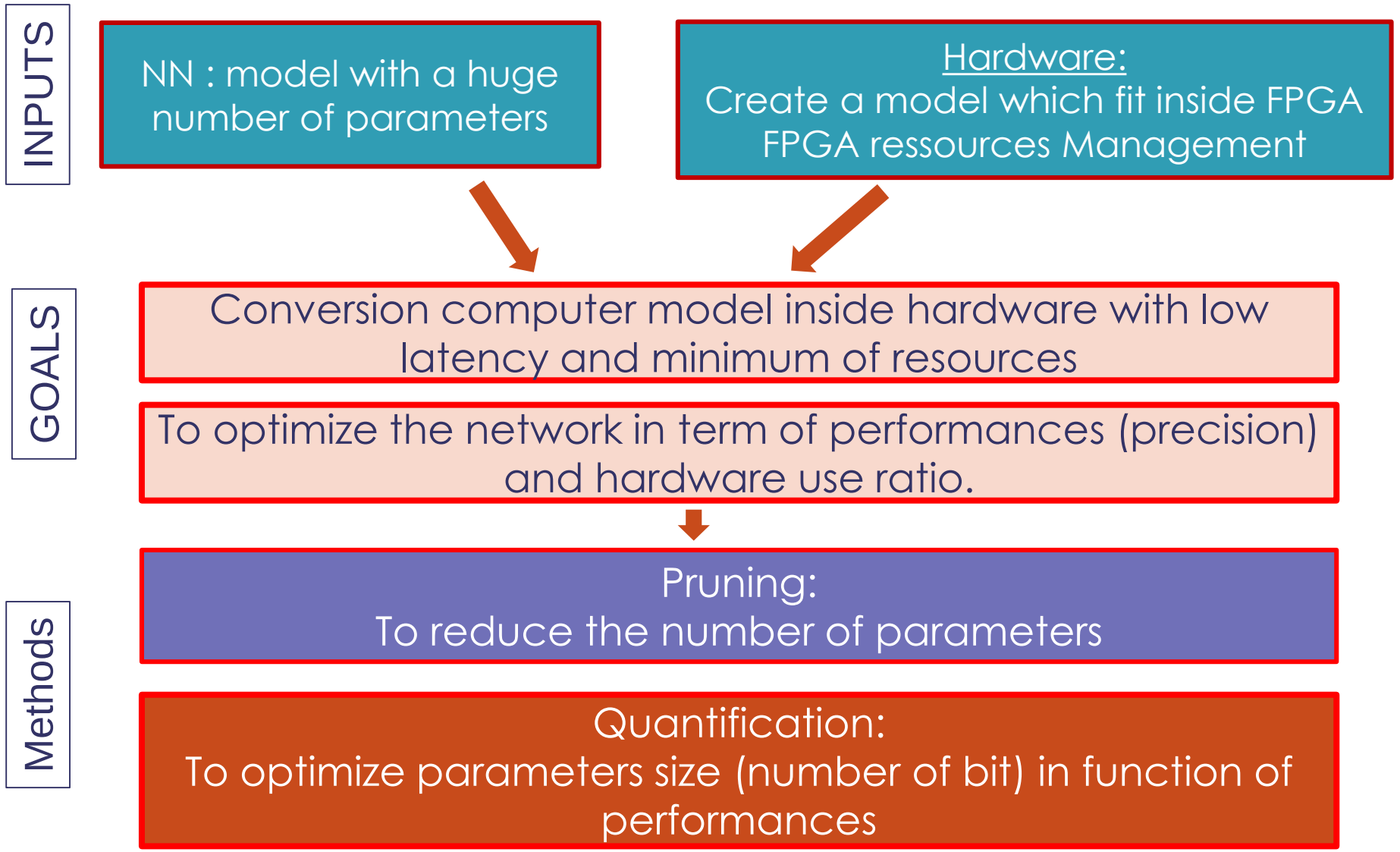
NMC

- **Designing embedded AI systems requires:**
  - *A knowledge of classic AI*
  - *An excellent understanding of hardware architectures*
  - *Specialization by hardware solution*
    - Resource Usage
    - Tools
    - Embedded functions
    - Use of network optimization tools

Methodology of design



Embedded approach: a question of optimization



R&T IN2P3 THINK

Testing Hardware Instantiations of Neural Kernels

Objectives: Test of Hardware Inferring Neural networK

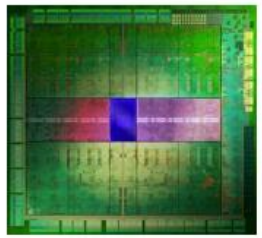
Jean-Pierre Cachemiche, Monnier Emmanuel, George Aad, Thomas Calvet, Arthur Ducheix, Etienne Fortin, **CPPM**,
Frédéric Magniette, **LLR**
Joana Fronteras-Pons, **IRFU/AIM**
Frédéric Druillole, Abdelkader Rebii, Raphael Bouet, **LP2IB**
David Etasse, **LPC**
Vladimir Gligorov, Le Dortz Olivier, **LPNHE**
Fatih Bellachia, Lafrasse Sylvain, **LAPP**
Claude Girerd, **LP2IL**

2020 - 2023

THINK Technologies Selection

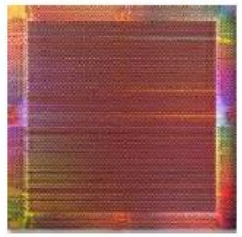
CPUs

MPPA



GPUs

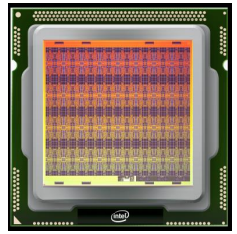
nVidia



FPGAs

DE10-Agilex

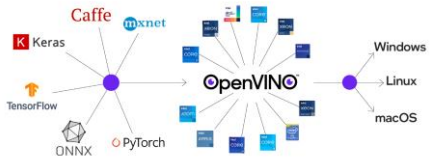
ZCU102, ZCU104

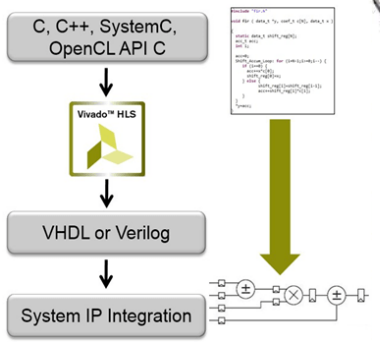


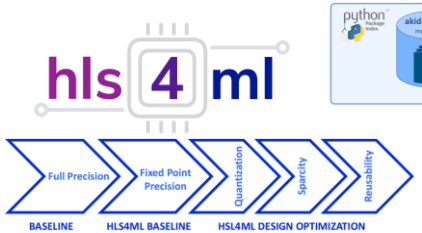
NMC

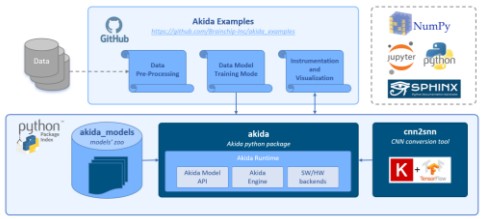
VCK190

https://think.in2p3.fr/



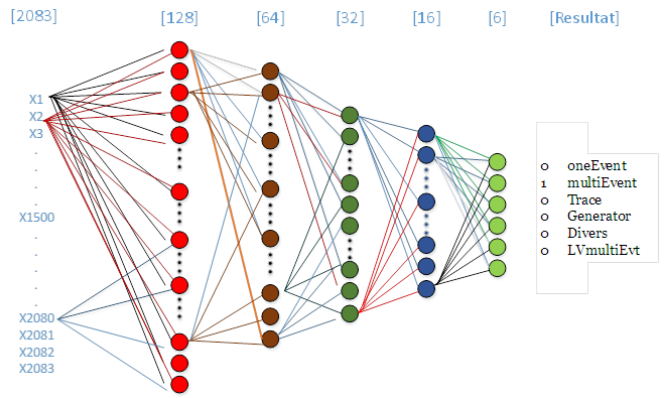






Example: Zynq Soc and HLS4ML

Keras/QKeras training phase
Save model weights



VIVADO PRJ
VIVADO ML
VITIS ML

Performances analysis

Weight, bias and activation Coefficients analysis

Pruning:
Reduce the size of the NN

HLS4ML Configuration:
{ Model, Layers, implementation}

NN python to HLS

HLS to VHDL:
VIVADO – Pipeline/Unroll –

qTools: Energy consumption analysis

Keras/ QKeras

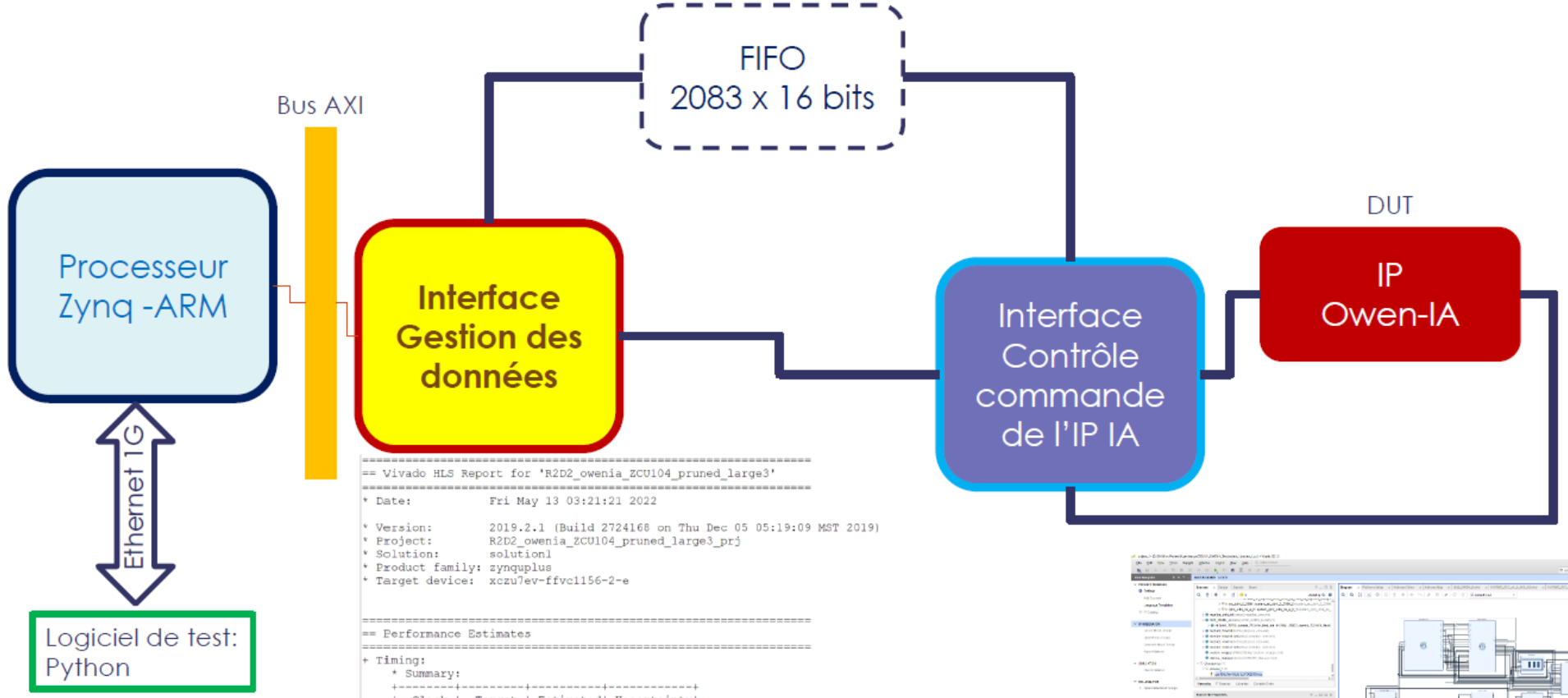
Tensorflow_model_ optimization

HLS4ML framework

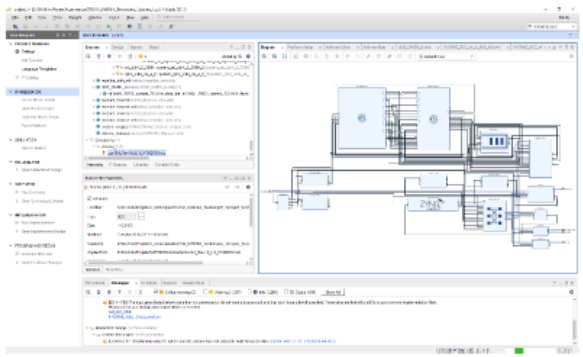
VIVADO HLS

Qtools & QKeras

Owen Test in real condition

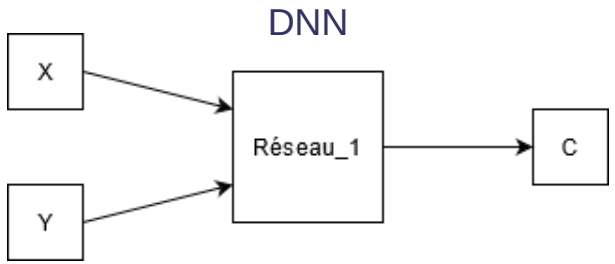


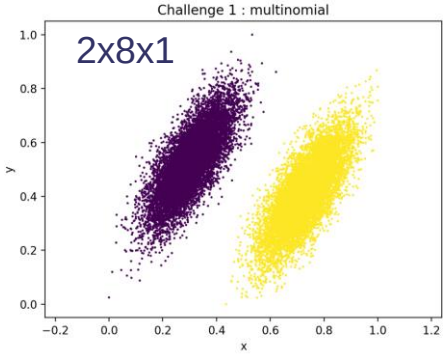

```
================================================================
== Vivado HLS Report for 'R2D2_owenia_ZCU104_pruned_large3'
================================================================
* Date:           Fri May 13 03:21:21 2022

* Version:        2019.2.1 (Build 2724168 on Thu Dec 05 05:19:09 MST 2019)
* Project:        R2D2_owenia_ZCU104_pruned_large3_prj
* Solution:       solution1
* Product family: zynquplus
* Target device:  xczu7ev-ffvc1156-2-e


================================================================
== Performance Estimates
================================================================
+ Timing:
    * Summary:
    +--------+---------+----------+------------+
    |  Clock |  Target | Estimated| Uncertainty|
    +--------+---------+----------+------------+
    |ap_clk  | 5.00 ns | 4.339 ns |   0.62 ns  |
    +--------+---------+----------+------------+

+ Latency:
    * Summary:
    +---------+---------+-----------+-----------+------+------+----------+
    | Latency (cycles)  |   Latency (absolute)  |   Interval  | Pipeline |
    |   min   |   max   |    min    |    max    |  min |  max |   Type   |
    +---------+---------+-----------+-----------+------+------+----------+
    |     2086|     2087| 10.430 us | 10.435 us |  2087|  2088| dataflow |
```

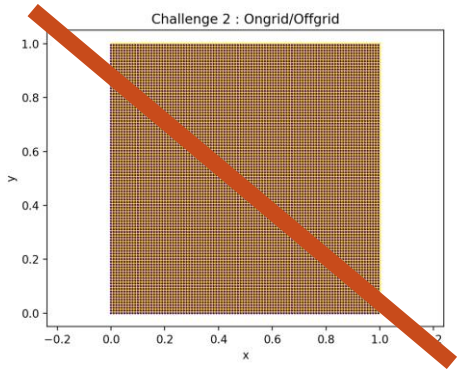


17

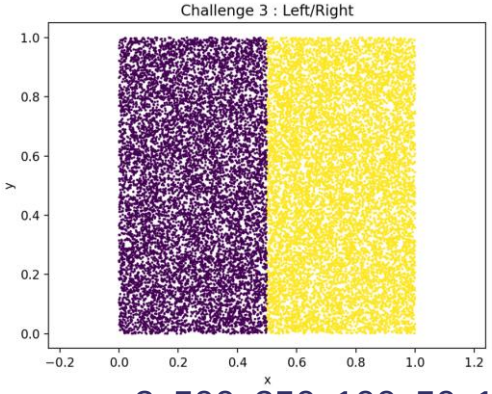
Solution Comparaison: AI Challenges (F. Magniette LLR)



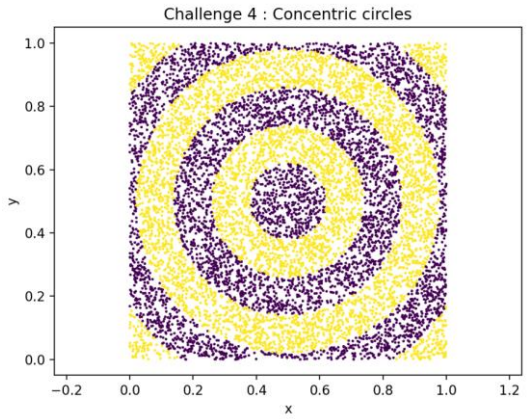




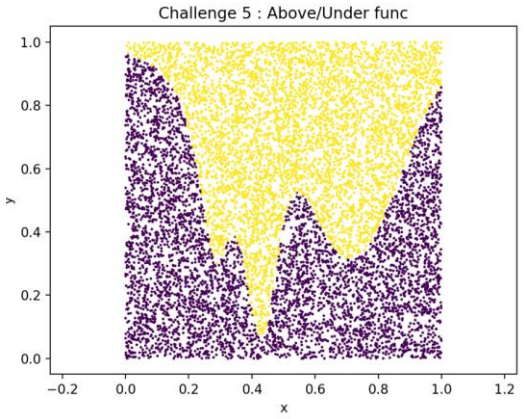
2x200x100x50x1



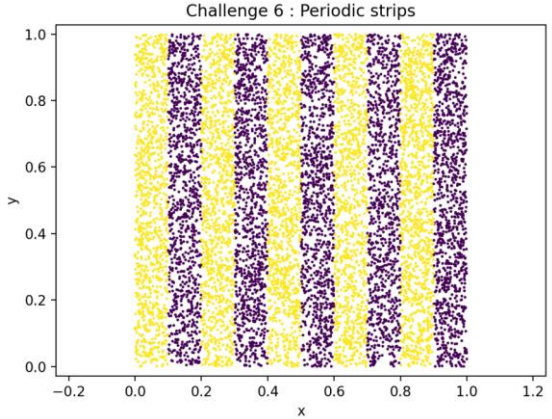
2x1000x500x250x100x50x1

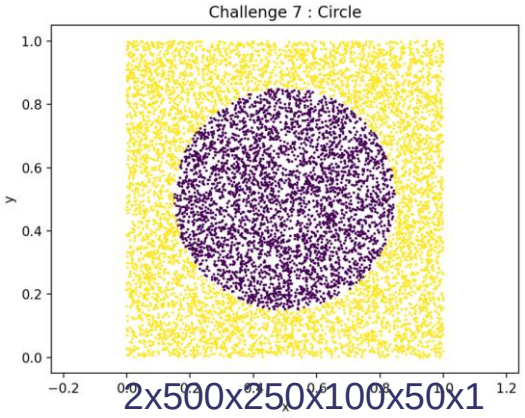


2x500x250x100x50x1

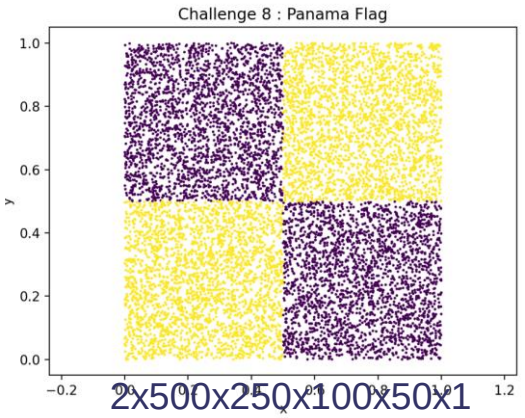


2x500x250x100x50x1





2x500x250x100x50x1



2x500x250x100x50x1

Discussion about FPGA: Spatial Accelerators

- **Zynq + HLS4ML: io_parallel / io_stream /Reusefactor / Resource / Latency**

	ARTY-Z7 CH1	ARTY Z7 CH3	ARTY Z7 CH3 Optimisé	ZCU102 CH3	CH4	ZCU102 CH4 Qbit<16,2>	ZCU102 CH5	ZCU102 CH7 Optim Ressource	ZCU102 CH7 Optim Latence	ARTY Z7 CH7 Optim HLS Stream
Nombre de cellules	**16**	**25801**	**25801**	**25801**	**658951**	**658951**	**156951**	**156951**	**156951**	**156951**
Horloge de reference	4,166ns	9,408ns	9,408ns	4,396ns		4,369ns	4,369ns	4,369ns	4,028ns	9,410ns
Temps de latence	70ns	19,804us	19,804us	2,510us		5,510us	5,510us	10,010us	10,015us	141us
BRAM	0%	41%	8%	6%		36%	18%	9%	15%	72%
DSP48E	21%	115%	11%	10%		59%	59%	12%	24%	6%
FF	2%	85%	28%	10%		28%	22%	32%	53%	167%
LUT	1%	280%	68%	46%		103%	113%	81%	104%	423%
URAM	0%	0%	0%	0%		0%	0%	0%	0%	0%

depend on VITIS-HLS #pragma
The way HLS handles vector/matrix before DSP

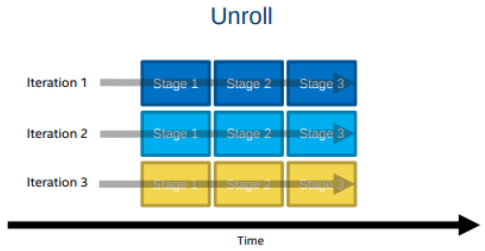
- **Intel Arria 10 + Intel HLS (expert HLS) :**

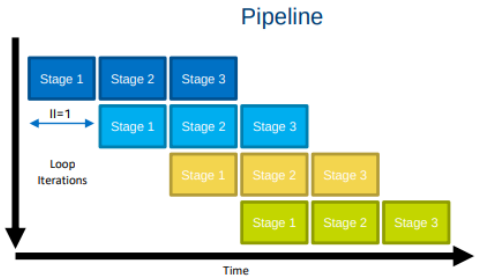
	ALUTs	FFs	RAMs	MLABs	DSPs
Ch1 vanilla	602 (0%)	547 (0%)	4 (0%)	2 (0%)	1.5 (0%)
Ch1 pipeline	610 (0%)	624 (0%)	4 (0%)	5 (0%)	1.5 (0%)
Ch1 unroll	515 (0%)	245 (0%)	4 (0%)	1 (0%)	0 (0%)
Ch1 u+p	515 (0%)	245 (0%)	4 (0%)	1 (0%)	0 (0%)

	ALUTs	FFs	RAMs	MLABs	DSPs
Ch3 vanilla	1 408 (0%)	1 809 (0%)	30 (1%)	12 (0%)	2.5 (0%)
Ch3 pipeline	2 093 (0%)	4 460 (0%)	32 (1%)	48 (0%)	2.5 (0%)
Ch3 unroll	118 041 (14%)	36 737 (2%)	5 (0%)	37 (0%)	0 (0%)
Ch3 u+p	27 524 (3%)	42 546 (2%)	1 855 (68%)	298 (1%)	75 (5%)

	ALUTs	FFs	RAMs	MLABs	DSPs
Ch4 vanilla	4 442 (0%)	6 415 (0%)	355 (1%)	20 (0%)	3.5 (0%)
Ch4 pipeline	5 555 (0%)	10 899 (0%)	362 (1%)	113 (0%)	3.5 (0%)
Ch4 unroll	Problème d'implémentation				
Ch4 u+p					

	ALUTs	FFs	RAMs	MLABs	DSPs
Ch5 vanilla	1 669 (0%)	2 193 (0%)	32 (1%)	16 (0%)	2.5 (0%)
Ch5 pipeline	2 617 (0%)	6 074 (0%)	35 (1%)	76 (0%)	2.5 (0%)
Ch5 unroll	569 259 (67%)	196 051 (11%)	6 (0%)	40 (0%)	0 (0%)
Ch5 u+p	17 560 (2%)	23 244 (1%)	507 (19%)	237 (1%)	50.5 (3%)





Question of HLS optimisation

R&D to come

Fully Firmware = spatial accelerator

- FPGA/SOM OK (optimisation Pb)
- SNN: not mature → to continue to investigate (ASIC)
- Build a VHDL Librairies without HLS (FPGA Optimisation)
- Reinforcement learning to create data quality

Edge Computing

- Hardware OK (CPU, GPU, FPGA (SoC)
- To test versatil NN model
- New architecture to test like VERSAL/STRATIX

Moke-up of an optimized instrument

- Teaching, people trainee
- Test new hardware, digital twin model
- Mixed model (1DRNN+1DCNN)
- Application to LHC
- Application to select rare events
- Develop/Select Embedded AI Framework

→ Embedded ML Technologies depends on Data Quality (simulation/emulation/data mainframe)