Hardware acceleration with FPGAs

J.-B. Sauvan

LLR CNRS / École polytechnique

GT09 Town Hall Meeting: Calcul, Algorithmes et Données – 18/10/2019

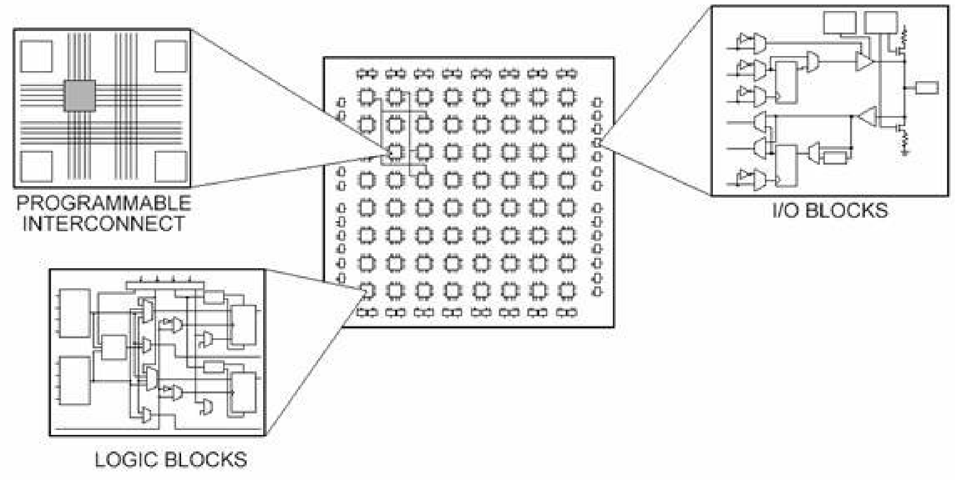
Introduction

- First FPGA in the beginning of the 80s
  - Constant growth since then
- Market dominated by Xilinx and Altera (acquired by Intel in 2015)
- FPGAs used mainly for
  - Telecom
  - Industry, Automotive, consumer electronics
- Using FPGA for hardware acceleration is a rather recent trend
  - Started between 2010-2015
- Disclaimers
  - I am neither a FPGA expert nor a computing expert, I am a physicist working with FPGAs for the (future) CMS trigger (for HL-LHC)
  - Xilinx FPGAs and tools are traditionally used in CMS, so my examples will be Xilinx products mainly (clearly I'm not covering everything related to FPGAs)

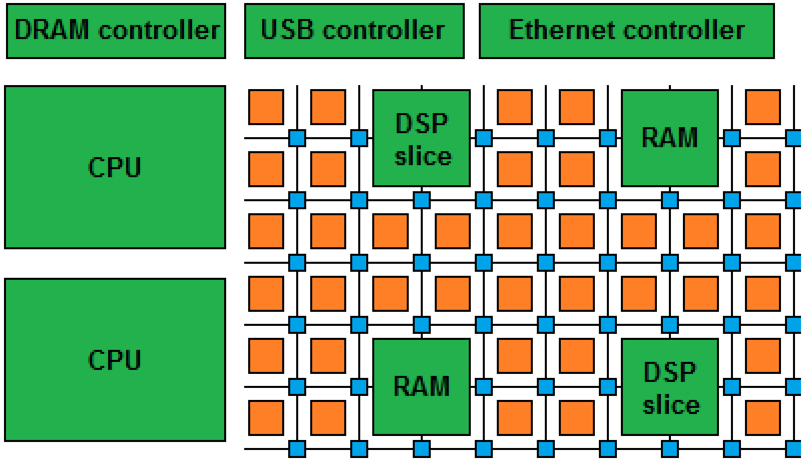
FPGA: a quick introduction

- FPGA = configurable logic
  - ○ Logic blocks
  - ○ Configurable interconnections
  - ○ I/O blocks
- Logic implemented in Lookup Tables (LUT)
- Additional embedded hardware
  - ○ RAM
  - ○ Multiplier-accumulator (DSP)
  - ○ CPU cores, controllers, etc.
- Important notes
  - ○ There is usually no floating point dedicated hardware in FPGAs
  - ○ The logic utilization efficiency of a FPGA is at most 75-80%

Main components of a FPGA



Additional hardware available



Modern FPGA: lots of hard, not-field-programmable gates

FPGAs in HEP

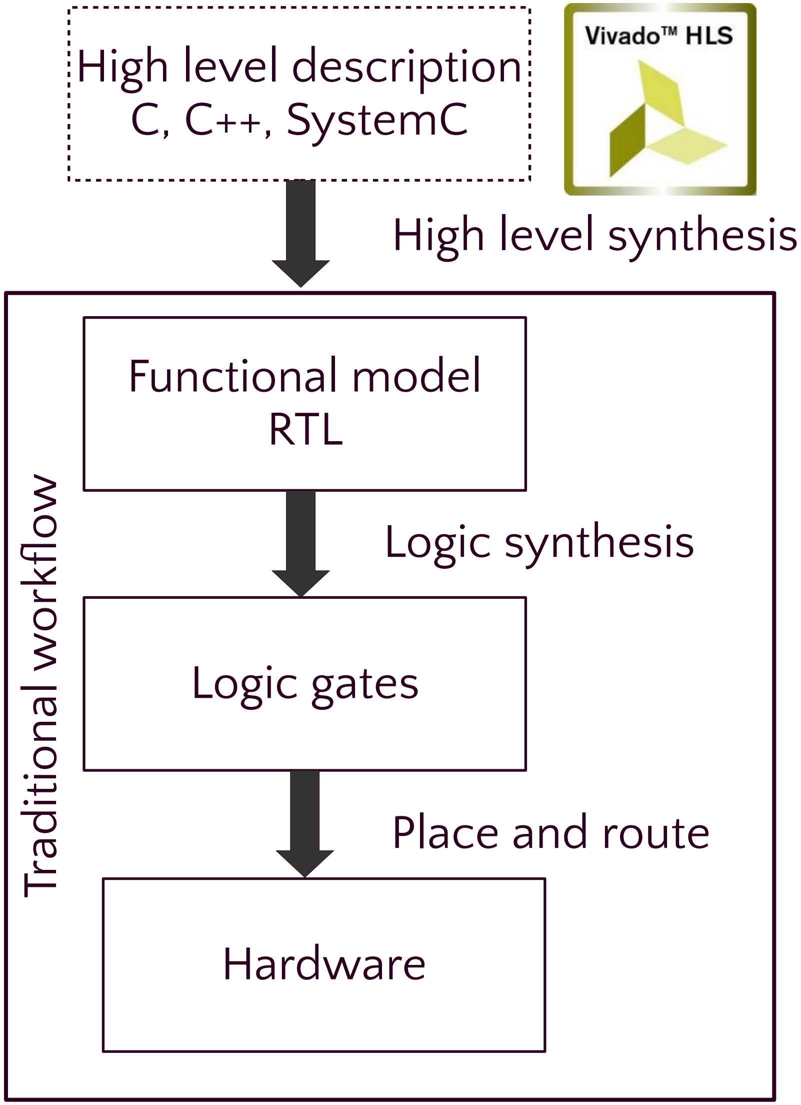
- FPGAs are traditionally used for trigger and DAQ in HEP since a long time
- Dedicated boards with large data throughput on optical fibers
  - e.g. CMS MP7 board, with almost 1 Tb/s in and 1Tb/s out
- Based on Telecom standards (MicroTCA, AdvancedTCA)

MP7 trigger board

MicroTCA crate filled with MP7 boards

FPGA (and ASIC) firmware development

- Traditional development
  - Code in Hardware Description Languages (e.g. VHDL, Verilog)
  - Logic synthesis tool to produce gate level description
  - Physical implementation with a placement and rooting tool
- High Level Synthesis
  - Add one level of abstraction (C, C++, etc.)
  - e.g. Xilinx Vivado HLS since 2012
  - Pragma directives to control the implementation
- HLS becomes more and more efficient
  - Though still useful to have an hardware view of the code



FPGA acceleration boards

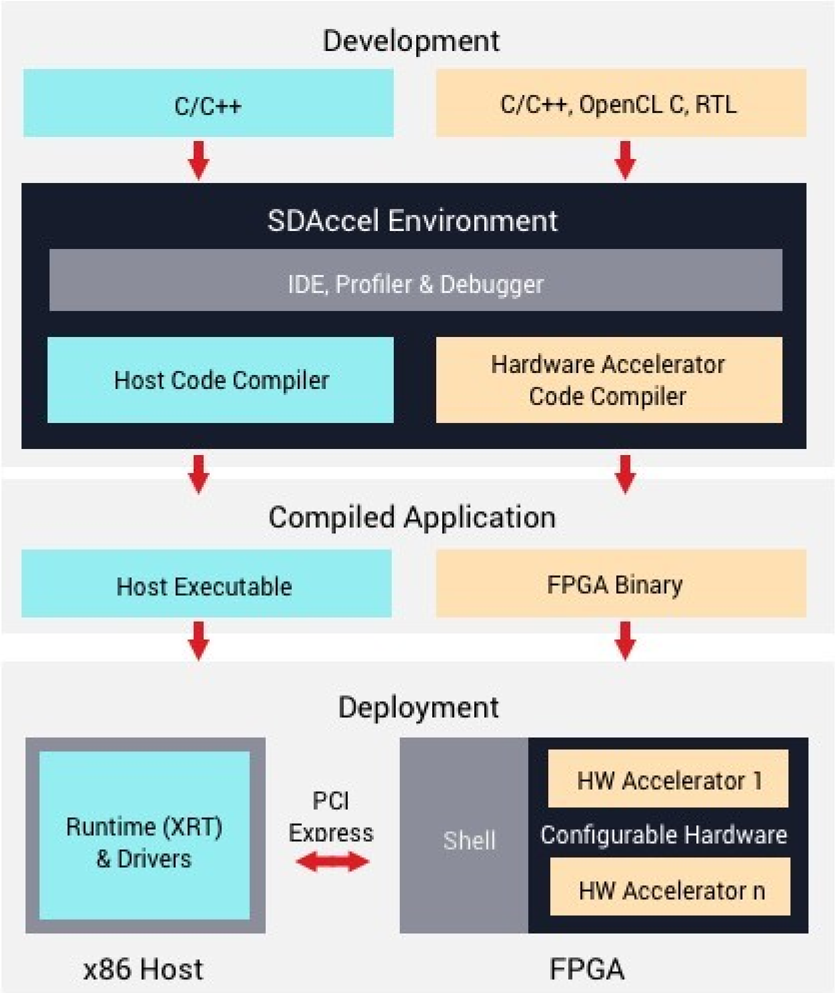
- One year ago Xilinx announced FPGA-based acceleration boards (Alveo)
  - PCIe form factor, single or dual slot
- Composed of (this is for one version of the board)
  - Programmable logic (Virtex Ultrascale+ architecture)
  - 32 GB DDR4
  - 8 GB HBM2 (3d-stacked High Bandwidth Memory)
  - 40-50 MB internal SRAM
- Programmable memory hierarchy
  - Useful for DNNs, which can have diverse memory requirements

Memory types and bandwidths

DDR4	32 GB	32 GB/s
HBM2	8GB	460 GB/s
Internal SRAM	40-50 MB	35 TB/s

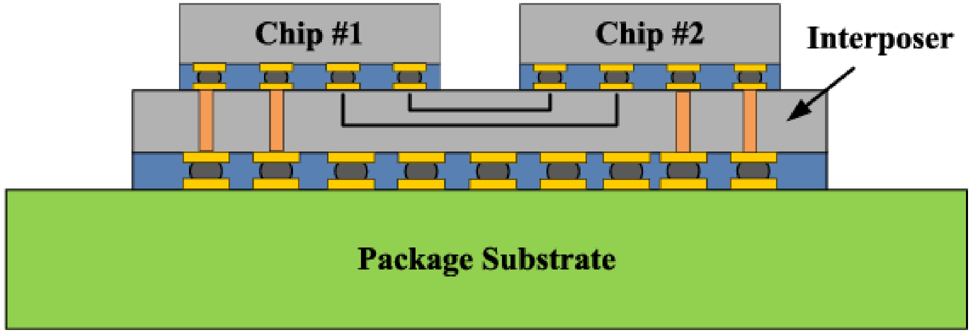
Accelerated applications development

- Dedicated tool to develop applications for Alveo boards
  - SDAccel
- High Level languages
  - Host application in C/C++
    - Built using GCC
  - FPGA-accelerated functions in HDL, C/C++, or OpenCL
    - Built using Vivado HLS + Vivado as backend

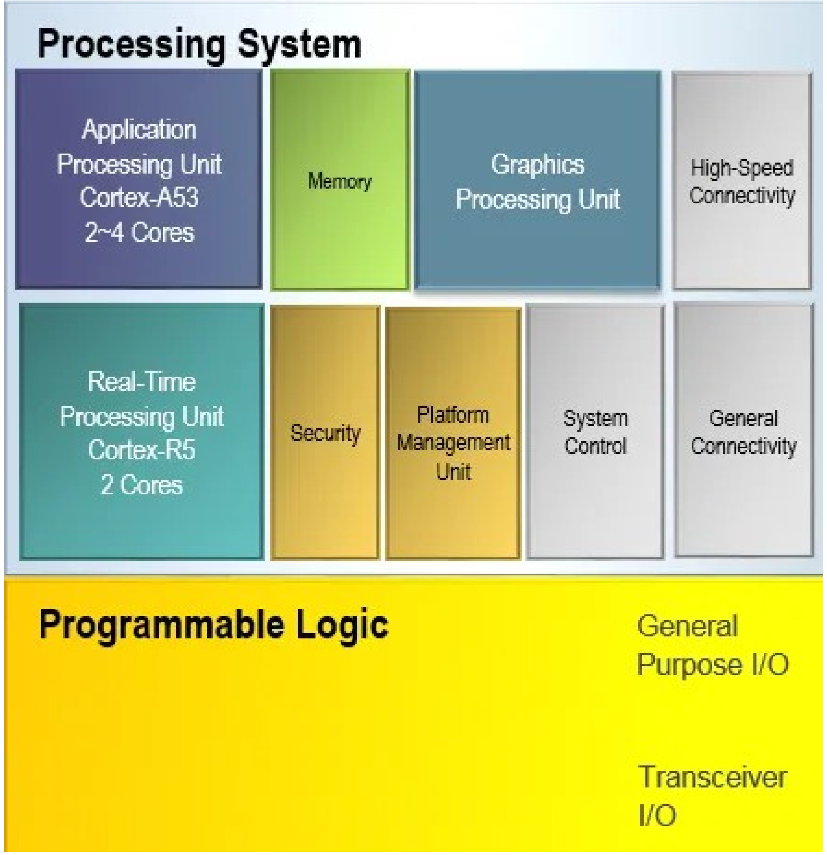


Multi-processor System-on-chip (MPSoC)

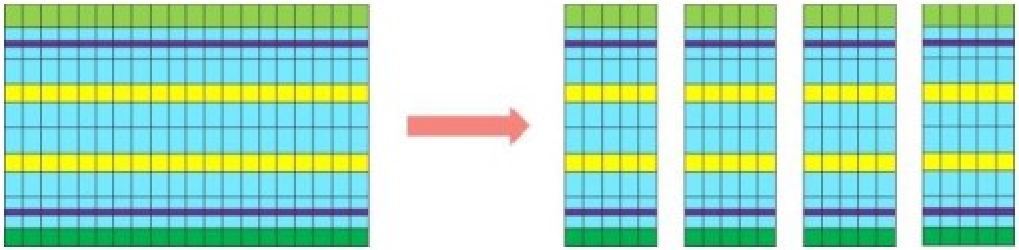
- Chips now use 3d technologies
  - ○ e.g. passive interposers with fast buses
- Interconnects FPGAs, memory, CPU, etc.
- Allows extremely fast data transfer
  - ○ e.g. "High Bandwidth Memory"



Multi-processor system

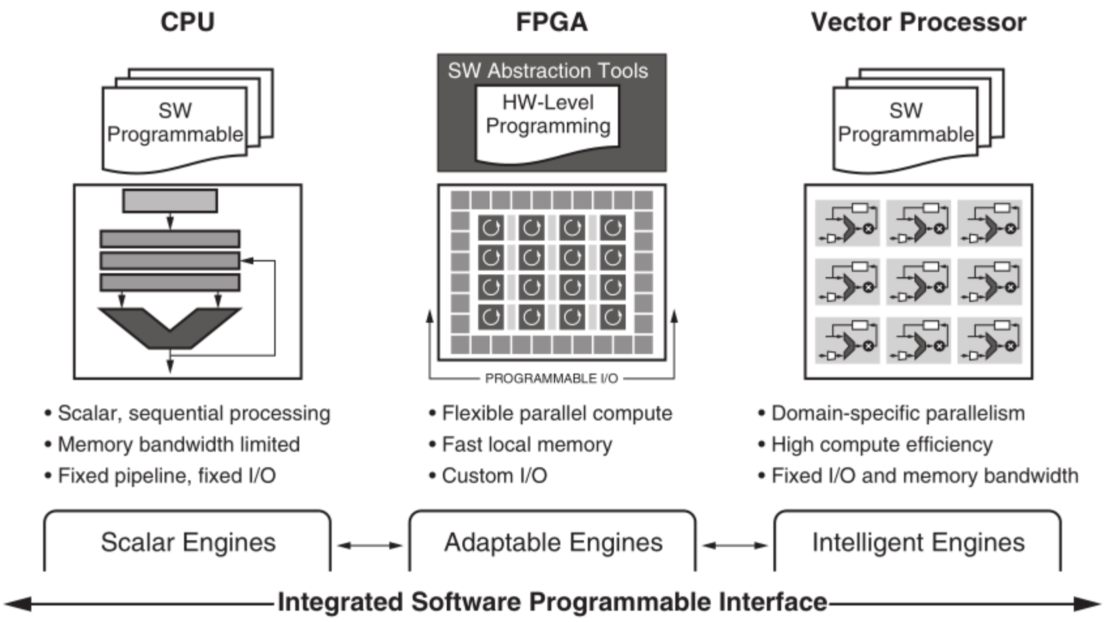


FPGA partitioning



Future acceleration platforms

- Adaptable logic (FPGA) combined with
  - ○ scalar processors (RISC)
  - ○ vector processors (both integer and floating point operations)
- Xiling Versal platform
  - ○ Announced 1 year ago
  - ○ 1st component available this year
- Integrated with dedicated software frameworks
  - ○ e.g. Vitis, announced a few weeks ago
- Interfaced with standard libraries
  - ○ e.g. for machine learning



CPU

SW Programmable

- Scalar, sequential processing
- Memory bandwidth limited
- Fixed pipeline, fixed I/O

FPGA

SW Abstraction Tools

HW-Level Programming

PROGRAMMABLE I/O

- Flexible parallel compute
- Fast local memory
- Custom I/O

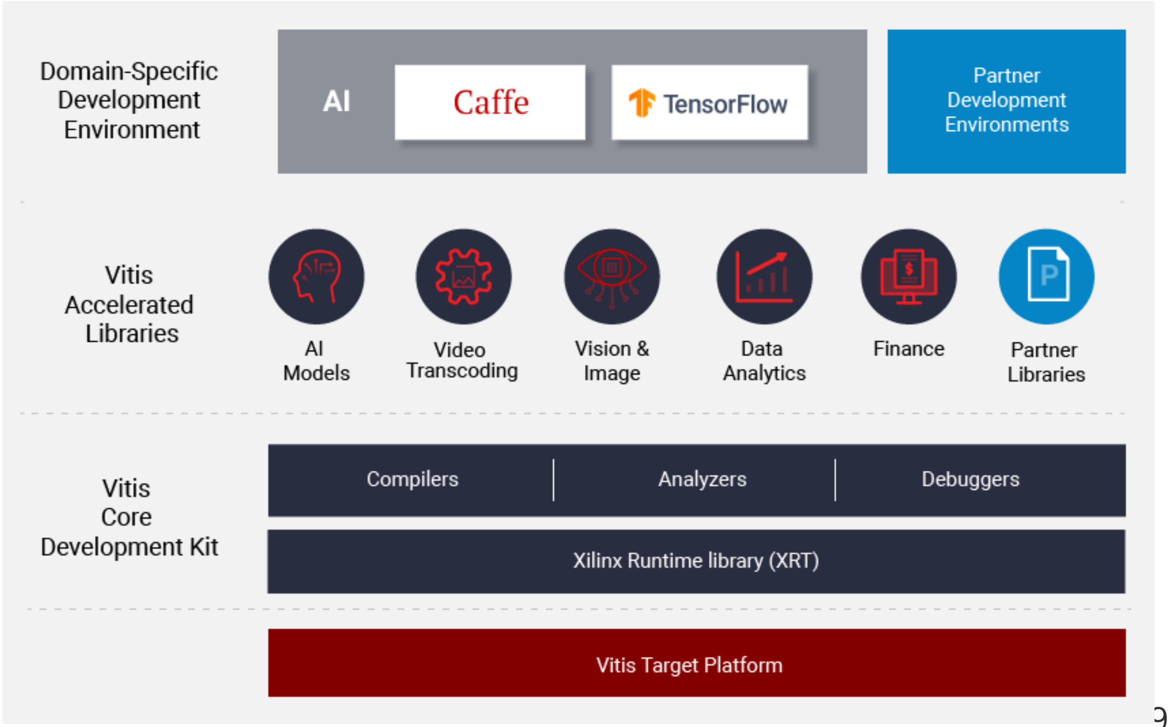
Vector Processor

SW Programmable

- Domain-specific parallelism
- High compute efficiency
- Fixed I/O and memory bandwidth

Scalar Engines ↔ Adaptable Engines ↔ Intelligent Engines

Integrated Software Programmable Interface



Domain-Specific Development Environment

AI

Caffe

TensorFlow

Partner Development Environments

Vitis Accelerated Libraries

AI Models

Video Transcoding

Vision & Image

Data Analytics

Finance

Partner Libraries

Vitis Core Development Kit

Compilers | Analyzers | Debuggers

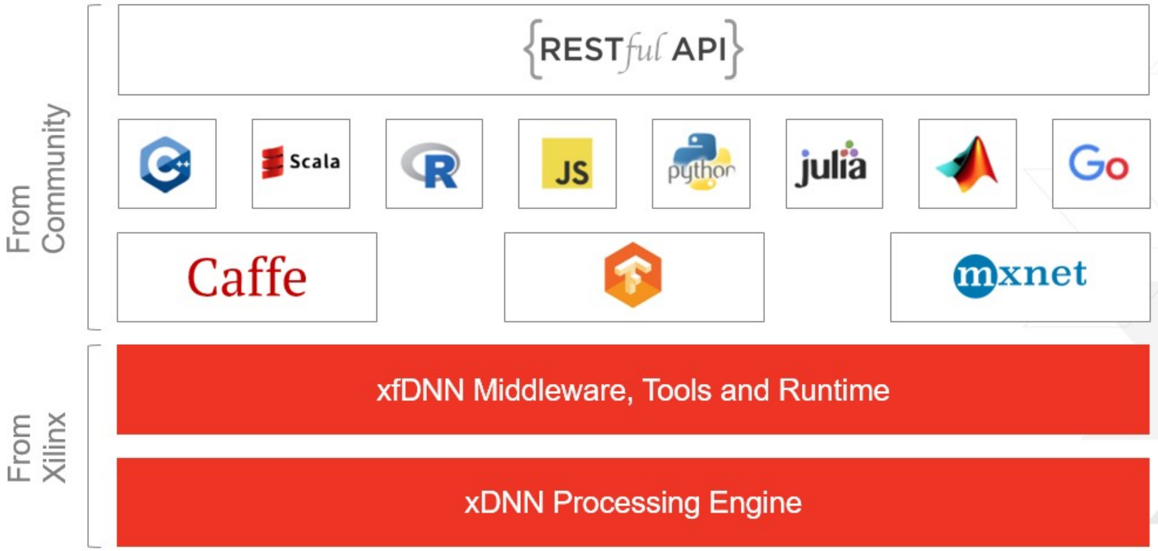
Xilinx Runtime library (XRT)

Vitis Target Platform

Applications

- Many compute-intensive tasks can be accelerated with programmable logic
  - Or with heterogeneous chips
- One particular target of FPGA companies is neural network inference
  - A lot of effort and money is being injected in the design of new hardware and software platforms
  - Partnerships with GAFAs et al. (e.g. Microsoft Azure, Amazon AWS, IBM)
- Also efforts in HEP (hls4ml, based on Xilinx Vivado HLS)

Xilinx MLSuite



Related developments at LLR

- Team of complementary people, with various expertise
  - Software and computing
  - Digital electronics and High Level Synthesis
  - Physics
- Growing set of hardware nodes used for R&D
  - CMS HGCAL platform
    - Server with 1 Xilinx Alveo U200 FPGA board
  - Labex P2IO ACP (Accelerated Computing for Physics) platform
    - Server with 2 Nvidia V100 GPU
    - Server with 1 Xilinx Alveo U280 FPGA board
    - Older GPUs
- Activities
  - Fast neural network inference for L1 triggers
  - Software platform development for neural network optimizations (" Innate")
  - Offline 3D object detection in HGCAL (Mask R-CNN)

Conclusion

- FPGAs exist since quite some time
- But they have been used for hardware acceleration only since recently
  - The interests in FPGA for acceleration is growing very fast
  - In particular for the acceleration of DNN inference
  - Things (hardware and software) are evolving almost on a daily basis
- Computing and FPGA experts are traditionally two sets of people with very different backgrounds
  - These two kinds of experts are there in IN2P3 labs
- Using FPGAs for hardware acceleration requires to build bridges between these people
- These experts may also need to adapt to significant changes of paradigm
  - Digital electronics experts need to acquire software expertise
  - Computing experts need (at least for the moment) to have some hardware knowledge