1st Mesoamerican Workshop on Reconfigurable X-ray Scientific Instrumentation for Cultural Heritage

HyperFPGA heterogeneous computing on MPSoC-FPGA at ICTP

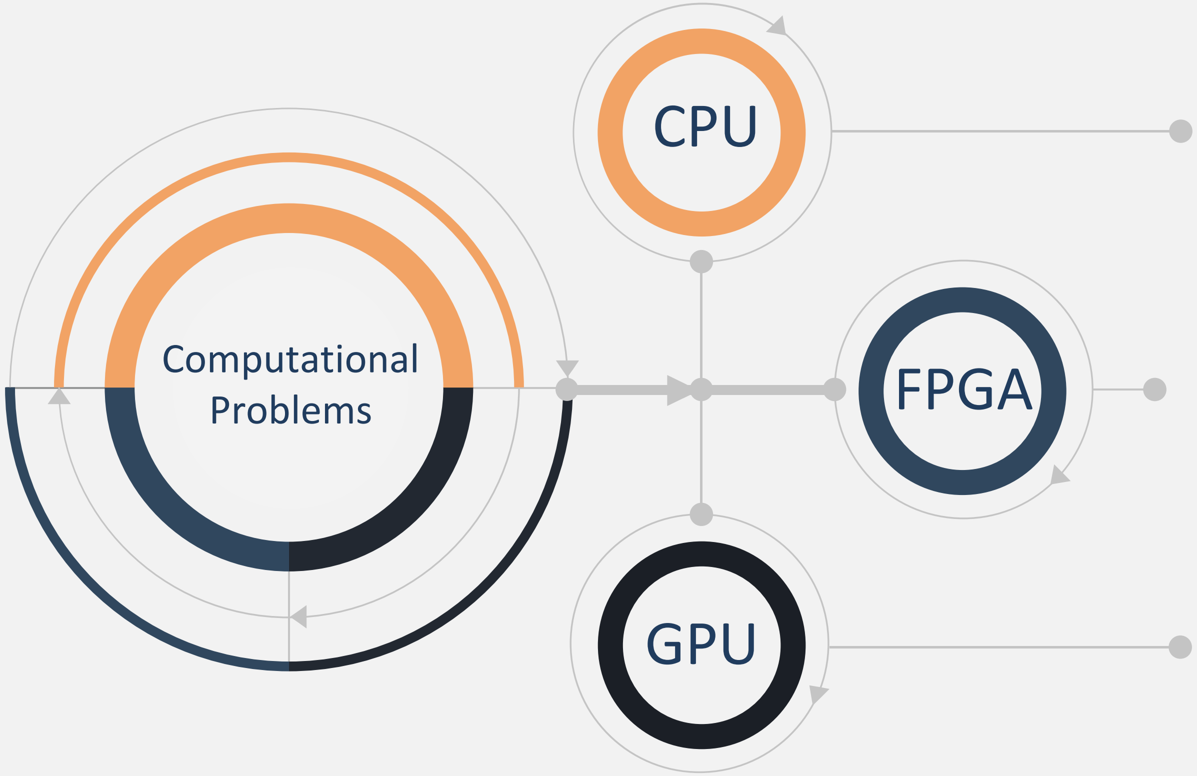
Antigua Guatemala, June 2025

Maynor Ballina

Introduction

- Heterogeneous computing
- Evolution of heterogeneous computing
- Distributed Computing
- Flynn Taxonomy
- Application of Heterogeneous Computing in Simulation
- The HyperFPGA cluster
- Infrastructure and management
- User Interface
- Jupyter Notebooks
- Broad spectrum of computational tasks
- Education and collaborative projects

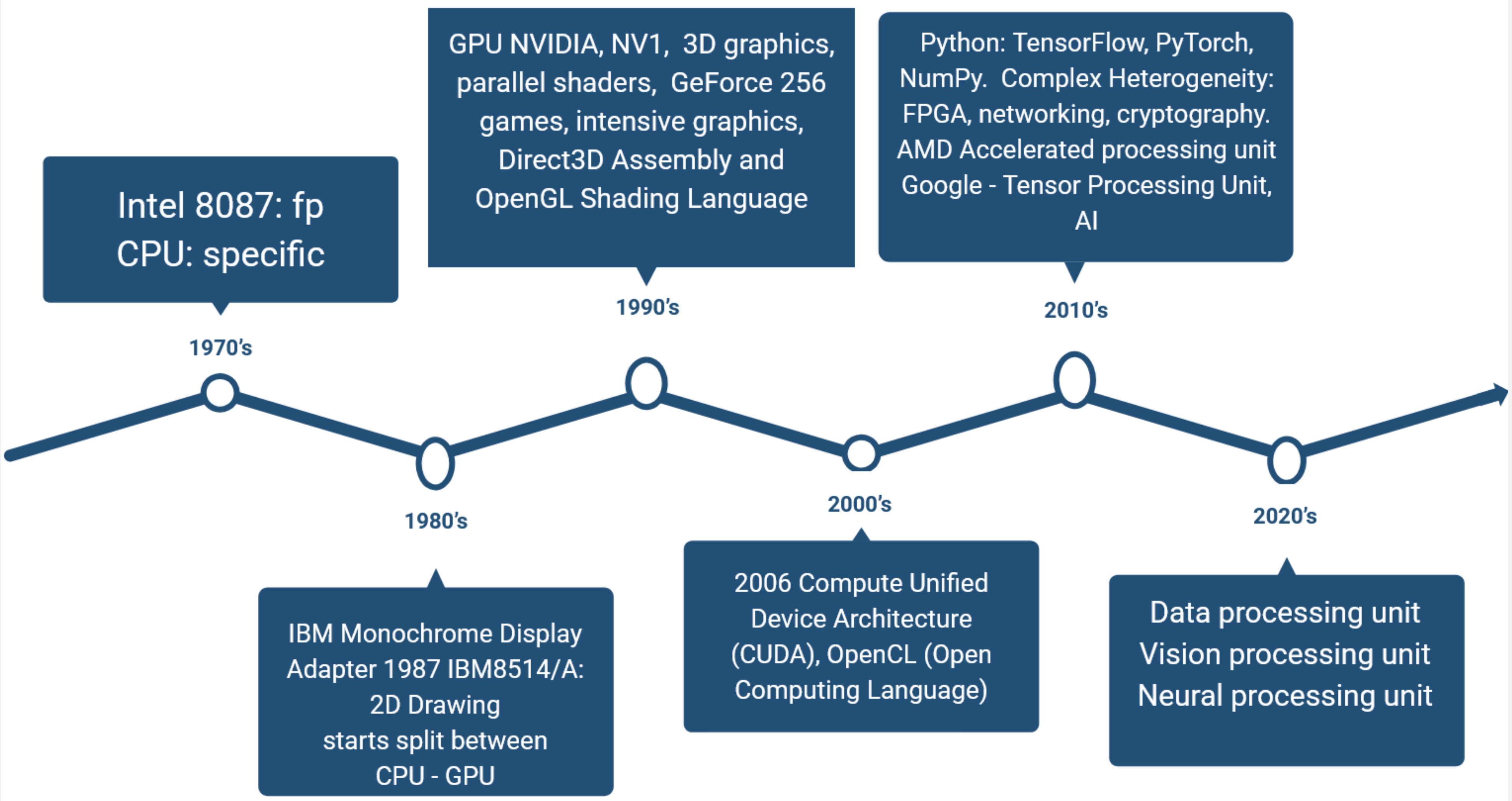
Heterogeneous Computing



It enhances computational performance by accelerating workloads such as graphics processing, artificial intelligence, and scientific simulations through specialized hardware.

The coordinated use of different types of processing cores within a single system to maximize performance and energy efficiency.

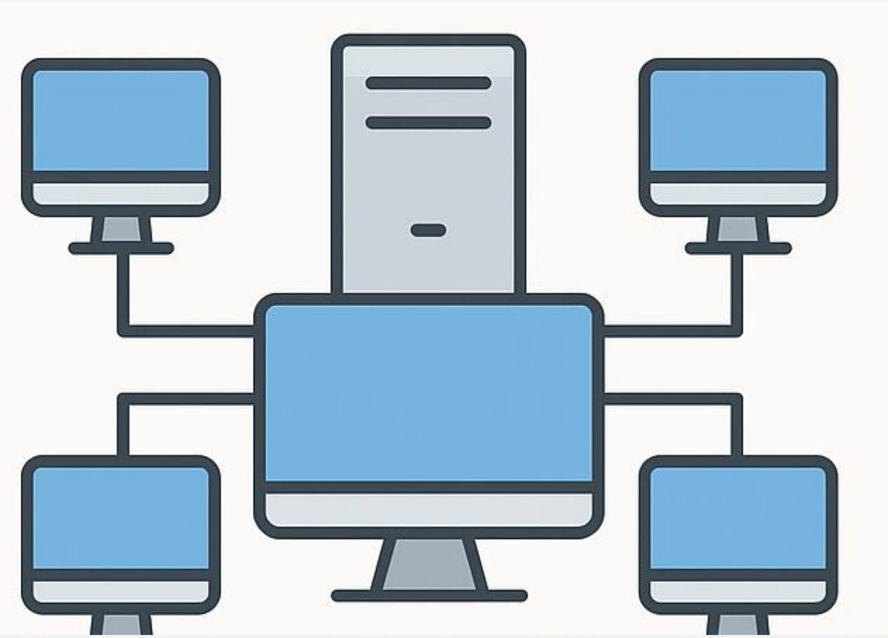
Evolution of Heterogeneous Computing



Distributed Computing

A model in which multiple nodes work together in a coordinated manner to solve a problem, sharing resources to improve performance, scalability, and fault tolerance.

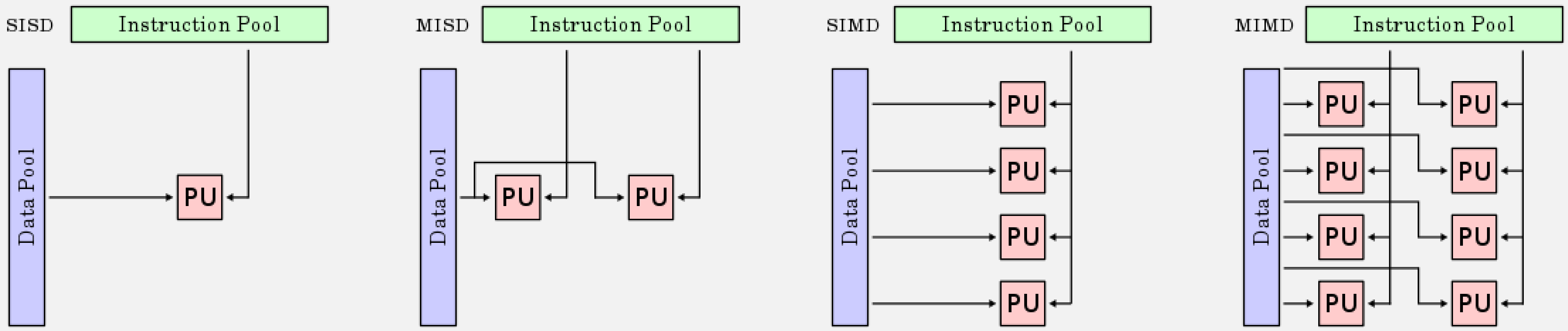
Distributed computing in clusters is a data processing approach that uses a set of interconnected nodes to work together as if they were a single machine.



Flynn's taxonomy

Classifies computer architectures based on the number of instruction streams and data streams.

•**SISD**: Single Instruction, Single Data – Conventional CPU
•**SIMD**: Single Instruction, Multiple Data – GPUs, instruction set extensions in CPUs
•**MISD**: Multiple Instructions, Single Data – Parallel processing, error detection
•**MIMD**: Multiple Instructions, Multiple Data – Multiprocessors, supercomputers



Application of Heterogeneous Computing in Simulation

Molecular Dynamics
CPU, GPU, FPGA

Fluid Dynamics
CPU, GPU, FPGA

Quantum Computational Simulation
CPU, GPU, TPU, FPGA

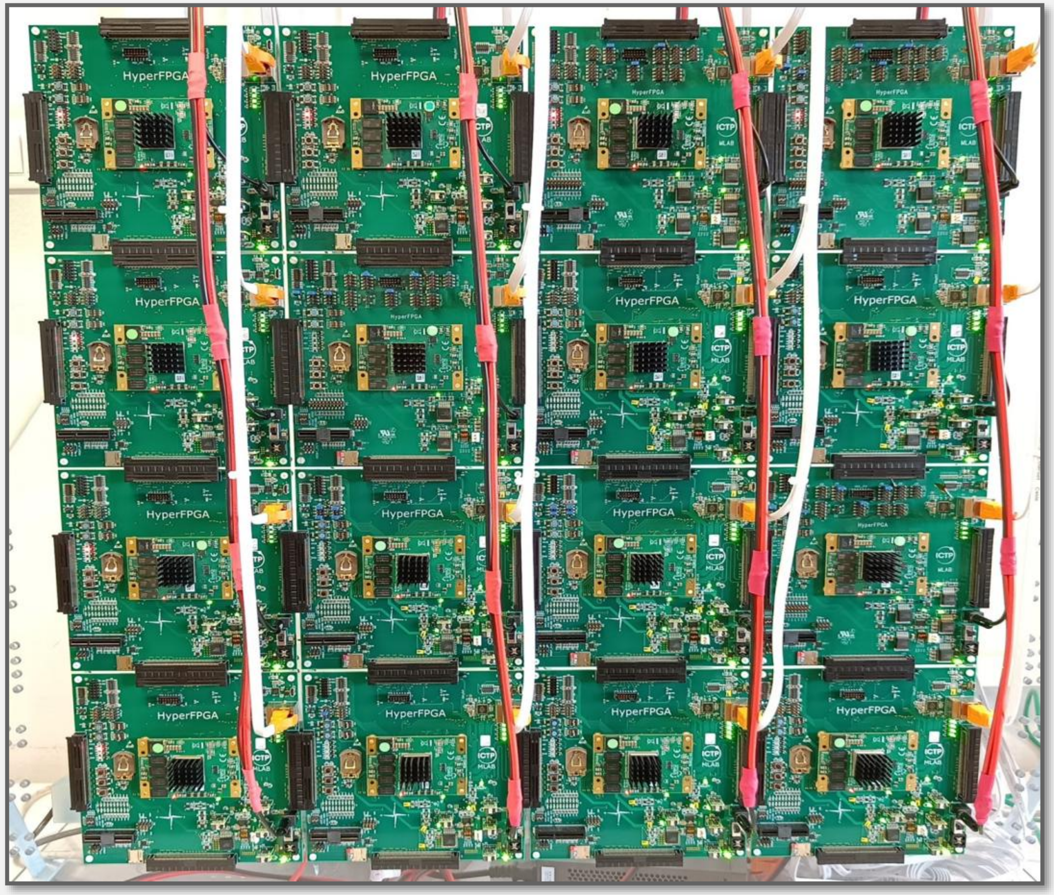
Cybersecurity
CPU, GPU, TPU, FPGA, DTU

Cosmological Models and Gravity Simulations
CPU, GPU, FPGA (Barnes-Hut)

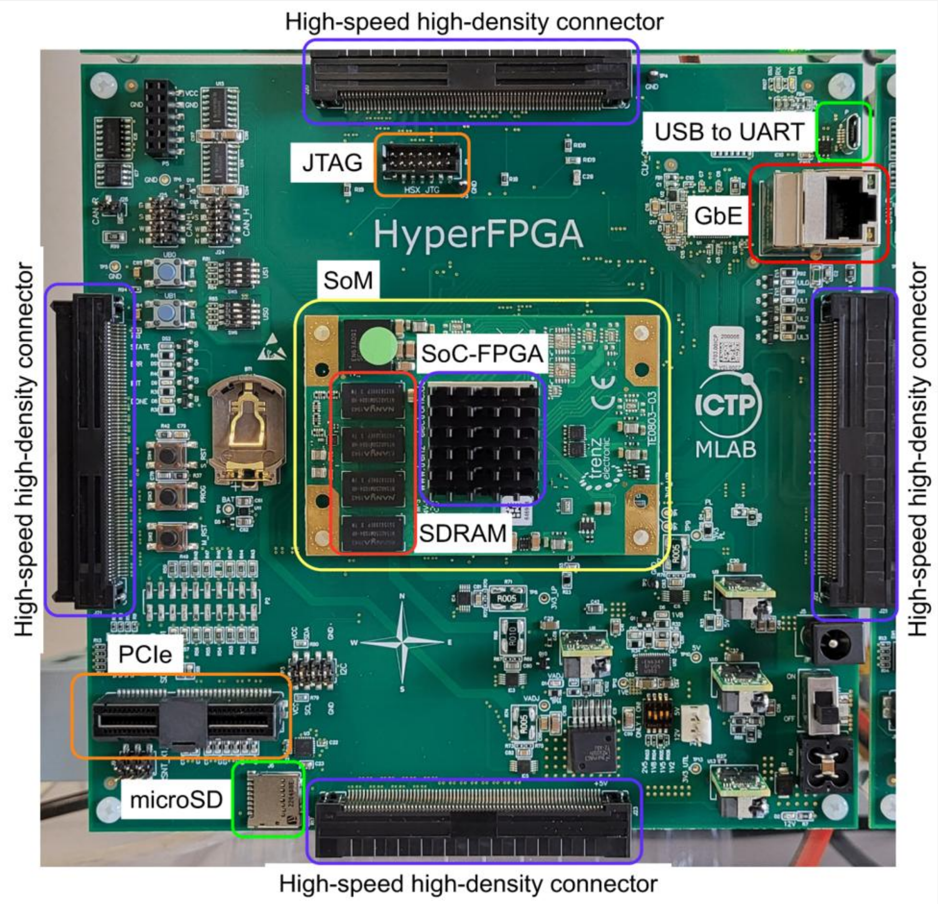
Montecarlo
CPU, GPU, FPGA

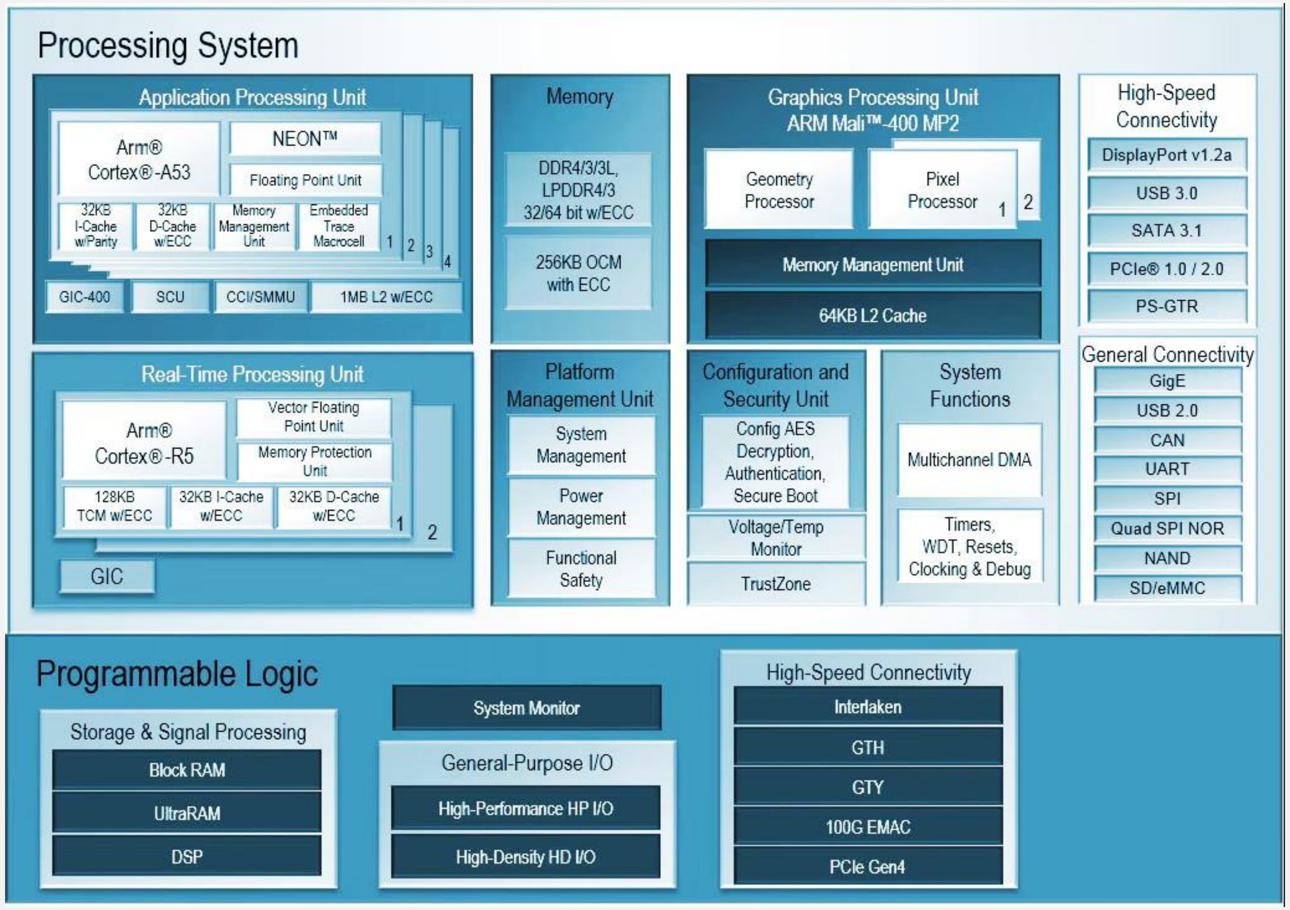
The HyperFPGA cluster

- MPSoC-FPGA-based experimental cluster.
- Zynq UltraScale+ MPSoC with APU, RPU, GPU.
- Fully connected, HP, HD, GTH, MIO, Ethernet.
- Debian OS



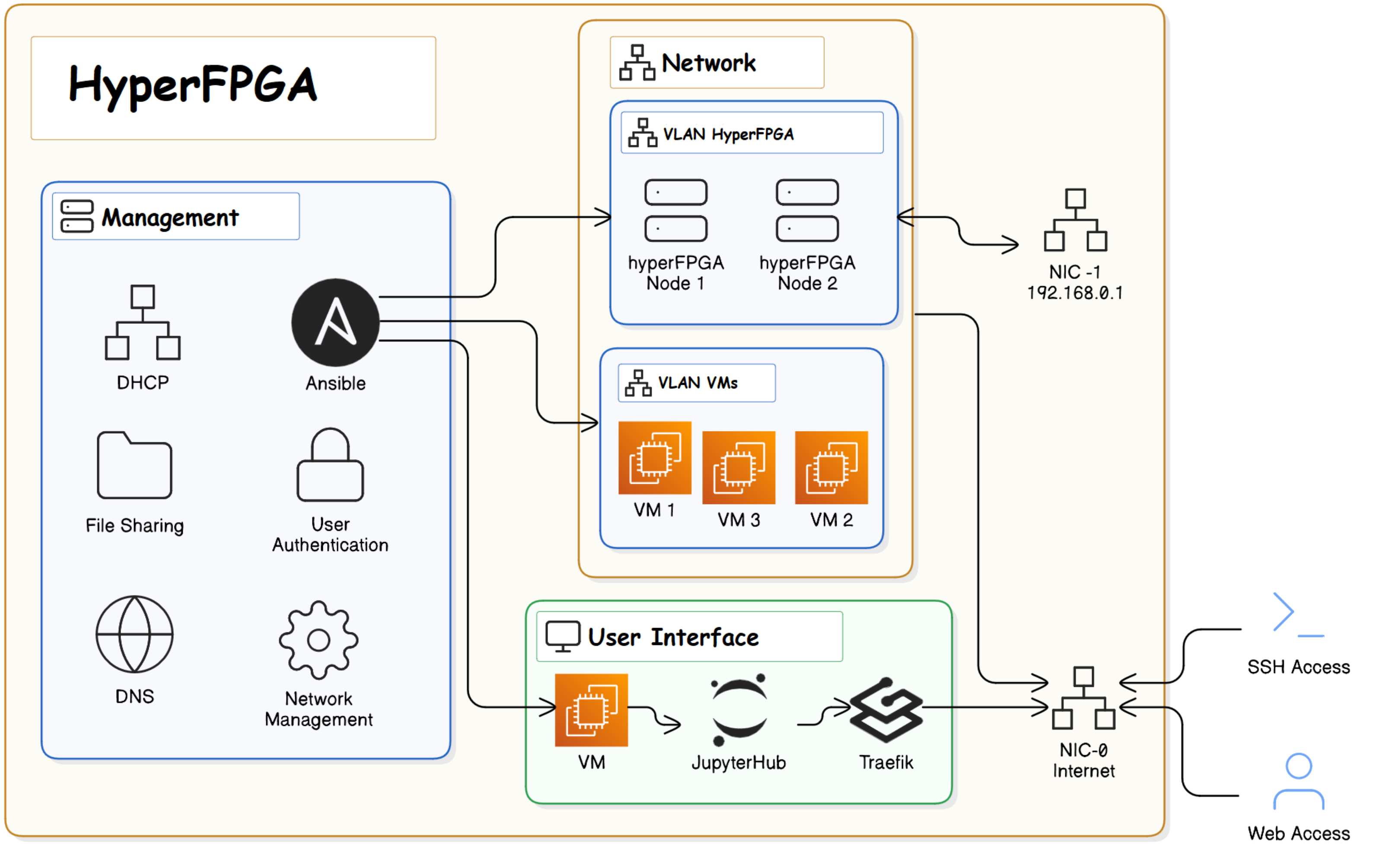
The HyperFPGA cluster





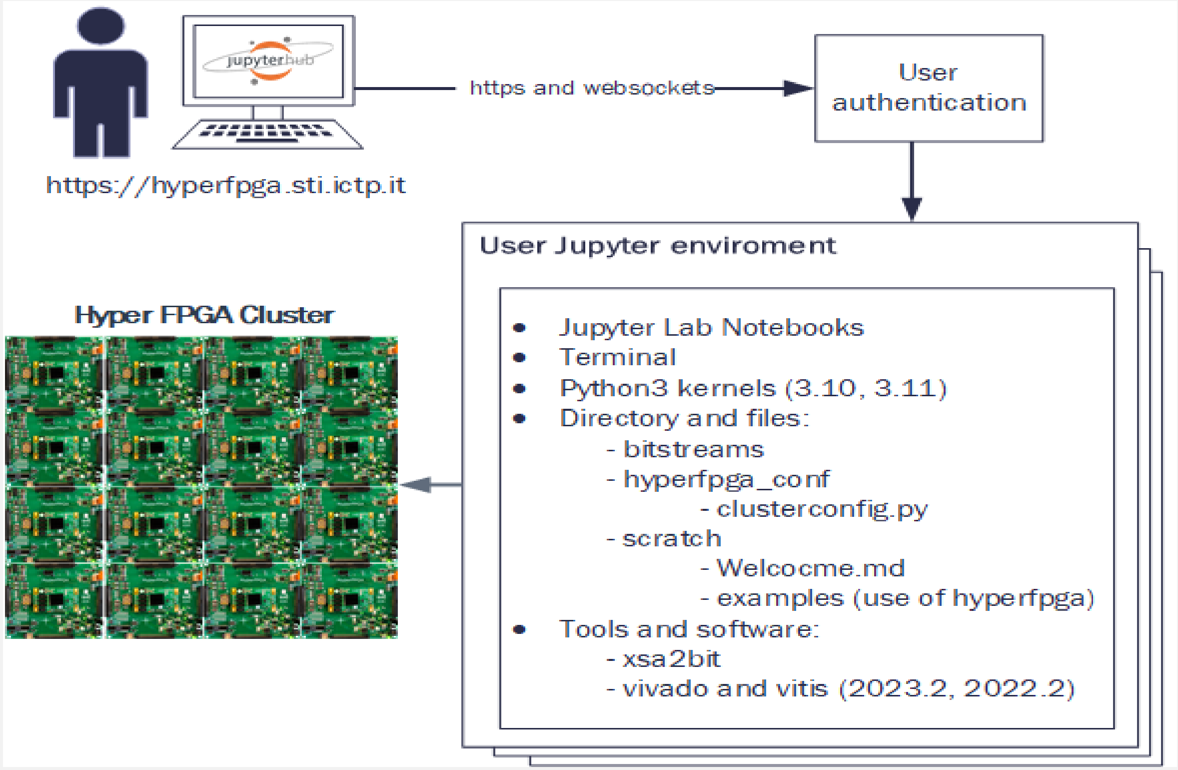
Infrastructure & Management

- Managed thought IaC principles.
- Automated using Ansible for provisioning and updates.
- Flexible resource allocation: 1:1, 1:N, N:1 user-node configurations.

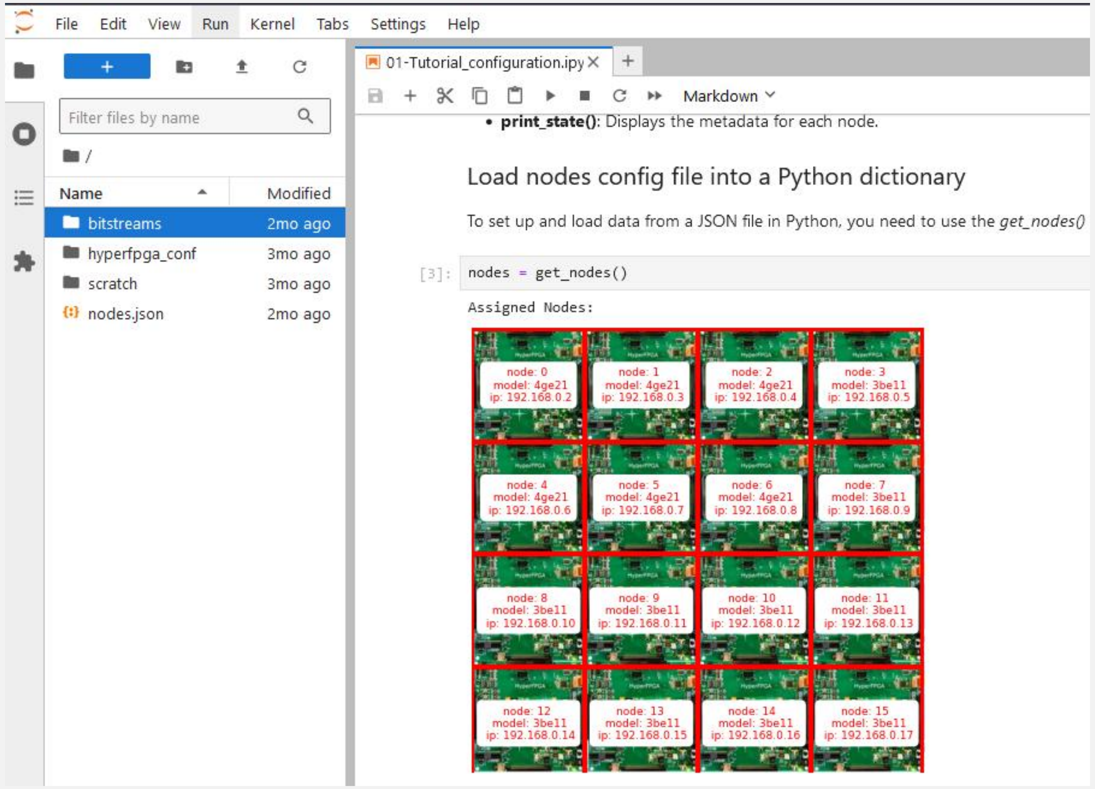


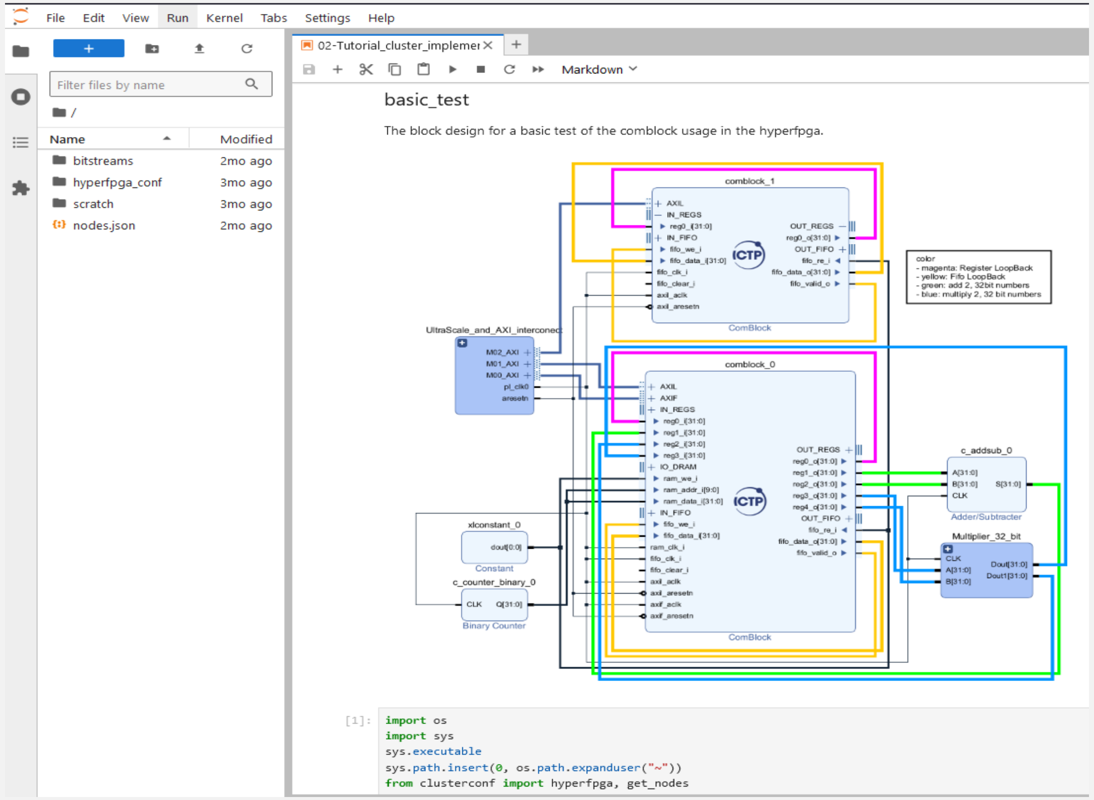
User Interface: JupyterHub

- Web-based, interactive platform.
- Roles: Admin (course content setup), Student (notebook interaction).
- Includes Vivado, Vitis, custom Python libraries for FPGA interaction.



Jupyter Notebooks





Broad spectrum of computational tasks

Parallel

HPC BASICS
IPYPARALLEL
MPI
SSH

Heterogeneous

OVERLAYS
CPU-FPGA

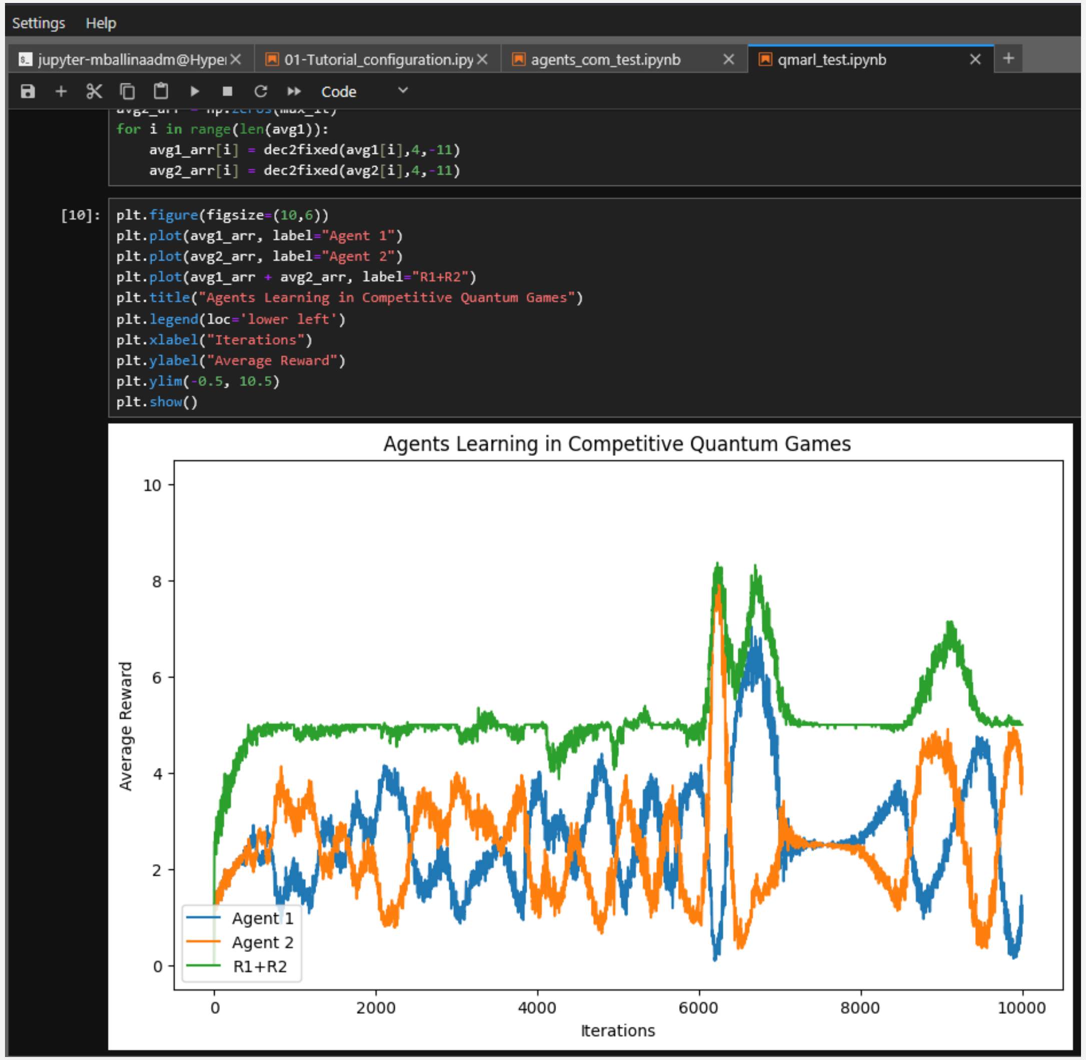
Accelerators

HLS
HLS4ML

Distributed

REPLICATE
COMPLEMENT

Distributing computing examples



Education and collaborative projects

Distributed computing,
DSP,
HPC fundamentals

From Algorithm to Hardware: Machine Learning in Embedded Systems

Distributed computing,
Heterogeneous computing,

Distributed computing

Used in 2 pilot courses with over 50 students across two countries and 3 different collaborative projects.

Conclusion:
Why is heterogeneous computing necessary?

Performance Optimization: GPUs excel at massive parallel processing, while CPUs are better suited for complex sequential and control tasks. FPGAs allow for specific customization of critical tasks and real-time optimizations.

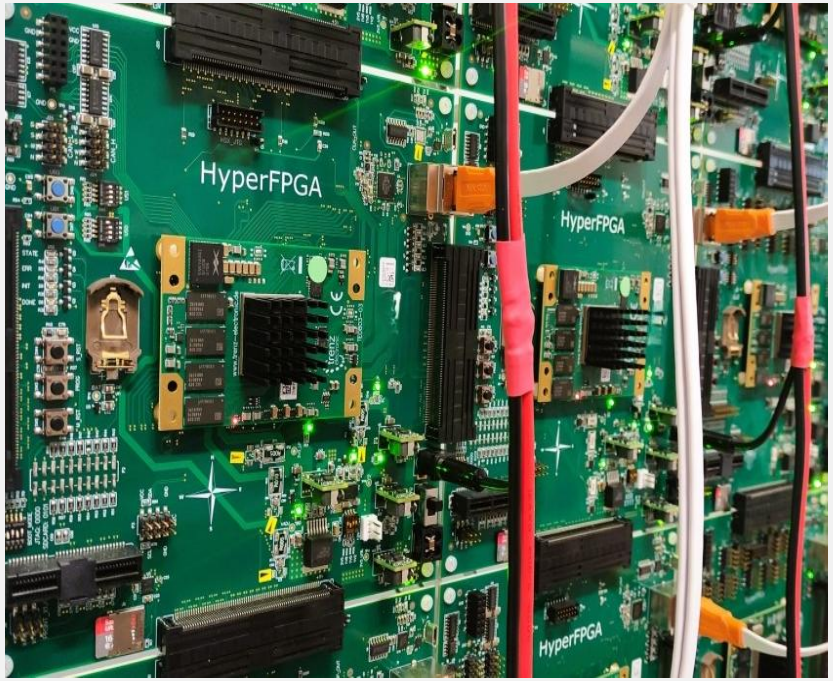
Energy Efficiency: Some accelerators, such as FPGAs, are more energy-efficient than CPUs or GPUs, which is crucial for simulations requiring extensive computational resources over long periods.

Scalability: Accelerators enable simulations to run efficiently at larger scales. Without heterogeneous computing, some simulations requiring real-time data processing or massive scale would be unfeasible.

Latency Reduction: In certain problems, such as those needing real-time simulations or immediate feedback, FPGAs and other accelerators can significantly reduce system latency.

Questions

Thank you for your attention!



Maynor Ballina
mballina@ictp.it