Towards Advanced Computing Architecture: HPC + AI +Q

Carlos Jaime BARRIOS HERNANDEZ, PhD.

cbarrios@uis.edu.co

Universidad Industrial de Santander

CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS

CAGE

Computo Avanzado y a Gran Escala
Advanced and Large Scale Computing
Research group

SCALAC

RISC2

About Me (Using AI Chat)

- Carlos Jaime Barrios Hernandez is a computer science professor and researcher in the field of high-performance computing and large-scale architectures. He is a director of the High Performance and Scientific computing Center (SC3UIS) at Universidad Industrial de Santander (UIS). He has co-founded and organized the Latin American Conference of High-Performance Computing (CARLA) and the Supercomputing and Distributed Systems Camping School (SCCAMP). Today, he is the general chair of the Advanced Computing System for Latin America and Caribe (SCALAC) and part of the board of international collaborations in advanced computing, mainly in HPC and Advanced Computing. He has published research papers on advanced computing, new trends in computing, and parallelism. He was a former DJ in France. Follow social networks. Carlos Jaime is a freeride snowboarder.
- Carlos J. Barrios is doctor in informatics at the Université Côte d'Azur, master in applied mathematics and informatics at the Université de Grenoble, both in France, and systems engineer at the Universidad Industrial de Santander in Bucaramanga, Colombia.

(Last week in Chamerousse, Isère, France)

@carlosjaimebh

HPC/Advanced Computing Challenges

Infrastructure

- **Post Moore Era Architectures**
  - Parallel Balancing, I/O, Memory Challenges
- Dark Sillico
- Exascale
  - Computer Efficiency (Processing/Energy Consumption)
- Hybrid Platforms (CISC+RISC+Others)
  - TPUs, ARM...
- Data Management / Data Centric
- Advanced Networks
- Fog/Edge
- **HPC@Pocket**
- ... Quantum Computing

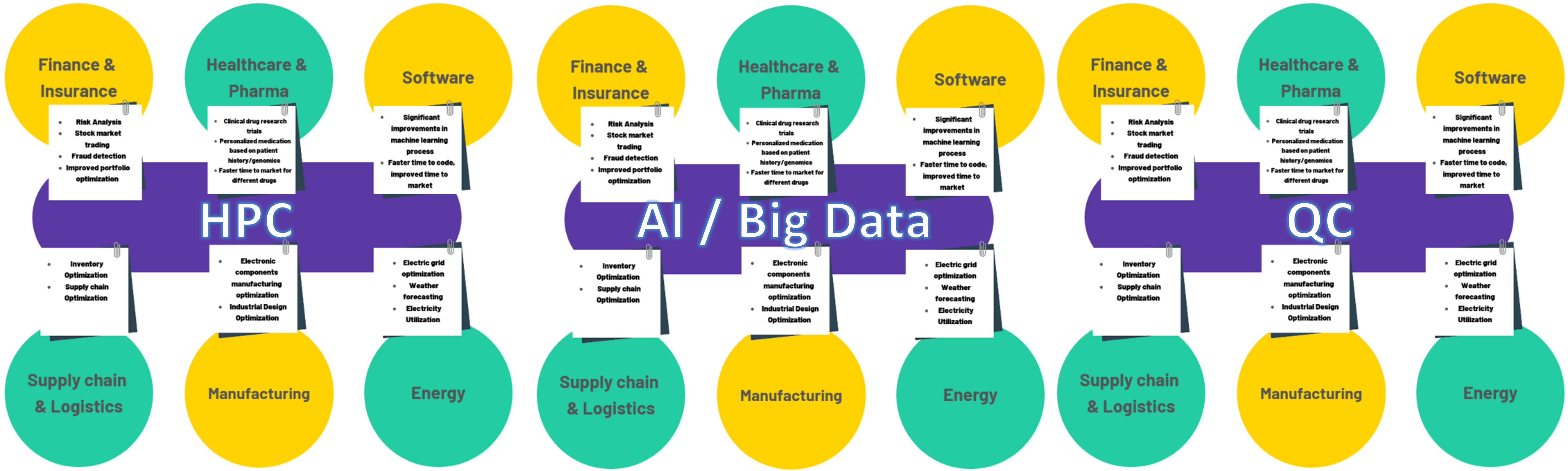
Platform

- Programmability
  - New Languages and Compilers
- Computing Efficiency
- Data Movement and Processing (In Situ, In Transit, Workflows)
- **HPC as a Service**
  - **Science Gateways, Containers**
- **Viz as a Service (In Situ)**
- Protocols
- IA and Deep Learning Frameworks
- Quantum Computing

Applications

- **IA and Deep Learning**
- Algorithms Implementation
- Use of Interpretators (as Python)
- Community versions
- Open Algorithms, Open Data
- **Utra Scale Applicatons**
- Quantum Applications
- and more...

Top Production Applications in Advanced Computing Systems



HPC

Finance & Insurance
- Risk Analysis
- Stock market trading
- Fraud detection
- Improved portfolio optimization

Healthcare & Pharma
- Clinical drug research trials
- Personalized medication based on patient history/genomics
- Faster time to market for different drugs

Software
- Significant improvements in machine learning process
- Faster time to code, improved time to market

Supply chain & Logistics
- Inventory Optimization
- Supply chain Optimization

Manufacturing
- Electronic components manufacturing optimization
- Industrial Design Optimization

Energy
- Electric grid optimization
- Weather forecasting
- Electricity Utilization

AI / Big Data

Finance & Insurance
- Risk Analysis
- Stock market trading
- Fraud detection
- Improved portfolio optimization

Healthcare & Pharma
- Clinical drug research trials
- Personalized medication based on patient history/genomics
- Faster time to market for different drugs

Software
- Significant improvements in machine learning process
- Faster time to code, improved time to market

Supply chain & Logistics
- Inventory Optimization
- Supply chain Optimization

Manufacturing
- Electronic components manufacturing optimization
- Industrial Design Optimization

Energy
- Electric grid optimization
- Weather forecasting
- Electricity Utilization

QC

Finance & Insurance
- Risk Analysis
- Stock market trading
- Fraud detection
- Improved portfolio optimization

Healthcare & Pharma
- Clinical drug research trials
- Personalized medication based on patient history/genomics
- Faster time to market for different drugs

Software
- Significant improvements in machine learning process
- Faster time to code, improved time to market

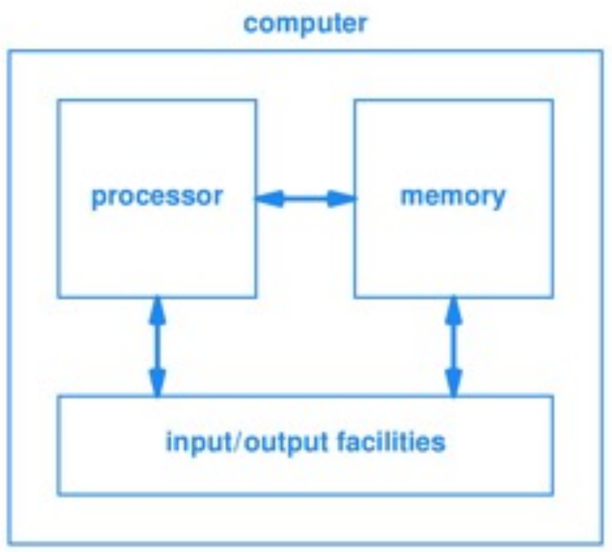
Supply chain & Logistics
- Inventory Optimization
- Supply chain Optimization

Manufacturing
- Electronic components manufacturing optimization
- Industrial Design Optimization

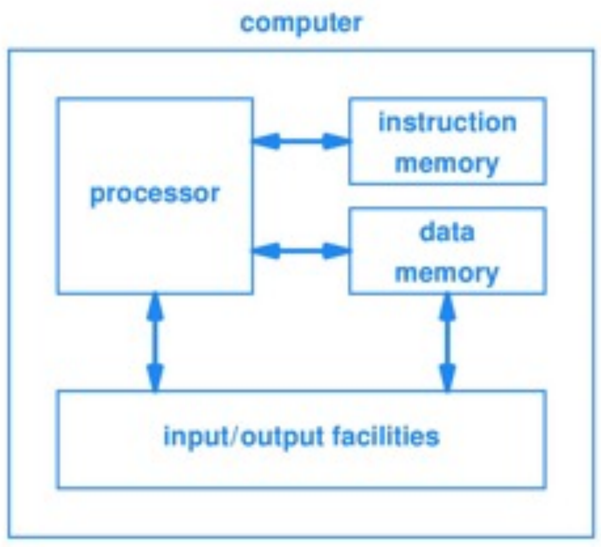
Energy
- Electric grid optimization
- Weather forecasting
- Electricity Utilization

Source: https://www.g2.com/

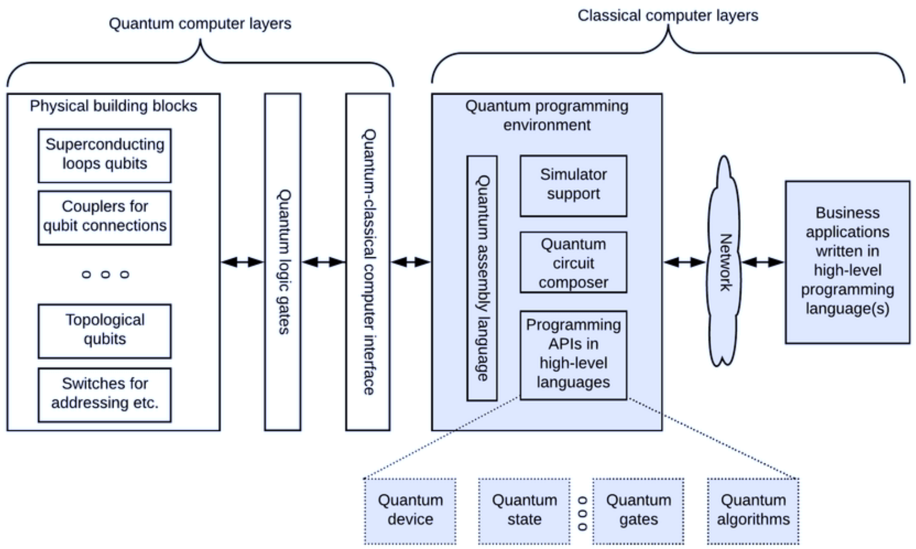
Computer Architecture Support
(i.e. QC Support)



+



=

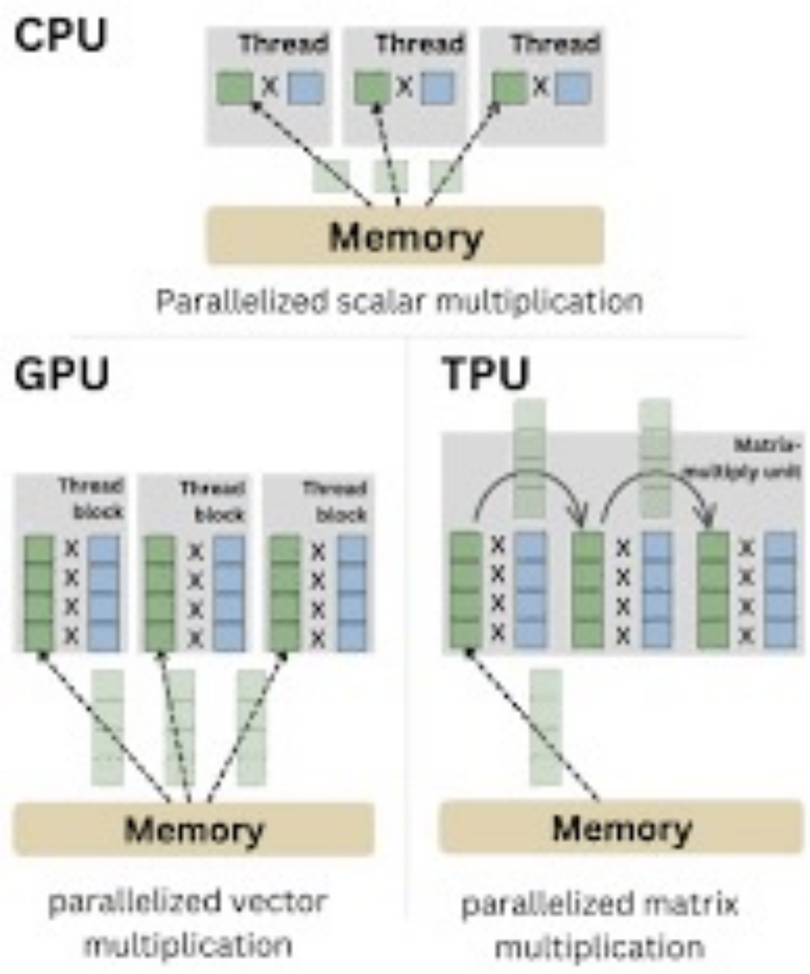

Quantum computer layers
Classical computer layers
Physical building blocks
Superconducting loops qubits
Couplers for qubit connections
Topological qubits
Switches for addressing etc.
Quantum logic gates
Quantum-classical computer interface
Quantum programming environment
Quantum assembly language
Simulator support
Quantum circuit composer
Programming APIs in high-level languages
Network
Business applications written in high-level programming language(s)
Quantum device
Quantum state
Quantum gates
Quantum algorithms

From https://eca.cs.purdue.edu/index.html
And Sodhi, Balwinder. Quality Attributes on Quantum Computing Platforms.

Main Topics

- Some Computer Architecture Features
- Open Questions (and our contribution)
- From HPC Architecture to Advanced Computing Architecture
- And more Open Questions..

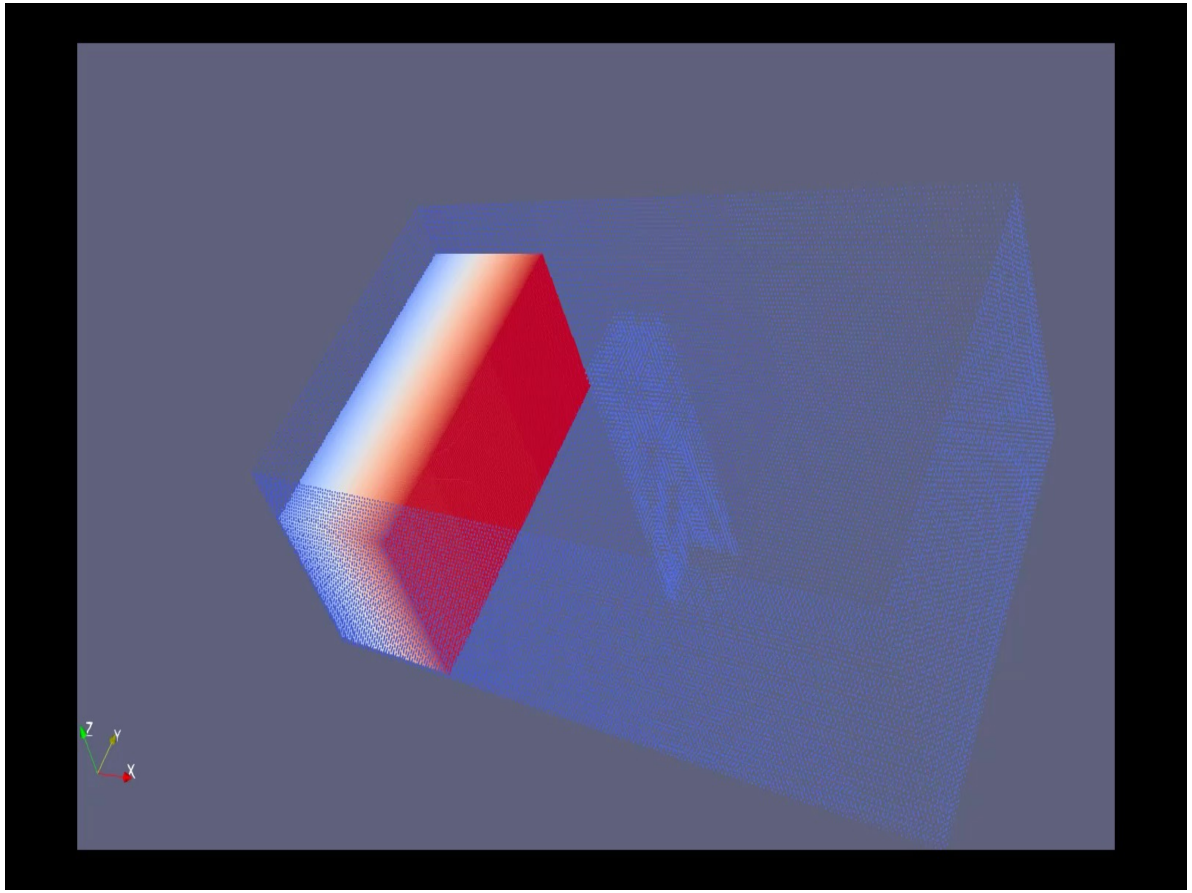
CPU/GPU+TPU Platform



CPU
Thread
Thread
Thread
Memory
Parallelized scalar multiplication
GPU
Thread block
Thread block
Thread block
Memory
parallelized vector multiplication
TPU
Matrix-multiply unit
Memory
parallelized matrix multiplication

- Good Points
  - Coarse/Fine-Grained Processing
  - Mixed Dense or Sparse Computation (ideal for A.I.)
  - Numerical Methods addressed Large Dependencies (Memory Latency, Memory Bandwidth) and Regularity (ideal for simulations)
  - Memory and FP advantages to simulate states or specific representation (as in the case of QC)
- Bad Points
  - Efficiency (*)
  - Programmability
  - Market Price (Now)
  - (Very) Specific Use

Simulation + Visualization using CPU+GPU



Multi-GPU DSPH Analysis Project Video
N. Gutierrez, S. Gelvez, J. Chacon, I. Gitler and C.Barrios

Computo Avanzado y a Gran Escala
Advanced and Large Scale Computing
Research group

https://dual.sphysics.org/

Open Question 1: How to exploit better parallelism to support computing and visualization (AI and Simulation)?

Production visualization: “Pure Parallelism”

From: Hank Childs

Lawrence Berkeley Lab & UC Davis

Production visualization with “pure parallelism”: same problems that processing

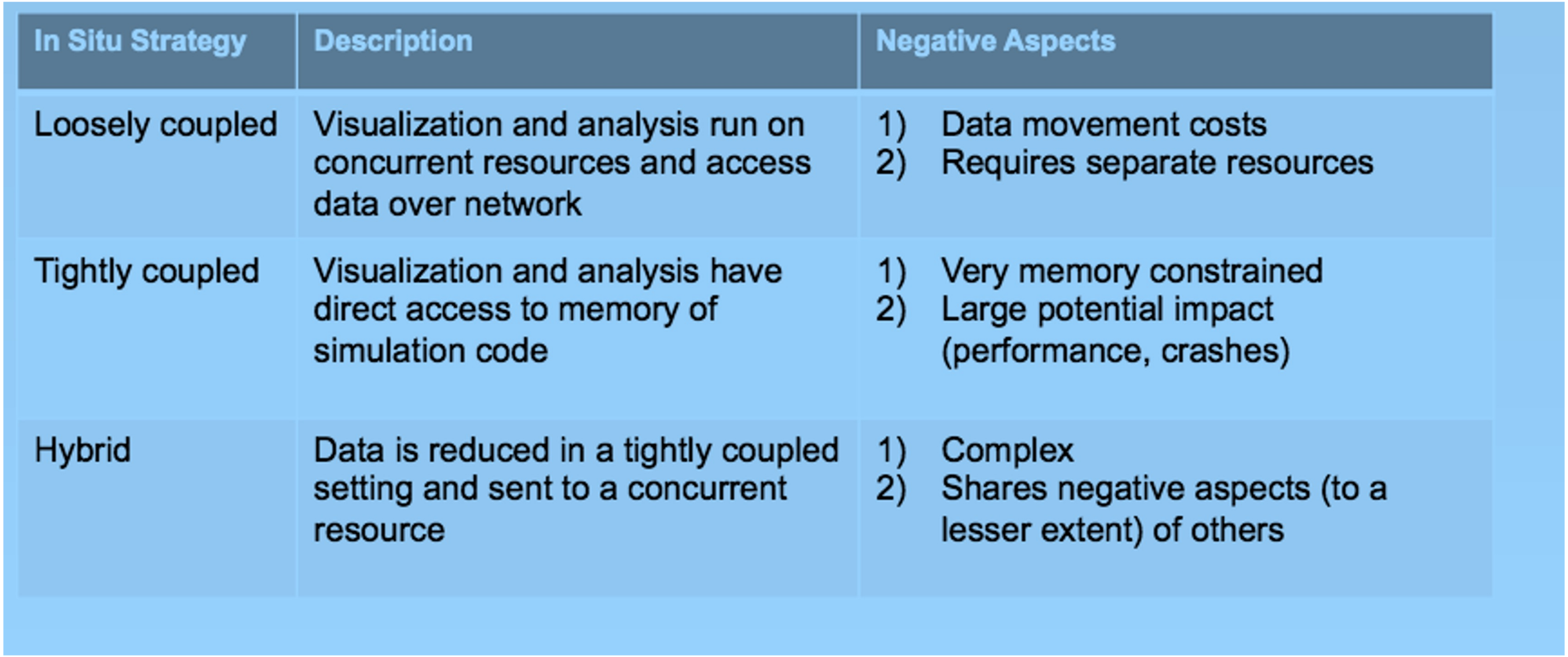
Pure parallelism emphasizes I/O and memory

- High Cost (Efficiency, Performance, Energy)
- Difficult to programming and use

Hardware Disruption

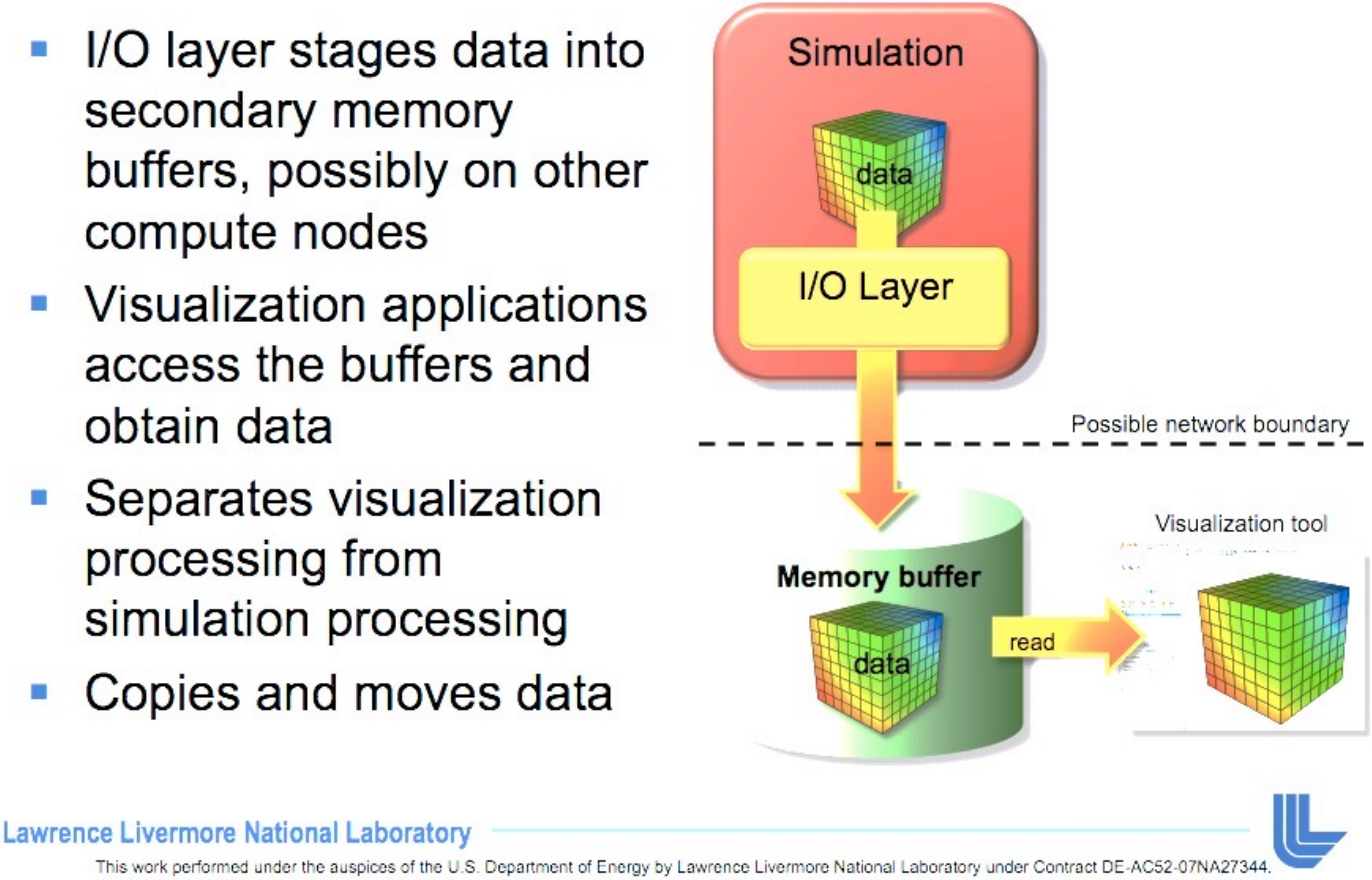
- Accelerators (GPUs, ARM, Xeon Phi)
- Specific Issues (i.e. TPUs, 3D Memory)

In Situ Strategies:



In Situ Strategy	Description	Negative Aspects
Loosely coupled	Visualization and analysis run on concurrent resources and access data over network	1) Data movement costs 2) Requires separate resources
Tightly coupled	Visualization and analysis have direct access to memory of simulation code	1) Very memory constrained 2) Large potential impact (performance, crashes)
Hybrid	Data is reduced in a tightly coupled setting and sent to a concurrent resource	1) Complex 2) Shares negative aspects (to a lesser extent) of others

Loosely Coupled



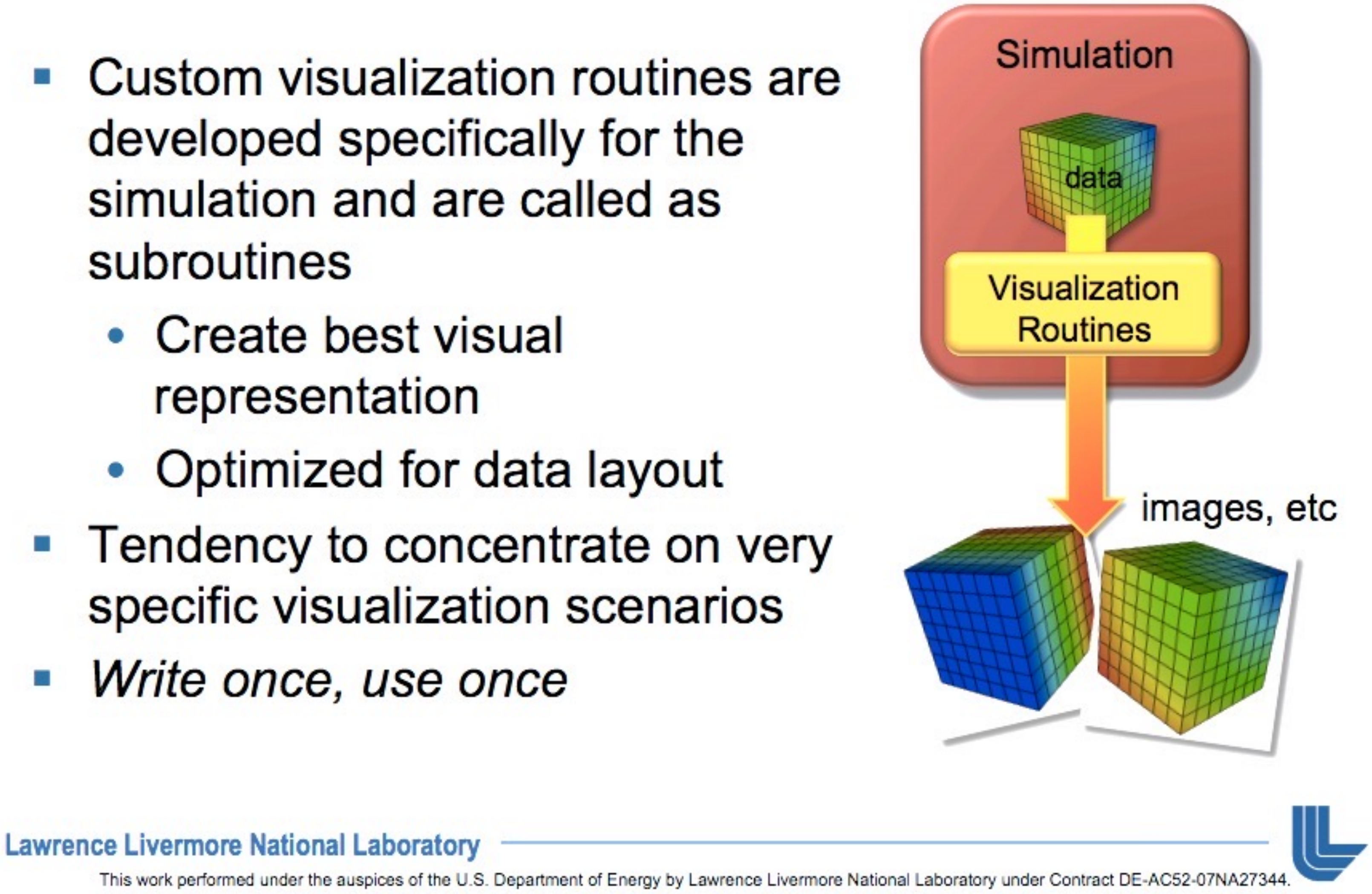
- I/O layer stages data into secondary memory buffers, possibly on other compute nodes
- Visualization applications access the buffers and obtain data
- Separates visualization processing from simulation processing
- Copies and moves data

Lawrence Livermore National Laboratory

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Demands Dynamic Memory

Thightly Coupled



- Custom visualization routines are developed specifically for the simulation and are called as subroutines
  - Create best visual representation
  - Optimized for data layout
- Tendency to concentrate on very specific visualization scenarios
- *Write once, use once*

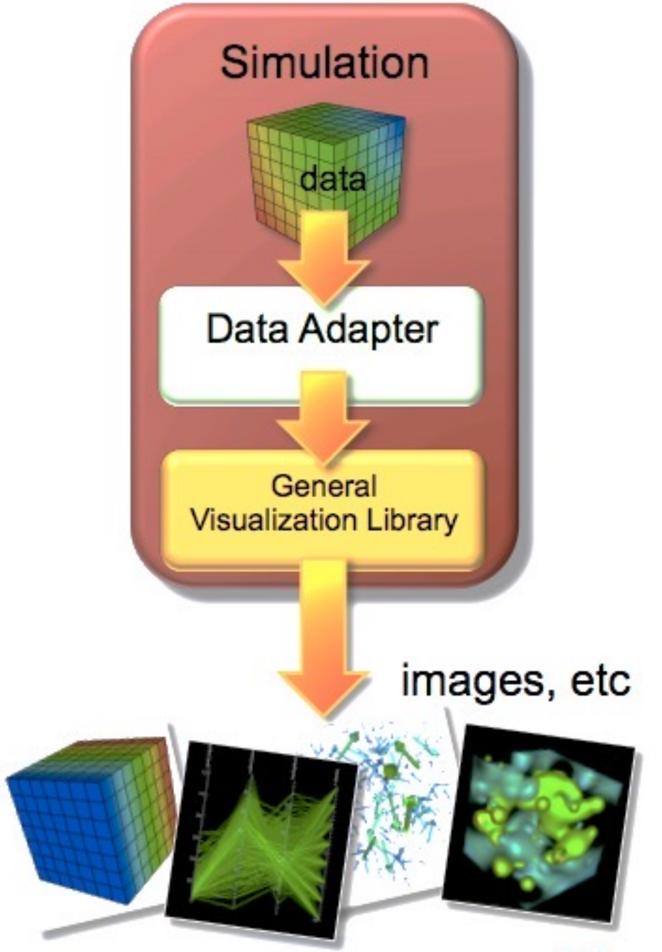
Lawrence Livermore National Laboratory

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Demands Dynamic Memory and a large amount of memory capabilities

Hybrid

- Simulation uses data adapter layer to make data suitable for general purpose visualization library
- Rich feature set can be called by the simulation
- Operate directly on the simulation's data arrays when possible
- *Write once, use many times*



Simulation

data

Data Adapter

General Visualization Library

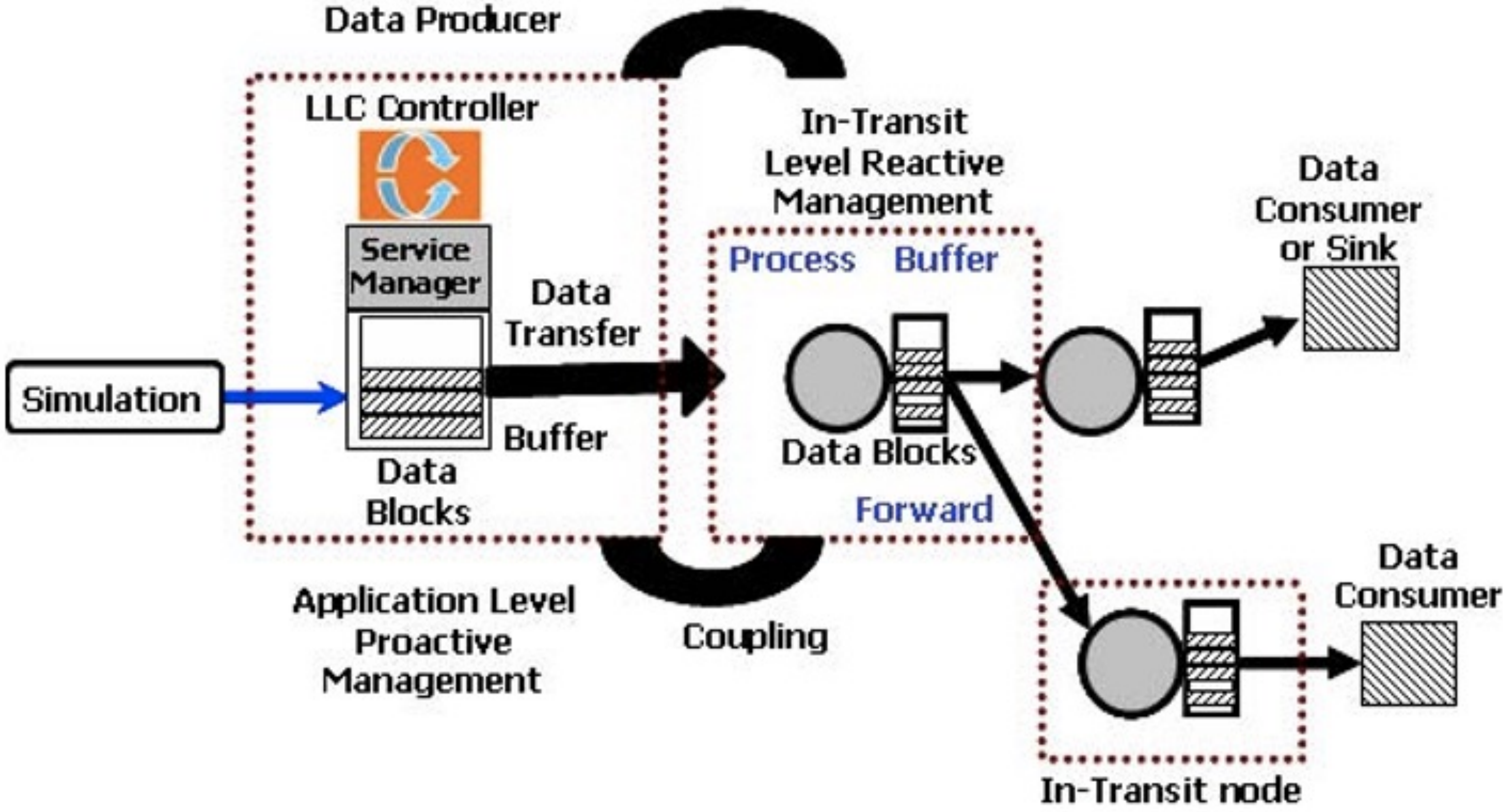
images, etc

Demands Dynamic Memory, a large amount of memory capabilities and specific algorithm approach

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

And In Transit?

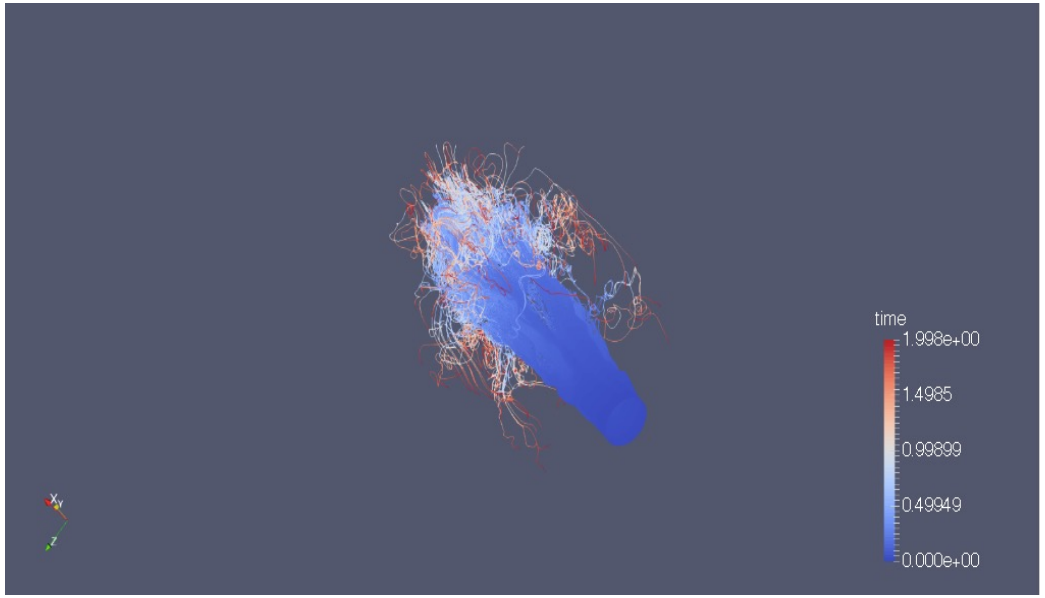
Analysis and Visualization is run on I/O nodes that receive the full simulation results but write information from analysis or provide run-time visualization



Super Computación y Cálculo Científico UIS

GROMACS
FAST. FLEXIBLE. FREE.

Our Contribution



Source: generated by code provided by VisLab Uni-KL. Rendered in Paraview.

A (new) Algorithm Analytics

Performance evaluation for seeds, steps, buffer depth.
A definition of metrics.
A new, more detailed evaluation.
After those, a new algorithm.

Platforms with In-Situ and In-Transit Strategies

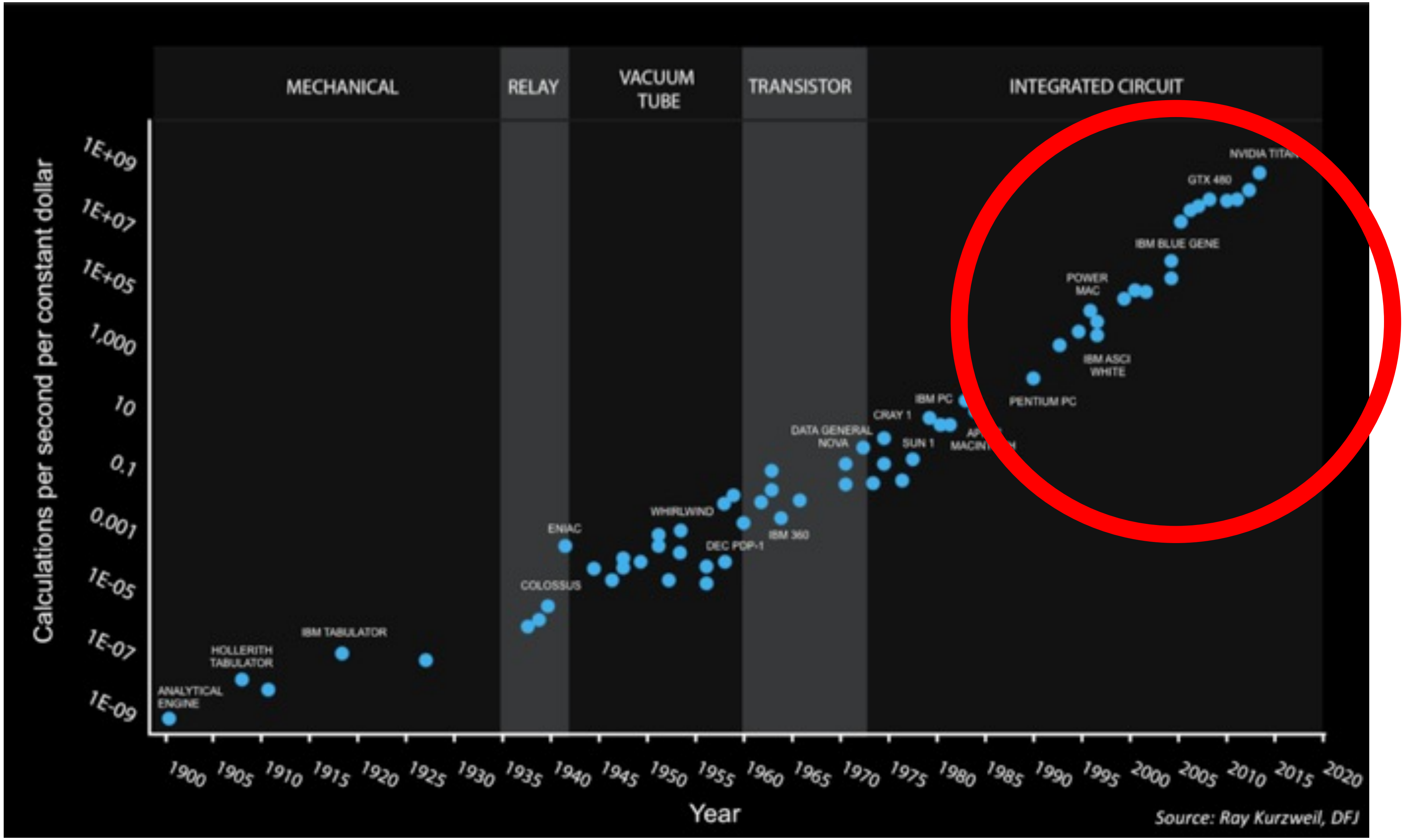
Tightly and Hybrid Approach
Exascale
Mixing Processing and Visualization Issues

Applications (Scientific Real Time)

GROMACS, NAMD, FlowVR…
High Availible Autonomous Systems
Specific Libraries and Frameworks
Ultrascale Software
Special In-Situ Tools (NVIDIA® VisIt)
Deep Learning Applications
Data Movement

Sergio Gelvez PhD. Thesis Visualisation Of Vector Fields In Parallel Environments: In-situ Approach Over Heterogeneous Architectures (Advising by K. Garth and C. J. Barrios, Collaborators: B. Raffin (INRIA) , J. Hernández (UniAndes) and B. Hernández (NVIDIA)

The (Post) Moore Era



Source: Ray Kurzweil, DFJ

After 120 years… The Moore's Law is Dead

Jack Dongarra

The Cambrian Explosion in Architecture for AI

Satoshi Matsouka Vision

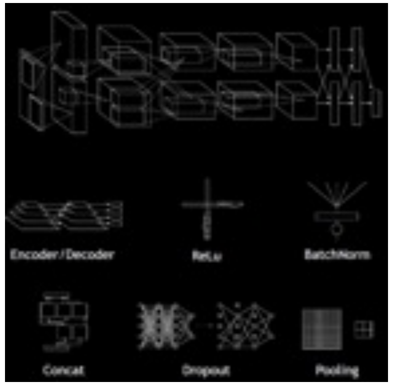
CPU

RAM

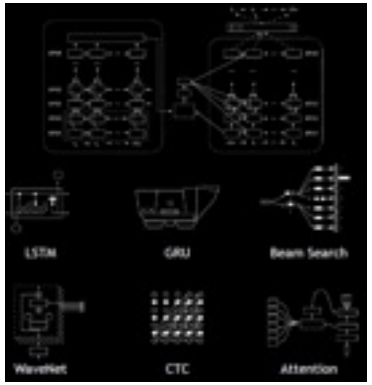
GPU

Storage

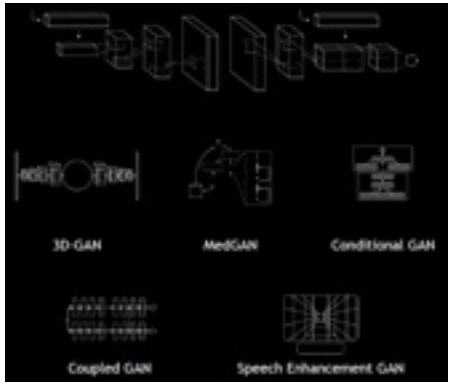
Convolution Networks



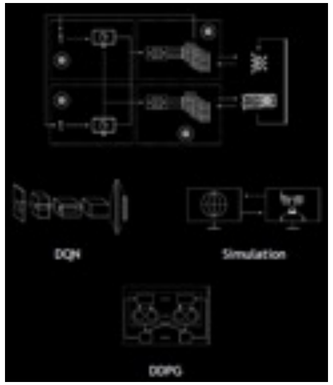
Recurrent Networks



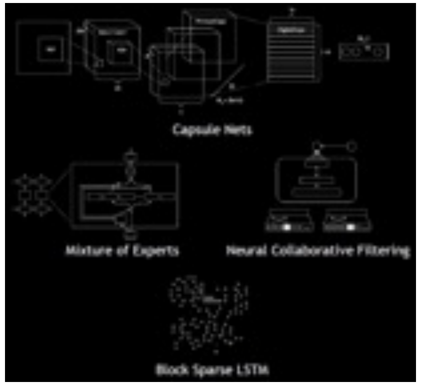
AR Netwroks



Deep Learning



New Networks



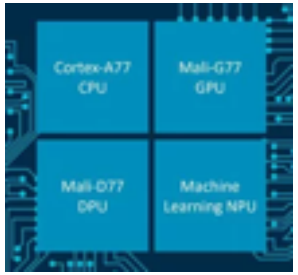
RPI

Nano

TX2

Xavier

FPGA



SiP

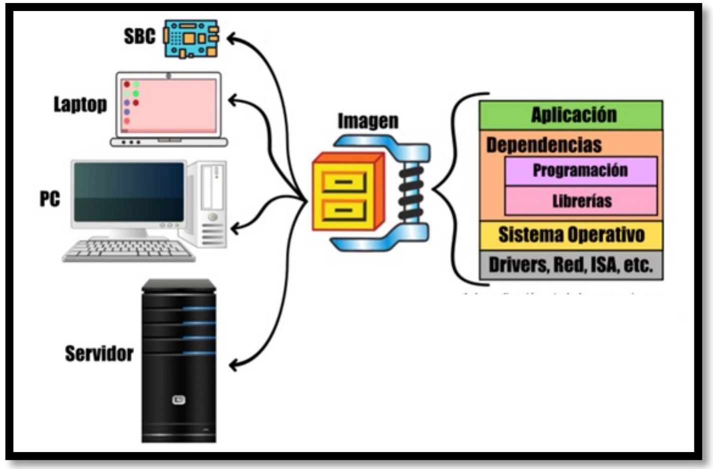
ASIC

TPU

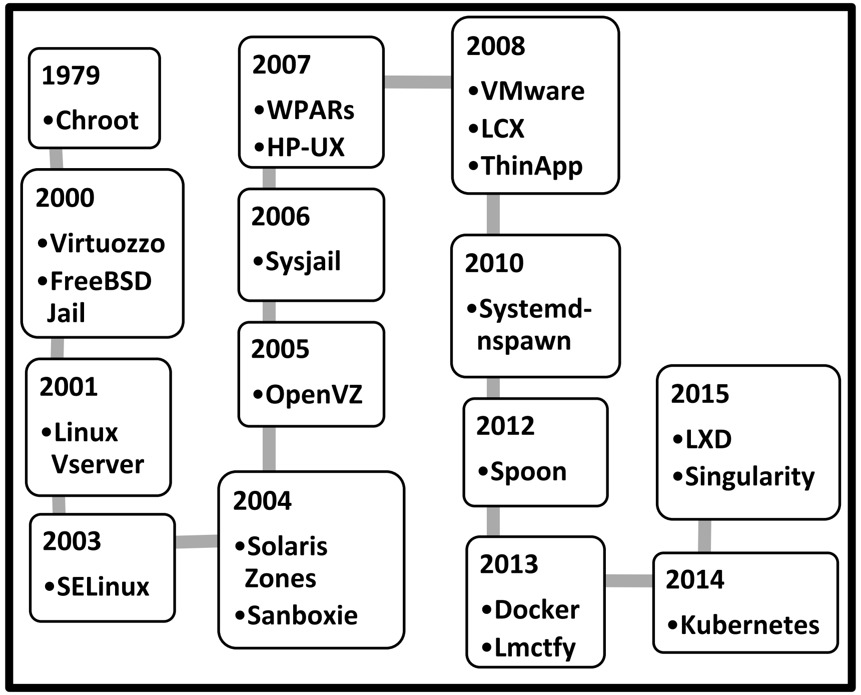
V100

Open Question 2: How to exploit Efficiently the Post Moore Architectures?

Virtualization or Containerization?



Virtualization



1979
- Chroot

2000
- Virtuozzo
- FreeBSD Jail

2001
- Linux Vserver

2003
- SELinux

2004
- Solaris Zones
- Sanboxie

2005
- OpenVZ

2006
- Sysjail

2007
- WPARs
- HP-UX

2008
- VMware
- LCX
- ThinApp

2010
- Systemd-nspawn

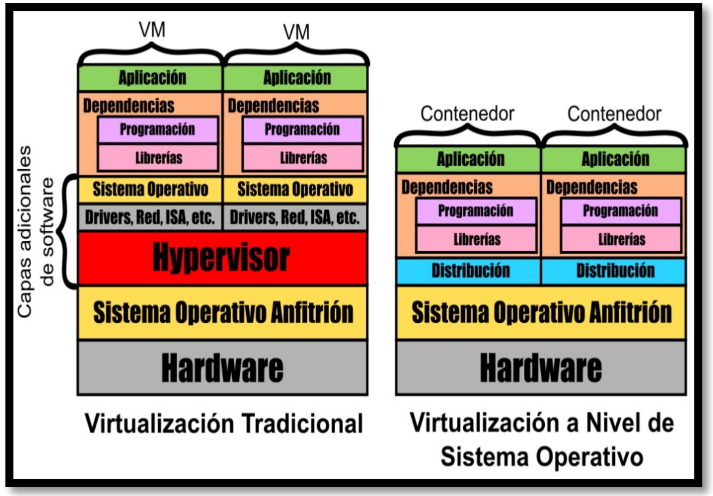
2012
- Spoon

2013
- Docker
- Lmctfy

2014
- Kubernetes

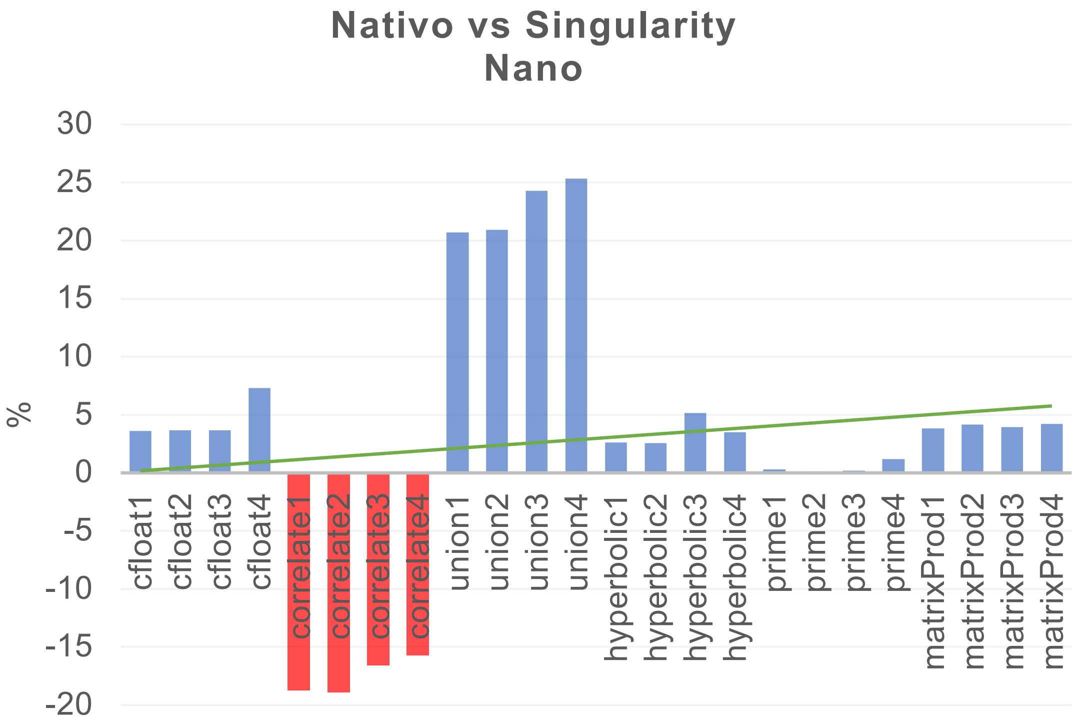
2015
- LXD
- Singularity

Evolution of Virtualization



Conteiners vs Virtualization?

Our Contribution: Performance Impact in Effective Deployment

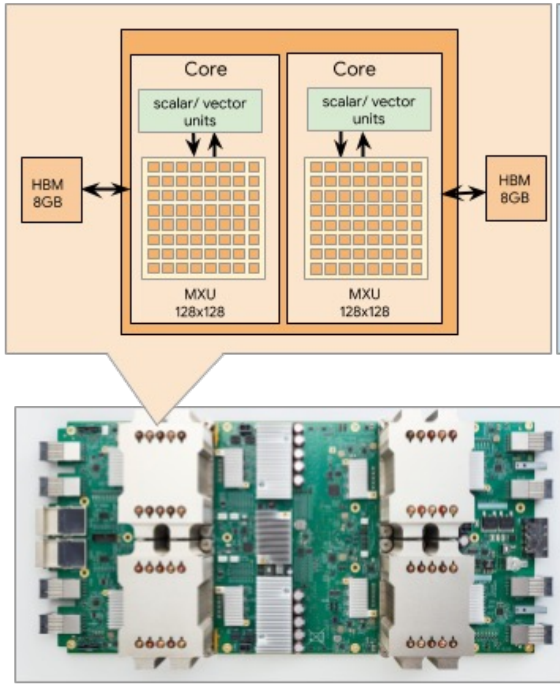


- Definition of Computing Efficiency:
  - In terms of Energy + ”computing element” + processing
- Definition of Post-Moore Era Architectures
  - Parallelism Support + Efficiency + Sustainability?
- Methodology to Analyze and (to predict) the impact of containerization
- Practical Approach to Scheduling Performance Evaluation

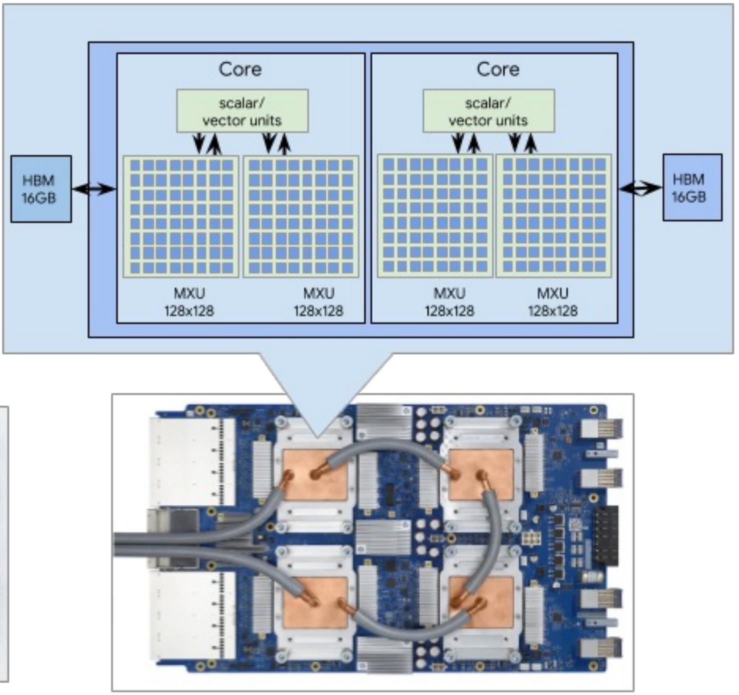
Pablo Rojas Thesis « Study of the deployment and execution of applications on post-moore architectures » co-advising with L.A. Steffennel

TPUs: Tensor Processing Units

Tensor Processing Unit (TPU) is an AI accelerator application-specific integrated circuit (ASIC) developed by Google and NVIDIA specifically for neural network machine learning

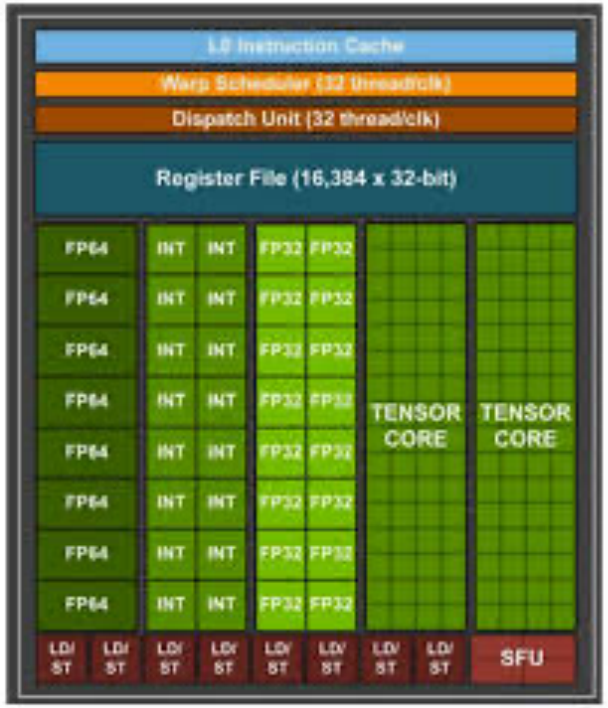


TPU v2 - 4 chips, 2 cores per chip



TPU v3 - 4 chips, 2 cores per chip

Google TPU

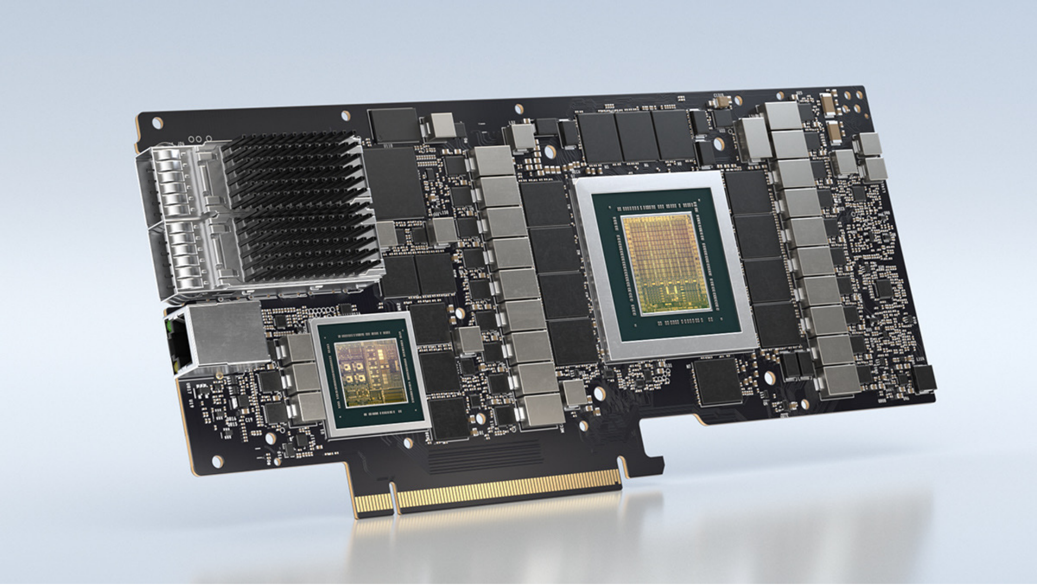


L0 Instruction Cache
Warp Scheduler (32 thread/clk)
Dispatch Unit (32 thread/clk)
Register File (16,384 x 32-bit)
FP64
INT
INT
FP32
FP32
TENSOR CORE
TENSOR CORE
LD/ ST
SFU

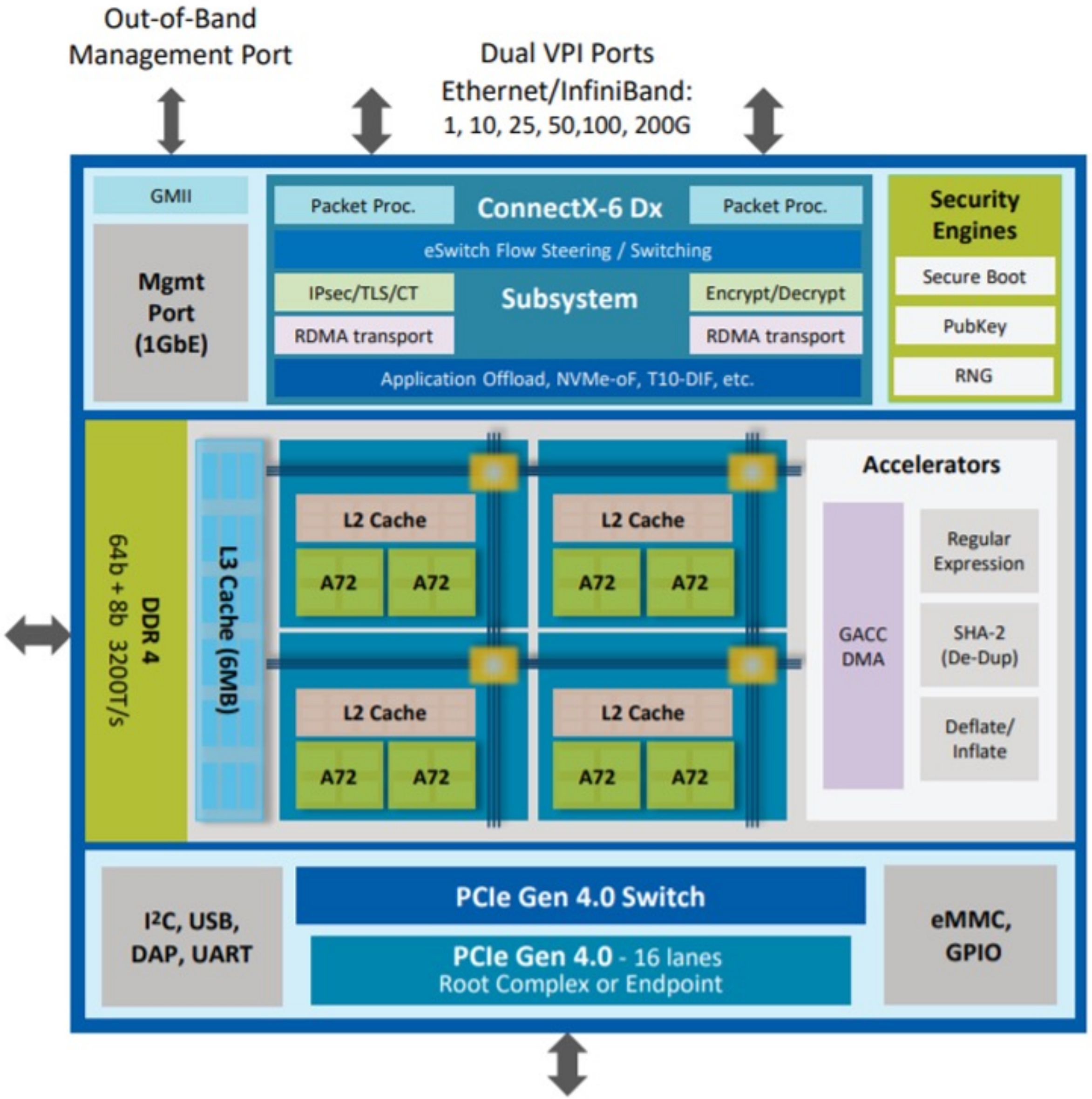
Source: NVIDIA

NVIDIA TPU

DPU Architecture



NVDIA DPU



Out-of-Band Management Port

Dual VPI Ports Ethernet/InfiniBand: 1, 10, 25, 50,100, 200G

GMII

Mgmt Port (1GbE)

Packet Proc.

ConnectX-6 Dx

Packet Proc.

eSwitch Flow Steering / Switching

IPsec/TLS/CT

Subsystem

Encrypt/Decrypt

RDMA transport

RDMA transport

Application Offload, NVMe-oF, T10-DIF, etc.

Security Engines

Secure Boot

PubKey

RNG

DDR 4 64b + 8b 3200T/s

L3 Cache (6MB)

L2 Cache

A72

A72

L2 Cache

A72

A72

L2 Cache

A72

A72

L2 Cache

A72

A72

Accelerators

GACC DMA

Regular Expression

SHA-2 (De-Dup)

Deflate/ Inflate

I²C, USB, DAP, UART

PCIe Gen 4.0 Switch

PCIe Gen 4.0 - 16 lanes Root Complex or Endpoint

eMMC, GPIO

Open Question 3: How to Achieve Efficiency and Scalability in HPC Architectures that Support AI and Big Data?

Two Approach to Contribute to Deal with the Question:

- **Computing Architectural Approach**
- **Algorithm Approach**

COMPUTING ARCHITECTURAL APPROACH

What is the problem?

The need to have increasingly efficient computational resources with better performance, among which are greater processing capacity and memory available for the execution of the training of these models..

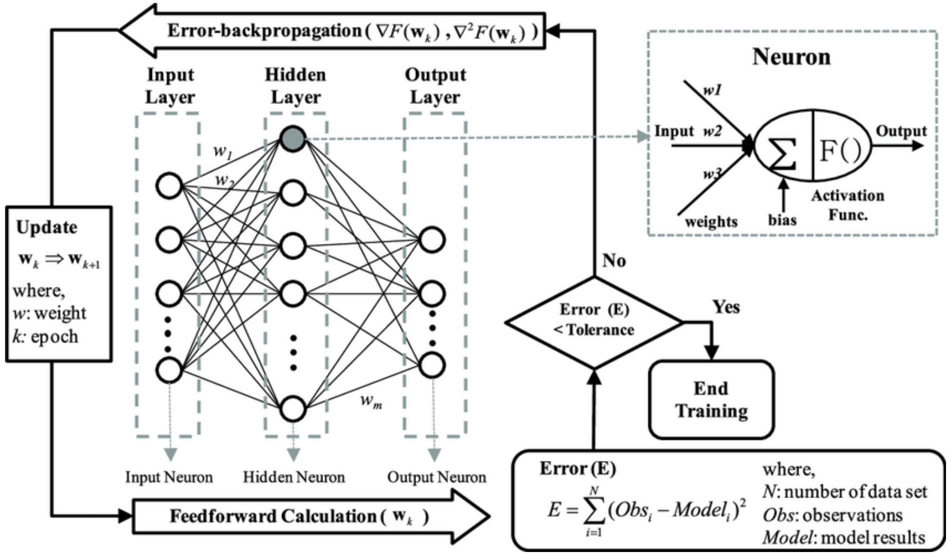
→

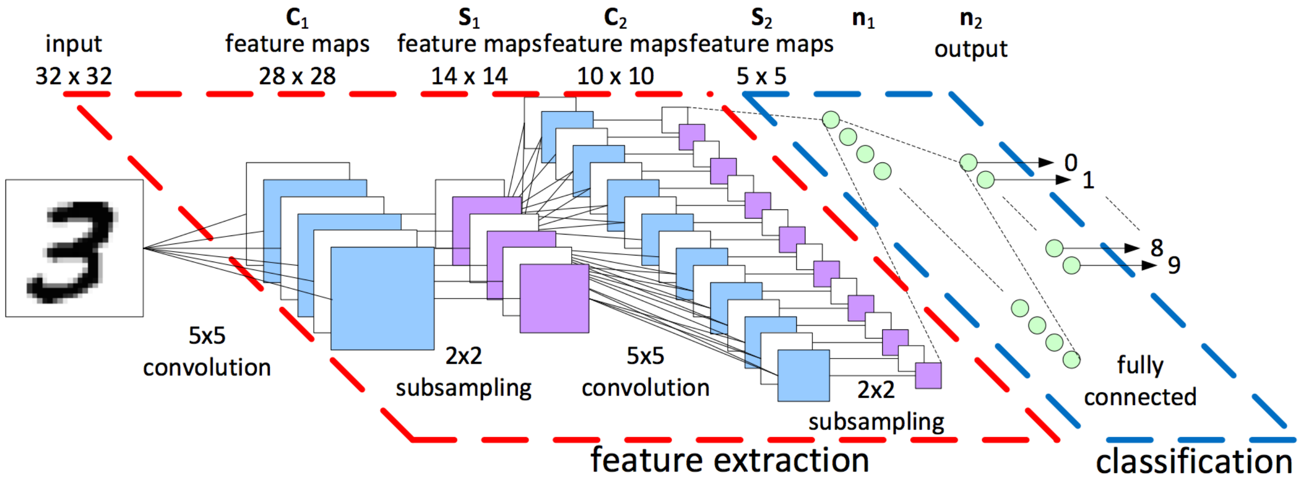
The deep learning model training algorithm requires a significant amount of memory that often exceeds the capabilities of the GPU and, in some cases, even the memory of the CPU.

New methods for training the model have been created to solve this problem, such as **Model Parallelism**, **Data Parallelism**, and **Pipeline Parallelism**. However, these methods have required increasingly specialized hardware that does not necessarily reduce the memory footprint, but distributes memory requirements across devices such as servers, GPUs, and TPUs.

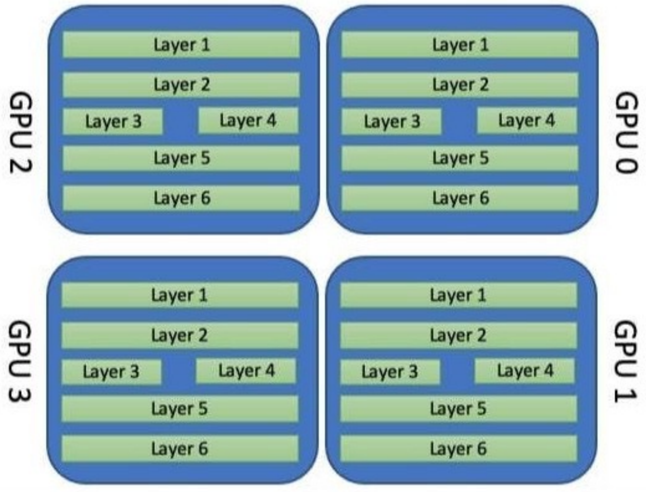
COMPUTING ARCHITECTURAL APPROACH

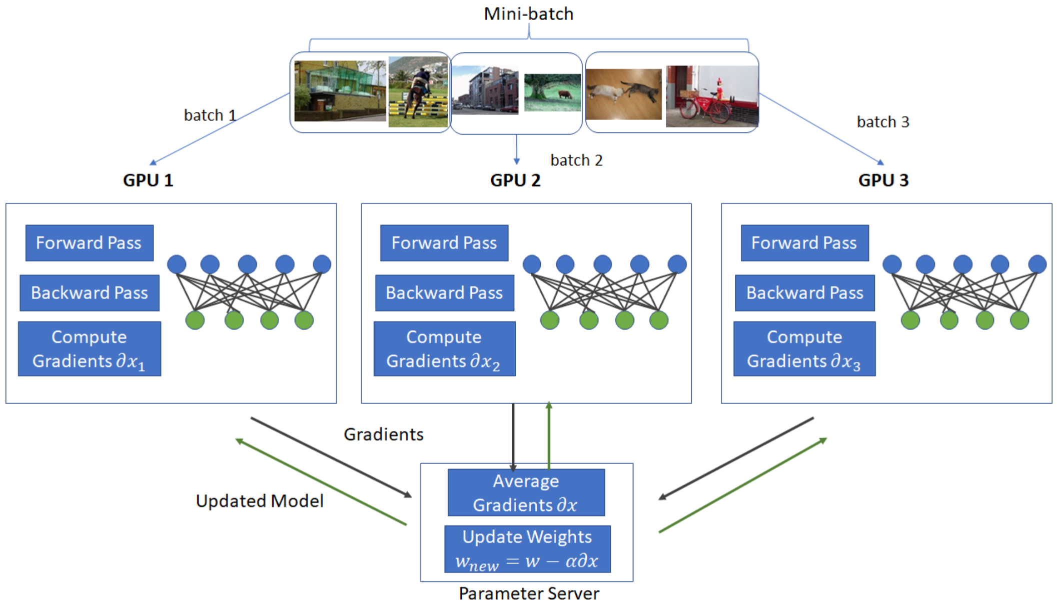
BACKPROPAGATION





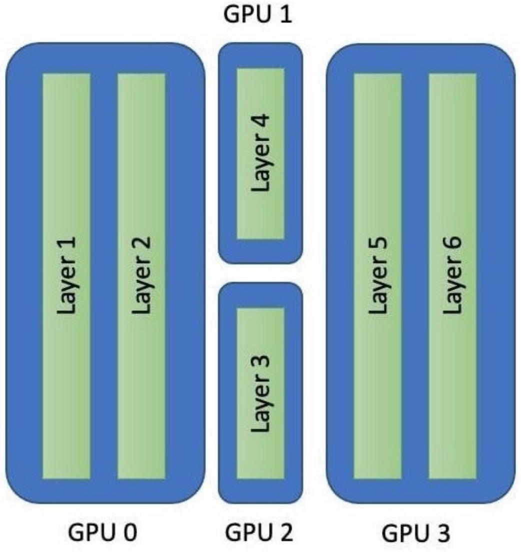
DATA PARALLELISM

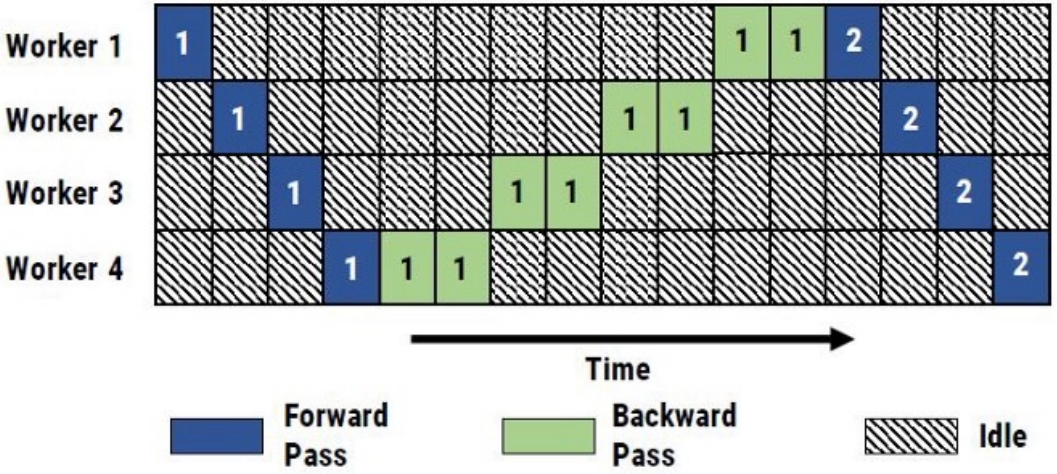



Mini-batch
batch 1
batch 2
batch 3
GPU 1
GPU 2
GPU 3
Forward Pass
Backward Pass
Compute Gradients ∂x_1
Forward Pass
Backward Pass
Compute Gradients ∂x_2
Forward Pass
Backward Pass
Compute Gradients ∂x_3
Gradients
Updated Model
Average Gradients ∂x
Update Weights $w_{new} = w - \alpha \partial x$
Parameter Server

COMPUTING ARCHITECTURAL APPROACH

MODEL PARALLELISM

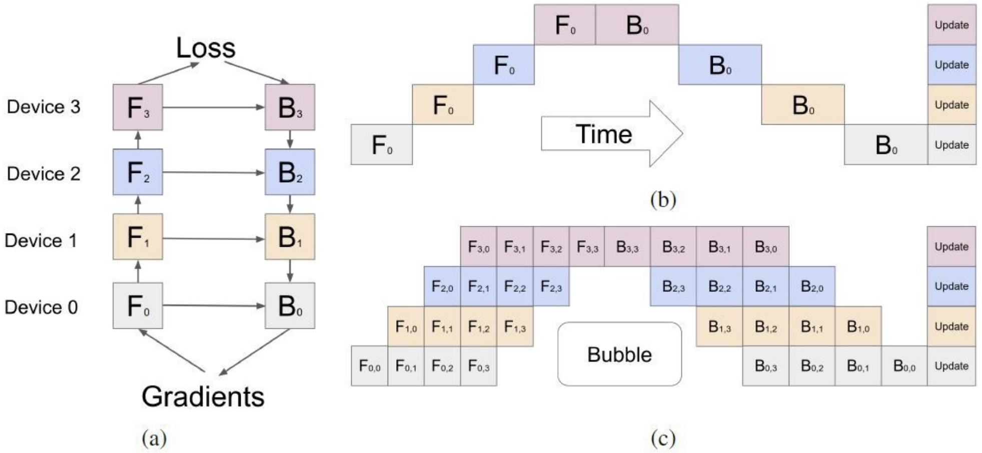


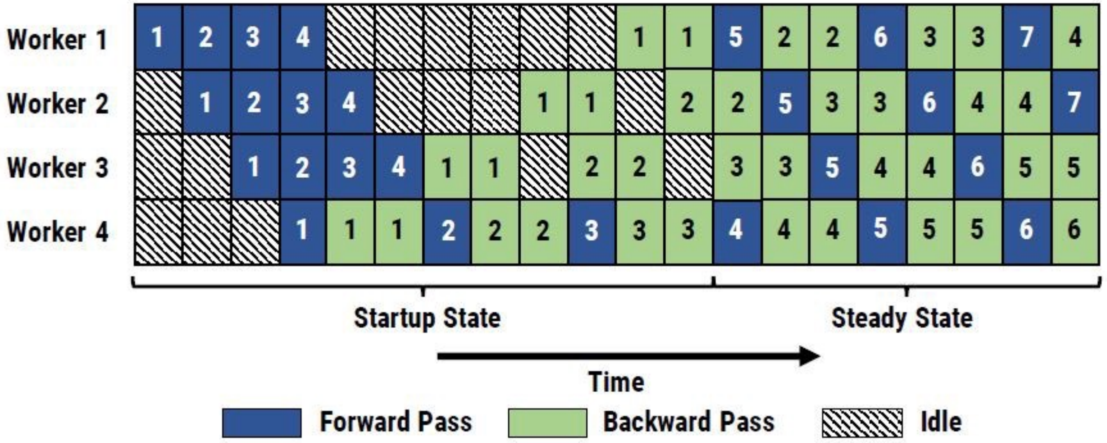


Worker 1	🟦 1	▨	▨	▨	▨	▨	▨	▨	▨	▨	🟩 1	🟩 1	🟦 2	▨	▨	▨
Worker 2	▨	🟦 1	▨	▨	▨	▨	▨	▨	🟩 1	🟩 1	▨	▨	▨	🟦 2	▨	▨
Worker 3	▨	▨	🟦 1	▨	▨	▨	🟩 1	🟩 1	▨	▨	▨	▨	▨	▨	🟦 2	▨
Worker 4	▨	▨	▨	🟦 1	🟩 1	🟩 1	▨	▨	▨	▨	▨	▨	▨	▨	▨	🟦 2

🟦 Forward Pass 🟩 Backward Pass ▨ Idle

PIPELINE PARALLELISM


Loss
Device 3
Device 2
Device 1
Device 0
F_3
F_2
F_1
F_0
B_3
B_2
B_1
B_0
Gradients
(a)
F_0
B_0
F_0
B_0
F_0
B_0
F_0
B_0
Time
Update
Update
Update
Update
(b)
$F_{3,0}$ $F_{3,1}$ $F_{3,2}$ $F_{3,3}$ $B_{3,3}$ $B_{3,2}$ $B_{3,1}$ $B_{3,0}$
$F_{2,0}$ $F_{2,1}$ $F_{2,2}$ $F_{2,3}$ $B_{2,3}$ $B_{2,2}$ $B_{2,1}$ $B_{2,0}$
$F_{1,0}$ $F_{1,1}$ $F_{1,2}$ $F_{1,3}$ $B_{1,3}$ $B_{1,2}$ $B_{1,1}$ $B_{1,0}$
$F_{0,0}$ $F_{0,1}$ $F_{0,2}$ $F_{0,3}$ $B_{0,3}$ $B_{0,2}$ $B_{0,1}$ $B_{0,0}$
Bubble
Update
Update
Update
Update
(c)

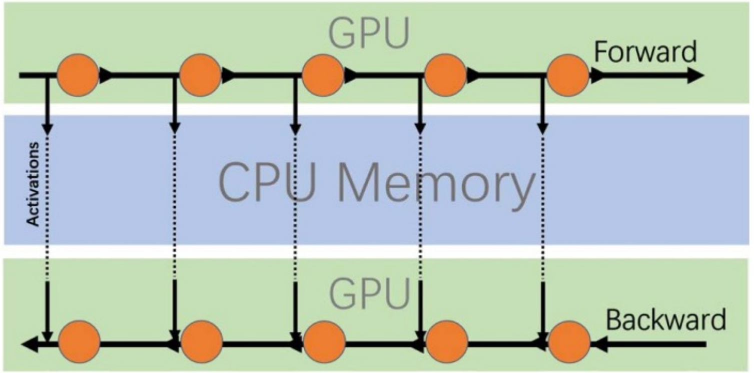


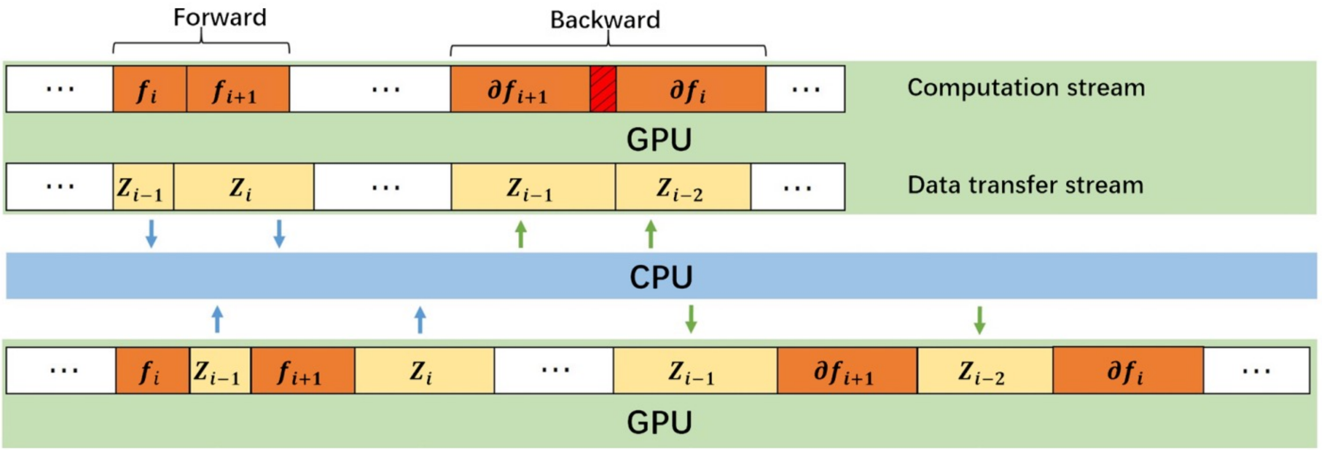
Worker 1	🟦 1	🟦 2	🟦 3	🟦 4	▨	▨	▨	▨	▨	▨	🟩 1	🟩 1	🟦 5	🟩 2	🟩 2	🟦 6	🟩 3	🟩 3	🟦 7	🟩 4
Worker 2	▨	🟦 1	🟦 2	🟦 3	🟦 4	▨	▨	▨	🟩 1	🟩 1	▨	🟩 2	🟩 2	🟦 5	🟩 3	🟩 3	🟦 6	🟩 4	🟩 4	🟦 7
Worker 3	▨	▨	🟦 1	🟦 2	🟦 3	🟦 4	🟩 1	🟩 1	▨	🟩 2	🟩 2	▨	🟩 3	🟩 3	🟦 5	🟩 4	🟩 4	🟦 6	🟩 5	🟩 5
Worker 4	▨	▨	▨	🟦 1	🟩 1	🟩 1	🟦 2	🟩 2	🟩 2	🟦 3	🟩 3	🟩 3	🟦 4	🟩 4	🟩 4	🟦 5	🟩 5	🟩 5	🟦 6	🟩 6

🟦 Forward Pass 🟩 Backward Pass ▨ Idle

COMPUTING ARCHITECTURAL APPROACH

CPU OFFLOADING





Torres, L. A., Barrios, C. J., & Denneulin, Y. (2021). Computational Resource Consumption in Convolutional Neural Network Training – A Focus on Memory. *Supercomputing Frontiers and Innovations*, *8*(1), 45–61. https://doi.org/10.14529/jsfi210104

Our Contribution: A New Parallelization Approach in Deep Learning Using CPU/GPU Architectures for Memory Optimization

Thesis of Alejandro Torres co-advising with Yves Denneulin

Universidad Industrial de Santander
CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS

CAGE

Computo Avanzado y a Gran Escala
Advanced and Large Scale Computing
Research group

UGA
Université Grenoble Alpes

LABORATOIRE D'INFORMATIQUE DE GRENOBLE

Is it possible to optimize memory usage in training deep neural network models by distributing or parallelizing the Pipeline between the CPU and the GPU/TPU?

By distributing the Pipeline between the CPU and the GPU/TPU, can better results be obtained in training times while maintaining the accuracy of the model prediction?

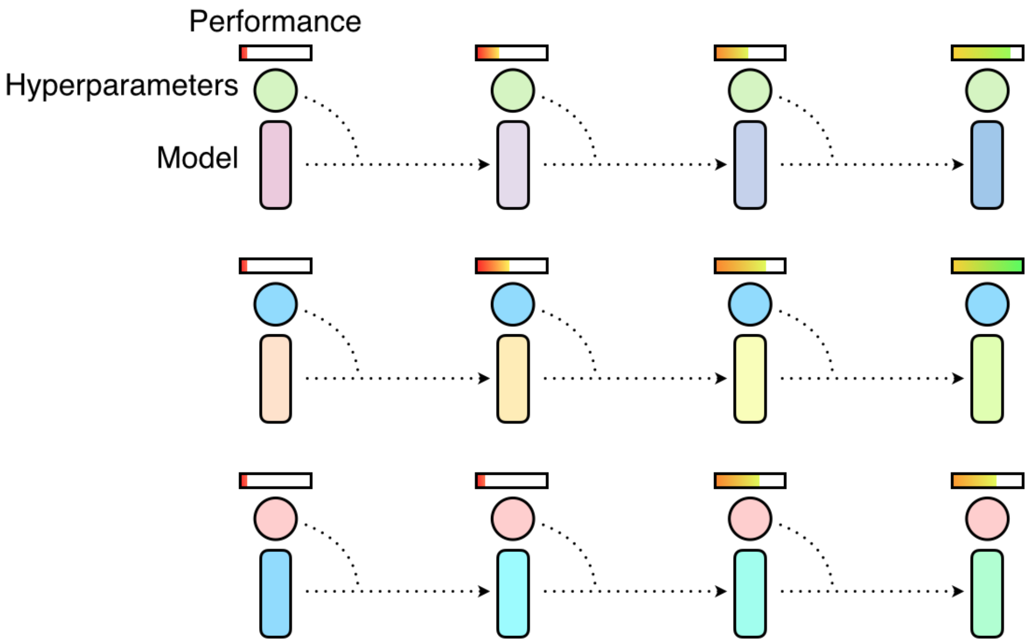
By having greater storage capacities in the CPU memory to use it as an active actor, is it possible to increase the size of the input batch and thus improve the efficiency of the training?

Does using the CPU and GPU/TPU simultaneously during training involve more or less energy expenditure when comparing training time Vs. Accuracy Vs. Power Consumption?

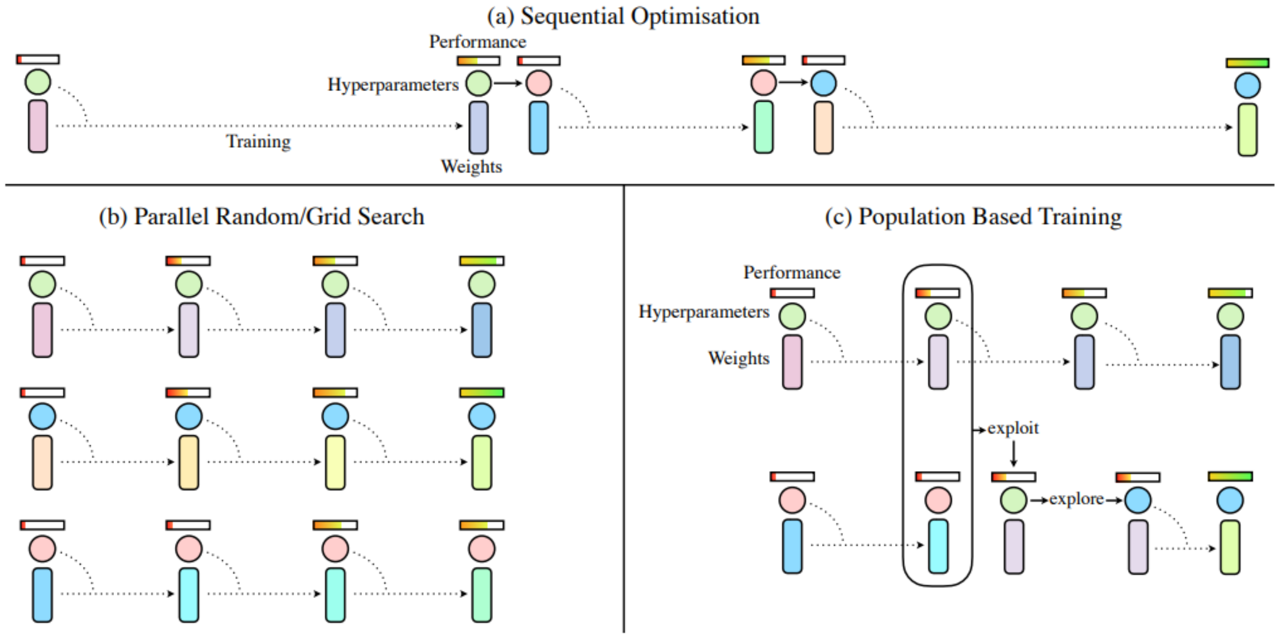
ALGORITHM APPROACH

Important Aspects:

- Complexity of deep learning models.
- Optimization of the search for hyperparameters in large-scale architectures.
- Population based training
- Generalized DL models
- Evolutionary algorithms in PBT
- Minimizing memory size in the produced model.
- Using AI to bring world-class specialist expertise to everyones, at lower cost.
- Expert care, anywhere.



Our Contribution: Hyperparameters Approach

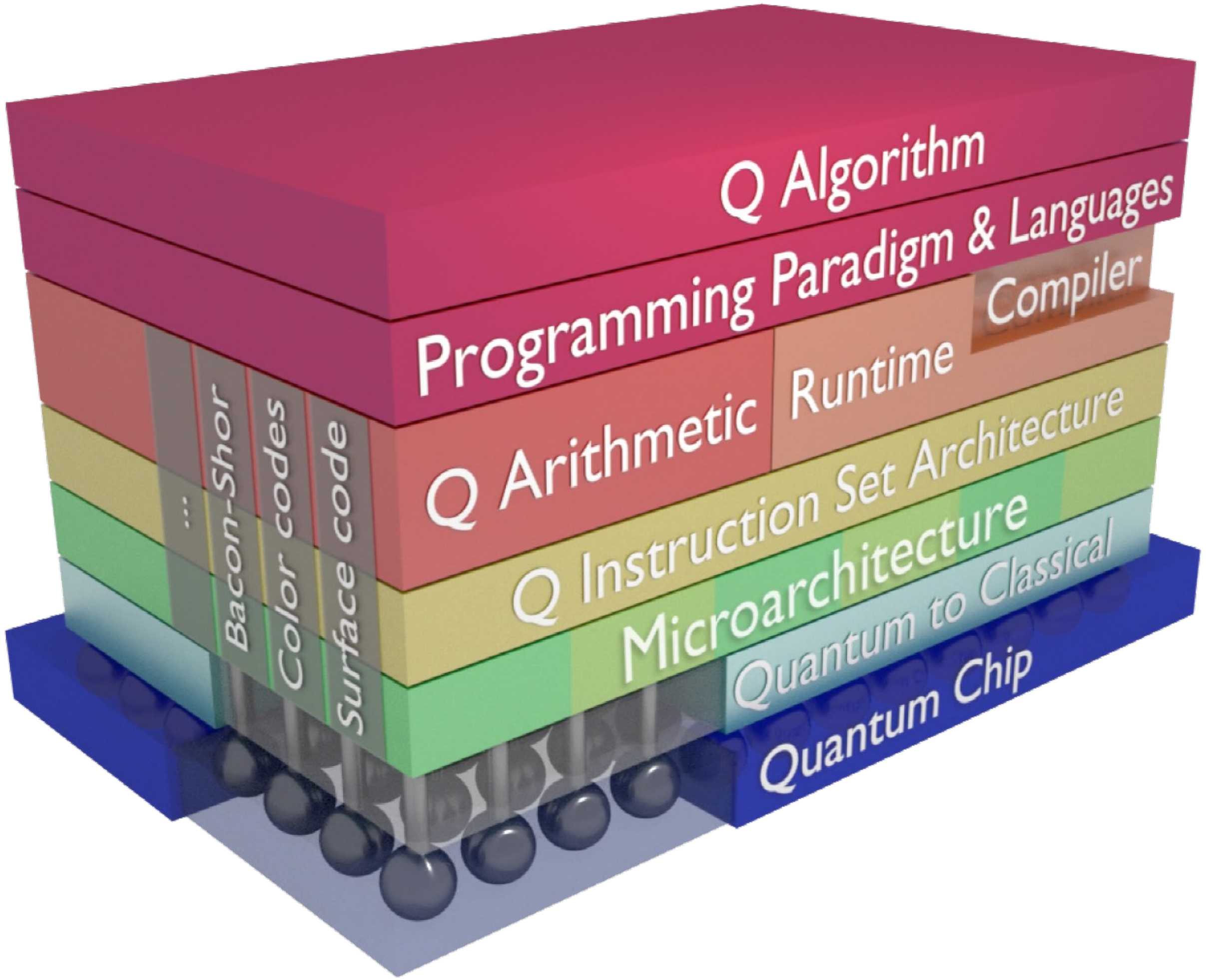


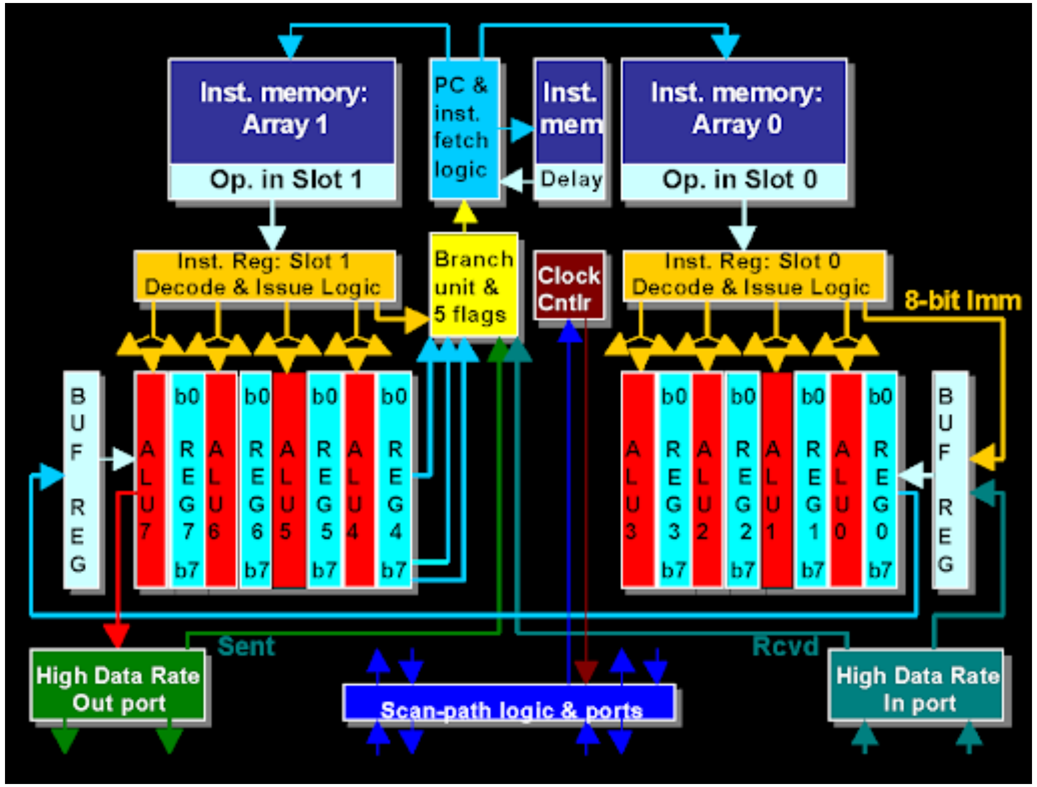
- Better understanding of PBT-based training mechanisms using distributed computational architectures
- Framework that implements efficient and scalable PBT mechanisms that, through evolutionary algorithms, allows finding generalizable models that minimize memory consumption.
- PBT techniques allow obtaining more optimal generalized models that consume less memory,.

Felix Mejia Thesis, Computational efficiency of the implementation of algorithms in Deep Learning applications for health in large-scale architectures in co-advising with M. Riveill and Collaboration with J. A. Garcia.

Universidad Industrial de Santander CONSTRUIMOS FUTURO Super Computación y Cálculo Científico UIS CAGE Computo Avanzado y a Gran Escala Advanced and Large Scale Computing Research group UNIVERSITÉ CÔTE D'AZUR i3S sophia antipolis Inria INVENTEURS DU MONDE NUMÉRIQUE UNIVERSITÄT BERN

Quantum Processor Unit Architecture*





*Simplest Approach

Quantum Computing Over HPC

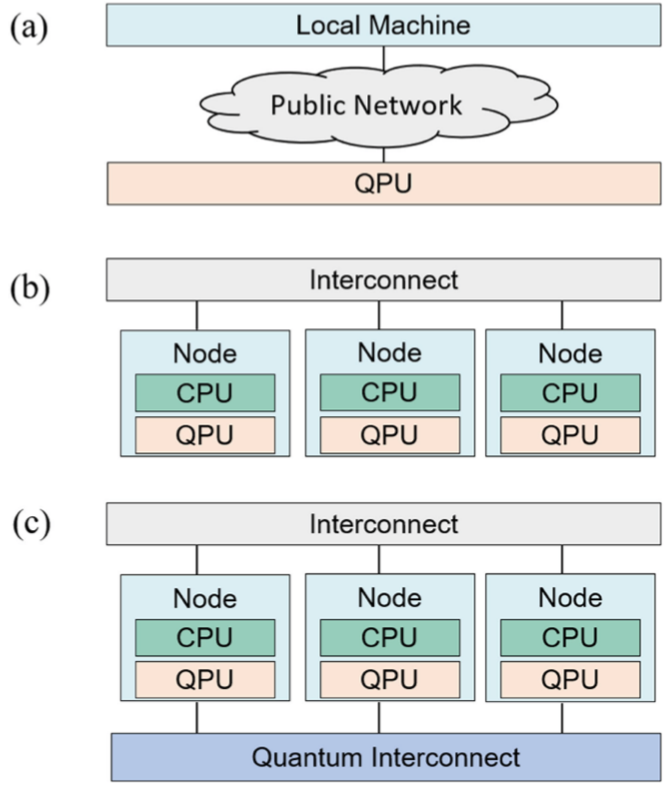


FIGURE 2. Three macroarchitectures for integrating quantum computing with conventional computing. (a) A local machine remotely accesses a QPU through public cloud network. (b) A network of quantum-accelerated nodes communicate through a common interconnect. (c) A network of quantum-accelerated nodes communicate through both conventional and quantum networks.

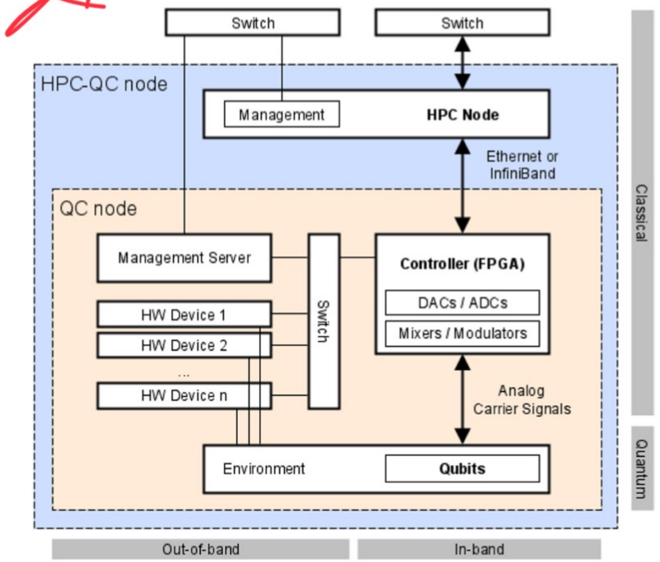


FIGURE 3. A component diagram representing the microarchitecture of a HPC-QC node with a common interconnect as depicted in Figure 2(b). The diagram shows the major components needed for the operation of a QPU within the HPC node infrastructure. Individual components are grouped into so-called out-of-band and in-band scopes and are placed on the left-hand and right-hand side of the figure, respectively. The QPU, which contains the qubits and is capable of processing quantum information, is depicted at the lower part, whereas classical information processing components are shown in the upper part of the figure. Several hardware (HW) devices control the QPU environment, which has a direct effect on qubit properties and thus the quality of execution of instructions.

DEPARTMENT: EXPERT OPINION

Quantum Computers for High-Performance Computing

Travis S. Humble, Alexander McCaskey, Dmitry I. Lyakh, and Meenambika Gowrishankar, *Quantum Computing Institute, Oak Ridge National Laboratory, Oak Ridge, TN, 37830, USA*

Albert Frisch, *Alpine Quantum Technologies, Innsbruck, 6020, Austria*

Thomas Monz, *Alpine Quantum Technologies, Innsbruck, 6020, Austria and also Institut für Experimentalphysik, Universität Innsbruck, Innsbruck, 6020, Austria*

Quantum computing systems are developing rapidly as powerful solvers for a variety of real-world calculations. Traditionally, many of these same applications are solved using conventional high-performance computing (HPC) systems, which have progressed sharply through decades of hardware and software improvements. Here, we present a perspective on the motivations and challenges of pairing quantum computing systems with modern HPC infrastructure. We outline considerations and requirements for the use cases, macroarchitecture, microarchitecture, and programming models needed to integrate near-term quantum computers with HPC system, and we conclude with the expectation that such efforts are well within reach of current technology.

High-performance computing (HPC) systems define the pinnacle of modern computing by drawing on massively parallel processing. This leading paradigm for HPC often relies on specialized accelerators and highly tuned networks to optimize data movement and application performance, whereby many computational nodes are connected by high-bandwidth networks to support shared information processing tasks. Existing computational nodes also support highly concurrent execution with multithreaded processing, and technology trends indicate that future node designs will integrate heterogeneous processing paradigms that include conventional central processing units (CPUs), graphics processing units (GPUs), field-programmable gate arrays (FPGAs), and other specialized processors.[1] The components of these future computational nodes must be tightly integrated to balance data movement with processing power and workload in order to optimize overall system performance.

By comparison, quantum computers (QCs) represent a young yet remarkable advance in the science and technology of computation that are often cited as rivals or successors to state-of-the-art conventional high-performance computing (HPC) systems. The source of this proposed advantage of QCs derives from quantum information processing in which information is encoded in the quantum state of physical systems such as atoms, electrons, and photons.[2] These quantum physical systems present the unique features of quantum coherence and quantum entanglement that permit quantum computing to reduce exponentially the computational time and memory needed to solve many problems from chemistry, materials science, finance, and cryptanalysis among other application domains. The advantage afforded to quantum computing is therefore aptly named the "quantum computational advantage," and there is now a fervent effort to realize quantum computing systems that demonstrate this advantage. Notably, recent efforts have focused on besting the world's leading HPC systems to great effect.[3,4,5]

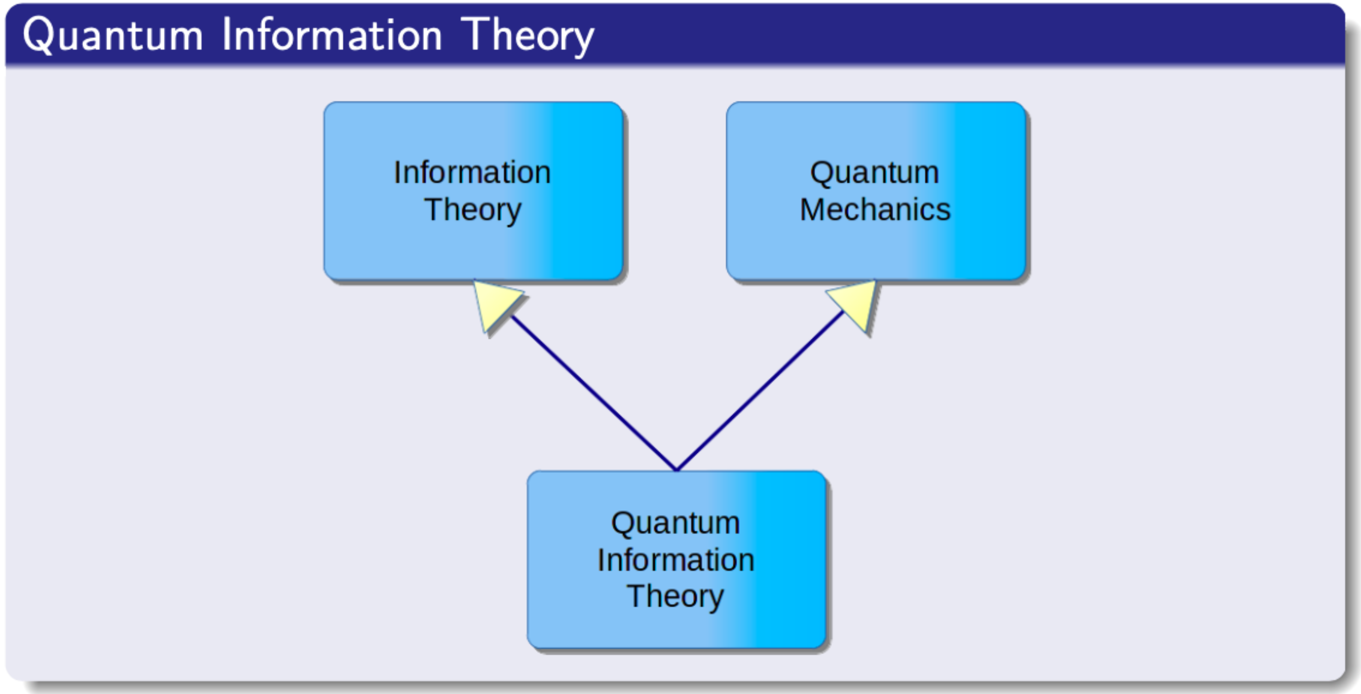
Many of the most promising applications of quantum computing overlap strongly with existing applications of HPC,[6] which begs the question of how QCs may be integrated with modern HPC to accelerate these

Please see the Acknowledgements section at the end of the article for a special statement regarding the copyright.

0272-1732 © 2021 IEEE
Digital Object Identifier 10.1109/MM.2021.3099140
Date of current version 14 September 2021.

Open Question 4: And Quantum Computing?

Our Contribution: Quantum Computing Theory for Quantum Computing Applications



UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE

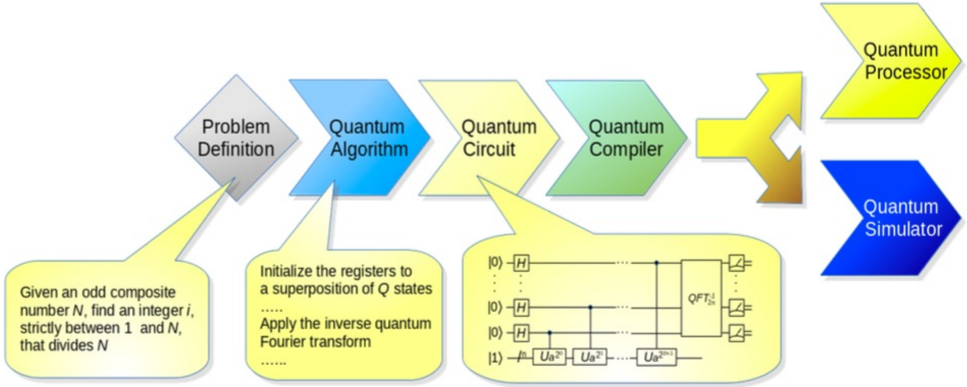
CReSTIC

Universidad Industrial de Santander UIS CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS

CAGE

Computo Avanzado y a Gran Escala
Advanced and Large Scale Computing
Research group


Problem Definition
Quantum Algorithm
Quantum Circuit
Quantum Compiler
Quantum Processor
Quantum Simulator
Given an odd composite number N, find an integer i, strictly between 1 and N, that divides N
Initialize the registers to a superposition of Q states
.....
Apply the inverse quantum Fourier transform
......

The term **quantum algorithm** is generally used for those algorithms that incorporate some essential feature of **quantum computing**, such as superposition or entanglement. By using this special features, we can speed up significantly the calculation, that is called **quantum parallelism**.

G. Diaz PhD. Thesis about Classica Resources Consumption in Quantum Computing Simulators (Co-Advising with L. A. Steffenel)

Final Note: A New Approach of the HPC/HPDA Platforms for Unified Advanced Computing Support (! Or ?)

Why an Advanced Computing Platform Vision (and Not Only HPC)?

(Inspired by the Accelerated/Hybrid Computing World)

Programming Approaches

Libraries

Accuracy and Acceleration

Directives

Easily Use

Programming Interpreters

Programming Languages

Maximum Flexibility

Development Environment

Versions Store
Developer Hubs, Community Platforms, Pipeline Environments

IDE
Linux, Mac and Windows Debugging and Profiling

Debuggers, Profiling and Performance Visualizers

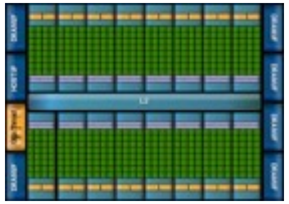
(Open) Compiler Tool Chain

Linkers, Assembly in Open Source or Corporate Development

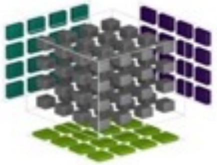
Enables compiling new languages to platforms, and languages to other architectures

System (HW/MW/SW) Capabilities

Post Moore and Non-Von Newman

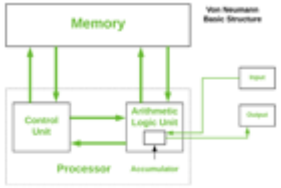


Novel Abstractions and Models

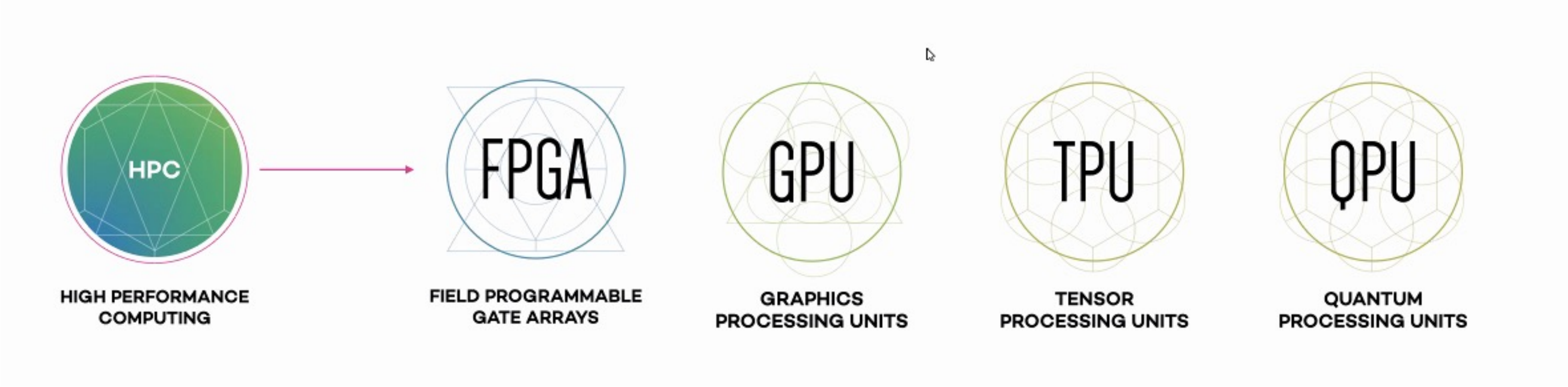


New Computing

Classical Computing

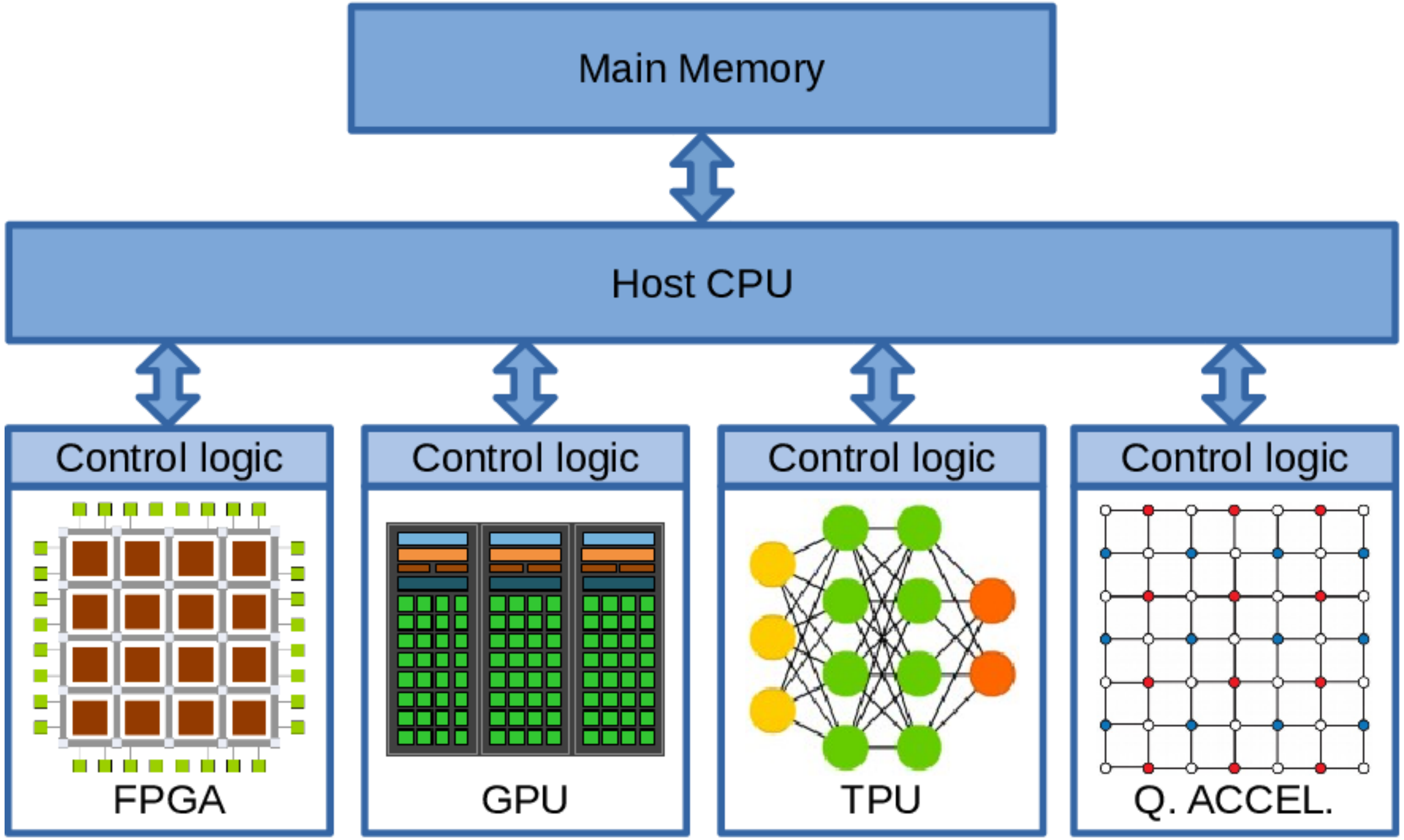


The Many-Architectures Challenge: How to exploit better Advanced Computing Architectures (for All)?


HPC
HIGH PERFORMANCE COMPUTING
FPGA
FIELD PROGRAMMABLE GATE ARRAYS
GPU
GRAPHICS PROCESSING UNITS
TPU
TENSOR PROCESSING UNITS
QPU
QUANTUM PROCESSING UNITS

From Why Quantum Computing is Integral to the Future of HPC'. By William "Whurley" Hurley, CEO of Strangeworks

Conclusion: HPC/Advanced Computing Systems



From : Bertels, K., Sarkar, A., Hubregtsen, T., Serrao, M., Mouedenne, A.A., Yadav, A., Krol, A.M., Ashraf, I., & Almudever, C.G. (2020). Quantum Computer Architecture Toward Full-Stack Quantum Accelerators. IEEE Transactions on Quantum Engineering, 1, 1-17.

Questions?

CARLA 2023
LATIN AMERICA HIGH PERFORMANCE COMPUTING CONFERENCE
Cartagena de Indias, Colombia
September 18-22

Universidad Industrial de Santander
CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS

CAGE

Computo **A**vanzado y a **G**ran **E**scala
Advanced and Large Scale Computing
Research group

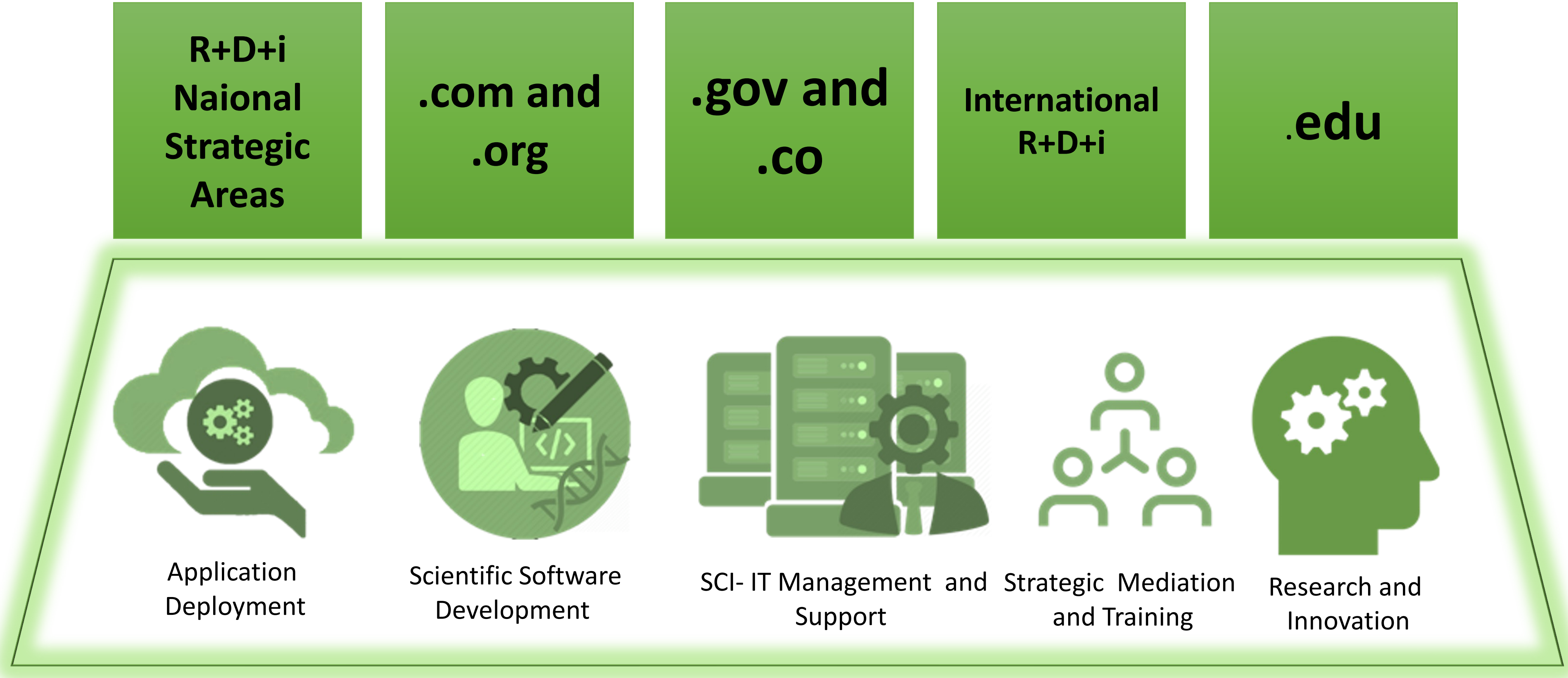
SCALAC

RISC2

LaRedCCA

@carlosjaimebh

What is SC3UIS?



R+D+i Naional Strategic Areas

.com and .org

.gov and .co

International R+D+i

.edu

Application Deployment

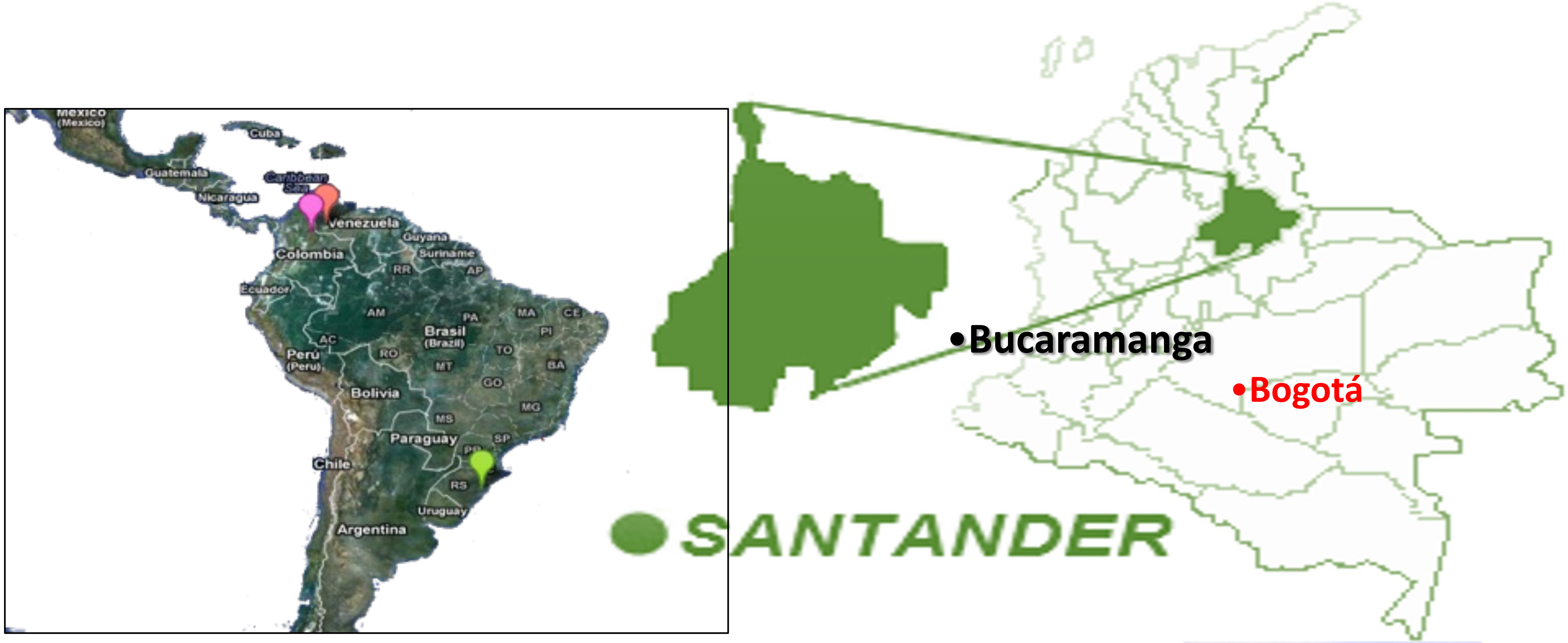
Scientific Software Development

SCI- IT Management and Support

Strategic Mediation and Training

Research and Innovation

Where is SC3UIS?



Universidad Industrial de Santander
CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS

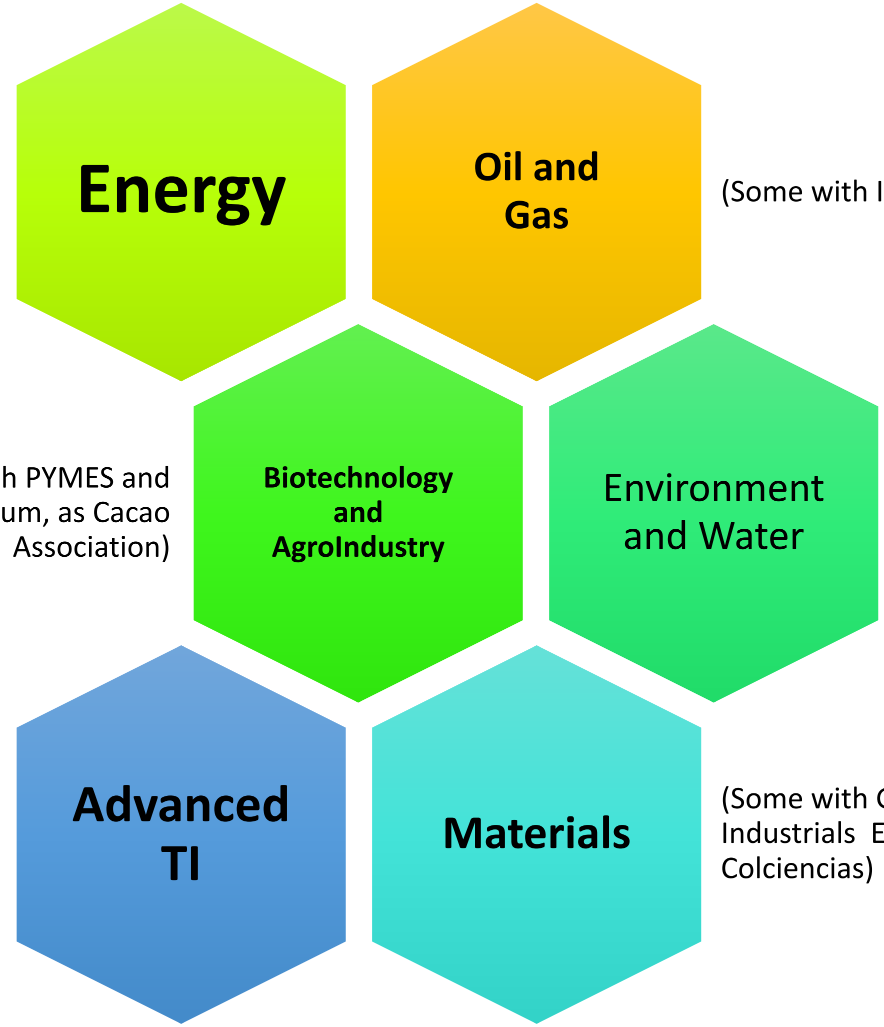
Guatiguará
PARQUE TECNOLÓGICO

SC3UIS at UIS (@UIS) and Guatiguara Technology Park (@PTGuatiguara)

- Founded in 1948 (Following the German /French Polytechnic Model)
- Public State University
- 8 Campus in the Department
  - 4 at Metropolitan Zone of Bucaramanga
  - 4 in Other Regional Cities (Barrancabermeja, Socorro, Malaga, Barbosa)
- 25000 Students (2300 Postgraduate Students)
- 530 Faculty (4 at SC3UIS)
- ***Support and R+D+I and General Training of SC3UIS***

- Guatiguara Site was created in 1989 (New Foundation at 2007 as Technology Park)
- 8 Industrial Corporations
- 3 National Labs
- National Core Repository and ANR Site
- 5 Centers
- ***High Performance Computing Data Center***
  - ***GUANE-1 and CHAMAN are here!***
- ***R+D+I and Specialized Training Site of SC3UIS***

R+D+i Axes (@PTGuatiguara)



(Some with ICP-Ecopetrol)

(Some with PYMES and Consortium, as Cacao Association)

(Some with Gouvernment and Industrials Entities, i.e. Colciencias)

2017 Important Numbers

4 Patents
5 Spin Off in Incubation Process (Potentially for 2018 more than 10)
3 Big International Collaborations (more than 5M USD)

2018 New Axes:
Healthcare
New Generation of Automotive Motors
Human and Social Development

Gracias…

Follow us: @sc3uis

Universidad Industrial de Santander

CONSTRUIMOS FUTURO

Super Computación y Cálculo Científico UIS