High Performance Quantum Computing

Andrés Gómez Tato, PhD.

AQADOC Workshop, Paris

Ocr. 2nd, 2024

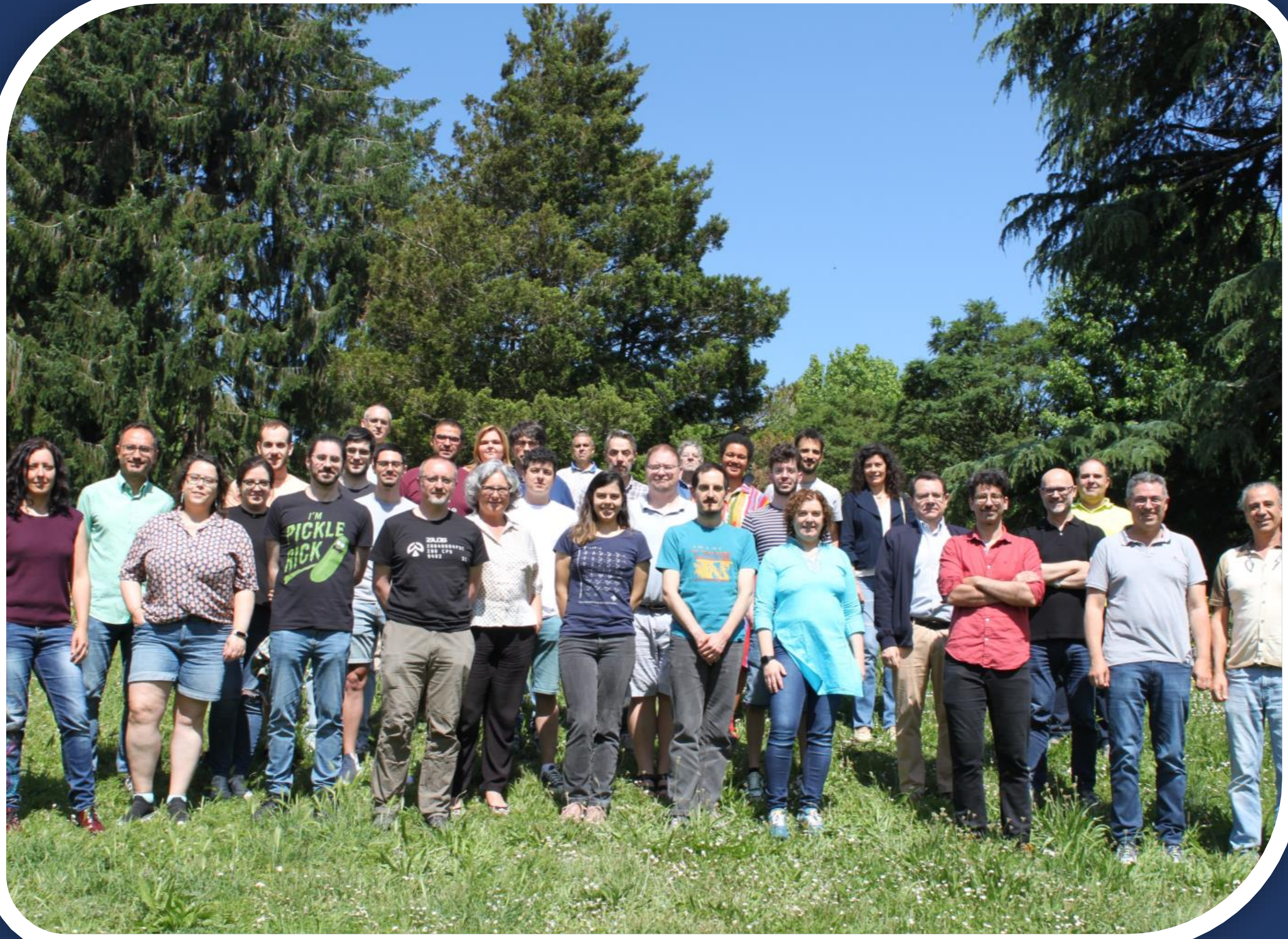
CESGA
GALICIA SUPERCOMPUTING CENTER

Who are we?

- **<u>Mision</u>**: contribute to the advance of science and technology via research and application of high performance computing and comunications
- CESGA offered supercomputing services for R&D&I to more than **1000 researchers** from Universities, National Research Centers, etc
- **Research Center** in the areas of Classical Computing, Quantum Computing, Life Sciences, and Earth Sciences

CESGA Staff

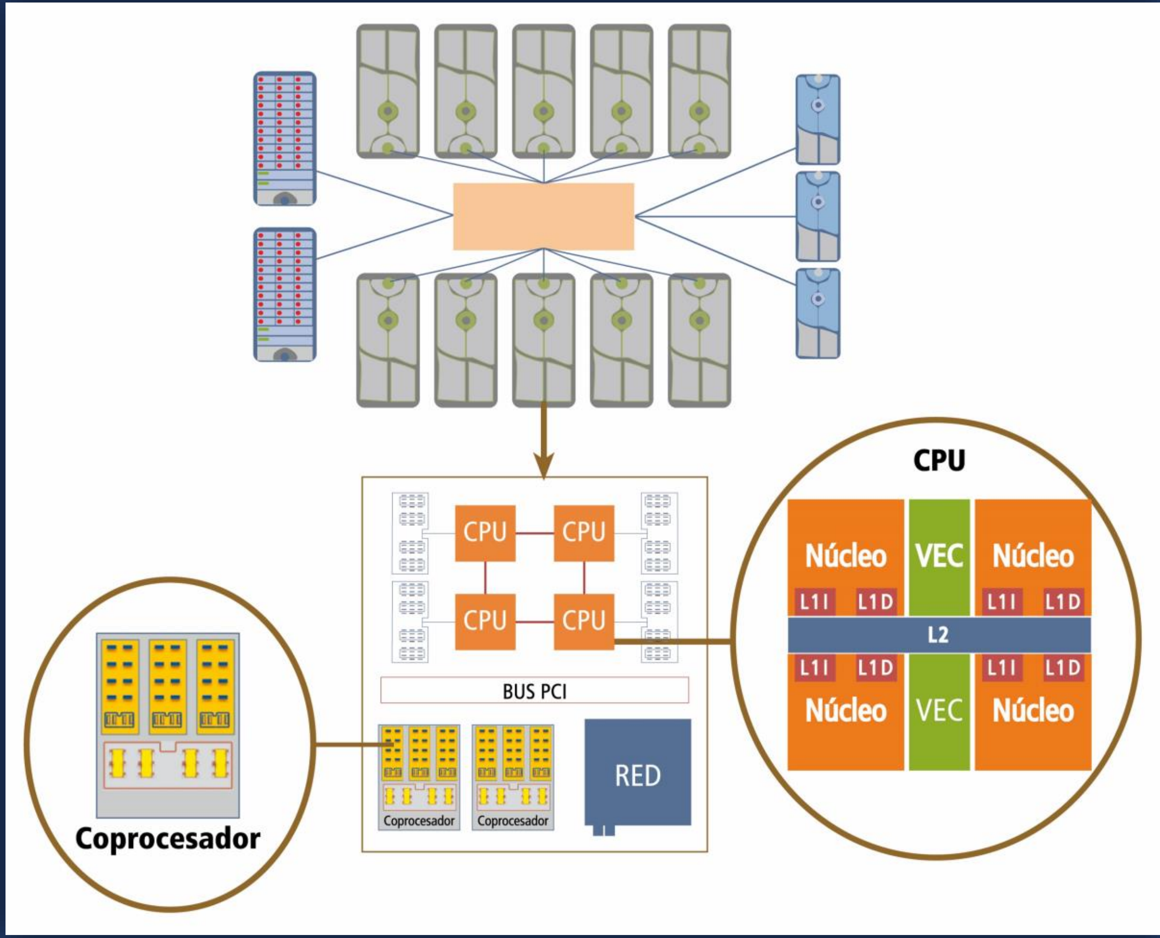
- **Around 50 people** in total (and **growing**!)
- **Technical staff**: Physicists, Computer Scientists, Telecomunicaction engineers, Matematicians, etc.
- Most departments **do research** and/or **support**



1 What is a classical supercomputer?

Supercomputer

Finisterrae III: 22.848 cores, 157 GPUs, 126TB RAM



Finisterrae III: 22.848 cores, 157 GPUs, 126TB RAM

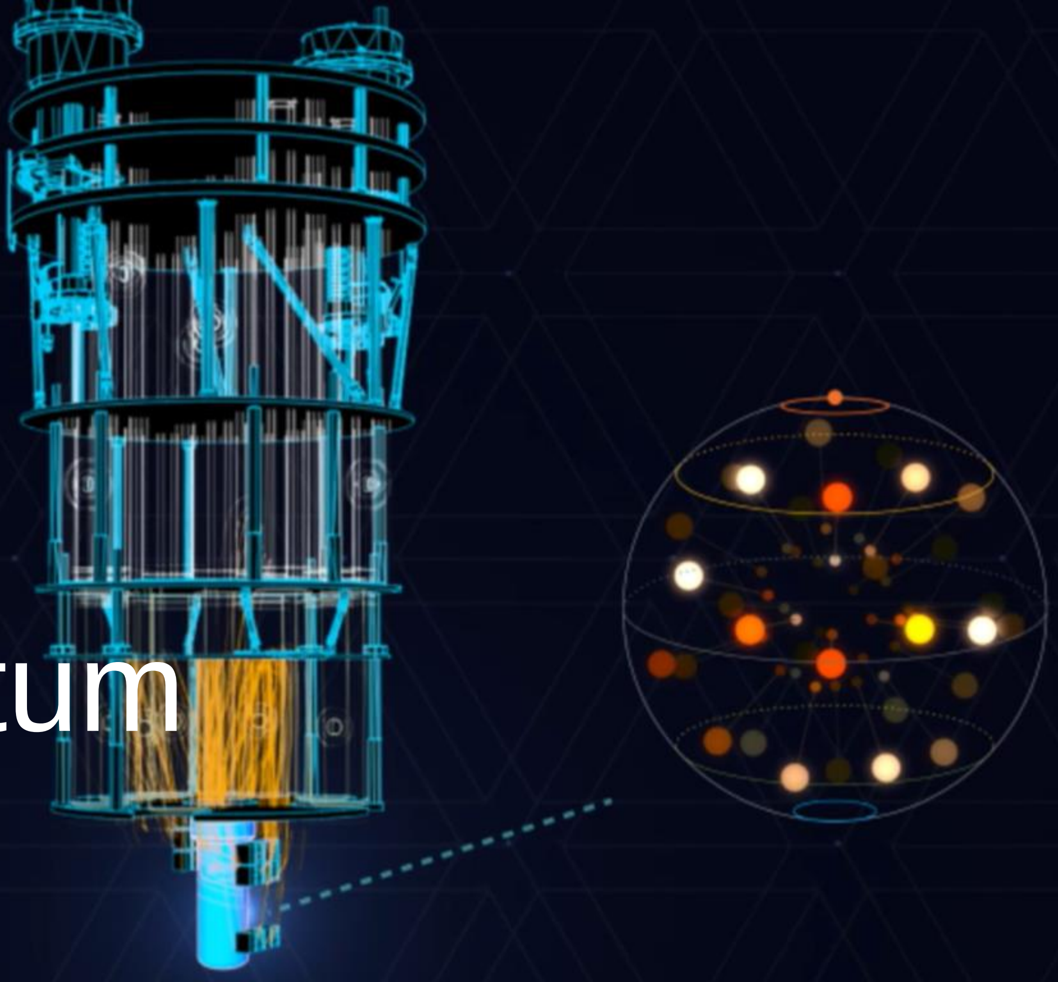
	Thin nodes	GPU nodes	Vis nodes	Transfer nodes	Fat nodes	QLM nodes
QUANTITY	256	66	16	2	17	Eviden Q-Activa emulator up to 30 qubits
PROCESSOR	2xIntel Xeon Ice Lake 8352Y 32 Cores 2.2 GHz					
MEMORY	256 GB				2 TB (16) 8 TB (1)	
LOCAL DISK	960 GB SSD NVMe				1920 GB SSD NVMe	
GPU		2x Nvidia A100 (64) 5x Nvidia A100 (1) + 256 GB RAM 8x Nvidia A100 (1) + 256 GB RAM	Nvidia T4			
NETWORK	Infiniband HDR 100					

- ❑ 714 Processors
- ❑ 22.848 cores
- ❑ 141 NVIDIA A100
- ❑ 126 TB RAM
- ❑ Infiniband HDR 100
- ❑ **4,36 PetaFLOPS**

XUNTA DE GALICIA · CSIC · Unión Europea — Fondo Europeo de Desarrollo Regional "Una manera de hacer Europa" · Gobierno de España · Ministerio de Ciencia e Innovación · RES Red Española de Supercomputación

CESGA
GALICIA SUPERCOMPUTING CENTER

2 What is Quantum Computing?



Many types of Quantum Processing Units

- Quantum simulator [1],
- Adiabatic Quantum Computer [2],
- Topological Quantum Computer [4].
- Continuous Variable Quantum Computer [5],
- **Universal Quantum Computer or Digital Quantum Computer [3],**
- Analog-digital quantum computer ,
- etc.

[1] Reviewed in Georgescu , I.M., Ashhab , S., & Nori , F. (2014). Quantum simulation. *Reviews of Modern Physics , 86* (1), 153–185.
http://doi.org/10.1103/RevModPhys.86.153 **arXiv:1308.6253**
[2] Reviewed in Albash , T., & Lidar , D.A. (2016). Adiabatic Quantum Computing. arxiv:1611.04471
[3] Proposed in Deutsch, D. (1985). http://doi.org/10.1098/rspa.1985.0070 and
Deutsch, D. (1989). http://doi.org/10.1098/rspa.1989.0099
[4] Lahtinen V., Pachos JK. SciPost Phys. 3, 021 (2017) **arXiv:1705.04103**
[5] Lloyd S. & Braunstein, AL Phys.Rev.Lett. 82 (1999) 1784-1787. arXiv:quant-ph/9810082

Some important facts about QC/DQC

- **NO, it will not multiply faster (at least in the short term)**
  - Think in a new type of algorithms.
  - Oriented to solve specific, difficult problems for classical computing.
  - In most cases, small input and a lot of computation (it is NOT always easy to insert classical information into quantum states)

- **Possible advantages (to be demonstrated empirically):**
  - Solve problems that a classical computer could not (*Quantum Supremacy*). Example: breaking RSA keys (Shor's Algorithm).
  - Obtain a better solution than classical algorithms. Examples: Optimization in Manufacturing; Quantum Machine Learning.
  - Obtain an equal or equivalent solution, but in less time .
  - Obtain an equal, equivalent or even slightly worse (but usefull) solution, consuming less energy

- **All algorithms are hybrid (quantum-classical)**
  - The scalability of quantum algorithms may be conditioned by classical computing (or even not exist).

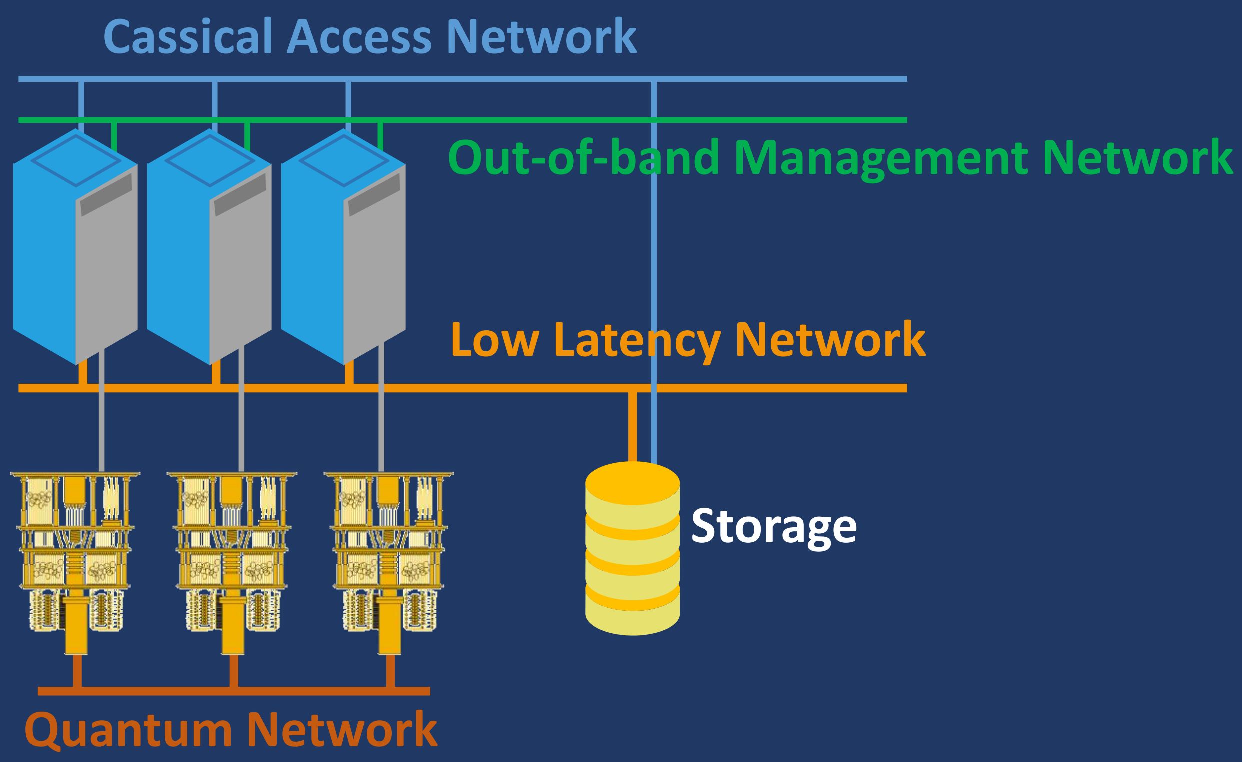
3 High Performance Quantum Computing

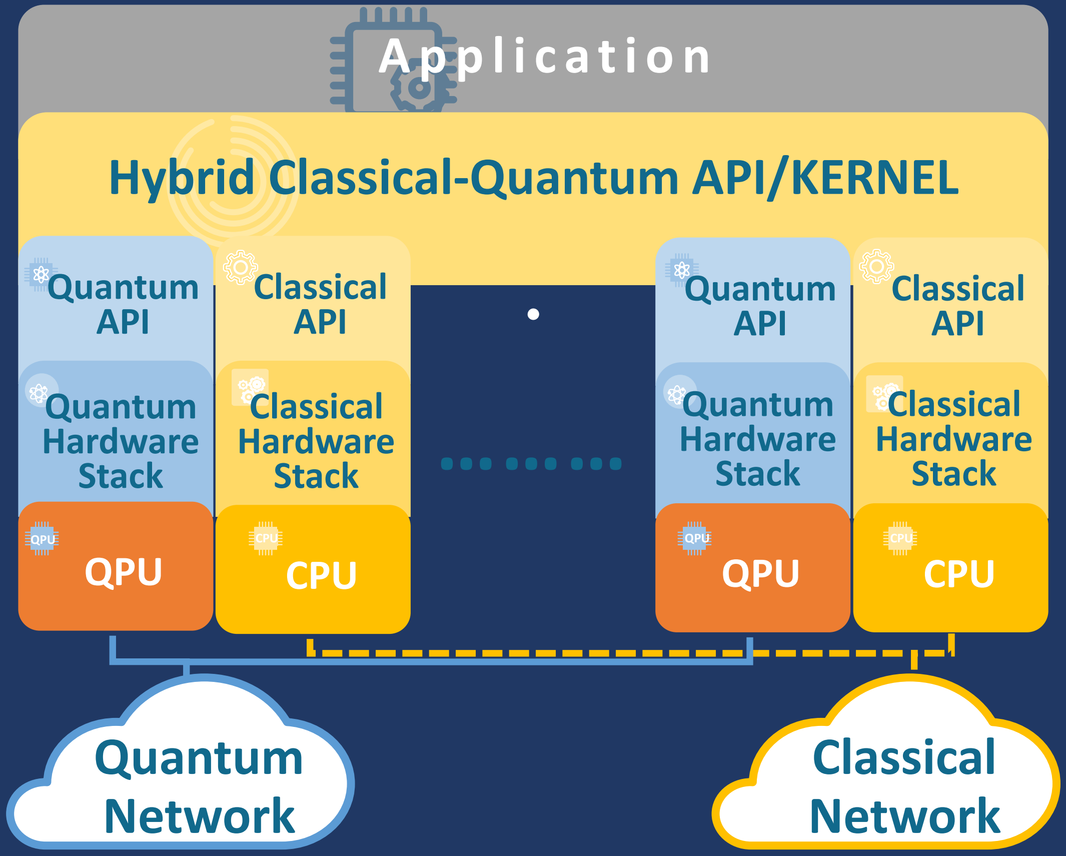
The future?

Rebranding...

- CPU → "Classical" Processing Unit
- GPU → Graphical Processing Unit
- **QPU → Quantum Processing Unit**
  - **i.e., another accelerator of an HPC node**

High Performance Quantum Computers

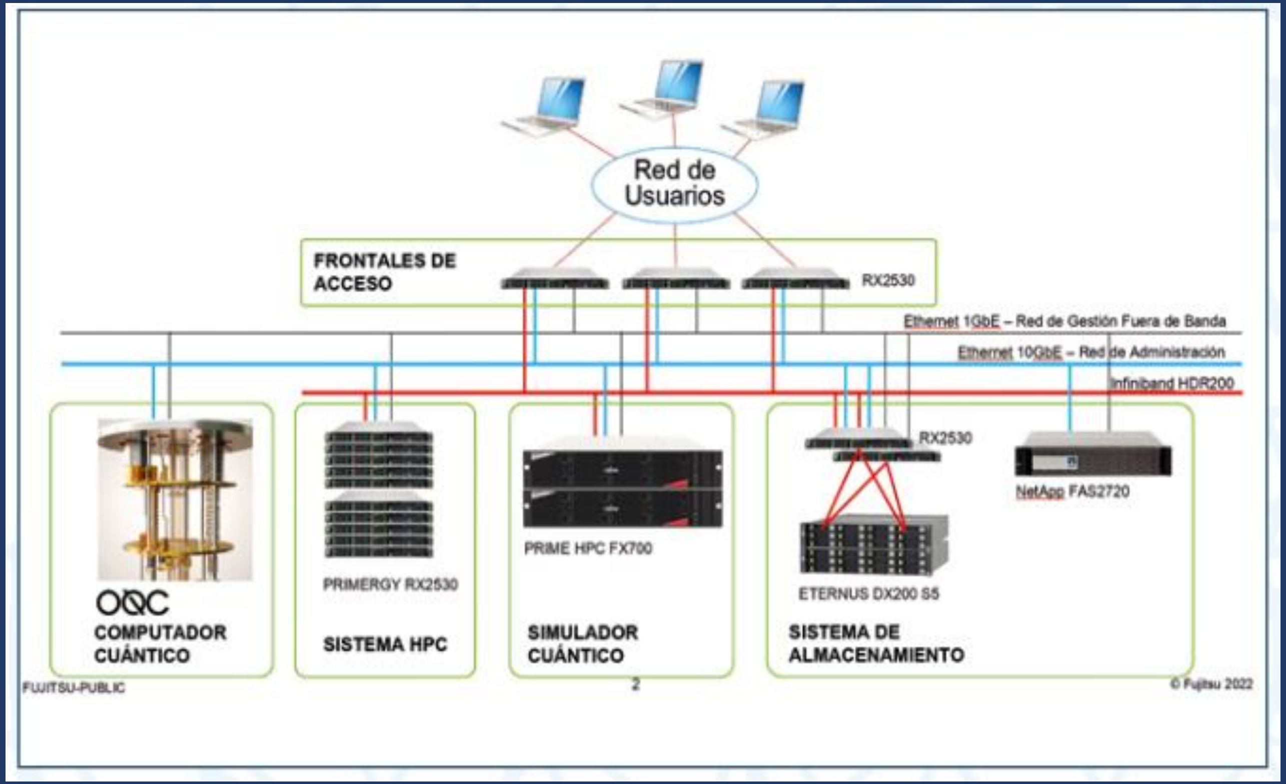




IN A SUPERCOMPUTER CENTER, RESOURCES (CPU/CORE/NODE, GPU, QPU, FPGA, ETC.) ARE ALLOCATED EXCLUSIVELY TO ONE PROCESS/JOB

Qmio

A PILOT FOR A HPQC INFRASTRUCTURE



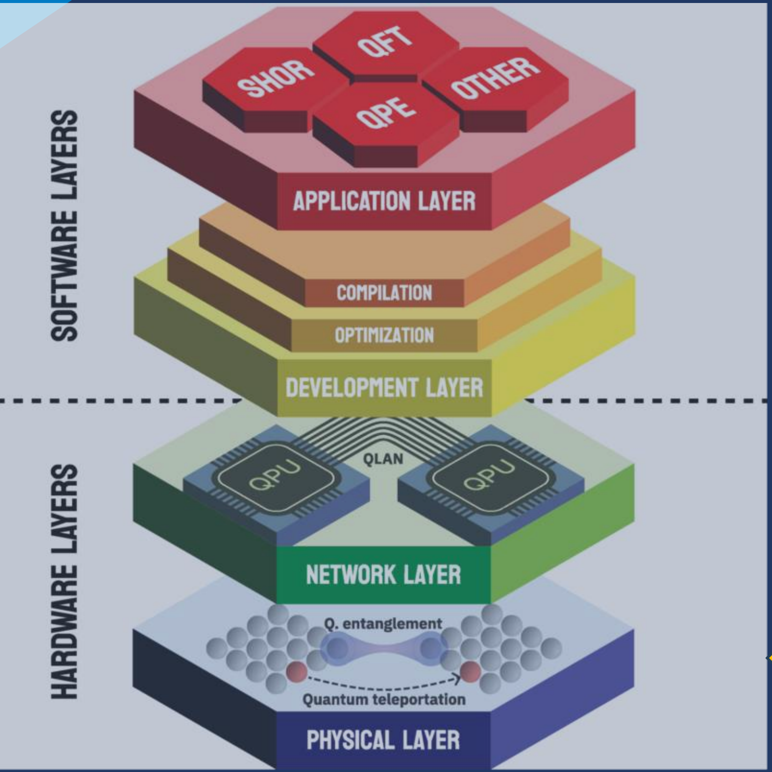
Despliegue de una infraestructura basada en tecnologías cuánticas de la información que permita impulsar la I+D+i en Galicia. Operación financiada por la Unión Europea, a través del FONDO EUROPEO DE DESARROLLO REGIONAL (FEDER), como parte de la respuesta de la Unión a la pandemia de la COVID-19.
PROGRAMA OPERATIVO
FEDER
2014-2020

Una manera de hacer Europa

AXENCIA GALEGA DE INNOVACIÓN
UNIÓN EUROPEA
Fondos Europeos

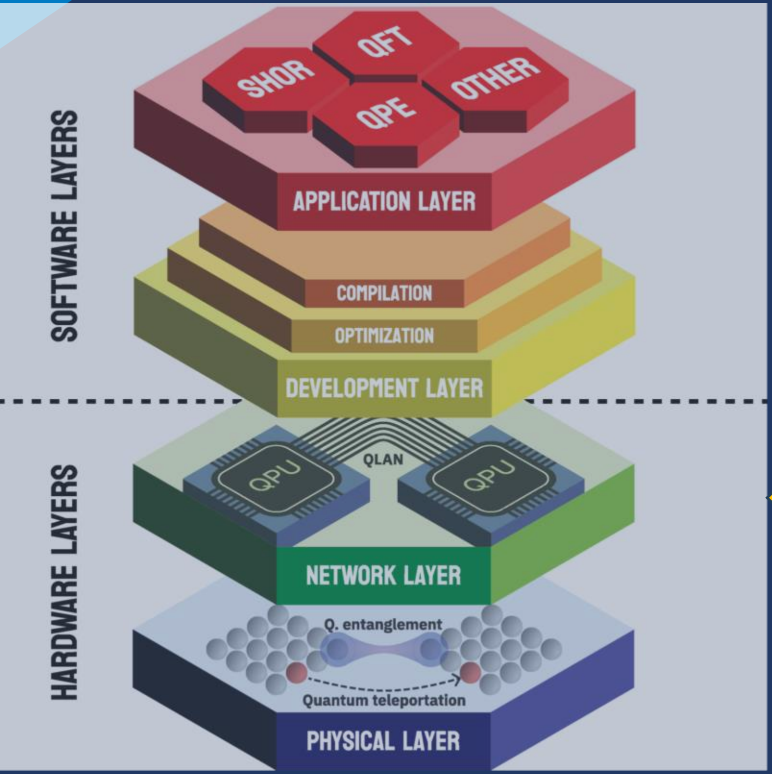
4 Supercomputers and Distributed Quantum Computing

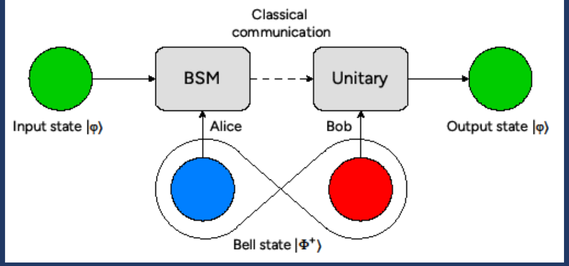
QPUs in a Supercomputing center



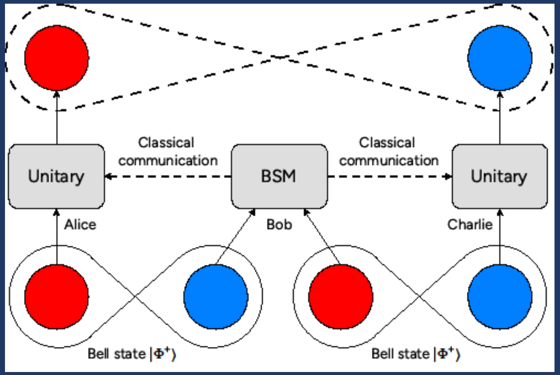
- Multiple hardware available.
  - Whis is the best?
  - Should we support many?
- Calibration.
  - CESGA makes a daily calibration plus one weekly
- Stability. Can we be running for months/years ?
- **Reproducibility. Produce the same results always**
- External influences. Is there any external factors that can affect the QPUs? (cosmic rays, visits to the computer room, maintenance works – hammers, etc. -, etc.)
- Control of temperatures.
- Aging. How long will a QPU work?
- Etc.

Network

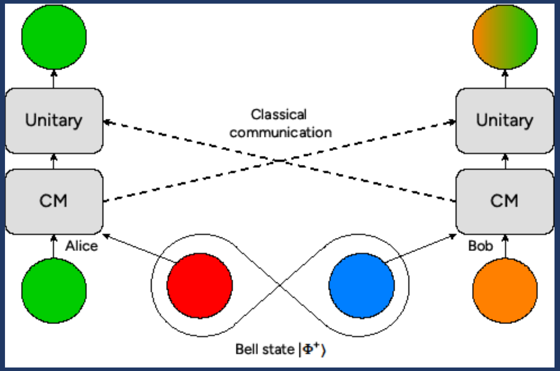




Teleporting/Teledata

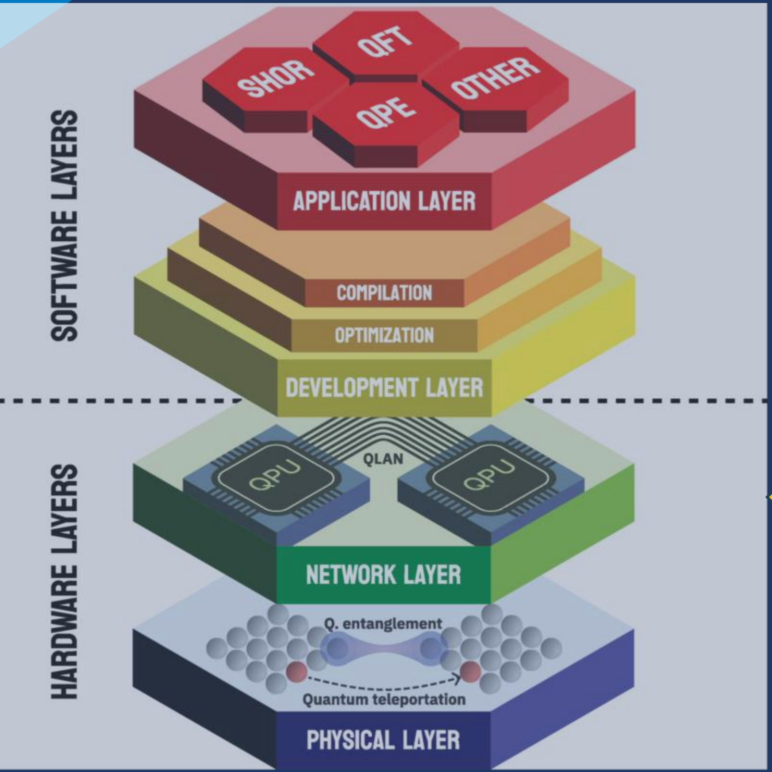


Entanglement Swapping



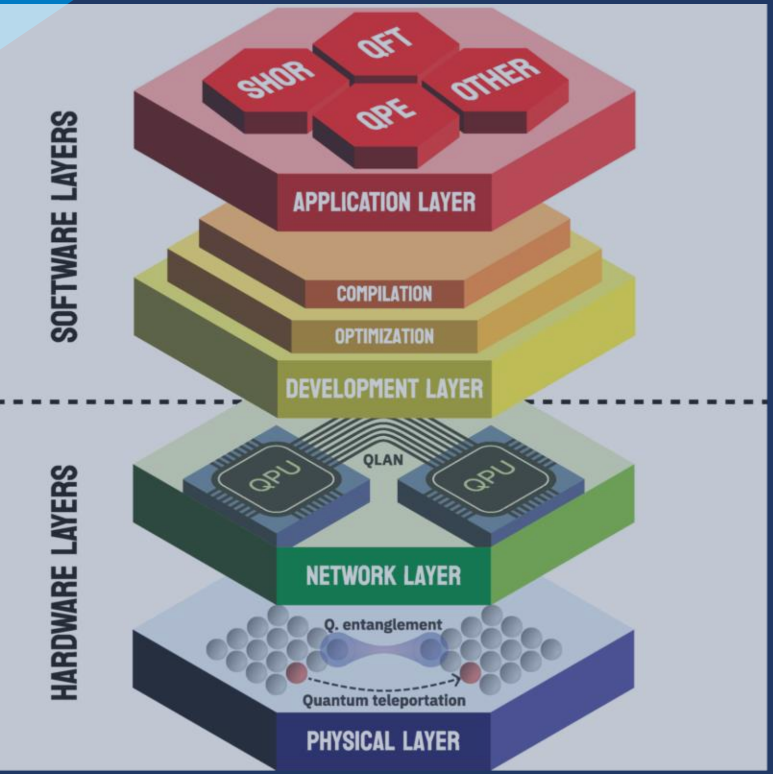
Telegate

Networking. Some research issues



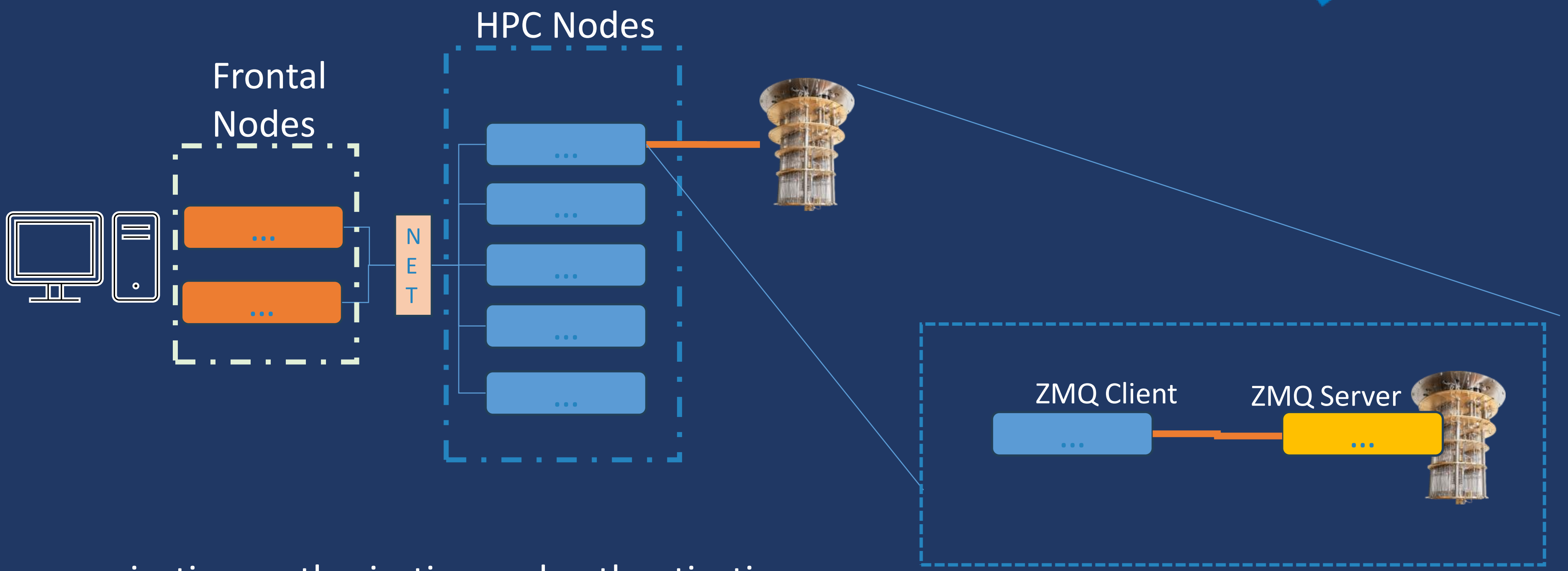
- Transduction
- Which network should be used: high-speed classical network or dedicated network
- **Architecture: Switches, routers, ?**

Integration CPU-QPU



- Conectivity
- Programming models

Qmio



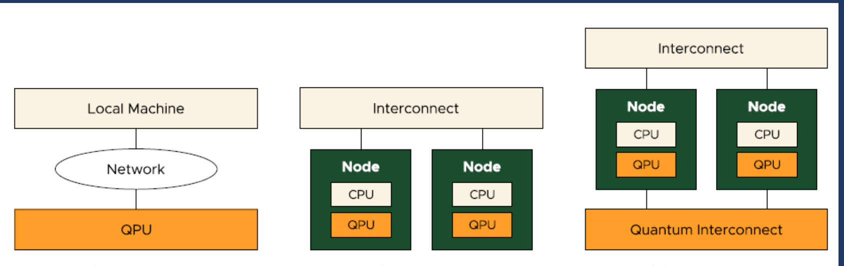
- Remove communication, authorization and authentication layers
- Remove compiling on QPU: program must be compiled in classical node before – **Compile once, execute many**

- Supports:
  - Partially OpenQASM 3.0 (2.0)
  - QIR

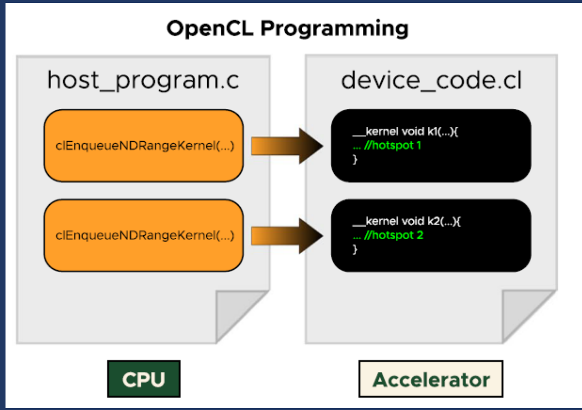
In collaboration with: FUJITSU OQC

Ref: To be published soon

Programming High Performance Quantum Computers



(a) Scheme 1. (b) Scheme 2. (c) Scheme 3.



The Journal of Supercomputing (2024) 80:11682–11703
https://doi.org/10.1007/s11227-023-05879-9

Check for updates

QPU integration in OpenCL for heterogeneous programming

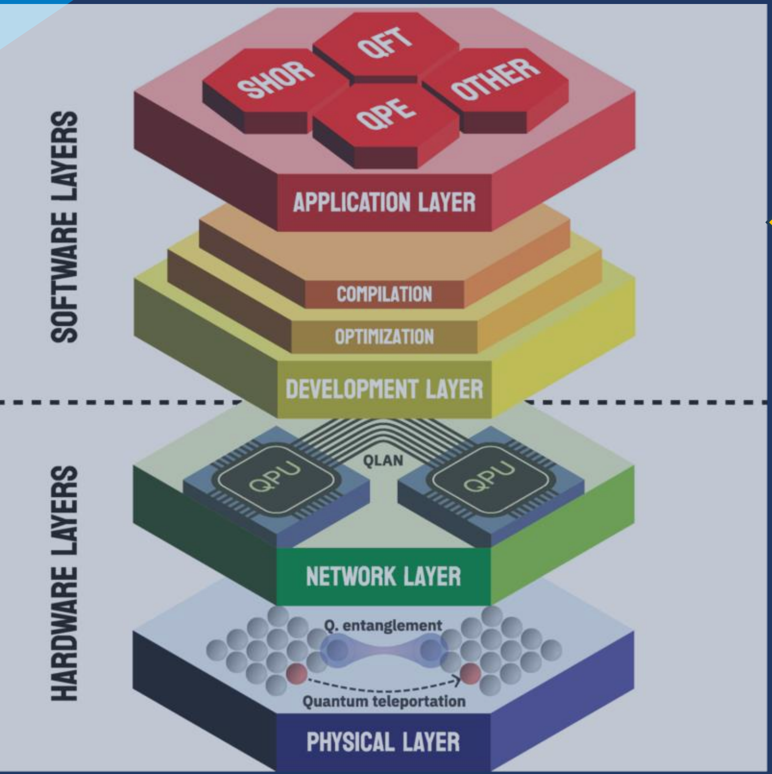
Jorge Vázquez-Pérez[1] · César Piñeiro[1] · Juan C. Pichel[1] · Tomás F. Pena[1] · Andrés Gómez[2]

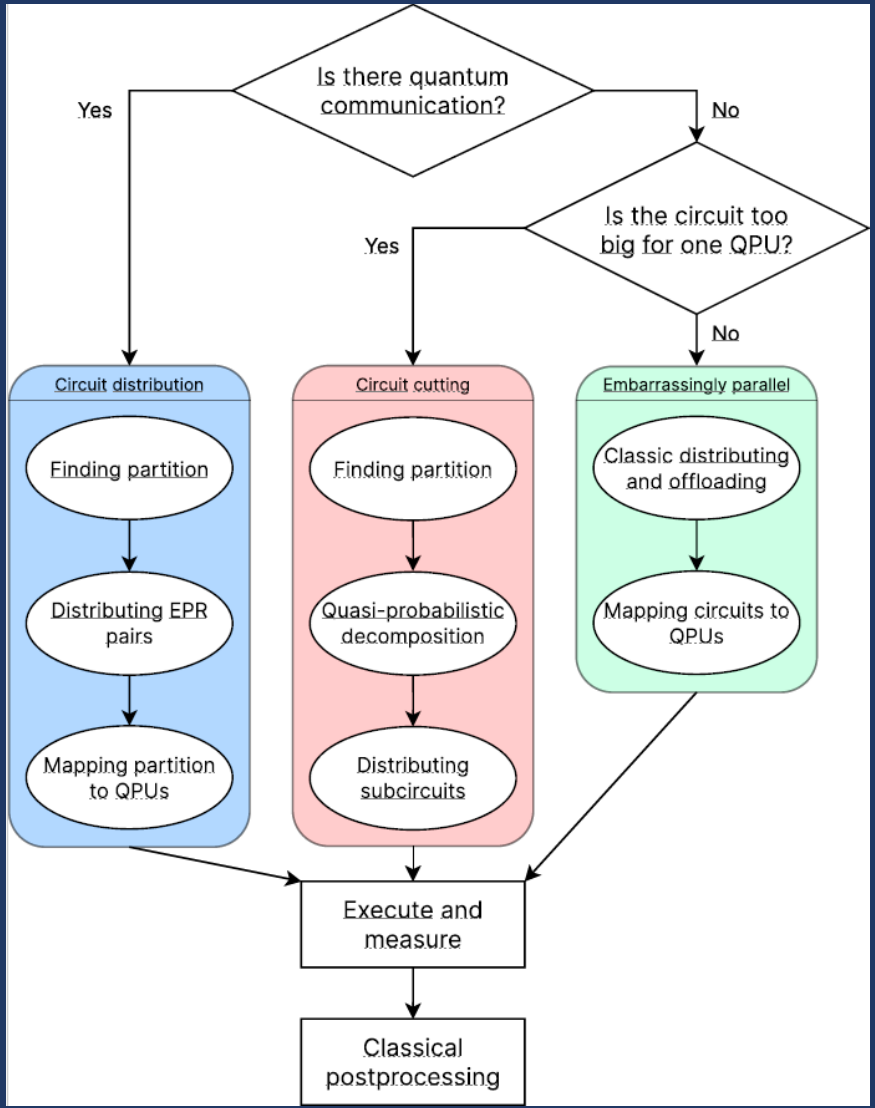
Accepted: 23 December 2023 / Published online: 31 January 2024

DOI:10.1007/s11227-023-05879-9

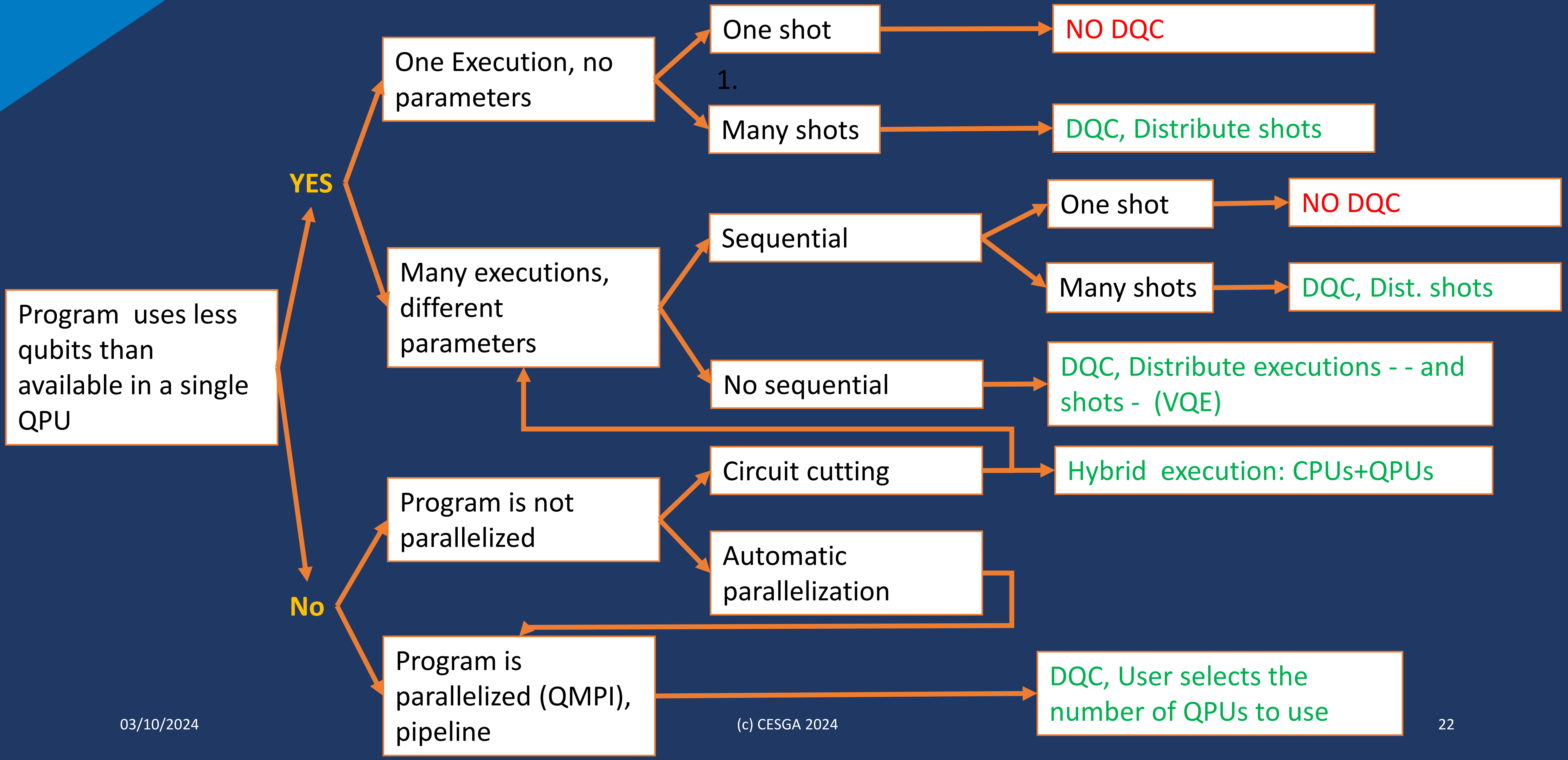
- Standard programming framework for accelerators
- Almost independent of the hardware: QIR
- One program, many systems.
- Demonstrated using QULACS emulator for QPE and Shor.
- Next steps:
  - Integrate with Qmio using QIR
  - Move from OpenCL to SYCL

Compilation – Execution of Programs





Execution on Distributed Quantum Computers



A possible simple example for DQC

- QAOA:
  - 32 qubits
  - 8 layers → 16 parameters
  - Optimizer: Differential Evolution, 32 individuals

- Available infrastructure:
  - 8 QPUs, each one attached to one classical node.
  - All QPUs with the same architecture.
  - 36 Qubits/QPU

- Distributed allocation:
  - 4 individuals/QPU
  - No parallelization on shots.
  - Model: master/workers

Remarks on Execution on Distributed Quantum Computers

- This is a simple scenario. More complex scenarios are possible.
  - Final users have a big imagination.

- In an HPC center, usually user selects the resources to use for each execution.
  - DQC program/schedulers must adapt to this selection

- Several models of QPU can be available/can be used by the program
  - QPUs can have different capabilities/capacities

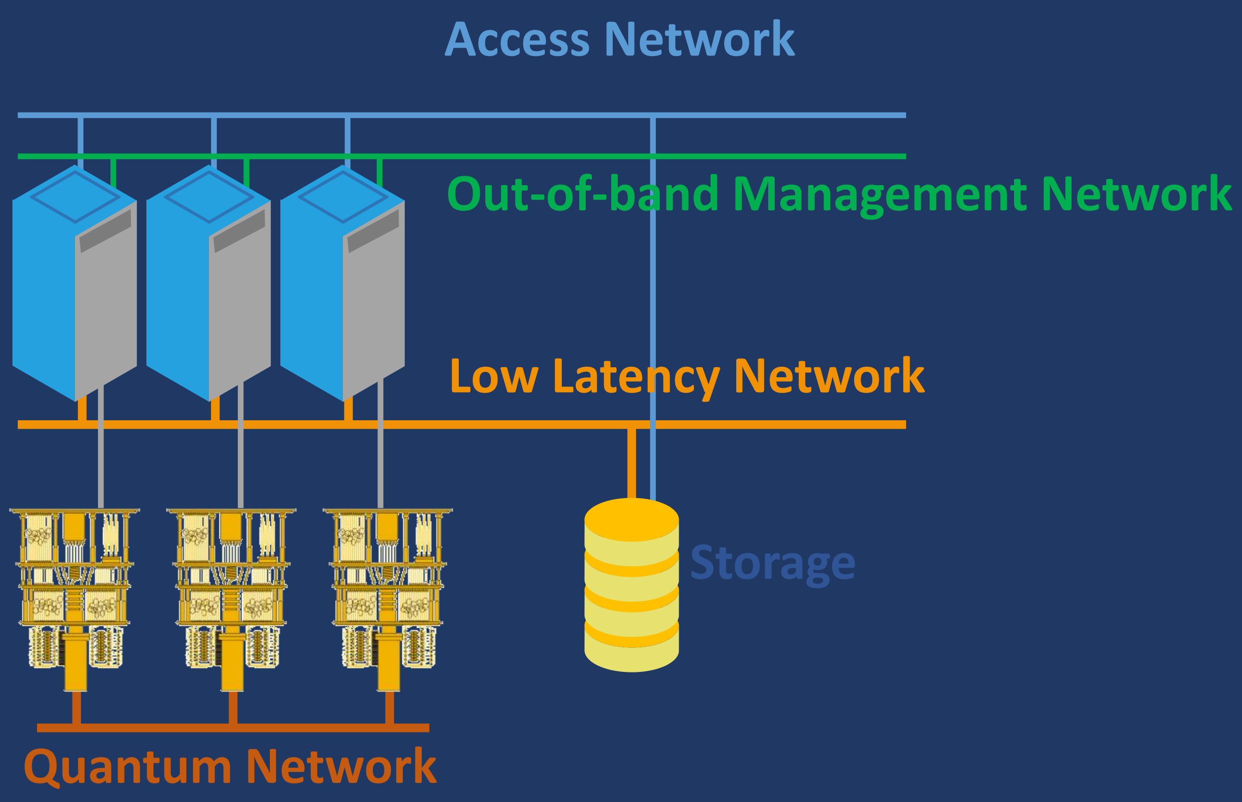
- DQC means also Distributed Classical Computing: Classical resources must be allocated as well.

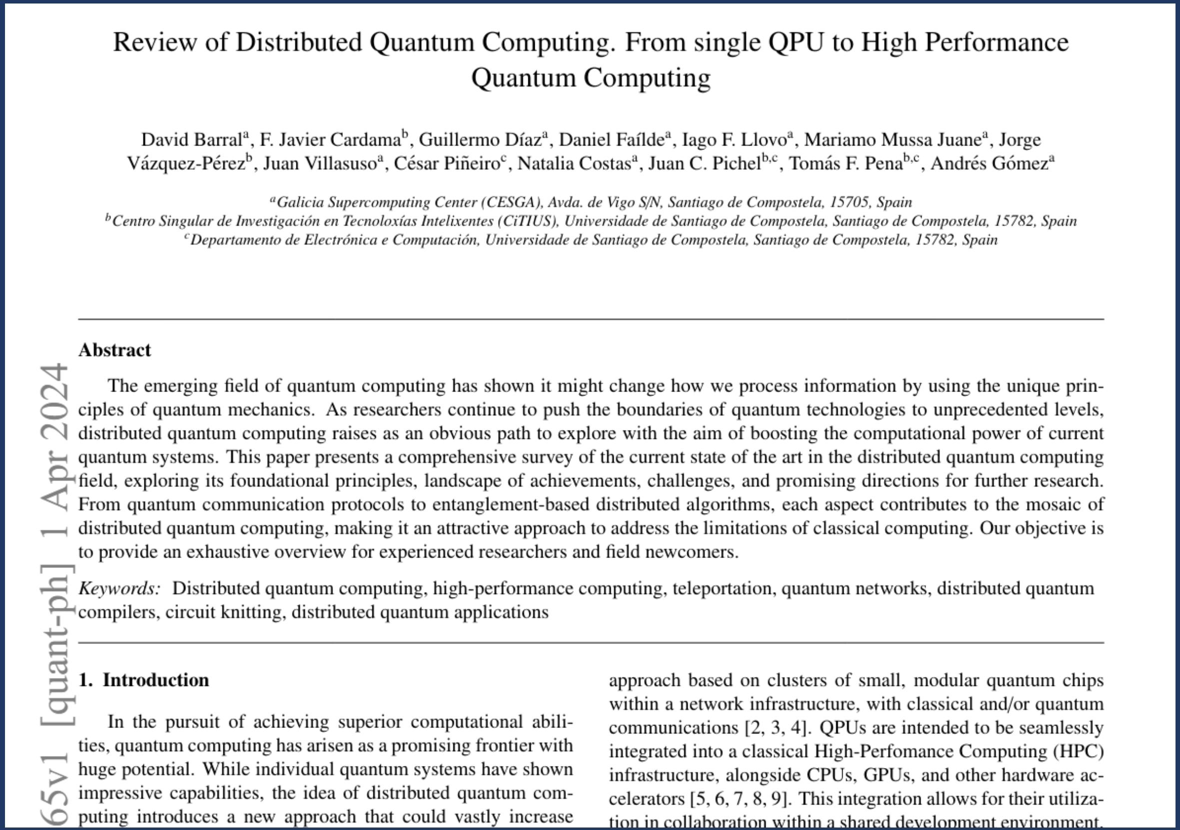
- In short-term, other models of usage can be applied:
  - Time sharing
  - QPU segmentation

5 Where are we?

The present

High Performance Quantum Computers





Review of Distributed Quantum Computing. From single QPU to High Performance Quantum Computing

David Barral[a], F. Javier Cardama[b], Guillermo Díaz[a], Daniel Faílde[a], Iago F. Llovo[a], Mariamo Mussa Juane[a], Jorge Vázquez-Pérez[b], Juan Villasuso[a], César Piñeiro[c], Natalia Costas[a], Juan C. Pichel[b,c], Tomás F. Pena[b,c], Andrés Gómez[a]

[a]*Galicia Supercomputing Center (CESGA), Avda. de Vigo S/N, Santiago de Compostela, 15705, Spain*
[b]*Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, Santiago de Compostela, 15782, Spain*
[c]*Departamento de Electrónica e Computación, Universidade de Santiago de Compostela, Santiago de Compostela, 15782, Spain*

Abstract

The emerging field of quantum computing has shown it might change how we process information by using the unique principles of quantum mechanics. As researchers continue to push the boundaries of quantum technologies to unprecedented levels, distributed quantum computing raises as an obvious path to explore with the aim of boosting the computational power of current quantum systems. This paper presents a comprehensive survey of the current state of the art in the distributed quantum computing field, exploring its foundational principles, landscape of achievements, challenges, and promising directions for further research. From quantum communication protocols to entanglement-based distributed algorithms, each aspect contributes to the mosaic of distributed quantum computing, making it an attractive approach to address the limitations of classical computing. Our objective is to provide an exhaustive overview for experienced researchers and field newcomers.

Keywords: Distributed quantum computing, high-performance computing, teleportation, quantum networks, distributed quantum compilers, circuit knitting, distributed quantum applications

1. Introduction

In the pursuit of achieving superior computational abilities, quantum computing has arisen as a promising frontier with huge potential. While individual quantum systems have shown impressive capabilities, the idea of distributed quantum computing introduces a new approach that could vastly increase approach based on clusters of small, modular quantum chips within a network infrastructure, with classical and/or quantum communications [2, 3, 4]. QPUs are intended to be seamlessly integrated into a classical High-Perfomance Computing (HPC) infrastructure, alongside CPUs, GPUs, and other hardware accelerators [5, 6, 7, 8, 9]. This integration allows for their utilization in collaboration within a shared development environment.

https://arxiv.org/abs/2404.01265

Parallel compilers

NetQIR: An Extension of QIR for Distributed Quantum Computing

Jorge Vázquez-Pérez[1], F. Javier Cardama[1], César Piñeiro[2], Tomás F. Pena[1,2], Juan C. Pichel[1,2], and Andrés Gómez[3]

[1]Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS)
[2]Departamento de Electrónica e Computación, Universidade de Santiago de Compostela
[3]Galicia Supercomputing Center (CESGA)

***Abstract*—The rapid advance of quantum computing has highlighted the need for scalable and efficient software infrastructures to fully exploit its potential. While current quantum processors have significant scalability problems due to the limited number of qubits on each chip, distributed quantum computing offers a promising solution by networking multiple quantum processing units (QPUs). To support this paradigm, robust intermediate representations (IRs) are crucial for translating high-level quantum algorithms into executable instructions across distributed systems. This paper introduces NetQIR, an extension of Microsoft's Quantum Intermediate Representation (QIR), specifically designed for distributed quantum computing. NetQIR is designed to meet the specific needs of distributed quantum systems by incorporating functions to manage quantum and classical communications between QPUs. The main objective is to facilitate the development of new distributed compilers by improving the integration and execution of quantum programmes in a distributed infrastructure, taking advantage of modular architectures to improve scalability. By extending QIR to support**

algorithms on quantum devices [17], [20]. High-level quantum programming languages such as Q# [32], Quipper [16] or Qiskit [2] facilitate the development of quantum algorithms by abstracting the complexities of quantum hardware [30].

For the efficient development of these software tools, quantum code compilers will play a crucial role. A compiler is a software that translates high-level languages into low-level instructions that quantum processors can execute [1]. In classical computing, the concept of IR is introduced as an abstract-machine code to facilitate the development of new compilers [31]. This concept is extended in the world of quantum computing to allow a common IR for target high-level languages and a starting point for low-level instructions. The main objectives of using an IR is to optimise the quantum code and ensure compatibility with various hardware backends [18], [24].

2408.03712 (arxiv.org)

CESGA
GALICIA SUPERCOMPUTING CENTER

Quantum communications @ CESGA
Quantum Software @CESGA

Tomás Fernández

Juan Villasuso

Dani Viqueira

Fran Cardama

J.C. Pichel

César Piñeiro

Jorge Vázquez
Mariamo Mussa
Daniel Faílde
Lois Orosa
Andrés Gómez
David Barral
Guillermo Díaz
Natalia Costas
Iago Fernández
Ignacio López

Centro Singular de Investigación en Tecnoloxías Intelixentes

MICIN through the European Union NextGenerationEU recovery plan (PRTR C 17 I 1 and by the Galician Regional Government through the “Planes Complementarios de I+D+i con las Comunidades Autónomas” in Quantum Communication. Simulations were performed using the Finisterrae III Supercomputer funded by the project CESGA 01 FINISTERRAE III

Qmio>

Do you want to know more? Download Qmio> app JUST NOW:

- *Android*
- *The other*

CESGA
GALICIA SUPERCOMPUTING CENTER

Visit us at:

Source: https://pamgrout.com/2021/06/14/we-need-your-weird-and-wonderful

Thanks!

andres.gomez.tato@cesga.es

Download Qmio> app: