Forum TERATEC 23

Unlock the future

31 MAI & 1er JUIN 2023 • Au Parc Floral, Paris

Un événement organisé par infoprodigital Teratec

EVIDEN

Hybrid Computing in HPC

Cyril ALLOUCHE
VP, Quantum R&D
31/05/2023

an atos business

Content overview

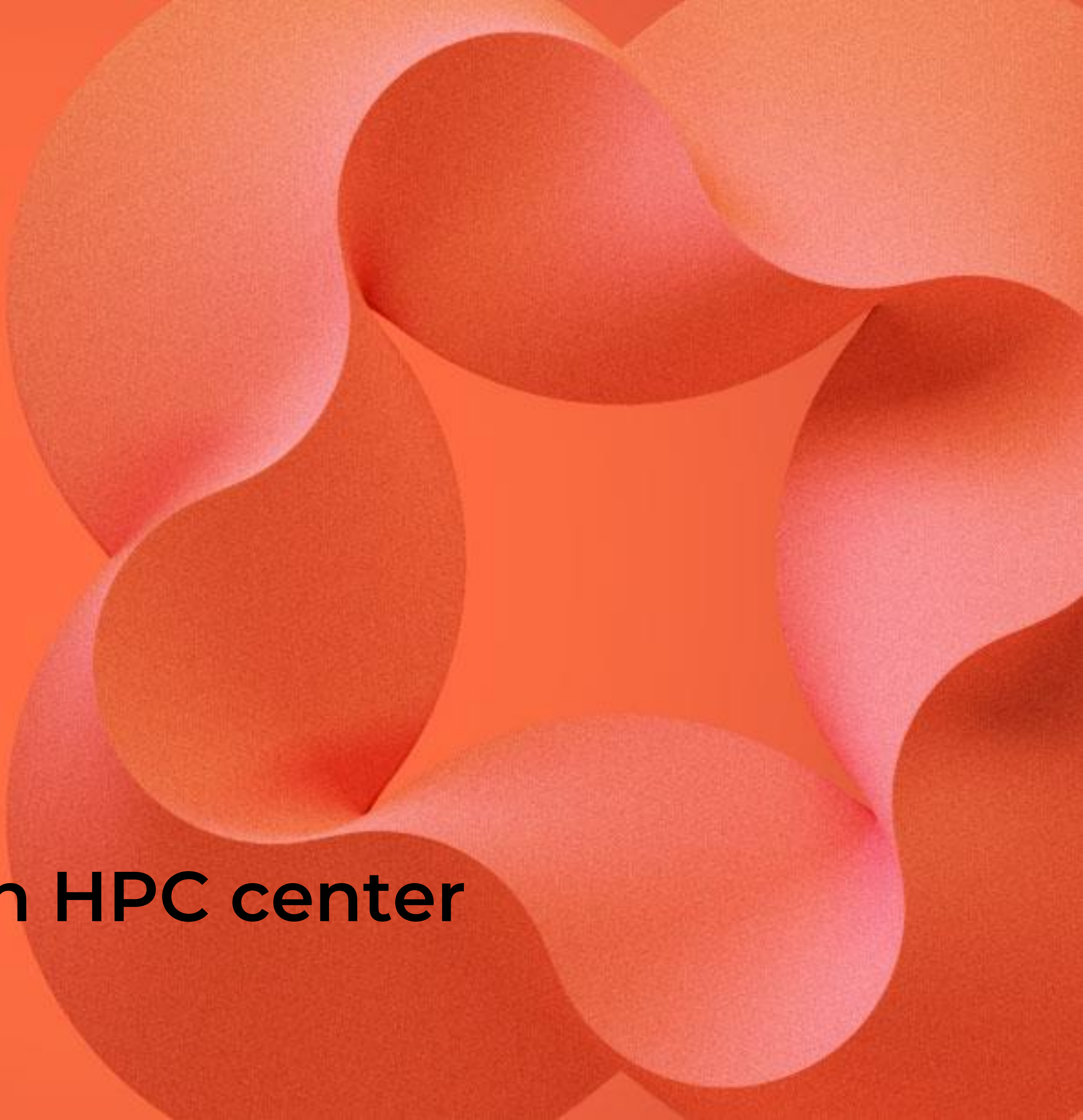
01

NISQ QPUs into an HPC center

02

Towards an High Performance Hybrid Computing (HPHC) Framework

EVIDEN



01 NISQ QPUs into an HPC center

Noisy Intermediate Scale Quantum

Defining NISQ

NISQ *(Noisy Intermediate Scale Quantum)*

→ ~hundreds of noisy qubits

→ ~hundreds instructions

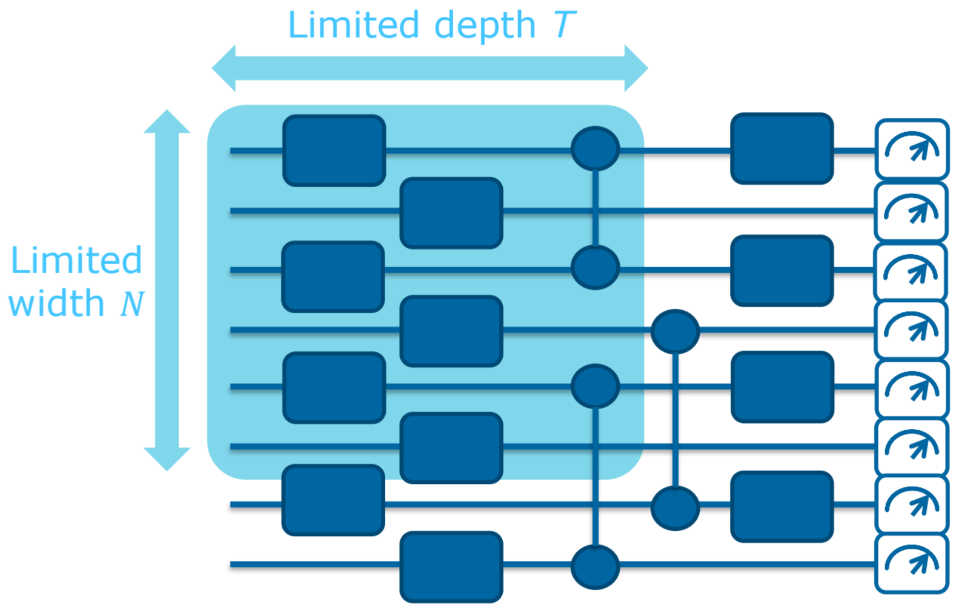
Programming model:

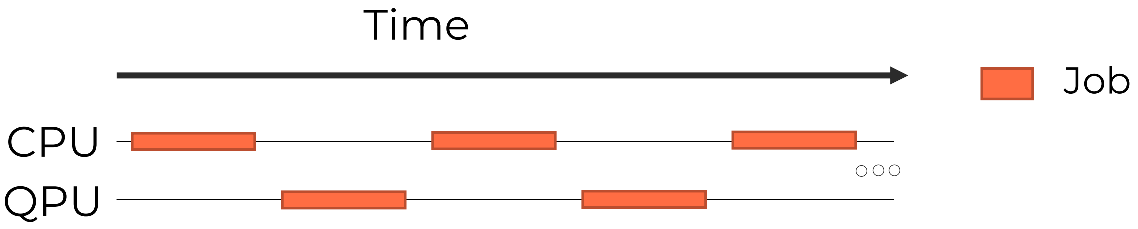
Control flow managed by CPU

Quantum circuits created by CPU

Repeated evaluation of circuit by QPU

⇒ **QPU online slave of CPU**

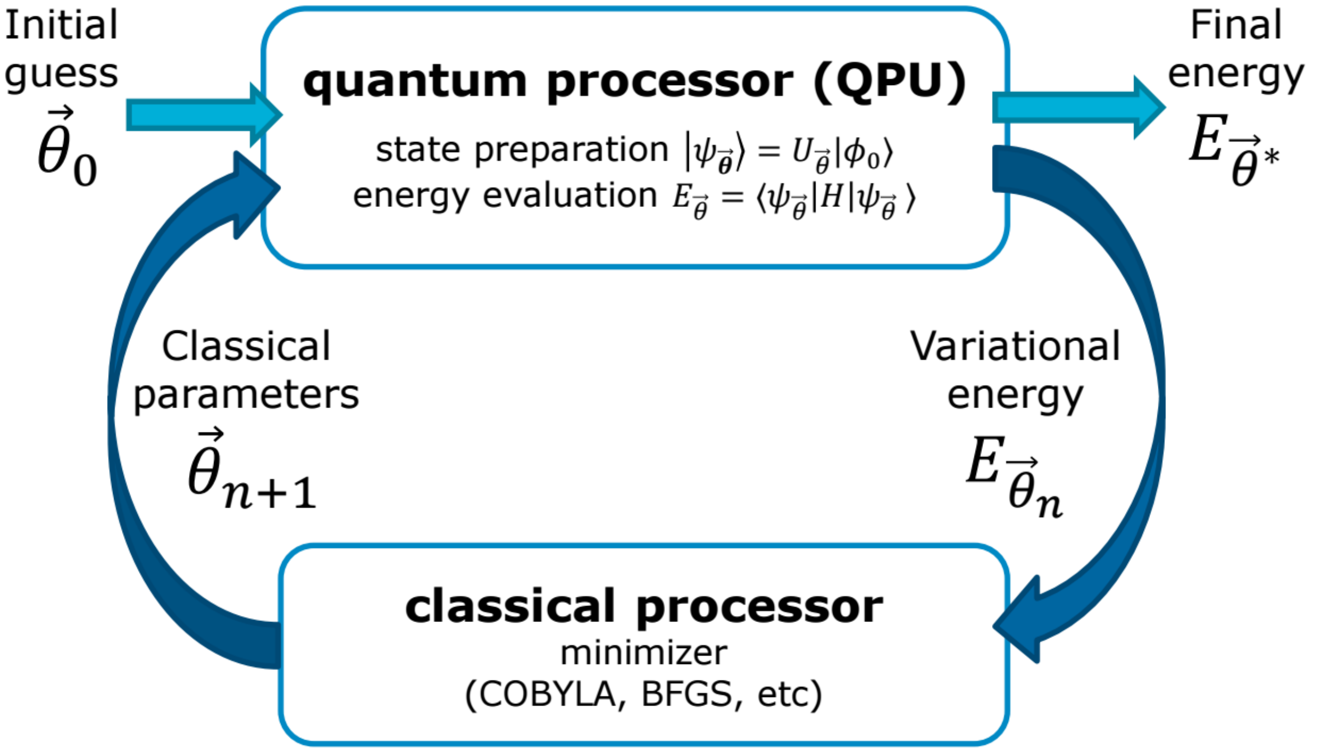




NISQ Algorithm(s)

The variational algorithm

Variational Quantum Eigensolving (VQE) [Peruzzo ‘14]



Initial guess $\vec{\theta}_0$

quantum processor (QPU)

state preparation $|\psi_{\vec{\theta}}\rangle = U_{\vec{\theta}}|\phi_0\rangle$

energy evaluation $E_{\vec{\theta}} = \langle\psi_{\vec{\theta}}|H|\psi_{\vec{\theta}}\rangle$

Final energy $E_{\vec{\theta}^*}$

Classical parameters $\vec{\theta}_{n+1}$

Variational energy $E_{\vec{\theta}_n}$

classical processor

minimizer

(COBYLA, BFGS, etc)

NISQ in a HPC environment

- NISQ programming languages : mostly interpreted, mostly Python

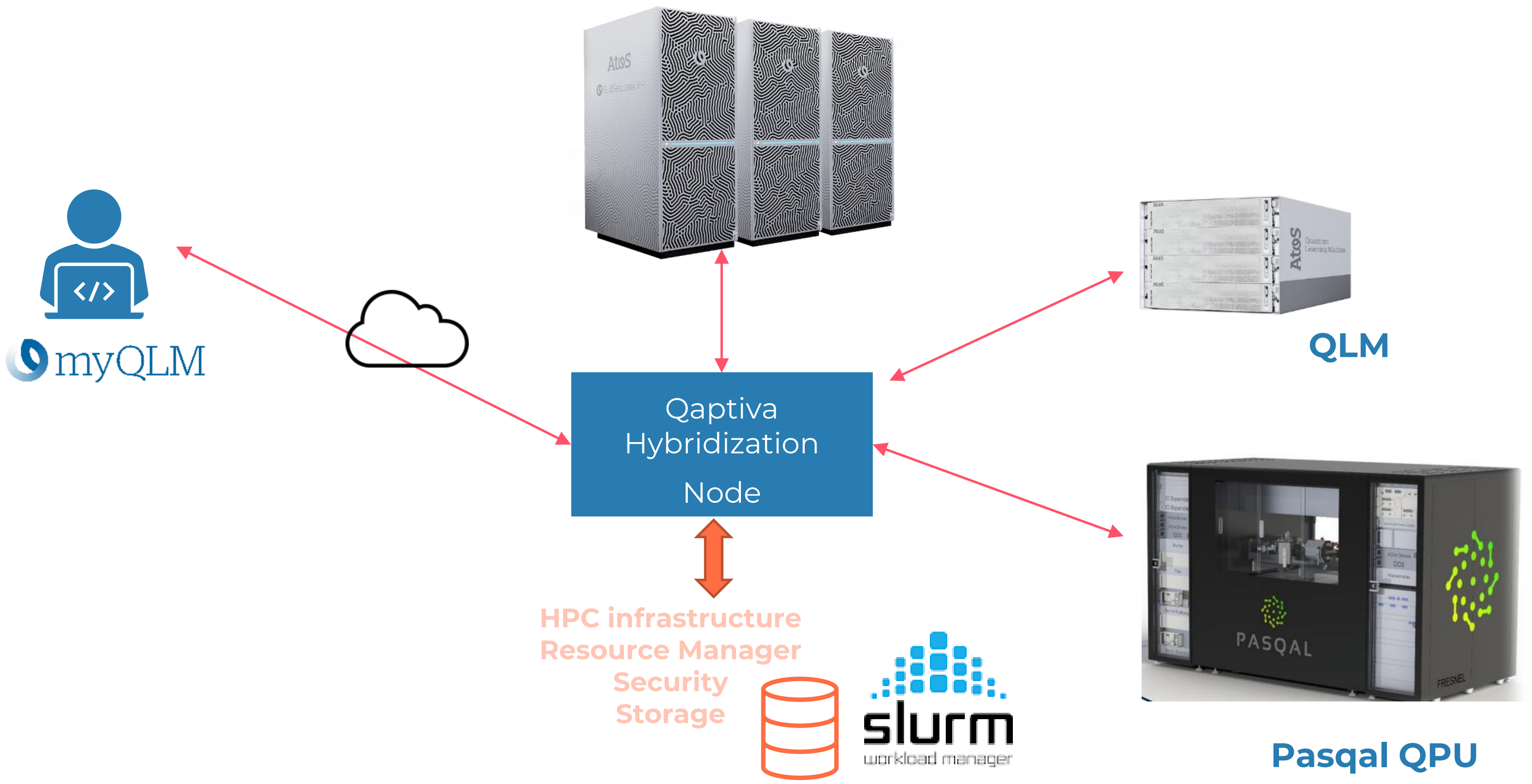
- QPU scalability: ?

- CPU scalability: ?

- Challenge : scheduling QPUs

Integrating NISQ QPUs into an HPC datacenter

- EuroHPC project HPC-QS, France HQI



myQLM

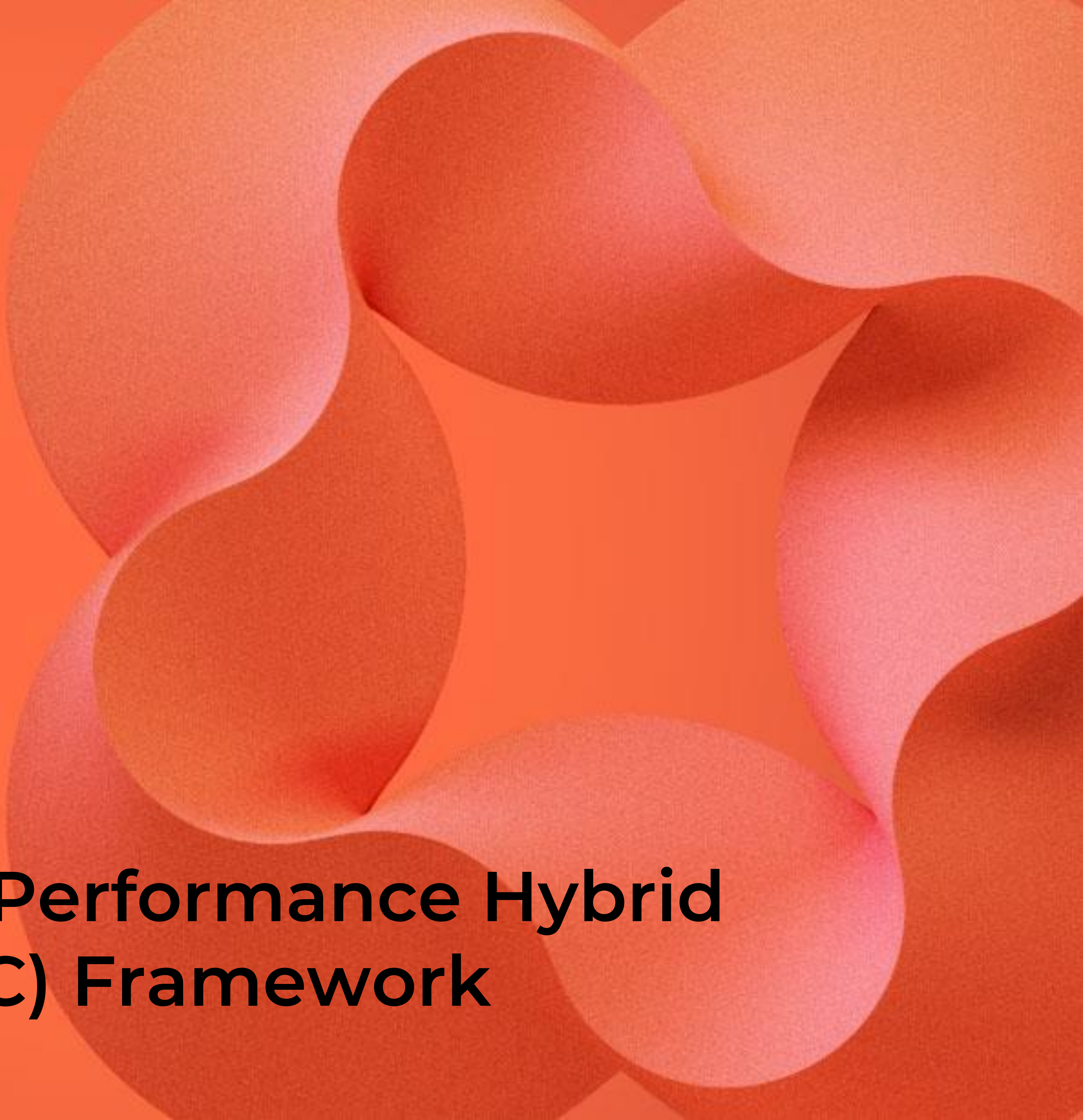
Qaptiva Hybridization

Node

QLM

Pasqal QPU

HPC infrastructure
Resource Manager
Security
Storage



02 Towards an High Performance Hybrid Computing (HPHC) Framework

High Performance Hybrid Computing

Defining HPHC

HPHC *(High Performance Hybrid Computing)*

→ ~thousands of perfect logical qubits (with QEC)

→ Multi-QPUs

→ Use of QPUs in HPC centers

Long term

Entire application, composed of classical and quantum parts

HPC programming languages (compatibility with C, C++, Fortran, etc.)

What will an HPHC program look like?

› Architecture of HPHC quantum devices ?

QPUs will have classical capabilities

Architecture of an hybrid quantum device

QPUs will be composed of:

- A *controller* receiving instructions and scheduling them on the *quantum part*
- A *quantum part* being the core of the QPU

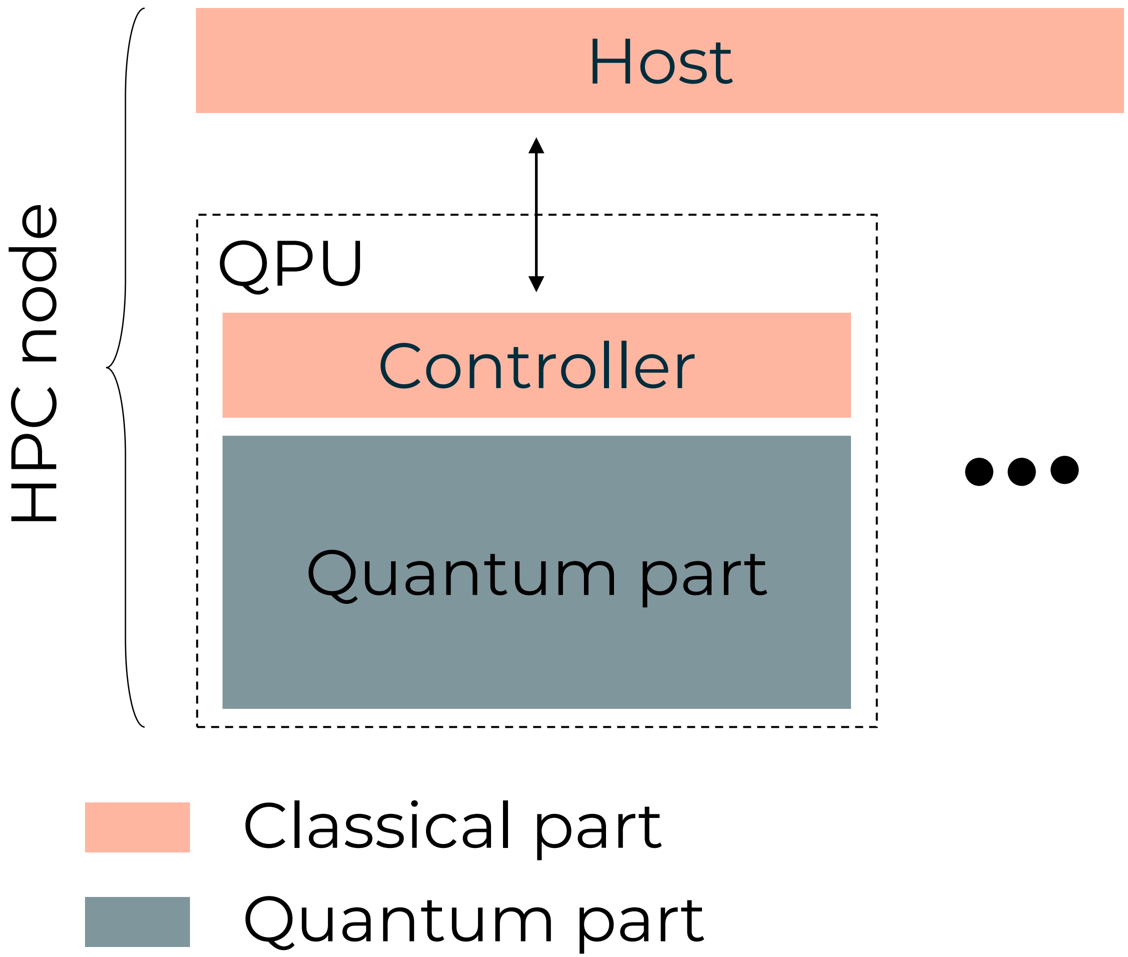
Controller purpose:

Manage quantum part

Quantum error correction

● ● ●

Execute classical user code?



Quantum capabilities

Defining quantum specific operations

Quantum routines are reversible

Quantum routines controllable

Access to quantum memory

Safe uncomputation should be used to reset a register

Typing quantum memory

Q-Pragma – A C++ Framework for LSQ computing

A framework composed of a library and some pragmas

Q-Pragma C++ framework:

Pragmas to extend C++ language, to add:

- Hybridization capabilities
- Quantum capabilites

A library providing:

- Quantum types
- Quantum routines
- …

Q-Pragma example

```
#pragma quantum routine
void bell_pair(const qbool & qb0,
               const qbool & qb1) {
    H(qb0);
    CNOT(qb0, qb1);
}

int main() {
    …;
    ::bell_pair(qb1, qb2);
    ::bell_pair.dag(qb1, qb2);
    ::bell_pair.ctrl(qc, qb1, qb2);
}
```

Perspectives for Q-Pragma

- Open source specification
- Federate a community from HPC
- Continue co-design , guided by HPC use cases

EVIDEN

Questions

Thank you!

For more information, please contact:
Cyril ALLOUCHE