LibKet: A Software Framework for Quantum-Accelerated Scientific Computing

2021 SIAM Conference on Computational Science and Engineering

March 1st, 2021

Matthias Möller[1] and Carmen G. Almudever[2]

[1]Delft University of Technology (m.moller@tudeflt.nl)

[2]Technical University of Valencia (cargara2@disca.upv.es)

Who we are
Matthias Möller

Associate Professor of Numerical Analysis,
Department of Applied Mathematics, TU Delft

- Numerical methods for PDE problems (FEM, IGA, MPM, …)
- High-performance/accelerated computing
- Quantum-accelerated scientific computing

Nauman Ahmed

Giorgio Balducci

Merel Schalkers

Smaran Adarsh

Who we are
Carmen G. Almudever

Distinguished Researcher – Quantum Computing Architectures

Computer Engineering Department, Technical University of Valencia (UPV)

- Quantum software, mapping of quantum algorithms
- QEC and FT quantum computation
- Architecting and benchmarking of quantum computers
- Scalability of quantum hardware

Abid Moudenne

Medina Bandic

Nikiforos Paraskevoupolus

Matt Steinberg

Funding and support

4TU. CENTRE FOR ENGINEERING EDUCATION

TNO innovation for life

STILLWATER SUPERCOMPUTING, INC.

IBM Q™

rigetti QCS

OLCF
OAK RIDGE LEADERSHIP COMPUTING FACILITY

binder

https://tinyurl.com/thxvwrnk

siam 2021 | Conference on Computational Science and Engineering

Mini-tutorial at SIAM CSE21, March 1-5, 2021

LibKet: A Software Framework for Quantum-Accelerated Scientific Computing

Organizers: Carmen G. Almudever, Matthias Möller

Session 1 (MT5): Monday, March 1, 9:45 AM – 11:25 AM CST

Time	Content	Lecturer	Slides	Binder
9:45-09:55	Quantum-accelerated scientific computing	Matthias	slides	
09:55-11:25	Hands-on introduction to quantum computing with LibKet	Carmen/Matthias	slides	tutorial 01

Session 2 (MT6): Monday, March 1, 2:15 PM – 3:55 PM CST

Time	Content	Lecturer	Slides	Binder
2:15-3:00	Quantum computing today and future perspective	Carmen	slides	
3:00-3:55	Hands-on scientific computing with LibKet	Matthias		tutorial 02 tutorial

Click now since first launch can take some time

ILib

binder

https://tinyurl.com/thxvwrnk

binder

Starting repository: mmoelle1/LibKet/master

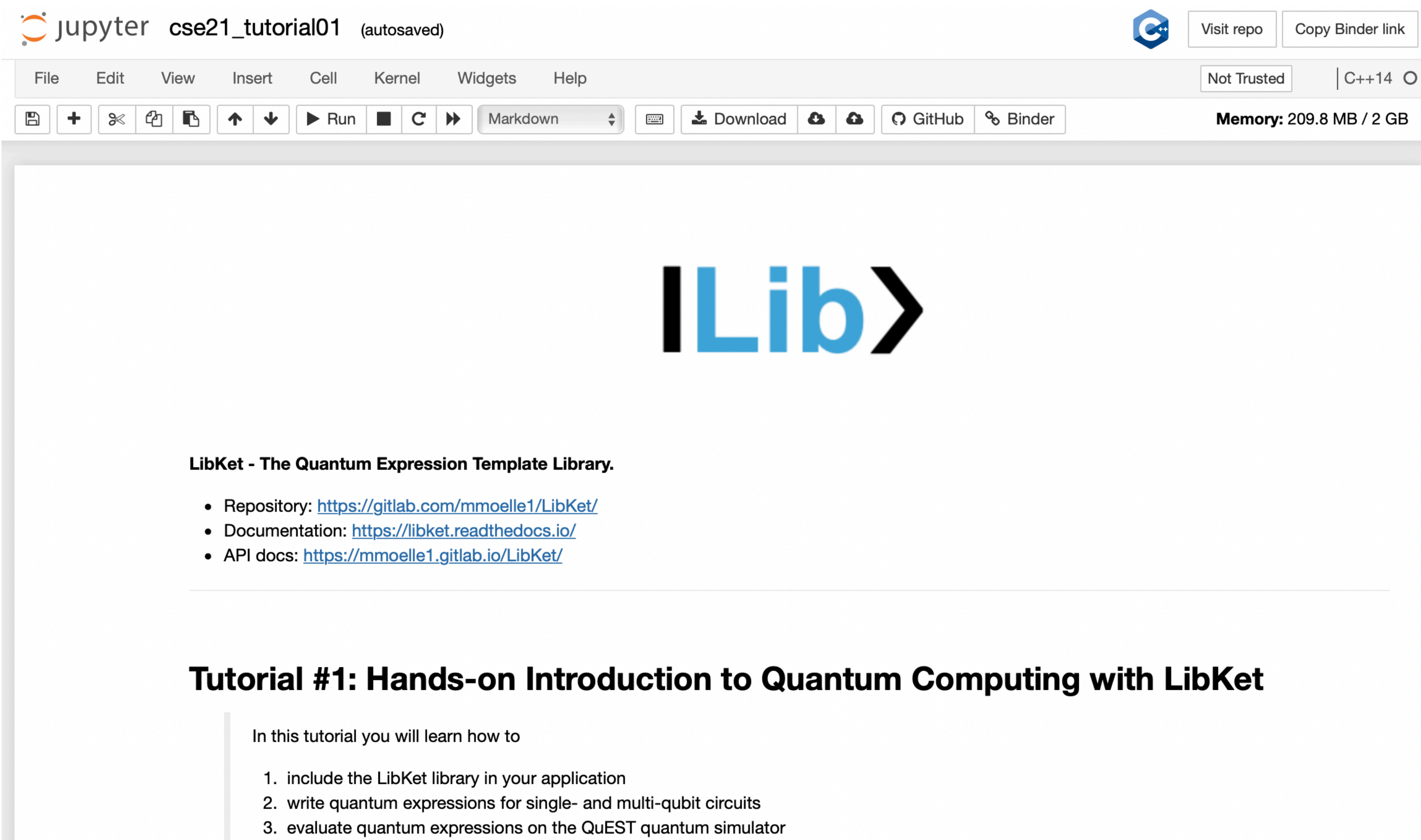
Your session is taking longer than usual to start!
Check the log messages below to see what is happening.

Build logs hide

```
 ---> 29bf3cb5b44f
Step 48/51 : RUN ./postBuild
 ---> Running in 9345aed0e8bf
Cloning into '/home/jovyan/LibKet'...
warning: redirecting to https://gitlab.com/mmoelle1/LibKet.git/
Submodule 'external/OpenQL' (https://github.com/QE-Lab/OpenQL.git) registered for path 'external/OpenQL'
Submodule 'external/QuEST' (https://github.com/QuEST-Kit/QuEST.git) registered for path 'external/QuEST'
Submodule 'external/armadillo' (https://gitlab.com/conradsnicta/armadillo-code.git) registered for path 'external/armadillo'
Submodule 'external/optim' (https://github.com/kthohr/optim.git) registered for path 'external/optim'
Submodule 'external/pegtl' (https://github.com/taocpp/PEGTL.git) registered for path 'external/pegtl'
```

LLib⟩

https://tinyurl.com/thxvwrnk



Quantum-Accelerated Scientific Computing

2021 SIAM Conference on Computational Science and Engineering
March 1st, 2021

Matthias Möller[1] and Carmen G. Almudever[2]
[1]Delft University of Technology (m.moller@tudeflt.nl)
[2]Technical University of Valencia (cargara2@disca.upv.es)

LLib⟩

Quantum Computing … why?

Quantum computers can solve problems that are intractable for even the most powerful classical supercomputers (e.g. simulation, search and optimization)

New materials (aerospace, automotive, energy)

New drugs (chemistry and pharma)

Better products and services (logistics, healthcare, finance)

Quantum Computing … first applications

PRL **103,** 150502 (2009) PHYSICAL REVIEW LETTERS week ending 9 OCTOBER 2009

Quantum Algorithm for Linear Systems of Equations

Aram W. Harrow,[1] Avinatan Hassidim,[2] and Seth Lloyd[3]

[1]*Department of Mathematics, University of Bristol, Bristol, BS8 1TW, United Kingdom*
[2]*Research Laboratory for Electronics, MIT, Cambridge, Massachusetts 02139, USA*
[3]*Research Laboratory for Electronics and Department of Mechanical Engineering, MIT, Cambridge, Massachusetts 02139, USA*

(Received 5 July 2009; published 7 October 2009)

Solving linear systems of equations is a common problem that arises both on its own and as a subroutine in more complex problems: given a matrix A and a vector $\vec{b}$, find a vector $\vec{x}$ such that $A\vec{x} = \vec{b}$. We consider the case where one does not need to know the solution $\vec{x}$ itself, but rather an approximation of the expectation value of some operator associated with $\vec{x}$, e.g., $\vec{x}^\dagger M \vec{x}$ for some matrix M. In this case, when A is sparse, $N \times N$ and has condition number κ, the fastest known classical algorithms can find $\vec{x}$ and estimate $\vec{x}^\dagger M \vec{x}$ in time scaling roughly as $N\sqrt{\kappa}$. Here, we exhibit a quantum algorithm for estimating $\vec{x}^\dagger M \vec{x}$ whose runtime is a polynomial of $\log(N)$ and κ. Indeed, for small values of κ [i.e., poly $\log(N)$], we prove (using some common complexity-theoretic assumptions) that any classical algorithm for this problem generically requires exponentially more time than our quantum algorithm.

DOI: 10.1103/PhysRevLett.103.150502 PACS numbers: 03.67.Ac, 02.10.Ud, 89.70.Eg

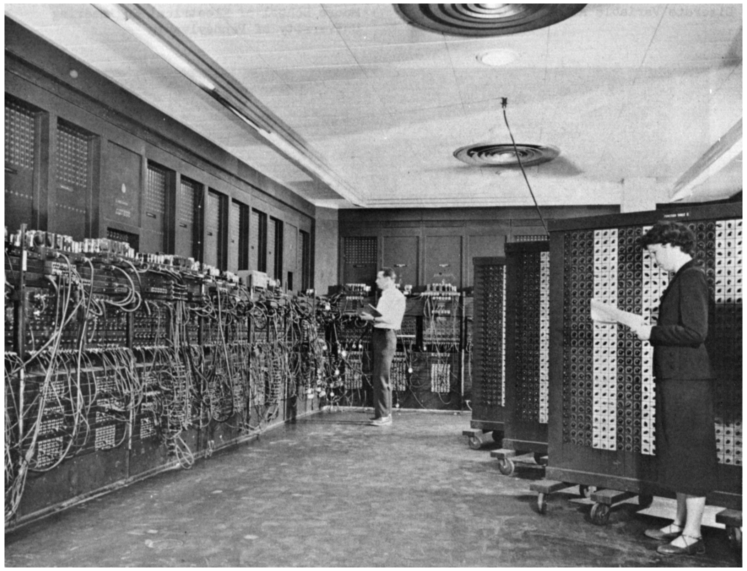
Variational Quantum Linear Solver

Carlos Bravo-Prieto,[1,2,3] Ryan LaRose,[4] M. Cerezo,[1,5] Yiğit Subaşı,[6] Lukasz Cincio,[1] and Patrick J. Coles[1]

[1]*Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA.*
[2]*Barcelona Supercomputing Center, Barcelona, Spain.*
[3]*Institut de Ciències del Cosmos, Universitat de Barcelona, Barcelona, Spain.*
[4]*Department of Computational Mathematics, Science, and Engineering & Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48823, USA.*
[5]*Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM, USA*
[6]*Computer, Computational and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

Previously proposed quantum algorithms for solving linear systems of equations cannot be implemented in the near term due to the required circuit depth. Here, we propose a hybrid quantum-classical algorithm, called Variational Quantum Linear Solver (VQLS), for solving linear systems on near-term quantum computers. VQLS seeks to variationally prepare $|x\rangle$ such that $A|x\rangle \propto |b\rangle$. We derive an operationally meaningful termination condition for VQLS that allows one to guarantee that a desired solution precision ϵ is achieved. Specifically, we prove that $C \geqslant \epsilon^2/\kappa^2$, where C is the VQLS cost function and κ is the condition number of A. We present efficient quantum circuits to estimate C, while providing evidence for the classical hardness of its estimation. Using Rigetti's quantum computer, we successfully implement VQLS up to a problem size of 1024×1024. Finally, we numerically solve non-trivial problems of size up to $2^{50} \times 2^{50}$. For the specific examples that we consider, we heuristically find that the time complexity of VQLS scales efficiently in ϵ, κ, and the system size N.

Quantum Computing … where are we?



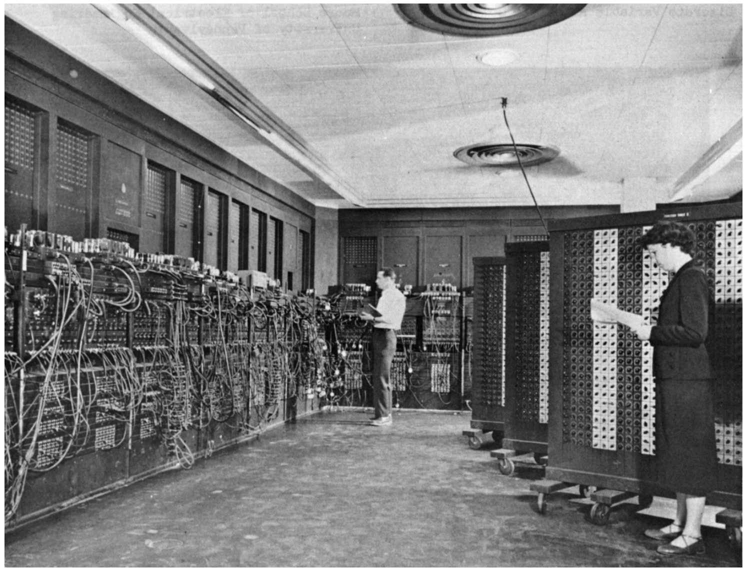
ENIAC 1947

Fugaku, #1 in TOP-500 Nov 2020

IBM Q 65-qubits, 2020

IBM Q 1000-qubits, 2023

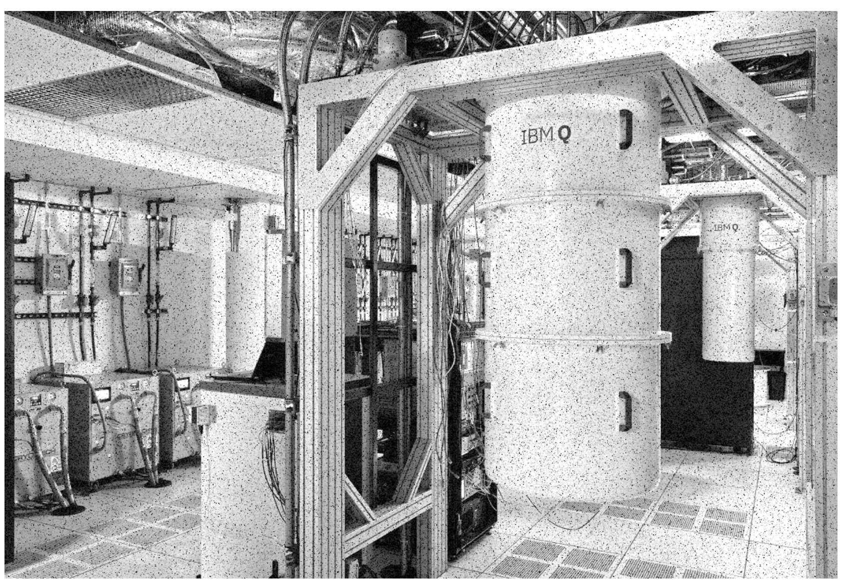
Quantum Computing … in its infancy!



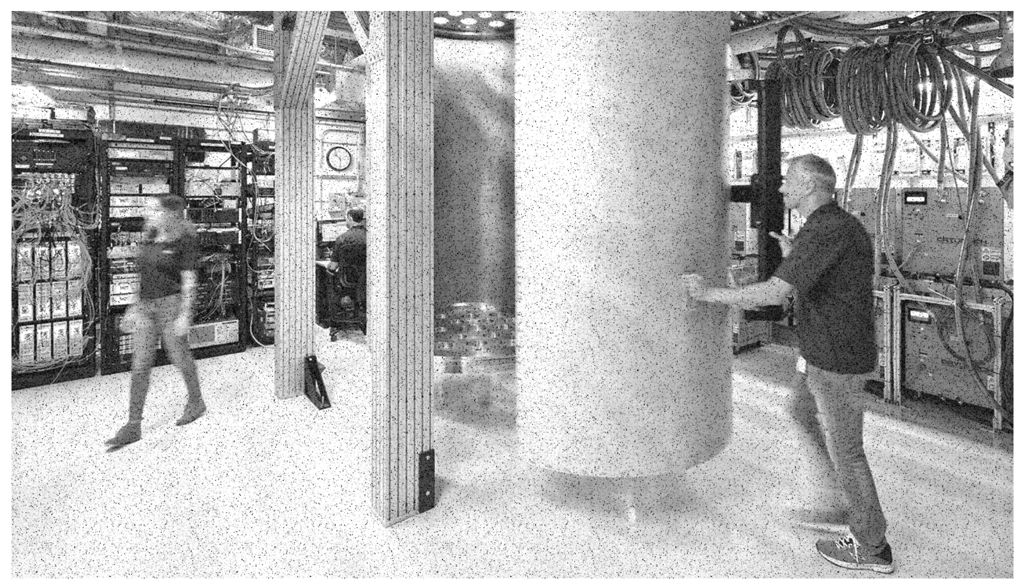
ENIAC 1947

Fugaku, #1 in TOP-500 Nov 2020

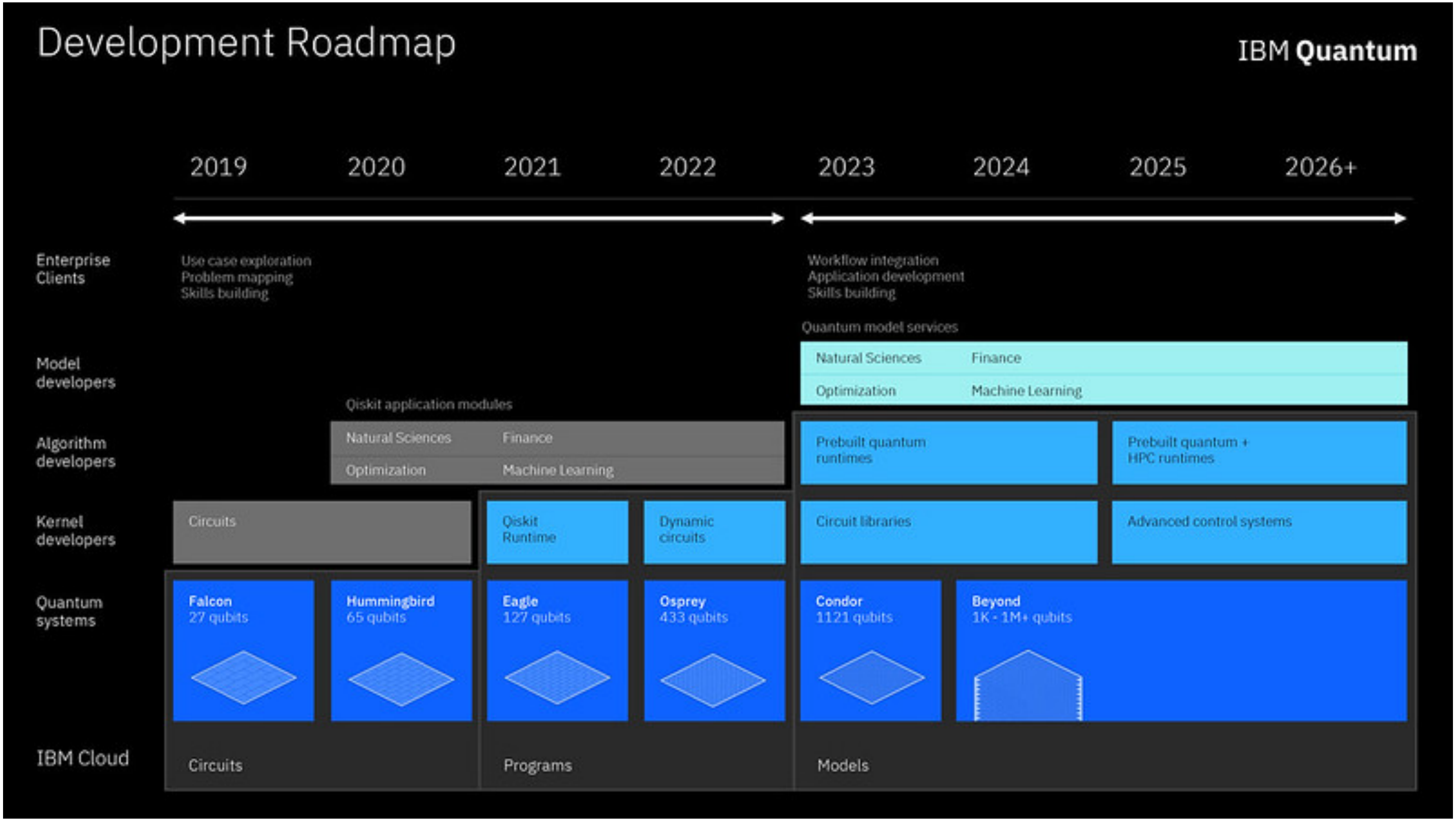
IBM Q 65-qubits, 2020



IBM Q 1000-qubits, 2023

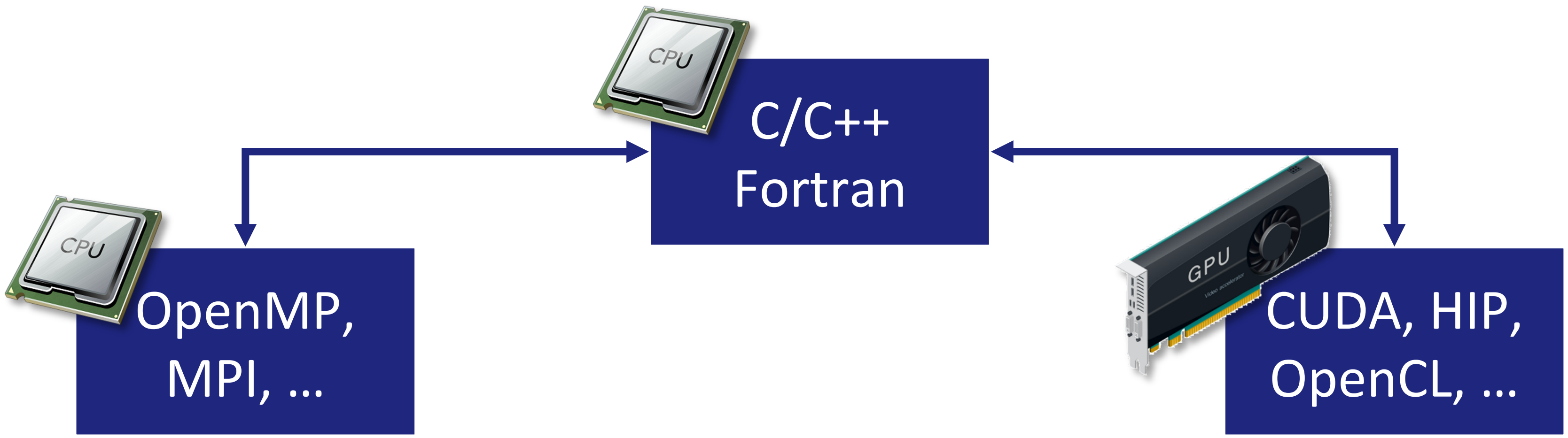


Quantum Computing … industry interest!



ILib

Scientific Computing



Quantum Computing

OpenQL

Forest

cQASM

Pennylane

OpenQASM

AQASM

QX

QuEST

pyQuil

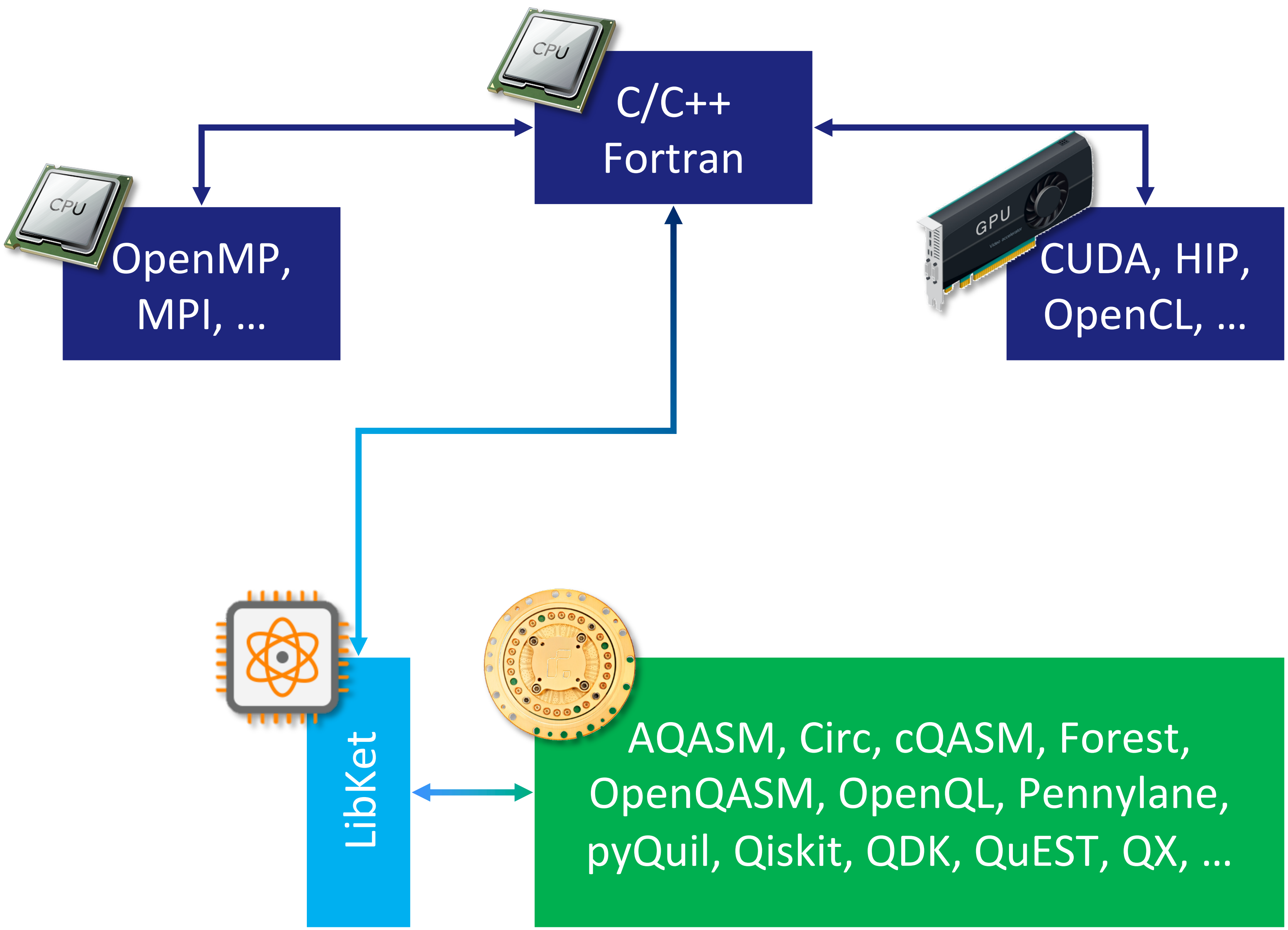
Cirq

QDM

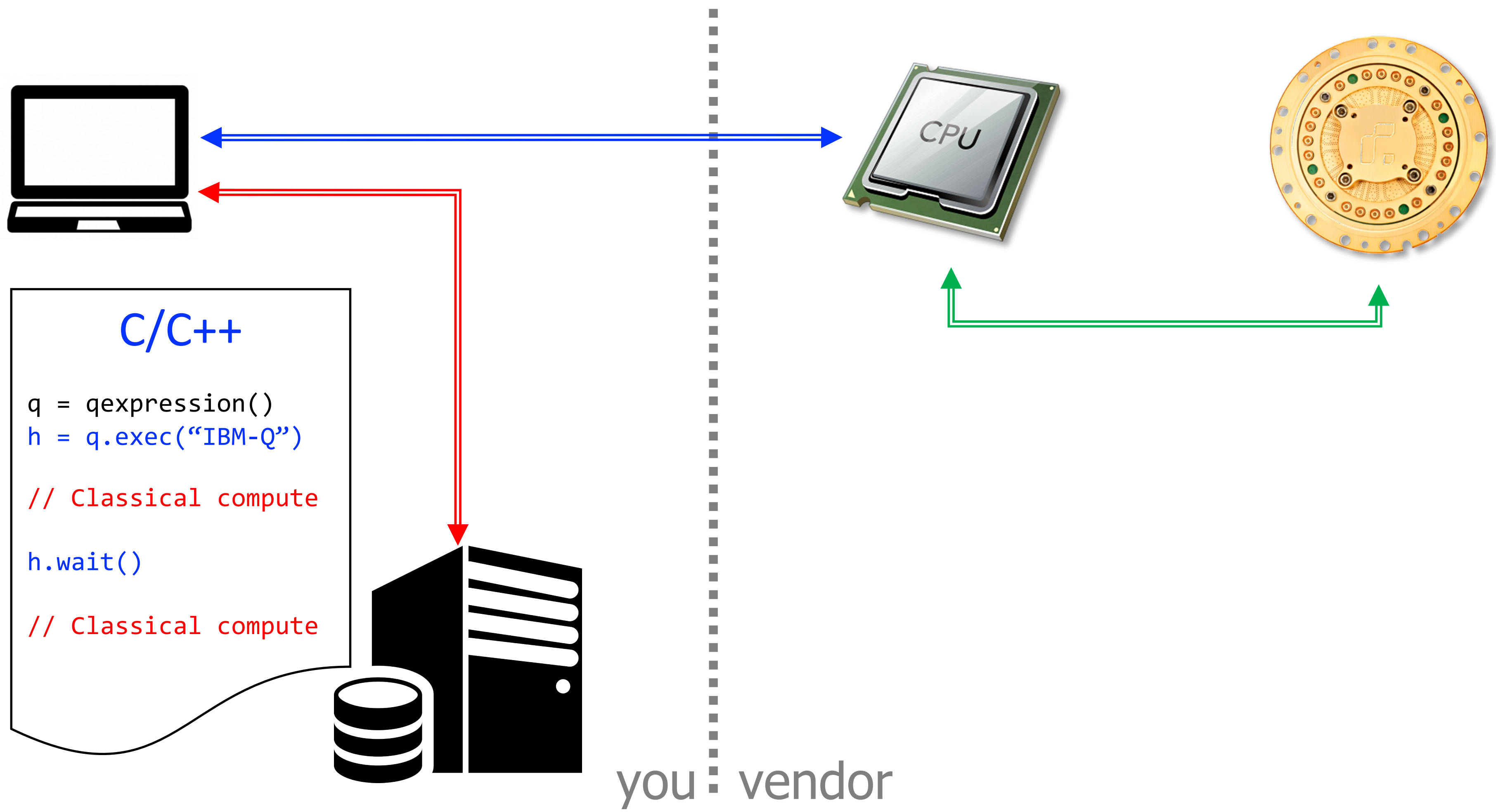
Qiskit

ILib

Quantum-*Accelerated* Computing

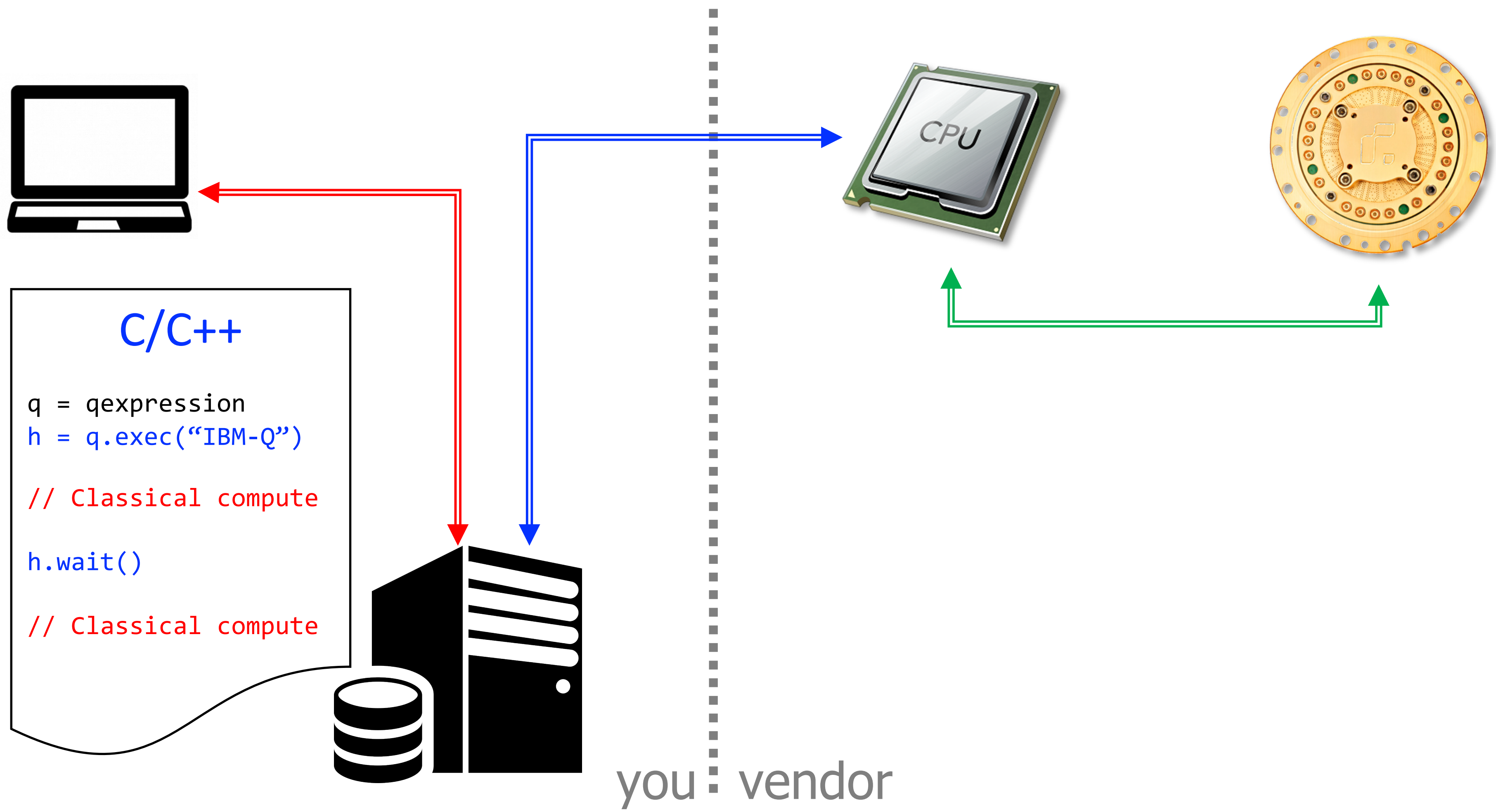


Workflow



C/C++
q = qexpression()
h = q.exec("IBM-Q")
// Classical compute
h.wait()
// Classical compute
CPU
you
vendor

Workflow



|Lib⟩