Conférence Quantum Computing et IBM Q
Institut des Actuaires

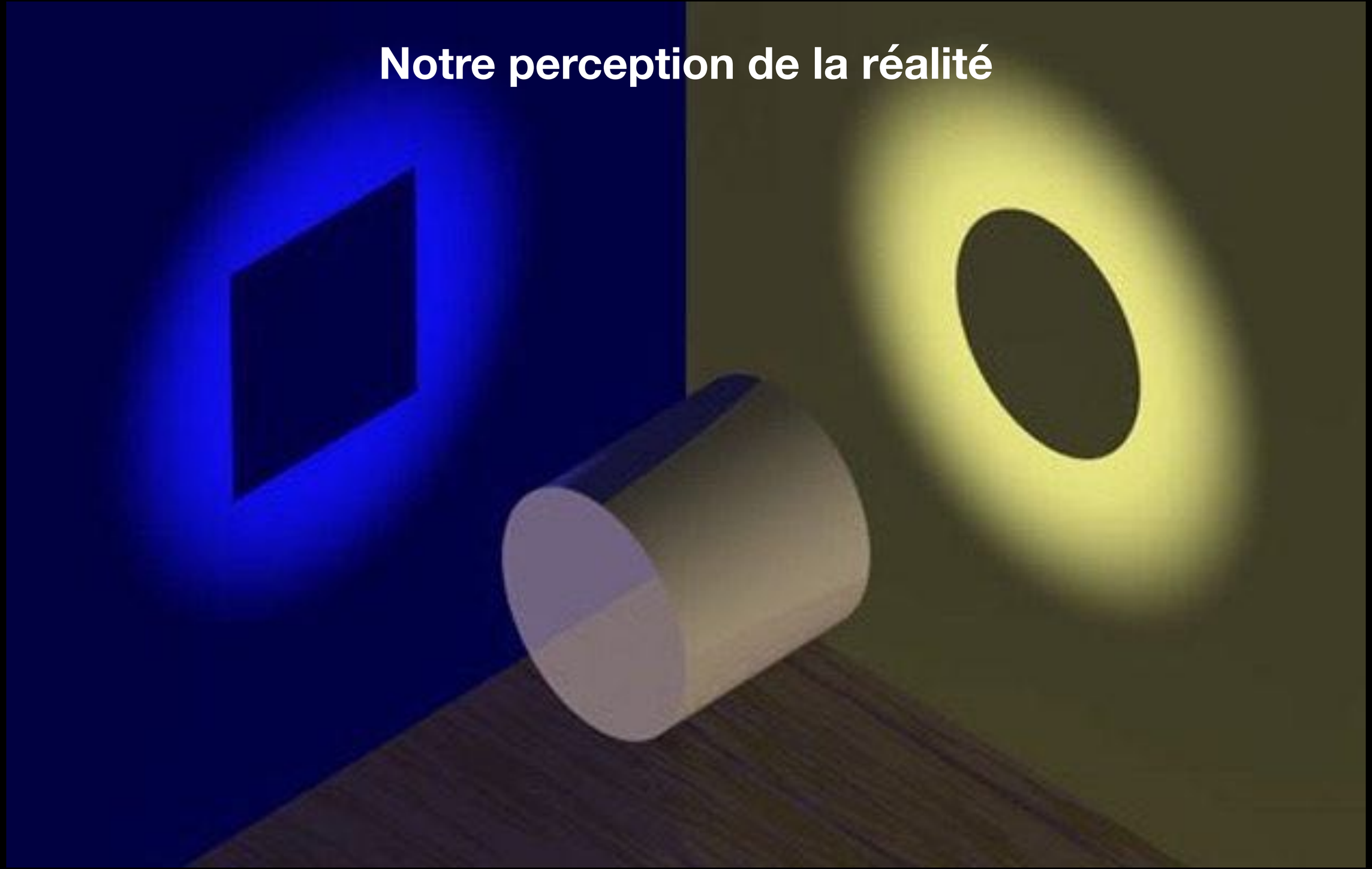
19 Juin 2018

Georges Uzbelger - IBM France

Présentation

- Georges Uzbelger
  - Advanced Analytics & Quantum Computing Leader
  - Development of Academic / Research Relations
    (Ecole Polytechnique, Sorbonne Université, Université Paris Dauphine, IHP)
- georges.uzbelger@fr.ibm.com
- LinkedIn

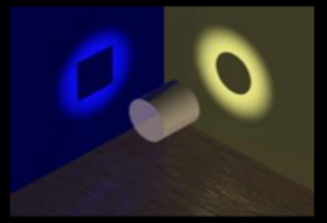
Notre perception de la réalité



Onde ou particule ?

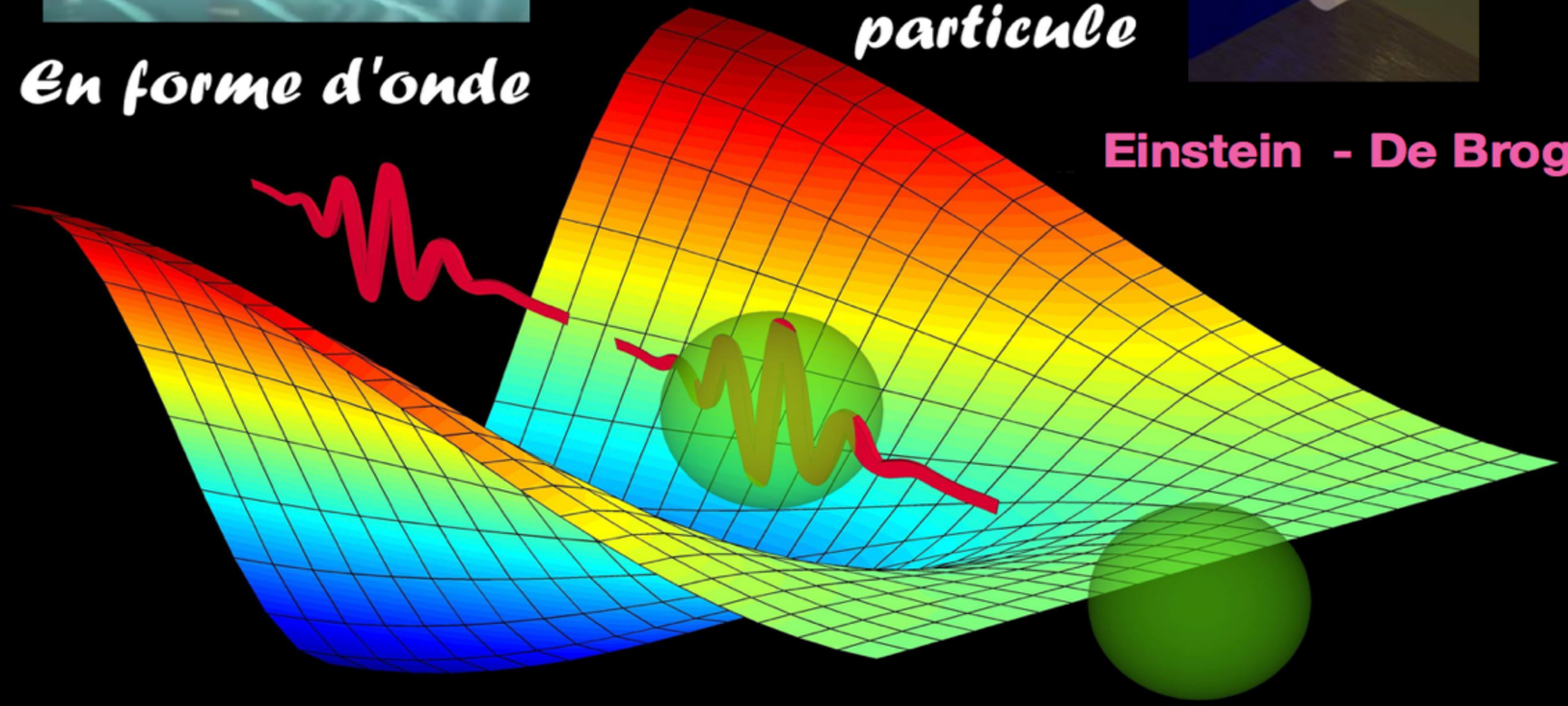
En forme d'onde

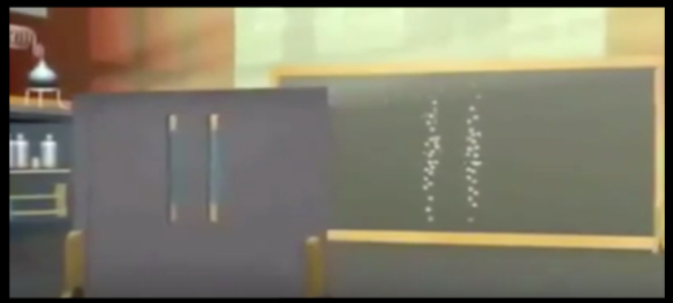
Onde et particule



$$E = h\nu$$
$$p = \frac{h}{\lambda}$$

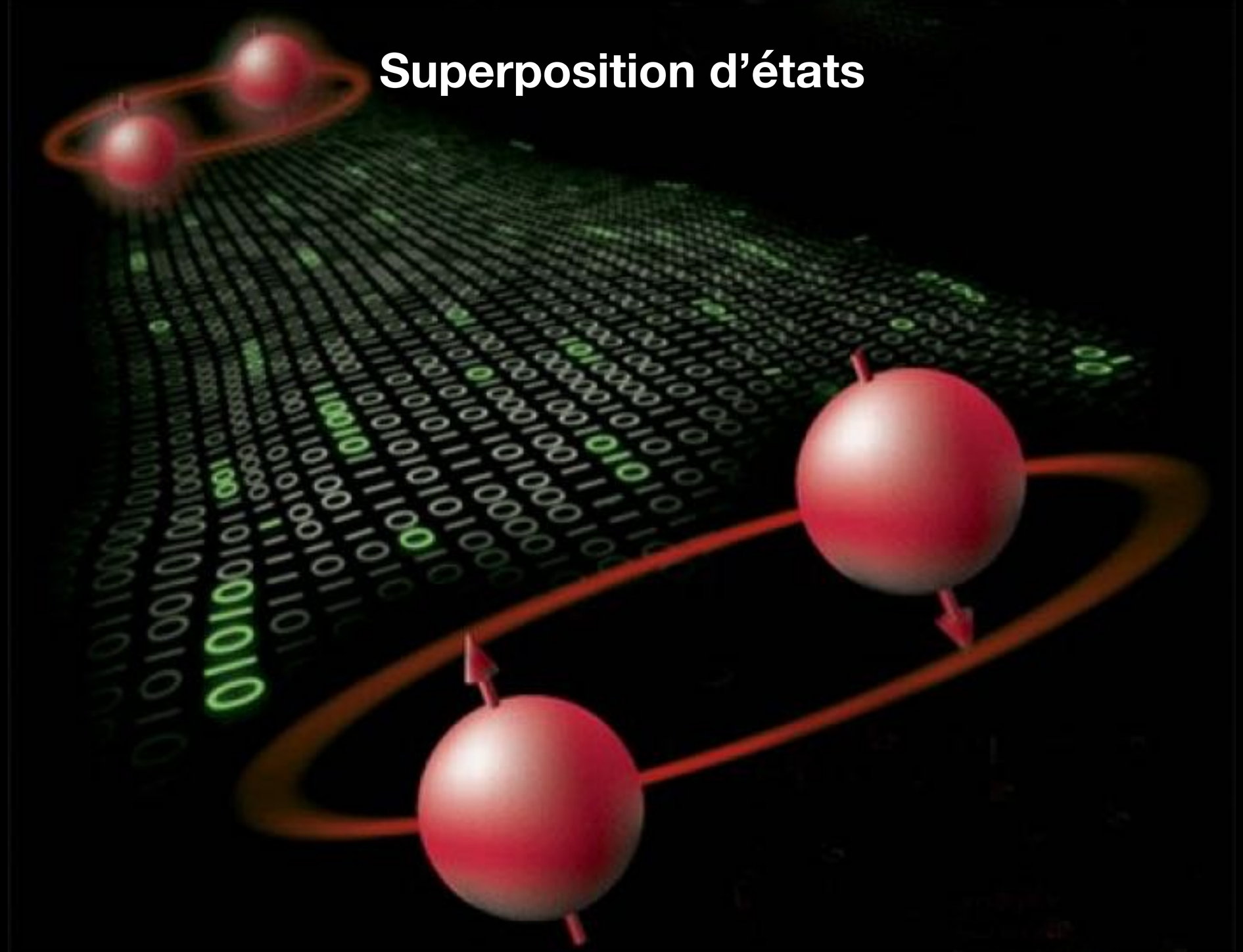
Einstein - De Broglie





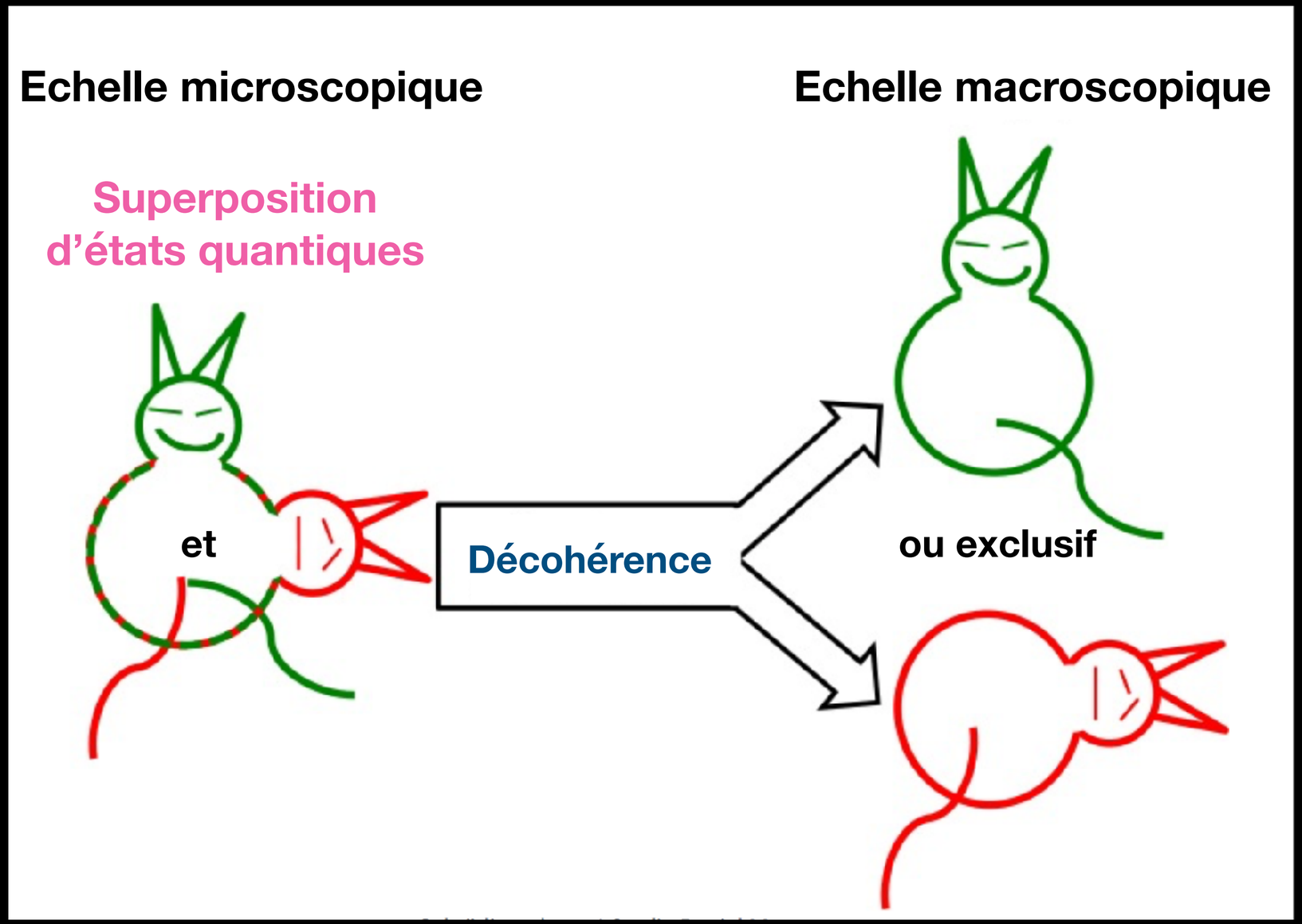
En forme de particule

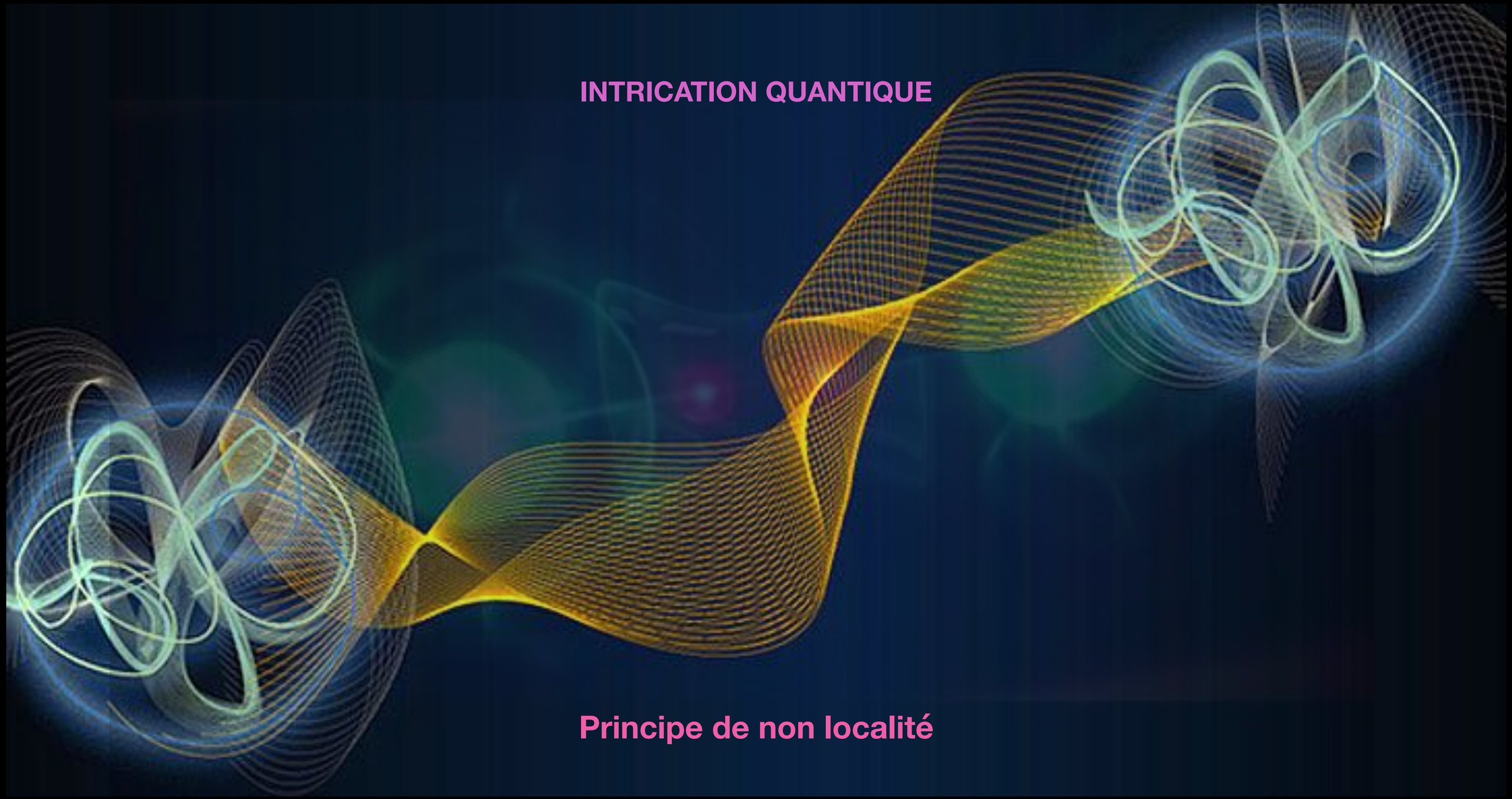
Superposition d’états




Notre chat de Schrödinger
est-il
mort ou vivant ?

Décohérence



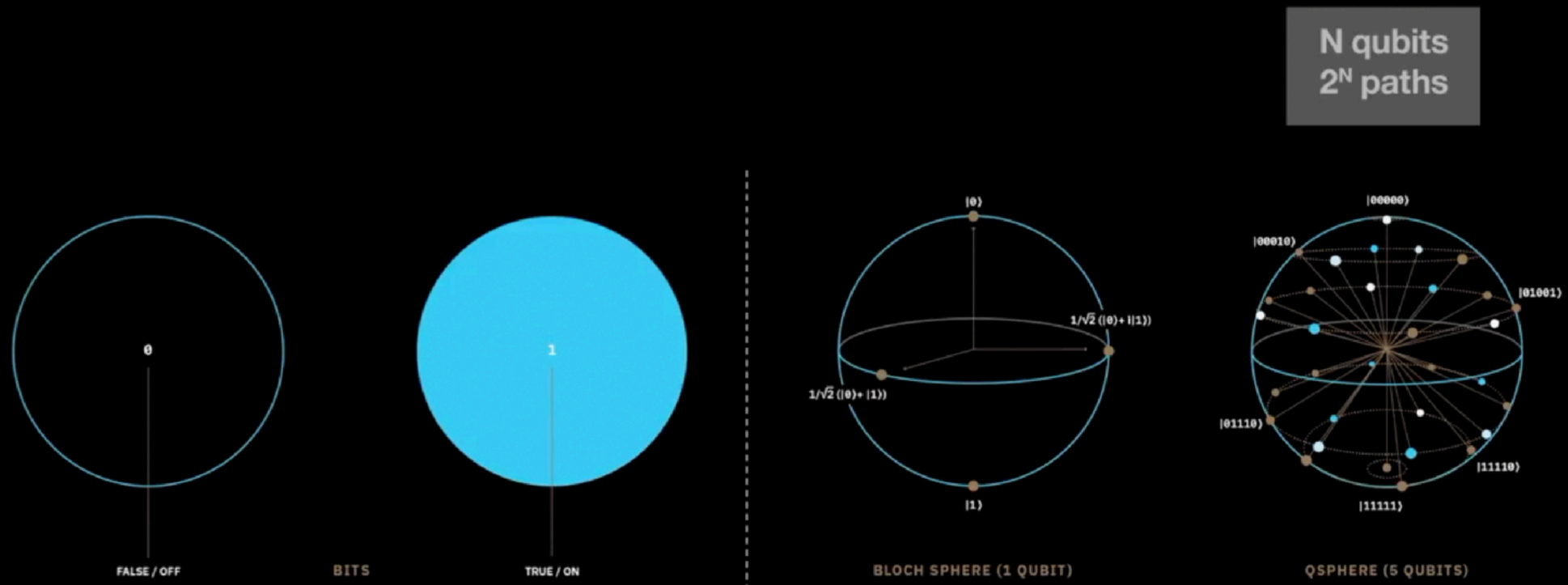

INTRICATION QUANTIQUE
Principe de non localité

History of Quantum Computing

Year	Event
1930er	Discovery of Quantum Mechanics
1970	Note on “quantum information theory” from Charlie Bennett (today IBM Fellow) and Stephen Wiesner
1981	Richard Feynman suggests to build a quantum computer
1984	Quantum key distribution Bennett/Brassard
1993	Quantum Teleportation Bennett
1994	Shor’s Factoring Algorithm
1995	Quantum Error Correction
1996	DiVincenzo Criterias to build an Quantencomputer
2001	Experimentally factoring the number 15
2004	Circuit QED
2007	The Transmon Superconducting Qubit is invented
2012	Qubit coherence time improvements
2014	Shor Algorithmus with NMR Steffen/Chuang/Vandersypen
2015	Error correction with superconductive Qubits
2016	IBM Quantum Experience & 5 Qubits
2017	The First Universal Quantum Computers for Business and Science with 20 Qubits

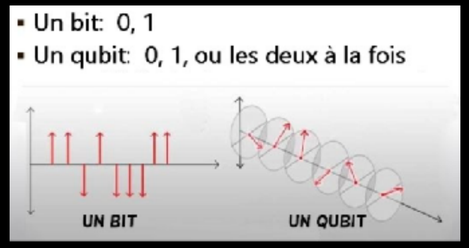
10 influential figures in the history of quantum mechanics. Left to right: Max Planck, Albert Einstein, Niels Bohr, Louis de Broglie, Max Born, Paul Dirac, Werner Heisenberg, Wolfgang Pauli, Erwin Schrödinger, Richard Feynman.

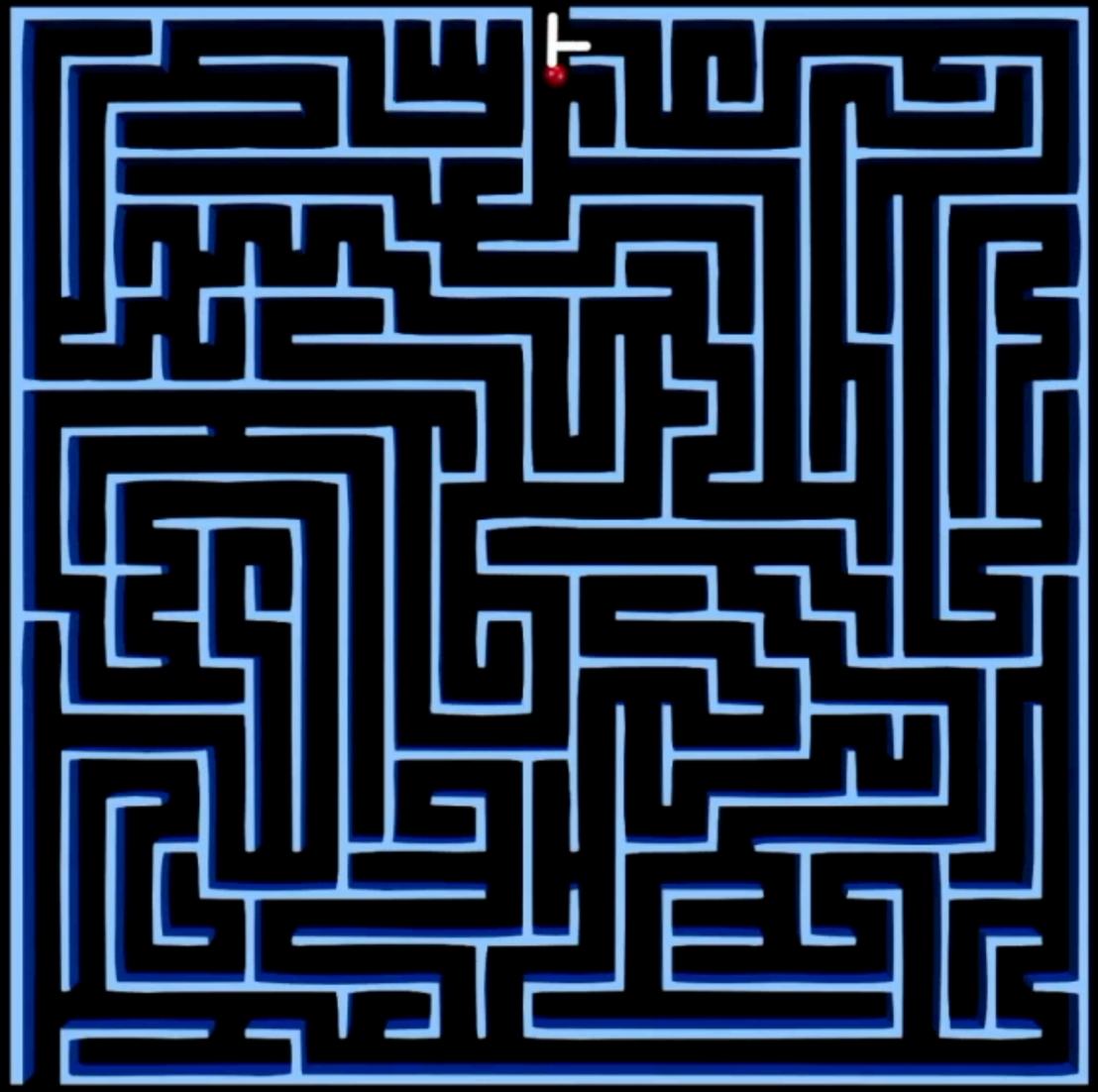
Superposition d’états



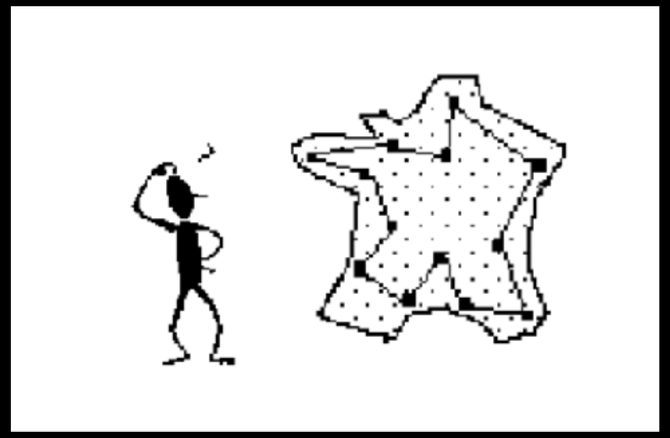
Etats classiques

Etats quantiques

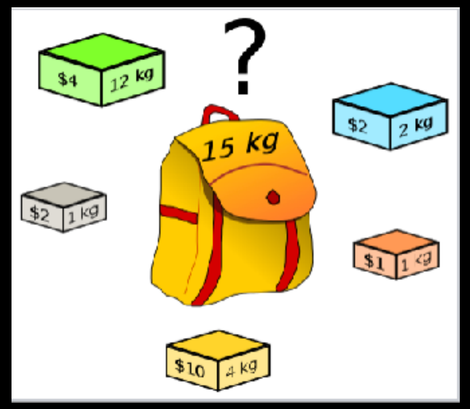




Classe de problèmes NP



Problème
du voyageur de commerce



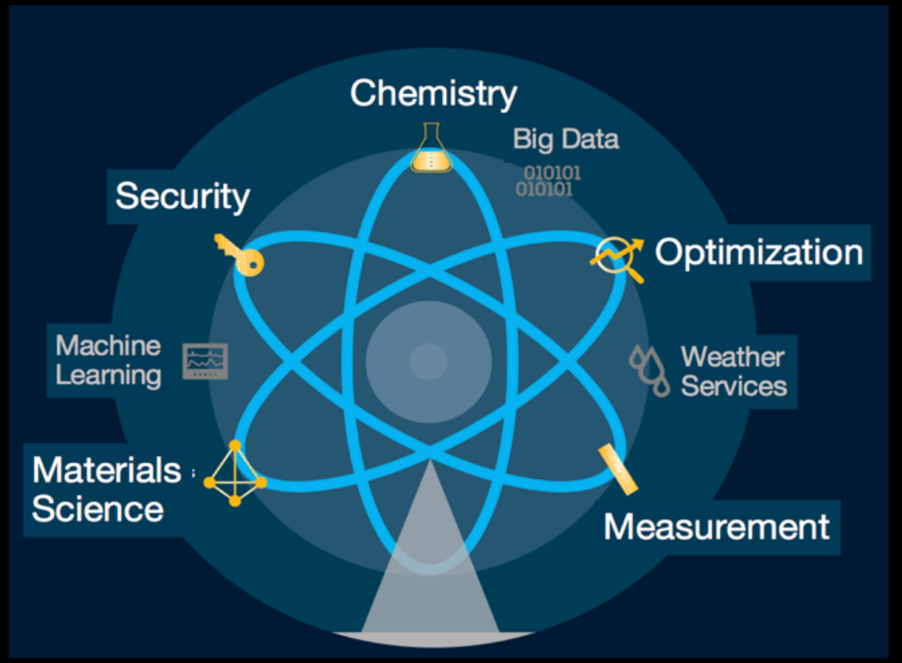
Problème
du sac à dos

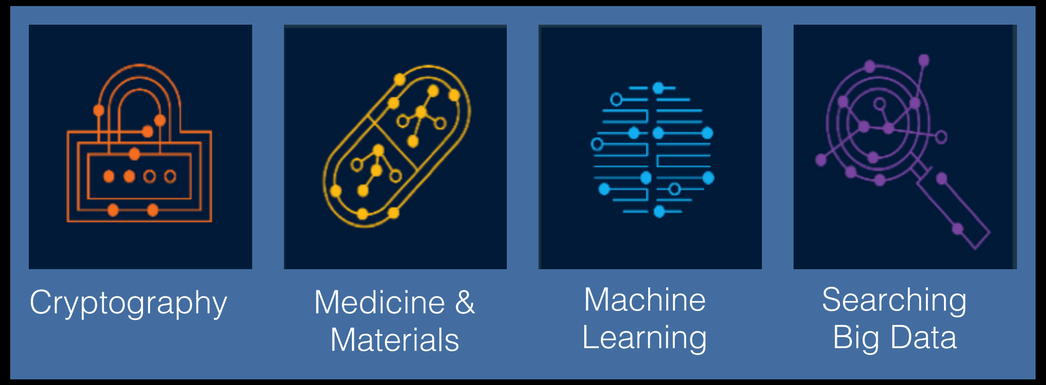
Problème SAT
(Résolution d'une formule logique propositionnelle)

Classe de problèmes NP-COMPLET

Applications éligibles par industrie

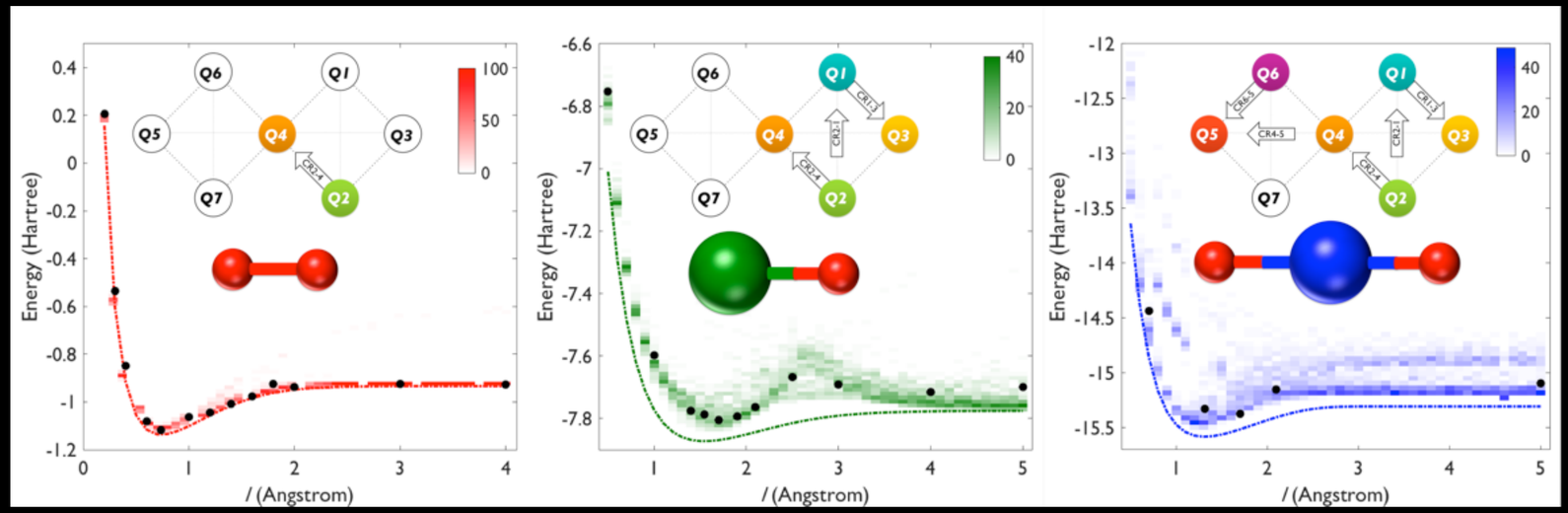
- **R&D (“Sciences de Matériaux”)**
  - Molecular Simulation et Quantum Chemistry
  - Material Sciences
- **Banques**
  - Risques (Méthode de Monte Carlo, ...)
  - Gestion de portefeuilles
- **Energie / Télécom / Transport**
  - Logistique, Géoloc, Sensing, Supply chain (optim)
  - Allocation de ressources
- **Santé**
  - Génétique / Séquençage (optim, graphe, ...)
- **Défense & Sécurité**
  - Confidentialité (Cryptographie, ...)





World's first demonstration of modeling LiH Lithium hybride (LiH) and Beryllium hybride (BeH2) using a quantum computer.

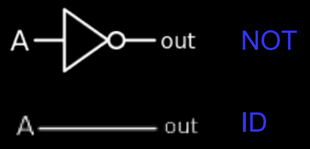
2 qubits 4 qubits 6 qubits



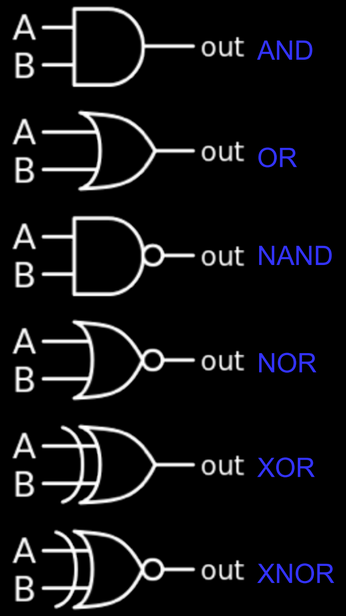
N electrons ~ N qubits

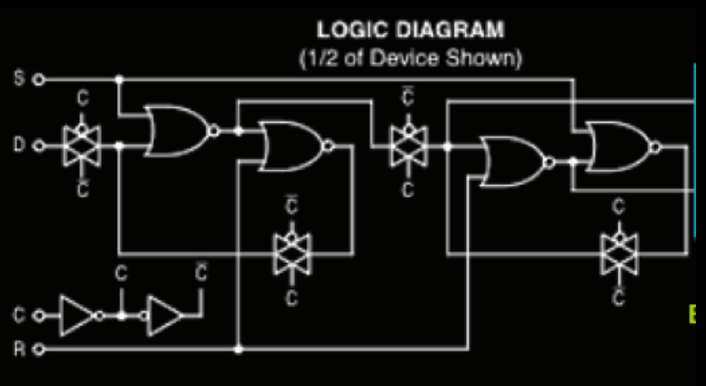
Bit

1 Bit Gates



2 Bit Gates

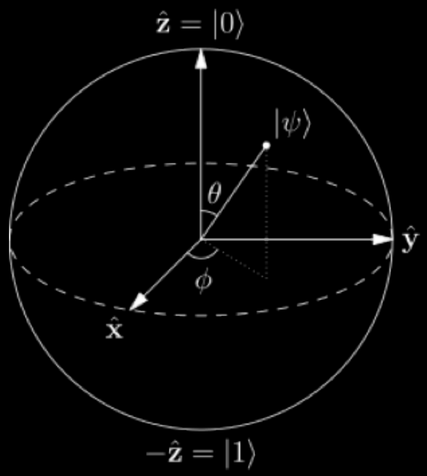




Qubit

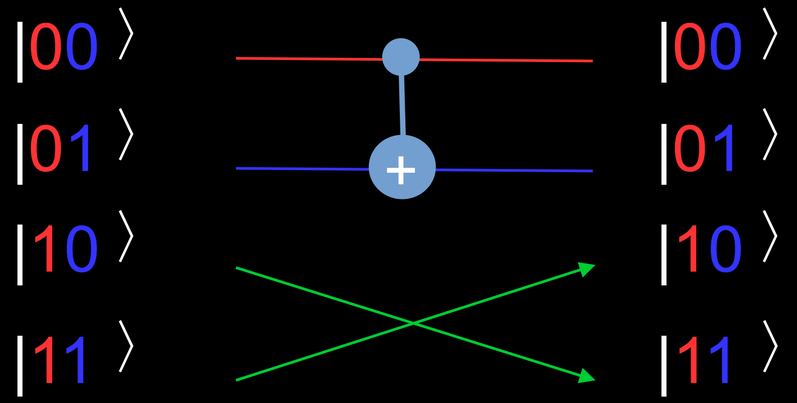
IN

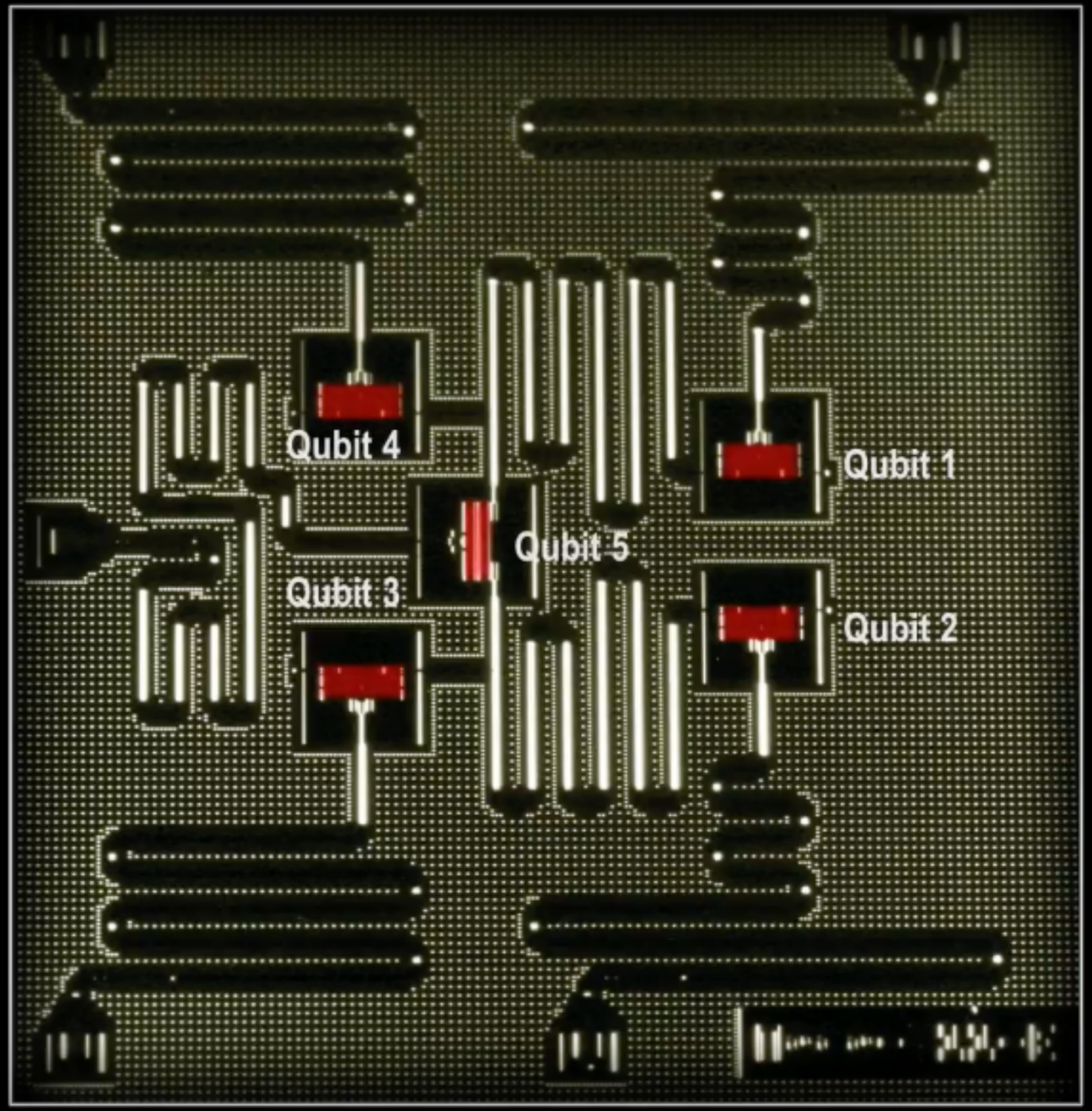
OUT



1-qubit Gate
→ Rotation of Bloch-Sphere
→ Adding to length and latitudee

2-qubit Gate: CNOT
→ Controlled NOT
→ Exchange of |10〉 and |11〉

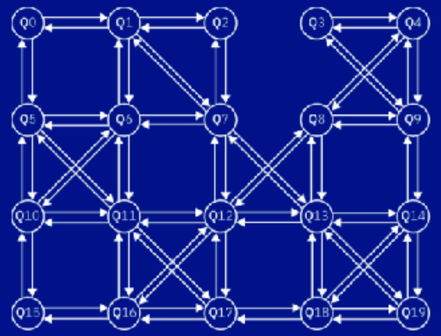



Qubit 4
Qubit 1
Qubit 5
Qubit 3
Qubit 2

Processeurs QUBITS IBM Q

Backend: QS1_1 (20 Qubits)

ACTIVE | AVAILABLE TO HUBS, PARTNERS, AND MEMBERS OF THE IBM Q NETWORK

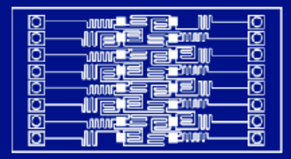


Last Calibration: 2018-02-27 22:24:11

	Q0	Q1	Q2	Q3	Q4	Q5	Q6
Frequency (GHz)	4.84	4.48	4.86	5.03	5.01	4.91	4.89
T1 (µs)	77.79	90.08	74.89	63.23	70.35	55.70	139.82
T2 (µs)	35.56	28.54	31.85	19.87	35.94	43.16	44.53
Gate error (10^{-3})	2.01	2.56	6.64	7.27	1.96	1.95	1.96
Readout error (10^{-2})	8.30	31.50	20.70	21.10	7.20	9.50	15.45
MultiQubit gate error (10^{-2})	CX0_1 3.85	CX1_0 3.85	CX2_1 14.18	CX3_4 10.00	CX4_3 10.00	CX5_0 1.27	CX6_1 11.39
	CX0_5 1.27	CX1_2 7.64	CX2_6 12.85	CX3_9 5.05	CX4_8 4.69	CX5_6 2.21	CX6_2 12.85
		CX1_6 11.39	CX2_7 9.61		CX4_9 3.47	CX5_10 1.90	CX6_5 2.21
		CX1_7 22.80				CX5_11 1.58	CX6_7 3.67
							CX6_10 2.35
							CX6_11 2.98

Backend: ibmqx5 (16 Qubits)

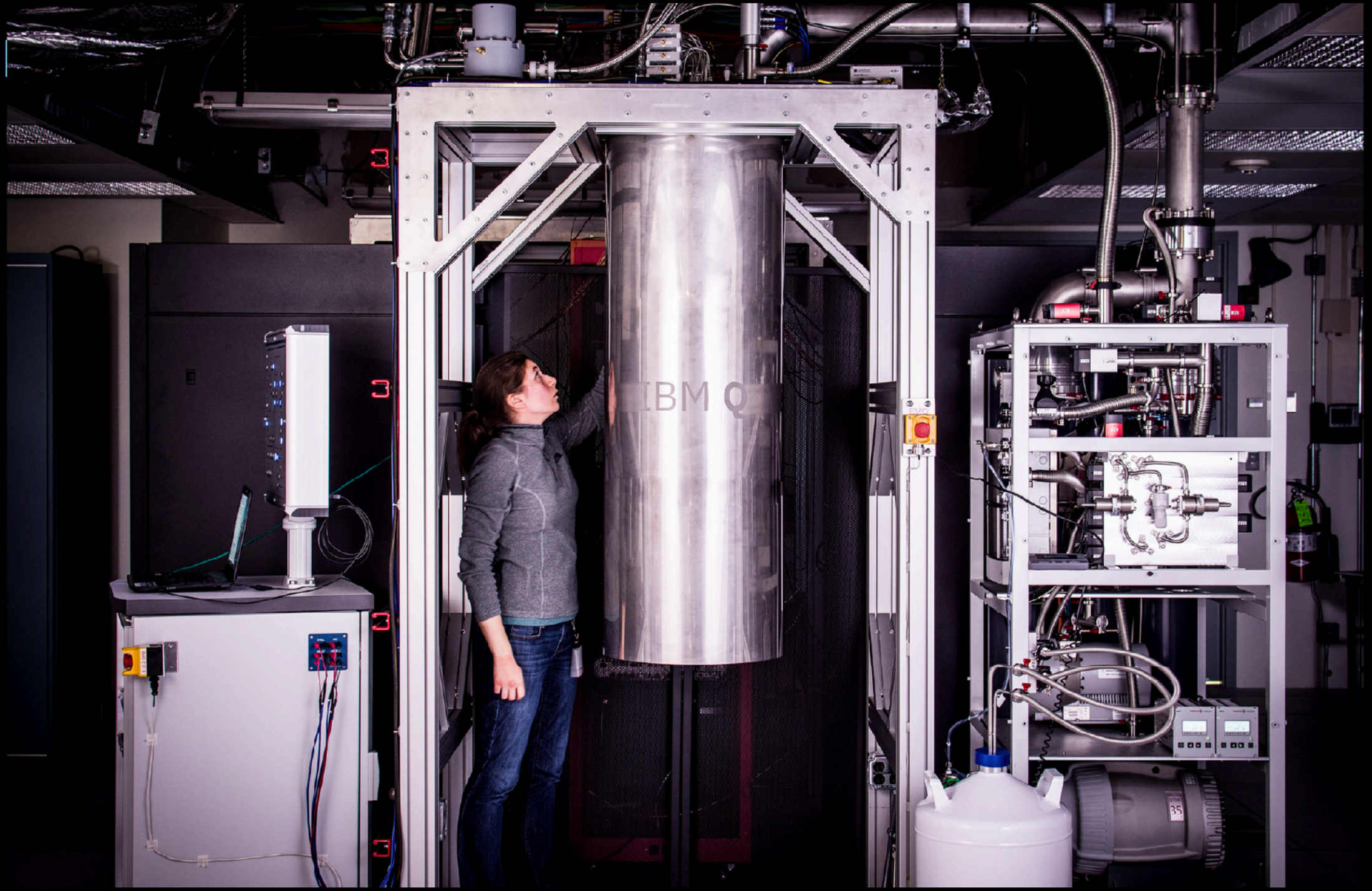
ACTIVE | AVAILABLE ON QISKIT



Last Calibration: 2018-02-27 13:40:32
Fridge Temperature: 0.0133997 K

More details

	Q0	Q1	Q2	Q3	Q4	Q5	Q6
Frequency (GHz)	5.26	5.40	5.28	5.08	4.98	5.15	5.31
T1 (µs)	39.60	40.80	34.70	38.20	29.80	30.10	43.60
T2 (µs)	34.90	70.80	41.60	60.70	59.60	18.70	65.10
Gate error (10^{-3})	2.19	3.37	3.81	2.61	1.34	4.07	1.55
Readout error (10^{-2})	5.43	6.40	4.09	6.64	6.61	5.77	3.87
MultiQubit gate error (10^{-2})		CX1_0 5.21	CX2_3 4.92	CX3_4 3.90		CX5_4 5.69	CX6_5 5.36
		CX1_2 5.51		CX3_14 5.50			CX6_7 2.73
							CX6_11 3.34


IBM Q

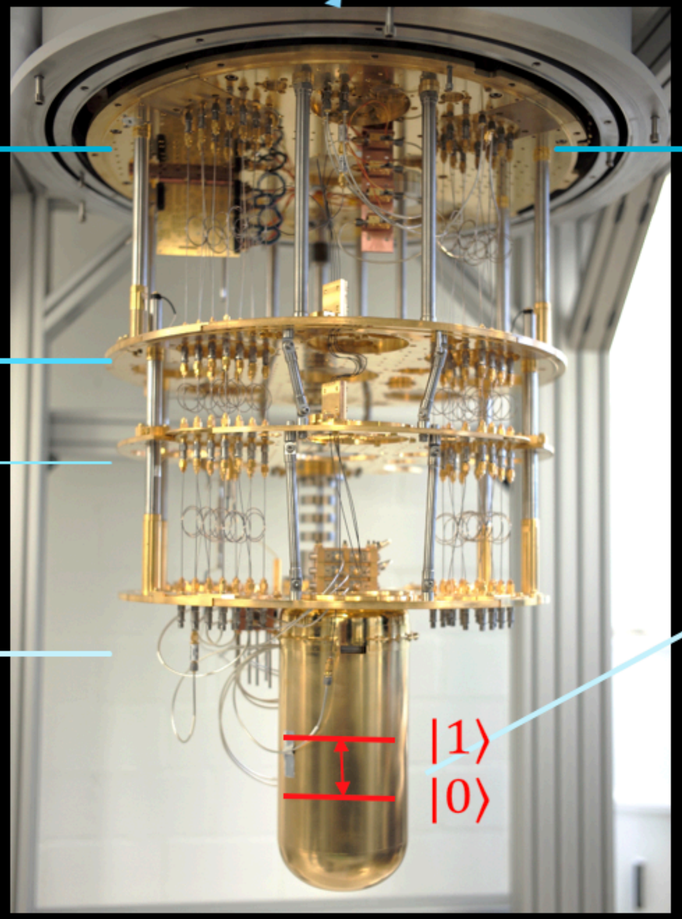
Microwave electronics

2.7K

0.8K

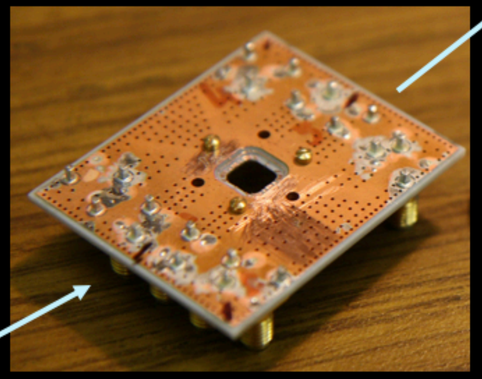
0.1K

0.02K

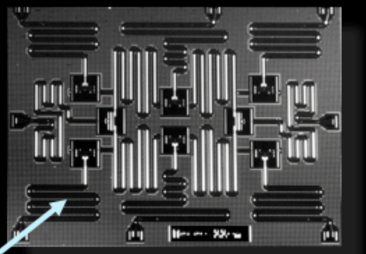


Dilution cryostat

-270°C



PCB with the qubit chip at 20mK
Protected from the environment by multiple shields



Chip with superconducting qubits and resonators

Quantum Volume is a metric of computational value...

IBM Q

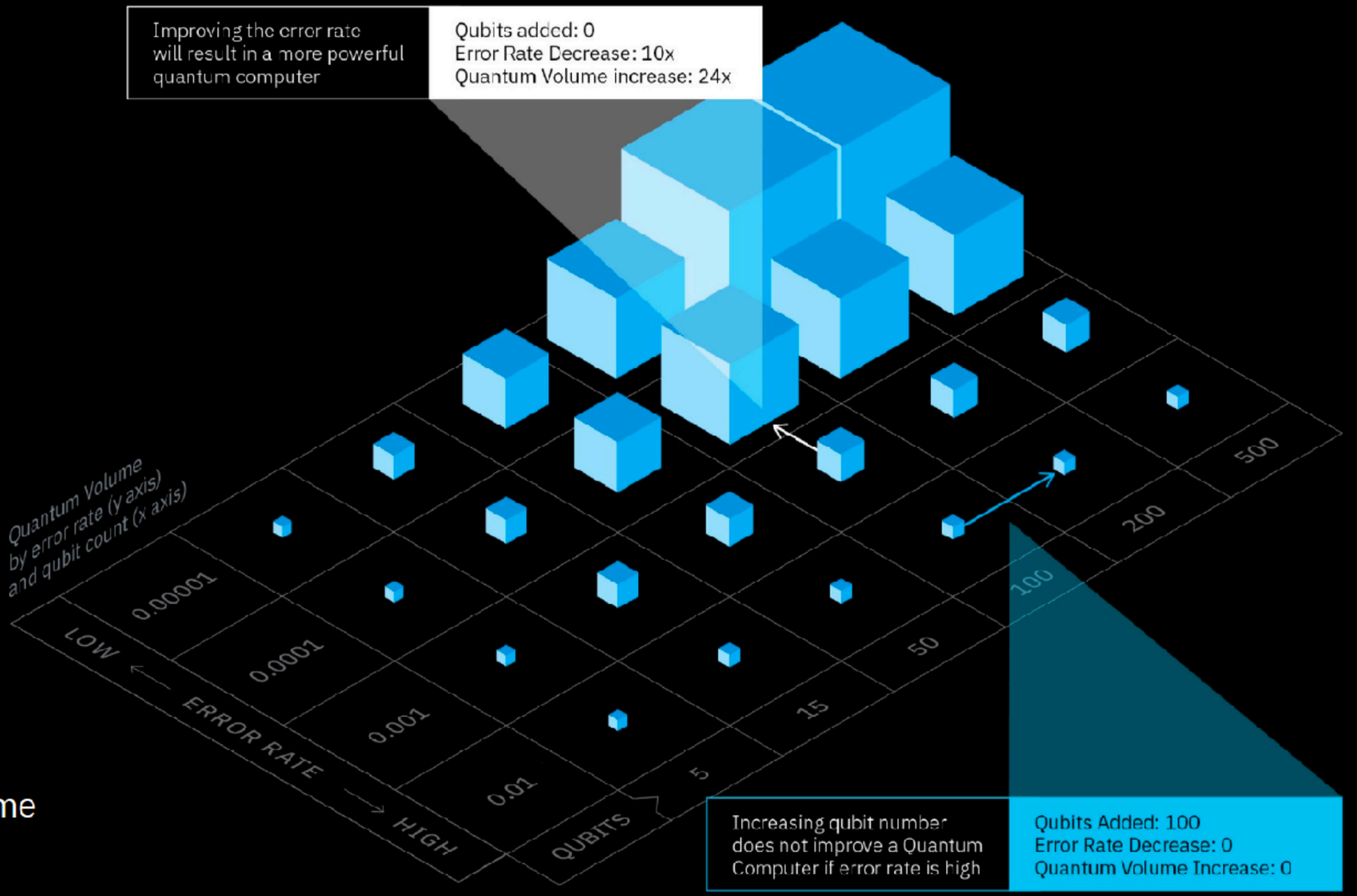
Qubit Count
(more is better)

Errors
(less is better)

Connectivity
(more is better)

Gate set
(more is better)

Technical description:
https://ibm.biz/qiskit-quantum-volume



IBM Q experience

IBM Q

80,000+ users

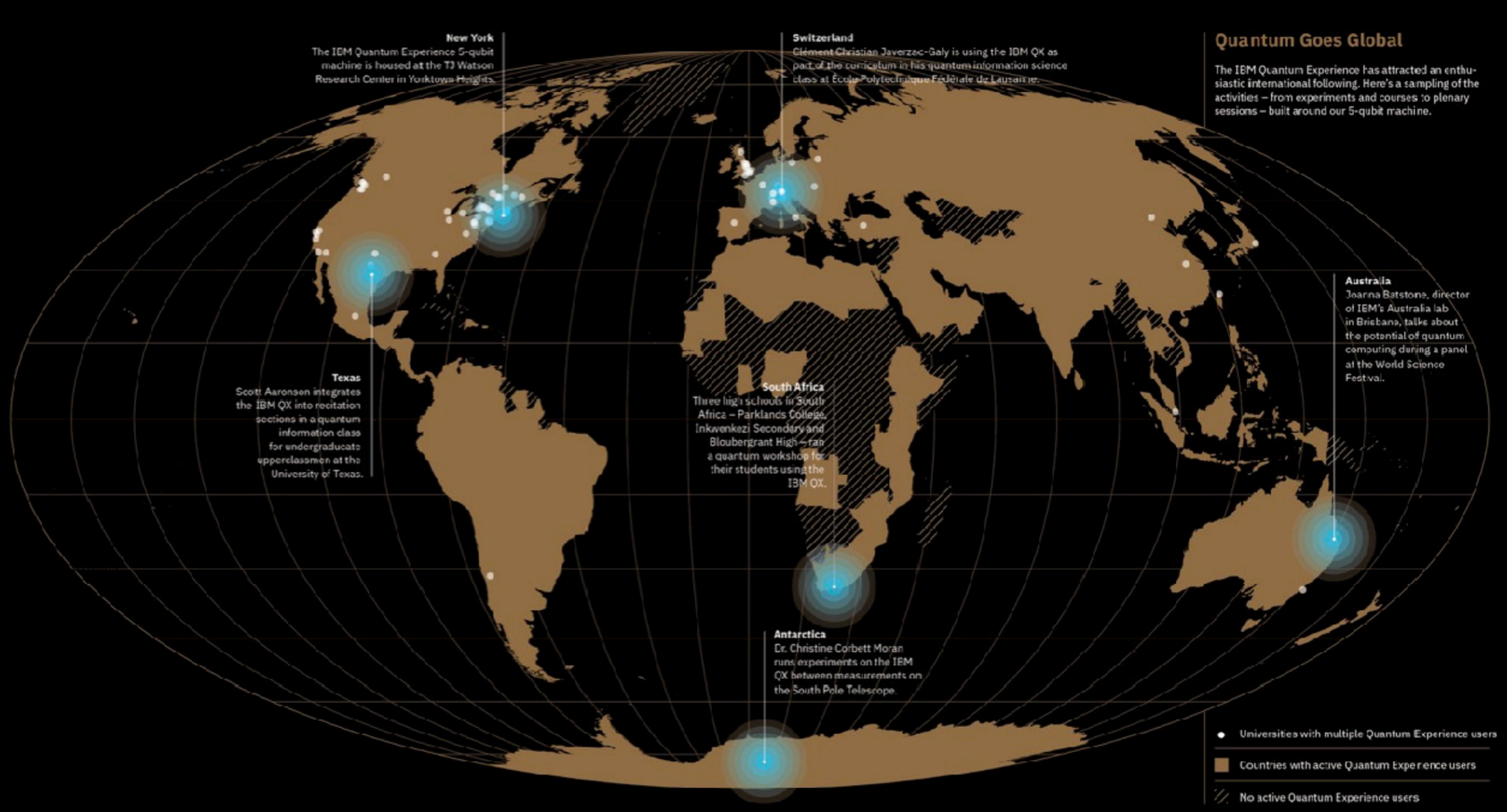
All 7 continents

> 1500 colleges and universities, 300 high schools, 300 private institutions

Over 3.0 M experiments

75+ external papers

Recent upgrade to include an additional 20 Q backend



Access to the world's most advanced quantum resources...

IBM Q

IBM Q will provide access to quantum systems
and the tools and enablement required to
support your quantum strategy

Access to Quantum Hardware

- Shared systems
- 17-qubit systems in 2017
- 50-qubit upgrades

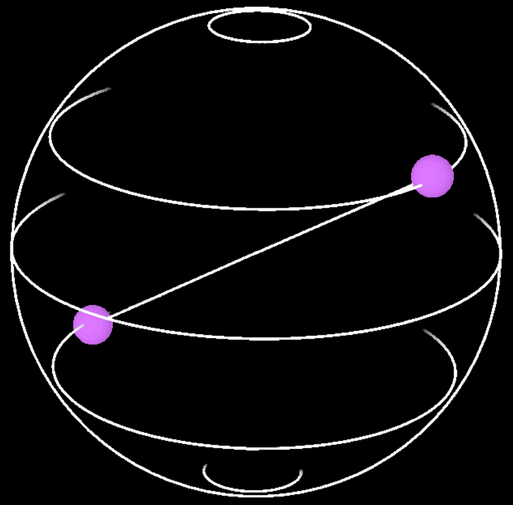
Enabled with a Quantum Software Stack

- OPENQASM support
- Accessible API
- Quantum QISKIT SDK
- Example Jupyter notebooks
- Quantum Simulator

IBM is building the larger ecosystem to ensure success...

IBM Q

QISKit

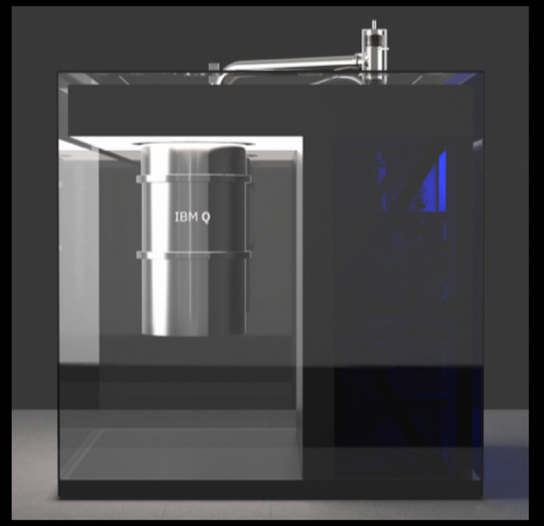


Open SDK and libraries at qiskit.org

IBM Q Experience

Free public access to introductory resources ibm.com/ibmq

IBM Q Systems



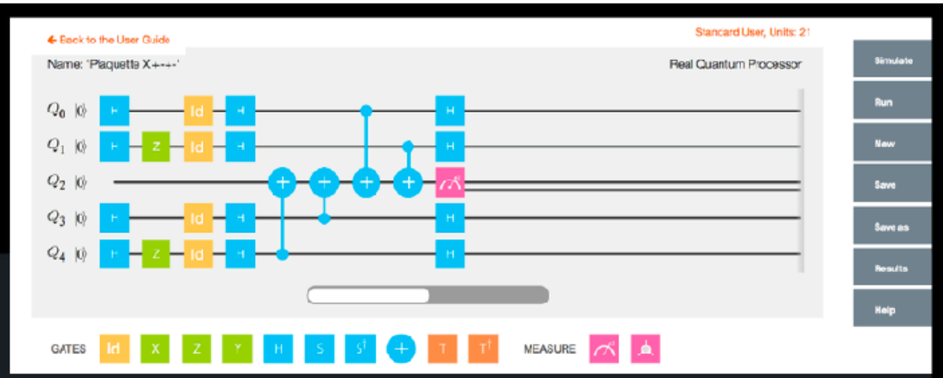
To support identification and launch of first commercial applications

IBM Q Network: QISKit (Quantum Software Information Kit)

https://www.qiskit.org

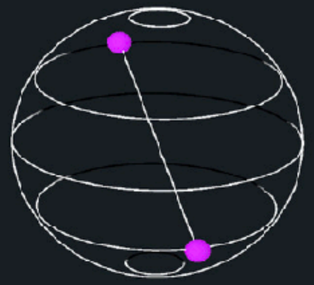
QISKit underpins everything that we do. It must remain the number one platform for quantum computing and win versus the competition in the coming years.

http://www.ibm.com/quantumexperience



IBM QX Features

- Tutorial
- Simulation
- Graphical programming
- QASM language
- API (SDK coming soon)
- Active user community



QISKit

Quantum Information Software Kit

Latest version v0.3.6

The Quantum Information Software Kit (QISKit for short) is a software development kit (SDK) for working with OpenQASM and the IBM Q experience (QX).

GitHub

Road map

Learn

Use QISKit to create quantum computing programs, compile them, and execute them on one of several backends (online Real quantum processors, and simulators).

Tutorials

Documentation

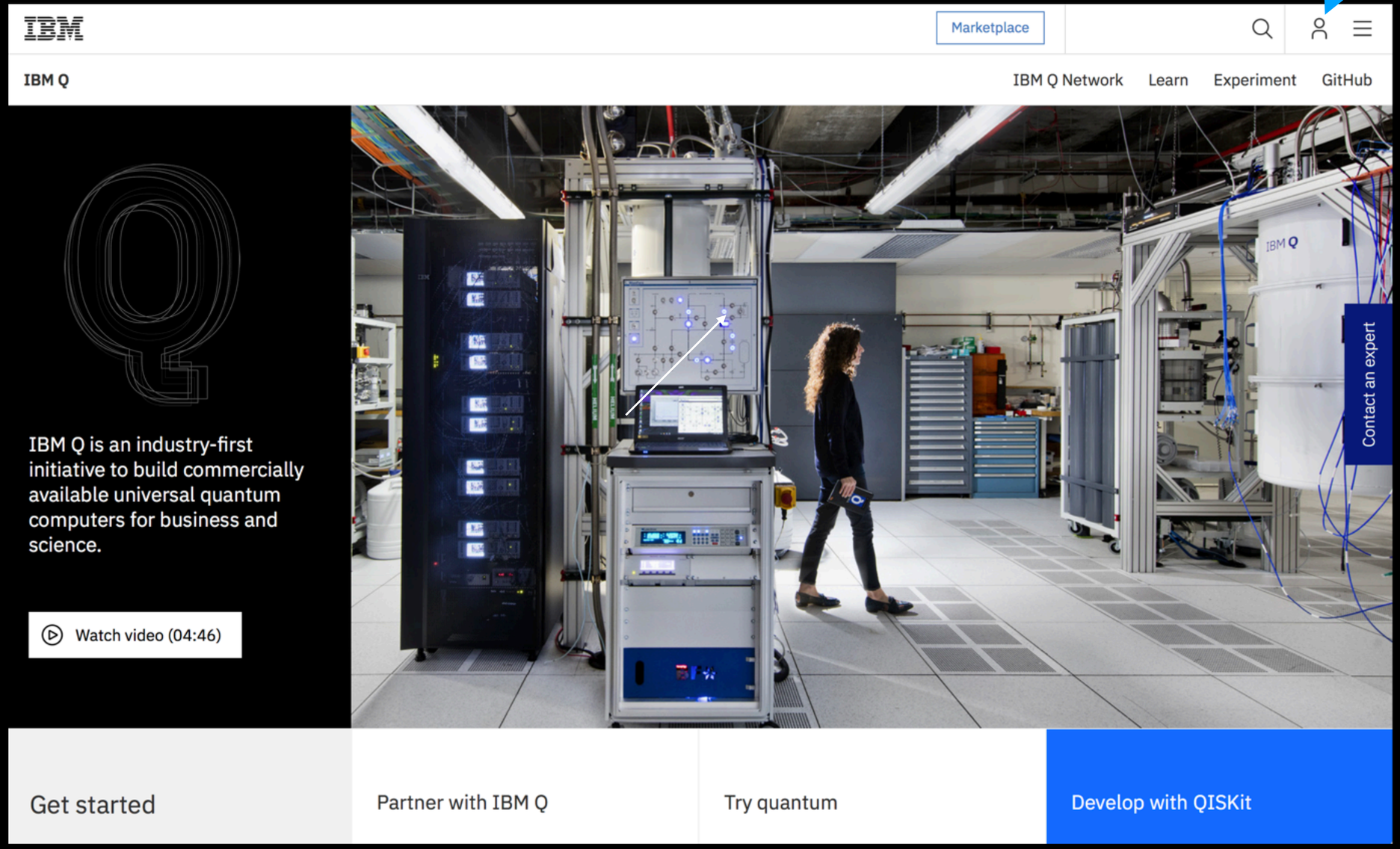
Run a quantum program

```
[python3] $ pip install qiskit
```

```
from qiskit import QuantumProgram
qp = QuantumProgram()
qr = qp.create_quantum_register('qr',2)
cr = qp.create_classical_register('cr',2)
qc = qp.create_circuit('Bell',[qr],[cr])
qc.h(qr[0])
qc.cx(qr[0], qr[1])
qc.measure(qr[0], cr[0])
qc.measure(qr[1], cr[1])
result = qp.execute('Bell')print(result.get_counts('Bell'
```

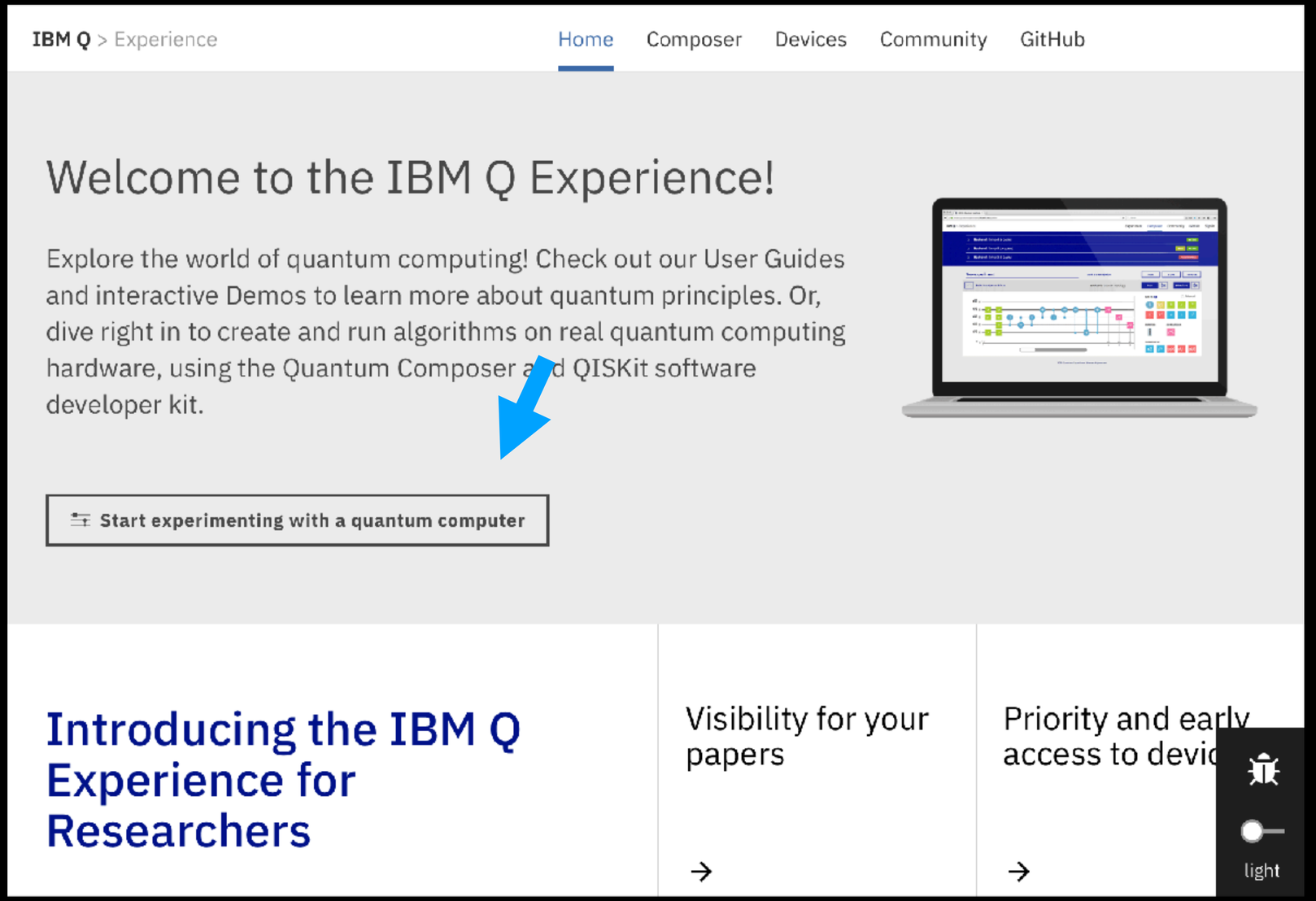
Création d'un compte IBM Q

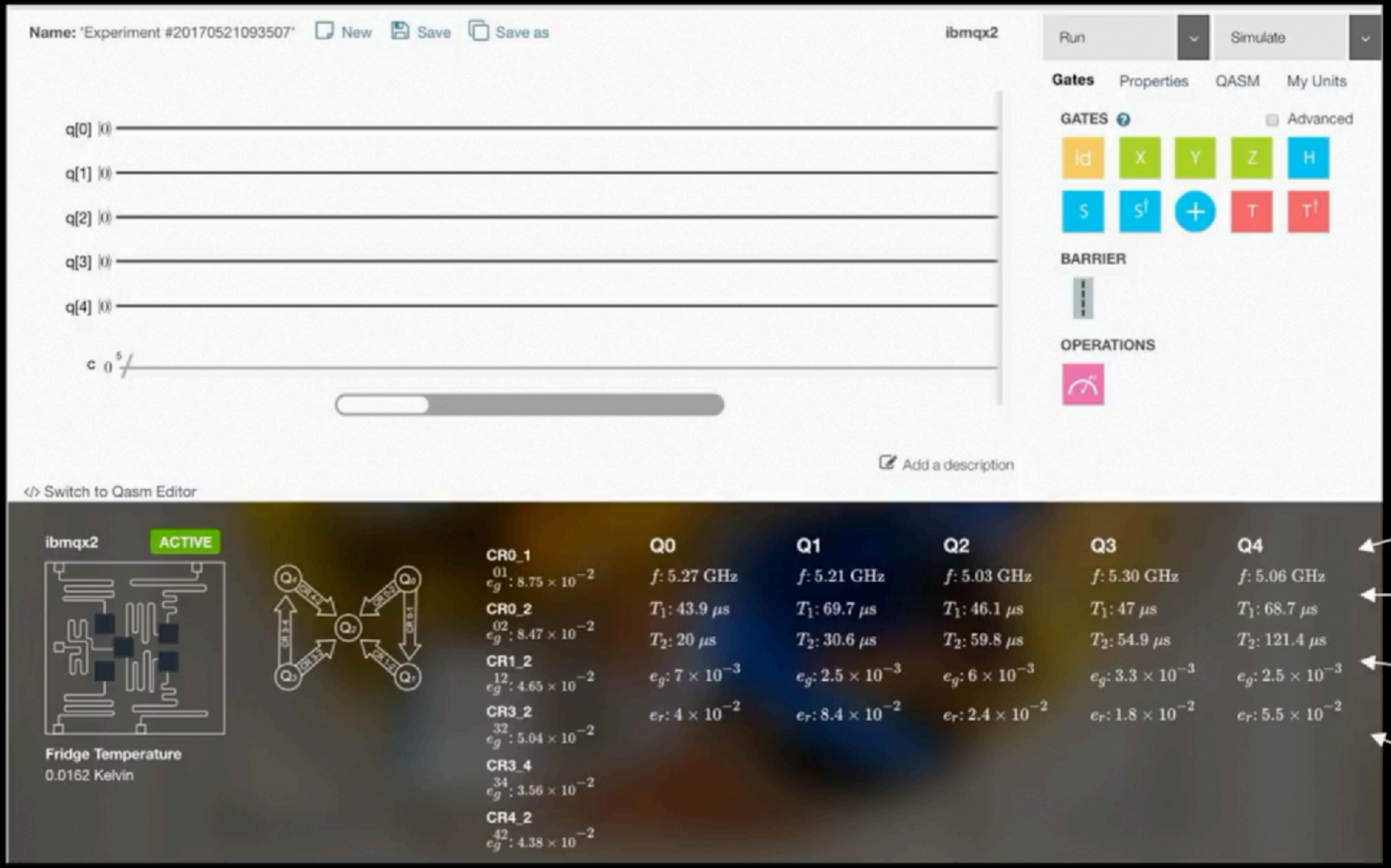
https://www.research.ibm.com/ibm-q/



Création d'un compte IBM Q

https://www.research.ibm.com/ibm-q/





Quantum gates

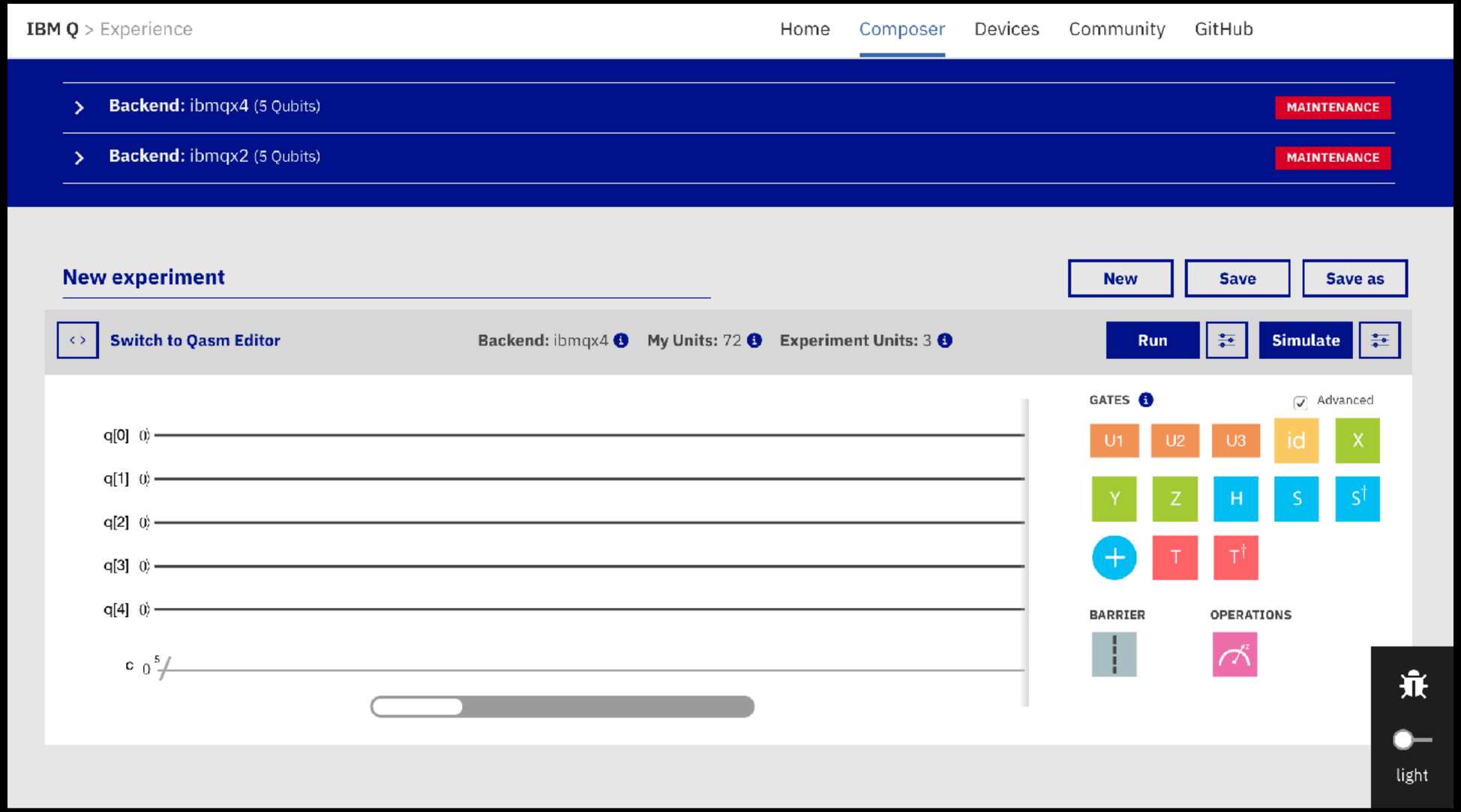
5-qubits

Each a diff. freq.

Good coherence

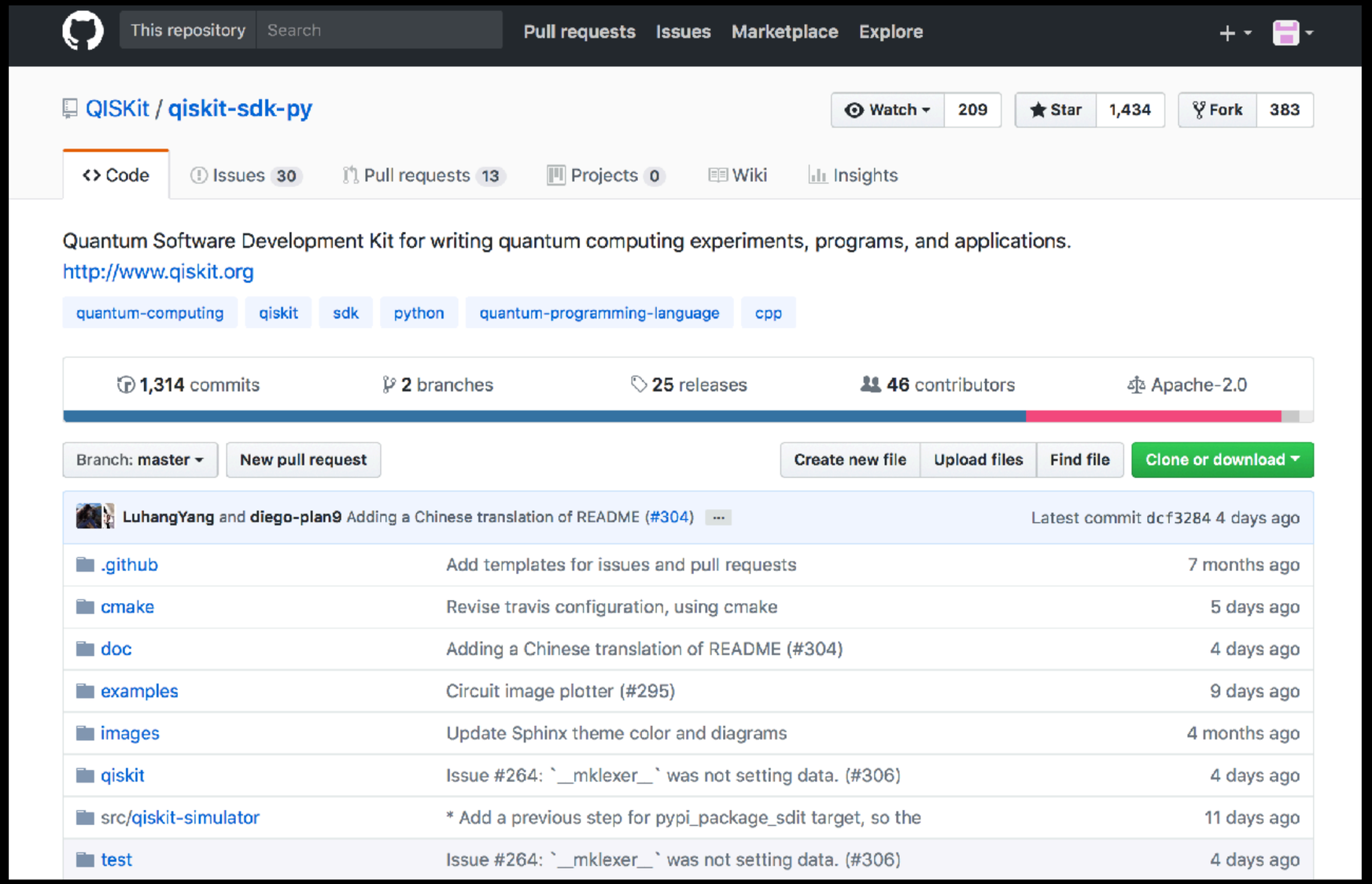
Low error rates

Expérience IBM Q



github

https://github.com/QISKit/qiskit-sdk-py



jupyter Superposition+Entanglement+Bell+States+and+the+CHSH+inequality (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Python 3

Markdown CellToolbar

IBM Quantum

Superposition, Entanglement, Bell states, and the C
with the Python Interface for the Web API of the IBM Quantum Experience

The latest version of this notebook is available on https://github.ibm.com/QuantumSD

For more information about how to use the Quantum Experience consult the IBM Qua the community.

Introduction

Many people tend to think quantum physics is hard math. This is actually not true, qua algebra classes you probably did in first year university or even at high school. What is no simple underlying theory. Instead you need to accept counter-intuitive ideas and in together we feel that these can be distilled into two principles.

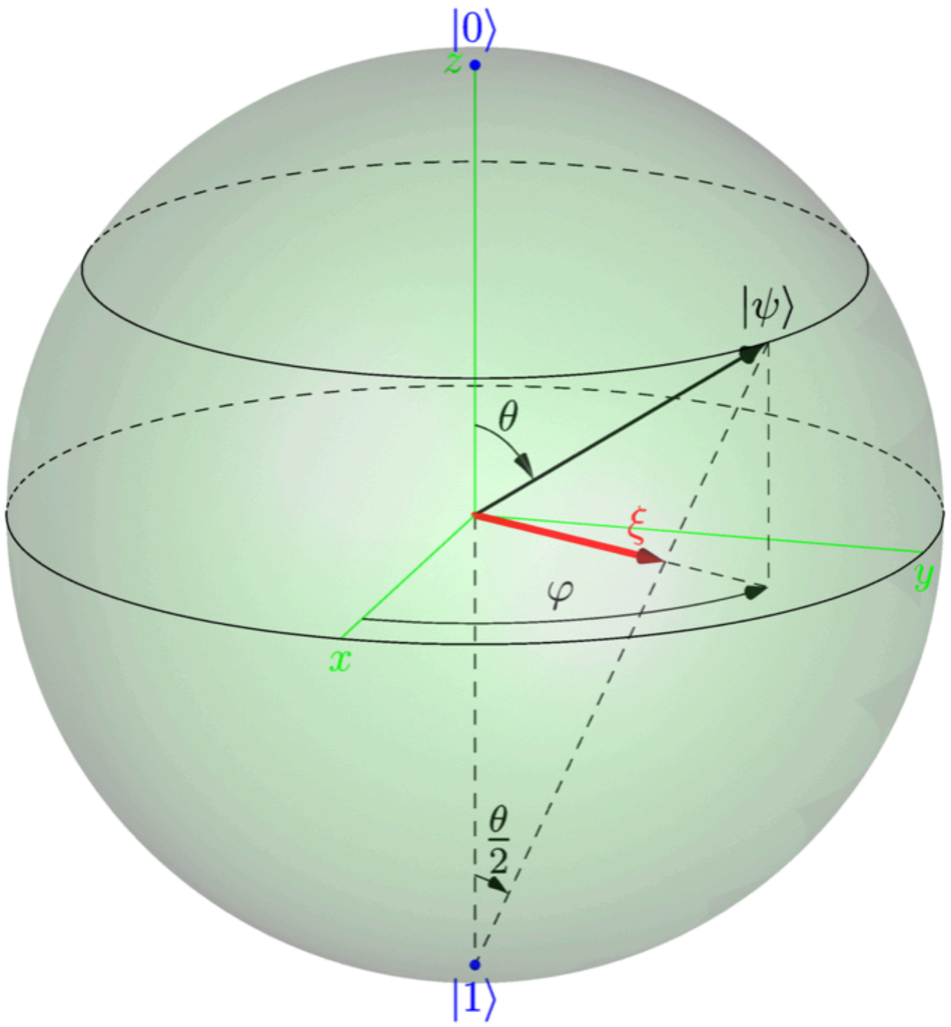
1. A physical system in a perfectly definite state can still behave randomly.
2. Two systems that are too far apart to influence each other can nevertheless beha random, are somehow strongly correlated.

In [26]:

```
from IBMQuantumExperience import IBMQuantumExperience
from IPython.display import Image, display
import matplotlib.pyplot as plt
import numpy as np
import scipy as sp
import scipy.linalg
%matplotlib inline

config = {"url": 'https://quantumexperience.ng.bluemix.net/api'} # base url for the
import Qconfig
api = IBMQuantumExperience.IBMQuantumExperience(Qconfig.APIemail,Qconfig.APIpasswor

def plotHistogramData(values, labels):
    """Plot a histogram of data"""
    N = len(values)
    ind = np.arange(N)  # the x locations for the groups
    width = 0.35       # the width of the bars
    fig, ax = plt.subplots()
    rects1 = ax.bar(ind, values, width, color='r')
    # add some text for labels, title and axes ticks
    ax.set_ylabel('Probabilities')
    ax.set_xticks(ind + (width/2.))
    ax.set_xticklabels(labels)
    def autolabel(rects):
        # attach some text labels
        for rect in rects:
            height = rect.get_height()
            ax.text(rect.get_x() + rect.get_width()/2., 1.05*height,
                    '%f' % float(height),
                    ha='center', va='bottom')
    autolabel(rects1)
    plt.show()
```

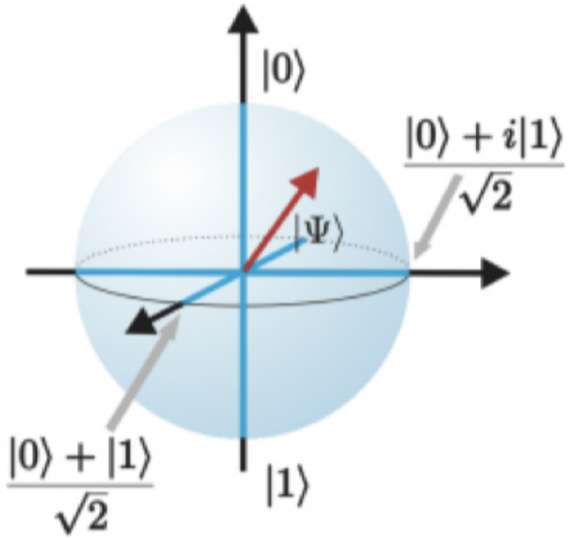
Sphère de Bloch (ou de Poincaré)

Qubit = point dans un espace de Hilbert de dim 2 sur le corps des complexes



$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle$$



Définition des portes quantiques

U1

The first physical gate of the Quantum Experience. It is a one parameter single-qubit phase gate with zero duration.

QASM Matrix

U2

The second physical gate of the Quantum Experience. It is a two parameter single-qubit gate with duration one unit of time.

QASM Matrix

U3

The third physical gate of the Quantum Experience. It is a three-parameter single-qubit gate with duration 2 units of gate time.

QASM Matrix

id

The identity gate performs an idle operation on the qubit for a time equal to one unit of time.

QASM Matrix

X

The Pauli X gate is a π-rotation around the X axis and has the property that $X \rightarrow X$, $Z \rightarrow -Z$. Also referred to as a bit-flip.

QASM Matrix

Y

The Pauli Y gate is a π-rotation around the Y axis and has the property that $X \rightarrow -X$, $Z \rightarrow -Z$. This is both a bit-flip and a phase-flip, and satisfies $Y = XZ$.

QASM Matrix

Z

The Pauli Z gate is a π-rotation around the Z axis and has the property that $X \rightarrow -X$, $Z \rightarrow Z$. Also referred to as a phase-flip.

QASM Matrix

H

The Hadamard gate has the property that it maps $X \rightarrow Z$, and $Z \rightarrow X$. This gate is required to make superpositions.

QASM Matrix

S

The Phase gate that is $\sqrt{Z}$ and has the property that it maps $X \rightarrow Y$ and $Z \rightarrow Z$. This gate extends H to make complex superpositions.

QASM Matrix

S†

The Phase gate that is the transposed conjugate of S and has the property that it maps $X \rightarrow -Y$, and $Z \rightarrow Z$.

QASM Matrix

The Controlled-NOT gate: a two-qubit gate that flips the target qubit (i.e. applies Pauli X) if the control is in state 1. This gate is required to generate entanglement and is the physical two qubit gate.

QASM Matrix

T

The Phase gate that is $\sqrt{S}$, which is a $\pi/4$ rotation around the Z axis. This gate is required for universal control.

QASM Matrix

T†

The Phase gate that is the transposed conjugate of T.

QASM Matrix

The barrier prevents transformations across this source line.

QASM Matrix

Measurement in the computational (standard) basis (Z).

QASM Matrix

if

Conditionally apply quantum operation

QASM Matrix

$|0\rangle$

Prepare qubits in the $|0\rangle$ state.

QASM Matrix

Quelques portes quantiques

Porte de Hadamard

$$M = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$|0> \rightarrow \frac{1}{\sqrt{2}}(|0> + |1>)$$

$$|1> \rightarrow \frac{1}{\sqrt{2}}(|0> - |1>)$$

Porte CNot

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Porte de Pauli X

$$M = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Porte de Pauli Y

$$M = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Porte de Pauli Z

$$M = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Porte SWAP

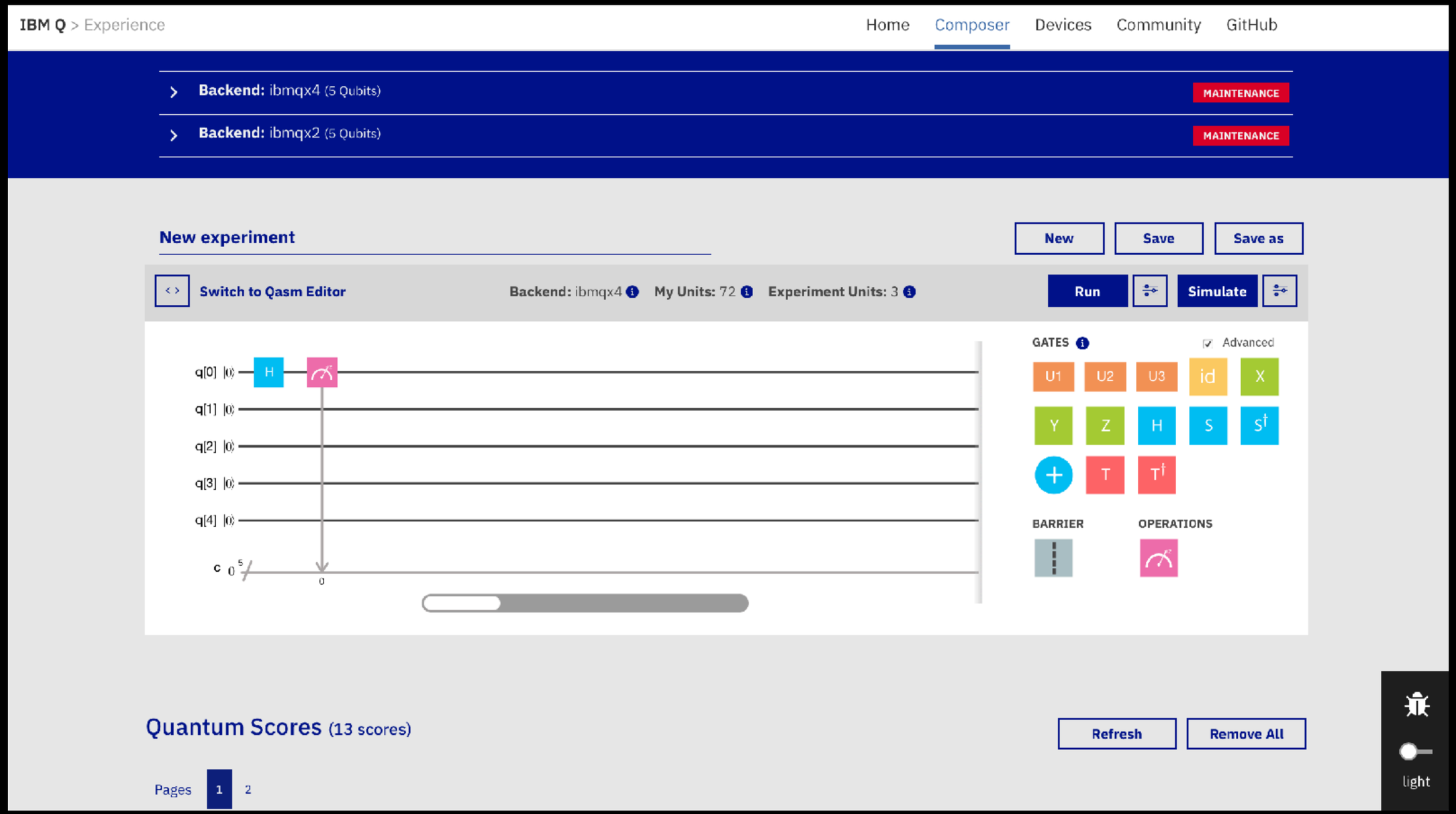
$$M = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Porte CCNOT (Toffoli)

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Portes réversibles et matrices unitaires (MM*=Id)

Exemple mise en oeuvre porte Hadamard



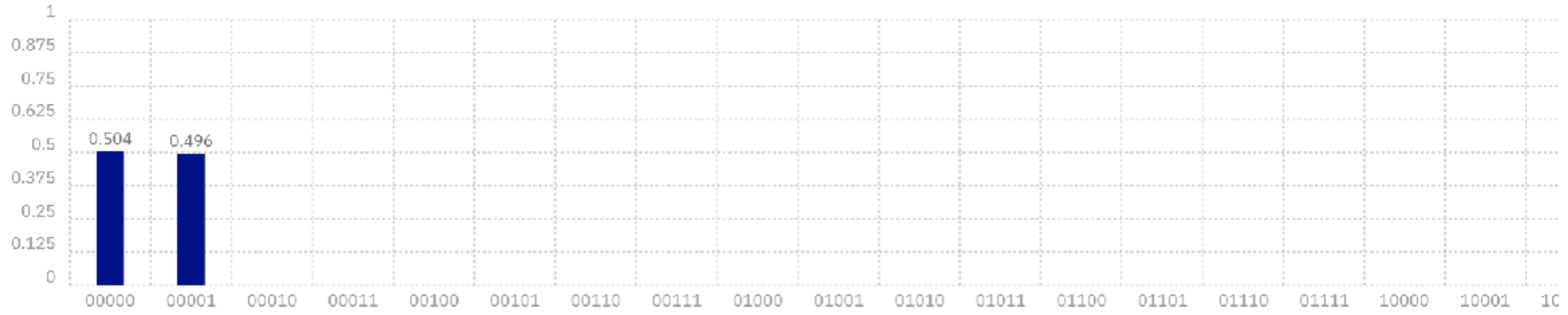
Exemple résultat porte Hadamard

Experiment #20180227223201

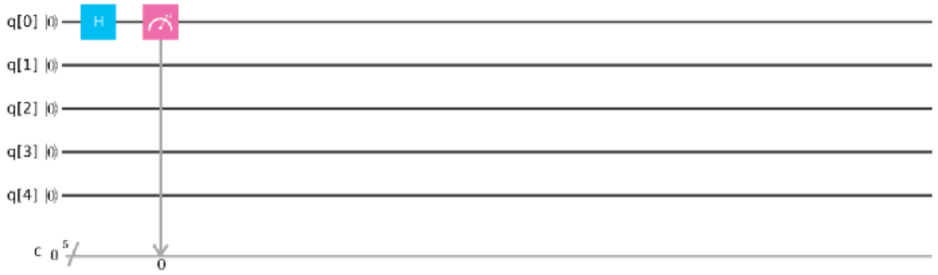
Device: ibmqx4

Quantum State: Computation Basis

Download CSV



Quantum Circuit



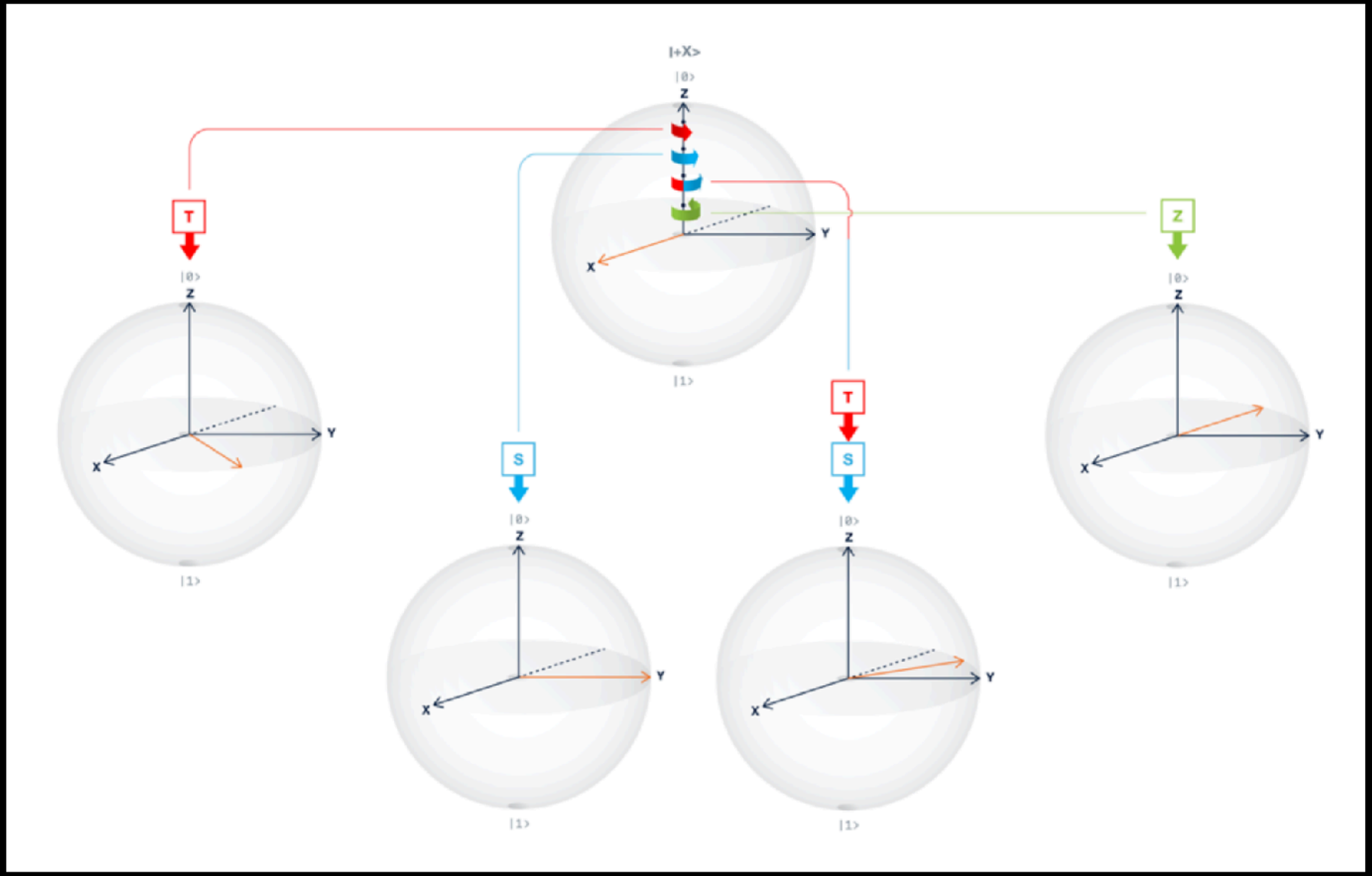
OPENQASM 2.0

```
include "qelib1.inc";

qreg q[5];
creg c[5];

h q[0];
measure q[0] -> c[0];

```

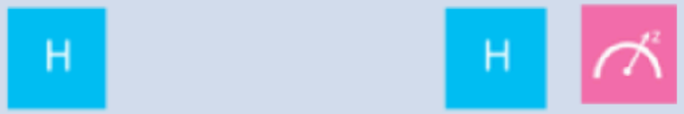
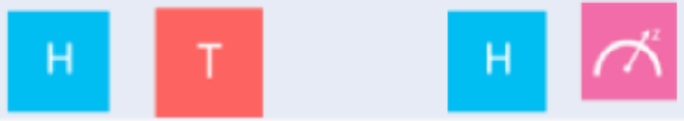
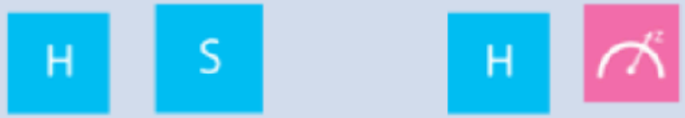
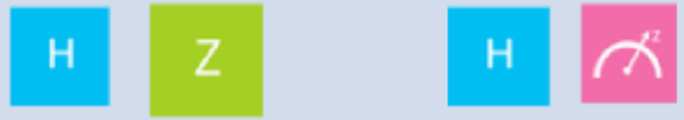
Open in Composer

Edit in QASM Editor

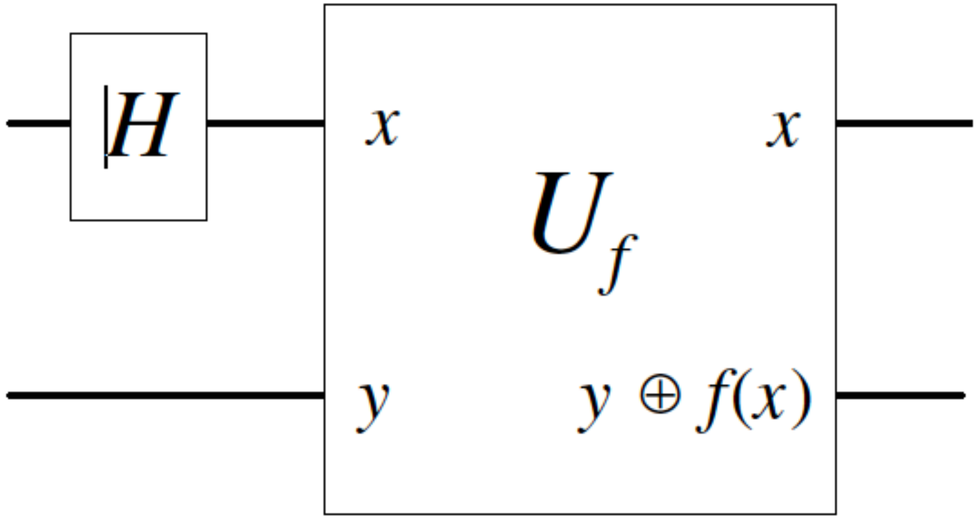
Différentes rotations suivant l’axe Z



Exemples de séquences de portes

Gate sequence	Rotation around Z	Probability of 0	Probability of 1
	0	1.0	0
	$\pi/4$	0.85	0.15
	$\pi/2$	0.50	0.50
	$3\pi/4$	0.15	0.85
	π	0	1

Le « parallélisme » quantique



$$|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$$

$$|0, 0\rangle \rightarrow \frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}}$$

Principaux algorithmes et librairies existants

- Algorithmes de Deutsch-Jozsa (1992)
  - Déterminer si une fonction binaire à valeur binaire est constante ou équilibrée

- Algorithme de Grover
  - Recherche d’éléments répondant à certains critères dans des tables

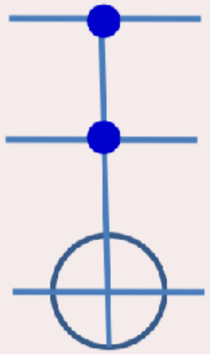
- Algorithme de shor
  - Factoriser un nombre composé

- Librairie HHL (Harrow, Hassidim, Lloyd)
  - Algèbre linéaire

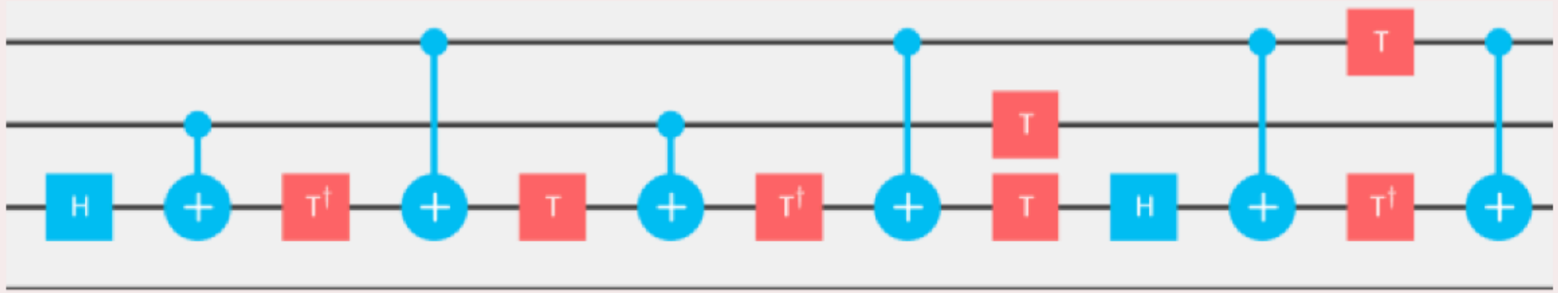
- Traitement du signal
  - Algorithme FFT

Exemples composition portes

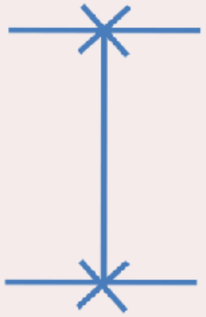
Toffoli



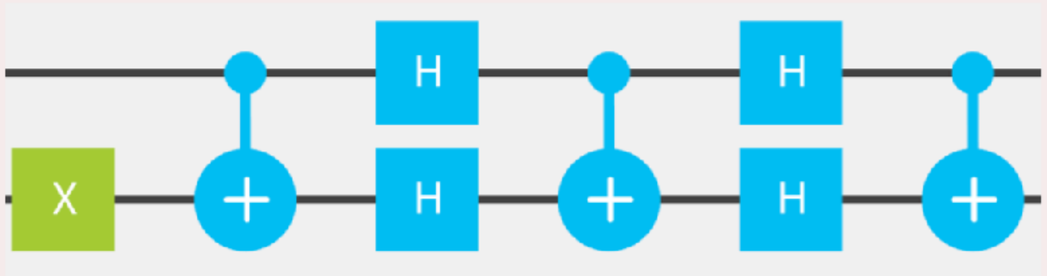
=



Swap Gates

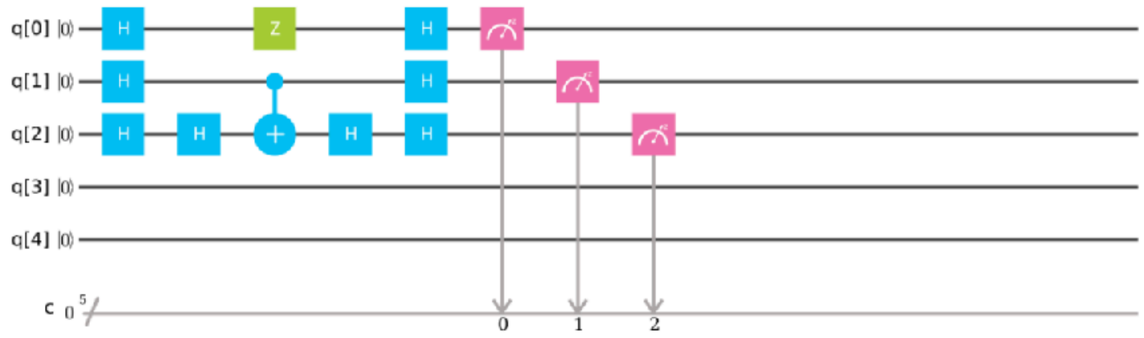


=



Exemple algorithme Deutsch-Josua

DJ N=3 Example



DJ N=3 Constant