

A large, white, stylized letter 'Q' is positioned on the left side of the slide. The background behind it is a dark blue gradient.

**Conférence Quantum Computing et IBM Q
Institut des Actuares**

19 Juin 2018

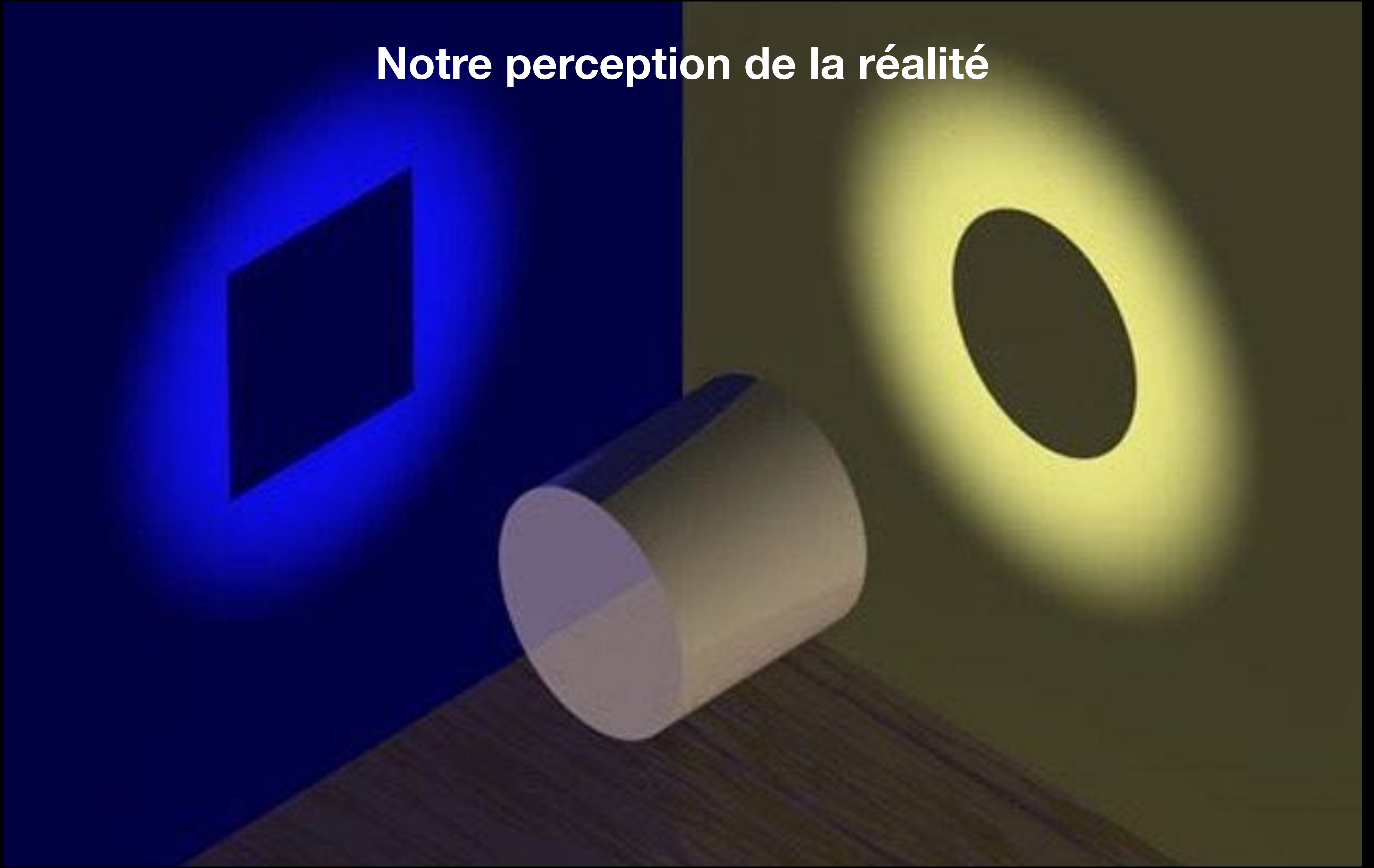
Georges Uzbelger - IBM France

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

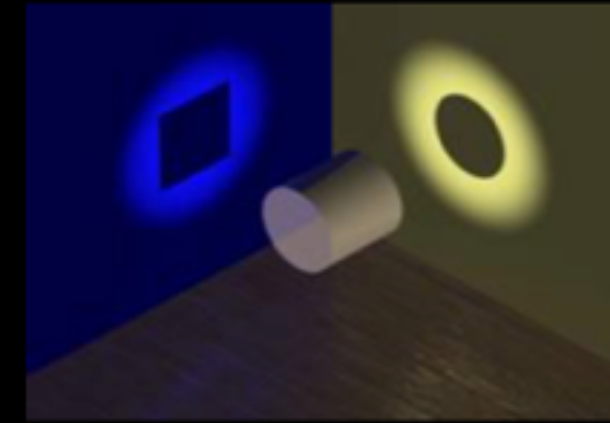


Onde ou particule ?



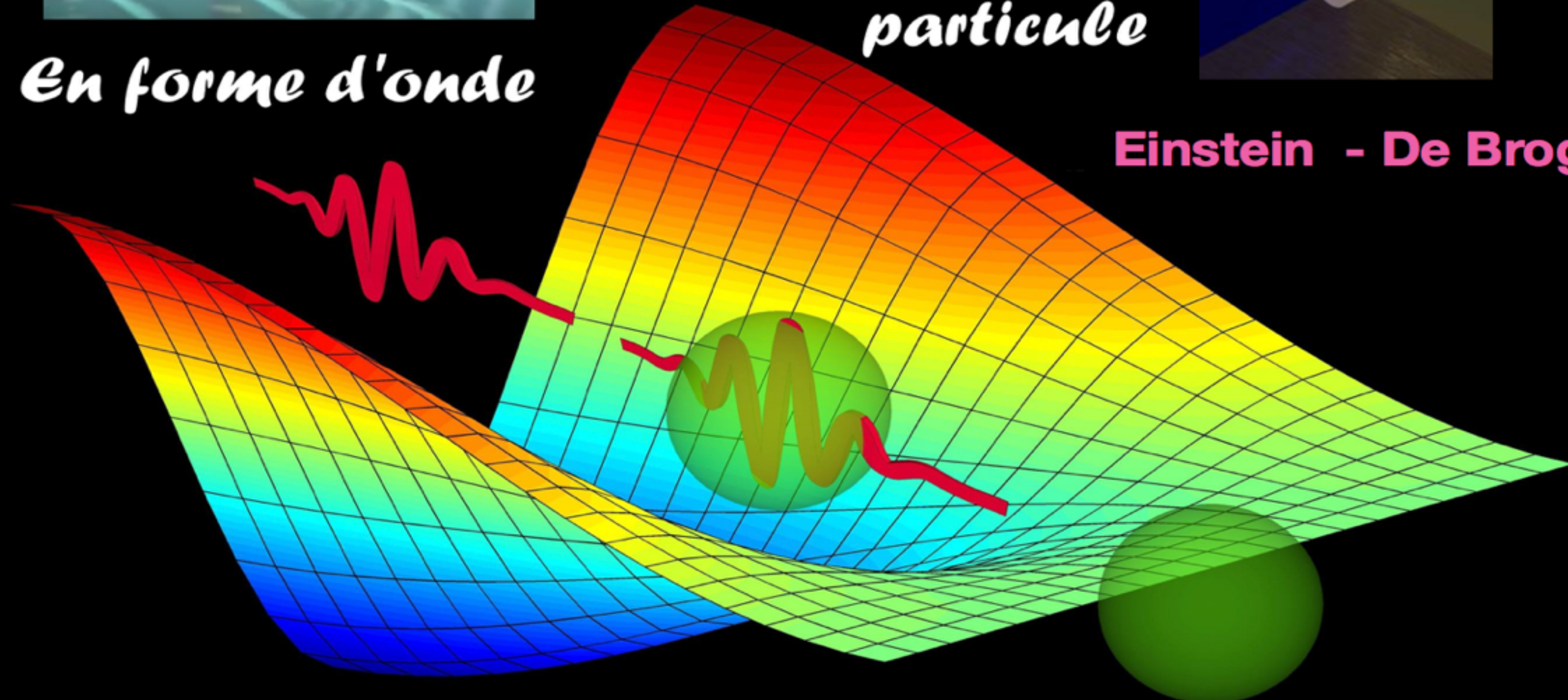
En forme d'onde

*Onde et
particule*

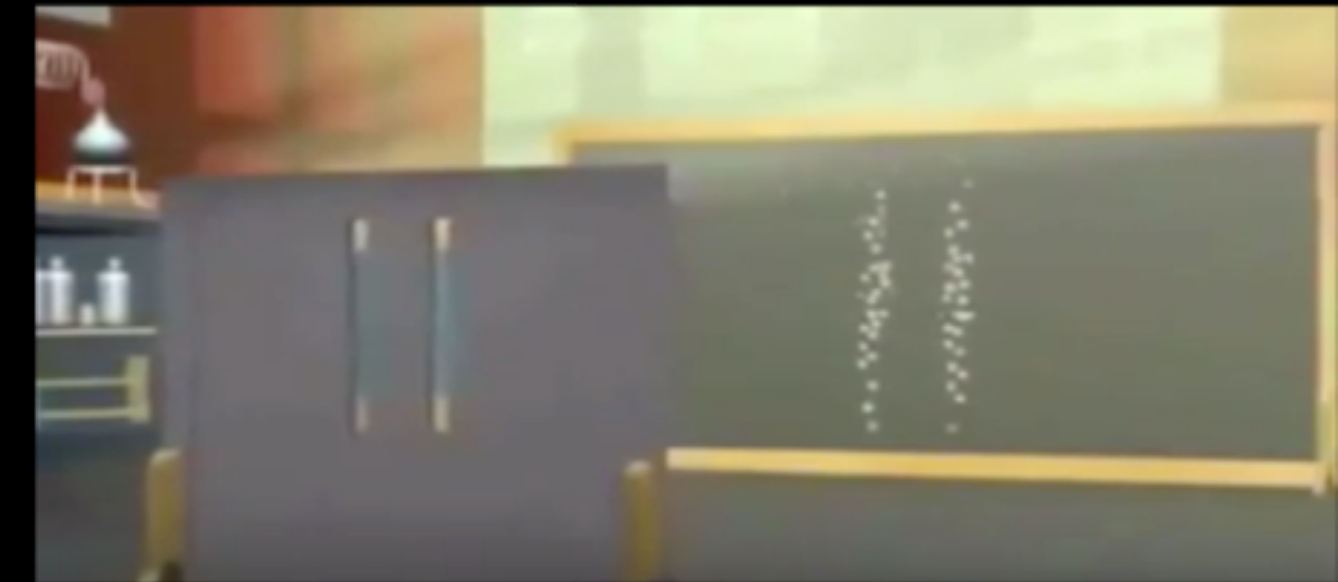


Einstein - De Broglie

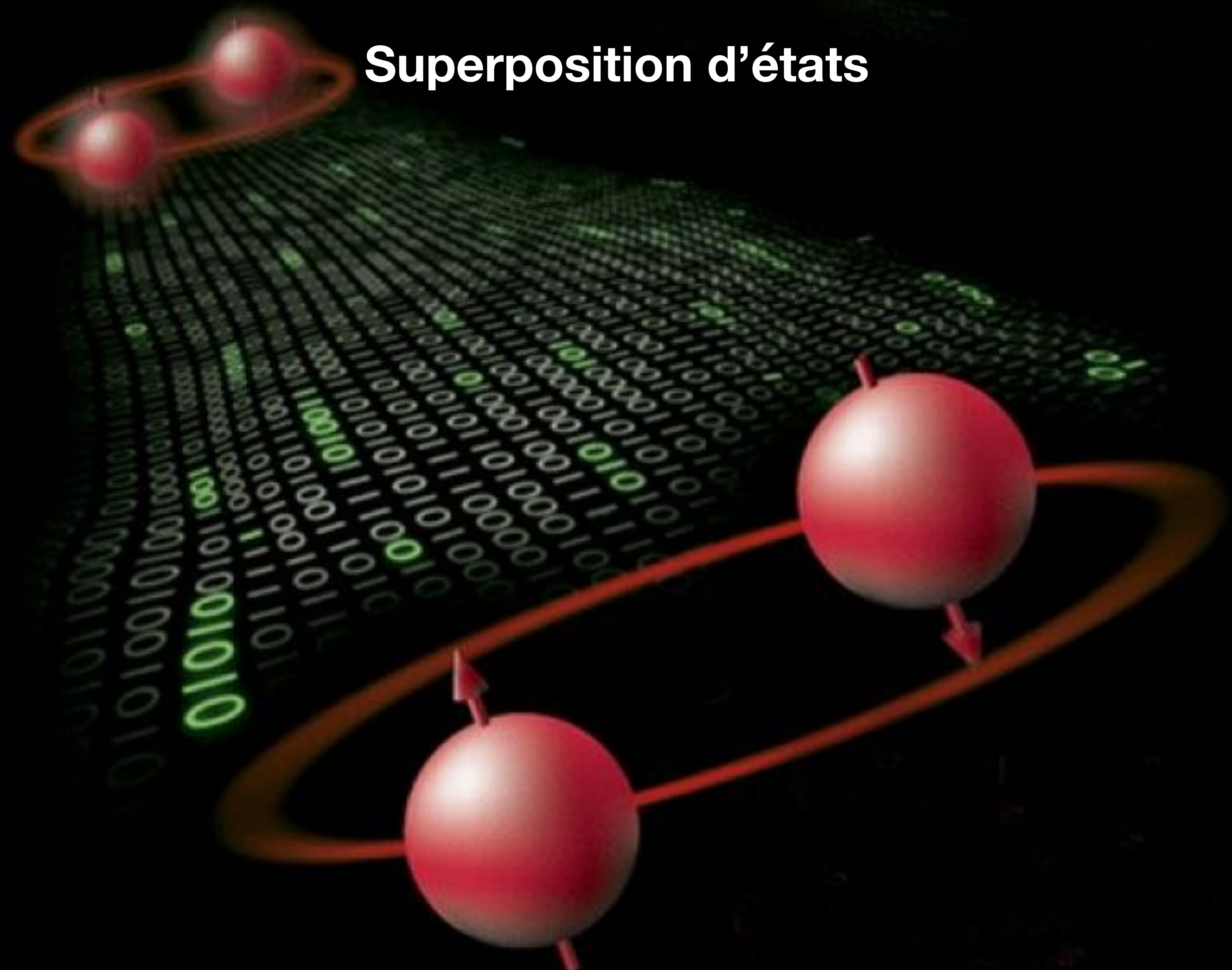
$$E = h\nu$$
$$p = \frac{h}{\lambda}$$



En forme de particule



Superposition d'états



**Notre chat de Schrödinger
est-il
mort ou vivant ?**

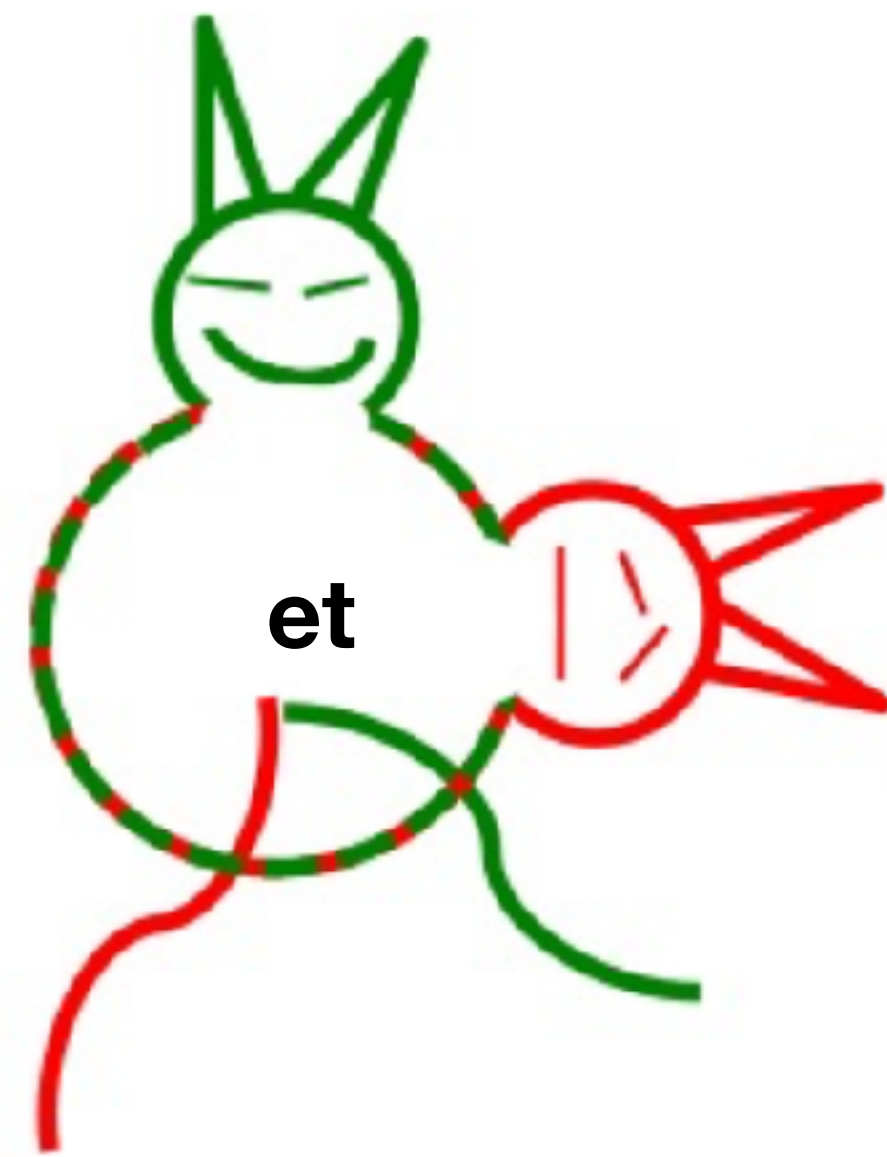


Décohérence

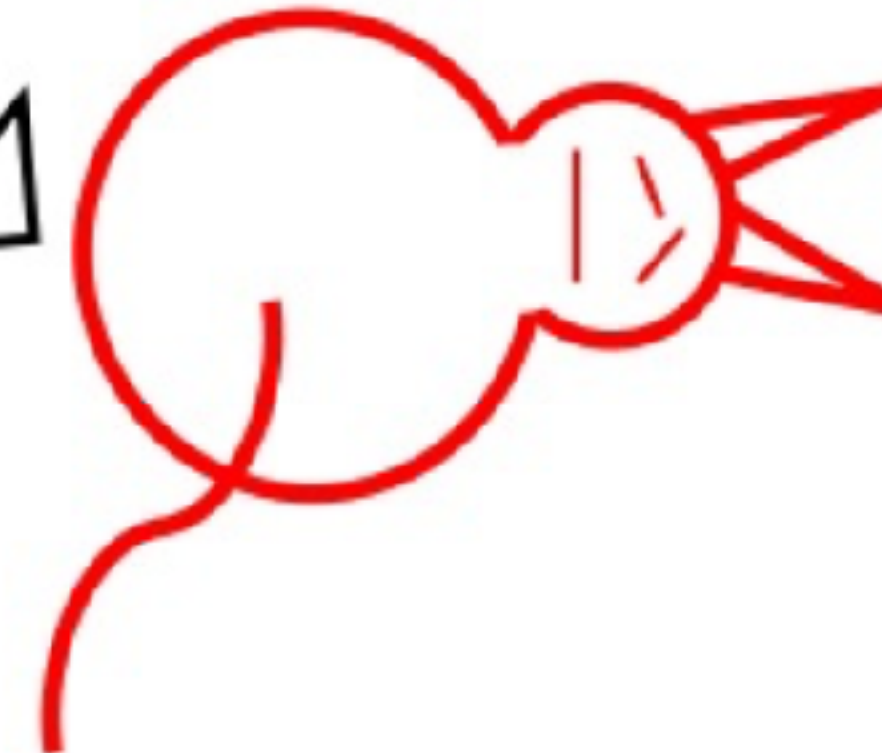
Echelle microscopique

Echelle macroscopique

Superposition
d'états quantiques



Décohérence



INTRICATION QUANTIQUE

An abstract visualization of quantum entanglement. The image features two complex, glowing waveforms on the left and right sides, connected by a central, narrower band. The waveforms are composed of many thin, overlapping lines that create a mesh-like appearance. The colors transition from a deep blue on the left to a bright yellow in the center, and back to a deep blue on the right. The background is a dark, gradient blue.

Principe de non localité

History of Quantum Computing

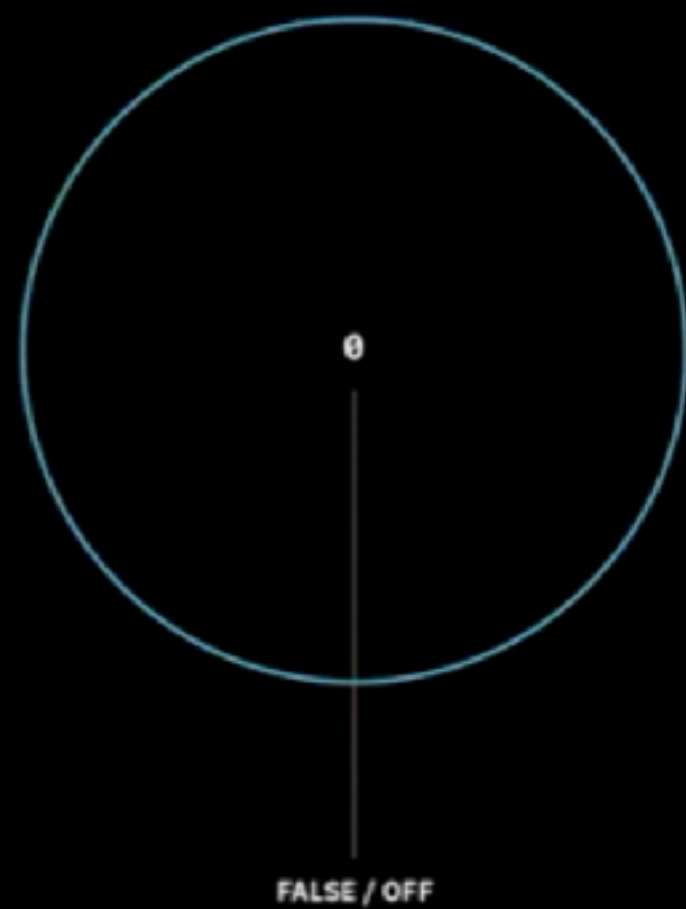
Year	Event
1930er	Discovery of Quantum Mechanics
1970	Note on “quantum information theory” from Charlie Bennett (today IBM Fellow) and Stephen Wiesner
1981	Richard Feynman suggests to build a quantum computer
1984	Quantum key distribution Bennett/Brassard
1993	Quantum Teleportation Bennett
1994	Shor’s 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



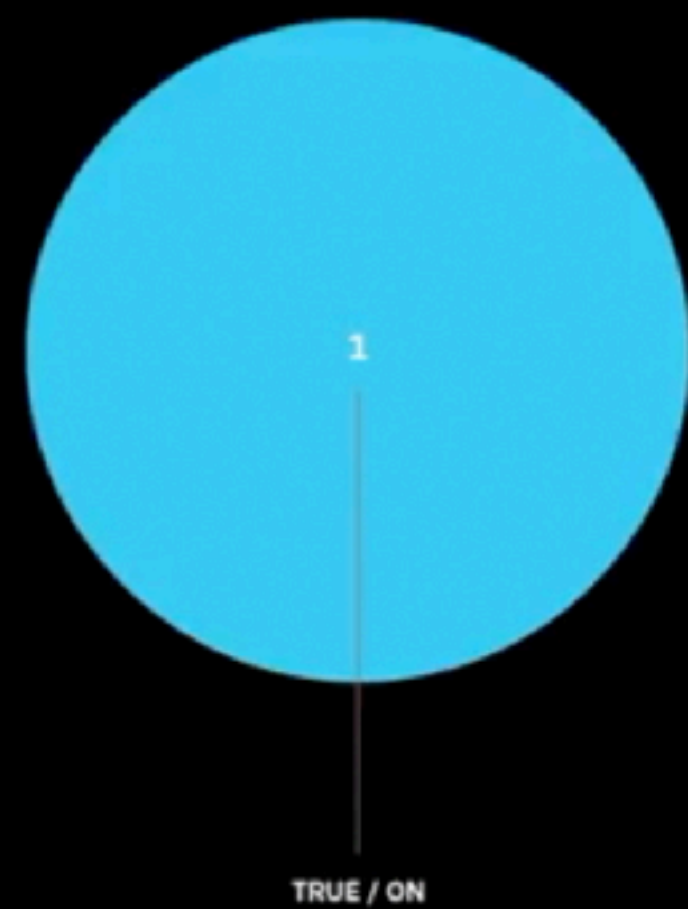
10 influential figures in the history of quantum mechanics. **Left to right:** [Max Planck](#), [Albert Einstein](#), [Niels Bohr](#), [Louis de Broglie](#), [Max Born](#), [Paul Dirac](#), [Werner Heisenberg](#), [Wolfgang Pauli](#), [Erwin Schrödinger](#), [Richard Feynman](#).

Superposition d'états

N qubits
 2^N paths

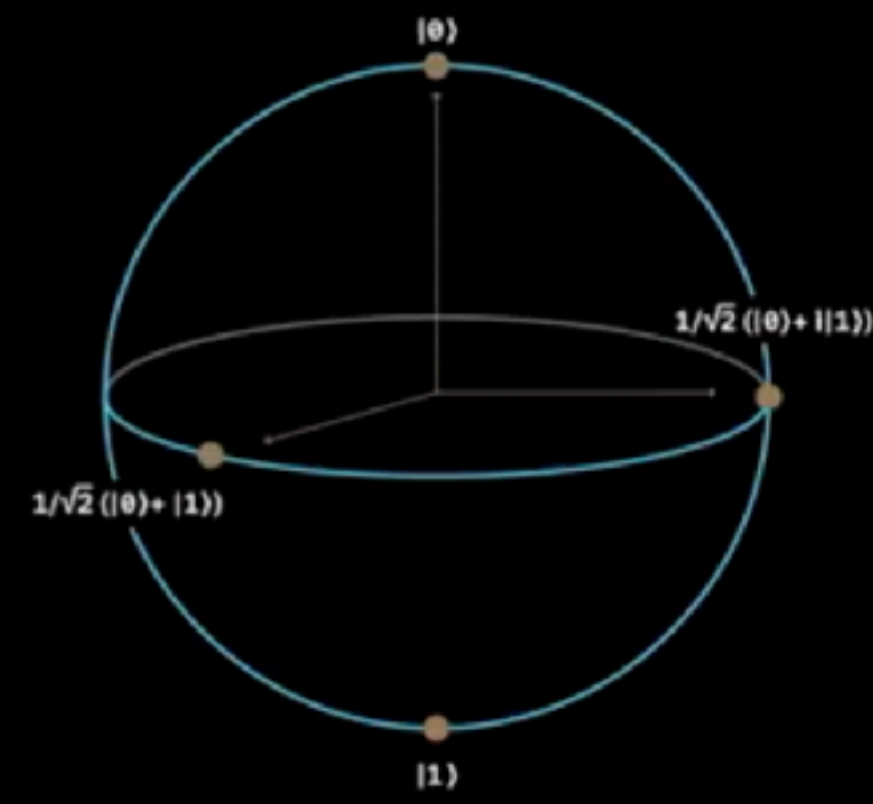


BITS

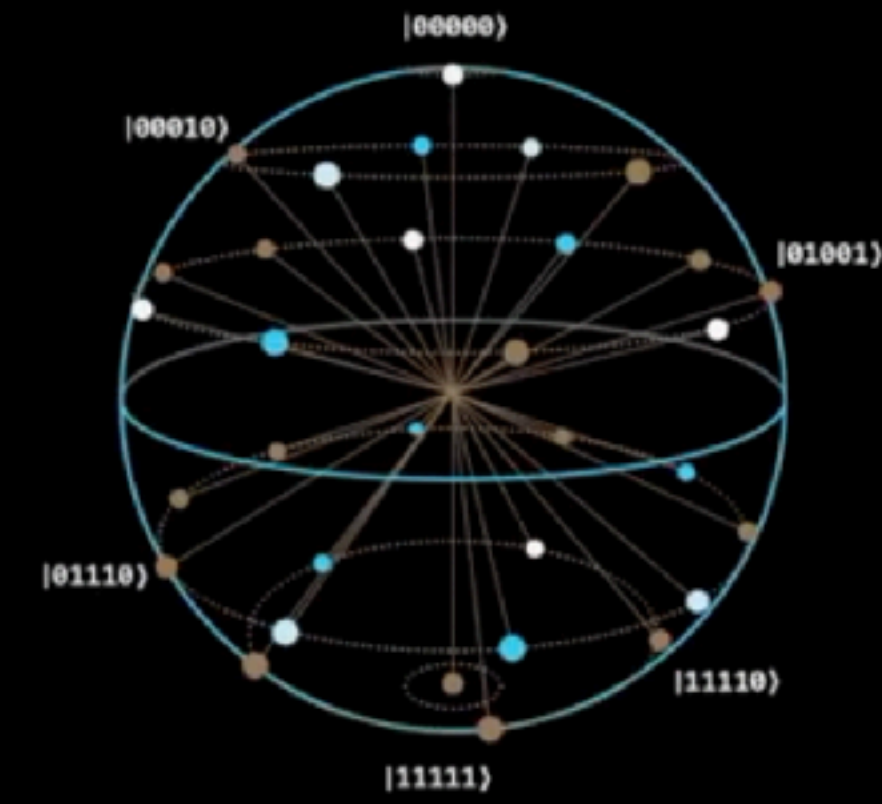


TRUE / ON

Etats classiques



BLOCH SPHERE (1 QUBIT)

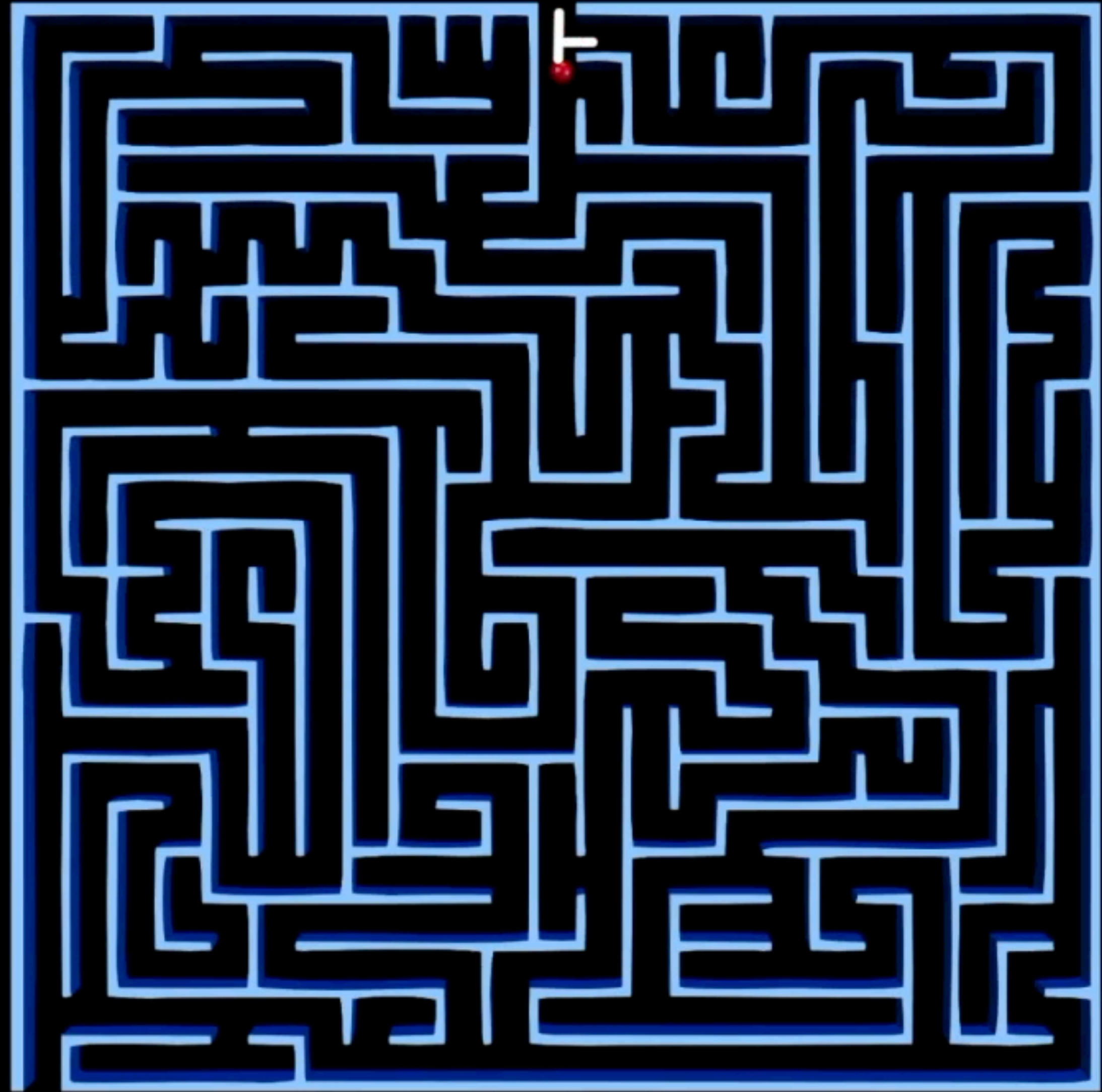


QSPHERE (5 QUBITS)

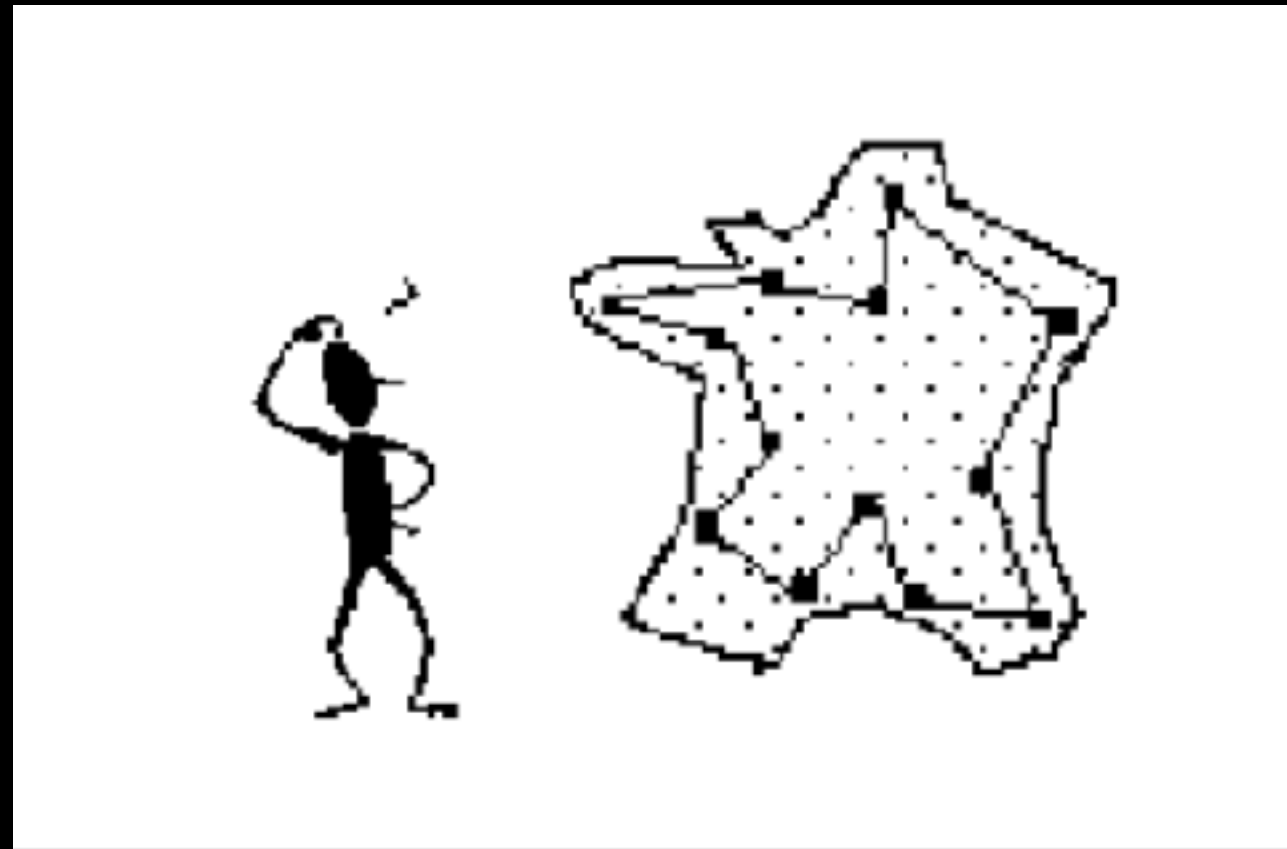
Etats quantiques

- Un bit: 0, 1
- Un qubit: 0, 1, ou les deux à la fois

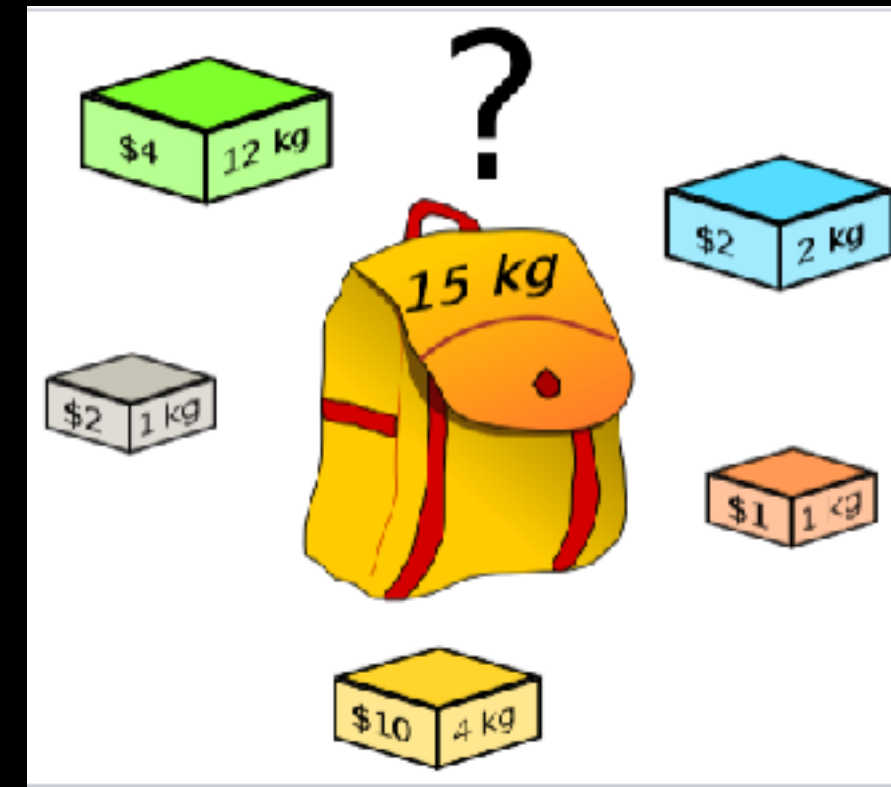
The diagram shows two parts. On the left, labeled 'UN BIT', there are two horizontal lines with red arrows pointing up and down, representing the discrete states 0 and 1. On the right, labeled 'UN QUBIT', there are two overlapping circles with red arrows pointing in various directions, representing the continuous space of quantum states.



Classe de problèmes NP



**Problème
du voyageur de commerce**



**Problème
du sac à dos**

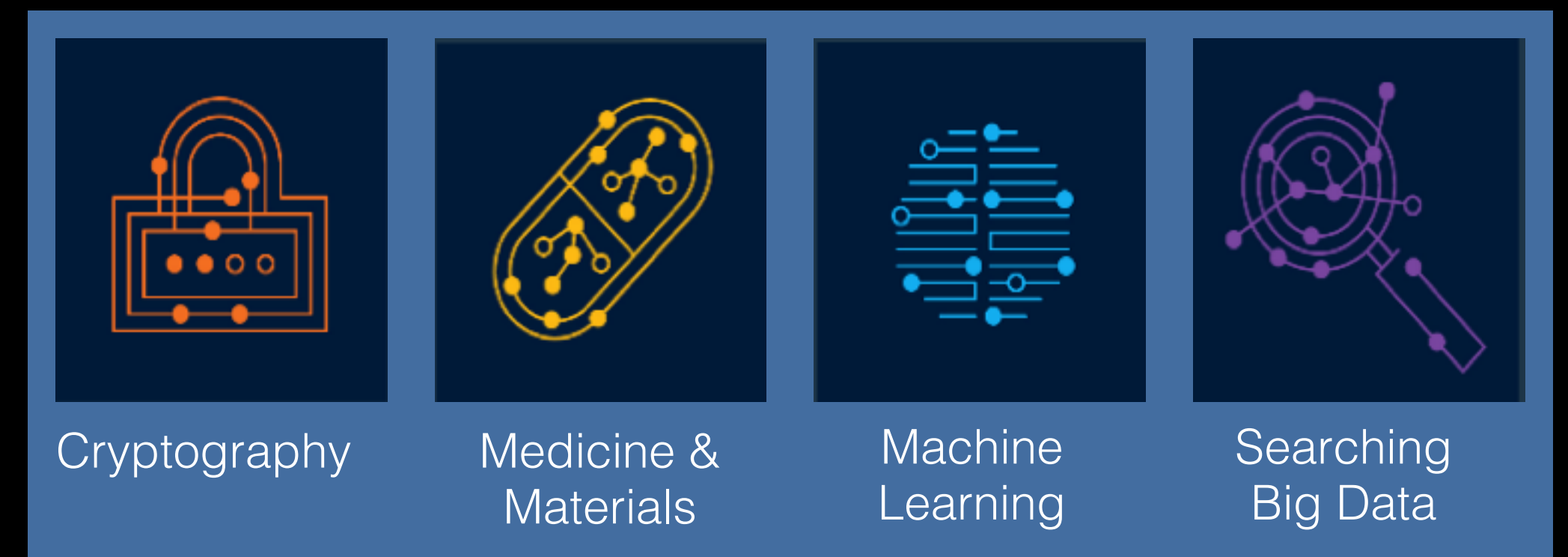
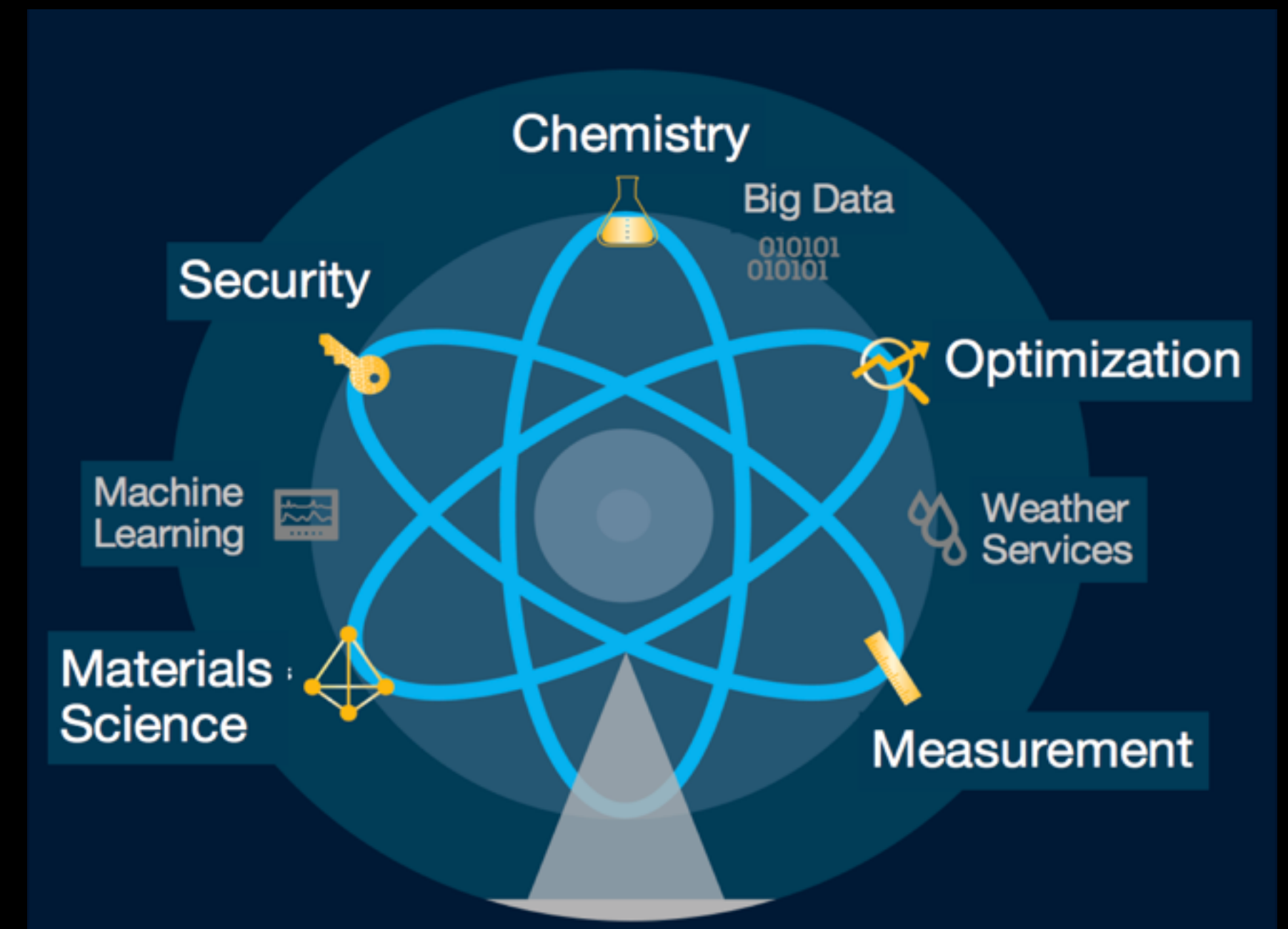
	7	9	6	3	5	
2			3	4	9	
1	3		5	7	8	
3	1	6	2	8		
		3	1		9	5
8	9	5			1	3
5	8		7	4		
	2	4	6	5	3	
			5		7	9

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, ...)



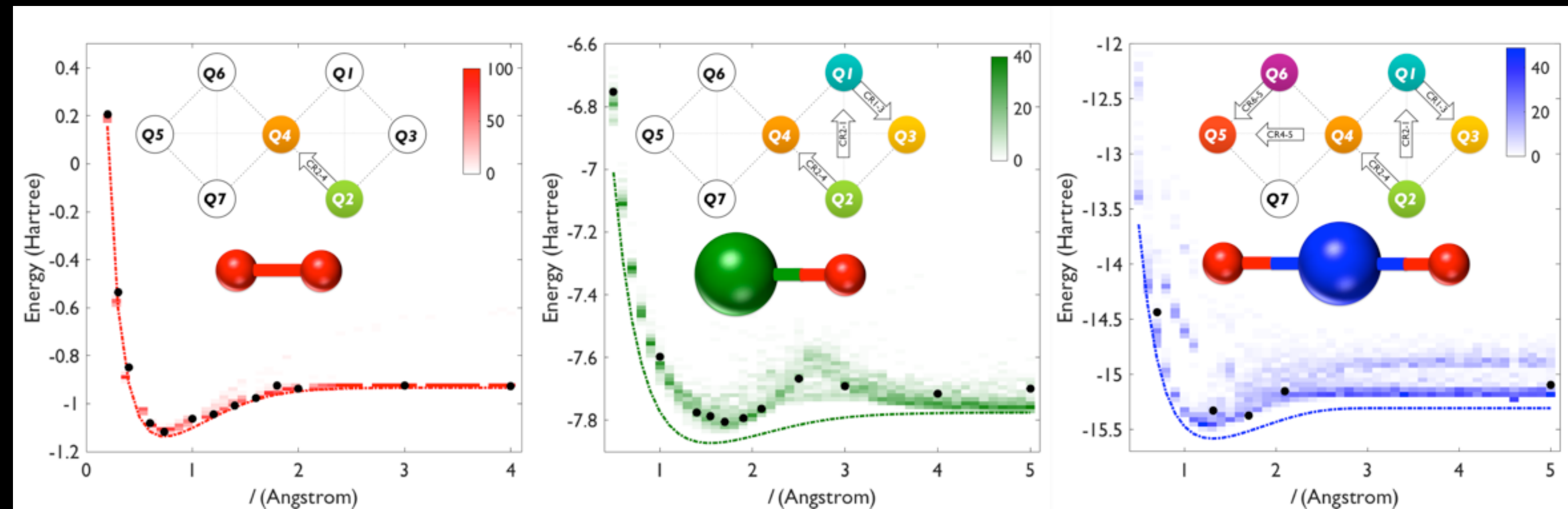


World's first demonstration of modeling LiH Lithium hydride (LiH) and Beryllium hydride (BeH₂) using a quantum computer.

2 qubits

4 qubits

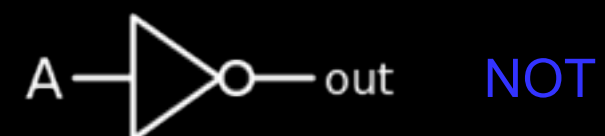
6 qubits



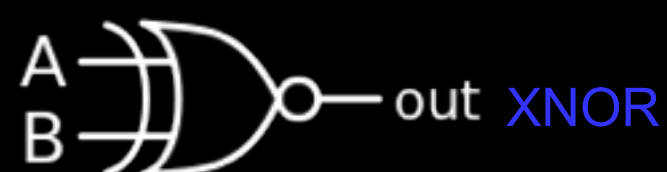
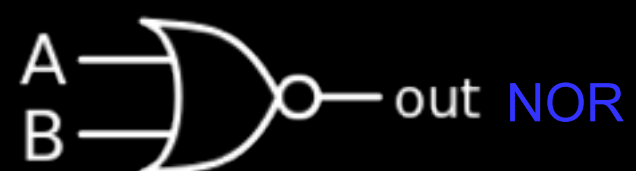
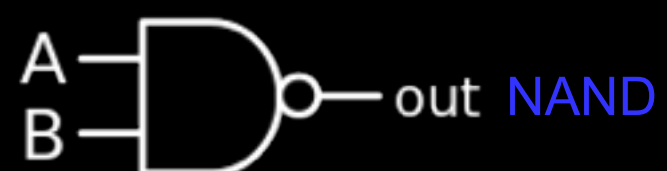
N electrons ~ N qubits

Bit

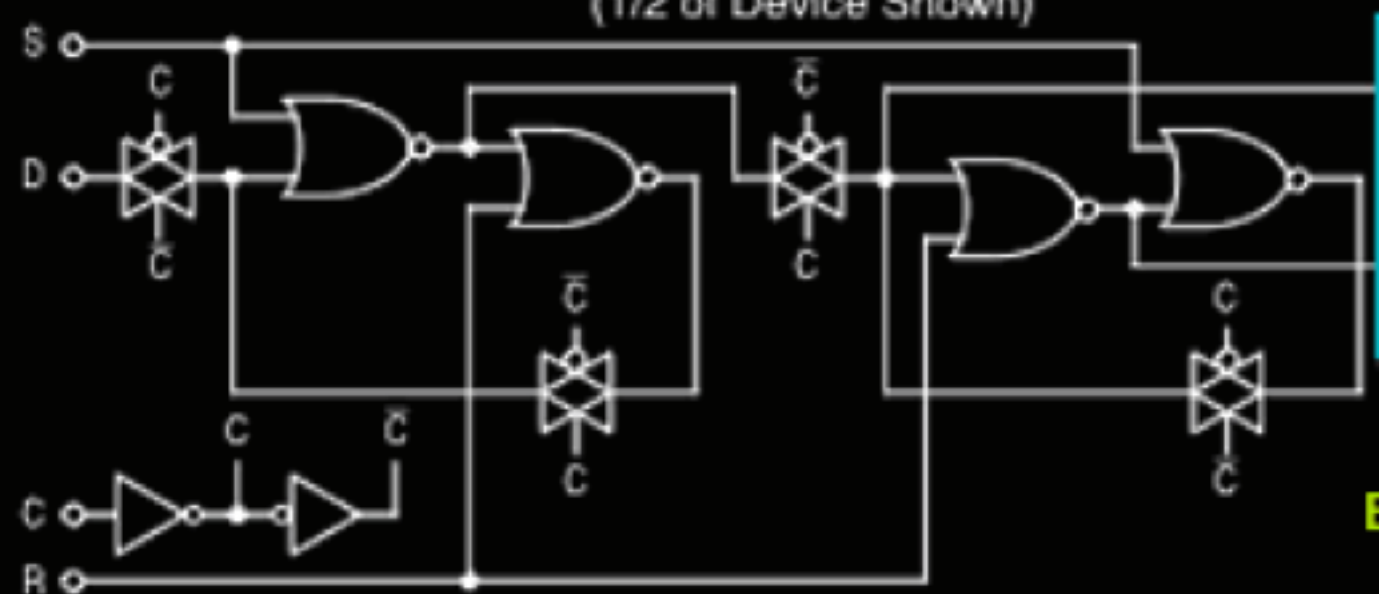
1 Bit Gates



2 Bit Gates



LOGIC DIAGRAM
(1/2 of Device Shown)



Qubit

IN

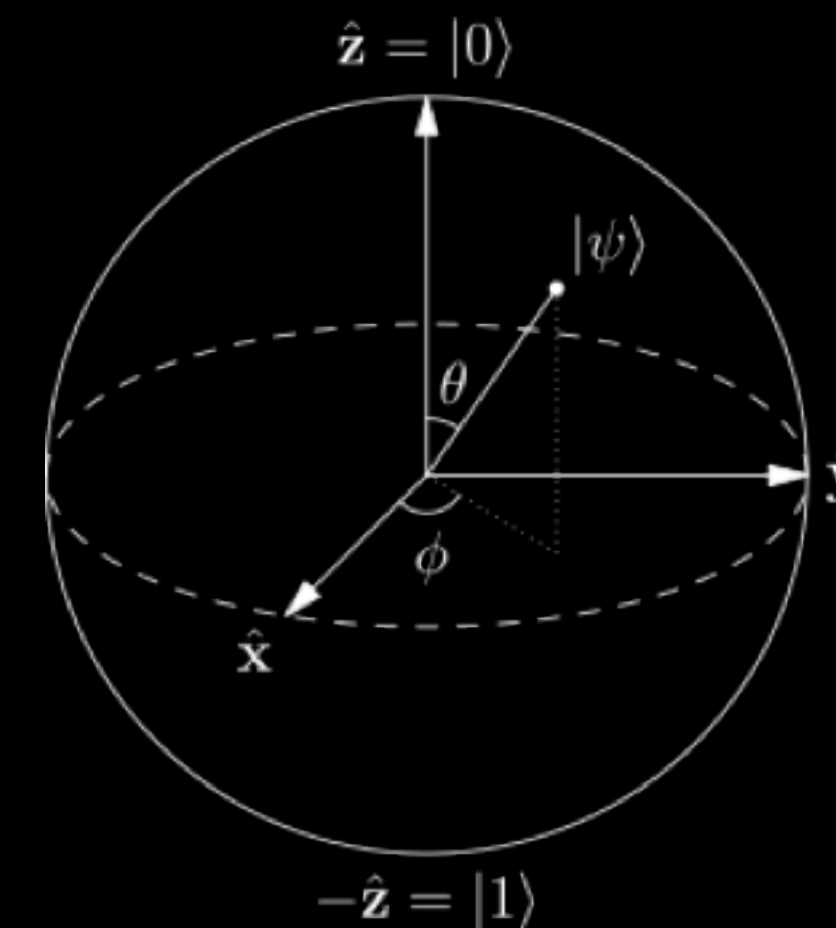


OUT

1-qubit Gate

→ Rotation of Bloch-Sphere

→ Adding to length and latitude



2-qubit Gate: CNOT

→ Controlled NOT

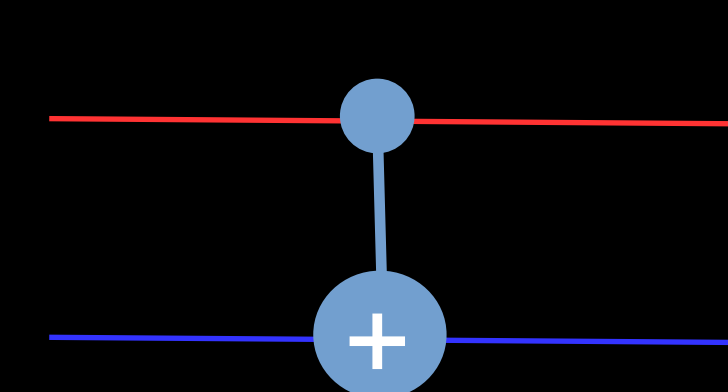
→ Exchange of |10> and |11>

|00>

|01>

|10>

|11>

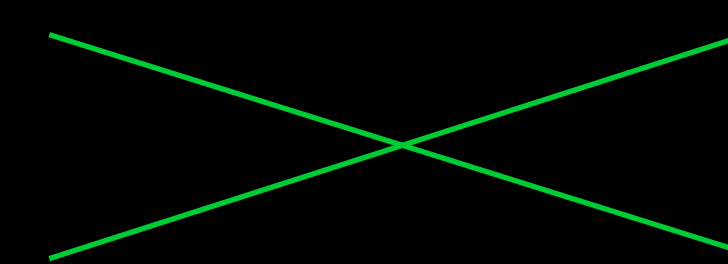


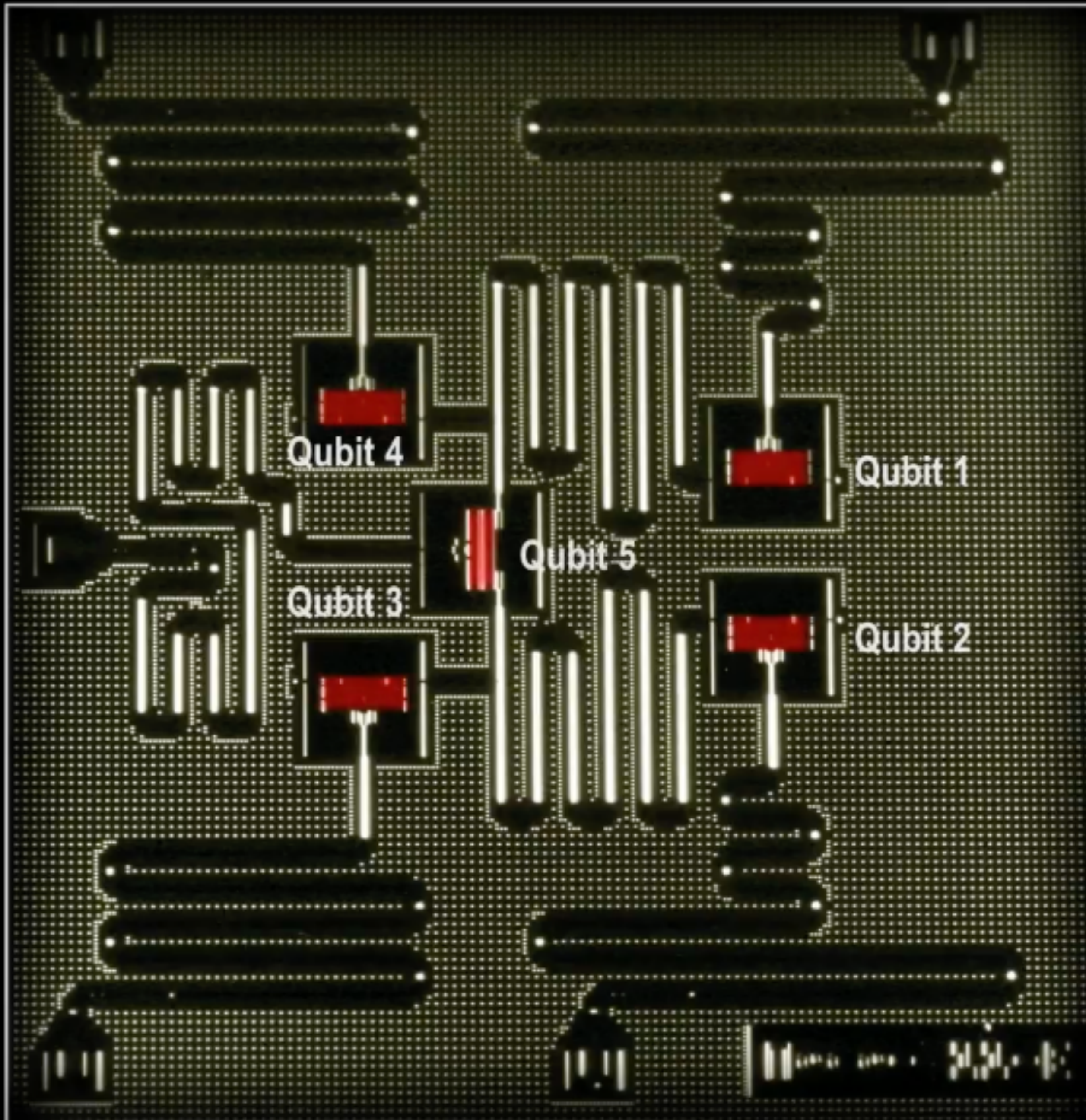
|00>

|01>

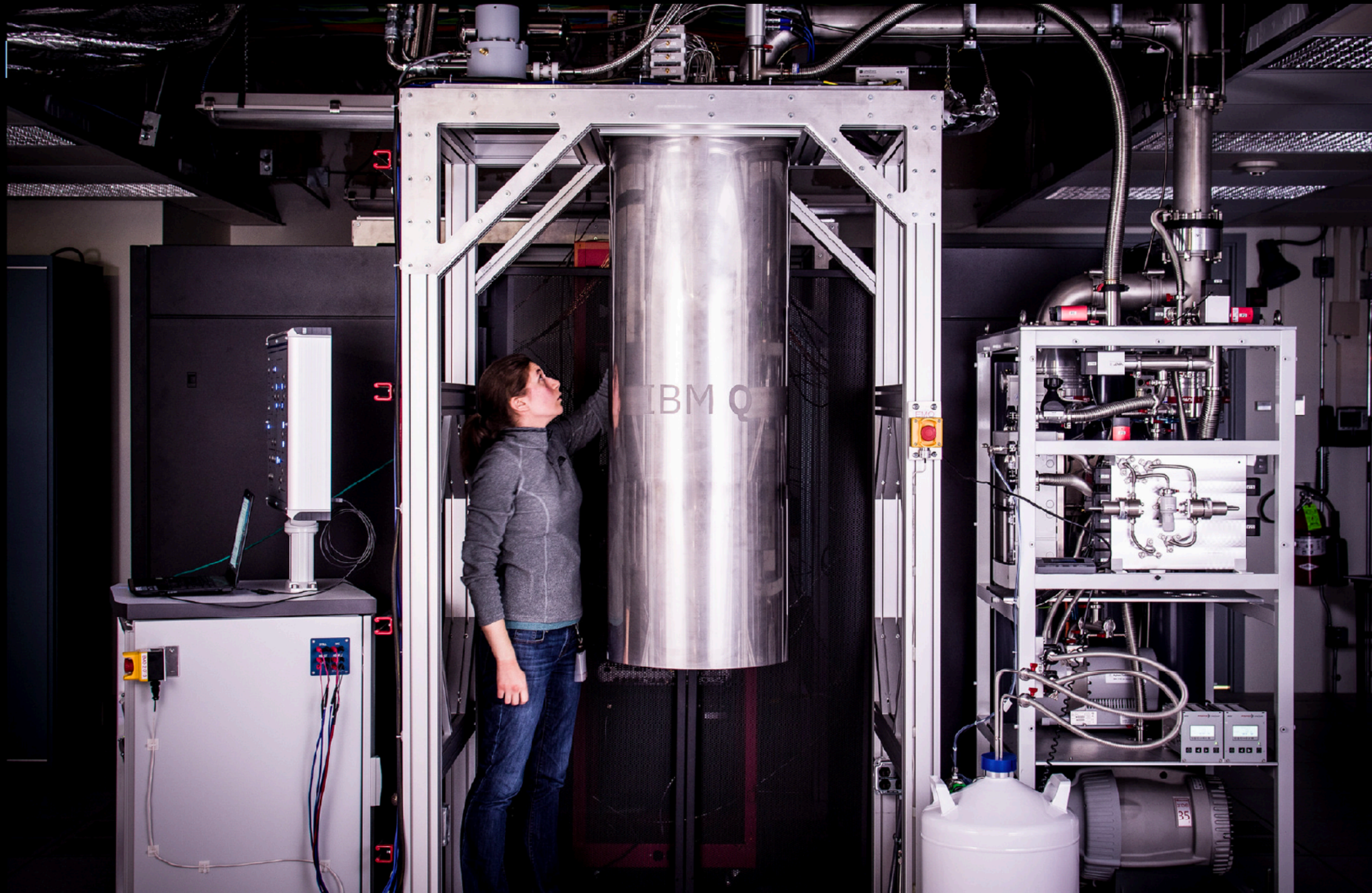
|10>

|11>





I

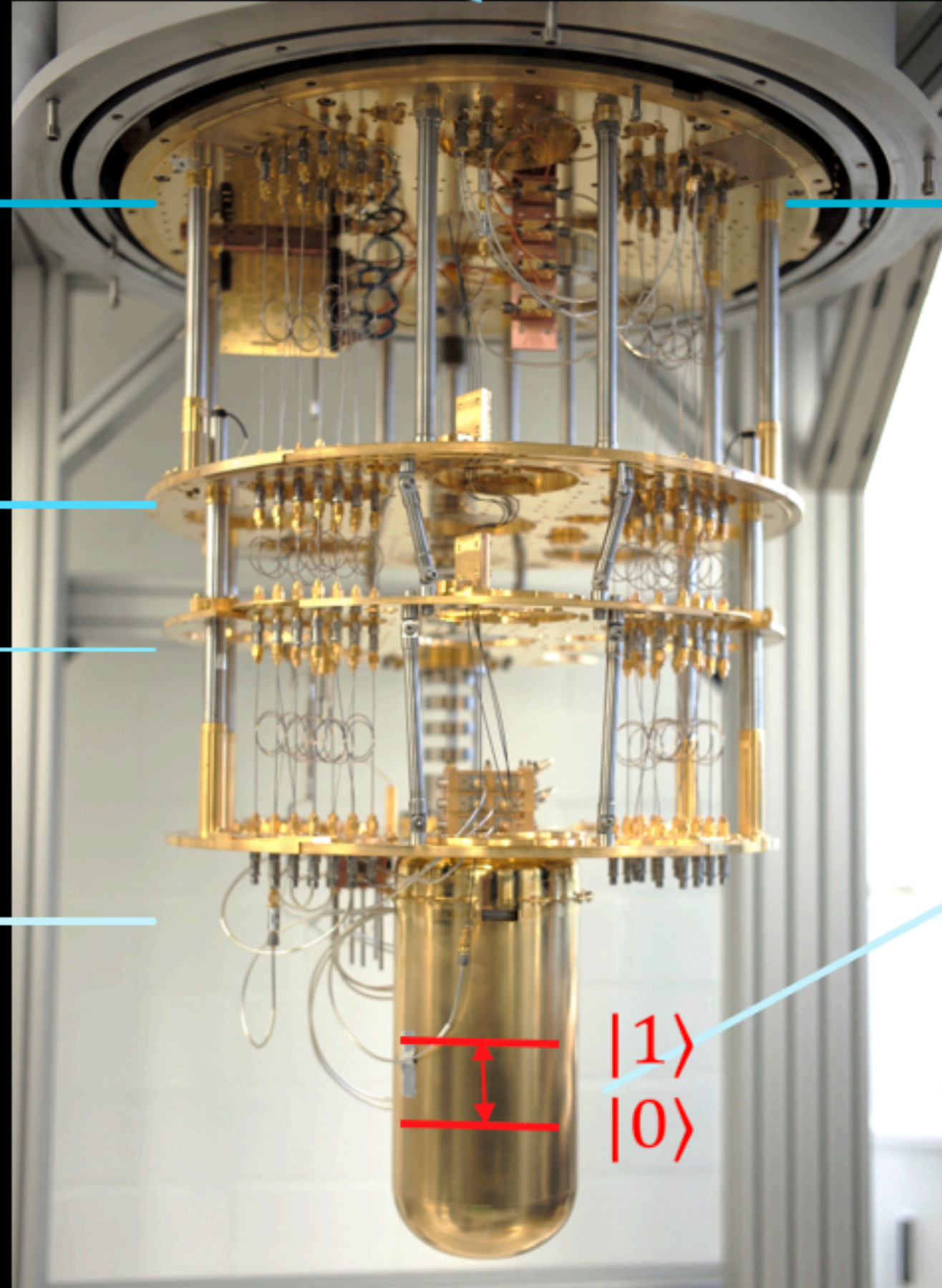


IBM



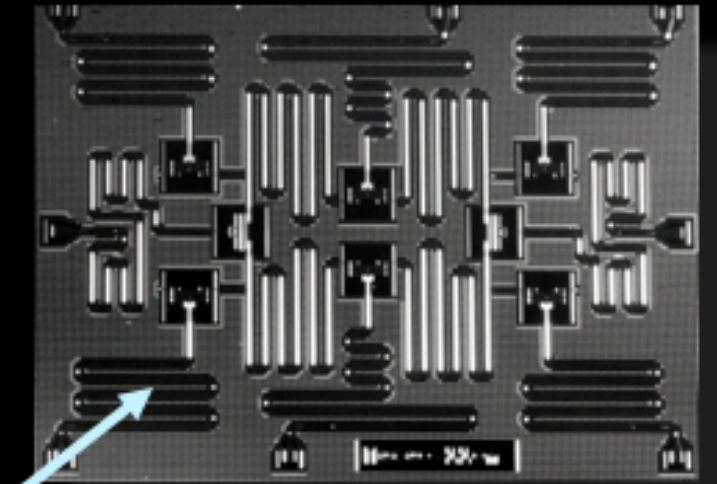
Microwave electronics

2.7K
0.8K
0.1K
0.02K

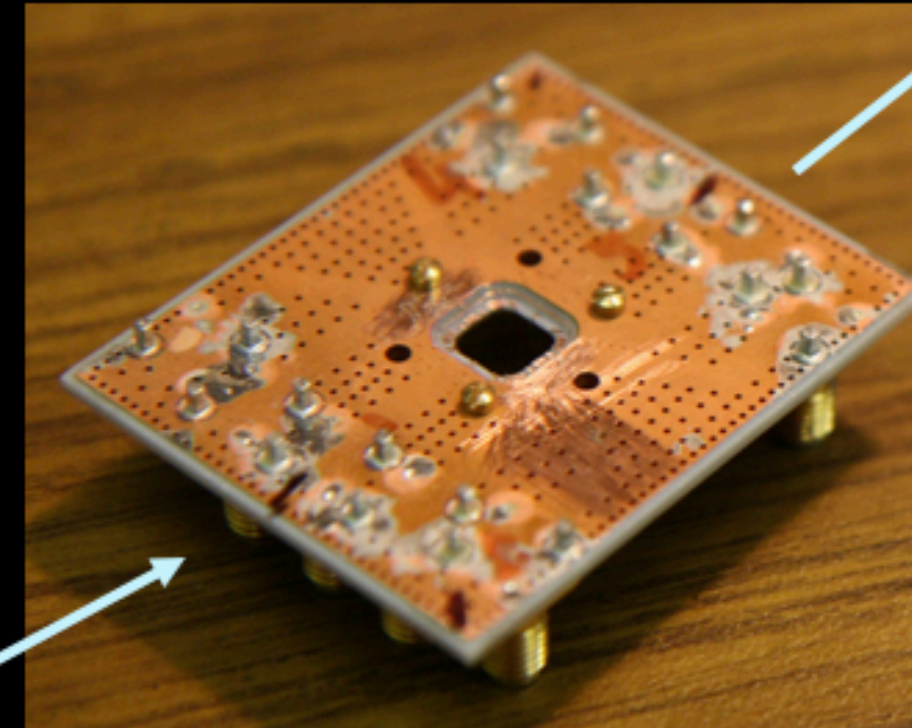


Dilution cryostat

-270°C



Chip with superconducting qubits and resonators



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



Quantum Volume is a metric of computational value...

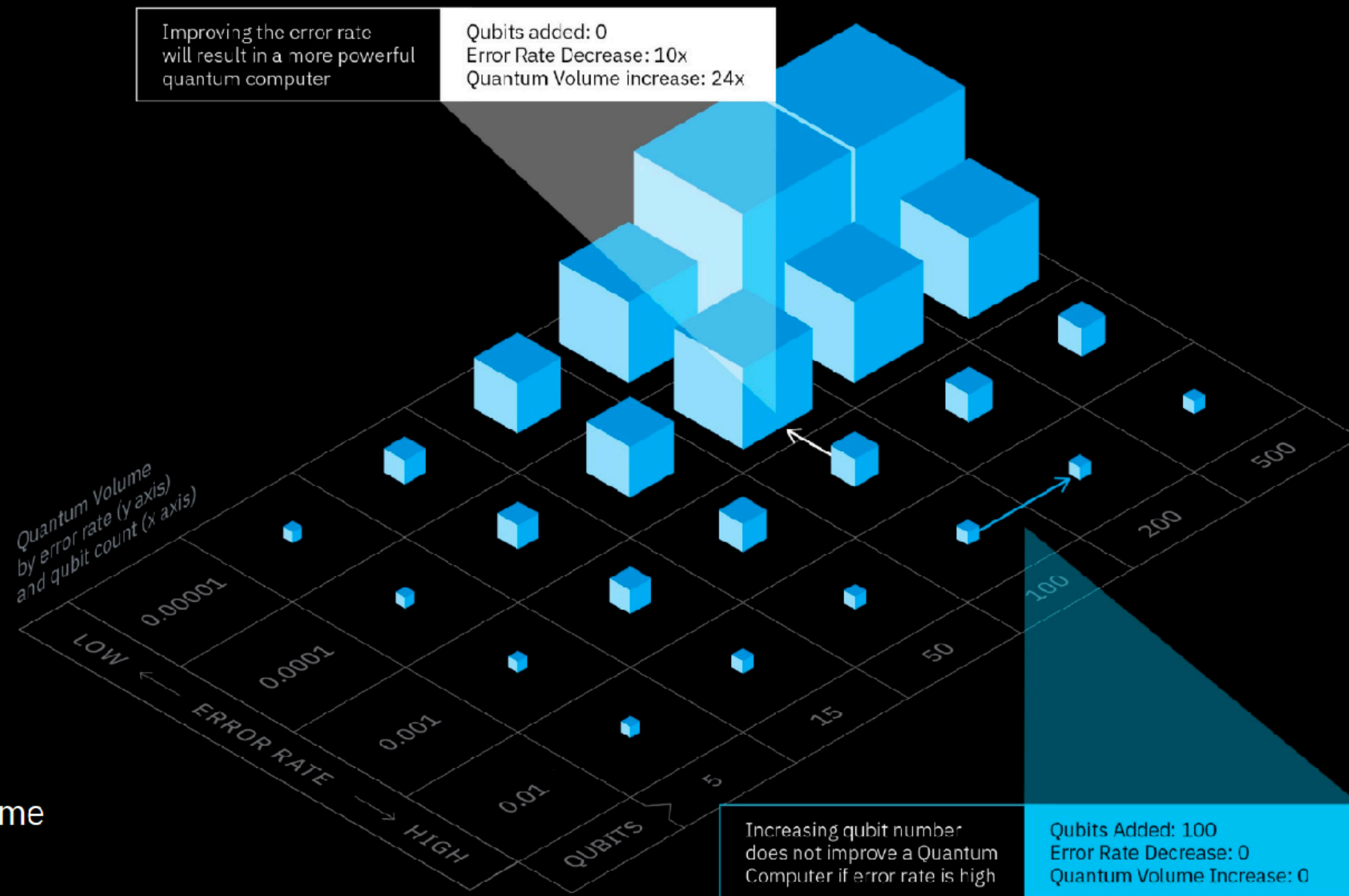
Qubit Count
(more is better)

Errors
(less is better)

Connectivity
(more is better)

Gate set
(more is better)

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



80,000+ users

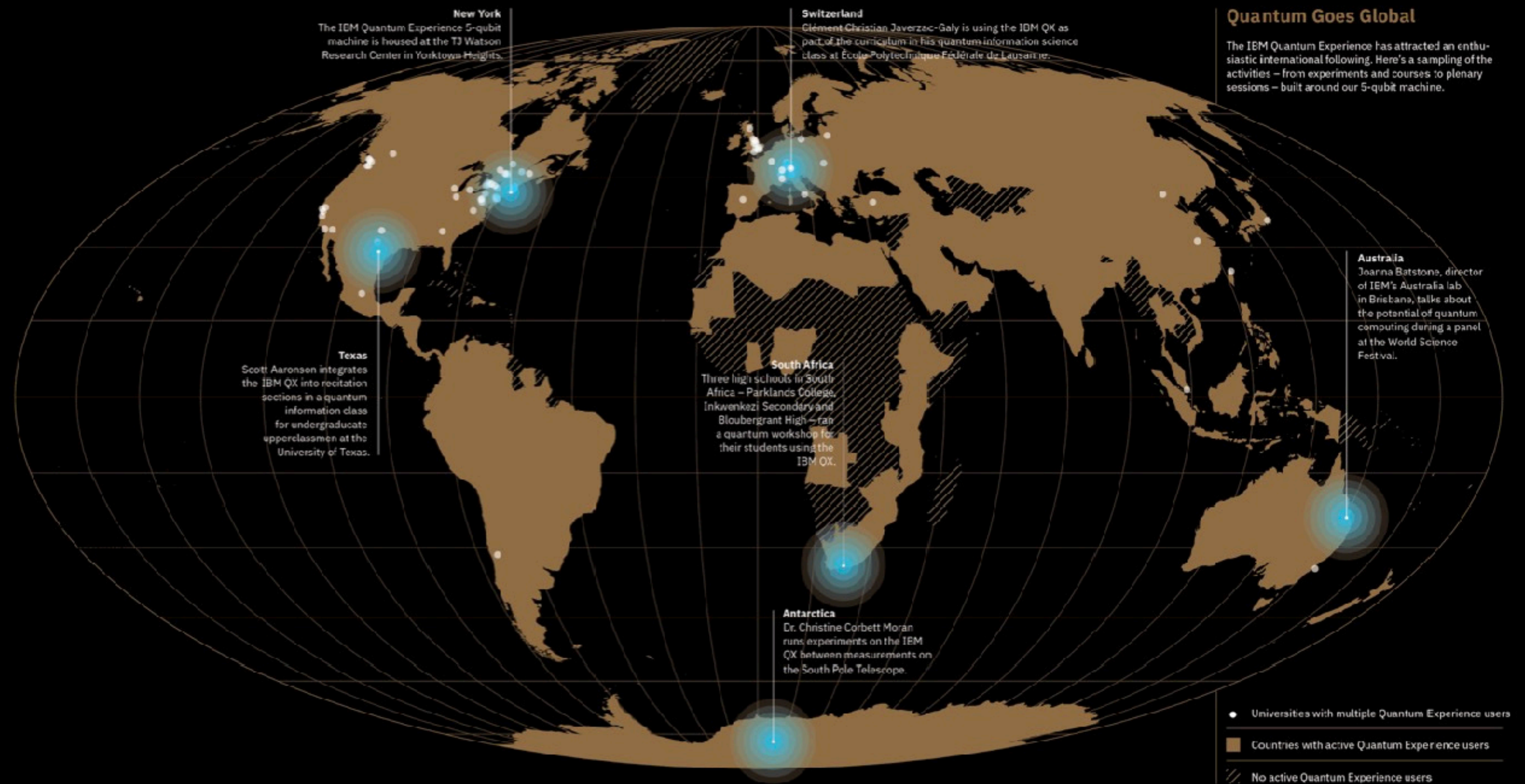
All 7 continents

> 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 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

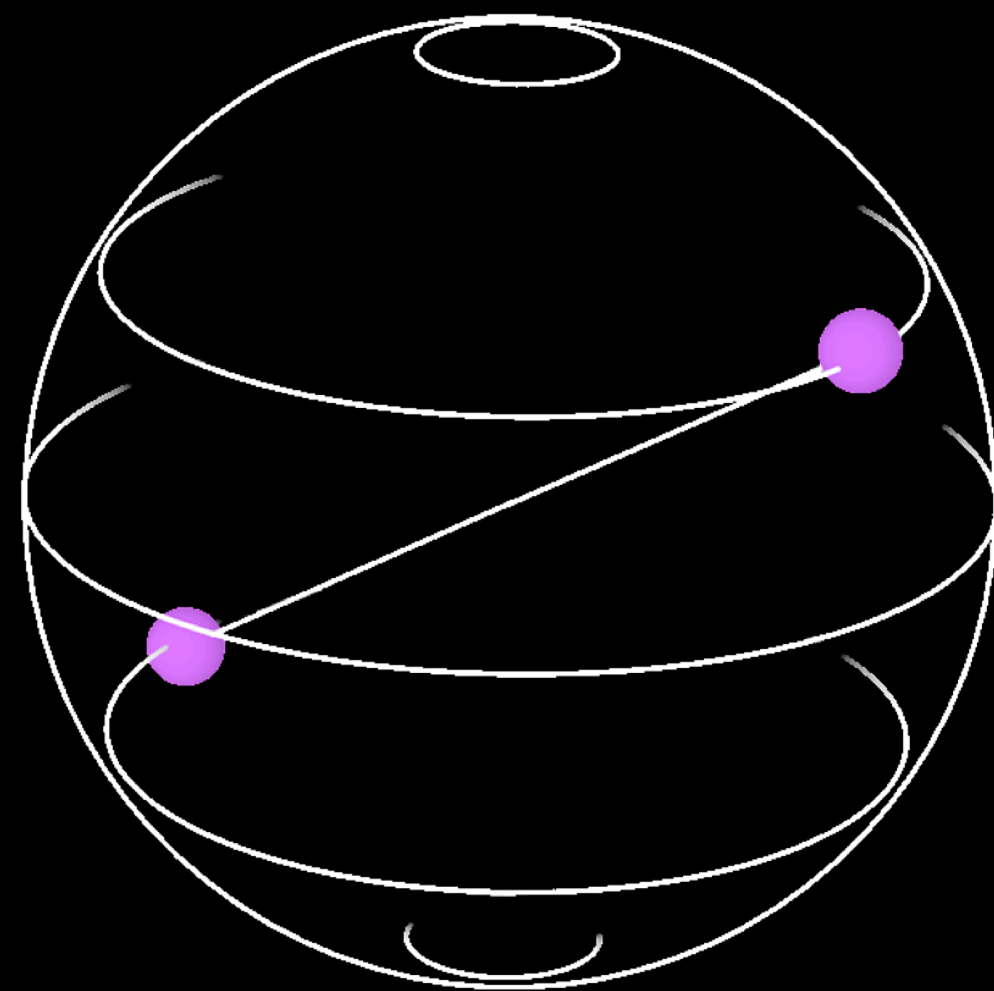
Enabled with a Quantum Software Stack

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

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

IBM Q

QISKit



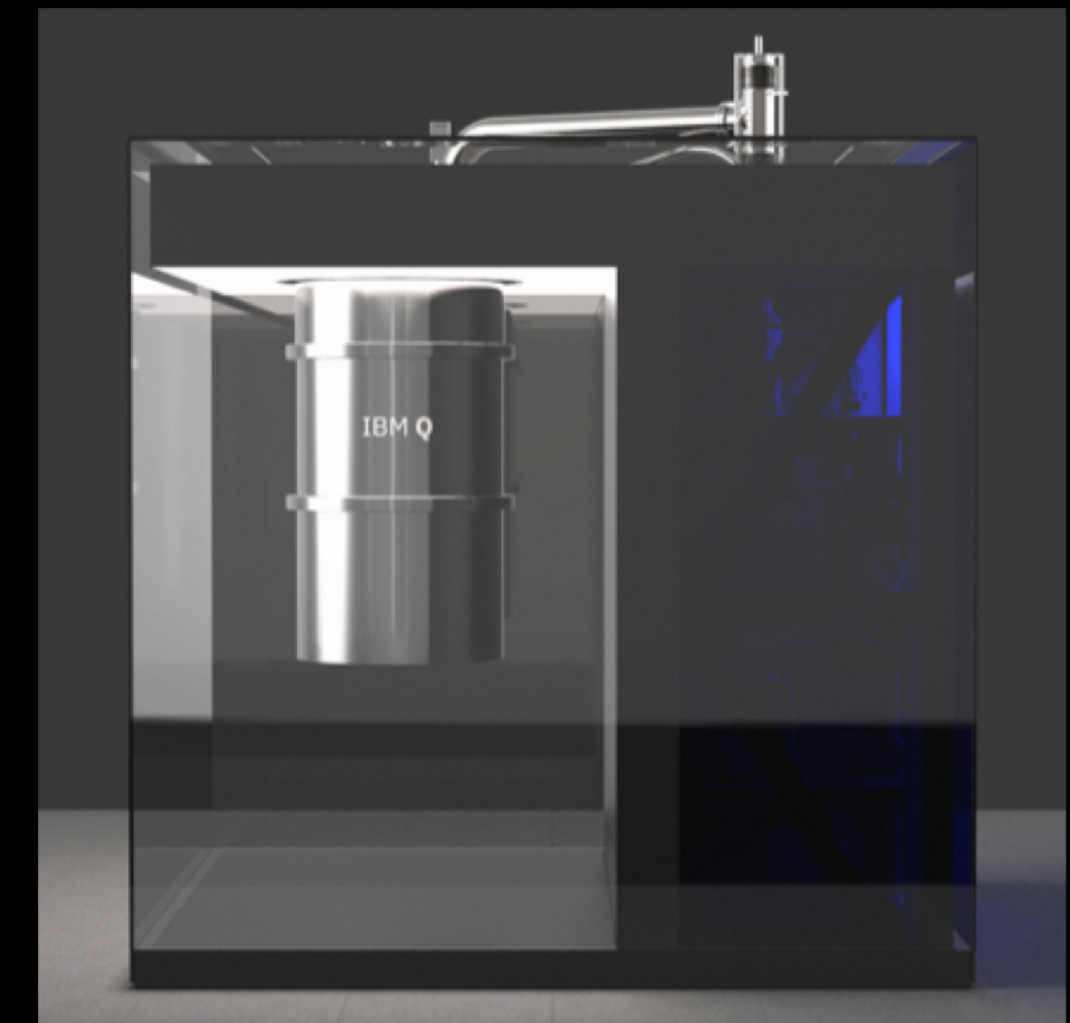
Open SDK and libraries at qiskit.org

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)

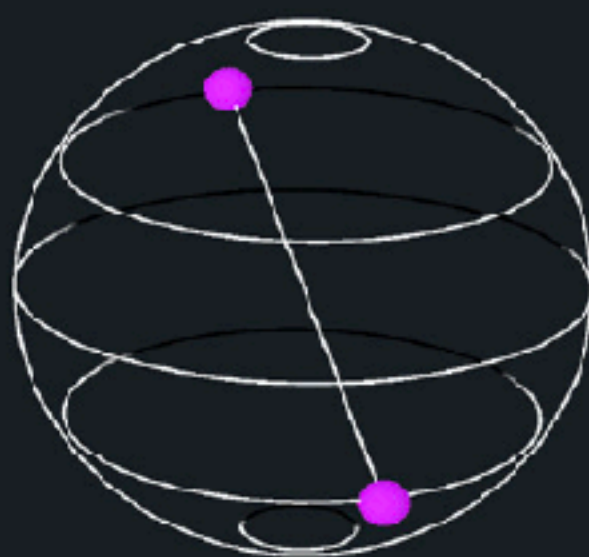
<https://www.qiskit.org>

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>

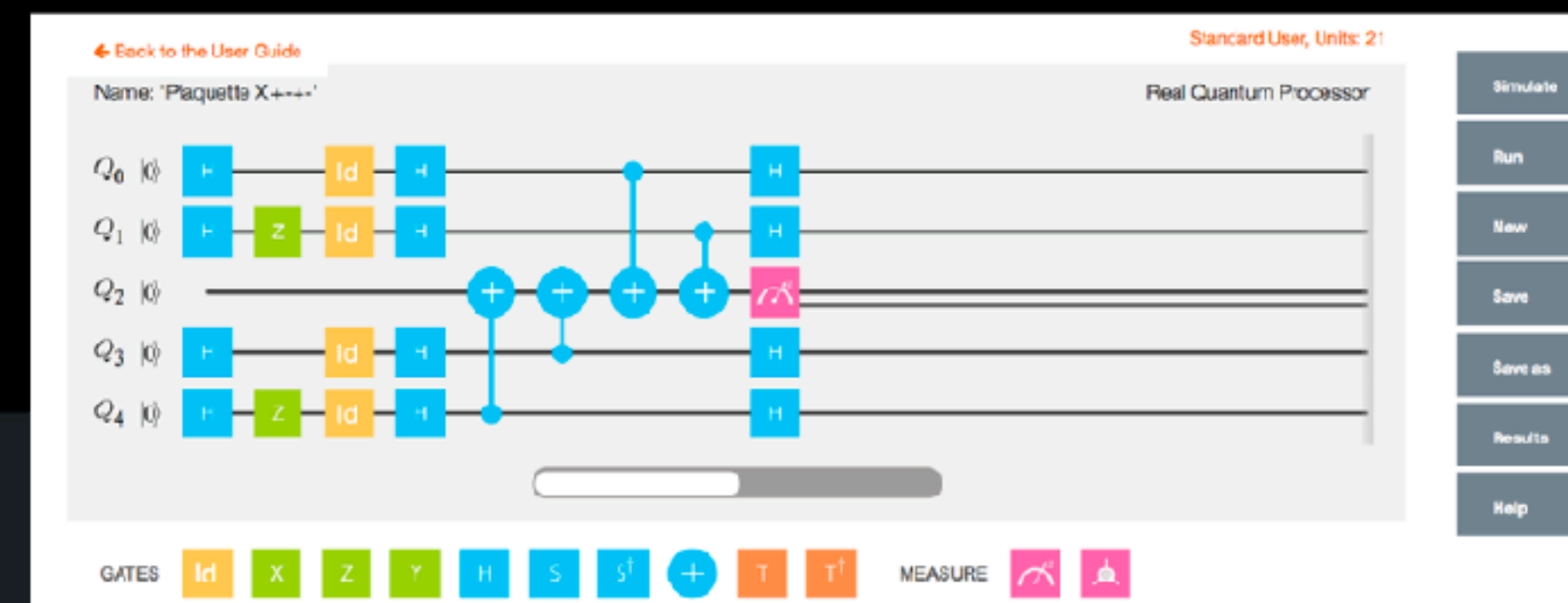
IBM QX Features

- 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](#)

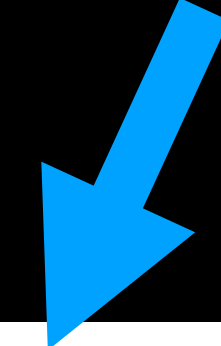
Run a quantum program

```
[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/>



IBM Marketplace

IBM Q IBM Q Network Learn Experiment GitHub

IBM Q is an industry-first initiative to build commercially available universal quantum computers for business and science.

Watch video (04:46)

Contact an expert

Get started Partner with IBM Q Try quantum **Develop with QISKit**

Création d'un compte IBM Q

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

The screenshot shows the IBM Q Experience website homepage. At the top, there is a navigation bar with the text "IBM Q > Experience" on the left and links for "Home", "Composer", "Devices", "Community", and "GitHub" on the right. The "Home" link is underlined. Below the navigation bar, the main heading reads "Welcome to the IBM Q Experience!". Underneath this heading is a paragraph of text: "Explore the world of quantum computing! Check out our User Guides and interactive Demos to learn more about quantum principles. Or, dive right in to create and run algorithms on real quantum computing hardware, using the Quantum Composer and QISKit software developer kit." A blue arrow points from the text "using the Quantum Composer and QISKit software developer kit." down to a button that says "Start experimenting with a quantum computer". To the right of the text is an image of a laptop displaying a quantum circuit diagram. Below the main content area, there are three columns of promotional text. The first column is titled "Introducing the IBM Q Experience for Researchers". The second column is titled "Visibility for your papers" and has a right-pointing arrow below it. The third column is titled "Priority and early access to device" and also has a right-pointing arrow below it. In the bottom right corner, there is a dark grey sidebar with three icons: a quantum circuit icon, a search icon, and a "light" theme toggle icon.

IBM Q > Experience

Home Composer Devices Community GitHub

Welcome to the IBM Q Experience!

Explore the world of quantum computing! Check out our User Guides and interactive Demos to learn more about quantum principles. Or, dive right in to create and run algorithms on real quantum computing hardware, using the Quantum Composer and QISKit software developer kit.

[Start experimenting with a quantum computer](#)

Introducing the IBM Q Experience for Researchers

Visibility for your papers

Priority and early access to device

light

Name: 'Experiment #20170521093507' New Save Save as

ibmqx2

Run Simulate

Gates Properties QASM My Units

GATES Advanced

Quantum gates palette:

- id (yellow)
- X (green)
- Y (green)
- Z (green)
- H (blue)
- S (cyan)
- S† (cyan)
- + (blue circle)
- T (red)
- T† (red)

BARRIER

Barrier icon: a vertical dashed line.

OPERATIONS

Operations icon: a pink square with a white circle and arrow.

Quantum gates

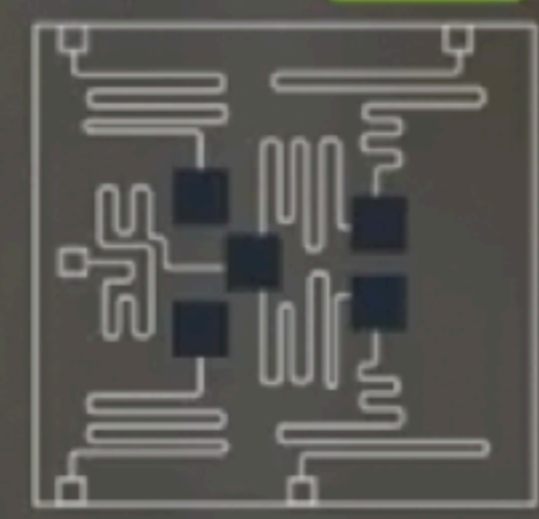


Add a description

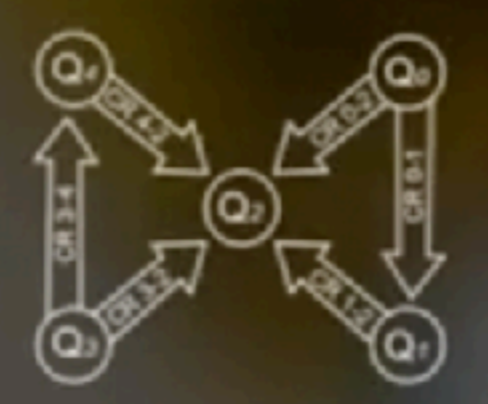
Switch to Qasm Editor

5-qubits
Each a diff. freq.
Good coherence
Low error rates

ibmqx2 **ACTIVE**



Fridge Temperature
0.0162 Kelvin



	Q0	Q1	Q2	Q3	Q4
CR0_1 e_g^{01} : 8.75×10^{-2}	f : 5.27 GHz	f : 5.21 GHz	f : 5.03 GHz	f : 5.30 GHz	f : 5.06 GHz
CR0_2 e_g^{02} : 8.47×10^{-2}	T_1 : 43.9 μ s	T_1 : 69.7 μ s	T_1 : 46.1 μ s	T_1 : 47 μ s	T_1 : 68.7 μ s
CR1_2 e_g^{12} : 4.65×10^{-2}	T_2 : 20 μ s	T_2 : 30.6 μ s	T_2 : 59.8 μ s	T_2 : 54.9 μ s	T_2 : 121.4 μ s
CR3_2 e_g^{32} : 5.04×10^{-2}	e_g : 7×10^{-3}	e_g : 2.5×10^{-3}	e_g : 6×10^{-3}	e_g : 3.3×10^{-3}	e_g : 2.5×10^{-3}
CR3_4 e_g^{34} : 3.56×10^{-2}	e_r : 4×10^{-2}	e_r : 8.4×10^{-2}	e_r : 2.4×10^{-2}	e_r : 1.8×10^{-2}	e_r : 5.5×10^{-2}
CR4_2 e_g^{42} : 4.38×10^{-2}					

Expérience IBM Q

IBM Q > Experience

Home Composer Devices Community GitHub

> Backend: ibmqx4 (5 Qubits) MAINTENANCE

> Backend: ibmqx2 (5 Qubits) MAINTENANCE

New experiment

[New](#) [Save](#) [Save as](#)

[Switch to Qasm Editor](#) Backend: ibmqx4 i My Units: 72 i Experiment Units: 3 i [Run](#) [Simulate](#)

q[0] 0) _____

q[1] 0) _____

q[2] 0) _____

q[3] 0) _____

q[4] 0) _____

c 0⁵ / _____

GATES i Advanced

U1 U2 U3 id X

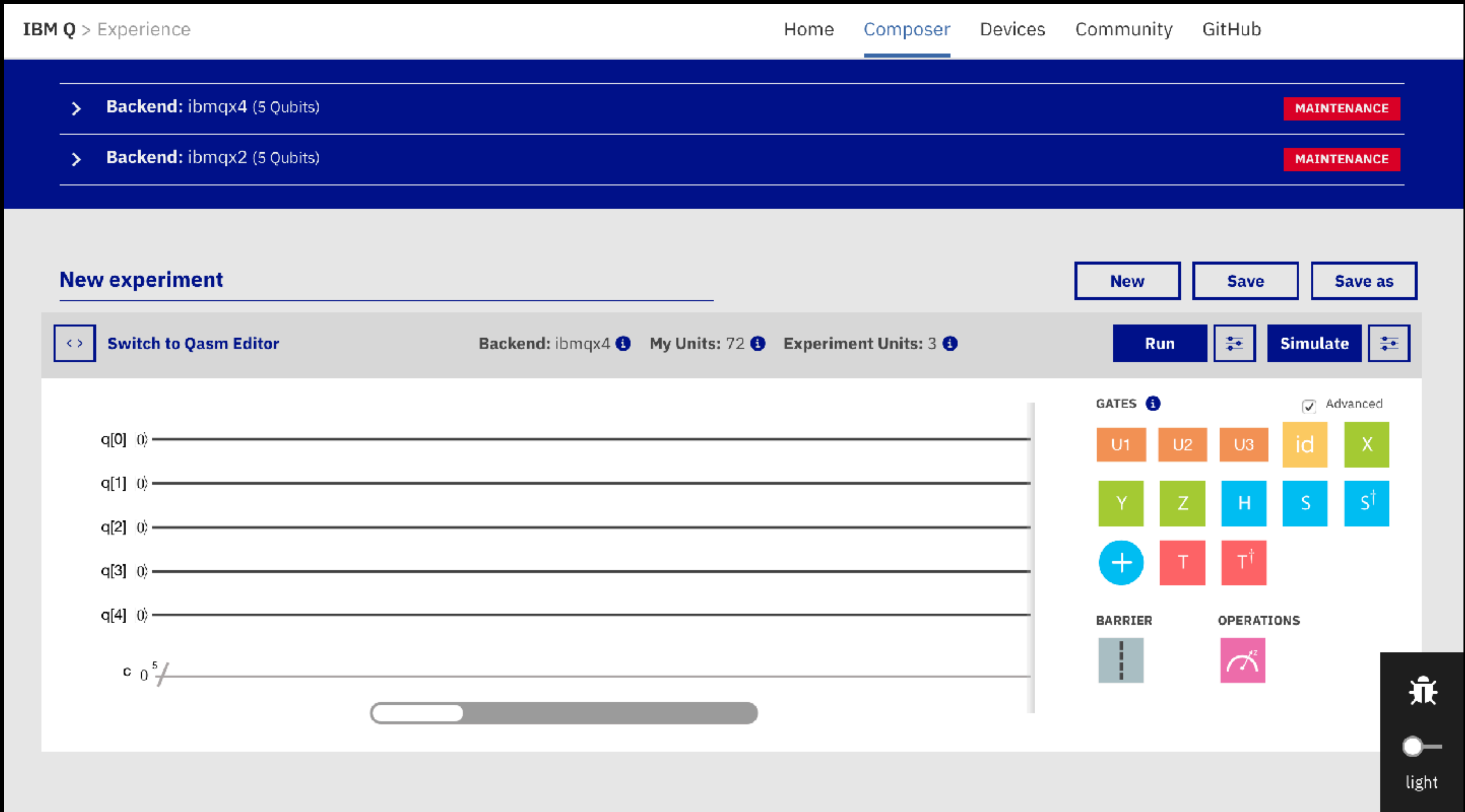
Y Z H S S†

+

T T†

BARRIER OPERATIONS

light



github

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

The screenshot shows the GitHub repository page for QISKit/qiskit-sdk-py. At the top, there is a navigation bar with the GitHub logo, 'This repository', a search bar, and links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. On the right side of the navigation bar, there are icons for adding a repository and a user profile. Below the navigation bar, the repository name 'QISKit / qiskit-sdk-py' is displayed, along with statistics: 'Watch 209', 'Star 1,434', and 'Fork 383'. A secondary navigation bar includes 'Code', 'Issues 30', 'Pull requests 13', 'Projects 0', 'Wiki', and 'Insights'. The main content area features a description: 'Quantum Software Development Kit for writing quantum computing experiments, programs, and applications.' followed by the website URL 'http://www.qiskit.org'. Below this, there are tags for 'quantum-computing', 'qiskit', 'sdk', 'python', 'quantum-programming-language', and 'cpp'. A summary bar shows '1,314 commits', '2 branches', '25 releases', '46 contributors', and 'Apache-2.0' license. A row of buttons includes 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The commit history section shows a recent commit by 'LuhangYang and diego-plan9' titled 'Adding a Chinese translation of README (#304)'. Below this, a list of folders and their corresponding commit messages and dates is provided.

Folder	Commit Message	Time Ago
.github	Add templates for issues and pull requests	7 months ago
cmake	Revise travis configuration, using cmake	5 days ago
doc	Adding a Chinese translation of README (#304)	4 days ago
examples	Circuit image plotter (#295)	9 days ago
images	Update Sphinx theme color and diagrams	4 months ago
qiskit	Issue #264: `__mklexer__` was not setting data. (#306)	4 days ago
src/qiskit-simulator	* Add a previous step for pypi_package_sdit target, so the	11 days ago
test	Issue #264: `__mklexer__` was not setting data. (#306)	4 days ago



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

© IBM Research

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 Quantum community](#).

Introduction

Many people tend to think quantum physics is hard math. This is actually not true, quantum algebra classes you probably did in first year university or even at high school. What is not so simple is the underlying theory. Instead you need to accept counter-intuitive ideas and in the end 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 behave randomly, are somehow strongly correlated.

In [26]:

```

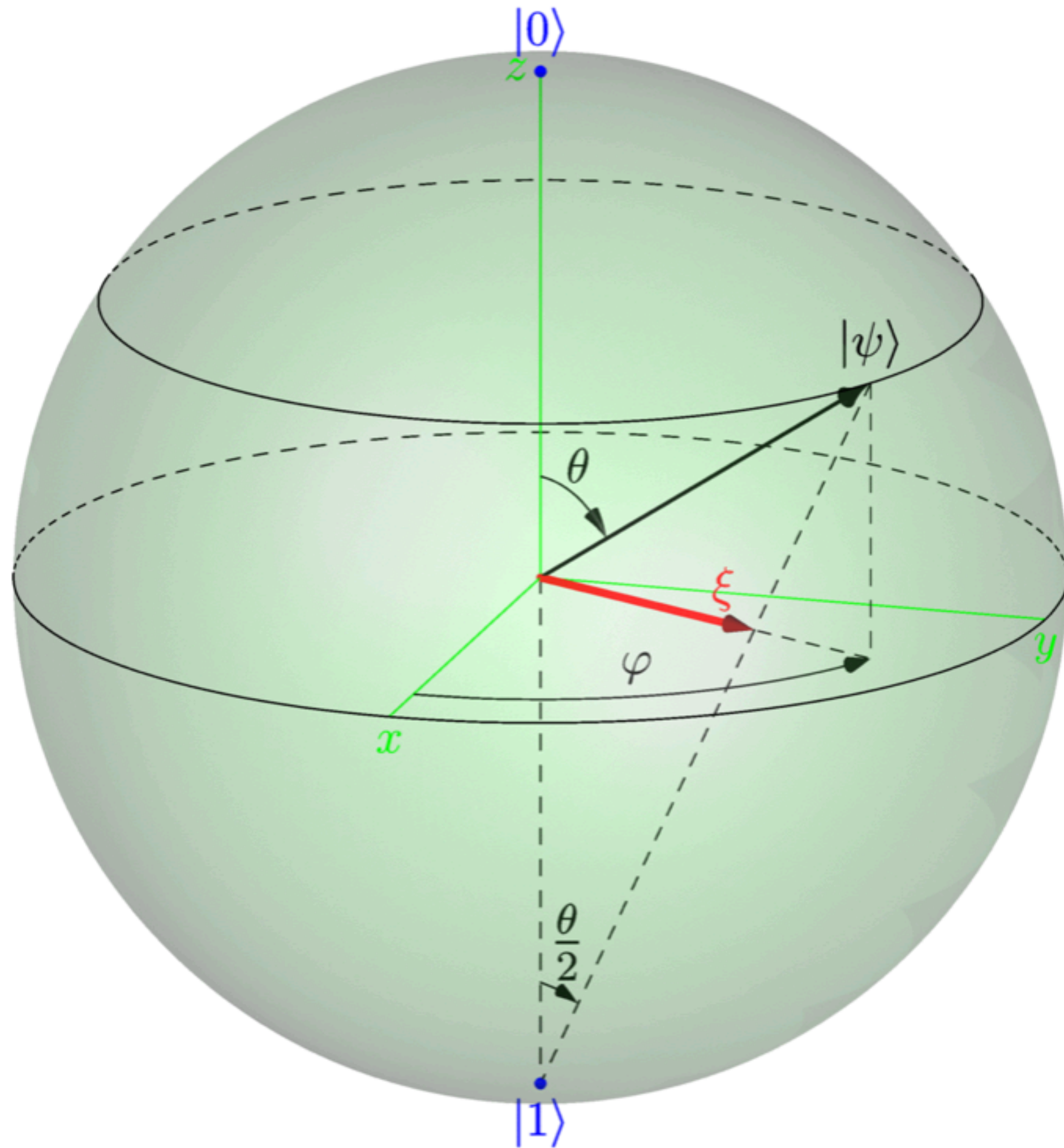
from IBMQuantumExperience import IBMQuantumExperience
from IPython.display import Image, display
import matplotlib.pyplot as plt
import numpy as np
import scipy as sp
import scipy.linalg
%matplotlib inline

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

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()
    
```

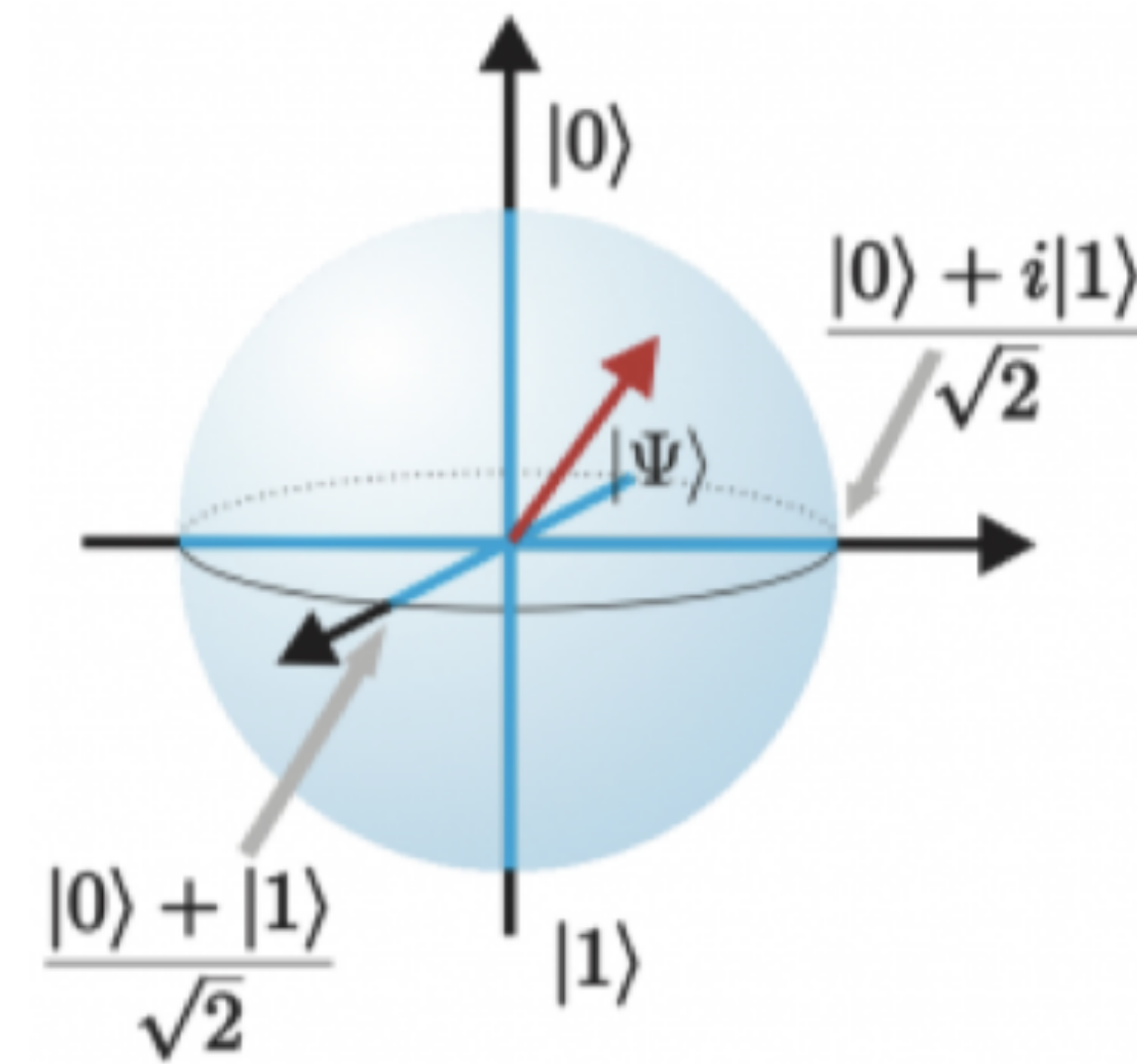
Sphère de Bloch (ou de Poincaré)

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.	 U2	The second physical gate of the Quantum Experience. It is a two parameter single-qubit gate with duration one unit of time.	 U3	The third physical gate of the Quantum Experience. It is a three-parameter single-qubit gate with duration 2 units of gate time.	 id	The identity gate performs an idle operation on the qubit for a time equal to one unit of time.
	QASM Matrix		QASM Matrix		QASM Matrix		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.	 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 $Y = XZ$.	 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.	 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		QASM Matrix		QASM Matrix		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.	 S†	The Phase gate that is the transposed conjugate of S and has the property that it maps $X \rightarrow -Y$, and $Z \rightarrow Z$.	 +	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.	 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		QASM Matrix		QASM Matrix		QASM Matrix
 T†	The Phase gate that is the transposed conjugate of T .		The barrier prevents transformations across this source line.		Measurement in the computational (standard) basis (Z).	 if	Conditionally apply quantum operation
	QASM Matrix		QASM Matrix		QASM Matrix		QASM Matrix
 $ 0\rangle$	Prepare qubits in the $ 0\rangle$ state.						
	QASM Matrix						

Quelques portes quantiques

Porte de Hadamard

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

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

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

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 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)

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

Porte de Pauli Z

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

Portes réversibles et matrices unitaires ($MM^* = \text{Id}$)

Exemple mise en oeuvre porte Hadamard

IBM Q > Experience Home Composer Devices Community GitHub

> Backend: ibmqx4 (5 Qubits) MAINTENANCE

> Backend: ibmqx2 (5 Qubits) MAINTENANCE

New experiment New Save Save as

<> Switch to Qasm Editor Backend: ibmqx4 ⓘ My Units: 72 ⓘ Experiment Units: 3 ⓘ Run ⚙ Simulate ⚙

q[0] |0> H M

q[1] |0>

q[2] |0>

q[3] |0>

q[4] |0>

c[0] 0 0

GATES ⓘ Advanced

U1	U2	U3	id	X
Y	Z	H	S	S†
+	T	T†		

BARRIER **OPERATIONS**

⋮	M
---	---

Quantum Scores (13 scores) Refresh Remove All

Pages 1 2

light

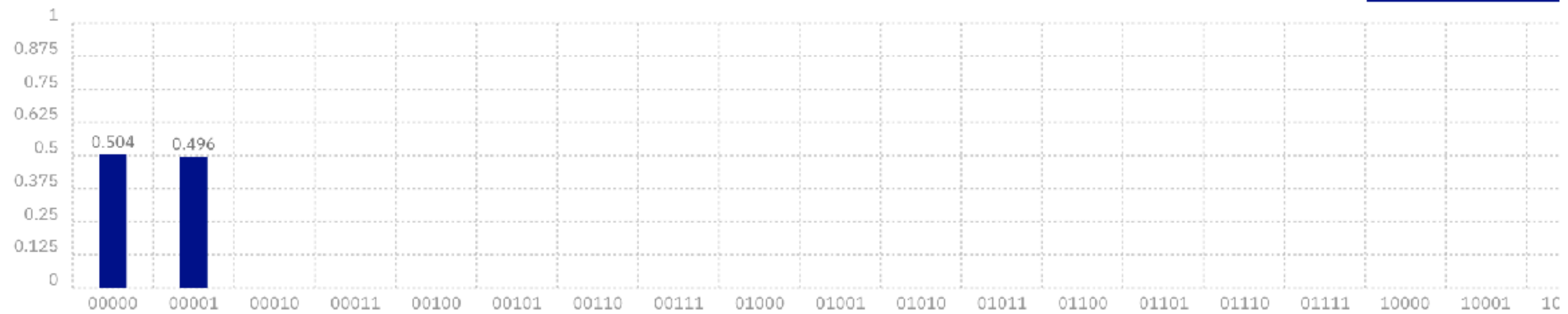
Exemple résultat porte Hadamard

Experiment #20180227223201

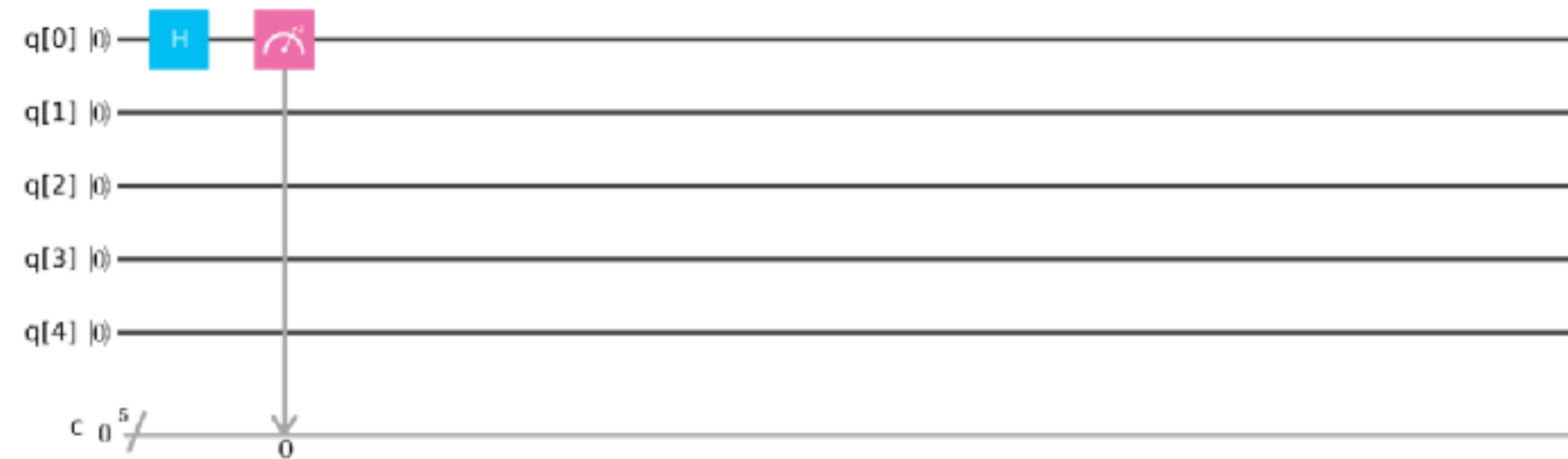
Device: ibmqx4

Quantum State: Computation Basis

Download CSV



Quantum Circuit



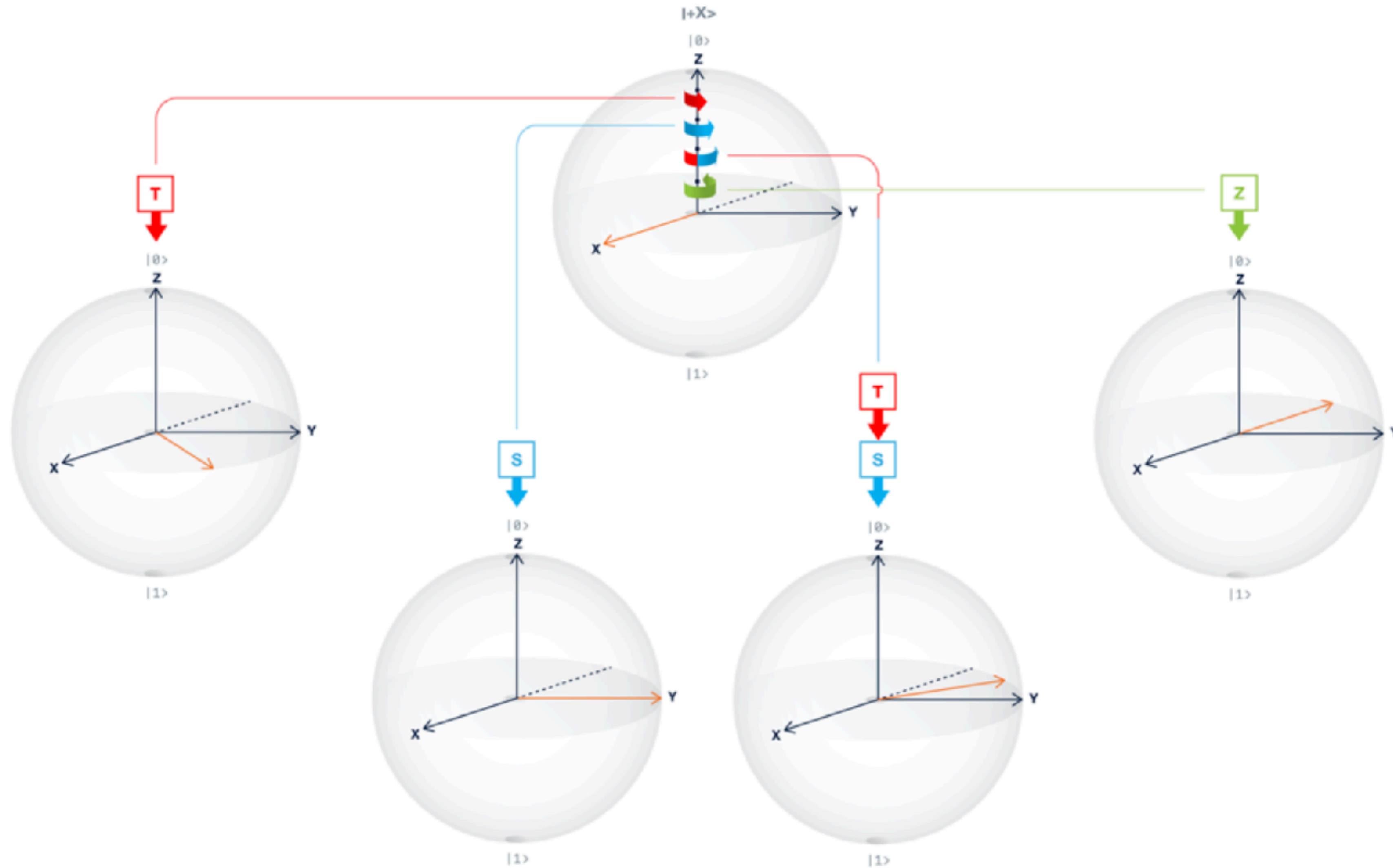
OPENQASM 2.0

```
1 include "qelib1.inc";
2
3 qreg q[5];
4 creg c[5];
5
6 h q[0];
7 measure q[0] -> c[0];
8
```





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
H H 	0	1.0	0
H T H 	$\pi/4$	0.85	0.15
H S H 	$\pi/2$	0.50	0.50
H S T H 	$3\pi/4$	0.15	0.85
H Z H 	π	0	1



Create +X state

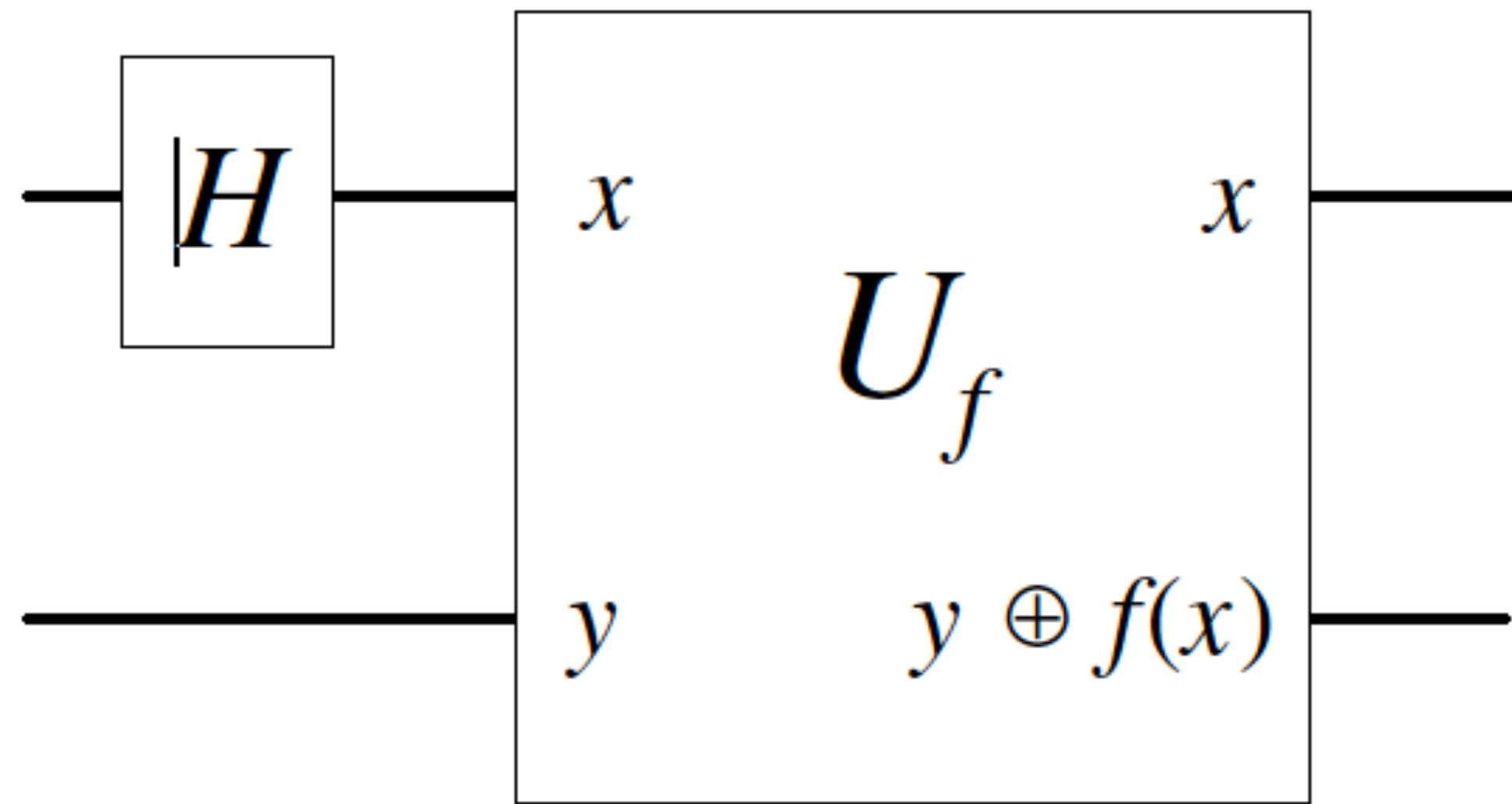


Change qubit phase



Measure in superposition basis

Le « parallélisme » quantique



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

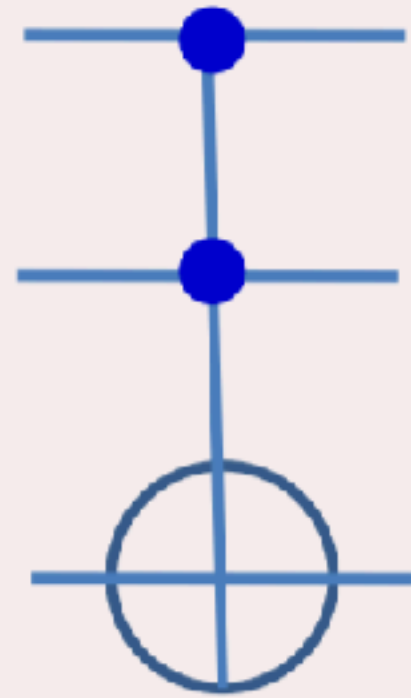
$$|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$$

Principaux algorithmes et bibliothèques existants

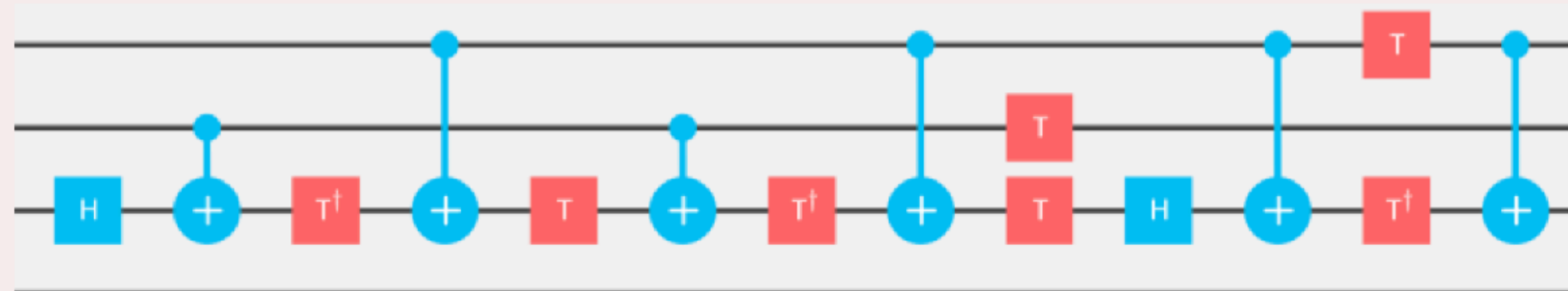
- Algorithmes de Deutsch-Jozsa (1992)
 - Déterminer si une fonction binaire à valeur binaire est constante ou équilibrée
- Algorithme de Grover
 - Recherche d'éléments répondant à certains critères dans des tables
- Algorithme de Shor
 - Factoriser un nombre composé
- Bibliothèque HHL (Harrow, Hassidim, Lloyd)
 - Algèbre linéaire
- Traitement du signal
 - Algorithme FFT

Exemples composition portes

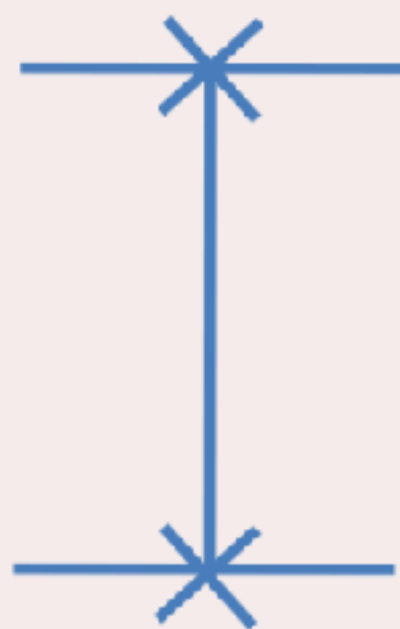
Toffoli



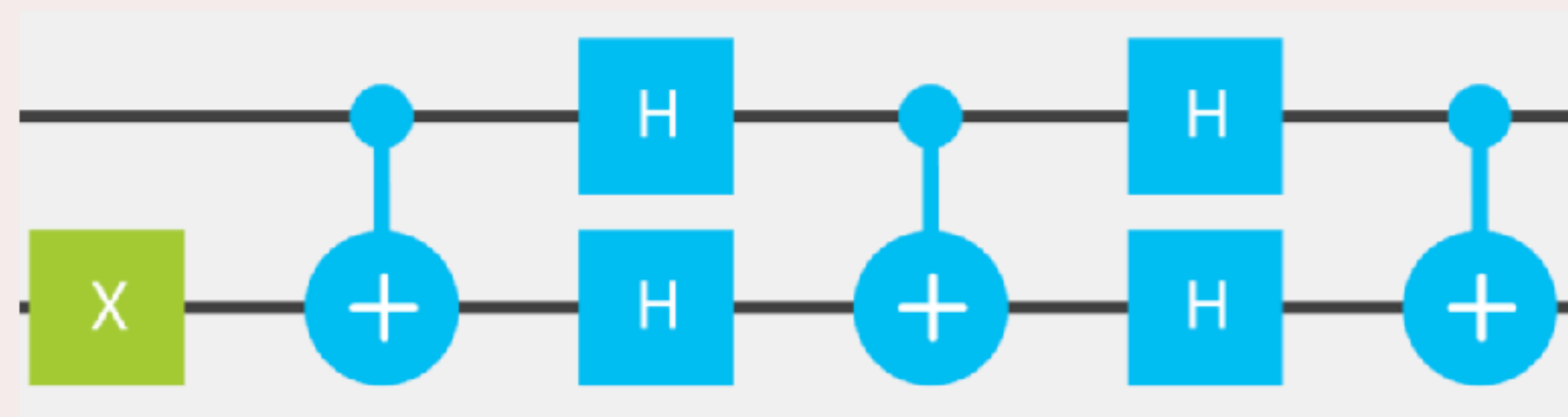
=



Swap Gates

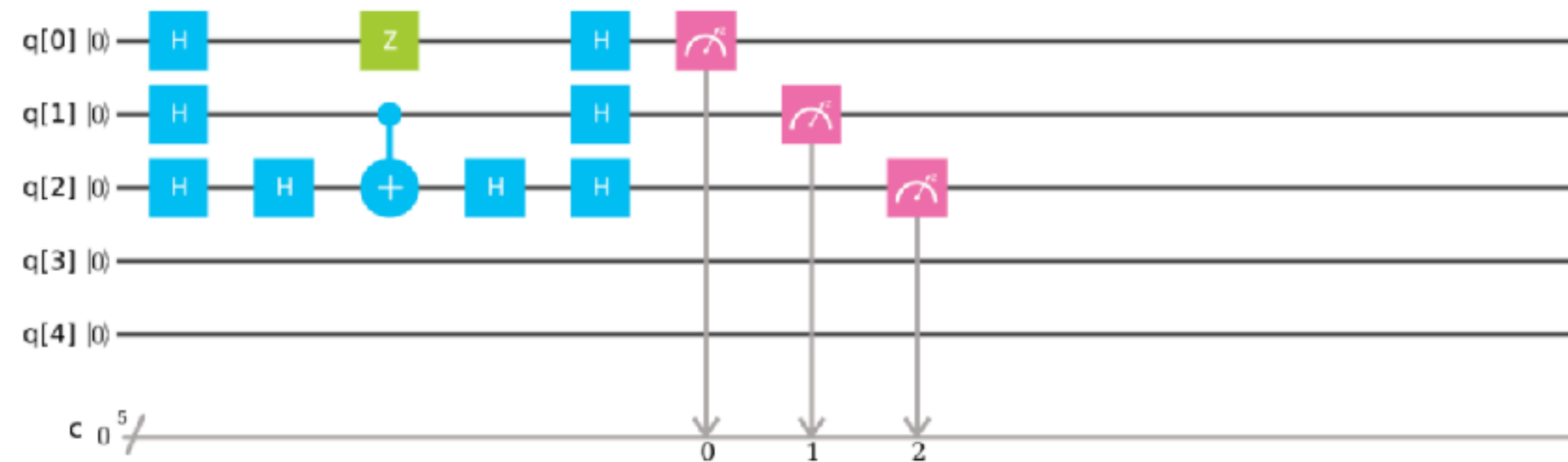


=



Exemple algorithme Deutsch-Josua

DJ N=3 Example



DJ N=3 Constant



