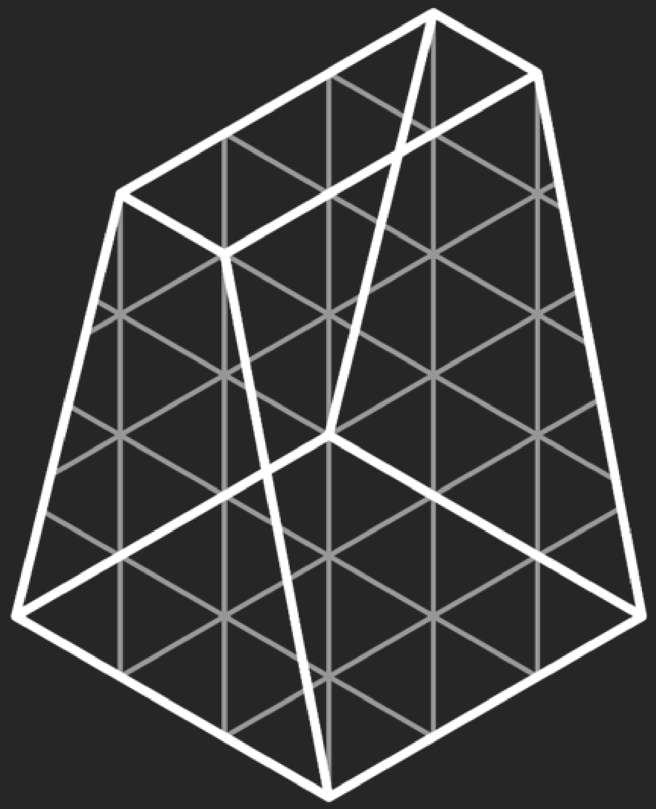
Microsoft



Quantum Computing mit Azure Quantum

Azure MeetUp Frankfurt
08/06/2022

Dr. Holger Sirtl
Microsoft
holger.sirtl@microsoft.com
@hsirtl

Agenda

Why do we need a new way of computing?

How is Quantum Computing different?

Is Quantum Computing ready for productive scenarios? What use cases are relevant?

How can you use it effectively today? What does Azure Quantum provide?

How can you prepare for this technology?

Moore‘s Law

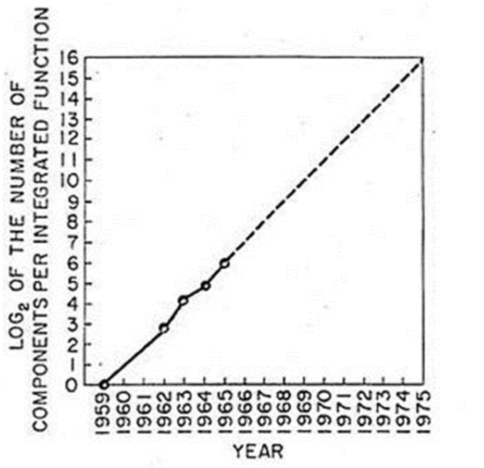
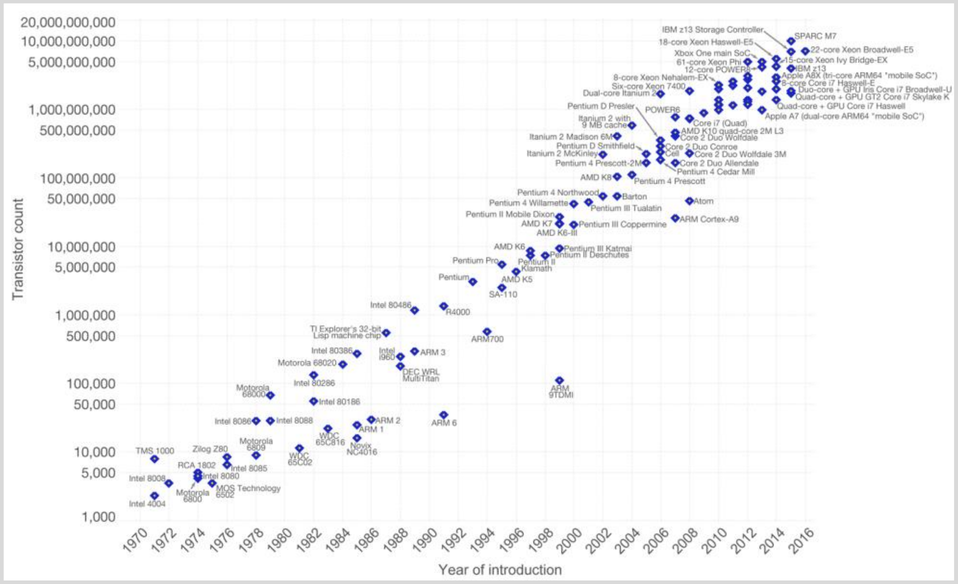
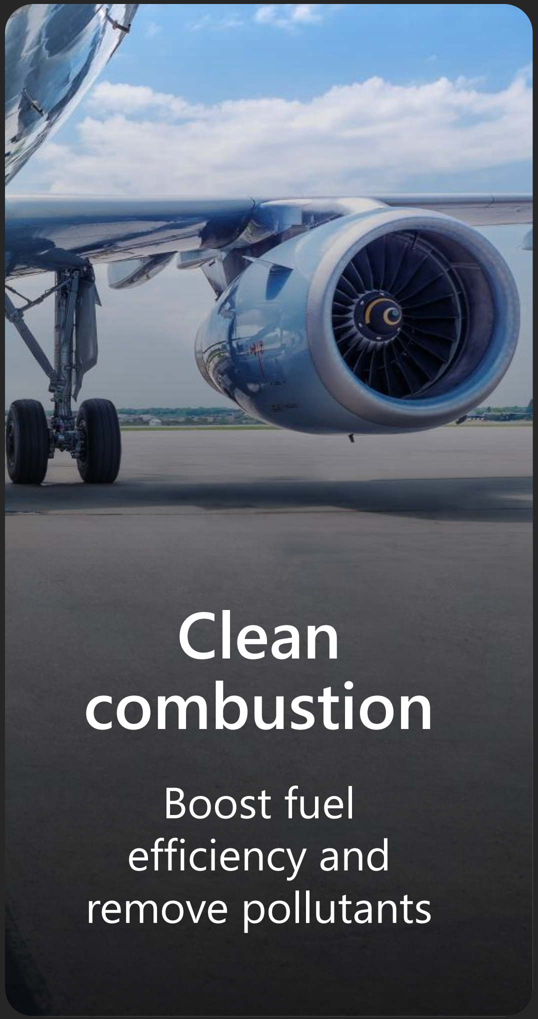


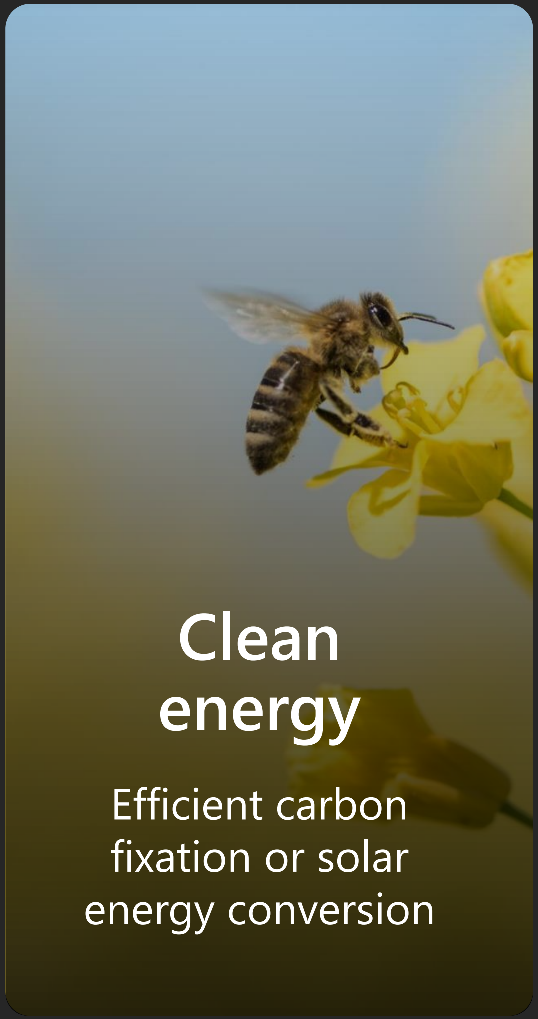
Fig. 2 Number of components per integrated function for minimum cost per component extrapolated vs time.

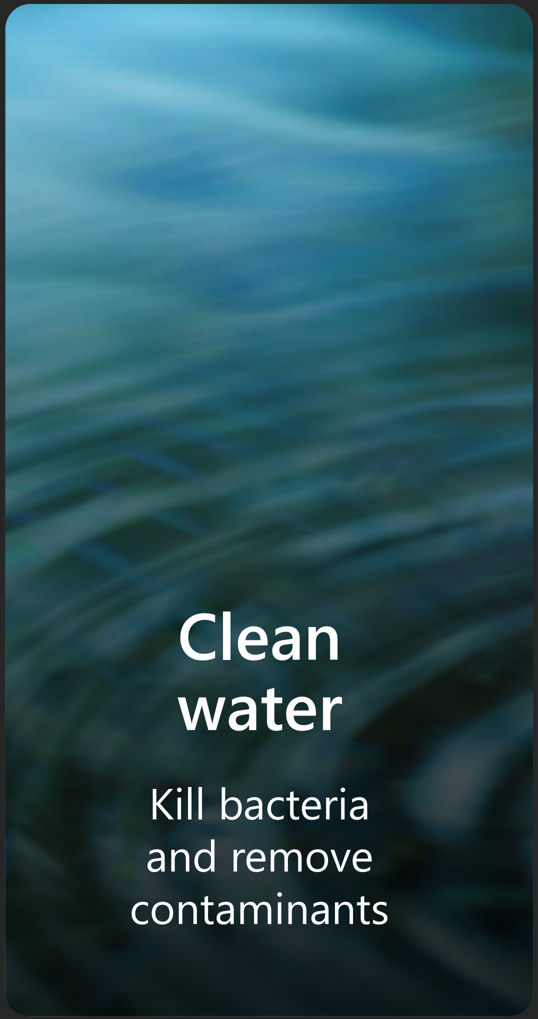


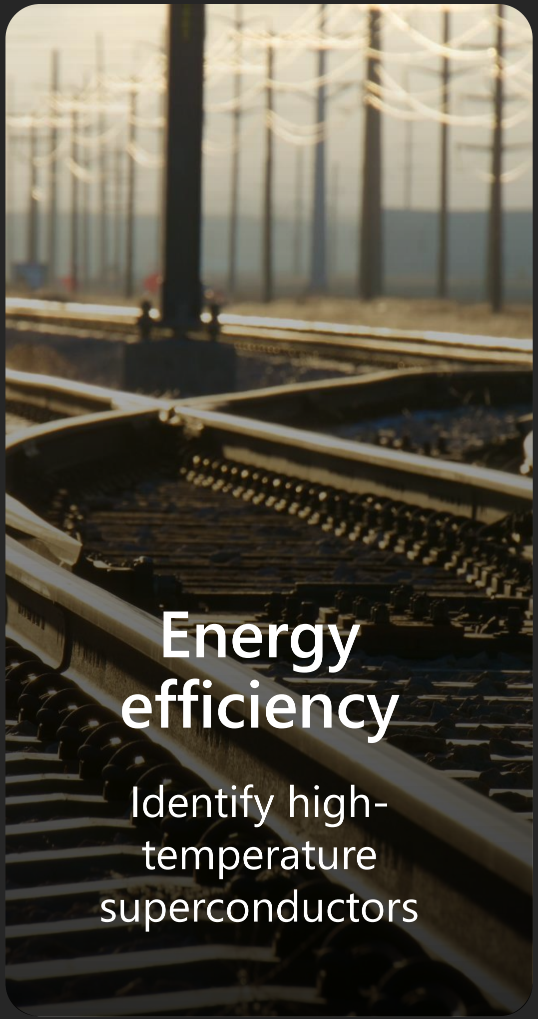
George Moore, known for **Moore's Law**
„the observation that the number of transistors on integrated circuits doubles approximately every two years"

Quantum will disruptively impact chemistry/materials science



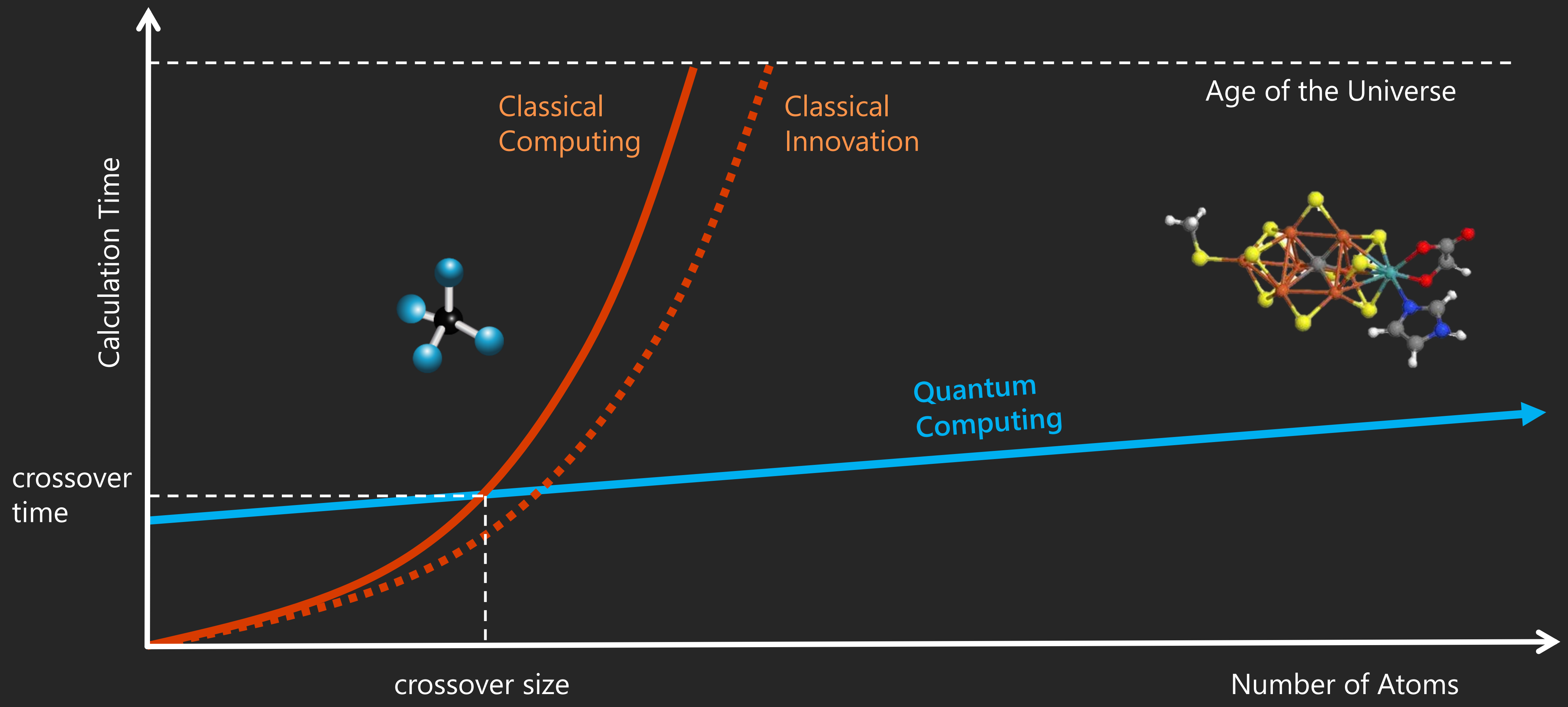






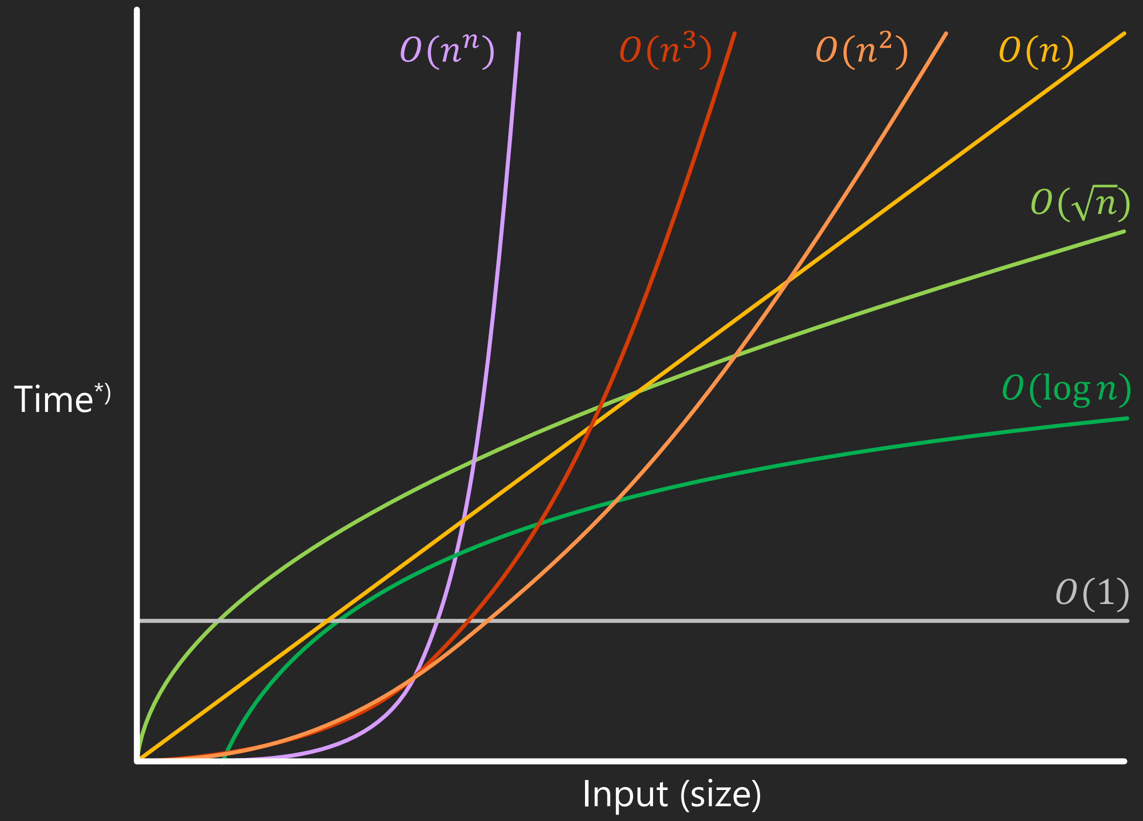
Microsoft

Quantum chemistry: classically intractable



Microsoft

Algorithm Complexity and Quantum Speedup



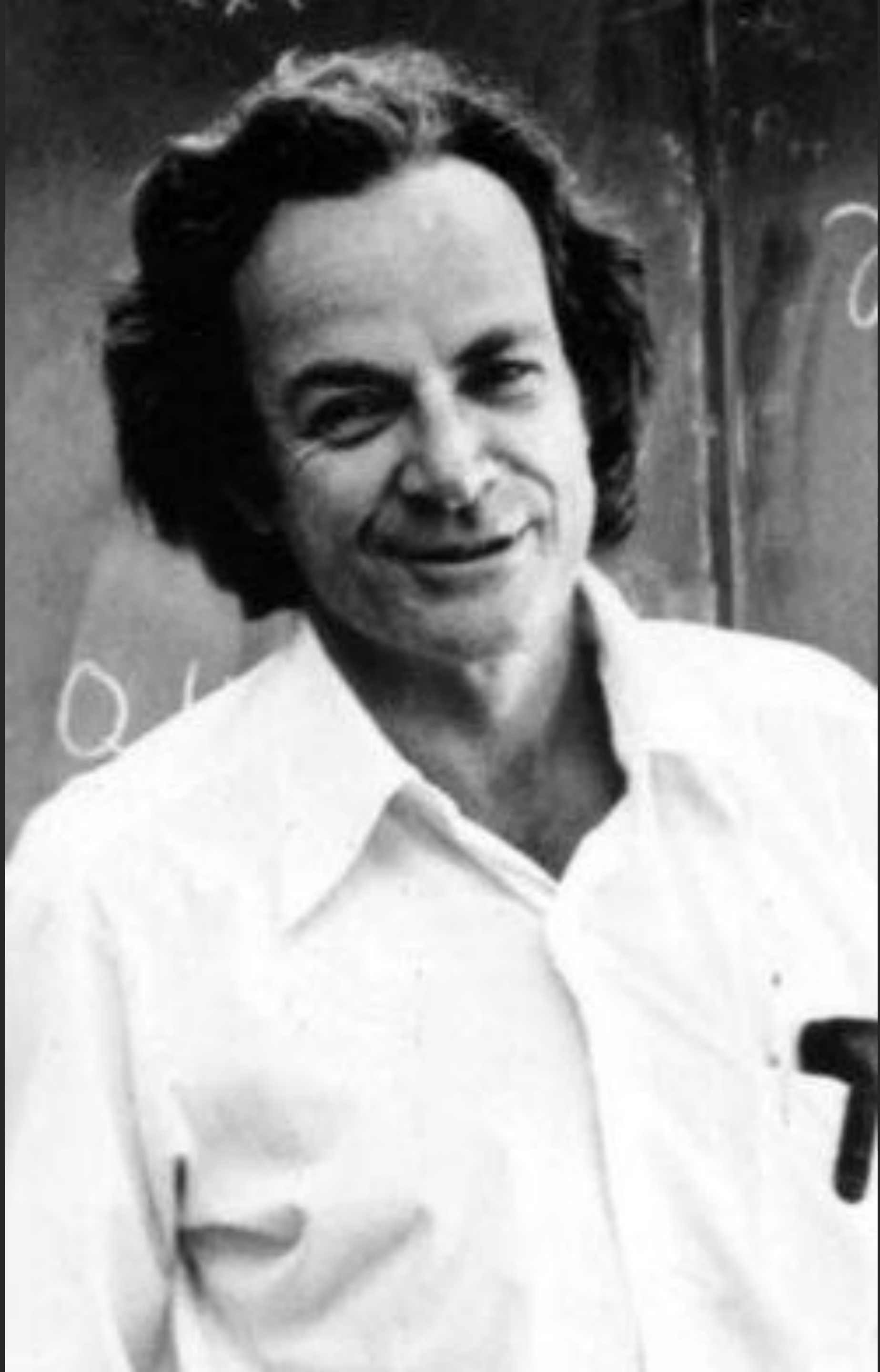
Algorithm	Classical Resources	Quantum Resources	Quantum Advantage
Simulation (quantum chemistry)	2^n (for n atoms)	n^c	Exponential
Factoring	2^n (for n digits)	n^3	Exponential
Linear Systems (Ax = b)	2^n (for n digits)	$\sim n$	Exponential
Optimization	2^n (for n items)	?	?
Search (unsorted / unstructured)	n (for n entries)	$\sqrt{n}$	Polynomial

*) graphs only for comparison, not correctly scaled

„Nature seems to be able to efficiently simulate itself. Since nature is quantum, maybe we need to use quantum mechanics in our computers."

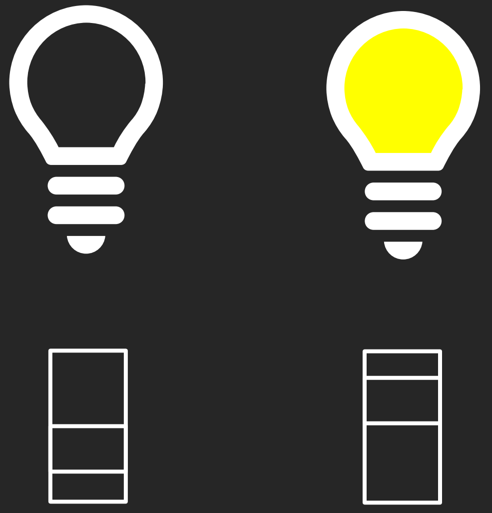
Richard Feynman

Theoretical Physicist, Nobel prize laureate 1965
1918 - 1988



Microsoft

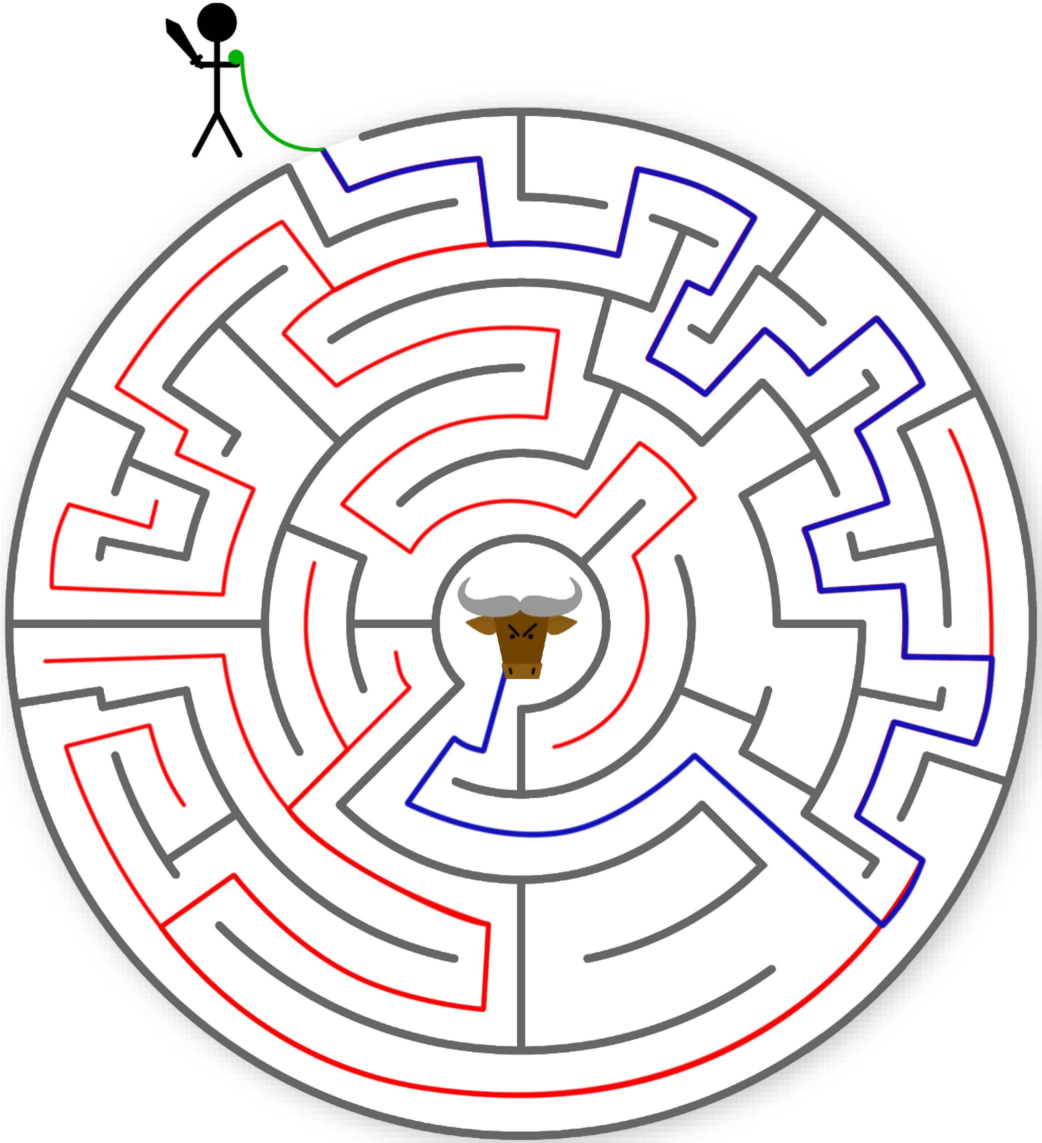
The Classical Bit



In classic computing, it's 0 **OR** 1, ...

0 1

Classical Bit
(cbit)

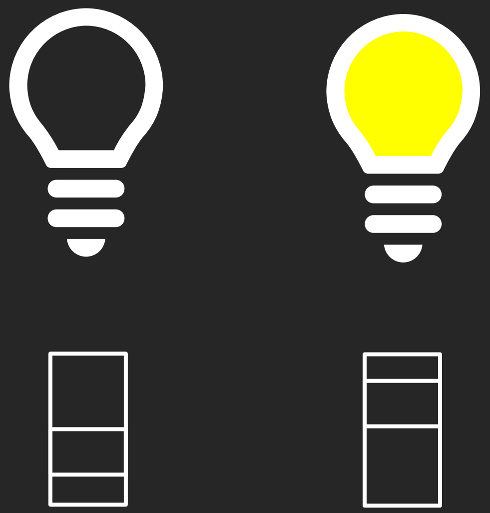


Superposition of a Quantum Bit (QuBit)

*In quantum computing, it's 0 **AND** 1.*

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

Quantum Bit
(qubit)



*In classic computing, it's 0 **OR** 1, ...*

$$0 \qquad 1$$

Classical Bit
(cbit)

Exponential Scaling

For simulating n qubits on a classical computer you need 2^n classical bits.

1 qubit → 2 bits

20 qubits → 16 MB

30 qubits → 16 GB

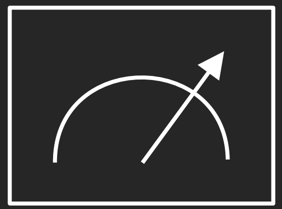
40 qubits → 16 TB

50 qubits → 16 PB

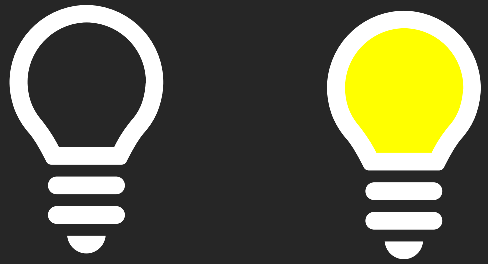
260 qubits → computer that has more bits then atoms in visible universe.

Microsoft

Measurement



Measuring makes the superposition state collapse to a „classic“ state.



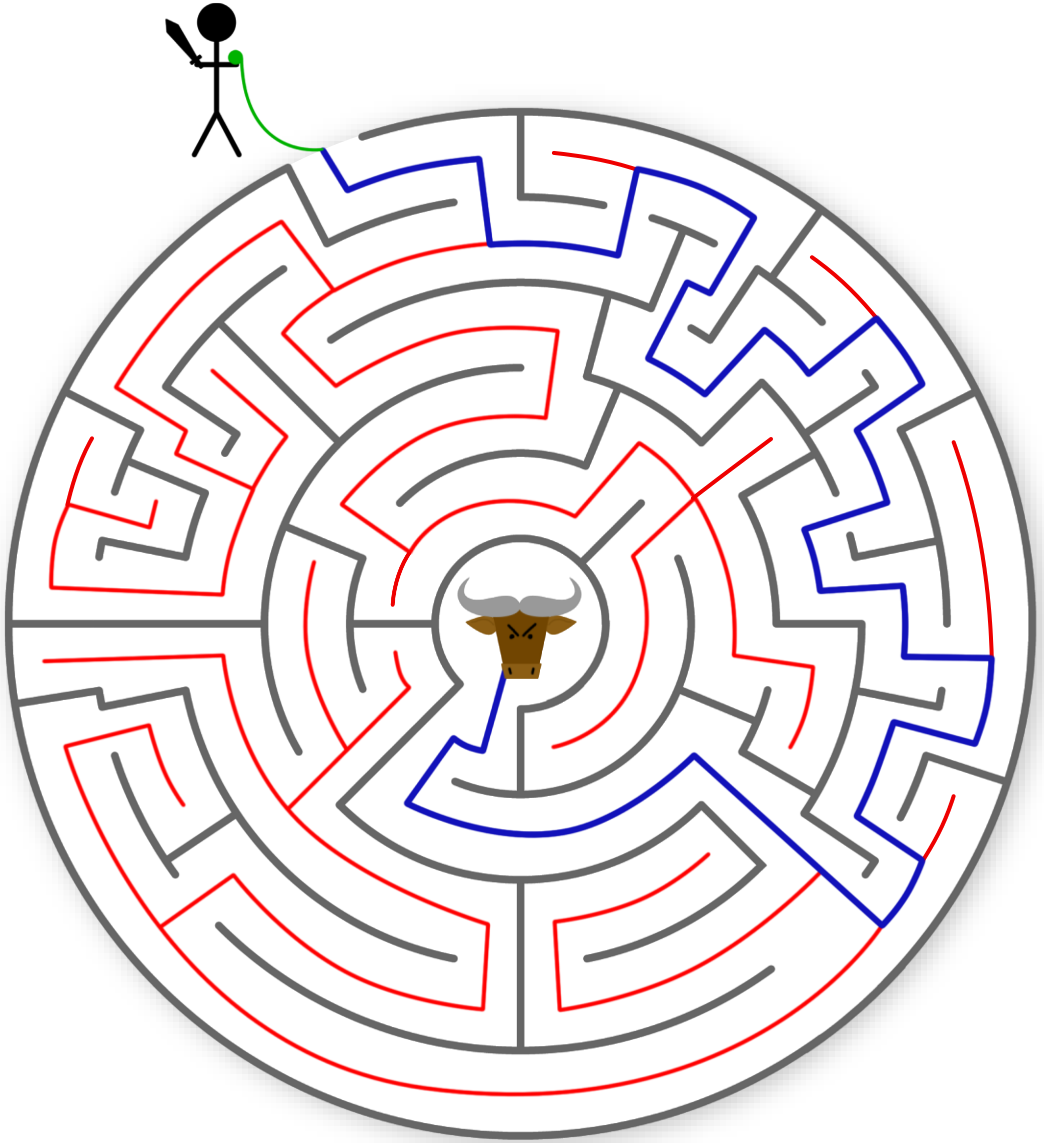
*The probabilities for measuring either 0 or 1 are determined by these values – called **amplitudes**.*

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

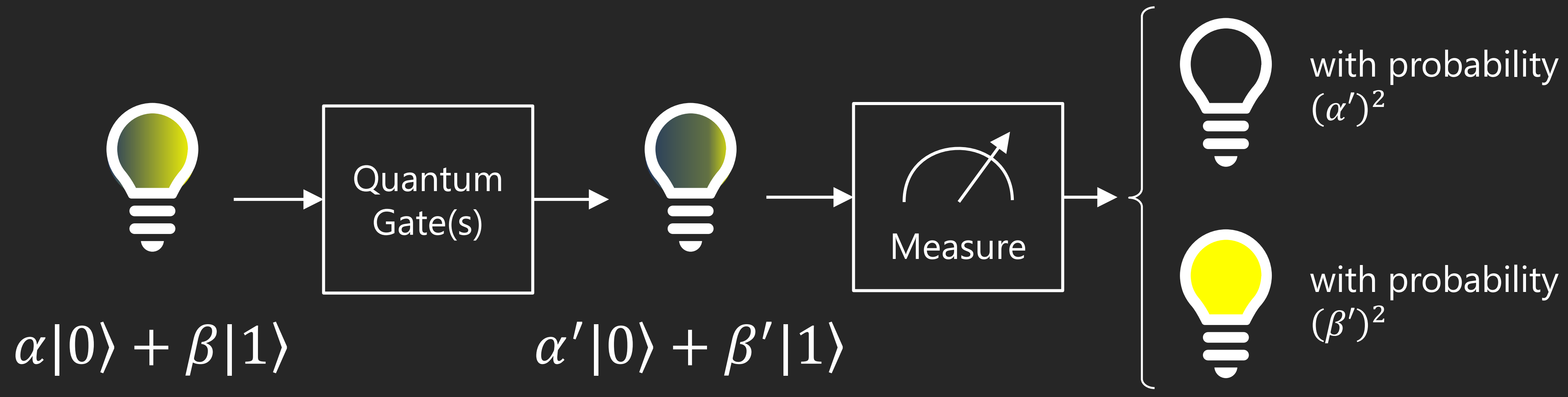
0 1

Quantum Bit (qubit)

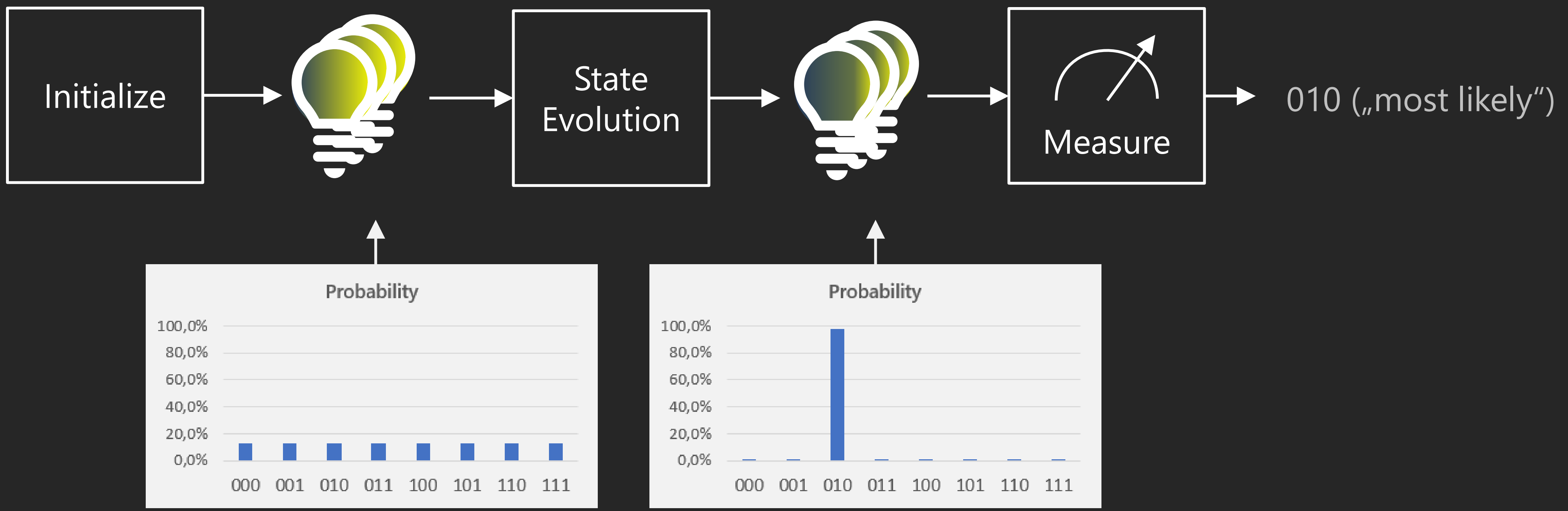
Classical Bit (cbit)



Quantum Gates



Quantum Algorithms



Microsoft

Summary of Quantum Effects

Superposition

allows storing and calculating with multiple states at the same time.

Quantum Gates

transfer one quantum state into another, using effects like interference, tunnelling, entanglement.

Measurement

returns one classical state with probability determined by the state amplitudes.

Use Case Categories for Quantum Computing

	Optimization	Simulation	Dynamic Systems	Cryptography
Methods	• Quantum Machine Learning & Optimization	• Simulation of quantum mechanical systems	• Improved solving of systems of linear equations	• Quantum cryptography
Applications	• Resource Optimization	• Simulation of molecules and chemical reations • Prediction of material characteristics	• Fluid simulation • Weather predictions	• Encrypted communication • Security

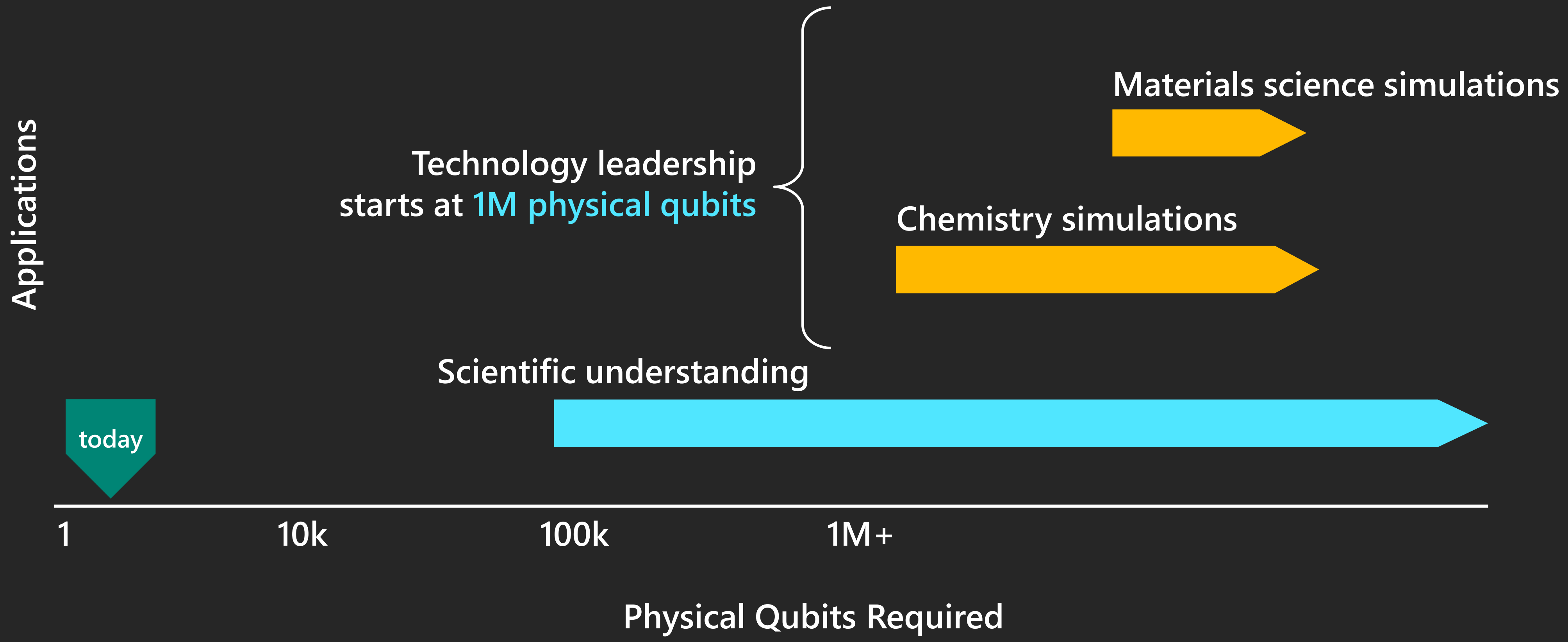
Source:
Quantum Machine Learning – Eine Analyse zu Kompetenz, Forschung und Anwendung
Fraunhofer, 2020

Industry Use Cases

	Pharma and chemical industry	Logistics and Mobility	Financial Services	Automotive and Manufacturing	Information Security
Phase 1: **Noisy Intermediate Scale Quantum (NISQ)** Physical qubits: <1000 Logical qubits: n/a Time: today-5 yrs.	• Clinical-trial site-selection optimization • Synthetic-data generation • Ground state analysis for small molecules	• Route and fleet optimization • Search problems	• Certified randomness • Collateral and portfolio optimization • Fraud detection	• Job scheduling optimization • Robot path optimization	• Secure encryption • Secure communication
Phase 2: **Quantum Supremacy** Physical qubits: <1M Logical qubits: <1000 Time: 5-15 yrs.	• Catalyst and enzyme design • Large-molecule simulation (e.g., protein folding)	• Weather simulation	• Trading strategy optimization • Risk assessments • Market simulation	• Supply-chain optimization • Material design • Fluid mechanics	• Breaking RSA cryptography
Phase 3: **Fault tolerant universal quantum computing** Physical qubits: <10M Logical qubits: 10.000 Time: >15-20 yrs.	• Personalized medicine			• Multiscale production optimization	

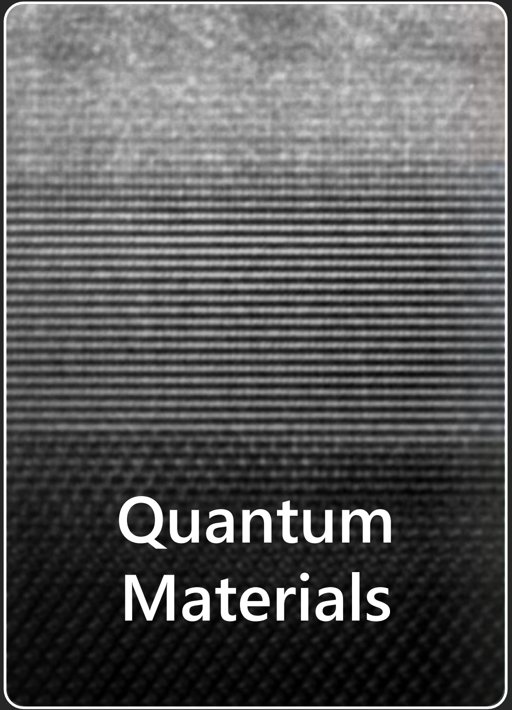
Sources:
Quantum Machine Learning – Eine Analyse zu Kompetenz, Forschung und Anwendung, Fraunhofer, 2020
Quantum computing use cases are getting real—what you need to know, McKinsey, 14/12/2021

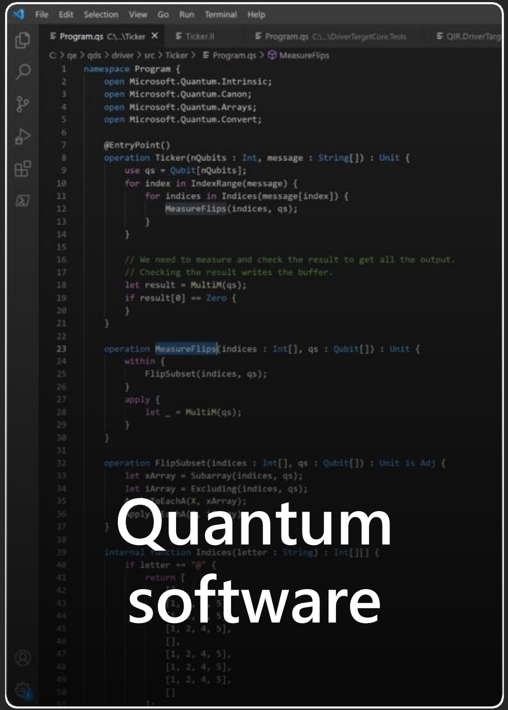
Impactful applications require at least 1M qubits



Microsoft’s unique approach to scalable quantum systems

Solve industrial scale at the physics layer with topological qubits while building the world’s best quantum computing platform





Quantum Computing at Microsoft

Research
Innovations across every layer of the quantum stack, from software and applications to control and devices.

Quantum Development Kit
Build quantum applications to run on quantum hardware, simulators, or optimization solvers.

Azure Quantum
Fully managed quantum and optimization service.

Azure Quantum Network
Community of solution partners, affiliates, customers, educators, or researchers.

Enterprise Acceleration Program
collaborate with experts from Microsoft to develop custom quantum solutions.

Azure Quantum

Fully managed quantum and optimization service.

COME WITH YOUR CODE

python
Cirq
Qiskit
Q#

DIVERSE HARDWARE

IONQ
PASQAL
QUANTINUUM
qci
rigetti

HIGH-PERFORMANCE QUANTUM CLASSICAL HYBRID

QIR ALLIANCE

INTEGRATE AND SCALE WITH AZURE

Serverless - AI

Edge Device – Azure Stack Hub

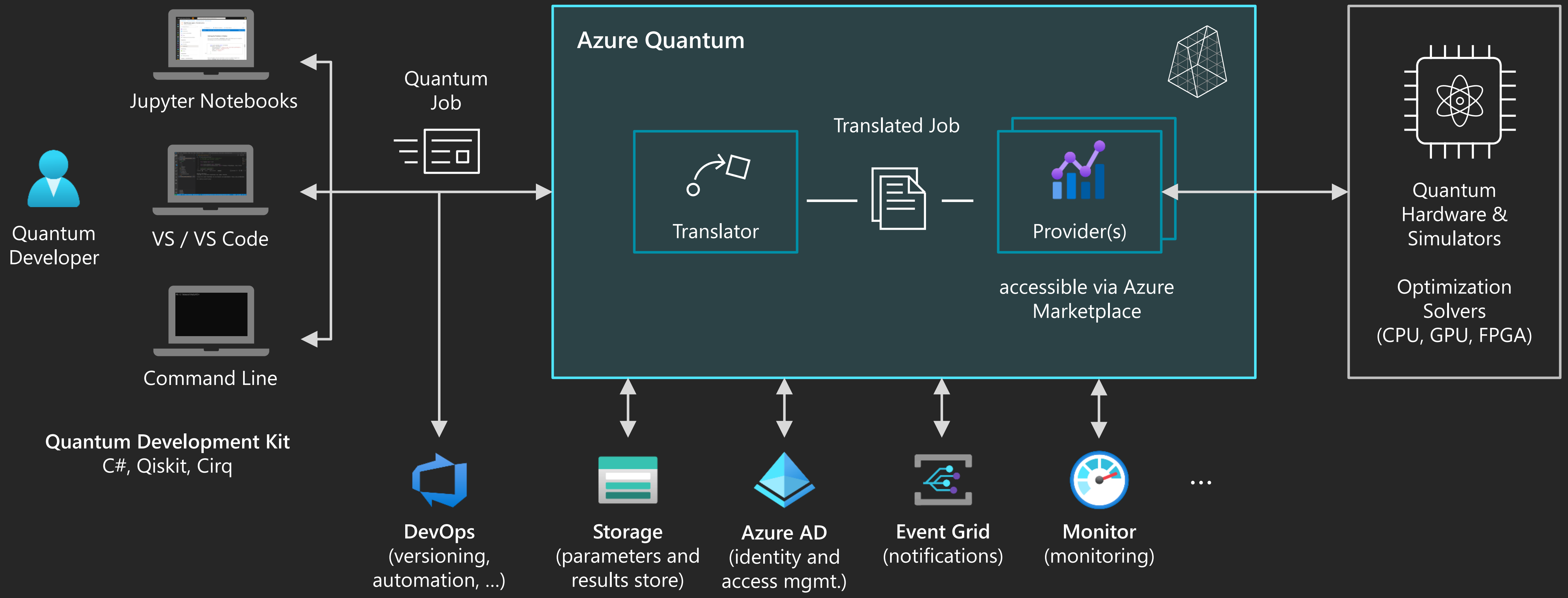
Infrastructure – Security

Tools - Github

https://azure.microsoft.com/en-us/services/quantum/

DEMO

Azure Quantum Architecture



Explore and innovate with Quantum Compute

Build	Integrate	Test	Run
Choose your language, tools, pre-built libraries	Connect classical to quantum, leverage the Azure platform	Simulate and estimate, tune for best quality & performance	code on quantum hardware
Q#, Qiskit, Cirq VS/VS Code Jupyter Notebooks QDK Libraries	Python (Qiskit, Cirq), .NET Industry and research libraries Azure Services: Azure DevOps, Azure ML, Azure Functions	Client & Cloud Simulators (Full State, Open Sys, etc.) QDK Resource Estimator	IONQ qci PASQAL TOSHIBA rigetti QUANTINUUM

Same quantum code

https://docs.microsoft.com/en-us/azure/quantum/overview-azure-quantum#workflow-of-the-quantum-software-development

Specifying Quantum Algorithms

Type of Quantum Computing

(Universal) Quantum Computing

Quantum Optimiziation (Adiabatic/ Annealing)

Algorithm Specification

Gate Circuit

describes the sequence of gates to be applied

Cost Function

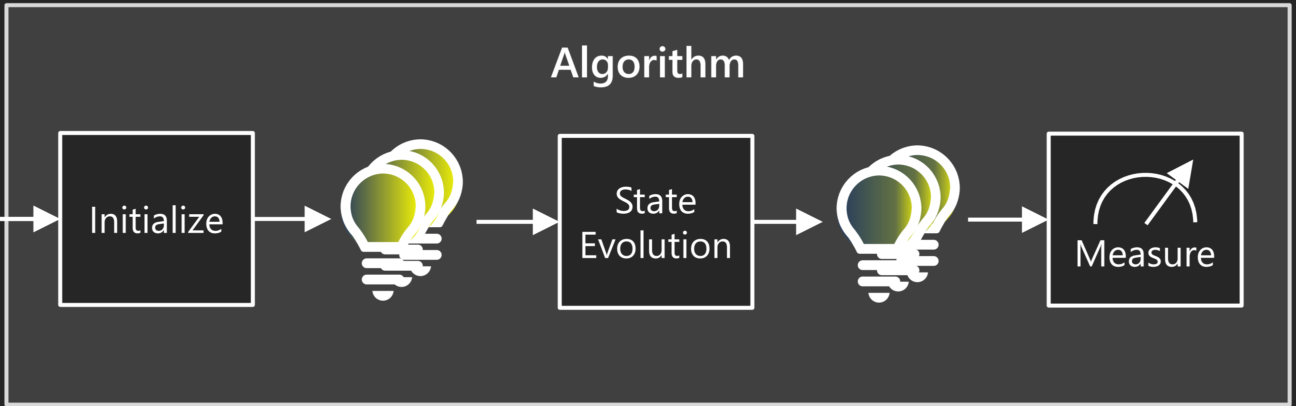
describes a function the system should find a minimum for

Algorithm

Initialize → State Evolution → Measure

Specifying Quantum Algorithms

Type of Quantum Computing	Algorithm Specification
(Universal) Quantum Computing	**Gate Circuit** describes the sequence of gates to be applied
Quantum Optimiziation (Adiabatic/ Annealing)	**Cost Function** describes a function the system should find a minimum for



Here, the most universal, most powerful quantum computers are used. But they are the hardest to build.

Here, specialized devices are used. Specialization allows for higher qubit-numbers. Today's systems support production scale compute.

Specifying Quantum Algorithms

Type of Quantum Computing	Algorithm Specification
(Universal) Quantum Computing	**Gate Circuit** describes the sequence of gates to be applied
Quantum Optimiziation (Adiabatic/ Annealing)	**Cost Function** describes a function the system should find a minimum for

```
namespace QuantumRNG {

    open Microsoft.Quantum.Intrinsic;
    open Microsoft.Quantum.Measurement;
    open Microsoft.Quantum.Canon;

    @EntryPoint()
    operation RandomNumberGenerator() : Result {
        // Allocate a qubit.
        use q = Qubit();
        // Put the qubit to superposition.
        H(q);
        // It now has 50% chance of being measured 0 or 1.
        // Measure the qubit value.
        return M(q);
    }
}
```

Specifying Quantum Algorithms

DEMO

Initialization

State Evolution

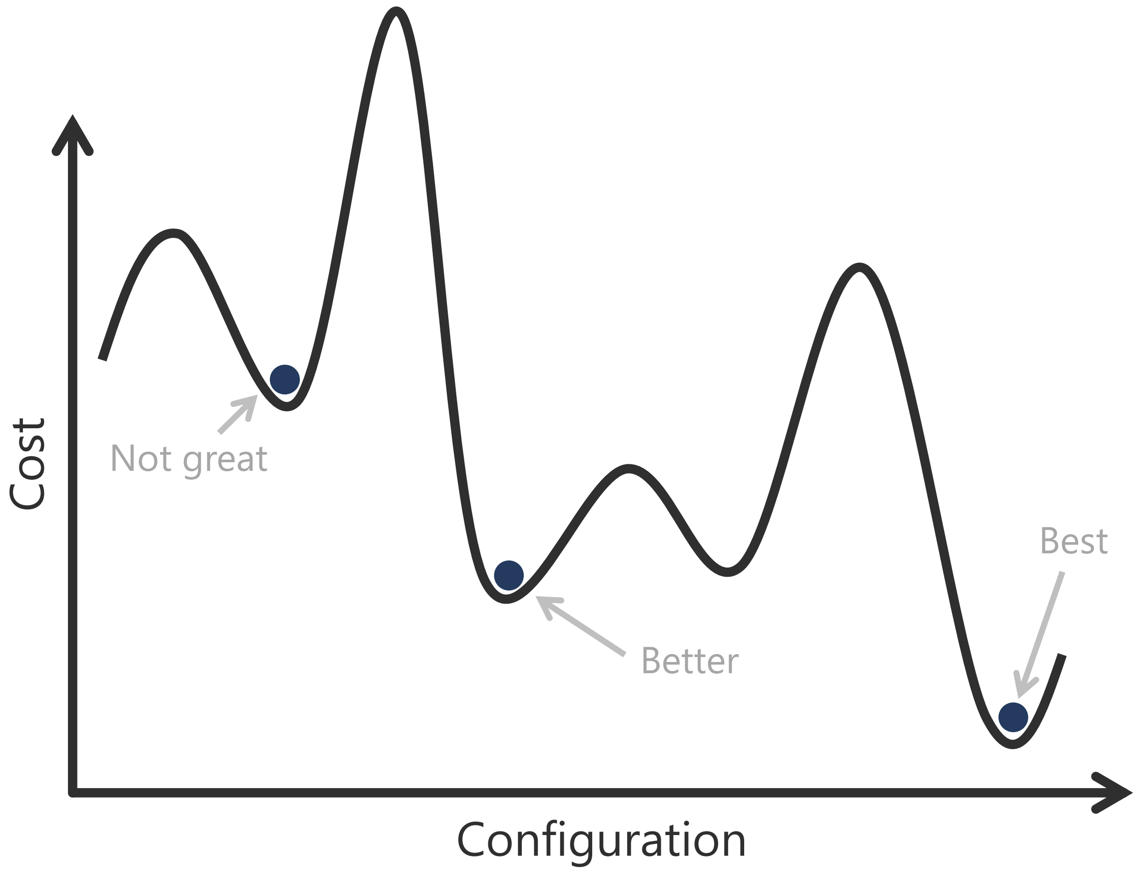
Measurement

```
namespace QuantumRNG {

    open Microsoft.Quantum.Intrinsic;
    open Microsoft.Quantum.Measurement;
    open Microsoft.Quantum.Canon;

    @EntryPoint()
    operation RandomNumberGenerator() : Result {
        // Allocate a qubit.
        use q = Qubit();
        // Put the qubit to superposition.
        H(q);
        // It now has 50% chance of being measured 0 or 1.
        // Measure the qubit value.
        return M(q);
    }
}
```

Optimization

Optimization problems are found in every industry, such as manufacturing, finance, transportation, and logistics.

One major challenge is to avoid getting stuck in a local minimum, and finding a global minimum instead.

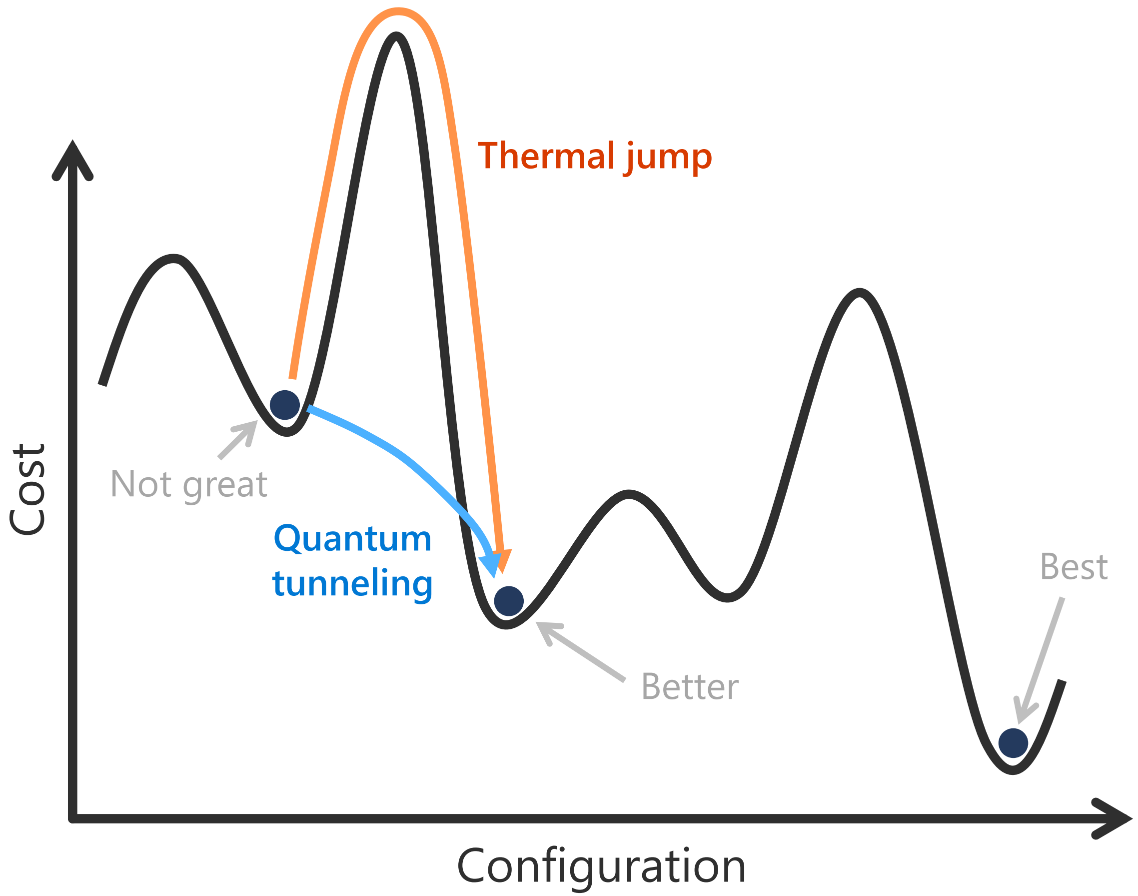


Microsoft

Optimization

Moving into the quantum age, optimizers have been developed that make use of quantum mechanics to accelerate optimization and escape local minima in the cost function landscape through quantum tunneling.

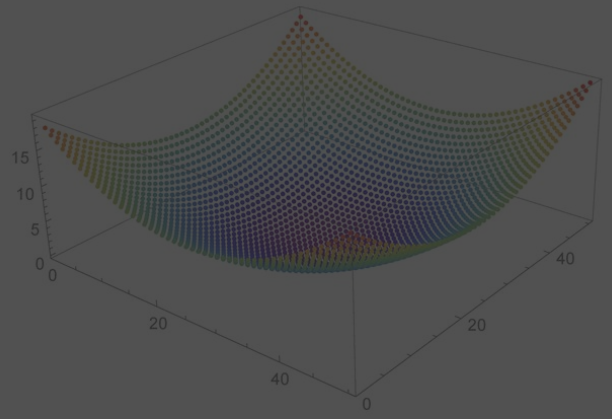
Emulating these quantum effects on classical computers has led to the development of new types of quantum solutions that run on classical hardware, also called “**quantum-inspired algorithms**”.



Microsoft

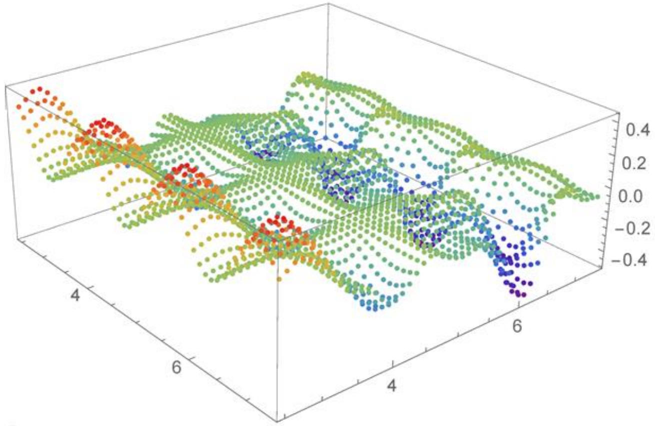
Types of Optimization Landscapes

Single, smooth landscape



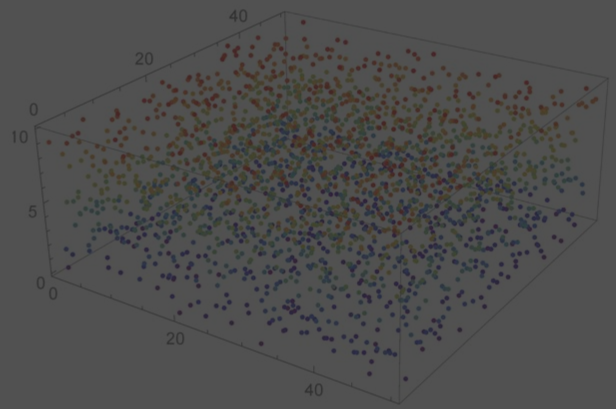
- One single minimum
- Problem is easily „classically“ solved (e.g. with gradient descent)
- QIO with no advantage

Structured, rugged landscape



- Challenge is to avoid getting stuck in any of the sub-optimal local minima
- Scenario where QIO can outperform other techniques

A scattered, random landscape



- No clear relation between cost parameters and cost value
- No algorithm can improve on a brute force search

Specifying Quantum Algorithms

Type of Quantum Computing

(Universal) Quantum Computing

Quantum Optimiziation (Annealing)

Algorithm Specification

Gate Circuit

describes the sequence of gates to be applied

Cost Function

describes a function the system should find a minimum for

```python
from azure.quantum import Workspace
from azure.quantum.optimization import Problem, ProblemType, Term
from azure.quantum.optimization import ParallelTempering

workspace = Workspace ( ... )

problem = Problem(name="My Simple Problem", problem_type=ProblemType.ising)
terms = [
    Term(c=-9, indices=[0]),
    Term(c=-3, indices=[1,0]),
    Term(c=5, indices=[2,0])
]

problem.add_terms(terms=terms)

solver = ParallelTempering(workspace, timeout=100)

result = solver.optimize(problem)
print(result)
```

Specifying Quantum Algorithms

DEMO

Cost Function Specification

Cost function:	$C = -9x_0 - 3x_0x_1 + 5x_0x_2$
Terms:	$-9x_0$ $-3x_0x_1$ $5x_0x_2$
Variables:	x_0, x_1 and x_2
Task:	assign values $+1$ and -1 to variables x_0, x_1 and x_2 so that C is minimized.

Solution: $x_0 = 1, x_1 = 1, x_2 = -1$

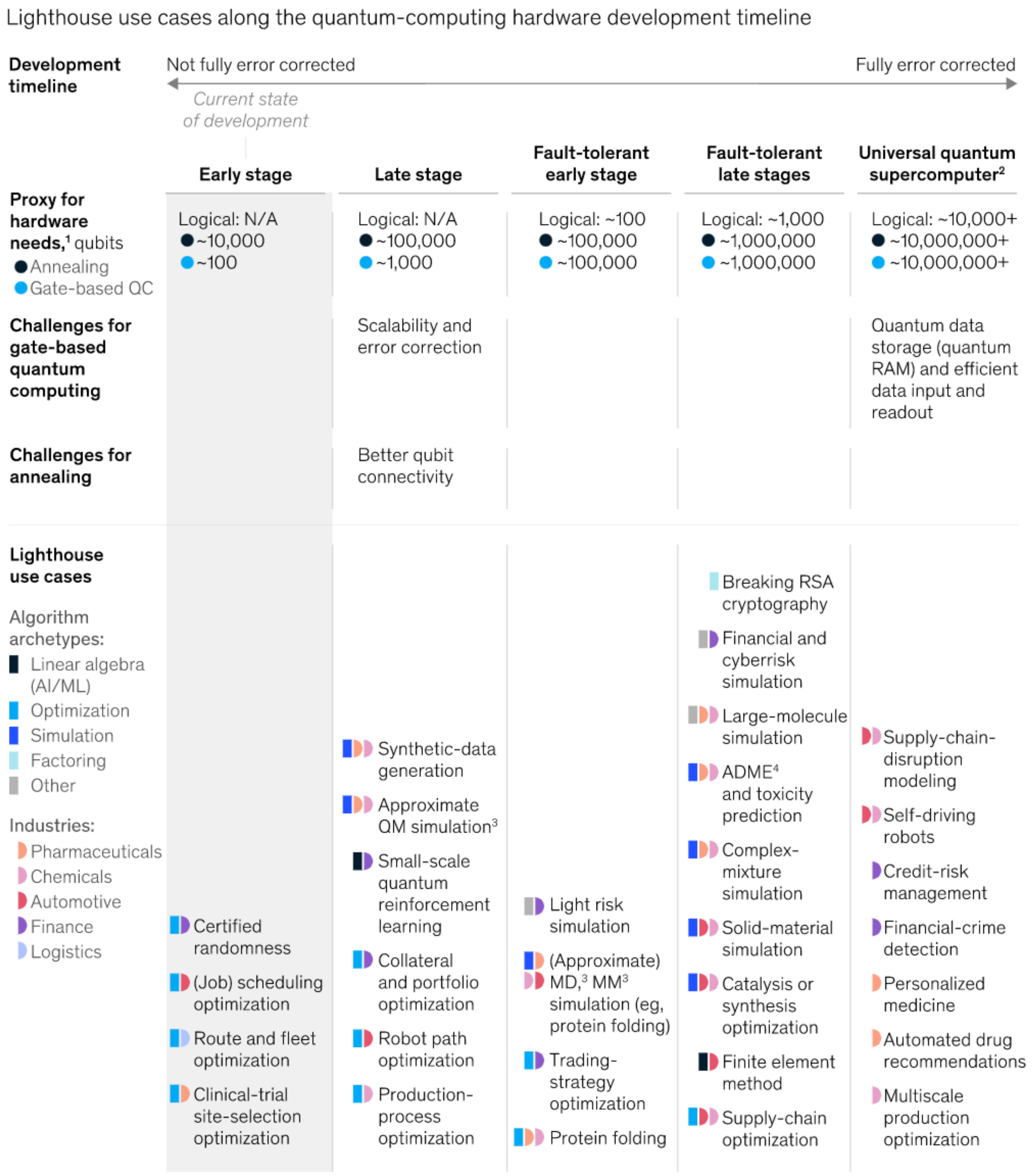
```
from azure.quantum import Workspace
from azure.quantum.optimization import Problem, ProblemType, Term
from azure.quantum.optimization import ParallelTempering

workspace = Workspace ( ... )

problem = Problem(name="My Simple Problem", problem_type=ProblemType.ising)
terms = [
    Term(c=-9, indices=[0]),
    Term(c=-3, indices=[1,0]),
    Term(c=5, indices=[2,0])
]

problem.add_terms(terms=terms)

solver = ParallelTempering(workspace, timeout=100)

result = solver.optimize(problem)
print(result)
```

Use Cases for Quantum Computing

Maturing quantum-computing hardware will make more use cases viable.

Outlook: Fault-tolerant quantum computing is expected between 2025 and 2030, based on announced hardware road maps for gate-based quantum-computing players.

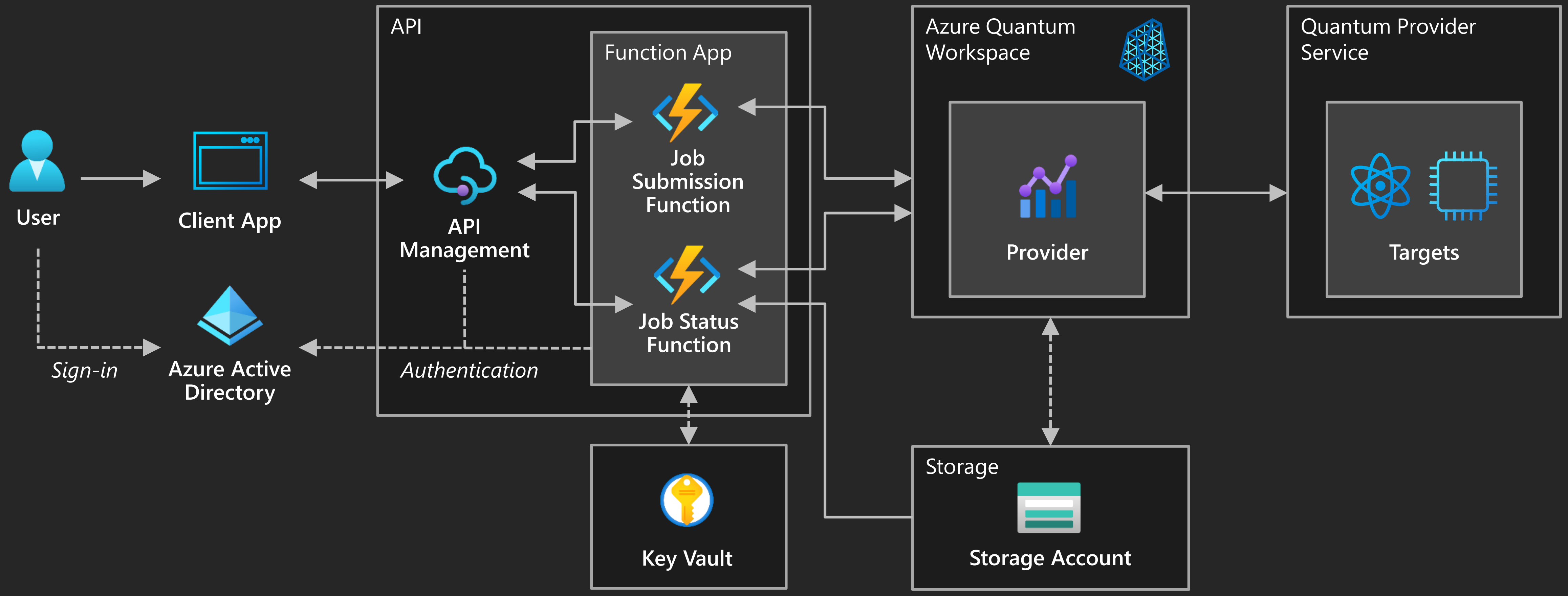
Source:
Quantum computing use cases are getting real—what you need to know
McKinsey, 14/12/2021
https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/quantum-computing-use-cases-are-getting-real-what-you-need-to-know



Microsoft

Exposing Quantum Algorithms via APIs

DEMO



https://docs.microsoft.com/en-us/azure/quantum/how-to-publish-qio-job-as-azurefunction

Quantum-ready starts now

Scaled quantum computing requires planning and investment today

Now to 5 years from now
Setting the quantum table

Use cases focused into cross between quantum and high-performance conventional computing

Business problems meaningfully addressable with quantum-inspired optimization (QIO)

Stakeholders align themselves to ecosystems, technologies, and partners

Preparing for the impact of disruptive technology

Post-quantum cryptography readiness

Greater than 5-year horizon
Enduring industrial advantage unlocked

Significant, global private and public investment into quantum yields scalable quantum systems

Complex industrial use cases unlocked

Organizations that have been planful in their preparation reap decisive early-adopter benefits

What Quantum-ready means for your company

Leaders should take 3 steps now:

1. **Understand the most significant opportunities and risks** in your industry arising as a result of Quantum technologies.

2. **Cultivate a team of quantum-enthusiasts or encourage** existing **innovation and disruptive technology team members** to lean into quantum and bring use cases forward.

3. **Invest in migrating and readying conventional compute architecture and workflows for quantum acceleration.**

Microsoft

Summary

Why do we need a new way of computing?

Some of today's challenges require more compute power than available with classic IT.

Quantum Computing promises exponential speedup for a certain class of use cases.

Quantum algorithms have similar structure

Initialization → state evolution → measurement

Specification of your algorithm depending on the type of Quantum Computing

Universal, gate-based compute → Gate Circuit

Optimization (Annealing) → Cost Function

Getting Started

Options for Getting Hands-on Experience

Azure Free Account

Azure Services free for 12 months.
Start with $200 Azure Azure credit.
https://azure.microsoft.com/en-us/free/

Azure Quantum Credits

$500 per quantum hardware provider with each workspace; additional $10.000 in credits via application.
Intended to help enable research and commercial impact with quantum.
https://aka.ms/aq/credits

Enterprise Acceleration Program

Collaborate with Microsoft experts to develop quantum solutions.
Join the Azure Quantum Network of solution partners, customer, research organizations.
mailto:QBizdesk@microsoft.com

Getting Started

Online Resources

Documentation
[Azure Quantum Documentation]
[Q# User Guide]

Learning
[Quantum Learning Resources]
[MS Learn: Quantum Computing foundations (8 modules)]
[Quantum Katas (self-paced tutorials)]

Samples
[Quantum Development Kit Samples]
[Q# Code Samples]
[Quantum Optimization Samples]

Customer Stories
[Azure Quantum Customer Stories]
[Industry Case Studies]

Blogs
[Microsoft Azure Quantum Blog]
[Q# Blog]