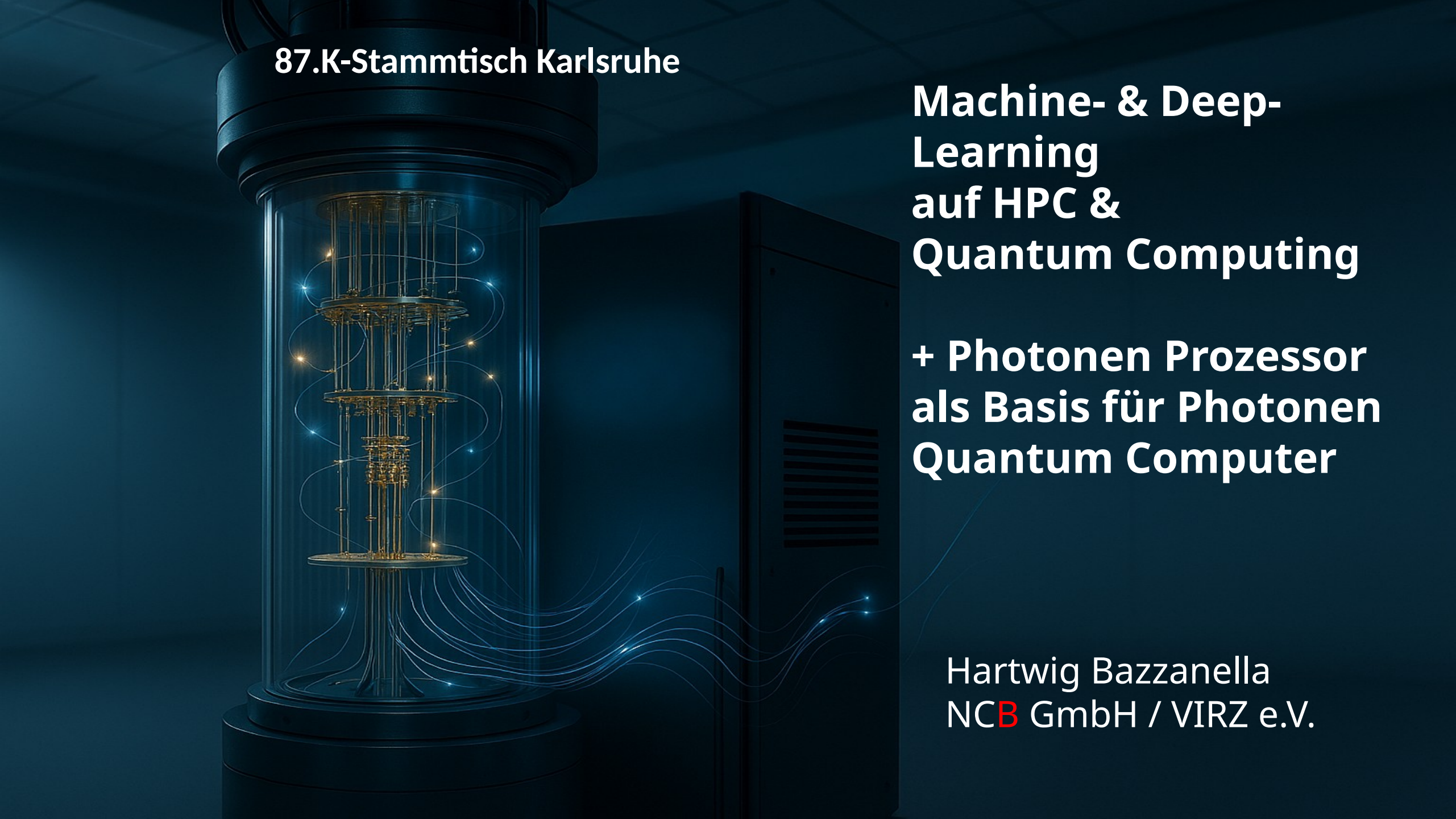




87.K-Stammtisch Karlsruhe

**Machine- & Deep-
Learning
auf HPC &
Quantum Computing**

**+ Photonen Prozessor
als Basis für Photonen
Quantum Computer**

Hartwig Bazzanella
NCB GmbH / VIRZ e.V.

Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning, usw. mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen QC – was ist eigentlich ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning, usw. mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen QC – was ist eigentlich ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

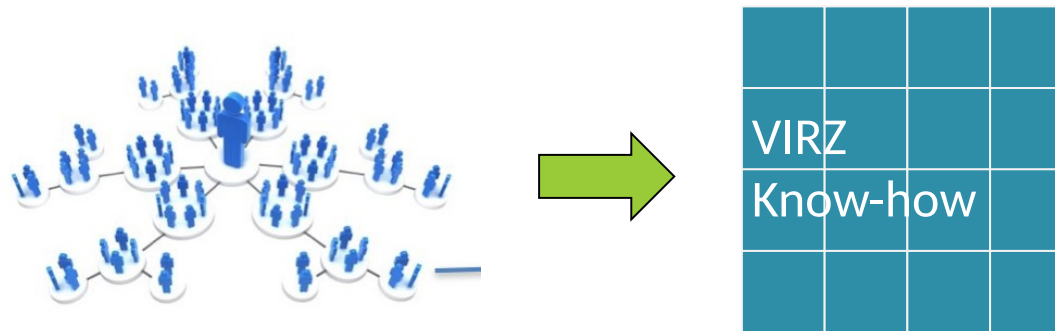


VIRZ e.V.

Verband Innovatives
Rechenzentrum e.V.

Warum VIRZ

- Verband Innovatives RechenZentrum
- Hohes Know-how aller Mitglieder



Warum VIRZ

- VIRZ-Mitglieder des VIRZ setzen sich zusammen aus
- Planer
- Experten
- RZ-Betreiber
- Forschung und Lehre / Öffentliche Hand
- Aktuell: Technische Hochschule Deggendorf



VIRZ -> Quantencomputer am Beispiel IBM



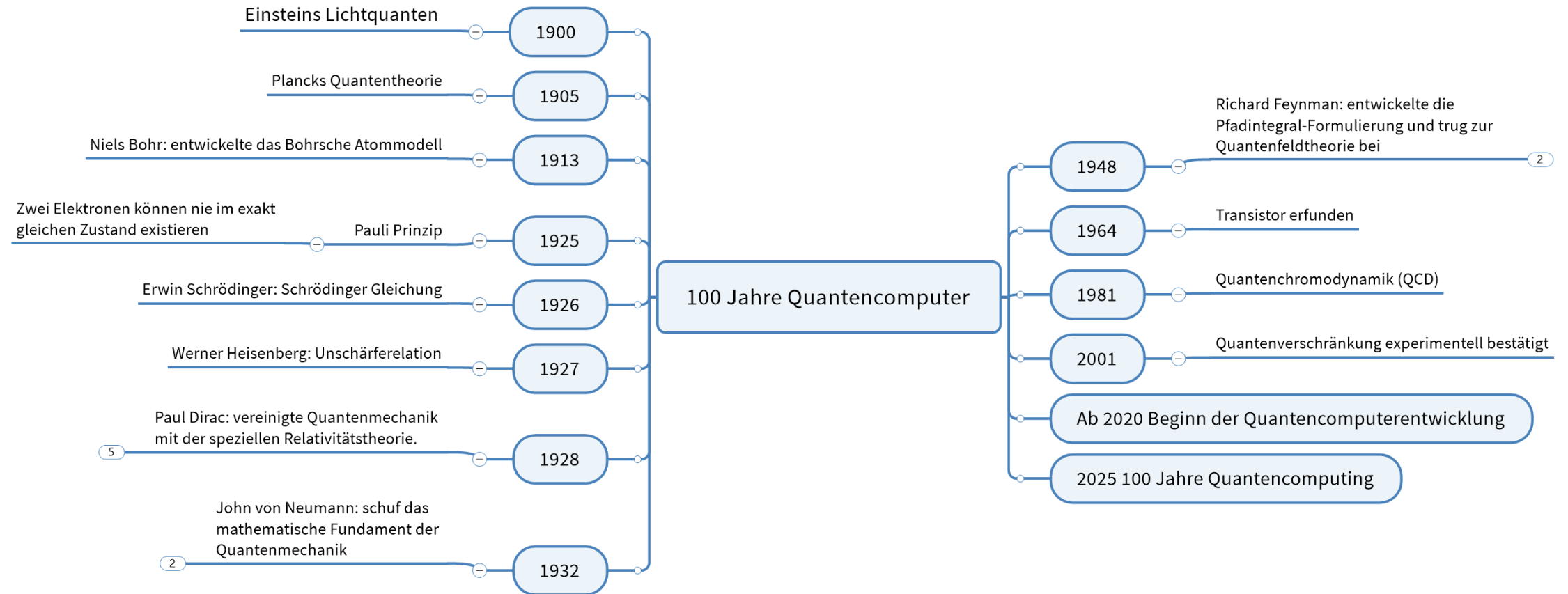
Die prägenden Köpfe und Konzepte

- Niels Bohr: entwickelte das **Bohrsche Atommodell** und später die Kopenhagener Deutung.
- Werner Heisenberg: formulierte **1927 die Unschärferelation**
- Erwin Schrödinger: entwickelte die **Schrödinger-Gleichung**.
- Paul Dirac: **vereinigte Quantenmechanik mit der speziellen Relativitätstheorie**.
- Wolfgang Pauli: führte das **Pauli-Prinzip** ein.
- John von Neumann: schuf das **mathematische Fundament der Quantenmechanik**.
- Richard Feynman: entwickelte die Pfadintegral-Formulierung und **trug zur Quantenfeldtheorie bei**.

Zentrale Entdeckungen und Anwendungen in der Quantentechnology

- **Quantenelektrodynamik** (QED) – die Beschreibung der Wechselwirkung von Licht und Materie.
- **Quantenfeldtheorie** (QFT) – das Standardmodell der Teilchenphysik basiert auf dieser Theorie.
- **Halbleiterphysik** – Grundlage für **Transistoren** und Mikroprozessoren.
- **Supraleitung, Quanten-Hall-Effekt, Bose-Einstein-Kondensate** – exotische makroskopische Quantenzustände.
- **Lasertechnologie, MR-Tomographie, GPS-Korrekturen** – alltägliche Anwendungen quantenphysikalischer Prinzipien.

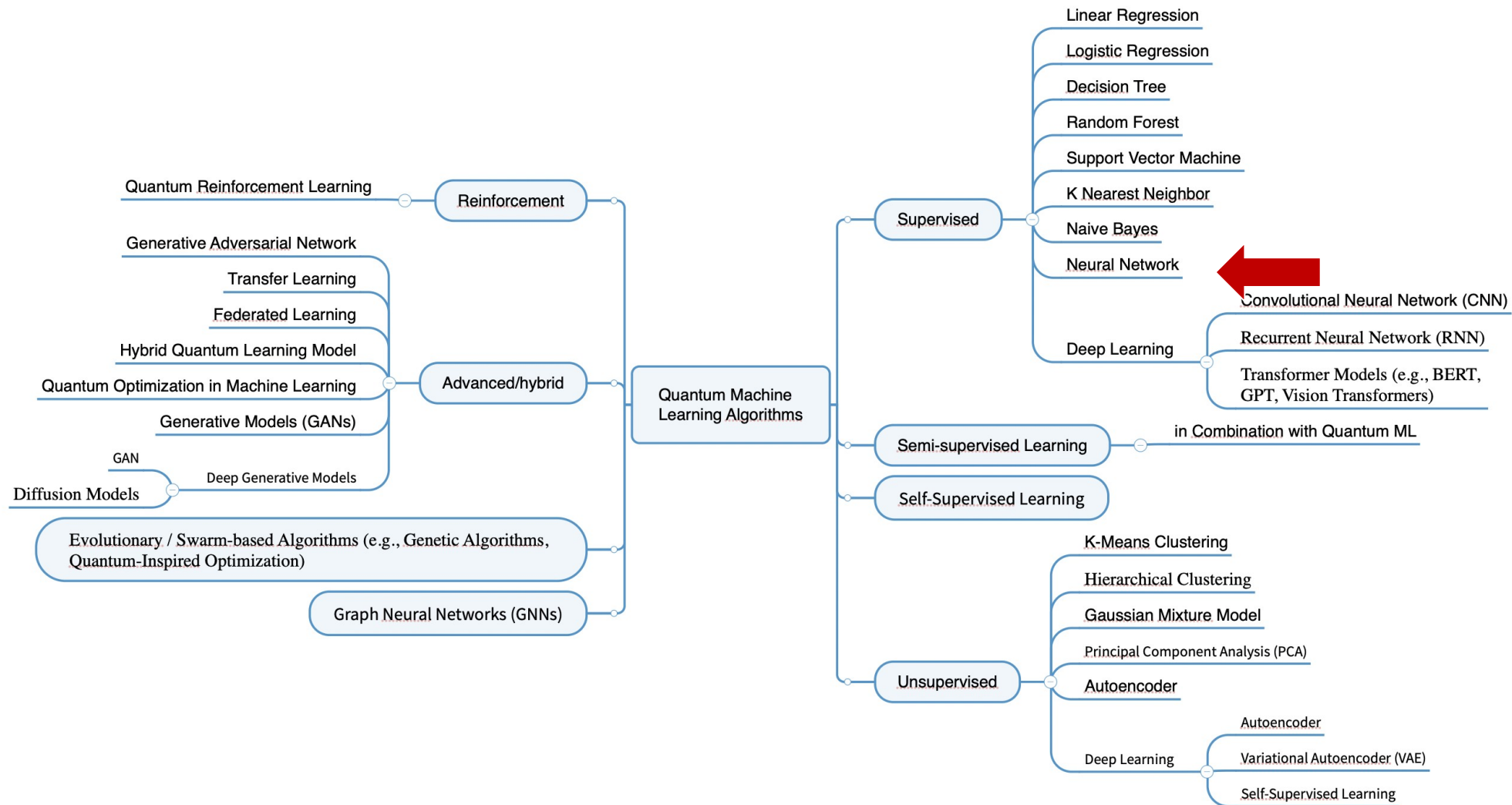
100 Jahre Quantenmechanik



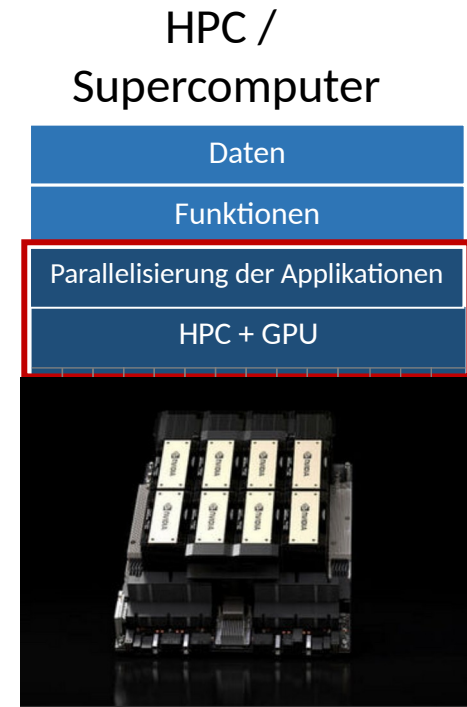
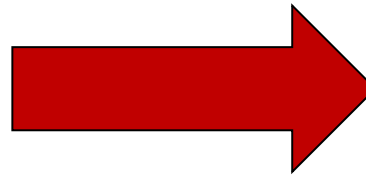
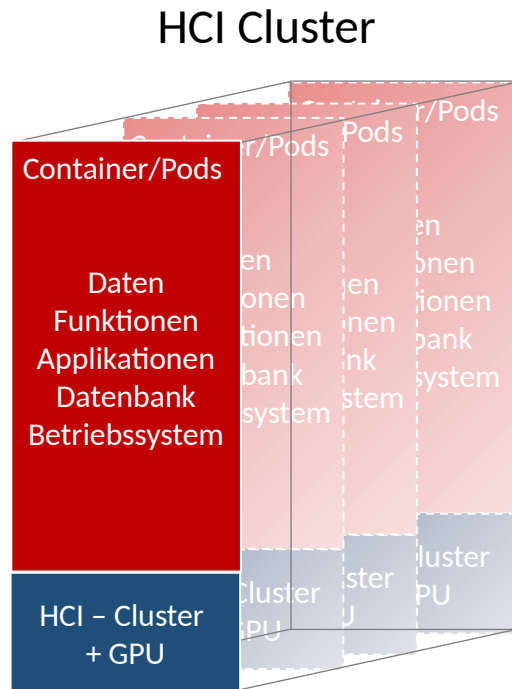
Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning, Neuronale Netze mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen QC – was ist eigentlich ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Quantum Machine Learning Algorithms



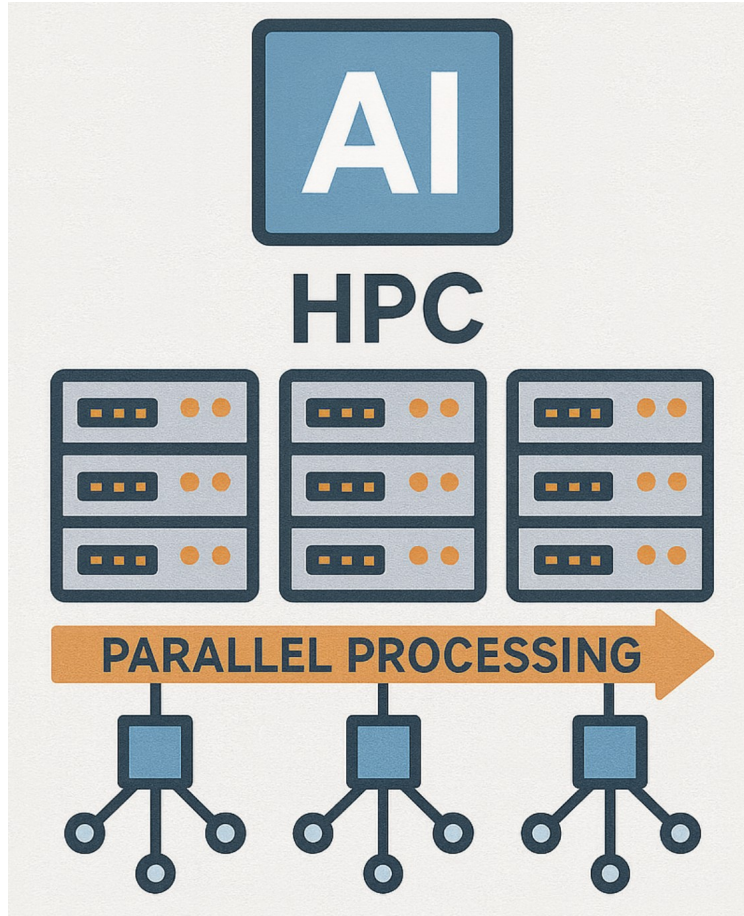
Parallelisierung mit HPC



NVIDIA
GPU-
Hardware

KI wird mit parallel laufenden HPC realisiert

Parallelisierung mit HPC am Bsp. Neuronale Netze



- Oben der Datensatz, der in Batches verteilt wird.
- Die HPC Nodes (GPU/TPU) berechnen die Vorwärts-/Rückwärtsdurchläufe.
- Über Gradient Synchronization (AllReduce) gleichen sie die Ergebnisse ab.
- Unten werden die globalen Modell-Gewichte aktualisiert.

Grundidee Neuronale Netze

- Neuronale Netze (NN, Deep Learning) sind sehr große **Matrizenmultiplikationen** und Vektoroperationen. Jedes Layer (z. B. Dense-, Convolution-, Transformer-Layer) lässt sich als lineare Algebra (Matrix $\times$ Vektor, Tensoroperationen) + nichtlineare Aktivierungsfunktionen beschreiben.
- HPC spezialisiert auf parallele Berechnungen großer Matrizen und Tensoren – ist es eine ideale Plattform für KI/NN-Workloads.

Dafür ist ein riesiger Platzbedarf notwendig!

Siehe Meta mit „RZ als Zelt“



Quelle: Midjourney / KI generiert

Meta arbeitet an mehreren Multi-Gigawatt Cluster für KI-Workloads.
Das erste Groß-RZ Codename „Prometheus“ soll 2026 in Betrieb gehen.

Ein 5 GW-Komplex „Hyperion“ soll folgen – ist so groß wie Manhattan!

Anwendung: KI in allen Facetten!!

Wo sind die Grenzen?

- HPC (klassisch, CPU-Cluster): skaliert linear mit der Anzahl der Knoten, aber **Speicherbandbreite und Kommunikation sind Flaschenhälse**.
- HPC + GPU: GPUs bringen massiv parallele Rechenkerne, also enormen Speed-up speziell bei Matrixmultiplikationen – aber **es bleibt klassisch deterministisch**.
- Hoher Energiebedarf – 50/100/150kW pro Rack

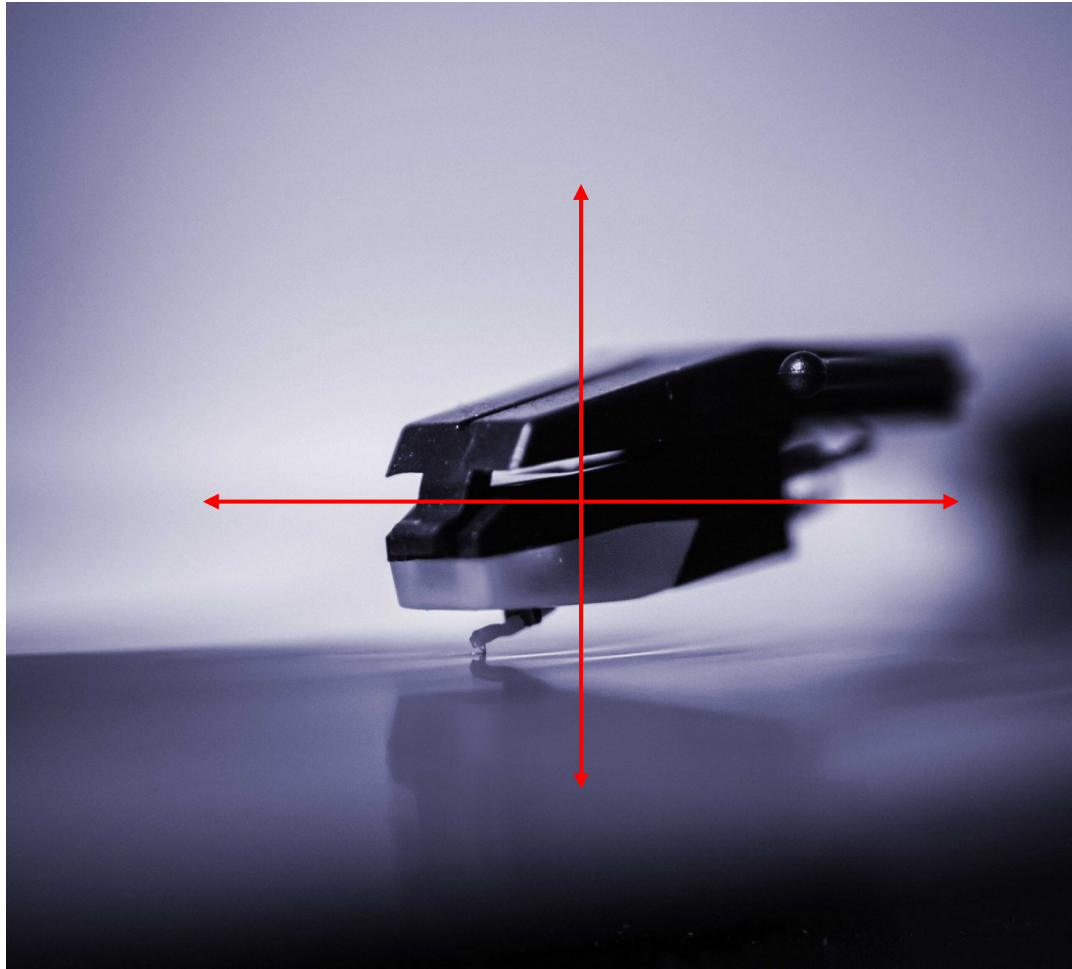
Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning, und Neuronale Netze mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen – was ist ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Heutige Computer

- Heute gängige Computer, Smartphones, Smartwatches oder Taschenrechner haben eines gemeinsam, sie arbeiten mit Bits.
- Ein Bit kann nur zwei Zustände annehmen: entweder eine [1] oder eine [0].

Digital -> Analog

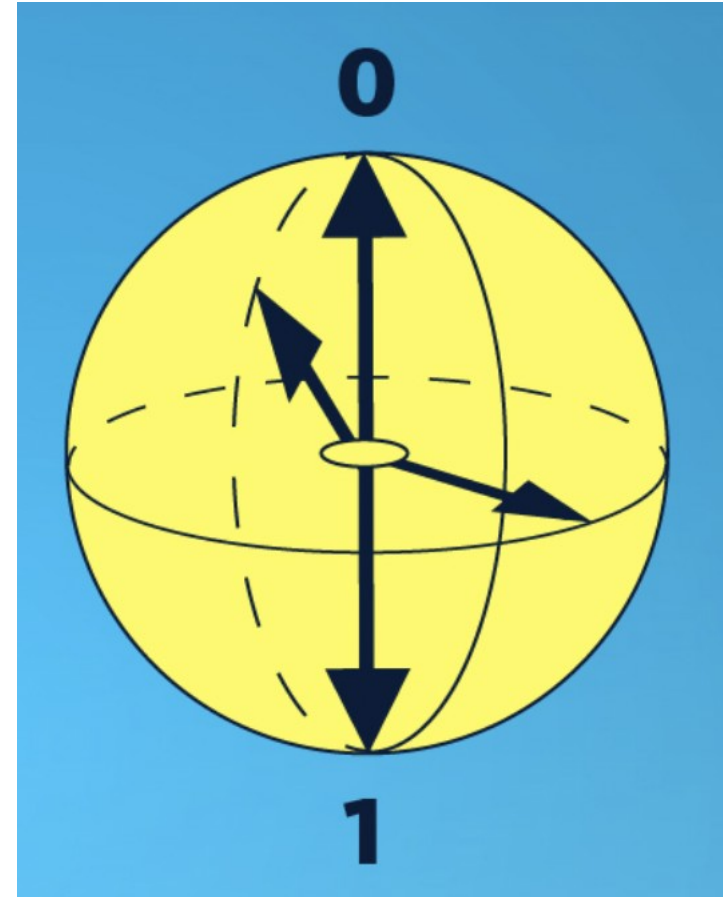


Qubits statt Bits

- Quantencomputer arbeiten mit „Qubits“ = „Quanten-Bit“.
- Wie ein Bit in einem klassischen Computer kann ein Qubit entweder im Zustand [1] oder [0] sein.
- Ein Qubit kann auch **gleichzeitig im Zustand [1] und [0]** sein oder auch in **theoretisch unendlich vielen Zuständen** dazwischen.
- Ein Qubit wäre eine in die Luft geworfene Münze, die sich schnell um sich selbst dreht. Bei ihr kann man nicht sagen, ob Kopf oder Zahl oben ist, sie ist in beiden Zuständen gleichzeitig.

Qubits statt Bits

- Dieses Qubit ist gleichzeitig im Zustand 0, 1 und auch noch in vielen weiteren Zuständen (Superposition)
- Mit zwei Bits kann ein normaler Computer die Zahlen von 0 bis 3 darstellen.
- Die beiden Bits $[0,0]$ ergeben die Zahl 0, mit $[0,1]$ ist die Zahl 1 gemeint. Mit $[1,0]$ die Zahl 2 und mit $[1,1]$ die Zahl 3.
- Die vier Zustandskombinationen, die die Zahlen 0 bis 3 darstellen, können theoretisch also durch nur einen Qubit und zur selben Zeit dargestellt werden.



Exponentieller Vorsprung durch Qubits

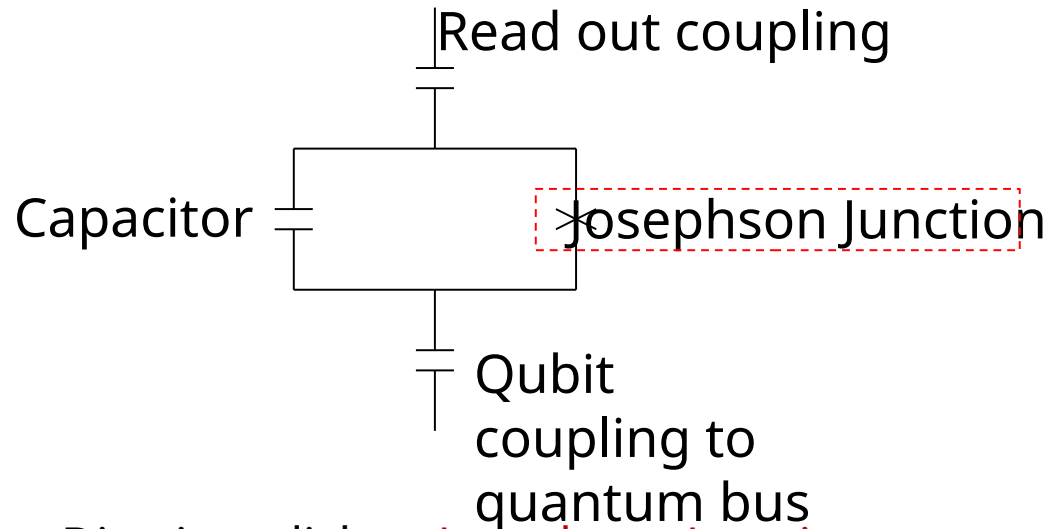
- Doch selbst wenn man von nur zwei verschiedenen Zuständen ausgeht, die ein Qubit gleichzeitig einnehmen kann, **dann verdoppelt jedes dazukommende Qubit die Anzahl der gleichzeitig darstellbaren Zustände.**
- Dieser Vorteil wächst exponentiell:
 - ❖ Drei Qubits können schon 8 Zustände gleichzeitig einnehmen,
 - ❖ **300 Qubits sogar 2^{300} =**
20370359763344860862684456884093781610514683936659362506361
4044935438129976333 6706183397376.
- Das ist eine größere Zahl, als Teilchen im Universum existieren!
- Für einen klassischen Computer ein No GO!

Quantencomputer

- Inneren Aufbau eines IBM-Quantencomputers mit dem typischen „Goldenen Kronleuchter“-Design
- Die **supraleitenden Schaltkreise** – also die **Qubits mit den Josephson-Kontakten** – sind hier allerdings nicht sichtbar. Sie befinden sich:
 - ❖ ganz unten am Ende dieser aufwendigen Kühleinheit,
 - ❖ auf einem **kleinen Chip**, der direkt auf wenige Millikelvin heruntergekühlt wird,
 - ❖ eingebaut in eine **Halterung am tiefsten Punkt** der Verdünnungskryostaten.
- Die vielen goldfarbenen Koaxialleitungen und Filter, die man hier sieht, führen nur die Steuer- und Auslesesignale sowie die Kühlung hinunter.



Josephson-Junctions (supraleitende Spulen / Kontakte)



Die eigentlichen **Josephson-Junctions** (supraleitende Spulen / Kontakte) sind

- winzige Strukturen aus Aluminium oder Niob auf einem Silizium- oder Saphirchip – nur Millimeter groß
- und liegen unterhalb dieser sichtbaren Verkabelung, im Chip am untersten Ende

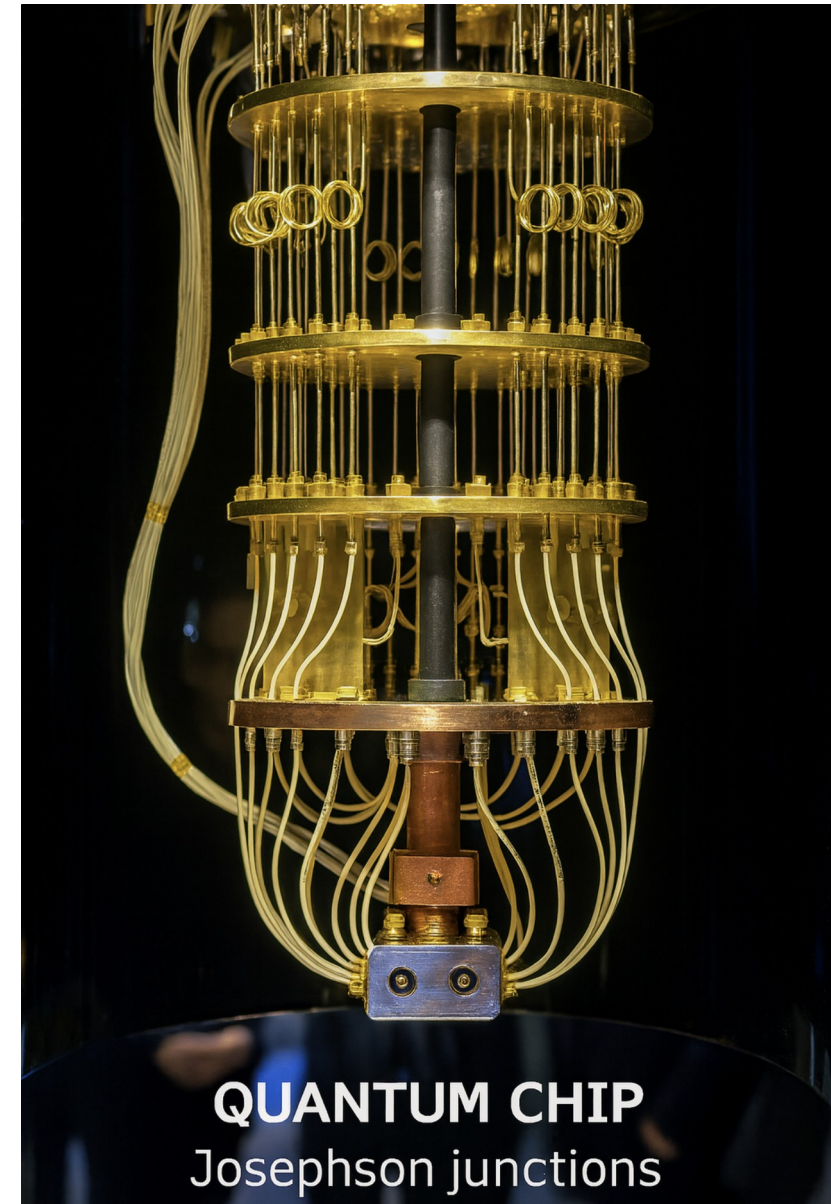
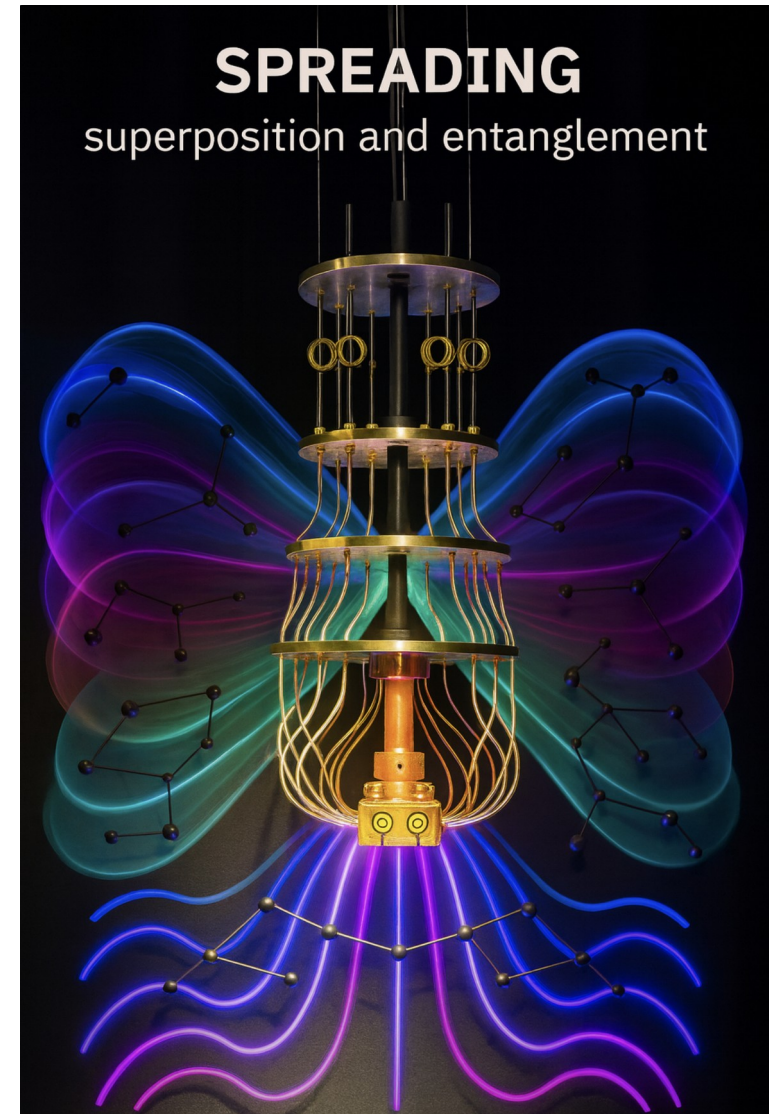


Illustration eines Quantencomputers, bei dem das Konzept der "Spreitung" (Superposition und Verschränkung) visualisiert wird

- Von den Josephson-Kontakten am Chip breiten sich wellenartige, halbtransparente Überlagerungen in verschiedenen Richtungen aus.
- Mehrere Qubits sind durch Linien verbunden, die Verschränkung symbolisieren.
- Farbcodierte Wellenfronten (blau, violett, grün) überlagern sich im Raum, um die gleichzeitige Existenz mehrerer Zustände darzustellen



Programmierung von Quantencomputer

- Man kann das grob mit einem Becher mit Würfeln vergleichen:
 - ❖ Solange der Würfelbecher geschüttelt wird, stoßen die Würfel in ihm immer wieder aneinander und an die Wand des Bechers.
 - ❖ Von außen weiß man aber nicht, in welchem Zustand sie in jedem Augenblick sind.
- Erst wenn man den Würfelbecher umdreht und die Würfel auf der Unterlage liegen bleiben, kann man das Ergebnis der Augenzahlen „auslesen“.
- Beim Würfeln beeinflussen sich die Würfel unkontrolliert, das Ergebnis ist zufällig.
- Beim Quantencomputer ist dagegen die Kunst, die vielen Qubits mit ihren Wechselwirkungen so anzustoßen, dass am Ende kein zufälliges, sondern ein sinnvolles Ergebnis ausgelesen wird.
- Der Haken: Sobald ein Ergebnis der Qubits abgefragt wird, ändert sich sein Zustand!

Kohärenzzeit

- Nutzung von Quantenzustände des Lichts.
- Einzelne Photonen müssen gezielt erzeugt, manipuliert und zur Wechselwirkung gebracht werden.
- Mit Folgen von
 - ❖ Strahlteilern und
 - ❖ Phasenschieberwerden hochgradig verschränkte Zustände einzelner Photonen erzeugt werden, an denen dann gezielte, problemspezifische Messungen vorgenommen werden.
- Die Kohärenzzeit ist ein Maß für die Dauer, die ein Qubit-Zustand für das Quantencomputing nutzbar bleibt, also noch nicht durch störende Einflüsse aus der Umgebung verändert wurde.

Rechnen mit Überlichtgeschwindigkeit



Klassische Daten vorbereiten

- Eingaben (z. B. Zahlen, Matrizen, Bitstrings) liegen als **klassische Bits** vor.
- Damit sie auf einem Quantencomputer verarbeitet werden können, musst du sie in **Qubit-Zustände** übersetzen.
- Beispiel: $0 \rightarrow |0\rangle$, $1 \rightarrow |1\rangle$.
- Mehrdimensionale Daten (z. B. Vektoren, Matrizen, Bilder) müssen oft **kodiert** werden (Amplitude Encoding, Basis Encoding, Angle Encoding).

Quantenkodierung (Encoding)

Es gibt verschiedene Methoden, digitale Daten auf Qubits zu bringen:

- **Basis Encoding**: Jedes Bit wird 1:1 auf ein Qubit abgebildet.
- **Amplitude Encoding**: Daten werden in die Wahrscheinlichkeitsamplituden geschrieben → sehr speichereffizient, aber schwer vorzubereiten.
- **Angle Encoding**: Werte werden in Rotationswinkel von Qubits übersetzt.

Messung (Measurement)

- Am Ende werden die Qubits gemessen → sie **kollabieren** in klassische Bits.
- Das Ergebnis sind wieder digitale Daten (0/1-Sequenzen), die klassisch interpretiert werden müssen (z. B. Wahrscheinlichkeit, Vektor, Optimierungslösung).

Klassisches Postprocessing

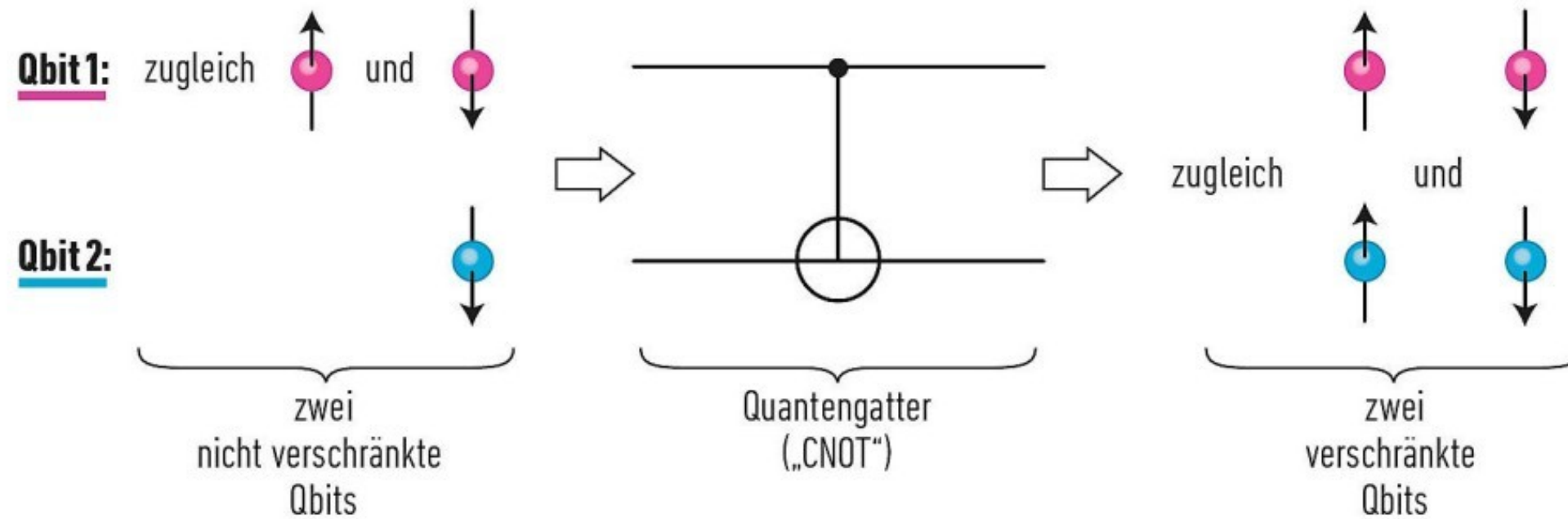
- Da Quantencomputer probabilistisch arbeiten, müssen viele Läufe (Shots) durchgeführt werden, um die Wahrscheinlichkeiten zu schätzen.
- Danach werden die Ergebnisse mit klassischer Software ausgewertet, d.h. im klassischen HPC

Quantenalgorithmus anwenden

- Mit **Quantengattern** (Unitary Operations) werden die Qubits bearbeitet.
- Algorithmen wie Shor, Grover, HHL oder QML-Ansätze arbeiten direkt auf diesem Zustand.
- Währenddessen **bleiben die Daten in Superposition** und können durch Interferenz und Verschränkung verarbeitet werden

Programmierung von Quantencomputer

Im Quantencomputer



F.A.Z.-Grafik Kaiser

IBM Quantum composer

IBM Quantum Composer

My first circuit | File | Edit | View

Visualizations seed 7918 | Sign in to run your circuit

Operations

Search

q[0]

q[1]

c2

OpenQASM 2.0

Open in Quantum Lab

```
1 OPENQASM 2.0;
2 include "qelib1.inc";
3 qreg q[2];
4 creg c[2];
```

Probabilities

Q-sphere

Probability (%)

Computational basis states

Phase

$\pi/2$

π

$3\pi/2$

0

Gates und Operationen
manipulieren Qubits.

Dies sind die building blocks der
Quantum circuits
= Quantenschaltung

IBM Quantum composer

The screenshot displays the IBM Quantum Composer web interface. The top navigation bar includes the 'IBM Quantum Composer' title, search, help, and user profile icons, along with a 'Sign in' button. Below the navigation bar, the 'My first circuit' tab is active, showing a workspace with 'File', 'Edit', and 'View' menus. The 'Visualizations seed' is set to 7918, and a 'Sign in to run your circuit' button is present.

The main workspace is divided into several sections. On the left, the 'Operations' panel shows a search bar and a grid of quantum gates including Hadamard (H), CNOT, T, S, Z, T†, S†, P, RZ, |0⟩, |1⟩, if, √X, √X†, Y, RX, RY, RXX, RZZ, U, RCCX, and RC3X. The central area contains a quantum circuit diagram with three qubits: q[0], q[1], and c2. These labels are enclosed in a red rectangular box, and a red arrow points from the text 'Qubits q(0) und q(2)' to this box. To the right of the circuit diagram, the 'OpenQASM 2.0' code editor is visible, showing the following code:

```
1 OPENQASM 2.0;
2 include "qelib1.inc";
3 qreg q[2];
4 creg c[2];
```

Below the circuit diagram, the 'Probabilities' panel shows a graph of 'Probability (%)' versus 'Computational basis states'. The 'Q-sphere' panel displays a Bloch sphere with a blue dot at the top pole, labeled |00⟩. A phase dial at the bottom left indicates the phase angle, with markers at 0, π/2, π, and 3π/2. A legend at the bottom right shows 'State' (checked) and 'Phase angle' (unchecked).

IBM Quantum composer

The screenshot displays the IBM Quantum Composer web application. The interface includes a top navigation bar with a search icon, a help icon, a user profile icon, and a 'Sign in' button. Below the navigation bar, there's a section for 'My first circuit' with tabs for 'File', 'Edit', and 'View'. The main workspace is divided into several panels:

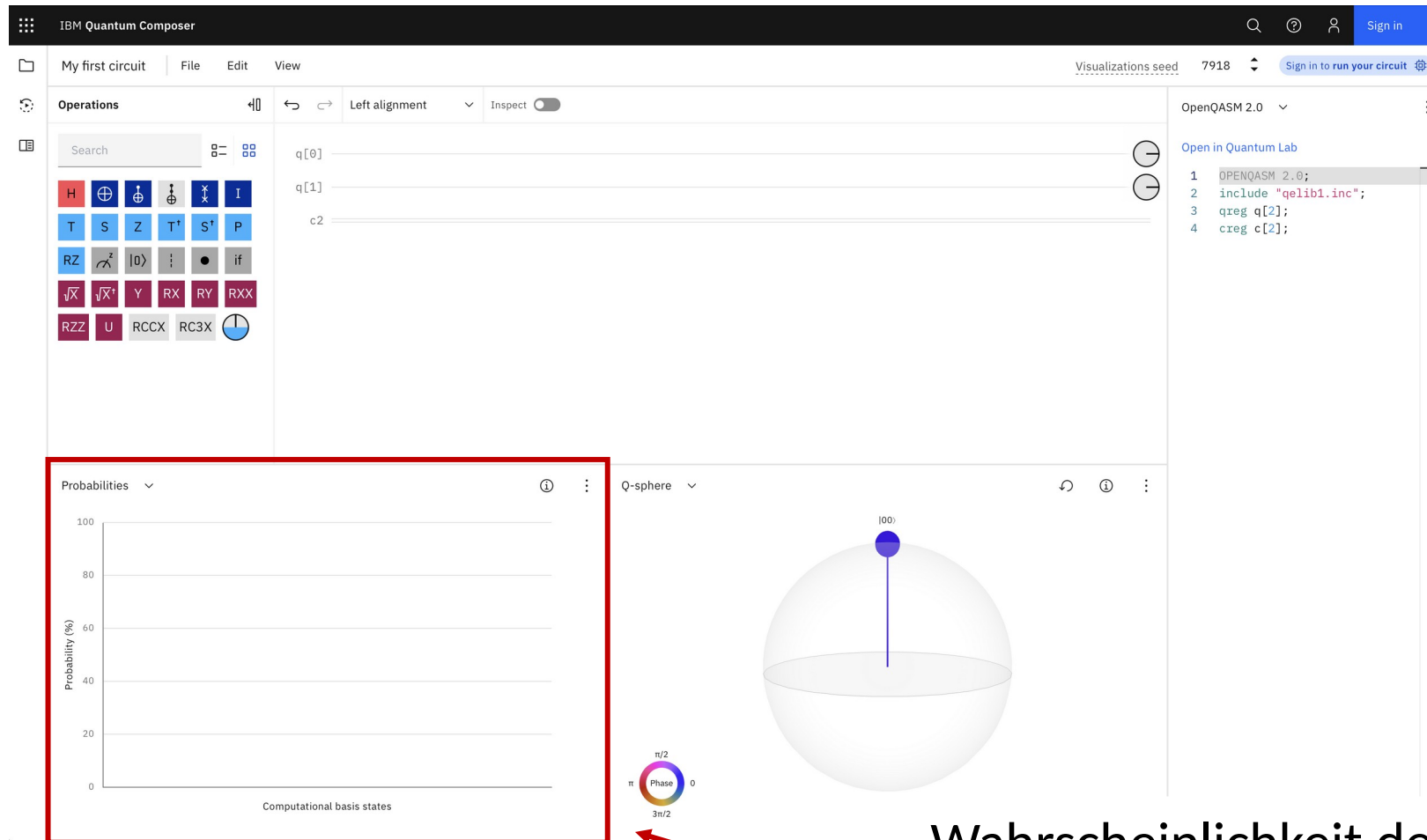
- Operations Panel:** A grid of quantum gates and operations including Hadamard (H), CNOT, T, S, Z, T†, S†, P, RZ, |0⟩, if, √X, √X†, Y, RX, RY, RXX, RZZ, U, RCCX, and RC3X.
- Circuit Editor:** A workspace with qubits q[0], q[1], and a classical register c2.
- Probabilities Panel:** A line graph showing the probability of computational basis states.
- Q-sphere Panel:** A 3D Bloch sphere visualization showing the state of a qubit, currently at the |00⟩ state.
- OpenQASM 2.0 Code Editor:** A text area on the right containing the following code:

```
1 OPENQASM 2.0;  
2 include "qelib1.inc";  
3 qreg q[2];  
4 creg c[2];
```

A red box highlights the OpenQASM 2.0 code editor, and a red arrow points from the text 'Programm Code' to it.

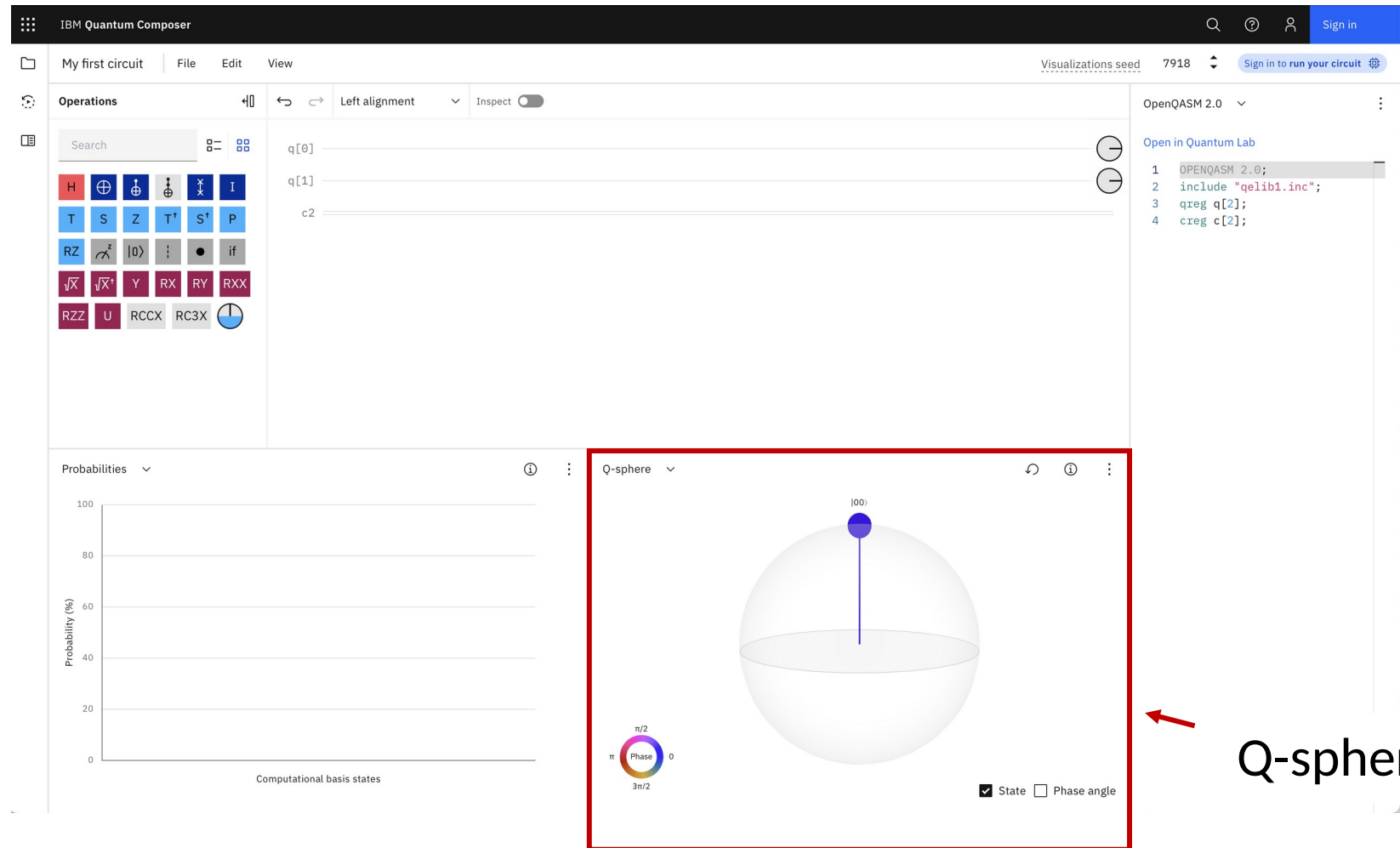
Programm
Code

IBM Quantum composer



Wahrscheinlichkeit der
Ergebnisse

IBM Quantum composer



Beispiel digitale Addition 2

+ 1 = 3

- Klassische Eingabe

- ❖ Zahl A = 2

- ❖ Zahl B = 1

- Ergebnis soll sein: 3

- Binärdarstellung

- ❖ $2 = 10$

- ❖ $1 = 01$

- ❖ $3 = 11$

Quanten-Schaltkreis (Qiskit)

Wir machen das mit einem kleinen „Halbaddierer“:

- Die Addition zweier Bits A und B ergibt Summe und Carry.
- Für einfache Fälle nutzen wir die Qiskit-Gatter: CNOT und CCX (Toffoli).

```
from qiskit import QuantumCircuit, Aer, execute

# Wir brauchen 3 Qubits (2 Eingaben + 1 Carry) und 3 klassische Bits
qc = QuantumCircuit(3, 3)

# Zahl A = 2 (binär 10) → wir setzen das erste Qubit auf 1
qc.x(0) # A = 1 auf Qubit 0

# Zahl B = 1 (binär 01) → wir setzen das zweite Qubit auf 1
qc.x(1) # B = 1 auf Qubit 1

# Volladdierer:
# Summe = A XOR B
qc.cx(0, 2)
qc.cx(1, 2)

# Carry = A AND B
qc.ccx(0, 1, 2)

# Messen
qc.measure([0,1,2], [0,1,2])

# Simulation
backend = Aer.get_backend('qasm_simulator')
job = execute(qc, backend, shots=1000)
result = job.result()
print(result.get_counts())
```

Ausgabe

```
{'011': 1000}
```

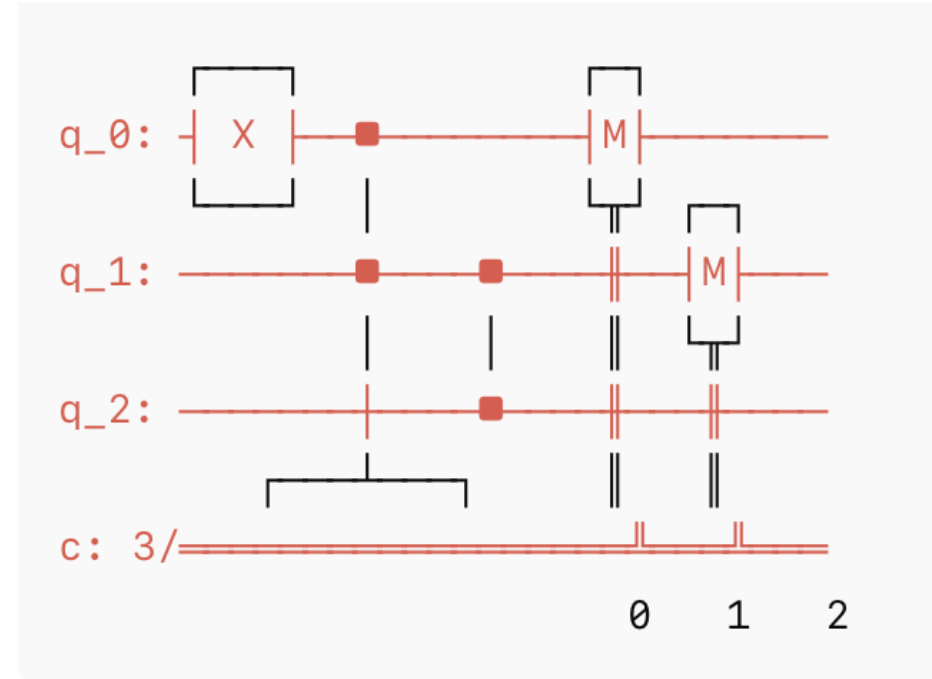
👉 Hier bedeutet 011:

- Qubit 0 = 1
- Qubit 1 = 1
- Carry = 0

→ also die Zahl 3.

Quanten-Schaltkreis (Qiskit)

- Q_0 repräsentiert Zahl A (Teil von 2 -> Bit „1“)
- Q_1 repräsentiert Zahl B (Teil von 1 -> Bit „1“)
- Q_2 ist das Carry-QuBit
- Die X-Gatter setzen die Ergebnisbits (digitale „1“)
- CX (CNOT) und CCX implementieren die Additionslogic
- Am Ende messen wir die klassischen Bits c0, c1, c3
- Das Messergebnis ist 011 (binär 3)



Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning und Neuronale Netze mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Grundlagen – was ist ein QuBit?
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Neuer „photonischer“ Weg

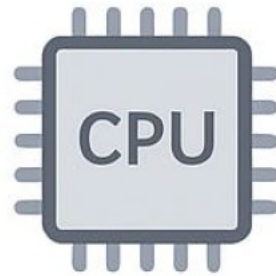
- Übergang von klassischen digitalen Daten zu einem **Photonen Prozessor** und dort Berechnung wie im HPC
- Übergang vom **Photonenergebnis** zur **Photonen-Quantenrepräsentation** (gleiches Medium - Photonen)
- Quantenberechnung
- Übergang zu einem klassischen Ergebnis



Vergleich CPU, Photonic Prozessor und Quantencomputer

- Photonenprozessor
 - ❖ führt Berechnungen mithilfe von Licht durch
 - ❖ Er kann für Matrixmultiplikationen eingesetzt werden
 - ❖ Alle Operationen erfolgen dabei gleichzeitig in einem optischen Netzwerk
- Quantencomputer
 - ❖ Nutzt die Gesetze der Quantenmechanik direkt als Rechenprinzip,
 - ❖ um bestimmt Probleme grundlegend anders (oft exponentiell schneller) zu lösen.

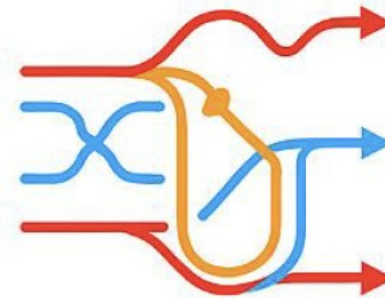
Classical
CPU



$0 \rightarrow 1$
 $1 \rightarrow 0$

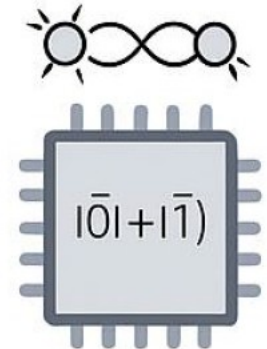
Electrical
signals

Photonic
Processor



Light

Quantum
Computer



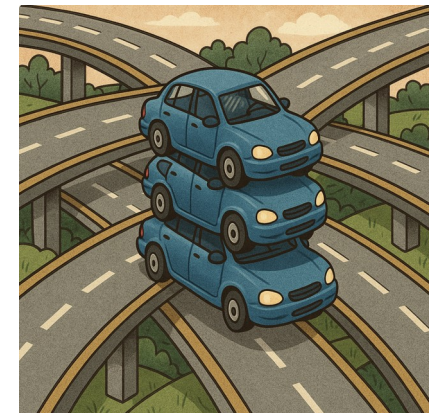
Qubits

Wie funktioniert ein Photonenprozessor

- CPU: Bisherige Technik mit 0/1
- Nutzt Licht nur als schnelleren „Kabelersatz“ und Rechenmedium für klassische Berechnungen.
- Er kann **keine** quantenmechanischen Effekte wie **Überlagerung** oder **Verschränkung** zur Berechnung verwenden.
- **Kodiert Daten in Intensität, Phase oder Polarisation**
- Führt Berechnungen über Interferenz und Modulation durch
- Besonders effizient bei Matrixoperationen (z. B. für KI)
- Arbeitet deterministisch, keine Quantenüberlagerung

Beispiel

- Photonenprozessor
 - ❖ Vierspurige Autobahn: auf jeder Spur fährt eine eigene Lichtfarbe (Frequenz)
 - ❖ Alles fährt klassisch, nur viel schneller und mit mehr Spuren.
- Quantencomputer
 - ❖ Beim Quantencomputer kann das Auto gleichzeitig auf allen Straßen eines Landes fahren
 - ❖ „Weiß“ am Ende, welche Route die beste war – das ist Quantenparallelität.



Beispiel

- Typische Paradebeispiel ist die **Matrixmultiplikation** für neuronale Netze, weil photonische Prozessoren hier ihre Stärken voll ausspielen.
- Neuronale Netze, Bildverarbeitung und viele HPC-Aufgaben bestehen zu großen Teilen aus der Operation:

$$C = A \times B$$

wobei **A** und **B** Matrizen sind, z. B. mit Gewichten und Eingabedaten.

- Ein photonischer Prozessor kann diese Multiplikation in einem einzigen optischen Durchlauf umsetzen, weil:
 - ❖ Die Werte in den Matrizen als **Lichtintensität** oder **Phasenverschiebung** codiert werden.
 - ❖ Interferometer-Netzwerke (z. B. Mach-Zehnder) die Multiplikation und Addition physikalisch analog durchführen.
 - ❖ **Mehrere Wellenlängen gleichzeitig verarbeitet werden (parallel).**

Beispiel Matrixmultiplikation

■ Eingabematrizen

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

■ Klassische Berechnung

$$C = \begin{bmatrix} 1 \cdot 5 + 2 \cdot 7 & 1 \cdot 6 + 2 \cdot 8 \\ 3 \cdot 5 + 4 \cdot 7 & 3 \cdot 6 + 4 \cdot 8 \end{bmatrix} = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

Beispiel Matrixmultiplikation

■ Kodierung:

- ❖ Die Elemente von **A** werden als feste Phasen- oder Amplitudenwerte in einem Interferometer-Netzwerk gespeichert.
- ❖ Die Elemente von **B** werden als Intensitäten verschiedener Eingangskanäle eingespeist.

■ Optische Berechnung:

- ❖ Lichtstrahlen interferieren in einer Gitterstruktur von Mach-Zehnder-Interferometern.
- ❖ Additionen passieren durch Überlagerung von Lichtwellen.
- ❖ Multiplikationen passieren durch Modulation der Amplitude.

■ Auslesen:

- ❖ Am Ausgang misst man die Lichtintensität → diese entspricht den Werten von **C**.

Beispiel Matrixmultiplikation - Warum ist das so schnell?

- Alle Multiplikationen und Additionen passieren gleichzeitig, weil Licht sich gleichzeitig durch alle Pfade bewegt. (Nicht Schritt für Schritt)
- Die Rechenzeit hängt nur von der Laufzeit des Lichts im Chip ab (Nanosekundenbereich), nicht von der Anzahl der Multiplikationen wie bei einer klassischen CPU.
- Energieverbrauch deutlich geringer als bei GPUs.

Beispiel

Matrixmultiplikation

- Optische Schaltung aus Mach-Zehnder-Interferometern (MZI), die Matrixelemente als Phasen- oder Amplitudenwerte im Licht kodieren.
 - **Links**: Eingabesignale (hier die Werte der Matrix B)
 - **Mitte**: Optisches Netzwerk (kodiert die Matrix A)
 - **Rechts**: Ausgangssignale = Ergebnis der Multiplikation $C = A \times B$
- **Kreise mit Zahlen** → **Eingabewerte** z. B. Lichtintensität proportional zum Zahlenwert.
- **Quadrate mit „MZI“** → **Mach-Zehnder-Interferometer**, dienen zum Skalieren und Kombinieren der Signale.
- **Linien** → **Optische Wellenleiter**, über die Lichtsignale laufen.

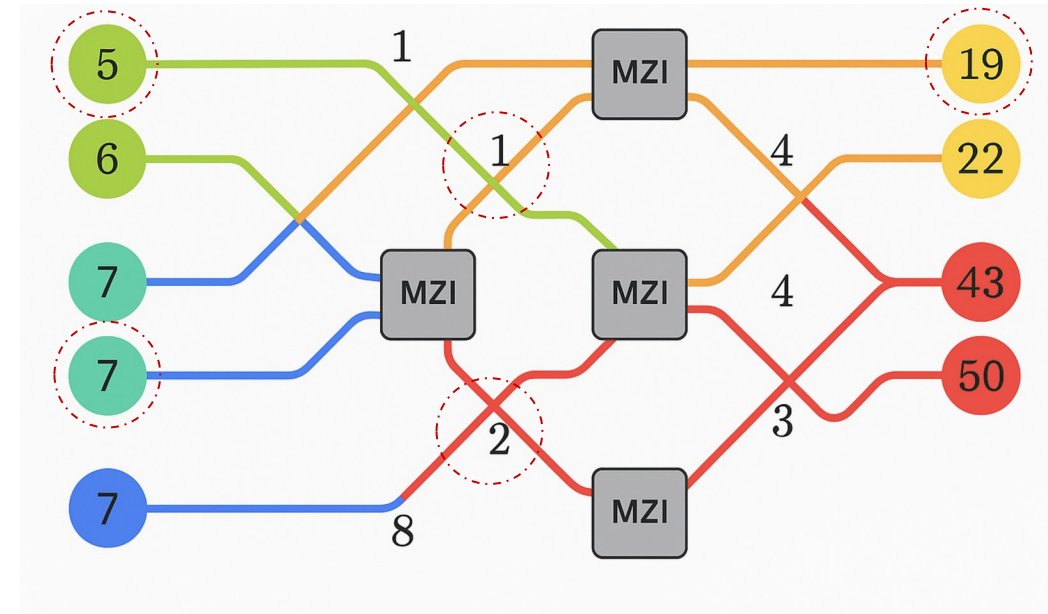
Beispiel Matrixmultiplikation

-
- Ablauf am Beispiel, wir wollen:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

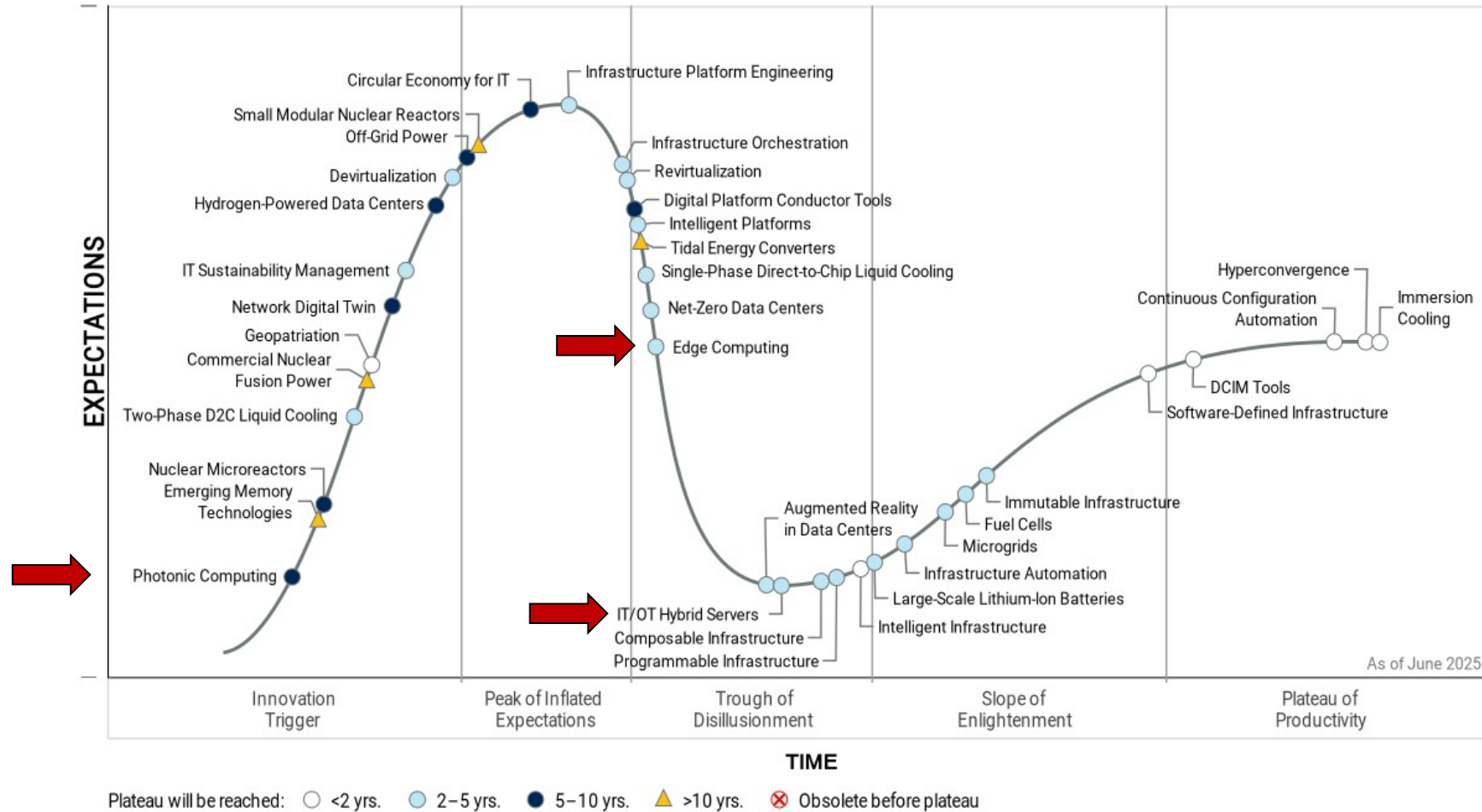
- Das optische Netzwerk (Mach-Zehnder-Interferometer) ist so programmiert, dass es die **Werte von A als Phasen/Amplituden** kodiert.
- Die Eingänge werden mit den **Werten von B als Lichtintensitäten** gespeist.
- Interferenz und Überlagerung der Lichtstrahlen ergeben am Ausgang rechts die Einträge von C mit den Summen 19,22,43,50 → genau das Produkt C.

Ablauf mit erste Zeile Matrix A
1,2 und erste Spalte von
Matrix B 5,7



Was sagt Gardner 2025

Hype Cycle for Data Center Infrastructure Technologies, 2025



Gartner

Korrekte Werte für jeden Pfad

Ergebnis C_{ij}	Zeile von A	Spalte von B	Rechnung
C_{11}	[1, 2]	[5, 7]	$1 \cdot 5 + 2 \cdot 7 = 19$
C_{12}	[1, 2]	[6, 8]	$1 \cdot 6 + 2 \cdot 8 = 22$
C_{21}	[3, 4]	[5, 7]	$3 \cdot 5 + 4 \cdot 7 = 43$
C_{22}	[3, 4]	[6, 8]	$3 \cdot 6 + 4 \cdot 8 = 50$

Rechnung

$\begin{bmatrix} 1, & 2 \end{bmatrix} \quad \begin{bmatrix} 5, & 7 \end{bmatrix}$

$\begin{bmatrix} 1, & 2 \end{bmatrix} \quad \begin{bmatrix} 6, & 8 \end{bmatrix}$

$\begin{bmatrix} 3, & 4 \end{bmatrix} \quad \begin{bmatrix} 5, & 7 \end{bmatrix}$

$\begin{bmatrix} 3, & 4 \end{bmatrix} \quad \begin{bmatrix} 6, & 8 \end{bmatrix}$

Ergebnis C_{ij}	Zeile von A	Spalte von B	Rechnung
C_{11}	[1, 2]	[5, 7]	$1 \cdot 5 + 2 \cdot 7 = 19$
C_{12}	[1, 2]	[6, 8]	$1 \cdot 6 + 2 \cdot 8 = 22$
C_{21}	[3, 4]	[5, 7]	$3 \cdot 5 + 4 \cdot 7 = 43$
C_{22}	[3, 4]	[6, 8]	$3 \cdot 6 + 4 \cdot 8 = 50$

Umsetzung in der optischen Schaltung

Eingangskanäle (links):

- Obere Eingangsleitung = erstes Element der Spalte von **B**
- Untere Eingangsleitung = zweites Element der Spalte von **B**

MZI-Netzwerk (Mitte):

- Enthält die festen Gewichtungen aus **A**
- Steuert, wie stark ein Eingangssignal zum jeweiligen Ausgang geleitet wird

Ausgänge (rechts):

- Vier Ausgänge liefern direkt $C_{11}, C_{12}, C_{21}, C_{22}$ in Lichtintensitäten, die proportional zu den Zahlenwerten sind

Zusammenfassung Photonprozessor vs. Quantencomputer

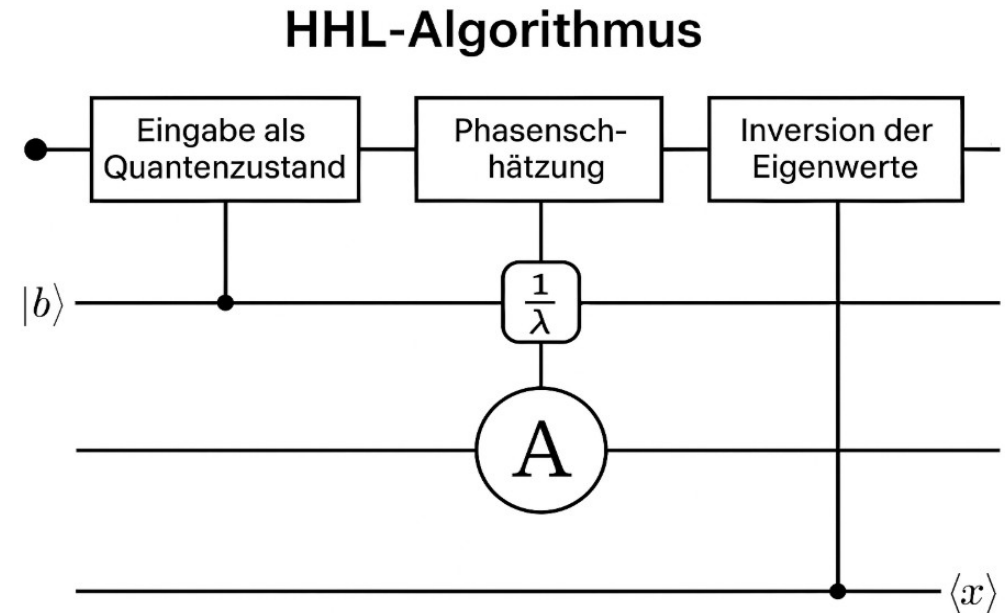
- Photonprozessor:
 - ❖ Klassische Berechnung mit Licht
 - ❖ Sehr schnell bei parallelen Operationen
 - ❖ Keine Quantenüberlagerung oder Verschränkung
- Quantencomputer:
 - ❖ Nutzt Qubits (Superposition, Verschränkung)
 - ❖ Exponentielle Parallelität bei bestimmten Problemen
 - ❖ Noch stark experimentell, hohe Fehlerraten

Wo sind die Grenzen?

- HPC (klassisch, CPU-Cluster): skaliert linear mit der Anzahl der Knoten, aber **Speicherbandbreite und Kommunikation sind Flaschenhälse**.
- HPC + GPU: GPUs bringen massiv parallele Rechenkerne, also enormen Speed-up speziell bei Matrixmultiplikationen – aber **es bleibt klassisch deterministisch**.
- Photonencomputer: kann Matrixoperationen (z. B. Fourier- oder unitäre Transformationen) hardware-intrinsisch in Lichtgeschwindigkeit parallelisieren. Vorteil: **extrem energieeffizient, große Matrizen in $O(1)$ -Zeit für bestimmte Operationen**.
- Quantencomputer: bringt bei speziellen Problemen einen exponentiellen Vorteil – aber nicht „alles schneller“.
- Matrixmultiplikation selbst ist klassisch effizient lösbar, **daher kein genereller Speed-up, sondern nur bei bestimmten AI-Algorithmen (z. B. HHL-Algorithmus für lineare Gleichungssysteme)**.

Wo sind die Grenzen?

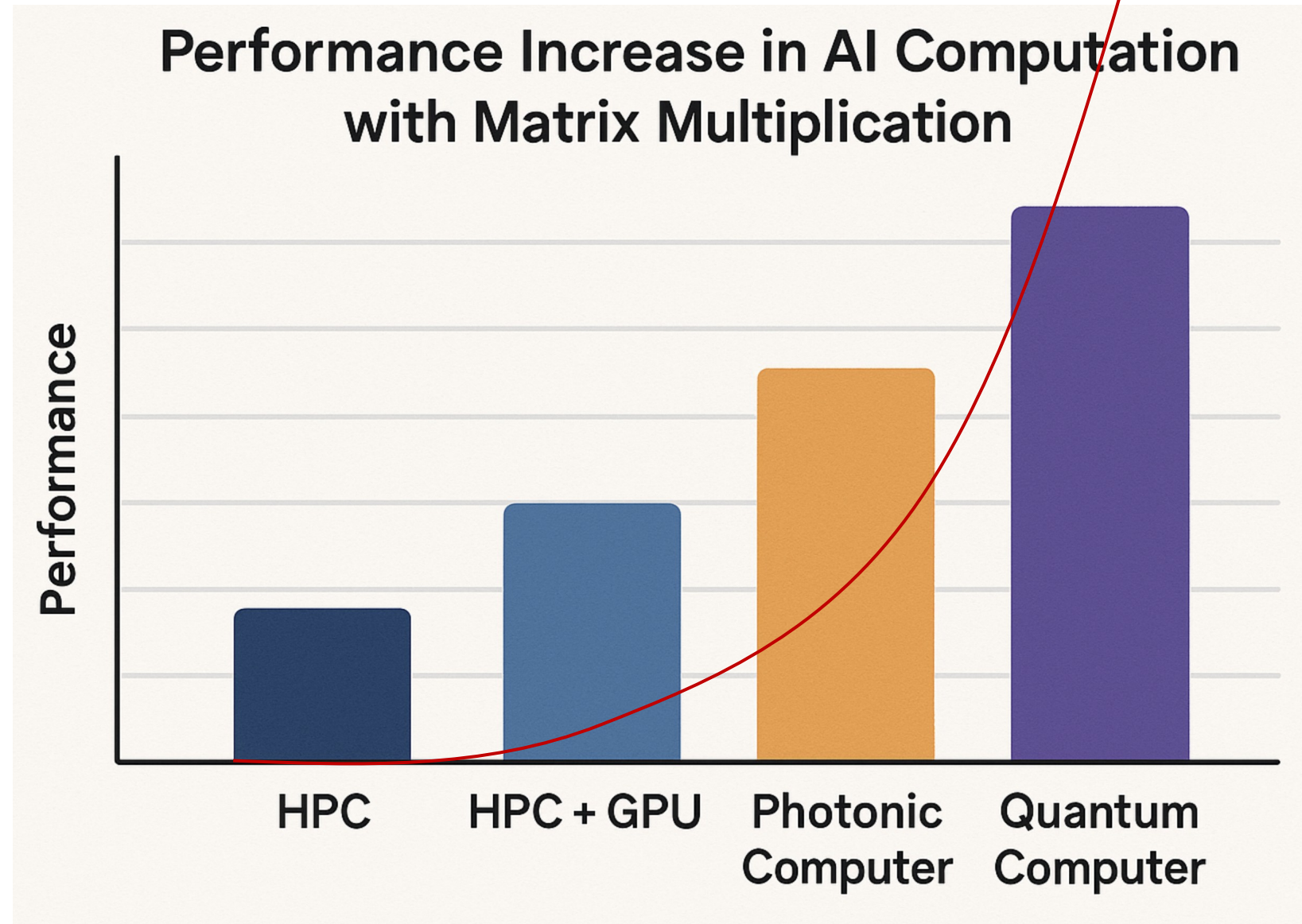
- Quantencomputer: **HHL-Algorithmus für lineare Gleichungssysteme**
 - ❖ Erfindern Harrow, Hassidim und Lloyd, 2009
 - ❖ ist ein Quantenalgorithmus zur Lösung von linearen Gleichungssystemen.
 - ❖ Gilt als einer der bekanntesten Beispiele, wie Quantencomputer bei bestimmten Problemen einen deutlichen Geschwindigkeitsvorteil gegenüber klassischen Computern haben könnten.



Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning und Neuronale Netze usw. mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen – was ist ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- **Wie PC und QC KI-Prozesse zukünftig unterstützen**
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Wo sind die Grenzen?



Wo sind die Grenzen

- HPC (CPU-Cluster): solide, aber **relativ stromhungrig pro Rechenoperation**.
- HPC + GPU (Tensor-/Matrix-Cores): **aktuell beste praktische Energieeffizienz für KI-Matmul** – besonders bei niedriger Präzision (FP16/FP8/INT8) und gutem Daten-Reuse.
- Photonencomputer (analog/optisch): theoretisch **extrem effizient im Rechenkern (Lichtinterferenzen)**, aber heute frisst **I/O/ADC-DAC/Lasernetzteil den Vorteil oft wieder auf**. Gut, wenn sehr große feste Matrizen mehrfach benutzt werden.
- Quantencomputer: **derzeit am ineffizientesten systemweit** (Kryotechnik, Fehlerraten, Overhead). Für Matmul kein genereller Vorteil; energieeffiziente Vorteile nur hypothetisch bei speziellen Algorithmen in ferner, fehlertoleranter Zukunft.

Zukunftsträumerei ?

- Z.B. Q.ANT aus Stuttgart
 - ❖ basiert auf gezieltem Einsatz von Photonik.
 - ❖ Prozessoren beherrschen nativ effiziente Matrix-Vektor-Multiplikationen, d.h. die zentrale Rechenoperation für KI, Machine Learning und neuronale Netze.
- Leibniz-Rechenzentrum rechnet jetzt mit Licht: Weltweit erster photonischer KI-Rechner von Q.ANT geht in Betrieb (22.7.25)

Deutschlands erster hybrider Quantencomputer am Leibniz-Rechenzentrum

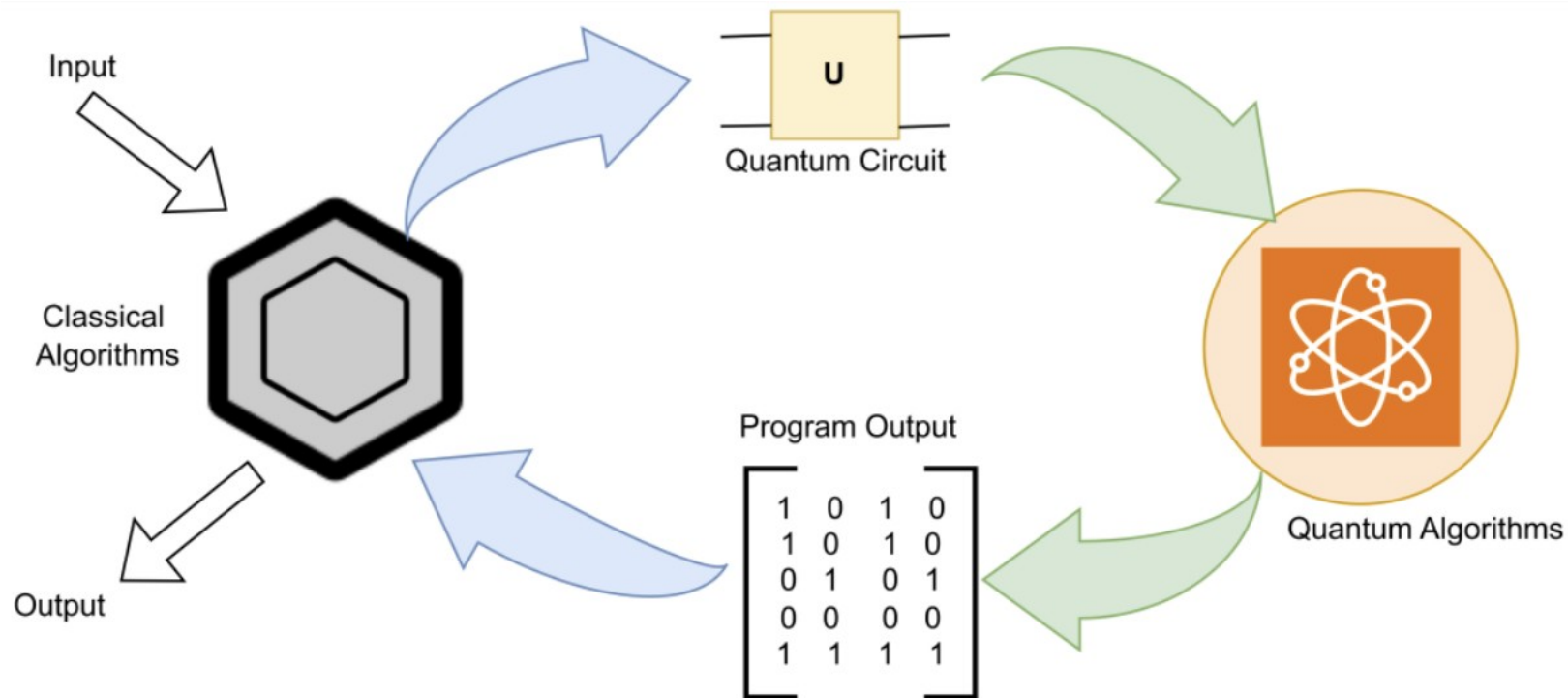
- SuperMUC-NG und das Quantensystem kooperieren!
- BMBF fördert mit 40 Millionen Euro diesen Meilenstein der IT namens Q-Exa,

Kopplung HPC & QC z.B. für Quanten-Maschinelles Lernen (QML).

Das ist die Basis für verteilte DC mit HPC/QC



Quanten Machine Learning Algorithms



Hybrider Quanten-Algorithmus über Quanten-Schaltung

Projekt IonLinQ Universität Hamburg – photonische Qubits

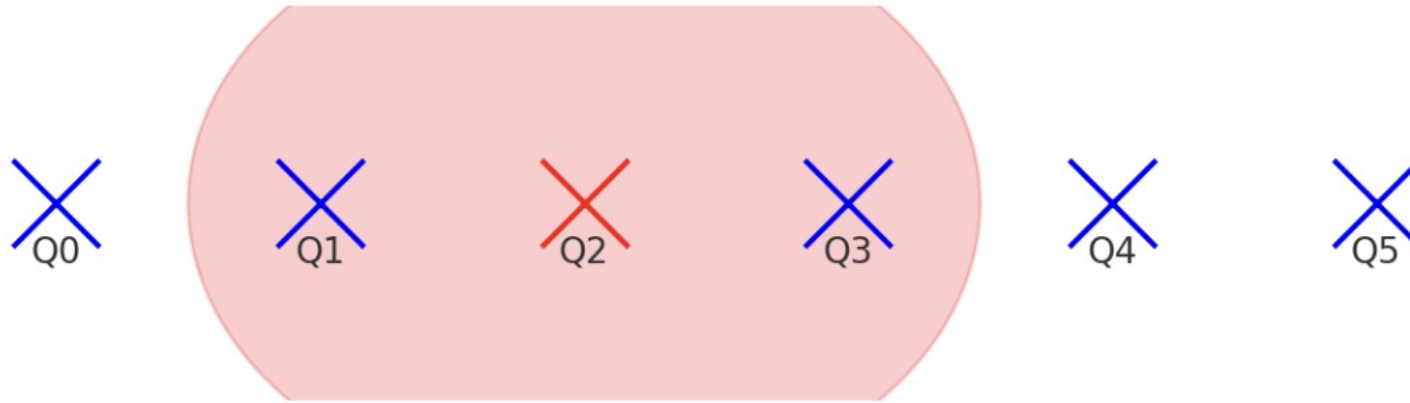
- Das Zentrum für optische Quantentechnologien der Universität Hamburg entwickelt **neuartige photonische Schnittstellen** für Ionen-basierte Quantenprozessoren.
- Quantensupercomputer können zahlreiche Durchbrüche ermöglichen und wichtig Algorithmen deutlich beschleunigen. Um das volle Potential von Ionen-basierten Quantencomputern zu nützen, sind enorme Systemgrößen erforderlich.
- Um dies zu erreichen, **können eine Vielzahl an Quantenprozessoren über ein Netzwerk zusammengeschlossen werden.**

Projekt IonLinQ Universität Hamburg – photonische Qubits

- Hierfür werden Schnittstellen zu photonischen Qubits benötigt, welche schnellen und kohärenten Informationsaustausch zwischen den Komponenten ermöglichen.
- Das Projekt IonLinQ entwickelt und testet **spezialisierte leitende Spiegelbeschichtungen für Mikroresonatoren**, die direkt in die Ionenfallen-Elektroden integriert werden
- Die optischen Mikroresonatoren **verstärken die Interaktion zwischen stationären Qubits und Photonen**, sodass Prozessoren mit hoher Bandbreite miteinander verbunden werden können.
 - ❖ Dies erlaubt eine reibungslose Parallelisierung der Algorithmen innerhalb des Quantensupercomputers um eine modulare skalierbare Rechner-Architektur zu realisieren.
 - ❖ Diese können dann an die Anforderungen relevanter Algorithmen angepasst werden.
 - ❖ **Gleichzeitig können die entwickelten mikrophotonischen Schnittstellen zur Anbindung anderer Qubit-Plattformen verwendet werden**, wie zum Beispiel Rydberg-Prozessoren.

Rydberg-Blockade: angeregtes Atom (rot) blockiert Nachbarn

Rydberg-Blockade: angeregtes Atom (rot) blockiert Nachbarn



Vereinfachtes Diagramm:

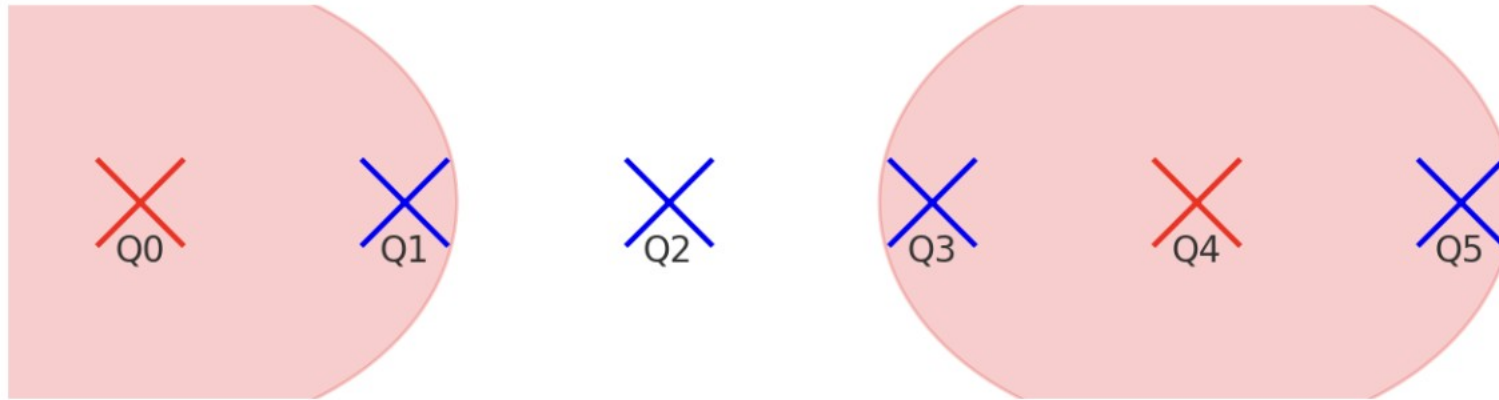
Das rote Atom ist in einem **Rydberg-Zustand**.

Die rote „Wolke“ zeigt den **Blockade-Bereich** – darin können Nachbaratome nicht gleichzeitig angeregt werden.

So lassen sich **kontrollierte Quanten-Gates** aufbauen.

Zwei Atome gleichzeitig angeregt (außerhalb der Blockade-Zone)

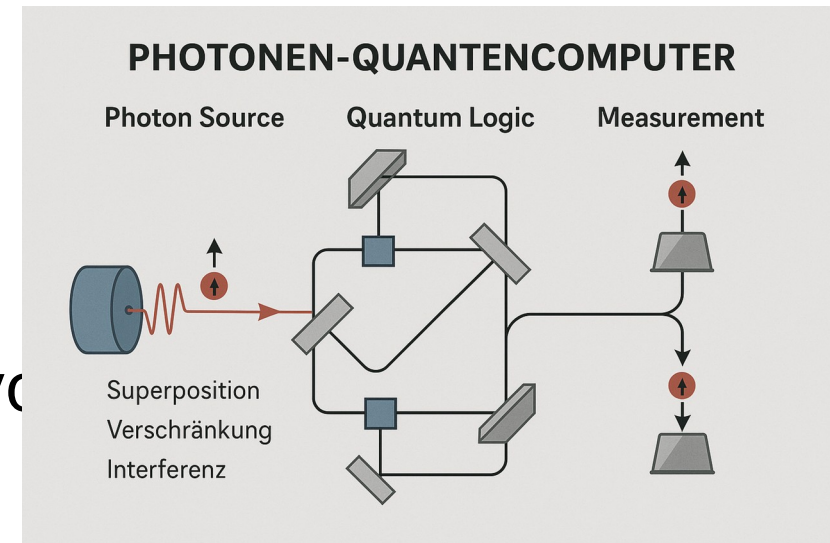
Zwei Atome gleichzeitig angeregt (außerhalb der Blockade-Zone)



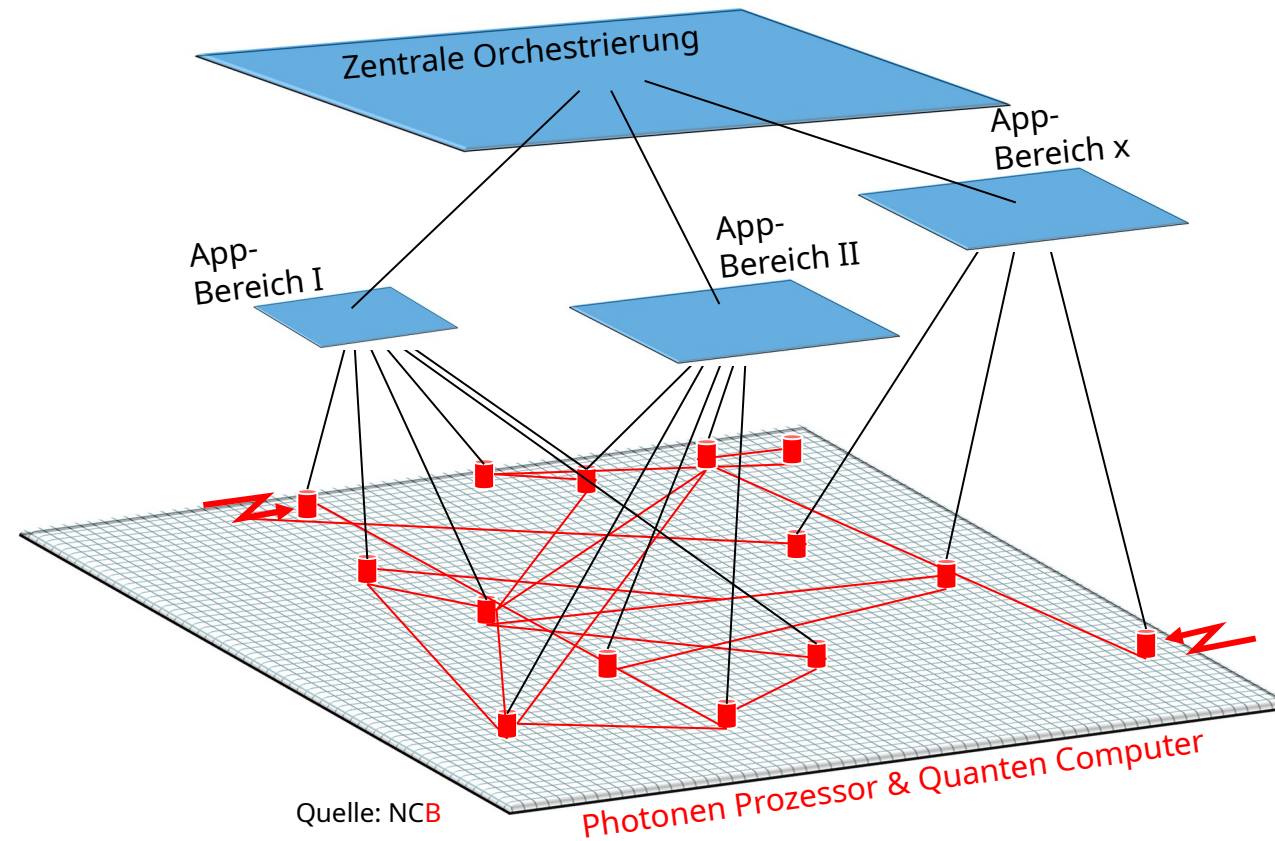
Zwei Atome (rot), **die gleichzeitig in den Rydberg-Zustand** angeregt sind. Da ihre **Blockade-Bereiche** (rote Kreise) sich nicht überschneiden, ist die Anregung gleichzeitig möglich. So entsteht die **kontrollierte Kopplung über Distanz** in einem Rydberg-Prozessor.

Photonen Quantencomputer

- Ein Photonen-Quantencomputer nutzt die Quantenzustände von Lichtteilchen, die über optische Netzwerke geschickt werden.
- Rechenoperationen entstehen durch **Interferenz** und **Verschränkung** der Photonen, gemessen werden sie mit Detektoren.
- Der große Vorteil
 - ❖ ist Stabilität und Kompatibilität mit Telekommunikationstechnik
 - ❖ die Schwierigkeit liegt in der Realisierung von zuverlässigen Mehr-Qubit-Gattern.

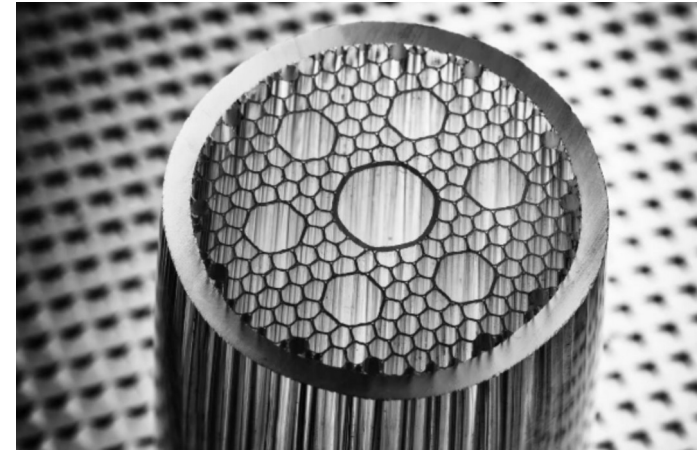


Kopplung über Photonen Computer und Quanten Computer (über Schnittstellen zu photonische QuBits?)



Einsatz von Hollow-Core-Faser

- Die **Hollow-Core-Faser** ist eine Microstructure Optical Fiber (MOF), die sich im Aufbau von den konventionellen Lichtwellenleitern unterscheidet.
- Ihr Prinzip basiert auf dem sogenannten Bandgap oder Bandlückeneffekt und nicht auf dem unterschiedlichen Brechungsindex zwischen Kernglas und Mantelglas.
- Die Latenzzeit wird dadurch um rund 30 Prozent reduziert, was einer Zeitersparnis von 1,5 Mikrosekunden pro Kilometer Übertragungsstrecke entspricht.
- Produkt: AccuCore HCF™ Hohlkernfaser von OFS Fitel, LLC.



Es reicht aber noch nicht

$$\begin{aligned}\lambda &:= 1550 \text{ nm} & L &:= 450 \text{ km} \\ a &:= 0.2 & \text{'Dämpfung dB/km'} \\ P_{ent} &:= 10^{-a \cdot \frac{L}{10}} = 1 \cdot 10^{-9} & \text{'Erfolgswahrscheinlichkeit'}\end{aligned}$$

Wenn die Dämpfung reduziert wird z.B. auf 0.01 dB/km dann ist die Erfolgswahrscheinlichkeit schon 0,355

$$\begin{aligned}\lambda &:= 1550 \text{ nm} & L &:= 450 \text{ km} \\ a &:= 0.01 & \text{'Dämpfung dB/km'} \\ P_{ent} &:= 10^{-a \cdot \frac{L}{10}} = 0.355 & \text{'Erfolgswahrscheinlichkeit'}\end{aligned}$$

gerade mal 30% !!

Das bedeutet, wir brauchen **bessere Lichtwellensysteme** oder **kürzere Übertragungswege**, d.h. es **müssen Verbindungsglieder benötigt werden** -> Quanten-Repeater!

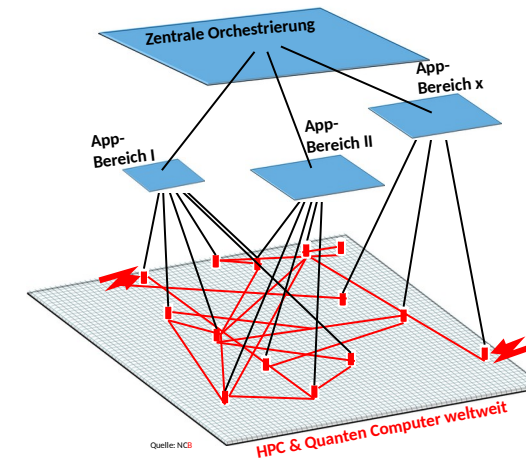
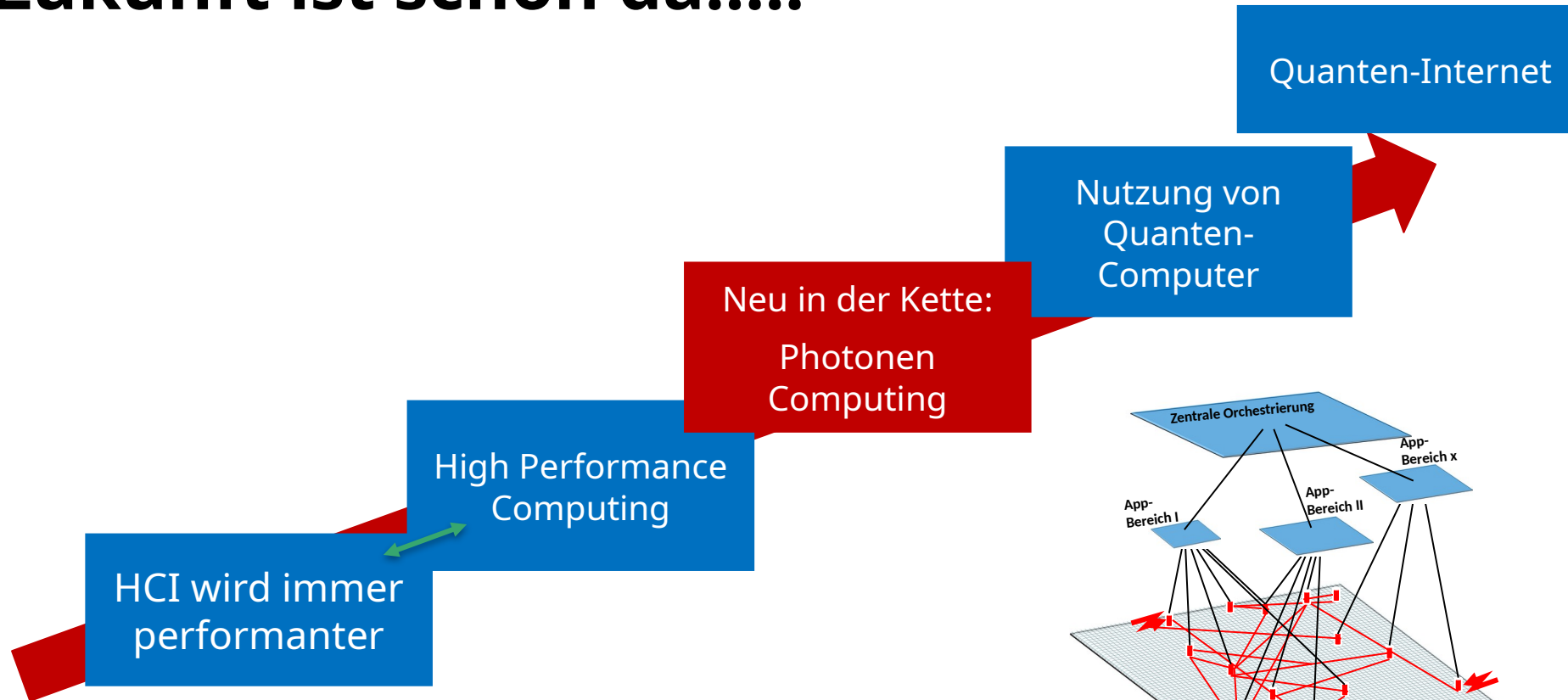
Agenda

- Wer ist VIRZ?
- Wie wird KI mit ML, Deep Learning und Neuronale Netze mit HPC umgesetzt und wo sind die Leistungsgrenzen?
- Grundlagen – was ist ein QuBit?
- Was sind Quantencomputer (QC) und Photonencomputer (PC) wo sind technischen Unterschiede
- Wie PC und QC KI-Prozesse zukünftig unterstützen
- Welche Entwicklungen sind für IT- und OT-RZs zu erwarten?

Wo geht die Reise hin?

- Quantencomputer mit hochverschränkten Photonen aus Halbleiter-Quantenpunkten
- Quantennetzwerke **verbinden Quantenprozessoren oder Quantensensoren miteinander** -> abhörsichere Kommunikation und leistungsstarke verteilte Sensornetzwerke.
- Simulation von Quantenphänomenen, etwa in der Materialforschung
- Forschung in der Medizintechniken
- Quanteninternet mit Photoncomputern und Photonen-Quantencomputer (Siehe Uni Hamburg Projekt „IonLinQ“)
- **IT und OT RZ je mit HPC / Photonenprozessor/Photonencomputer**

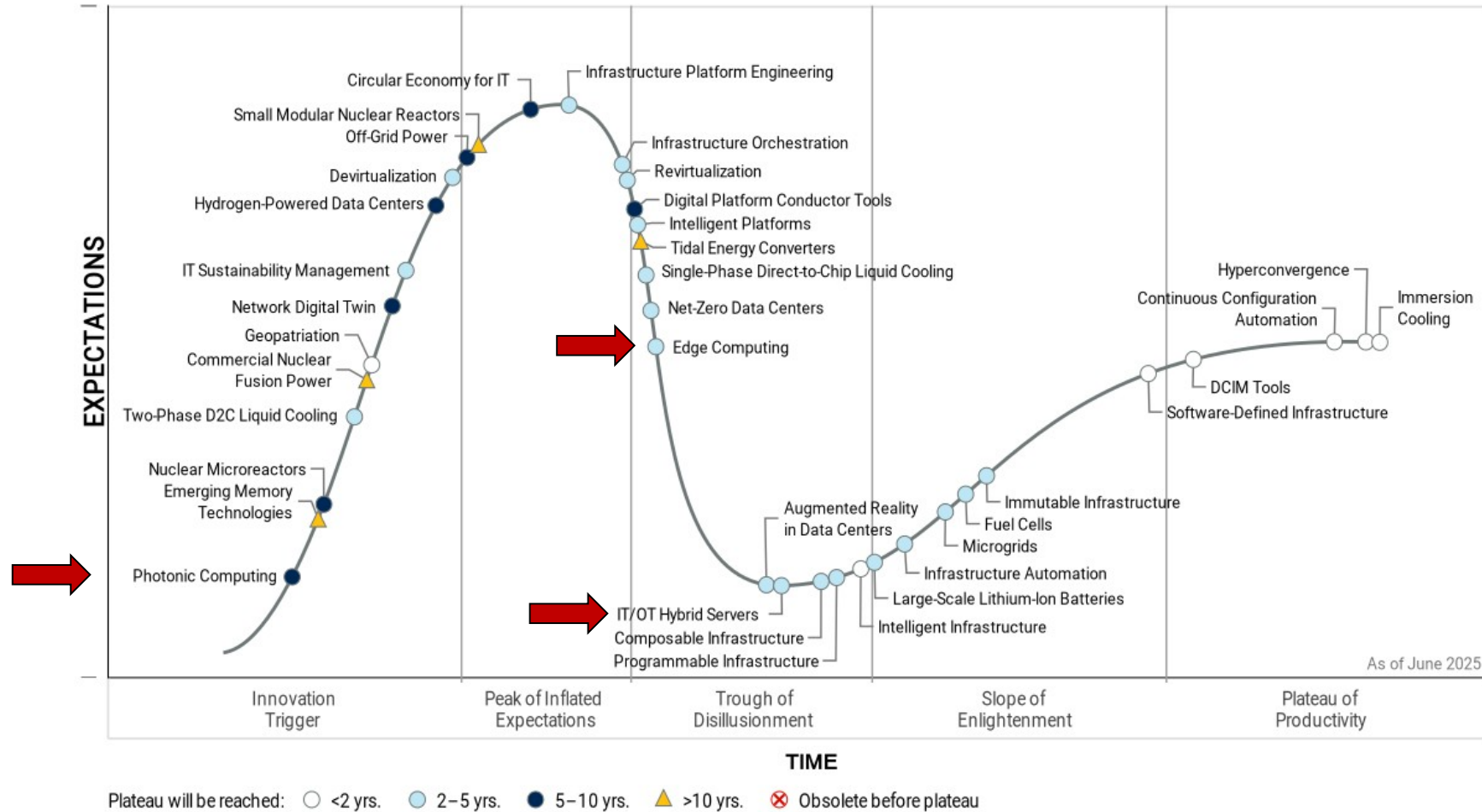
Die Zukunft ist schon da.....



und das alles in verteilten RZ!
Mit einem zus. Baustein, dem
Photonen Computer

Was sagt Gardner 2025

Hype Cycle for Data Center Infrastructure Technologies, 2025



Gartner

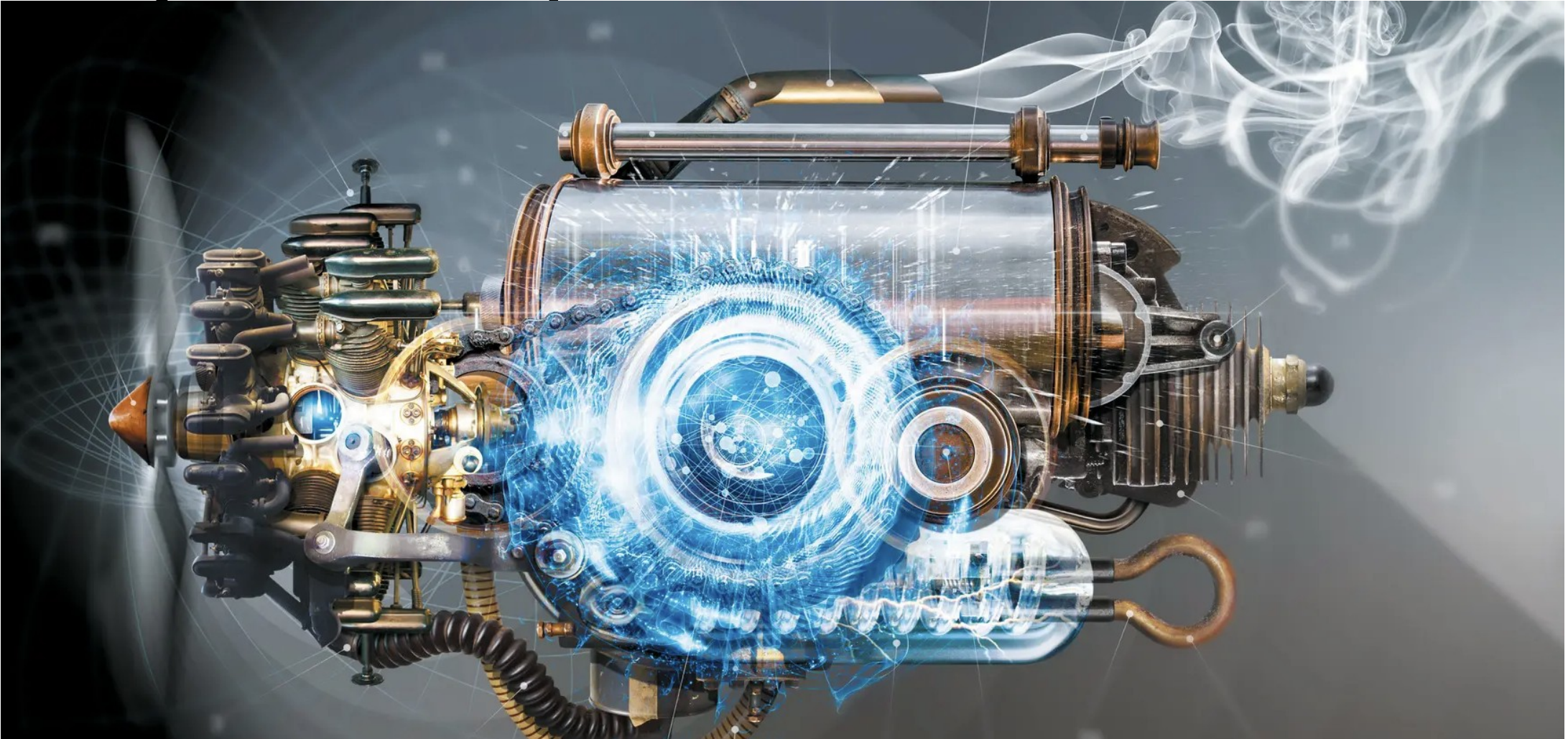


87.K-Stammtisch Karlsruhe

**Herzlichen Dank für Ihre
Aufmerksamkeit**

Hartwig Bazzanella
NCB GmbH / VIRZ e.V.
hbaz@ncb.de

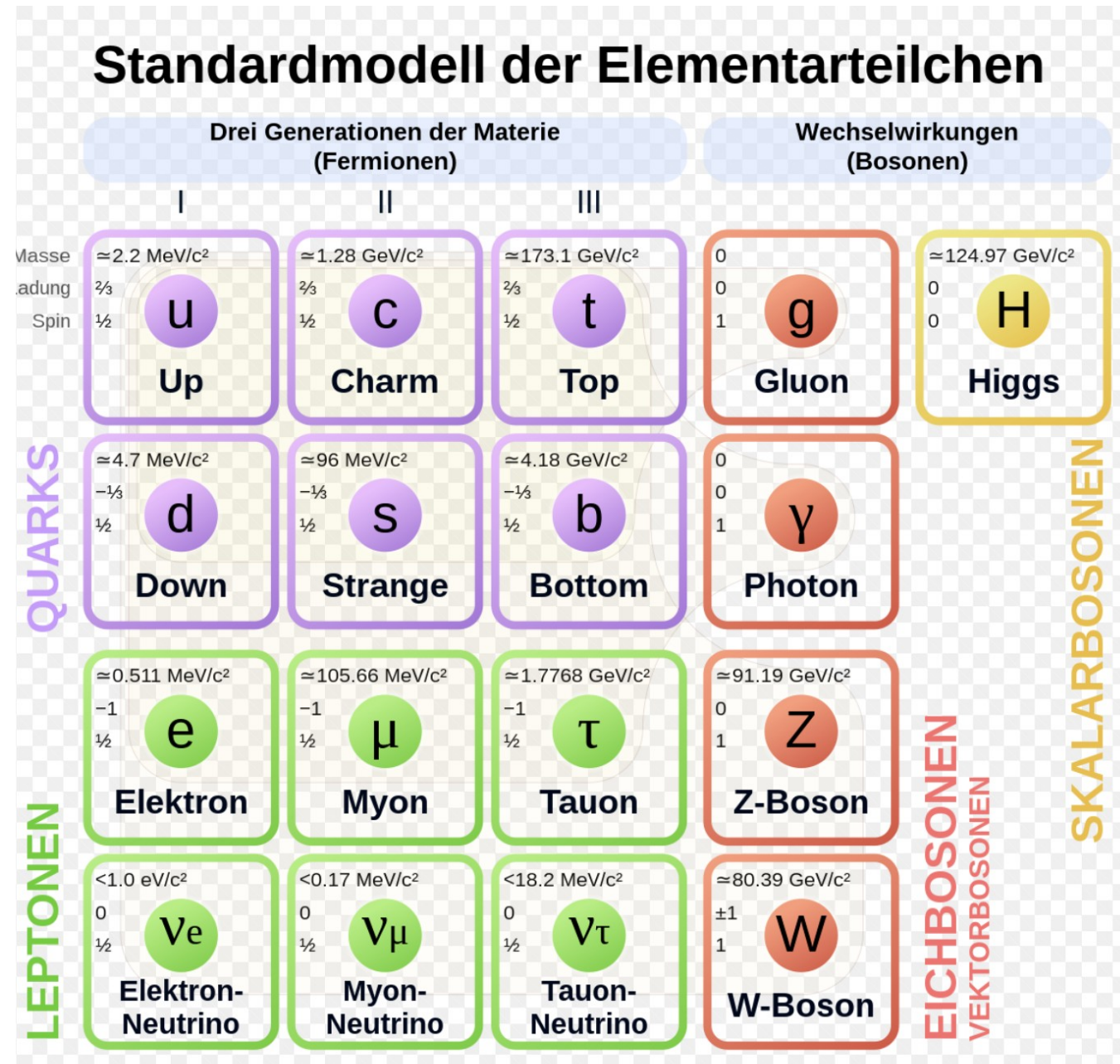
Die Quanten-Dampfmaschine



Steampunk Genre

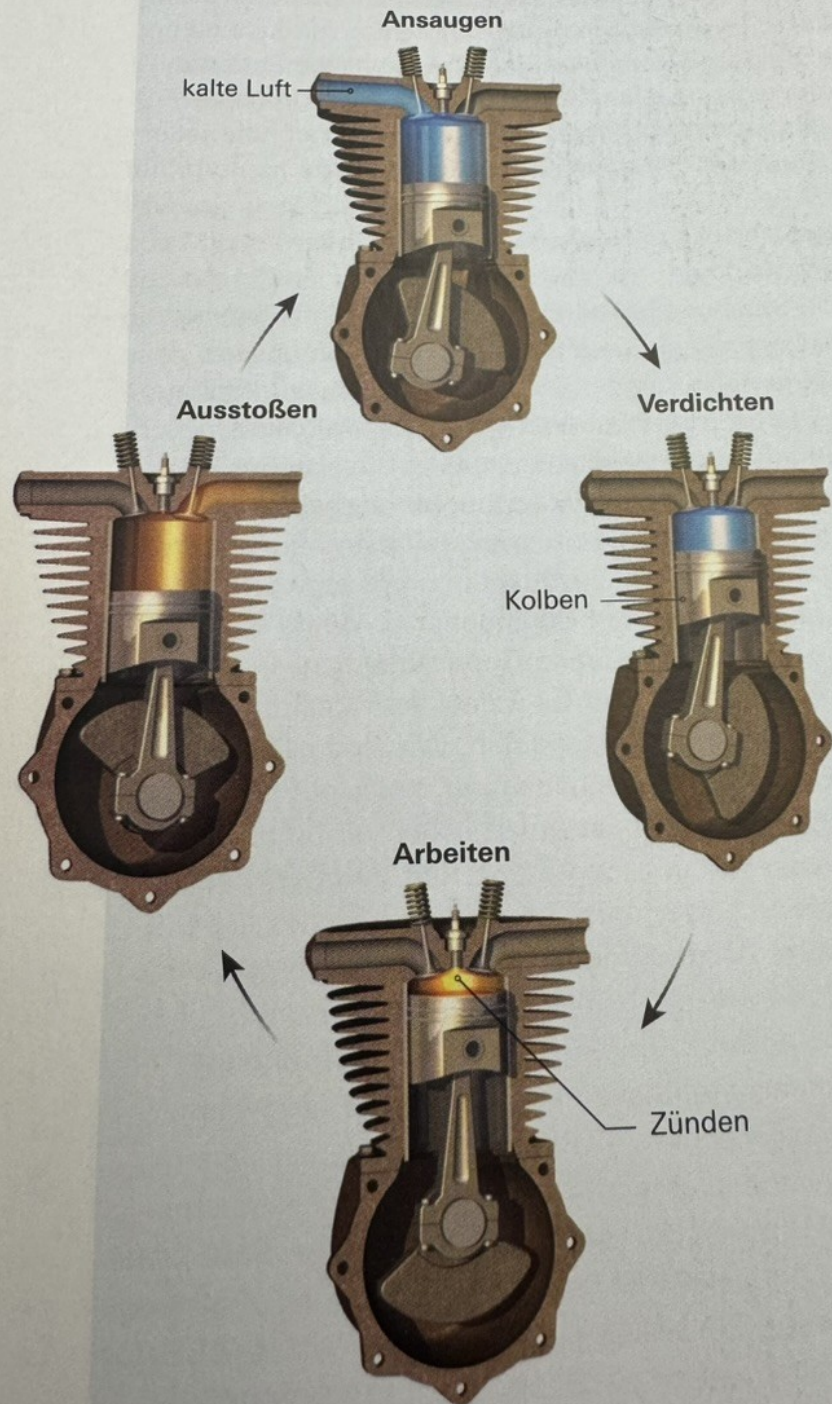
- Quanten Sternpunkt: Quantenphysik, Informationstheorie und Thermodynamik
- Werkzeugkasten der Quantenphysik: Bauteile auf Basis von
 - ❖ Photonen (Lichtteilchen)
 - ❖ Elektronen
 - ❖ und supraleitende Schaltkreise

Umwandlung von Fermionen in Bosonen erzeugt Energie, die gespeichert werden kann



MBL-Mobil – bzw. Quanten Dampfmaschine

- Herzstück einer modernen Konstruktion,
 - ❖ eine quantenmechanische Phase der Materie,
 - ❖ die sogenannte Vielteilchen-Lokalisierung (MBL, many-body localization)



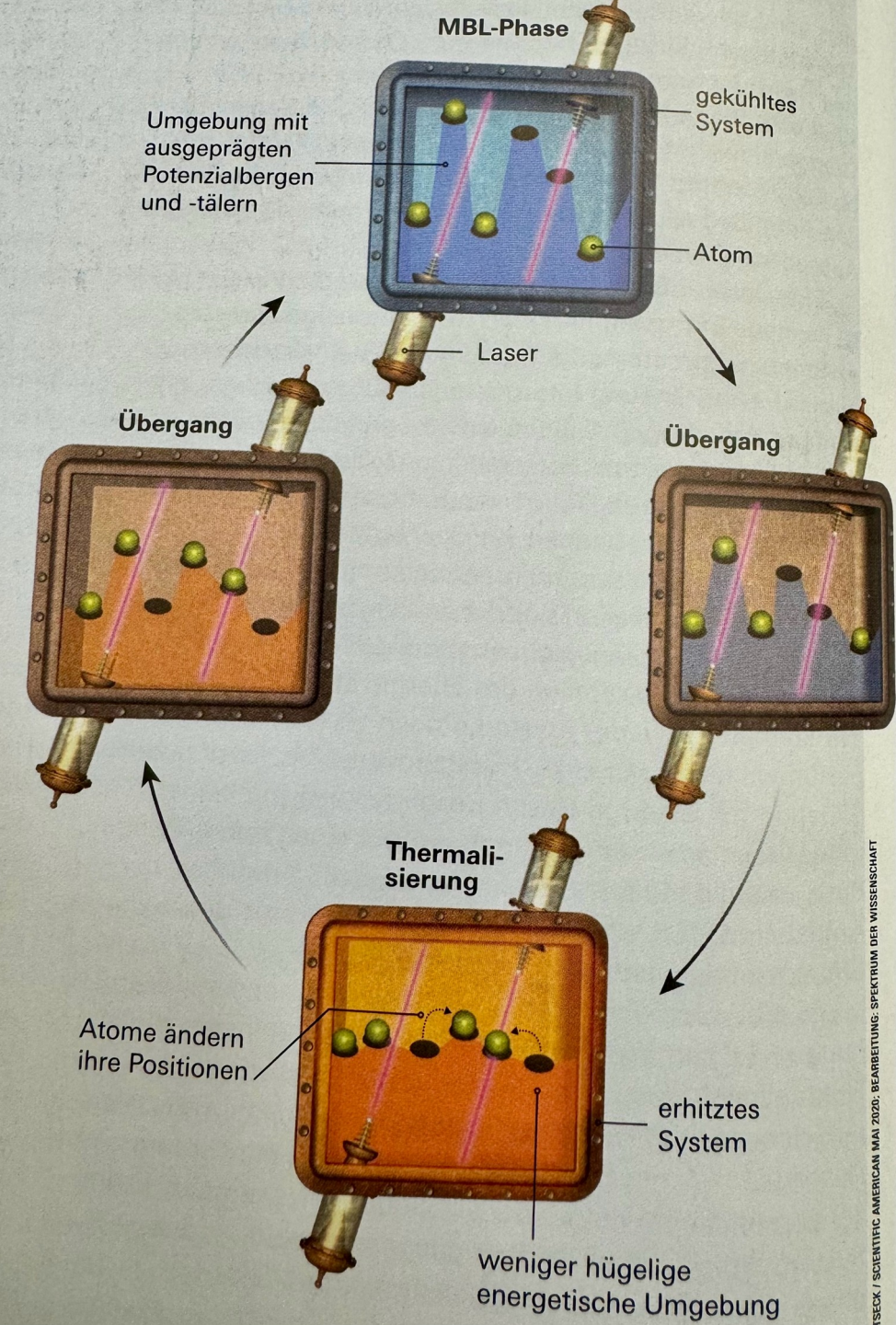
Vierstufiger Prozess:

- Ansaugen kalter Luft, Einspritzen von Kraftstoff
- Verdichtung des Gemisches durch den Kolben
- Im Arbeitstakt zünden des Gemisches, worauf es sich ausdehnt und den Kolben beschleunigt
- Ausstoß der heißen Luft und der Verbrennungsprodukte

Dampfmaschine

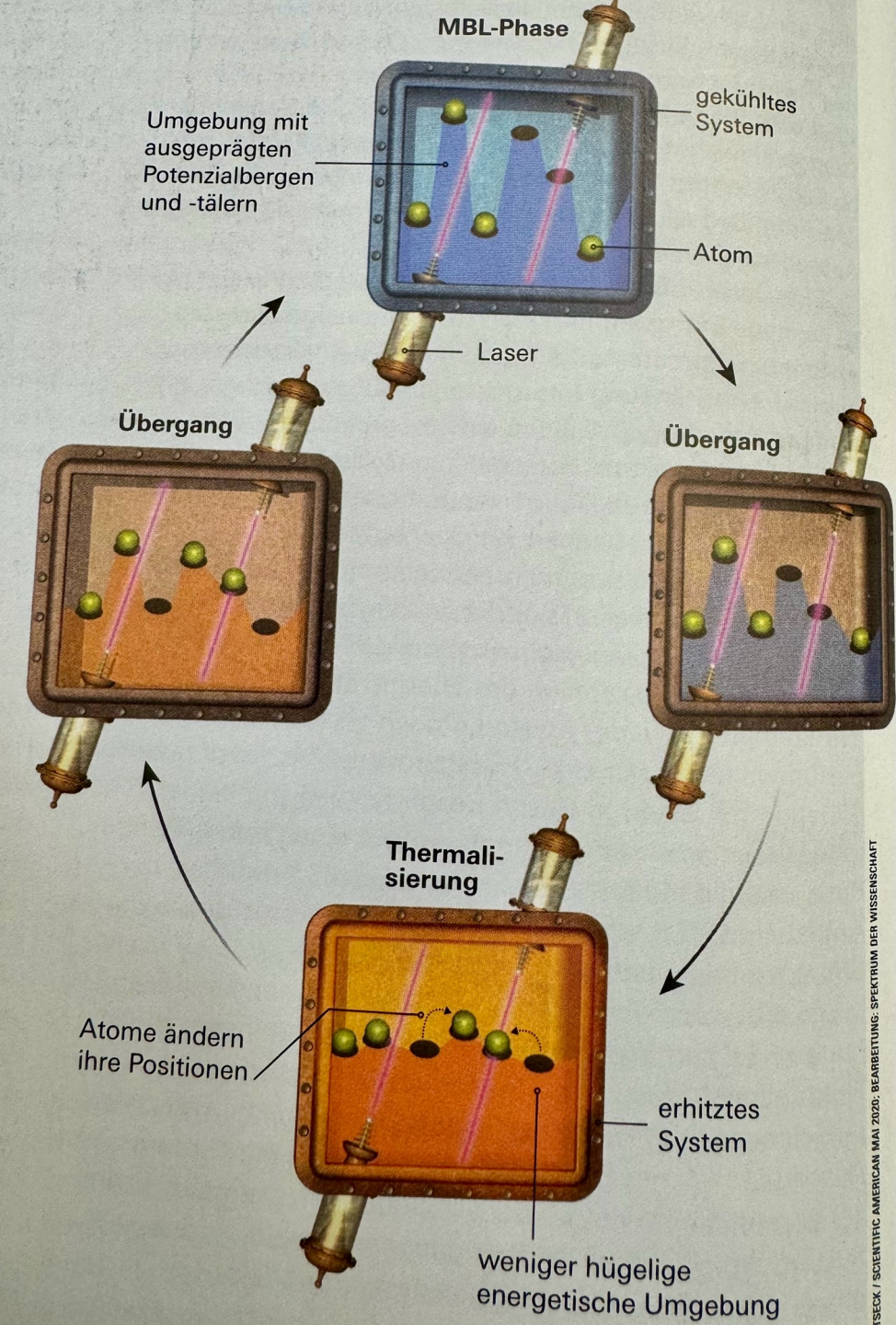
Vierstufiger Zyklus:

- Vielteilchen Lokalisierung: Atome sind in einer kühlen Umgebung mit stark unterschiedlichen Energien. Damit bleiben sie weitgehend auf Ihren Plätzen
- Laser steuern einen Übergang in eine Phase, in der die Energieunterschiede geringer sind und die Atome sich bewegen können.
- Die Laser bringen das System zurück in die MBL-Phase. Während des Übergangs leistet der Motor „Arbeit“



MBL-Mobil

- Im Vergleich der Leistungsdichte, d.h. Leistung pro Volumeneinheit, ist sie bei einem Automobil zehnmal so hoch wie das MBL
- Der MBL kann jede beliebige Größe haben, von zehn Teilchen bis zu beliebig vielen - im Parallelbetrieb
- Das Prinzip kann umgekehrt werden! Der Zyklus wird umgedreht und der Motor als Kältemaschine betrieben, die Wärme aus einer kalten Umgebung abführt und an eine wärmere abgibt.
- **Und: Es gibt Konzepte, wie dieses MBL als Energiespeicherung verwendet werden kann!**



Summary

- Allein diese Vorstellung zeigt das unendliche Potential durch den Einsatz von Quantencomputer
- Hier stehen wir erst ganz am Anfang
- Das bedeutet: Zukünftige Lösungen für den Energie-Speicher sind auch in ganz anderen Technologien denkbar!