

# AUFBAU EINES SOFTWARE-ÖKOSYSTEMS FÜR QUANTENCOMPUTER AM DLR

**Dr. Elisabeth Lobe**  
Institut für Softwaretechnologie (SC)  
Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)



# Forschungsfelder



Energie-



Kryptographie

HPC-  
Systeme



Bild: CARA

## Komplexe Probleme

- enormer Rechenbedarf
- stark vereinfachter Modelle oder teure Bau-und-Test-Zyklen

Material-  
simulation



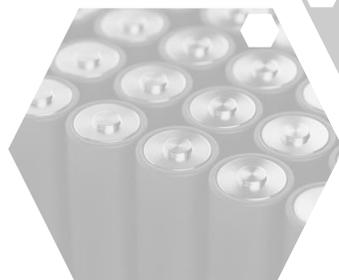
Klima-

Quanten-  
computer

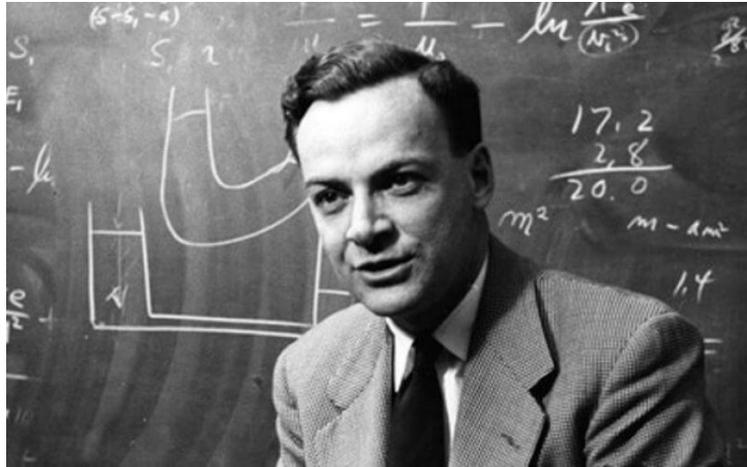


Bild: IBM

Batterie-  
forschung

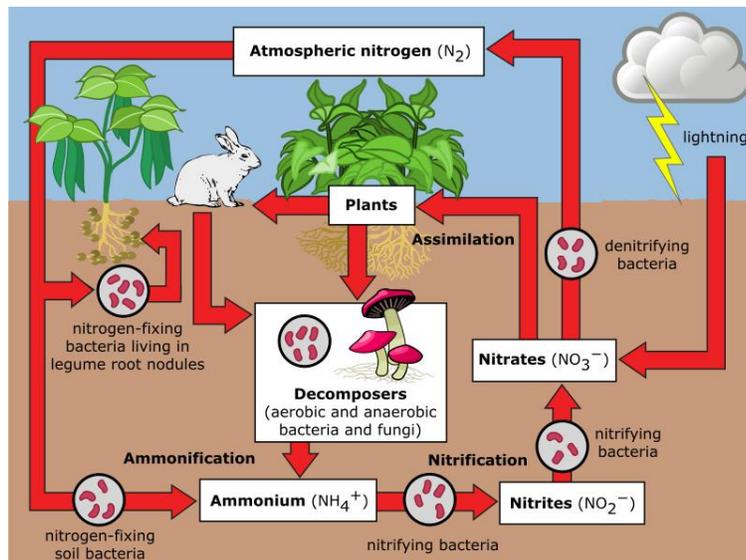


# Ursprüngliche Idee des Quantencomputings



“Nature isn’t classical, dammit,  
and if you want to make a simulation of nature,  
you’d better make it quantum mechanical,  
and by golly it’s a wonderful problem,  
because it doesn’t look so easy.”

Richard Feynman (1981)



Nitrogen Cycle, source: Johann Dréo, Wikipedia

## ▪ Herausforderung:

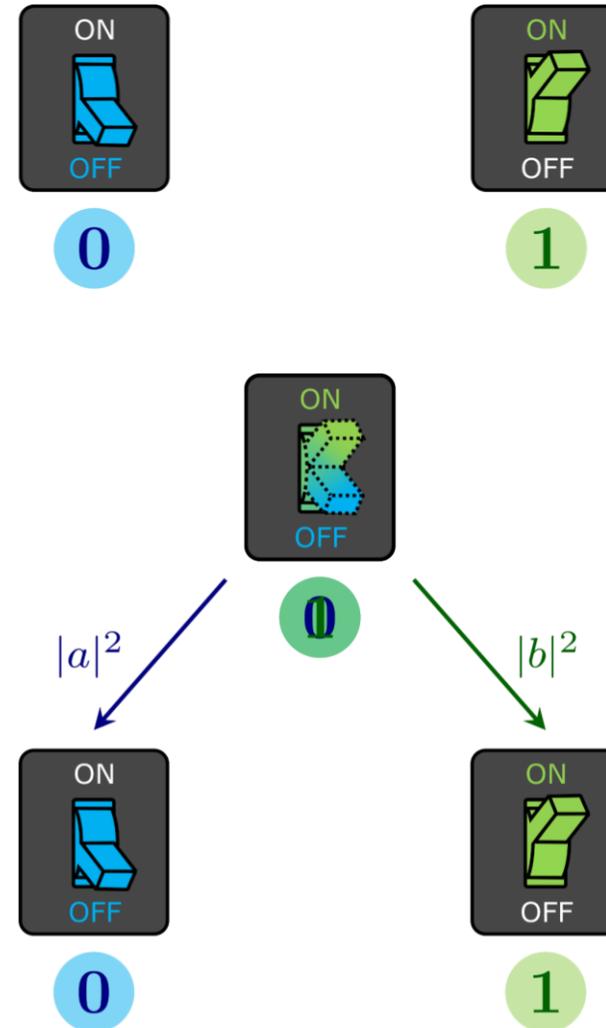
Die Dimension des Hilbert-Raumes eines Quantensystems skaliert **exponentiell** mit der Anzahl der Partikel

## ▪ Lösung:

Nutzen des exponentiell großen Hilbert-Raumes als **Ressource** für Computing

# Quantenbits (Qubits)

- Wert eines klassischen Bits ist **entweder „0“ oder „1“**
- Zustand eines Quantenbits ist Superposition von „0“ **und** „1“  
 $|\phi\rangle = a|0\rangle + b|1\rangle$
- Messung verändert Zustand in
  - $|\phi\rangle = |0\rangle$  mit Wahrscheinlichkeit  $|a|^2$
  - $|\phi\rangle = |1\rangle$  mit Wahrscheinlichkeit  $|b|^2$
- Ergebnis entspricht klassischer Information



# DLR QUANTUM COMPUTING INITIATIVE



**DLR  
Quantum Computing  
Initiative**

*Wir gestalten das Ökosystem  
Quantencomputing*

- Gebündelte Technologie, Expertise & Infrastruktur
- Forschung und Entwicklung auf Basis der DLR Technologiefelder
- Beschaffung von QC-Komplettsystemen und -Komponenten durch technologieoffene wettbewerbliche Ausschreibungen
- Ertüchtigung durch die Beauftragung der notwendigen Grundagentechnologien und Anwendungen

[qci.dlr.de](http://qci.dlr.de)



740 Millionen Euro über vier Jahre

80 %

Industrie, Startups  
und Zulieferer



20 %

Forschung und  
Entwicklung, DLR

Entwicklung von Quantencomputern,  
Enabling-Technologien, Software und Anwendungen

Wir bringen Quantencomputer in die industrielle Anwendung

**Wirtschaft**

Die DLR QCI ist ein zuverlässiger Ankerkunde für das QC-Ökosystem

**Ökosystem**

Technologische Vielfalt und eine enge Vernetzung für eine starke QC-Industrie

**Technologie**

Transfer aus der Grundlagenforschung in die wirtschaftlich relevante Anwendung

**Souveränität**

Sicherung von Knowhow und Expertise für einen souveränen Technologiezugriff



# DLR Quantum Computing Initiative

- Innovationszentren
- DLR-Institute
- Industriepartner

*Wir gestalten das Ökosystem  
Quantencomputing*

1

Einzigartiges  
Ökosystem

2

Innovations-  
zentren

19

DLR-  
Institute

22

Industrie-  
partner

6

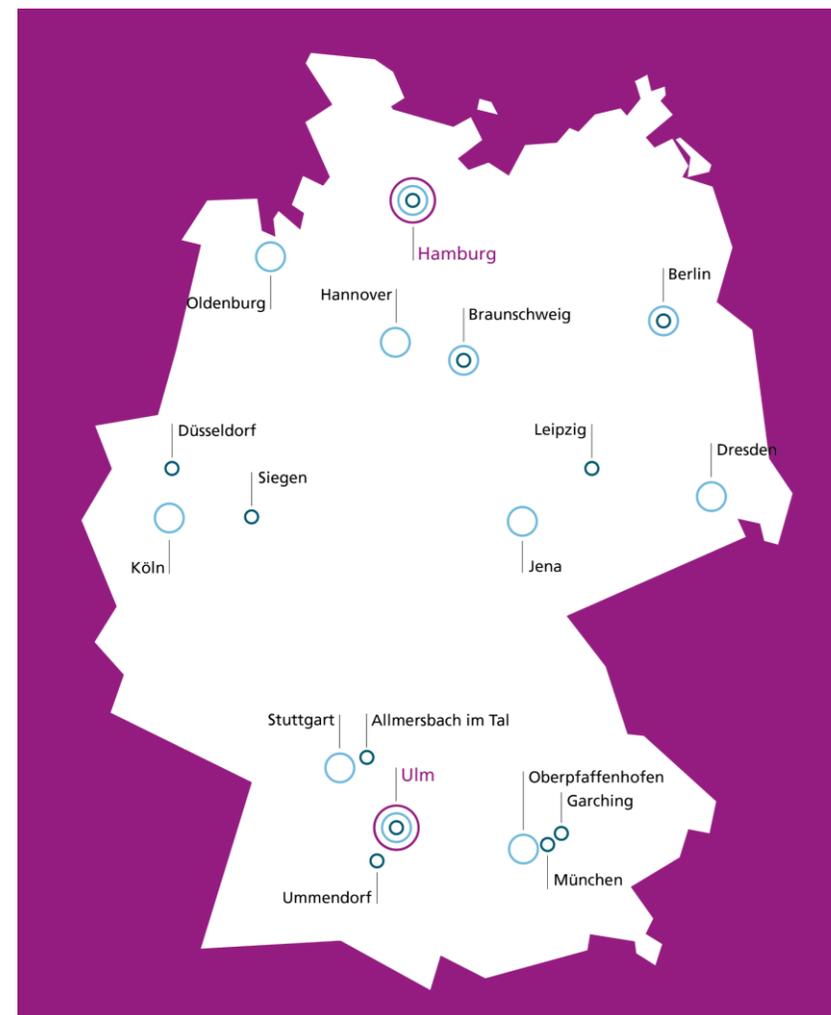
Qubit-  
Technologien

17

Hardware-  
projekte

22

SW / App-  
Projekte



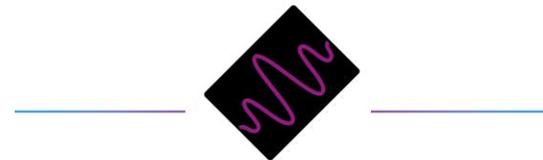
# QCI – Industriepartner



# Technologien



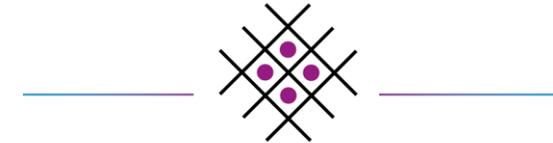
Es gibt viele Möglichkeiten, Qubits und damit Quantencomputer in Hardware zu realisieren. Deswegen beauftragen wir die Entwicklung unterschiedlicher Hardware-Plattformen.



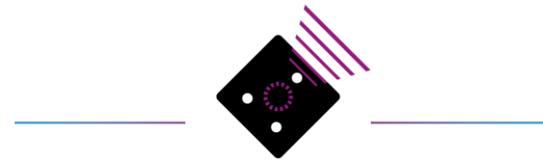
Analogrechner



Ionenfallen



Neutralatome



NV-Zentren



Photonen

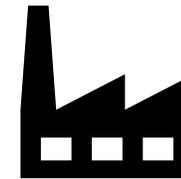


Spin-enabling

→ Welcher ist der beste Ansatz zum universellen, fehlerkorrigierten Quantencomputer?



# Potentielle Begünstigte



- Zugriff auf und Evaluation von verschiedenen Hardwareplattformen
- Benchmarking verschiedener Implementierungen
- Entwicklung neuer Quantenalgorithmen
- Nutzung von Quantenalgorithmen
- Schnelleres/besseres Lösen von Domänenproblemen



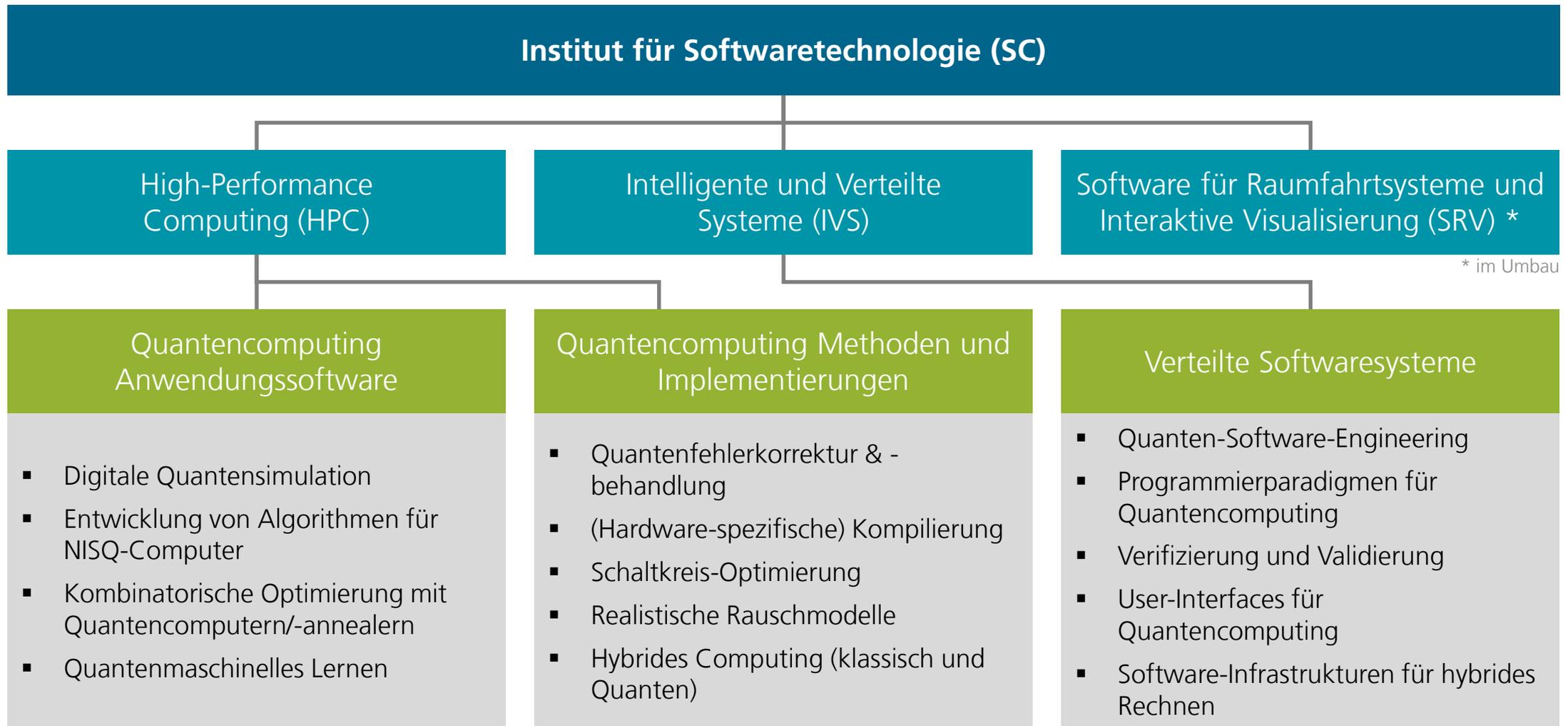
# QCI – Projektüberblick



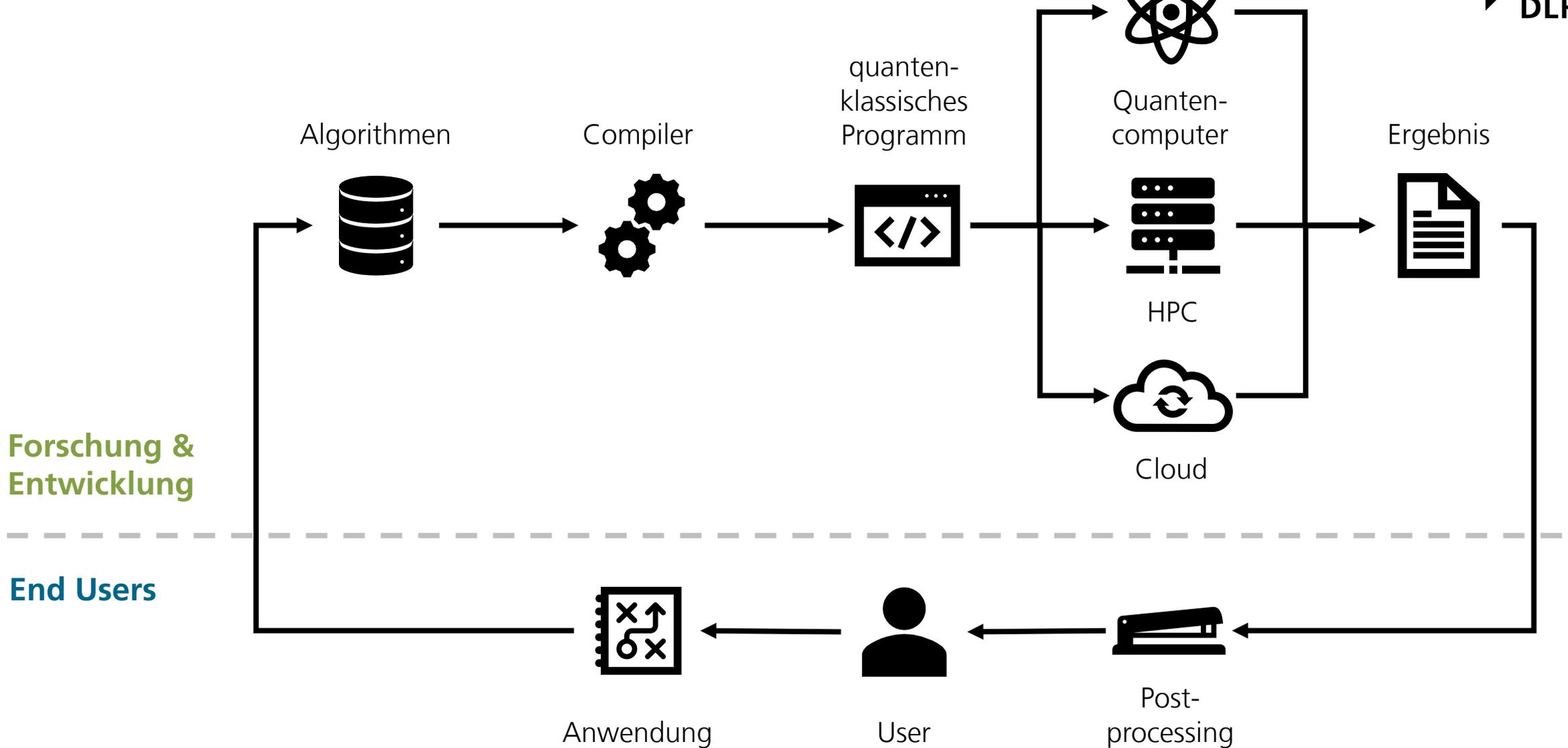
Hardware				Anwendungen	
Quantencomputer	Enabling-Technologien	Hardwarenahe Software	Middleware	Anwendungsprojekte	
<b>Legato</b> Ionenfallen Universal Quantum	<b>DIAQ</b> Spin-enabling Diatope	<b>ALQU</b> HW/SW-Kodesign DLR SC	<b>Quant<sup>2</sup>AI</b> Quanten-Maschinelles-Lernen DLR KI	<b>Attract'em</b> Optimierung DLR VE	<b>BASIQ</b> Materialwissenschaft DLR TT
<b>QSea I</b> Ionenfallen eleQtron   ParityQC   NXP	<b>PiQ</b> Photonen-enabling DLR OS	<b>AQuRA</b> Analoger Quantenrechner DLR QT	<b>QuTeNet</b> Quanten-Maschinelles-Lernen DLR KI	<b>Klim-QML</b> Quanten-Maschinelles-Lernen DLR PA	<b>QCMobility</b> Optimierung DLR QT   VF   LV   TS   SE
<b>QSea II</b> Ionenfallen eleQtron   ParityQC   NXP	<b>SQuAp</b> Spin-enabling Advanced Quantum	<b>CLIQUE</b> QC-Fernzugriff DLR SC   SP	<b>R-QIP</b> Quantenfehlerkorrektur DLR KN   SC   QT   RB	<b>QCoKalm</b> Quanten-Maschinelles-Lernen DLR DW	<b>Qlearning</b> Quanten-Maschinelles-Lernen DLR QT
<b>REDAC</b> Analogrechner Anabrid	<b>StarQ</b> Spin-enabling DLR QT	<b>IQDA</b> Hardware-nahe Software DLR SC		<b>QMPC</b> Optimierung DLR RB	<b>QUA-SAR</b> Optimierung DLR HR
<b>SuNQC</b> NV-Zentren SaxonQ	<b>TeufIQ</b> Spin-enabling DLR QT			<b>QuantiCoM</b> Materialwissenschaft DLR WF   MP   TT	<b>Quantity</b> Quanten-Kryptoanalyse DLR KN
<b>Toccatà</b> Ionenfallen Universal Quantum				<b>ToQuaFlics</b> Quanten-Maschinelles-Lernen DLR AS   SP	<b>NeMoQC</b> Quanten-Maschinelles-Lernen DLR KI
<b>UPQC</b> Photonen QuiX Quantum				<b>QC Mineral</b> Materialwissenschaft DLR MP	<b>QCOptSens</b> Optimierung DLR OS
<b>XAPHIRO</b> Ionenfallen QUDORA Technologies				<b>QI-Mozart</b> Optimierung DLR AS	
<b>XQi</b> NV-Zentren XeedQ					
<b>DiNAQC</b> Neutralatome planqc					



# QC @ DLR-SC



# Angestrebte Nutzererfahrung



# Entwicklungsstand Quantencomputer

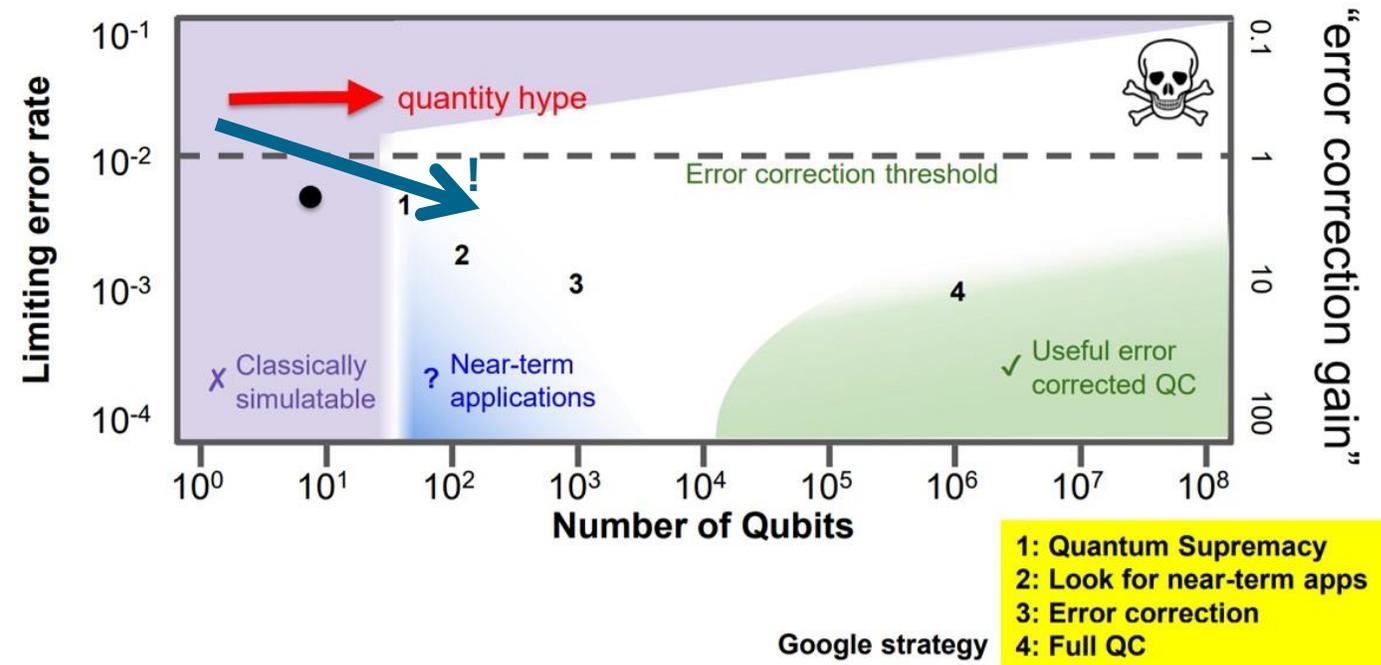
Siehe auch BSI-Studie  
<https://www.bsi.bund.de/qcstudie>



- Universale Quantencomputer sind intrinsisch anfällig für Fehler
  - Für die Implementierung der „Killer Apps“ (z.B. Shor) brauchen wir Quantenfehlerkorrektur (QEC)
  - QEC basiert auf Redundanz
  - Produziert signifikanten Overhead bzgl. Qubits und Gattern
- ➔ Müssen unter Threshold

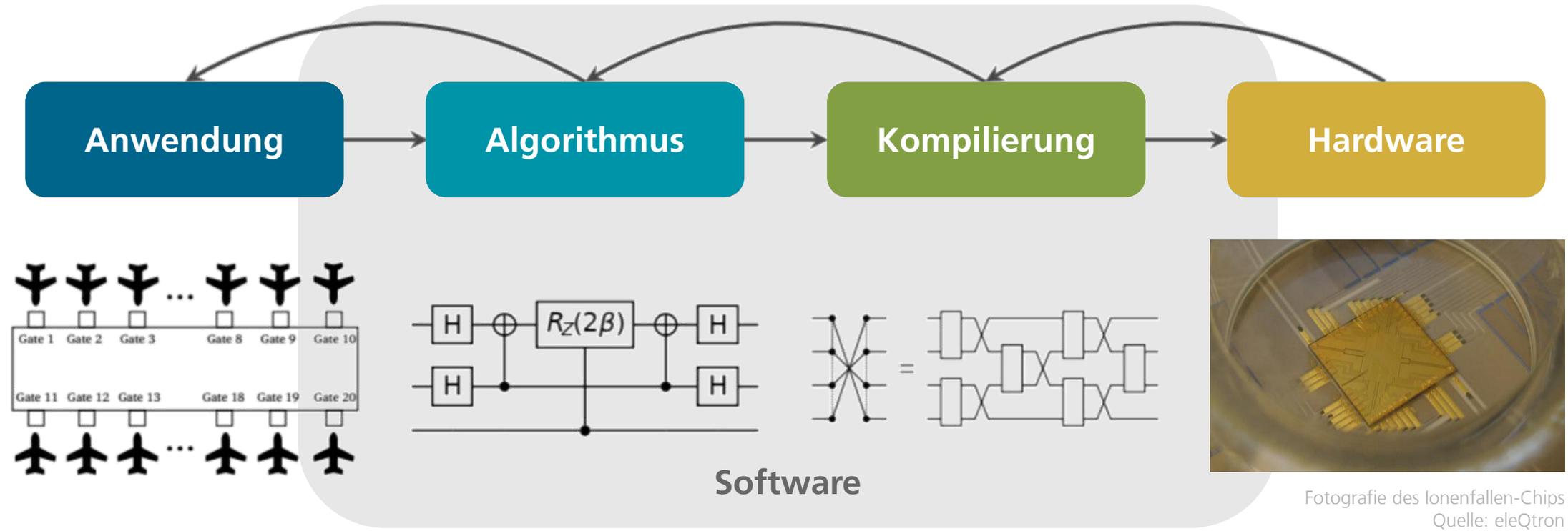
Quelle: John Martinis, Google.

([https://medium.com/@quantum\\_wa/quantum-computing-near-and-far-term-opportunities-f8ffa83cc0c9](https://medium.com/@quantum_wa/quantum-computing-near-and-far-term-opportunities-f8ffa83cc0c9))



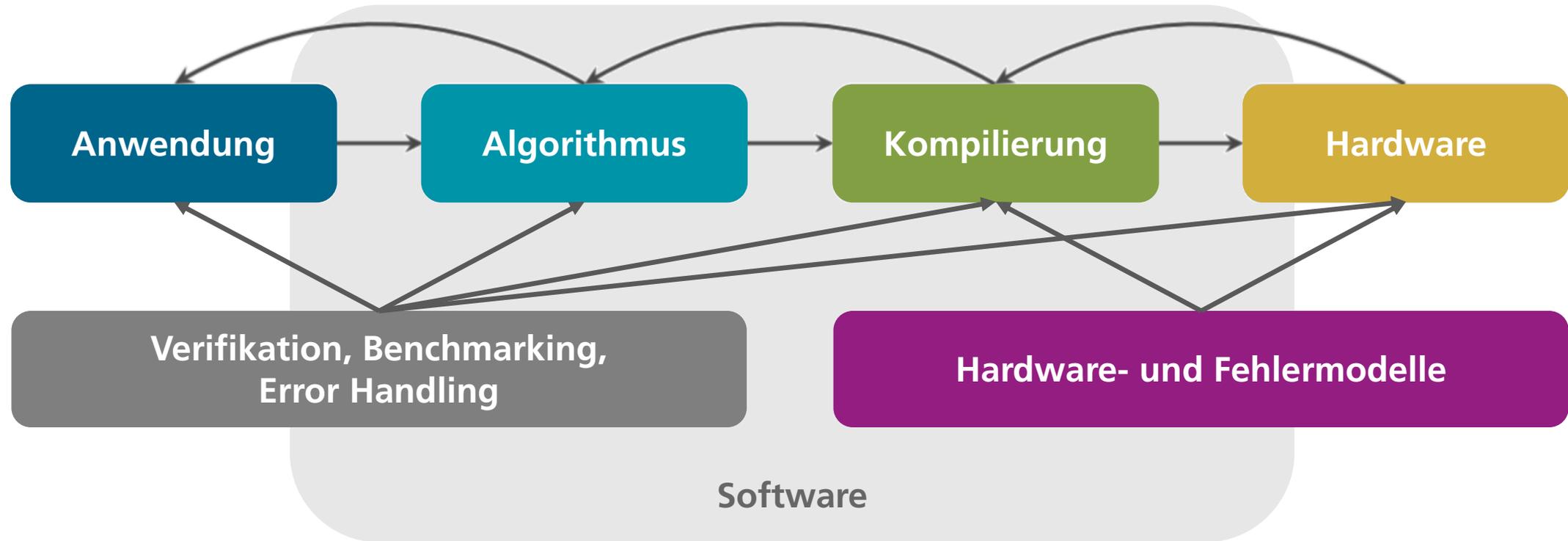
Quantenvorteil in naher Zukunft nur mit Noisy Intermediate-Scale Quantum (NISQ) Computern!

# Hardware-Software-Co-Design



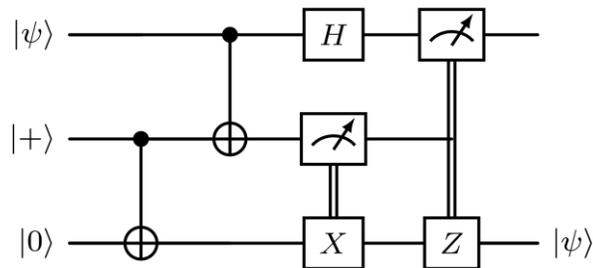
- **Identifikation** aussichtsreicher QC-Anwendungen von der einen sowie QC-Plattformen (digital, analog, topologisch) von der anderen Seite
- **Entwicklung** von neuen Algorithmen und Anwendungssoftware für NISQ-Computer mit Ziel des Quantenvorteils durch optimale Ausnutzung von Hardwarebesonderheiten

# Hardware-Software-Co-Design



- Zugriff und damit Adaptierbarkeit auf allen Ebenen des QC-Stacks
- Geeignete Abstraktionsschichten und Solution Patterns
- Ermöglichung von hybridem Rechnen
- Integration von SW/HW-Herstellern und Anwendern
- Vergleich von DLR QC-Hardware mit state-of-the-art klassischen Verfahren

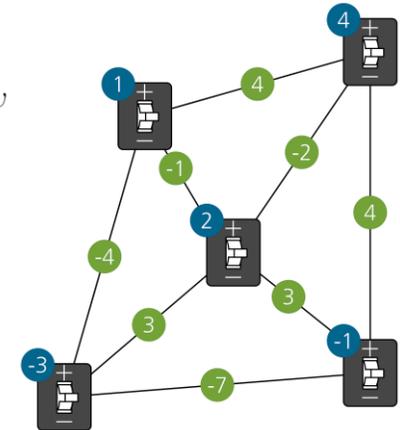
## Gatter-basiertes Quantencomputing



- Manipulation von Qubitregistern mittels Quantengattern
- Universell programmierbar mit Algorithmen, die durch Quantenschaltungen dargestellt werden
- Ausnutzung inhärenter quantenmechanischer Eigenschaften: Überlagerung und Verschränkung mit anderen Qubits

## Adiabatisches QC / Quantum Annealing

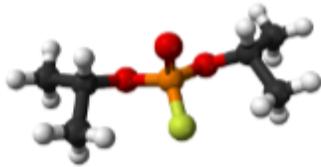
$$\min_{s \in \{-1,1\}^V} \sum_{v \in V} W_v s_v + \sum_{vw \in E} S_{vw} s_v s_w$$



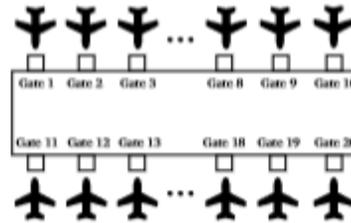
- Optimierung eines spezifischen Problems: Ising/QUBO
- Einzelner „Algorithmus“: Adiabatische Evolution eines Quantensystems
- Nur heuristische Realisierung des theoretischen Konzepts
- Sampling aus der Niedrigenergie-Verteilung

# Vielversprechende Anwendungen?

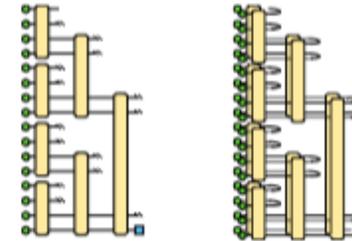
## Quanten- simulation



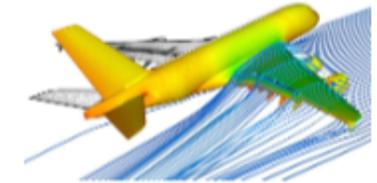
## Kombinatorische Optimierung



## Quantum- enhanced ML



## “Klassische” Simulation



heute

Zeit / benötigte  
Fehlerkorrektur

- Vibronische Struktur und Dynamik
- Atomistische Simulation von Legierungen
- Elektronenübertragung in Fotovoltaik

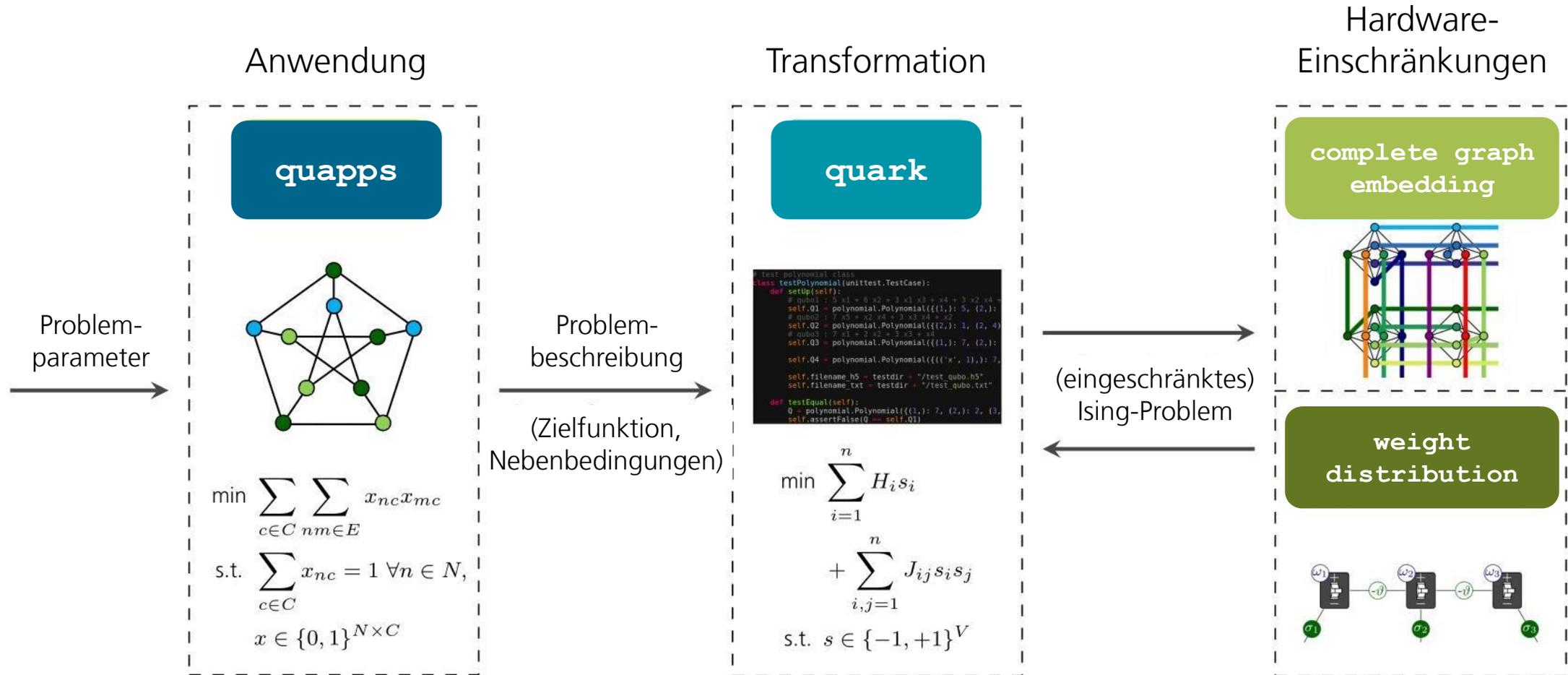
- Transmission Expansion Problem
- Multi-Roboter Faserbund-Leichtbau
- Beladungsoptimierung für Autoklavenprozesse

- Quantenzustandskompression von Hyperspektraldaten
- Berechnungsunsicherheiten von Gletschereis-Massebilanzen
- Systemmodellierung in der Solarenergieforschung

# Software-Schnittstelle für Kombinatorische Optimierung



<https://gitlab.com/quantum-computing-software/>



# Quantum Software Engineering

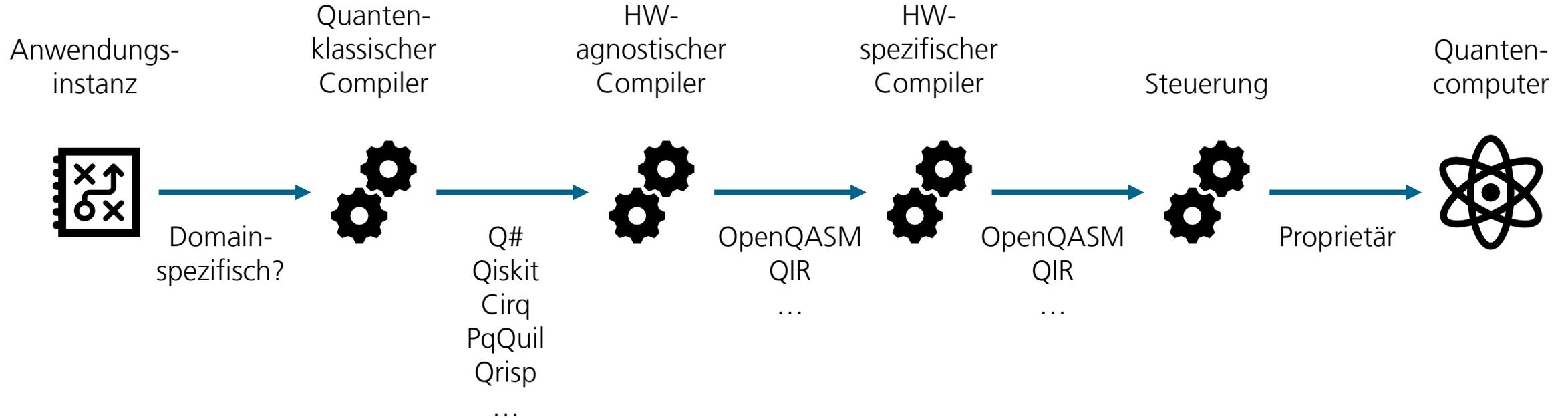


## Effiziente Entwicklung hochqualitativer Software mit wissenschaftlichen Methoden

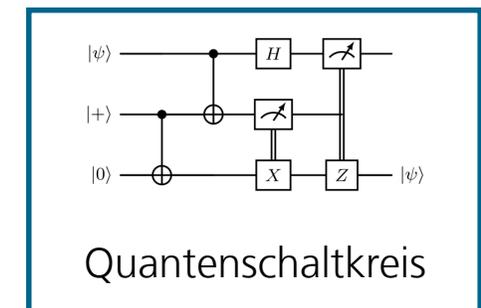
Requirements Engineering	Software Design	Modelle und Repräsentationen	Testen
<ul style="list-style-type: none"><li>▪ Involvieren von Domänenexperten</li><li>▪ Hardware setzt Grenzen</li></ul>	<ul style="list-style-type: none"><li>▪ hybride Formulierungen</li><li>▪ Interface-Definitionen</li><li>▪ Sicherstellen von Wiederverwertbarkeit</li><li>▪ Gerätecharakterisierung</li></ul>	<ul style="list-style-type: none"><li>▪ Schaltkreismodelle</li><li>▪ Intermediate Representations</li><li>▪ erste high-level Sprachen</li></ul>	<ul style="list-style-type: none"><li>▪ neue typische Fehler</li><li>▪ No-cloning Theorem und Messungen</li></ul>

- ➔ Quantencomputing unterscheidet sich von klassischer Software in wichtigen Aspekte
- ➔ Insbesondere im Umgang mit der statistischen Natur und individuellen Rauschcharakteristiken

# Modelle und Repräsentationen in der Kompilierung

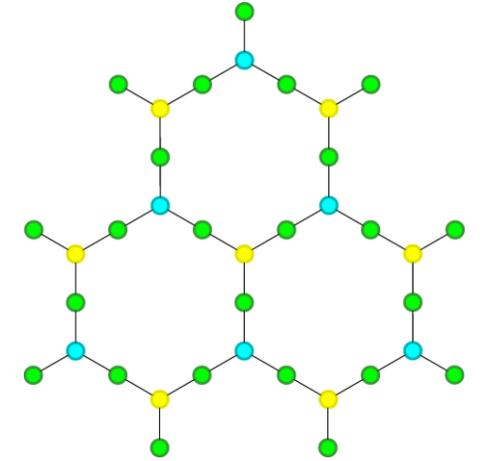


- Welche Grundoperationen (Quantengatter) sind verfügbar?
- Welche Fehlerrate haben einzelne Qubits oder Gatter?
- Welches Level an Parametrisierung habe ich?



# Kompilierung für Quantencomputing

- Anpassung an die jeweilige spezifische Hardware
  - **Transpiling**: Umformulierung von einer Beschreibung in eine andere
  - **Synthese**: Von der unitären Matrix zu einer Sequenz von Gattern
  - **Routing**: Umstrukturieren des Schaltkreises, um der Konnektivität der Hardware zu entsprechen
  - **Optimierung**: Reduktion der Gatterzahl, Minimierung des Rauschens
- Etablieren von Standards noch notwendig



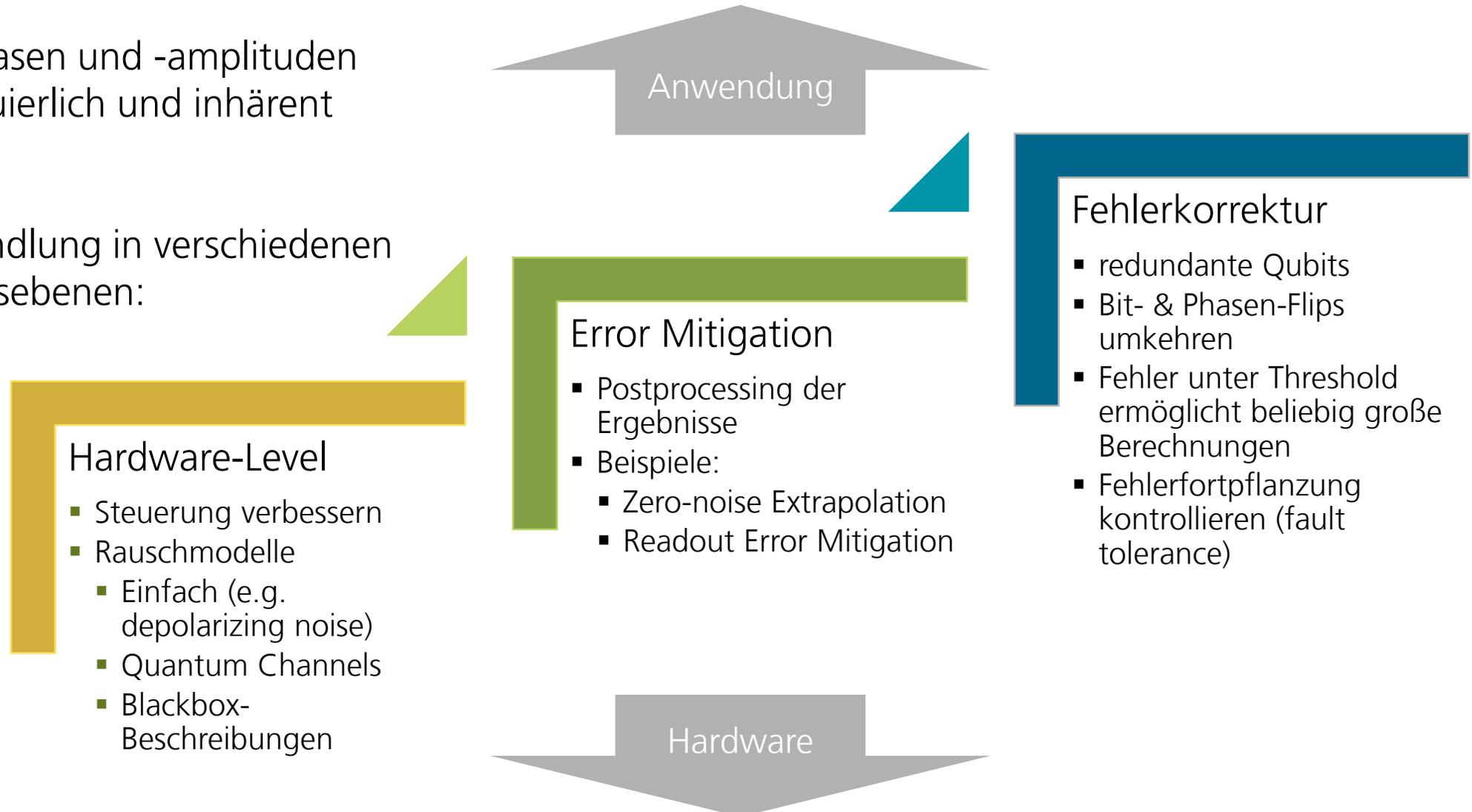
Heavy-Hexagon-Struktur des IBM Q Experience  
Credit: Jochen Szangolies

→ korrekter, effizienter, hardware-kompatibler Output für Quanten- und klassische Anteile

# Fehlerbehandlung

Quantenphasen und -amplituden sind kontinuierlich und inhärent instabil!

Fehlerbehandlung in verschiedenen Abstraktionsebenen:



# Quantensoftware Testen / Verifikation

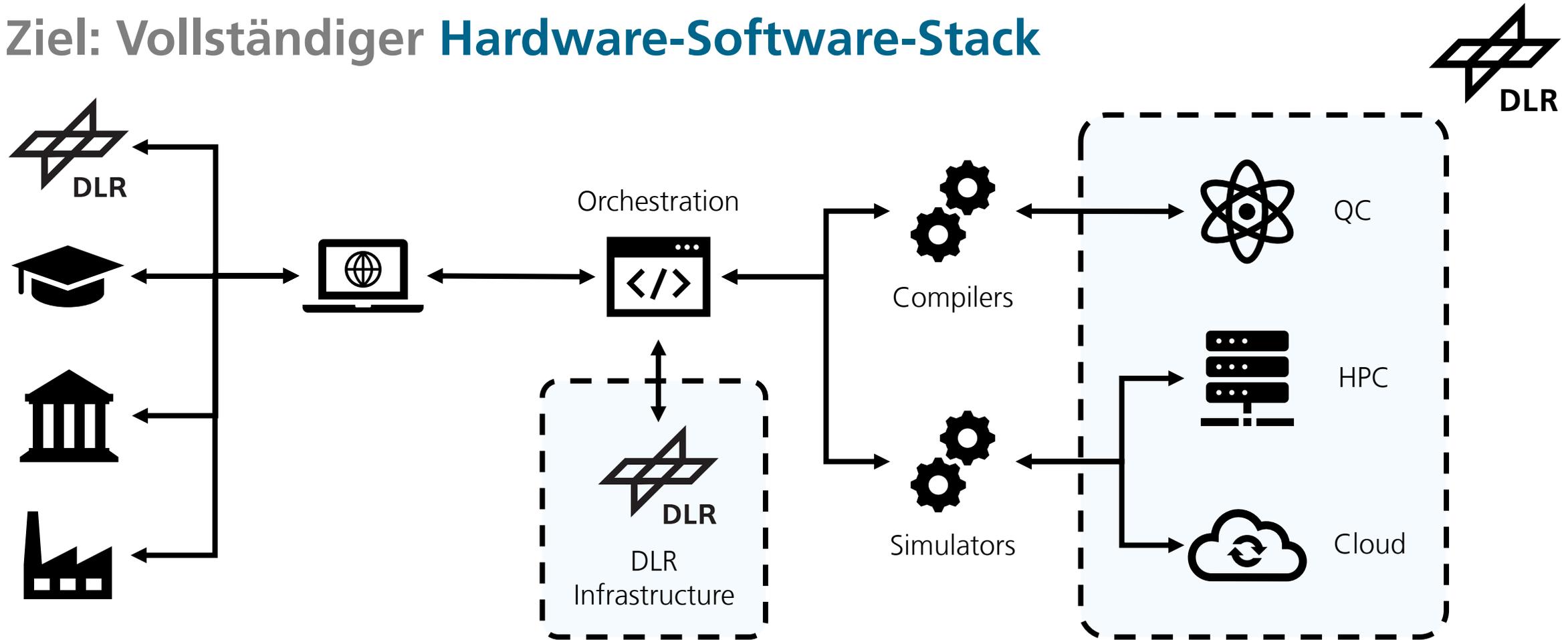


- Sicherstellen, dass die Software die Anforderungen erfüllt
- Ziel: hochqualitative Lösung (hohe Wahrscheinlichkeit das gewünschte Ergebnis zu erzielen)
- Challenge: Stochastische Natur + Rauschen + Gerätecharakteristiken
- Erforderliche Bestandteile:
  - Theoretischer Beweis: *der Quantenalgorithmus erfüllt Pre- und Postkonditionen*
  - Praktische Validierung: *der Code implementiert den gewünschten Algorithmus (Quanten- + klassischer Anteil)*
  - Funktionierende Toolchain: *der Code wird korrekt in einen ausführbaren Schaltkreis übersetzt + funktionierende Hardware*

→ Wissenschaftliche Fragestellung: Wenn Quantencomputer klassischen Rechnern überlegen sind, wie können wir die Richtigkeit überprüfen?

# DAS DLR-QC-ÖKOSYSTEM

# Ziel: Vollständiger Hardware-Software-Stack



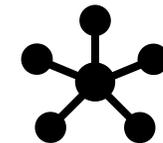
- Abbildung der vollständigen Kette von Anwender zu Hardware
- Integration in bestehende Architekturen durch langjährige HPC-Erfahrung
- Nutzbarmachung eines Quantenvorteils für Endanwender
- Integration von SW/HW-Herstellern und Anwendern

→ **QC-Hardware: nur ein kleiner Teil des Gesamtsystems**

# Systemanforderungen



- Intuitives Nutzer-Interface
  - Schnelle Iteration unterstützen
  - Ermöglichen einer transparenten und flexiblen Verbindung aus klassischem und Quantencomputing
- Orchestration und Datenmanagement
  - Organisieren klassischer und Quantenberechnungen
  - Balancieren der Nutzung von QC-, HPC-, Cloud-Ressourcen
  - Speichern von Messungsergebnissen unter Berücksichtigung der Datenlokalität
- Einbinden von verschiedenen Komponenten
  - Vereinheitlichter Zugang zu unterschiedlichen QC-Simulatoren
  - Balancieren von HPC- mit klassischen Berechnungen



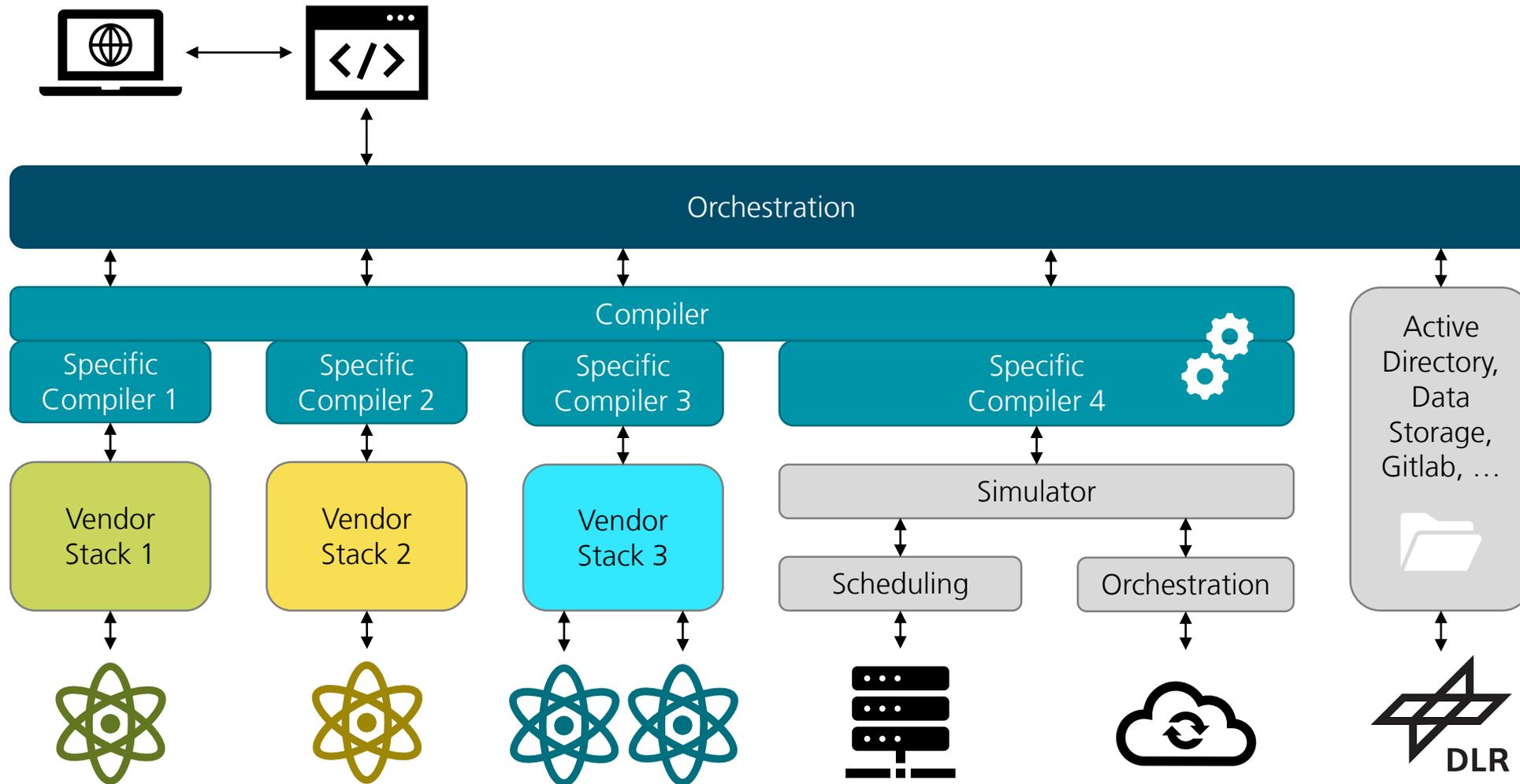
# Integrativer Quanten-Software-Stack

ALQU



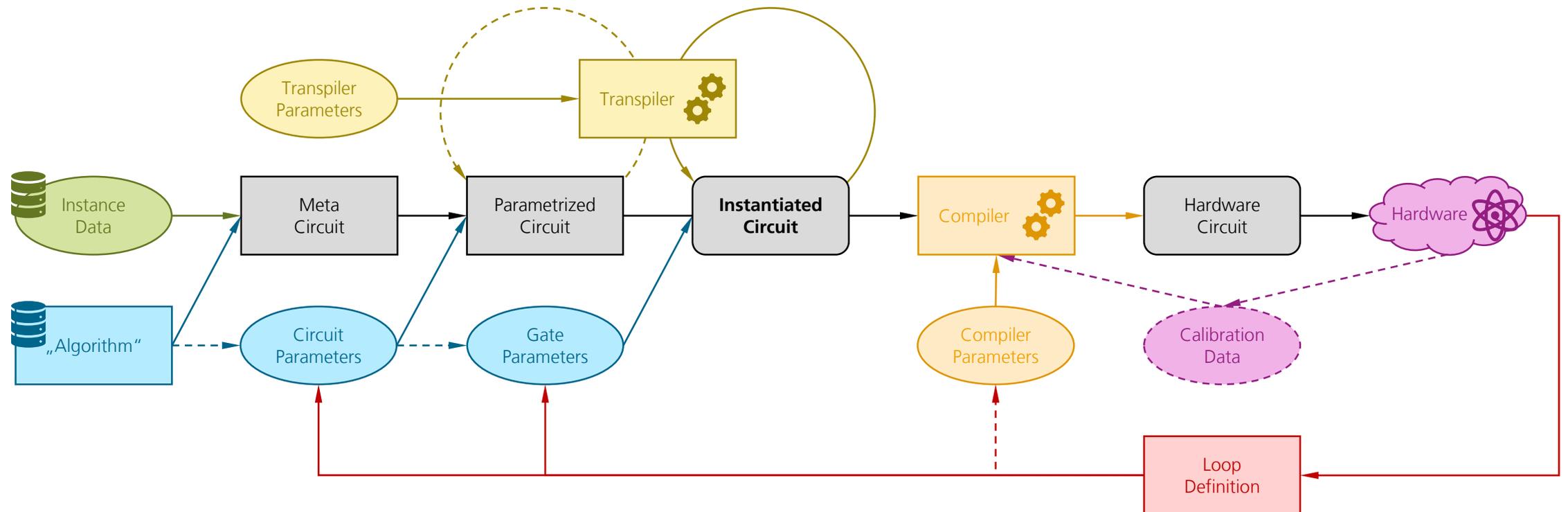
planqc

d-fine



# Schwierigkeit – Hybride Algorithmen

- Problem: Input-Output-Verhalten und Definition der Interaktivität mit klassischen Systemen
- Abbildung von Schleifen im Rahmen des Scheduling und der Orchestrierung



- Buch **Quantum Software - Aspects of Theory and System Design**

herausgegeben von Jaakov Exman (HIT, Israel), Ricardo Perez-Castillo (UCLM, Spanien), Mario Piattini (UCLM, Spanien), Michael Felderer (DLR-SC) mit Springer Nature als Open Access

- **Quantum Software Ecosystem Design**

Achim Basermann, Michael Epping, Benedikt Fauseweh, Michael Felderer, Elisabeth Lobe, Melven Röhrig-Zöllner, Gary Schmiedinghoff, Peter K. Schuhmacher, Yoshinta Setyawati, Alexander Weinert (DLR-SC)

- **Challenges for Quantum Software Engineering: An Industrial Application Scenario Perspective**

Cecilia Carbonelli (Infineon), Michael Felderer (DLR-SC), Matthias Jung (Uni Würzburg), Elisabeth Lobe (DLR-SC), Malte Lochau (Uni Siegen), Sebastian Lubber (Infineon), Wolfgang Mauerer (OTH Regensburg), Rudolf Ramler (SCCH), Ina Schaefer (KIT), Christoph Schroth (Fraunhofer IESE)

# Impressum



Thema: Aufbau eines Software-Ökosystems für Quantencomputer am DLR

Datum: 2023-04-26

Autor: Elisabeth Lobe

Institut: Institut für Softwaretechnologie (SC)  
Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)

Bildquellen: Alle Bilder „DLR (CC BY-NC-ND 3.0)“,  
sofern nicht anders angegeben