# ALQU: A DLR-QCI PROJECT

**ALgorithms for QUantum computer development using hardware-software-codesign**

DLR
Quantum Computing
Initiative

DLR

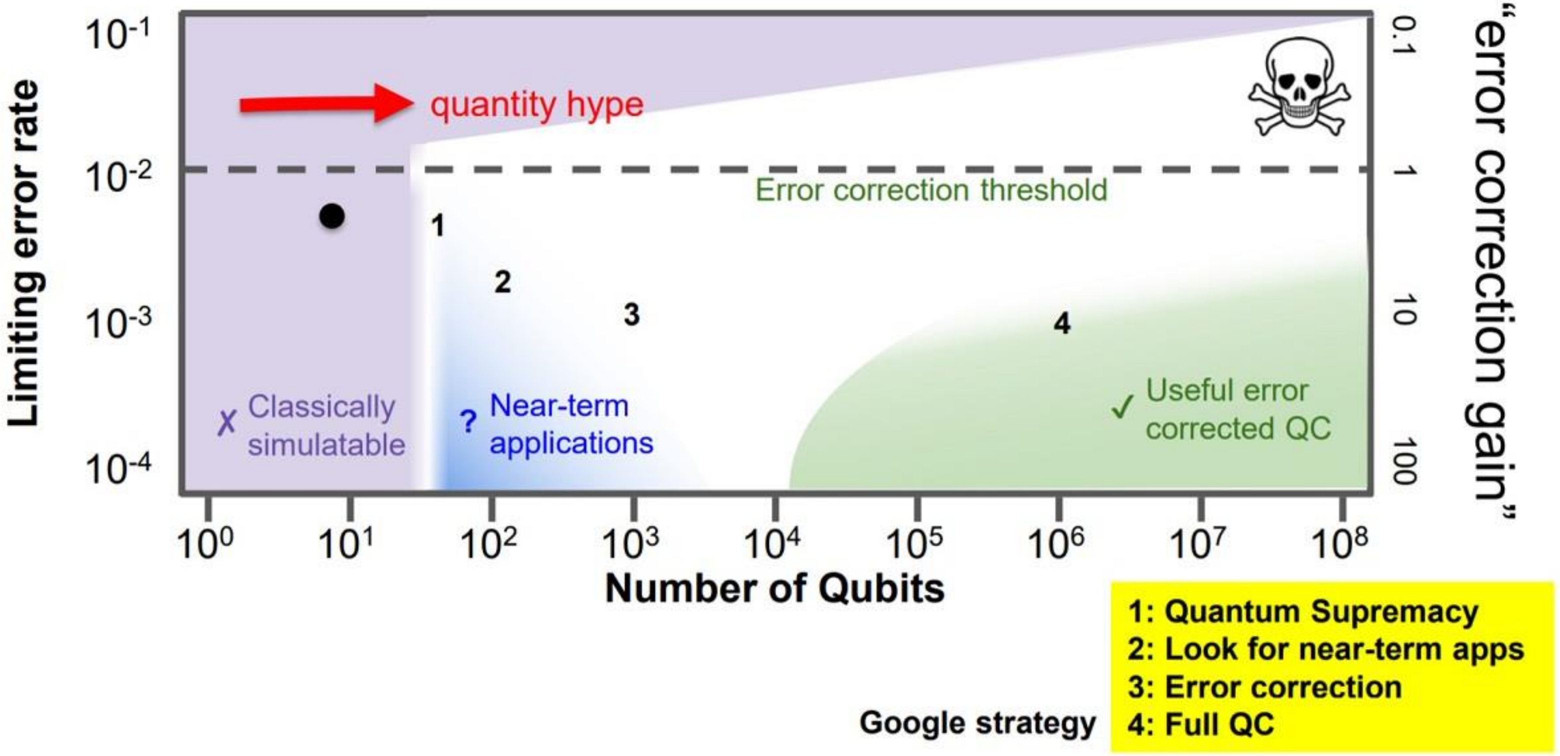
Project leader: Dr. Peter K. Schuhmacher, Institute for Software Technology (SC), 14.04.2023

# Background: State of Development of Quantum Computers

DLR

## Quantum Error Correction (QEC)

- Universal quantum computers are intrisically error-prone. To run killer apps like Shor‘s algorithm, we therefore need QEC.
- QEC relys on redundancy. Hence, it requires a significant overhead on reliable qubits and gates.



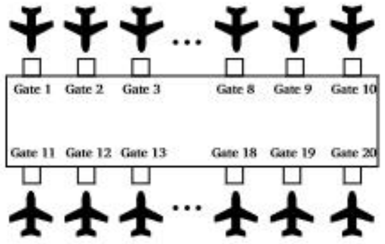

**If we want to achieve any quantum advantage within the near future, then we need to exploit the power of NISQ devices.**

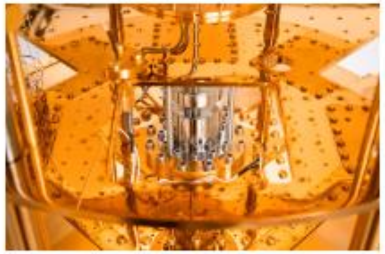


Figure: John Martinis, Google. (taken from https://medium.com/@quantum_wa/quantum-computing-near-and-far-term-opportunities-f8ffa83cc0c9 )

# Noisy Intermediate Scale Quantum Devices (NISQ)

- State-of-the-art quantum computers are rather small, noisy and have limited connectivity.
- Within the next years, we will be restricted to quantum computers **without QEC**.
- However, is it still possible to achieve quantum advantage?

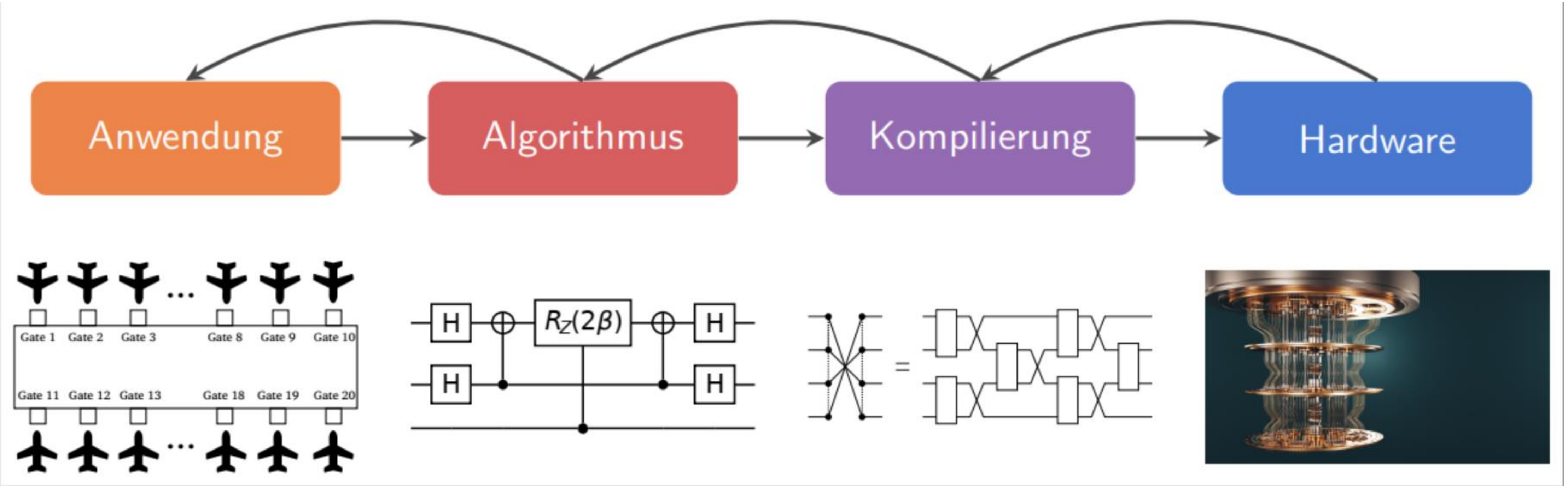



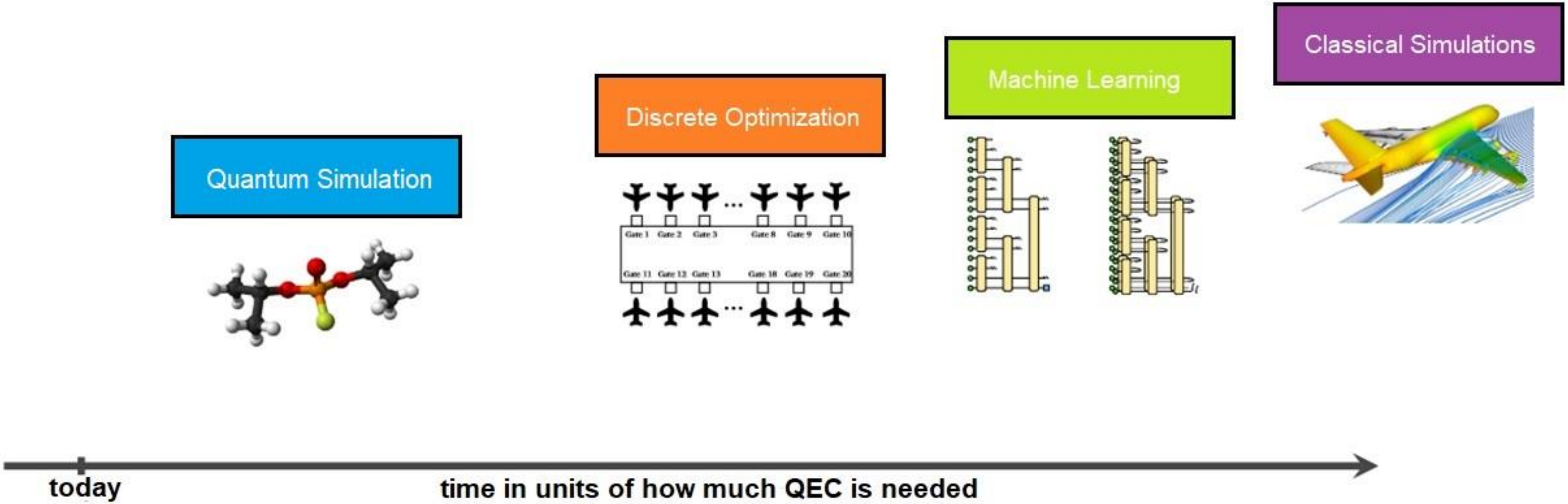
# Hardware-Software-Codesign

To persue the goal of quantum advantage, we should…

- … keep the hardware development in mind and
- … consider the error-prone and limited hardware within the software development process.



# What are promising applications for NISQ devices?*




*my own estimation without engagement

# ALQU: Work Packages

DLR

## HAP 1: Hardware focused

- Device aware compilation
- Control- and readout-algorithms

## HAP 2: Application focused

- Industrial scheduling problems
- Quantum simulation

## HAP 3: Software development

- Demonstrator
- Integration and test environment

| HAP 1: Fehlerbewusste Kompilierung und Ansteuerung von Quantencomputern | HAP 2: Anwendungsorientierte Algorithmenentwicklung im Hardware-Software-Codesign | HAP 3: Demonstrator / Integration und Testumgebung |
|---|---|---|
| AP 1.1: Algorithmen für fehlerbewusste Kompilierung | AP 2.1: Quantenalgorithmen für industr. Planungsprobleme | AP 3.1: Integration und Testumgebung |
| AP 1.2: Ansteuerungs- und Auslesealgorithmen f. Hardware | AP 2.2: Quantenalgorithmen für industr. Quantensimulation | AP 3.2: Demonstrator und experimentelle Studien |
| AP 1.3: Benchmarking und Software | AP 2.3: Quantenvorteil für fehlerbehaftete HW | |

# Industrial Contributions

| Industrial quantum software developer (IS) | Industrial end-user from the domain logistics / transport (IL) | Industrial end-user from the domain material science (IM) |
| --- | --- | --- |
| Supports the quantum compiler development | Delivers relevant use-cases from the domain of logistics and transport **(Discrete Optimization)** | Delivers relevant use-cases from the domain of material science **(Quantum Simulation)** |

Industrial hardware manufacturer for ion trap quantum computers (IH)

**No quantum computers ➡ no quantum computing!**