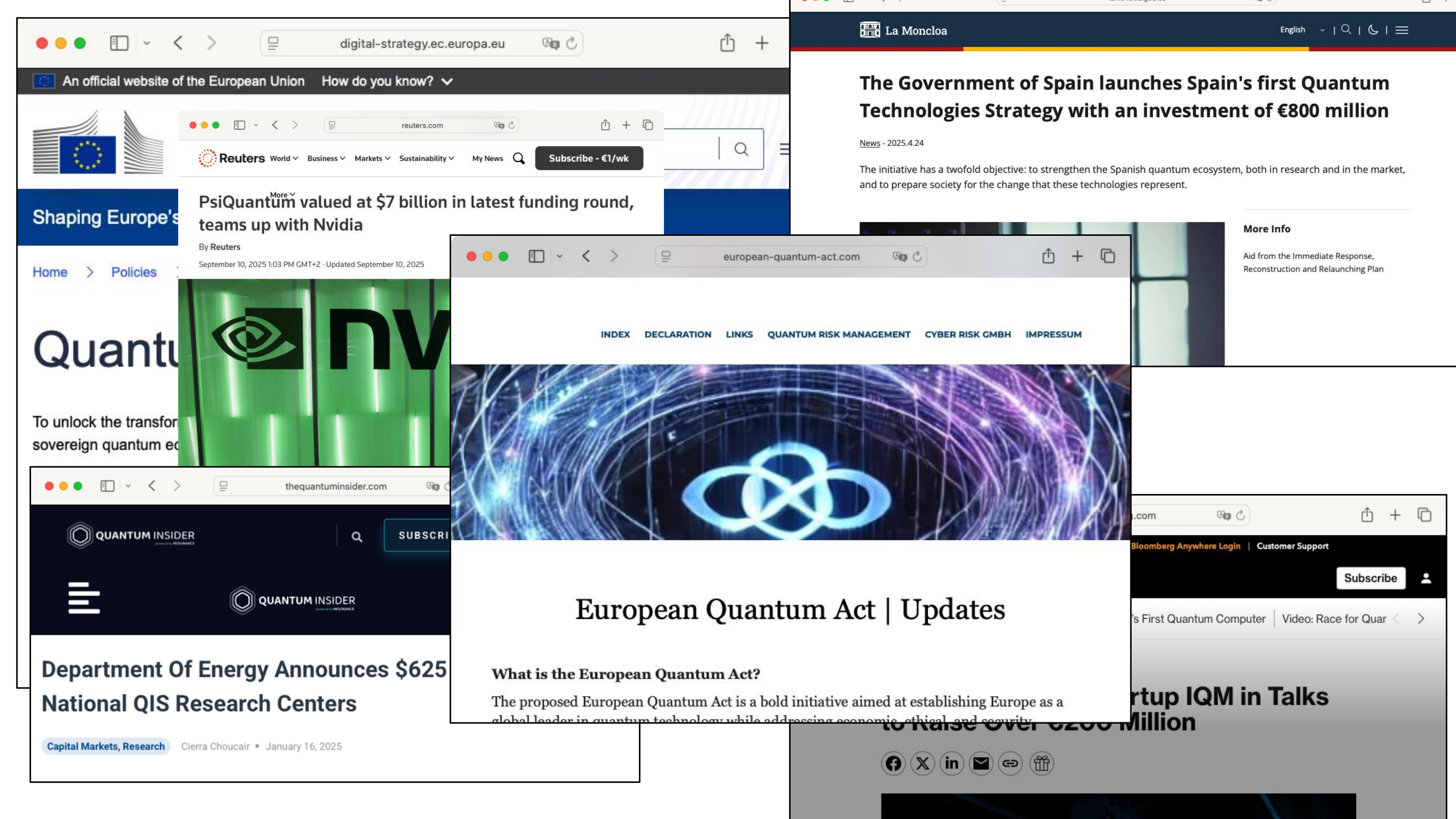
Quantum Software ante portas

How to prepare for a new computing paradigm?

Manuel Wimmer

Institute of Business Informatics - Software Engineering (BISE)

manuel.wimmer@jku.at


digital-strategy.ec.europa.eu
An official website of the European Union How do you know?
Shaping Europe's
Home > Policies
Quantu
To unlock the transfor
sovereign quantum ec
reuters.com
Reuters World Business Markets Sustainability My News Subscribe - €1/wk
More
PsiQuantum valued at $7 billion in latest funding round, teams up with Nvidia
By Reuters
September 10, 2025 1:03 PM GMT+2 · Updated September 10, 2025
La Moncloa
English
The Government of Spain launches Spain's first Quantum Technologies Strategy with an investment of €800 million
News - 2025.4.24
The initiative has a twofold objective: to strengthen the Spanish quantum ecosystem, both in research and in the market, and to prepare society for the change that these technologies represent.
More Info
Aid from the Immediate Response, Reconstruction and Relaunching Plan
european-quantum-act.com
INDEX DECLARATION LINKS QUANTUM RISK MANAGEMENT CYBER RISK GMBH IMPRESSUM
European Quantum Act | Updates
What is the European Quantum Act?
The proposed European Quantum Act is a bold initiative aimed at establishing Europe as a
thequantuminsider.com
QUANTUM INSIDER
SUBSCRI
Department Of Energy Announces $625
National QIS Research Centers
Capital Markets, Research
Cierra Choucair • January 16, 2025
.com
Bloomberg Anywhere Login | Customer Support
Subscribe
's First Quantum Computer | Video: Race for Quar
rtup IQM in Talks
to Raise Over €200 Million

What can we ***really*** expect from QC?

(unfortunately) Nothing New !

Every QC-solvable problem can be solved on a **classical machine** (with enough time).

Actually, we currently simulate QC.

But…

… for **some problems**, certain quantum algorithms have **exponentially better** runtime complexity.

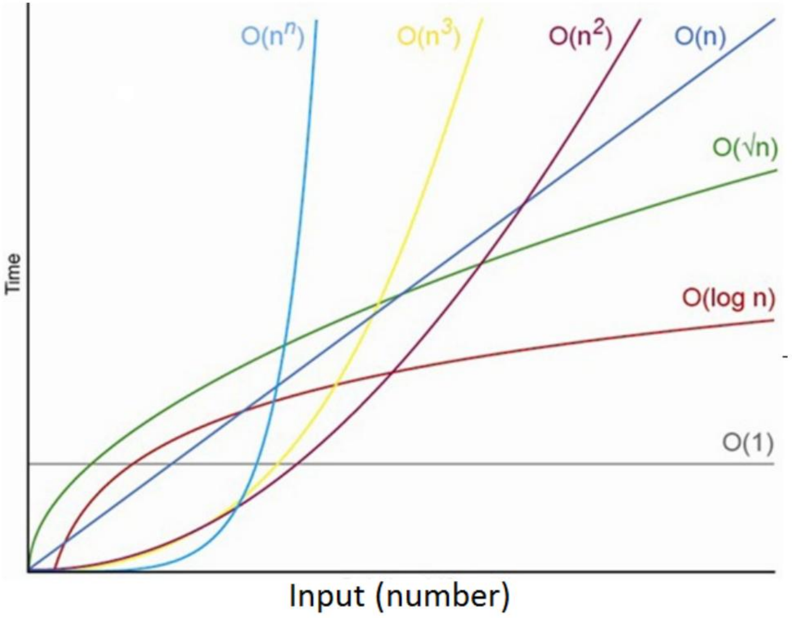
These algorithms are hard to find:

“*quantum computers are very difficult to reason about using classical intuition*”

Shor, Peter W. "Why haven't more quantum algorithms been found?" Journal of the ACM (JACM) 50.1 (2003): 87-90.

The Potential of QC

- Many complex problems are **intractable** for classical computing
  - **Exponentially growing** search spaces
- Classical computers **are reaching physical limitations**
- **Which problems** can quantum computers solve faster than classical computers can?
- **Aim: QC algorithms** which provide **polynomial/exponential speedup**



Source: Hidary (2019). Quantum Computing: An Applied Approach, Springer 2021

Where QC is expected to excel?

Quantum Simulation: Natural modeling of quantum systems (chemistry, materials, molecular reactions) by following Feynman’s original motivation.

Optimization Problems: Variational quantum algorithms (VQE, QAOA) and annealers tackle combinatorial or NP-hard problems.

Search and Sampling: Quadratic speed-up in unstructured search (Grover’s algorithm); quantum Monte Carlo for complex distributions.

Cryptanalysis: Factoring and discrete logarithms (Shor’s algorithm) → impact on RSA.

Machine Learning & Pattern Discovery: Quantum kernel methods, amplitude encoding, and quantum feature spaces for high-dimensional data.

Hybrid Workflows: Acceleration of subroutines (e.g., optimization or linear algebra) inside classical pipelines.

Example: Quantum Combinatorial Optimization

- **QCO** aims to find the set of discrete decision variables that minimize a certain constrained or unconstrained optimization function.
- **Problems** are usually non-convex (causing **many local optima**) and are often characterized by an **exponentially growing search space**

Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Benjamin Weder, Frank Leymann: Quantum Combinatorial Optimization in the NISQ Era: A Systematic Mapping Study. ACM Comput. Surv. 56(3), 2024

Example: Quantum Combinatorial Optimization

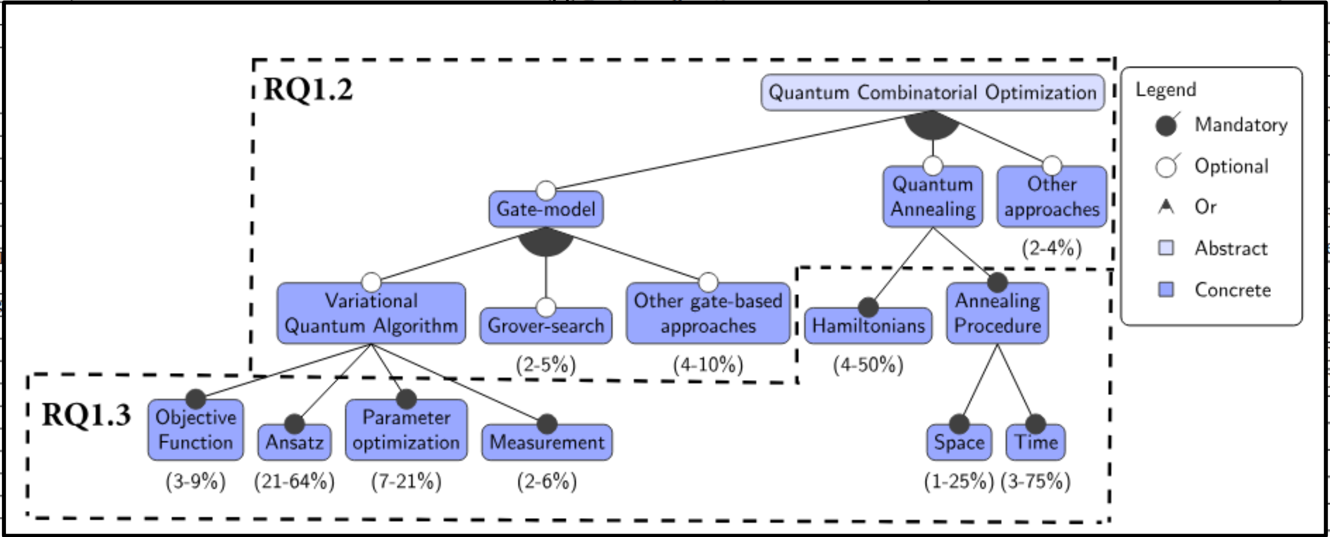
Standard Problem	Solution Approach
Graph Optimization	
Dense Subgraph Identification	Others: [24]
Graph Coloring	QAOA/VQE: [116, 134], QA: [113, 126, 134], Others: [118]
Graph Partitioning	QAOA/VQE: [117, 143], Others: [30, 139]
Graph Similarity	QAOA: [66], Others: [24]
k-Community Detection	QAOA/VQE: [89, 123], QA: [101, 115], Others: [74, 124, 139]
Maximum Clique	QA: [103, 110], Others: [24]
Max-Cut	QAOA/VQE: [5, 33, 48, 56, 60, 65, 81, 84, 92, 100, 121, 142, 154], [6, 10, 22, 23, 42, 51, 75, 76, 78, 125, 131, 137, 155] QA: [72], Others: [83, 94, 97, 151],
Maximum Independent Set	QAOA/VQE: [10, 100, 119, 120], QA: [148]
Minimum Multicut Problem	QA: [34]
Travelling Salesman Problem	QAOA/VQE: [96, 100], QA: [12, 45, 108, 109]
Vertex Cover	QAOA/VQE: [95, 96, 117]
Routing/Scheduling	
Job-Shop Scheduling	QA: [2, 156]
Network-flow Optimization (e.g., Routing)	QAOA/VQE: [153]
Vehicle Routing Problem	QAOA/VQE: [14, 61], QA: [2, 45, 59, 73]
Packing/Covering/Partitioning	
Consensus Clustering	QA: [32]
Knapsack Problem	QA: [45]
Linear Assignment Problem	QAOA/VQE: [112]
Market Split Problem	QAOA/VQE: [10, 100]
Minimum 2-sum Problem	Others: [90]
Number Partitioning	QAOA/VQE: [10, 78, 100]
Quadratic Knapsack Problem	QA: [149]
Set Packing	QAOA/VQE: [117]
Tiling Puzzle Problem	QA: [41]
SAT	
Max-2-SAT	QAOA/VQE: [87, 142, 143]
Max-3-SAT/3-SAT	QAOA/VQE: [10, 100], QA: [27, 49, 80, 88]
SAT Problems	QA: [9]
Diversification	
Max-sum Diversification	QA: [13]
Portfolio Optimization	QAOA/VQE: [10, 42, 78], QA: [55, 140] Others: [52]
No Category	
Quadratic Assignment Problem	Others: [90]

Real-world Problem/Domain	Standard Problem	Solution Approach
Graph Optimization		
Air Traffic Management	Conflict-resolution Problem	QA: [129]
Flight scheduling Frequency allocation Register allocation	Graph Coloring	QAOA/VQE: [104]
Smart charging electric vehicles	Max-k-Cut Maximum Independent Set	QAOA/VQE: [35]
Cluster head selection Wireless scheduling Satellite scheduling	Maximum Weighted Independent Set	QAOA/VQE: [28, 29, 77]
Satellite sub-constellation assignment	Weighted k-Clique Problem	QA: [12]
Routing/Scheduling		
Social workers problem	Combination of Vehicle Routing Problem and Scheduling Problem	QAOA/VQE: [1] Others: [8]
Military maintenance	Job-Shop Scheduling	QA: [12]
Robot routing	Routing	QA: [31, 106]
Binary paint shop problem	Scheduling	QAOA/VQE: [132]
Flight-gate assignment problem	Scheduling	QAOA/VQE: [127]
Multiple processor scheduling	Scheduling	QAOA/VQE: [117]
Nurse scheduling	Scheduling	QA: [69]
Railway dispatching problem	Scheduling	QA: [40]
Traffic flow optimization	Scheduling	QA: [67, 70, 147]
Transaction scheduling	Scheduling	QA: [17, 18]
Workflow scheduling	Scheduling	QA: [138]
Packing/Covering/Partitioning		
Tail assignment problem	Exact Cover Set Partitioning	Others: [133]
No Category		
Facility location allocation	Quadratic Assignment Problem	QAOA/VQE: [112] QA: [3]
Garden optimization problem	Quadratic Assigment Problem	QA: [25]
Item listing optimization problem	Quadratic Assignment Problem	QA: [102]

Example: Quantum Combinatorial Optimization

Standard Problem	Solution Approach
Graph Optimization	
Dense Subgraph Identification	Others: [24]
Graph Coloring	QAOA/VQE: [116, 134], QA: [113, 126, 134], Others: [118]
Graph Partitioning	QAOA/VQE: [117, 143], Others: [30, 139]
Graph Similarity	QAOA: [66], Others: [24]
k-Community Detection	
Maximum Clique	
Max-Cut	
Maximum Independent Set	
Minimum Multicut Problem	
Travelling Salesman Problem	
Vertex Cover	
Routing/Scheduling	
Job-Shop Scheduling	
Network-flow Optimization (e.g.,	
Vehicle Routing Problem	
Packing/Covering/Partitioning	
Consensus Clustering	
Knapsack Problem	
Linear Assignment Problem	
Market Split Problem	
Minimum 2-sum Problem	
Number Partitioning	
Quadratic Knapsack Problem	
Set Packing	
Tiling Puzzle Problem	
SAT	
Max-2-SAT	QAOA/VQE: [87, 142, 143]
Max-3-SAT/3-SAT	QAOA/VQE: [10, 100], QA: [27, 49, 80, 88]
SAT Problems	QA: [9]
Diversification	
Max-sum Diversification	QA: [13]
Portfolio Optimization	QAOA/VQE: [10, 42, 78], QA: [55, 140] Others: [52]
No Category	
Quadratic Assignment Problem	Others: [90]

Real-world Problem/Domain	Standard Problem	Solution Approach
Graph Optimization		
Air Traffic Management	Conflict-resolution Problem	QA: [129]
Flight scheduling Frequency allocation	Graph Coloring	QAOA/VQE: [104]
		A/VQE: [35]
		A/VQE: [28, 29, 77]
		[12]
		A/VQE: [1]
		ers: [8]
		[12]
		[31, 106]
		A/VQE: [132]
		A/VQE: [127]
		A/VQE: [117]
		[69]
		[40]
		[67, 70, 147]
		[17, 18]
		[138]
Packing/Covering/Partitioning		
Tail assignment problem	Exact Cover Set Partitioning	Others: [133]
No Category		
Facility location allocation	Quadratic Assignment Problem	QAOA/VQE: [112] QA: [3]
Garden optimization problem	Quadratic Assigment Problem	QA: [25]
Item listing optimization problem	Quadratic Assignment Problem	QA: [102]



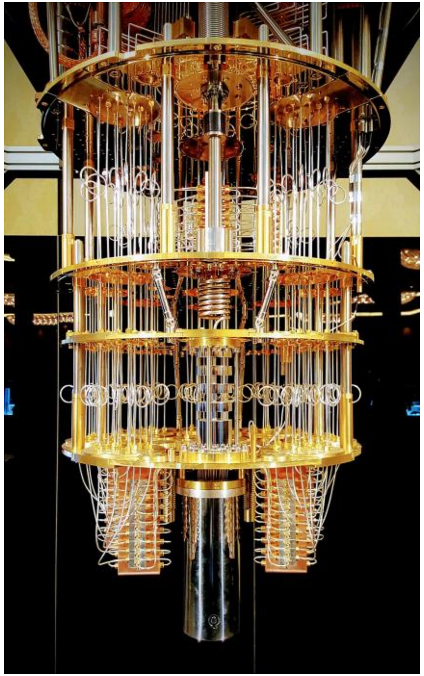
However…

Current Quantum Computers are…

- Room-filling machines
- Expensive
- Prone to error

Software running on these machines is…

- Often hardware-specific
- Low-level and complex
- Usually developed by physicists or mathematicians.

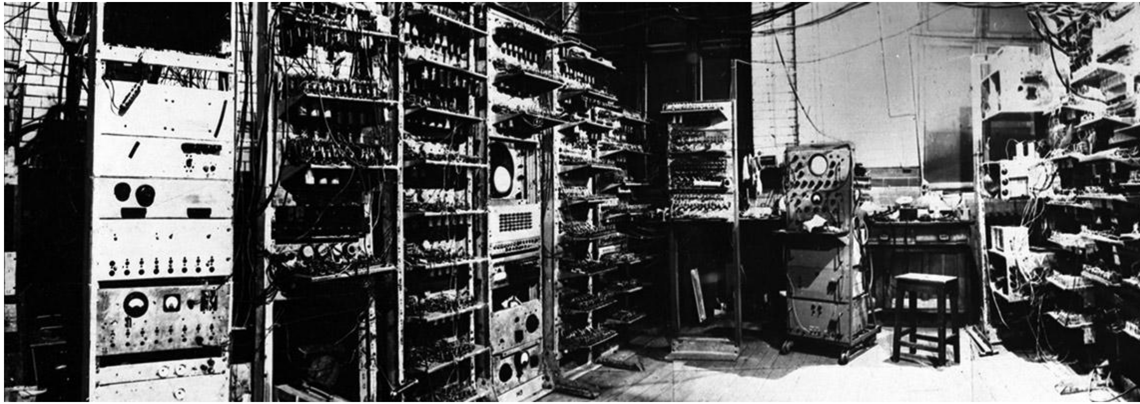


IBM Q: 50-qubit quantum computer
(*photo* © Lars Plougman)

Moguel, Enrique, et al. "A Roadmap for Quantum Software Engineering: Applying the Lessons Learned from the Classics." *Q-SET@QCE*. 2020.

However…

Small-Scale Experimental Machine



https://www.cs.manchester.ac.uk/about/history-and-heritage/

Early *computers* were…

- Room-filling machines
- Expensive
- Prone to error.

Software running on these machines was…

- Often hardware-specific
- Low-level and complex
- Usually developed by physicists or mathematicians.

JKU JOHANNES KEPLER UNIVERSITY LINZ

Moguel, Enrique, et al. "A Roadmap for Quantum Software Engineering: Applying the Lessons Learned from the Classics." *Q-SET@QCE*. 2020.

We’ve been there before!

- **QC** faces similar challenges **classical computing** faced in the mid of the last century
- Back then, a new profession emerged to solve these challenges
- People built abstractions & languages, invented techniques & tools
  - ➢ Software engineering discipline
- **QC is coming** with fundamentally different working principles.
  - ➢ Let’s prepare for it!

BISE @ JKU Linz

JOHANNES KEPLER
UNIVERSITY LINZ

BISE @ JKU Linz

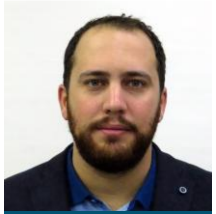
Christian Doppler Forschungsgesellschaft

Starting Point 2019

Birgit Breitschopf
Administration

Dominik Lamprecht
Software Technician

Felix Gemeinhardt
Pre-Doc



Antonio Garmendia
Post-Doc

Manuel Wimmer
Head of Institute

Iris Groher
Associate Professor

Christoph Stein
Pre-Doc

Stefan Klikovits
Post-Doc

Wolfgang Narzt
University Assistant

Luca Berardinelli
Post-Doc

Martin Eisenberg
Researcher
Pre-Doc

Sabine Sint
Researcher
Pre-Doc

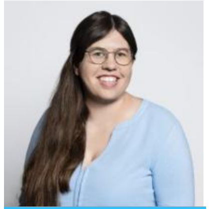
Bernhard Schenkenfelder
SCCH, Pre-Doc

Johannes Sametinger
Associate Professor

Rainer Weinreich
Assistant Professor

Reinhold Plösch
Associate Professor

Hermann Sikora
Honorary Professor



Bianca Wiesmayr
Post-Doc

Markus Feneberger
Researcher
Siemens

Marco Stadler
Researcher
LIT Secure

Abbas Rahimi
LIT, Pre-Doc

BISE @ JKU Linz

Foundations

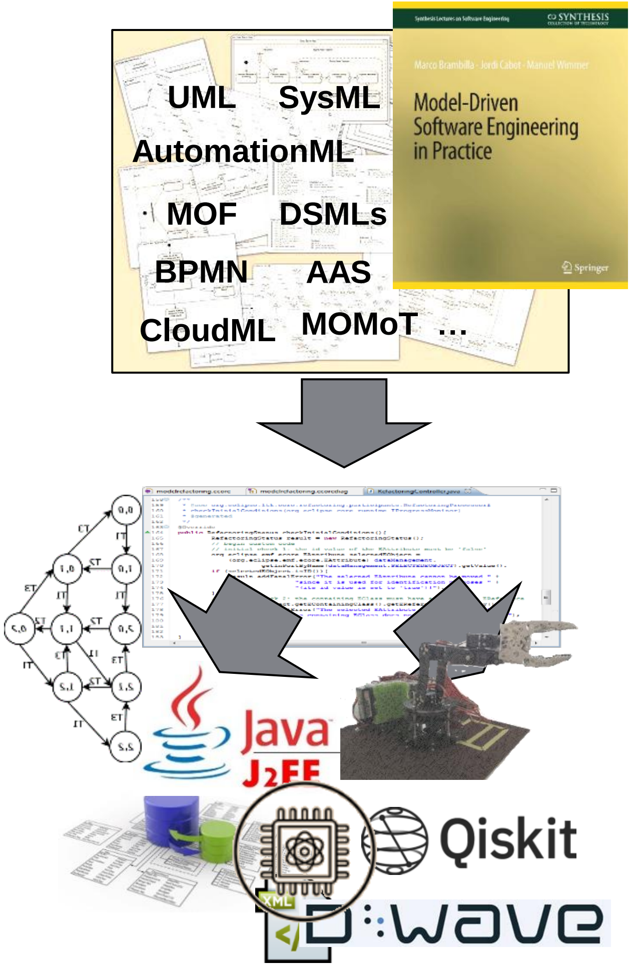
- **Languages**: UML, SysML, AutomationML, DSMLs, …
- **Methods**: MDSE, DSM, MPM, ….
- **Techniques**: transformation, code generation, tracing, …
- **Management**: evolution, versioning, variants, …
- **Tools**: editors, repositories, validators, engines, **AIware**, …

Applications

- **Software**: Web, Cloud, Databases, Workflows, IoT, …
- **Systems**: Industry 4.0/5.0, BIM, Tunneling, …
- **Emerging Paradigms**: AI, Digital Twins, **Quantum Computing**

Quantum Computing

- **Applications of QC** (k-community detection, graph matching, ...)
- **Quantum Software Engineering** (read SE 4 QC)



What to expect from the rest of this talk?

- **Quick outline of QC**
  - How does the technology work?
  - What can it do?
- **Quantum Software in addition to Classical Software**
  - Where are we?
  - Current challenges
- **How to approach Quantum Software research?**
  - Some insights how we approach this area
- **What's next?**

Quantum Technologies: *What are we speaking about?*

Quantum Communication

Quantum Sensing

Quantum Computing

Quantum Software is required for all these technologies, but especially for **Quantum Computing**

What is Quantum Computing?

Quantum computers process information using the **laws of quantum physics**,

not classical Boolean logic.

While a classical computer is in one of 2^N states at a time (where N is number of bits),
a **quantum computer** is in **probability distribution of 2^N states**.

Programming a QC is not about transforming one state through operations a computer can perform faster than a human. It is about manipulating a **distribution over all possible solutions** to a problem in a meaningful way.

Basic Concepts - Superposition

Classic bits: two states (0, 1)

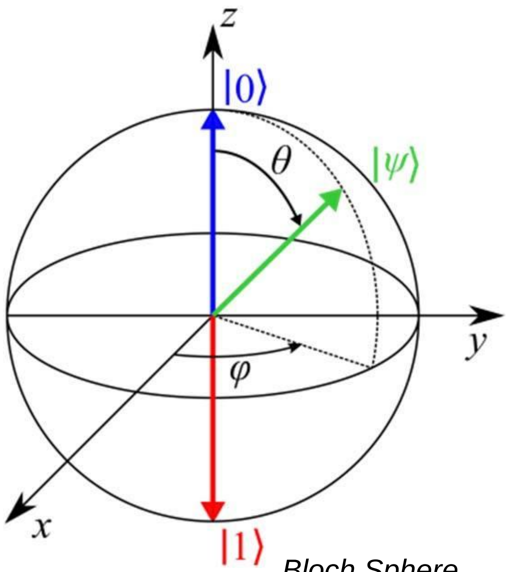
Quantum bits (Qubits): represent a **probability distribution** between 0 and 1

$|\psi\rangle = \alpha * |0\rangle + \beta * |1\rangle$, where $|\alpha|^2 + |\beta|^2 = 1$

Similar to classical bit 0,1 → $|0\rangle, |1\rangle$

Can also be a mixture → **superposition**

Upon measurement, states *collapse* to 0 or 1



Bloch Sphere

Basic Concepts - Entanglement

Correlation between **probability distributions of qubits**

- Probabilities of two qubits can be **entangled**

E.g.,: **Bell States** (completely entangled):

- $|\Psi_+\rangle = \frac{\mathbf{1}}{\sqrt{\mathbf{2}}}|\mathbf{00}\rangle + \frac{\mathbf{1}}{\sqrt{\mathbf{2}}}|\mathbf{11}\rangle$ ➜

  50% probability of measuring |00>
  50% probability of measuring |11>
  0% probability of measuring |01> or |10>

- When **one qubit** is **measured**, the shared superposition collapses.
  Right after measurement, **both qubits** have the **same value**

Entanglement is an important building block for achieving speed-ups in quantum algorithms

Side-note: Linking two quantum computers through entanglement gives exponential speedup

Jozsa, R., & Linden, N. (2003). On the Role of Entanglement in Quantum-Computational Speed-Up. *Mathematical, Physical and Engineering Sciences*, *459*(2036).

Basic Concepts - Quantum Parallelism

Because our system is in a **multitude of states** at **once** (superposition),

a **single operation** can **affect all** of these **states** at **once**.

There is no free lunch!

Due to the laws of quantum physics, **quantum operations have to be reversible**.

Most operations from **classical computing** are **not reversible** (e.g., AND, OR, …), and therefore, cannot directly be executed on a QC.

Basic Concepts - Measurement

Measurement destroys superposition

- Non-reversible quantum operation

Intermediate states of the quantum system are **not** accessible

Copying is not possible, see no-cloning theorem

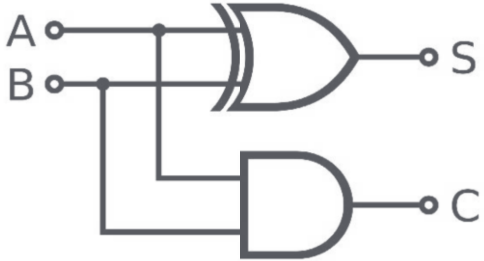
- To **reproduce** states, **repeated** state computation and measurement required

Wootters, W., Zurek, W. A single quantum cannot be cloned. Nature 299, 802–803 (1982)

What does a quantum program look like?

Recall: Classical Logic Circuit Diagram

Input	Computation	Output



Bits	Logic Gates (And/Or/Xor/...)	Bits

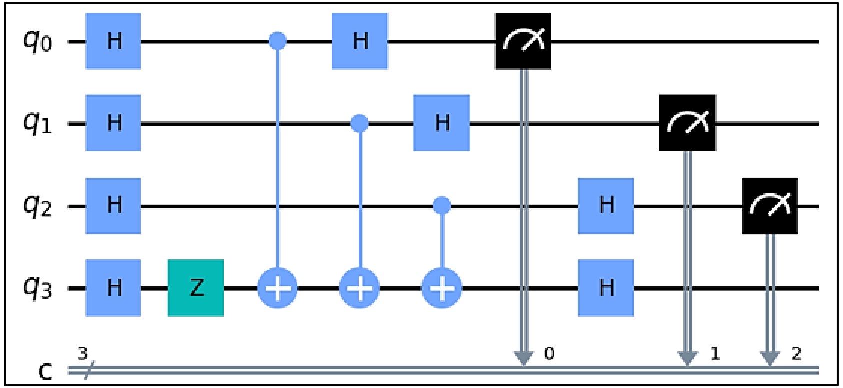
What does a quantum program look like?

Current Status: Quantum Circuit

Initialization Computation Measurements

Input/
Output

qubits



Quantum Operations
Gate Set {X, Z, H, …}

Quantum Gate

actions on the qubits

- **X (Pauli-X):** bit flip
- **Z (Pauli-Z):** phase flip
- **H (Hadamard):** creates superposition
- …

A **unitary matrix** *U* acting on a qubit

$$|\psi'\rangle = U|\psi\rangle$$

Gates are <u>reversible</u> (important property)

Typical Problem:
"Create a circuit that does …"

JOHANNES KEPLER UNIVERSITY LINZ

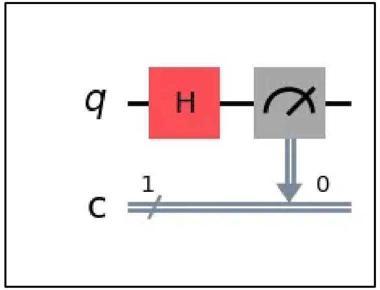
What does a quantum program look like?

Current Status: Quantum Circuit

Initialization **Computation** **Measurements**

Input/ Output

qubits



Quantum Operations
Gate Set {X, Z, H, …}

Quantum Gate

actions on the qubits

- **X (Pauli-X):** bit flip
- **Z (Pauli-Z):** phase flip
- **H (Hadamard):** creates superposition
- …

A **unitary matrix** *U* acting on a qubit

$$|\psi'\rangle = U|\psi\rangle$$

Gates are <u>reversible</u> (important property)

Typical Problem:
"Create a circuit that does …"

Current Limitations of Quantum Computing

- **Multitude of technical challenges**
  - Fragile qubits
  - Inaccuracy of quantum operations
  - …

- **Main paradigms**
  - Noisy Intermediate Scale Quantum (NISQ-era)
  - Fault-tolerant Quantum Computing

Current Era: NISQ
“Noisy Intermediate-Scale Quantum”

NISQ = *Noisy Intermediate-Scale Quantum* (term coined by John Preskill, 2018).

Size **50–1000 physical qubits**;

- Error-prone, short coherence times; no full error correction

Goal: Demonstrate *quantum advantage* for specialized tasks before large-scale fault-tolerant systems exist

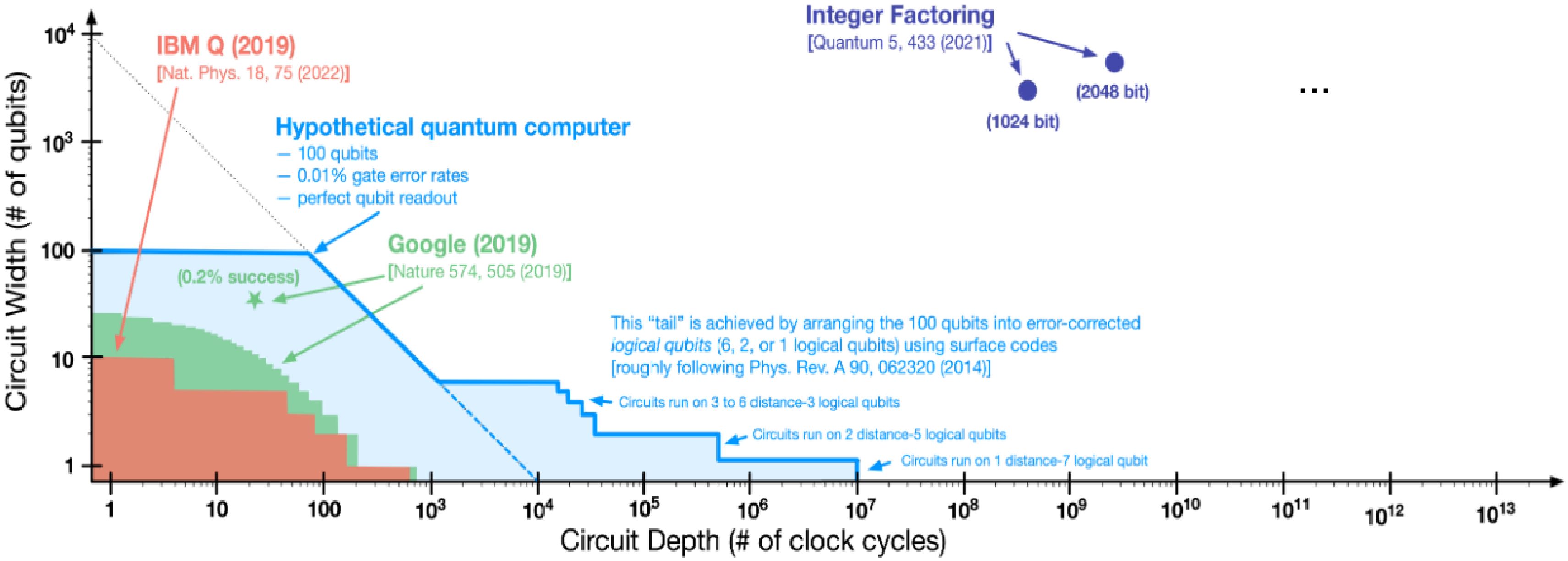
Approach: Use hybrid algorithms (VQE, QAOA, HQCC) — classical optimization controlling quantum circuits

Limitations: Noise limits circuit depth; results are probabilistic and hardware-specific

Software Challenge: Optimize and mitigate errors through compilation, calibration, and classical post-processing

Preskill, John. "Quantum computing in the NISQ era and beyond." Quantum 2 (2018)

Quantum Computing: Where are we?



JOHANNES KEPLER UNIVERSITY LINZ

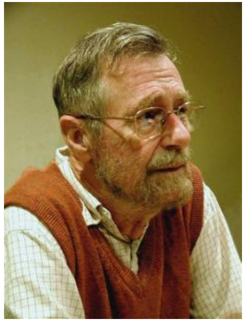
Proctor *et al.* Benchmarking quantum computers. *Nat Rev Phys* **7**, 105–118 (2025).

Example Technology Roadmap: IBM

JOHANNES KEPLER UNIVERSITY LINZ

https://www.ibm.com/quantum/technology

The Challenges Ahead



(*photo © Hamilton Richards*)

Edsger Dijkstra on the early challenges of software:

"*As long as there were **no machines**, **programming** was **no problem** at all;*

*when we had a few **weak computers**, **programming** became a **mild problem**,*

*And now we have **gigantic computers**, **programming** has become an **equally gigantic problem**.*"

(Excerpt from his 1972 Turing Award Lecture)

Complex Computing Infrastructure

How to specify?

Where can we find algorithms and guidelines how to select/ realize them?

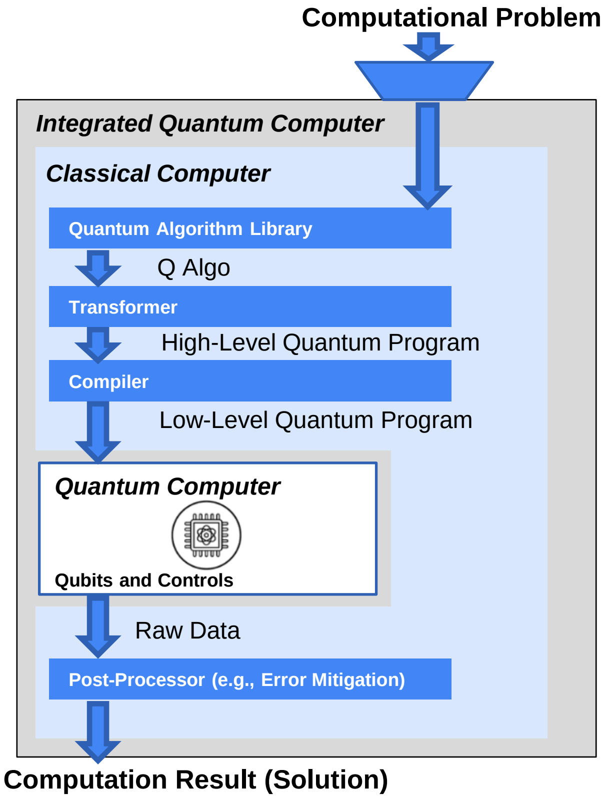
How to best program a QC?

What is the best deployment?

What is the best QC for a given problem?

What is the quality of the computed solution?

How to integrate the results in a larger system?



Requirements Engineer

Domain Expert

Algorithm Specialist

Quantum Programmer

Quantum Tester

QC Technology Expert

Quantum Physicist

Data Analyst

Software Engineer

JOHANNES KEPLER UNIVERSITY LINZ

Integrated Quantum Computer Illustration based on Proctor *et al.* Benchmarking quantum computers. *Nat Rev Phys* **7**, 105–118 (2025).

Abstraction levels

Classical Software

- Models, No-Code, 4GL, 5GL, …
- Python + Frameworks
- Java, …: objects
- C++: objects
- C: arrays, data types, …
- Assembler Language
- Machine language (00100101…)

Quantum Software

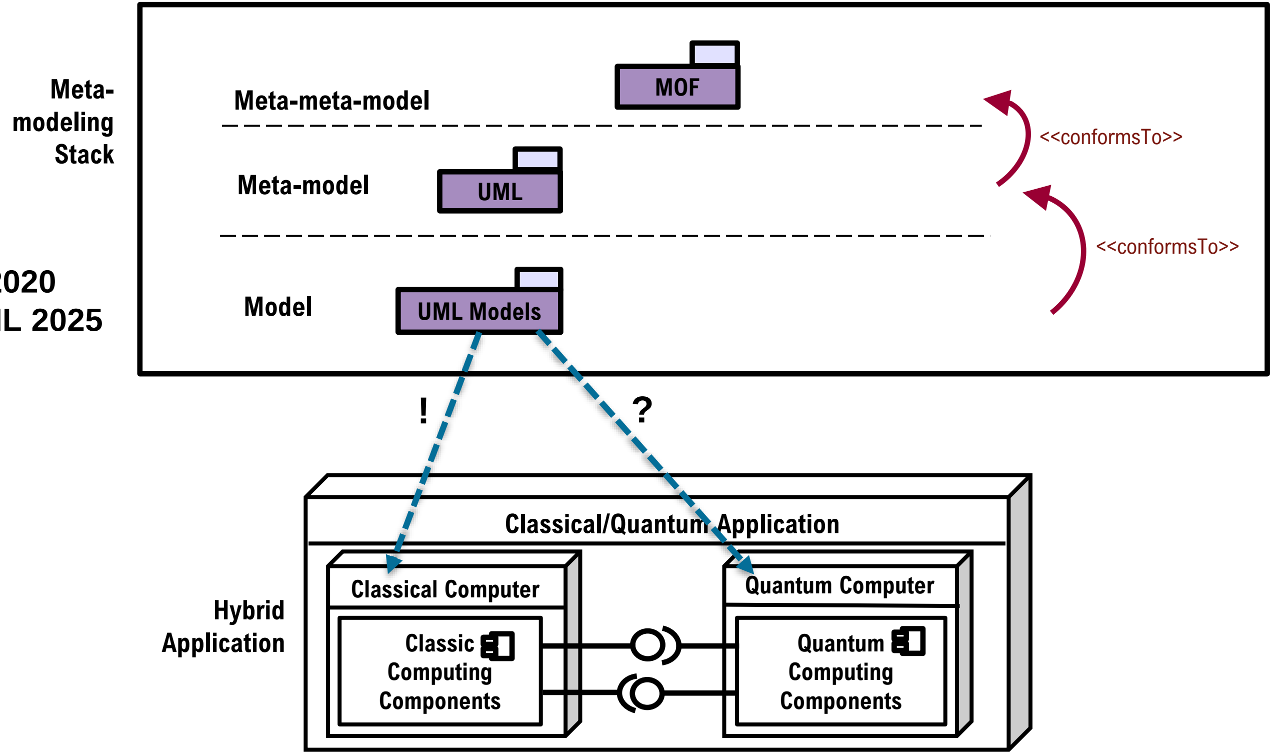
- *?Models, No-Code, 4GL, 5GL?*
- IBM Quantum Composer (qubits)
- Python + Frameworks (qubits)
- …
- Circuits before transpilation (qubits)
- Circuits after transpilation (qubits)
- Pulse-level, …

→ same abstraction level (qubits + gates)

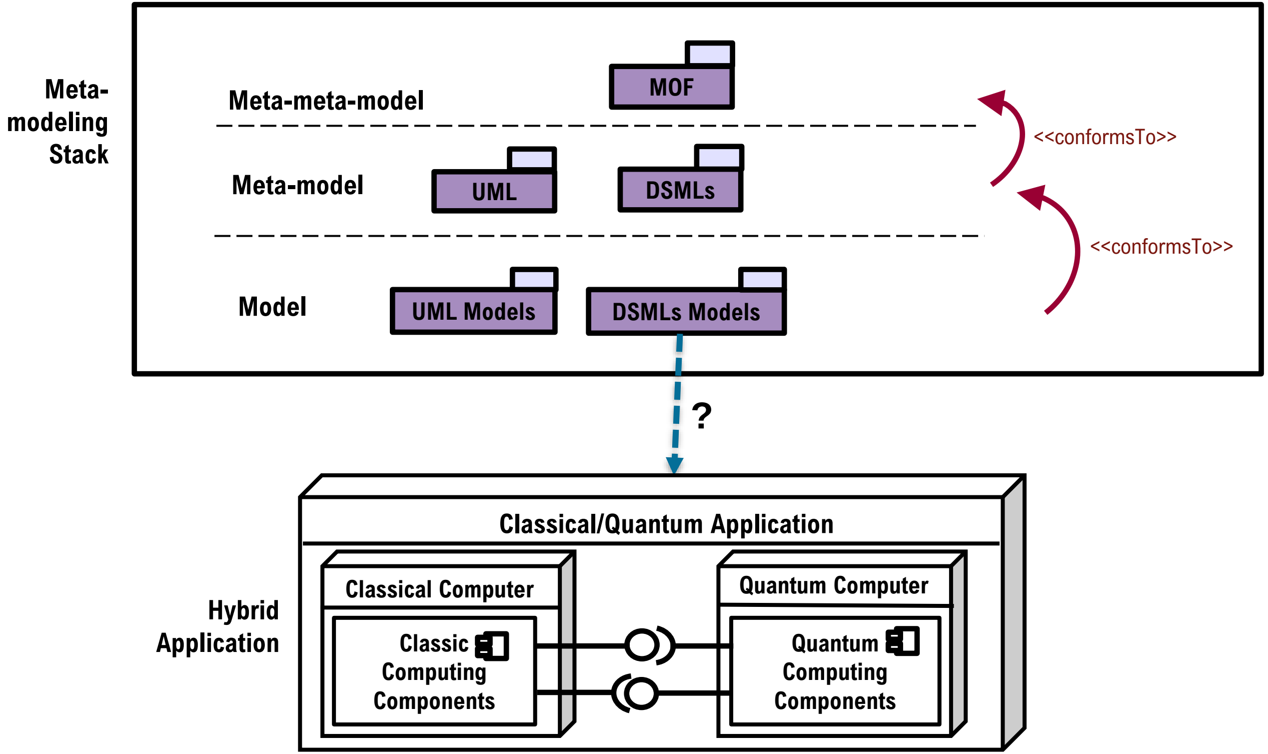
O. Di Matteo et al. "An Abstraction Hierarchy Toward Productive Quantum Programming," *IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2024.

And what about Modeling Languages?

Q-UML 2020
QuanUML 2025
…

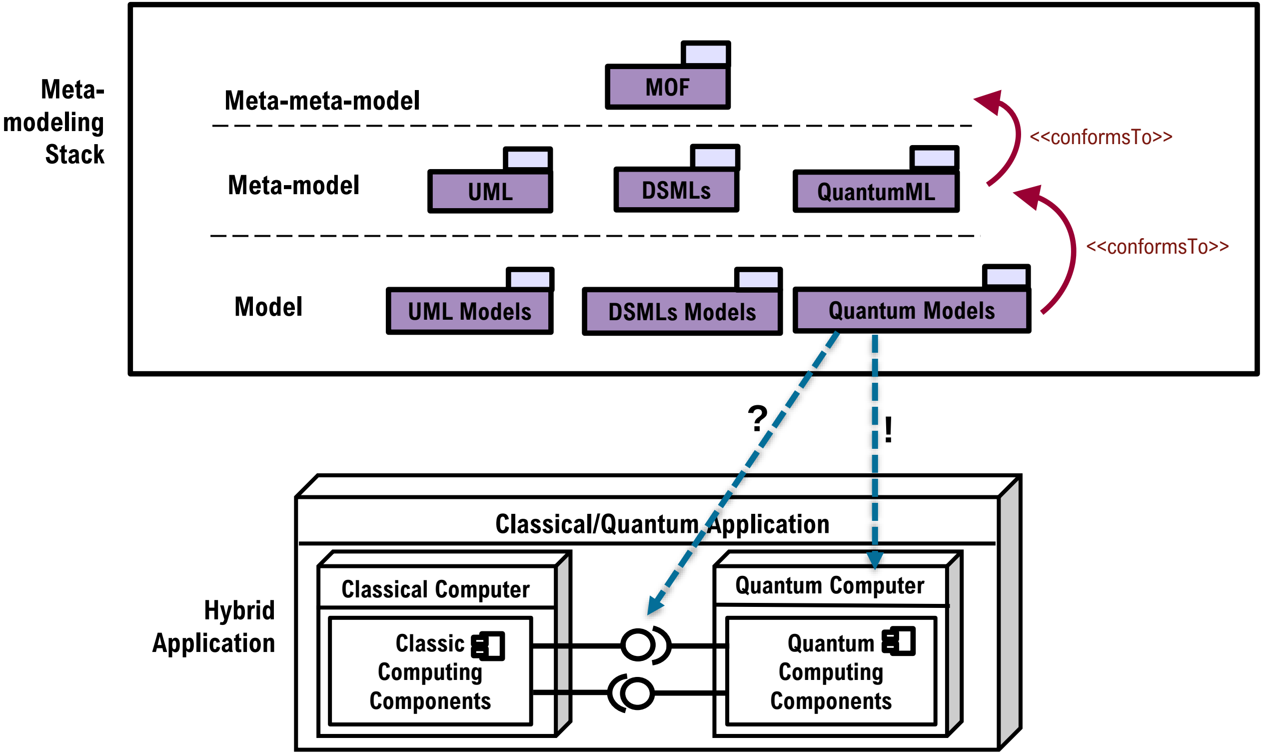


And what about Modeling Languages?



Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer:
Towards Model-Driven Quantum Software Engineering. Q-SE@ICSE, 2021.

And what about Modeling Languages?



Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Robert Wille. A Model-Driven Framework for Composition-Based Quantum Circuit Design. ACM Transactions on Quantum Computing 5(4), 2024.

Reality Check: Multitude of Challenges/Opportunities

QC community focus

- More and better **qubits** (hardware)
- Demonstrating **quantum advantage** (algorithms)

However, **software** is the next **bottleneck**

- **Abstraction gap**: gate-level (like assembly!)
- **No best practices**: Testing, debugging, verification
- **Tooling immaturity**: Limited IDE support, e.g., profilers, debuggers, validators, …
- **Skills gap**: Quantum physicists ≠ software engineers

Opportunity for computer scientists

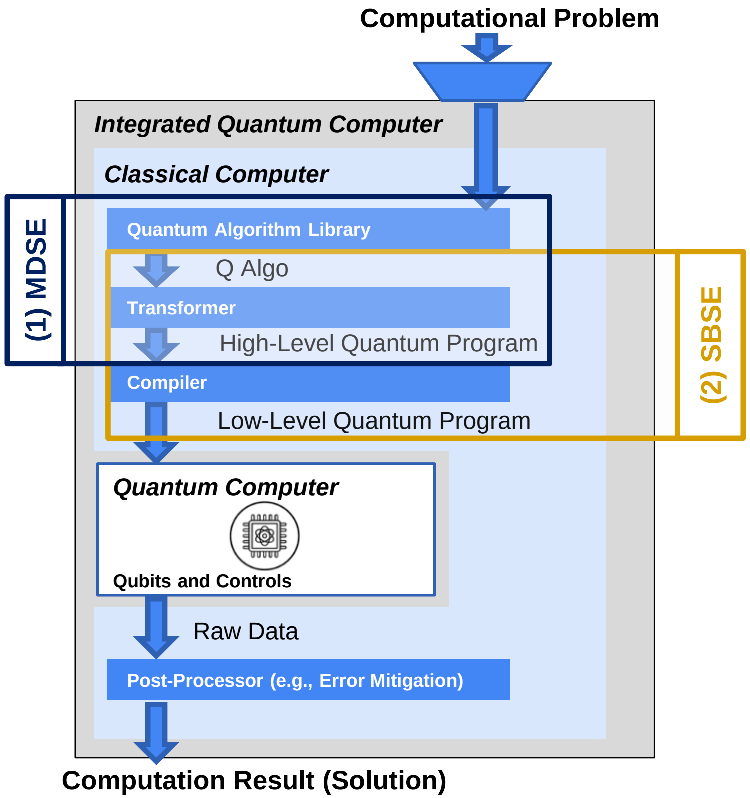
(Non-exhaustive List!)

- ☑ Model-driven engineering
- ☑ Search-based software engineering
- ☑ Testing and verification
- ☑ Software processes
- ☑ Service Oriented Computing
- ☑ Software Architecture
- ☑ Quantum AI

Juan Manuel Murillo et al.: Quantum Software Engineering: Roadmap and Challenges Ahead. ACM Trans. Softw. Eng. Methodol. 34(5): 154:1-154:48 (2025)

QSE Focus at BISE (JKU Linz)

Composition-based Quantum Circuit Design with Model-Driven Software Engineering



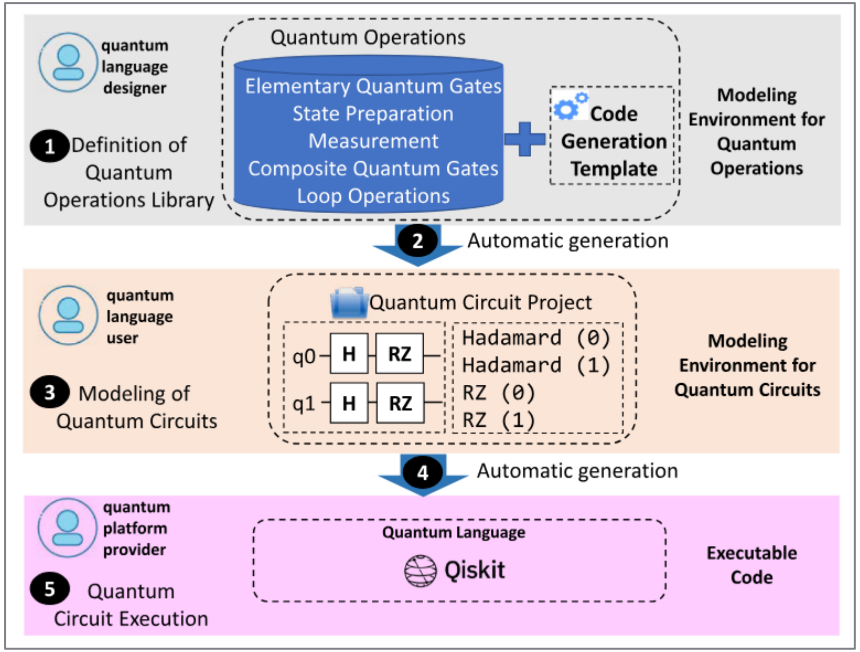
Quantum Circuit Synthesis and Debugging with Search-based Software Engineering

Integrated Quantum Computer Illustration based on Proctor *et al.* Benchmarking quantum computers. *Nat Rev Phys* **7**, 105–118 (2025).

https://github.com/jku-win-se/composition-quantum-circuit

Area #1: MDSE 4 QSE

Methodology

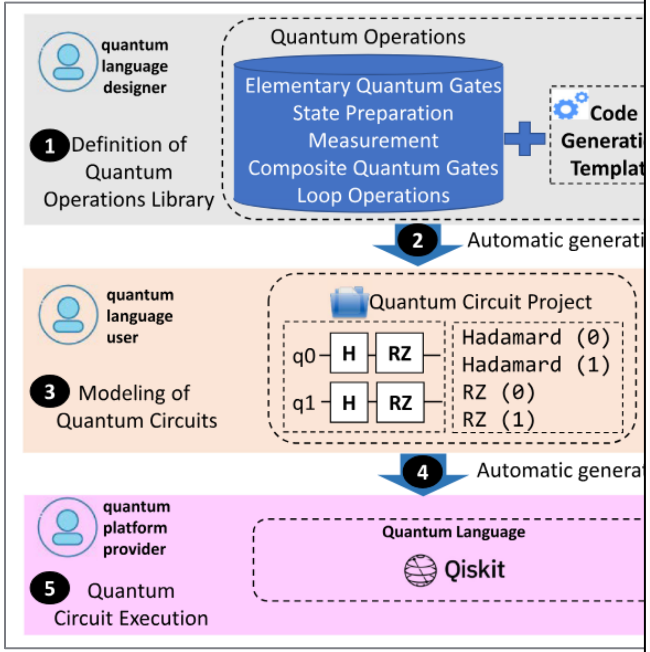


Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Robert Wille. A Model-Driven Framework for Composition-Based Quantum Circuit Design. ACM Transactions on Quantum Computing 5(4), 2024.

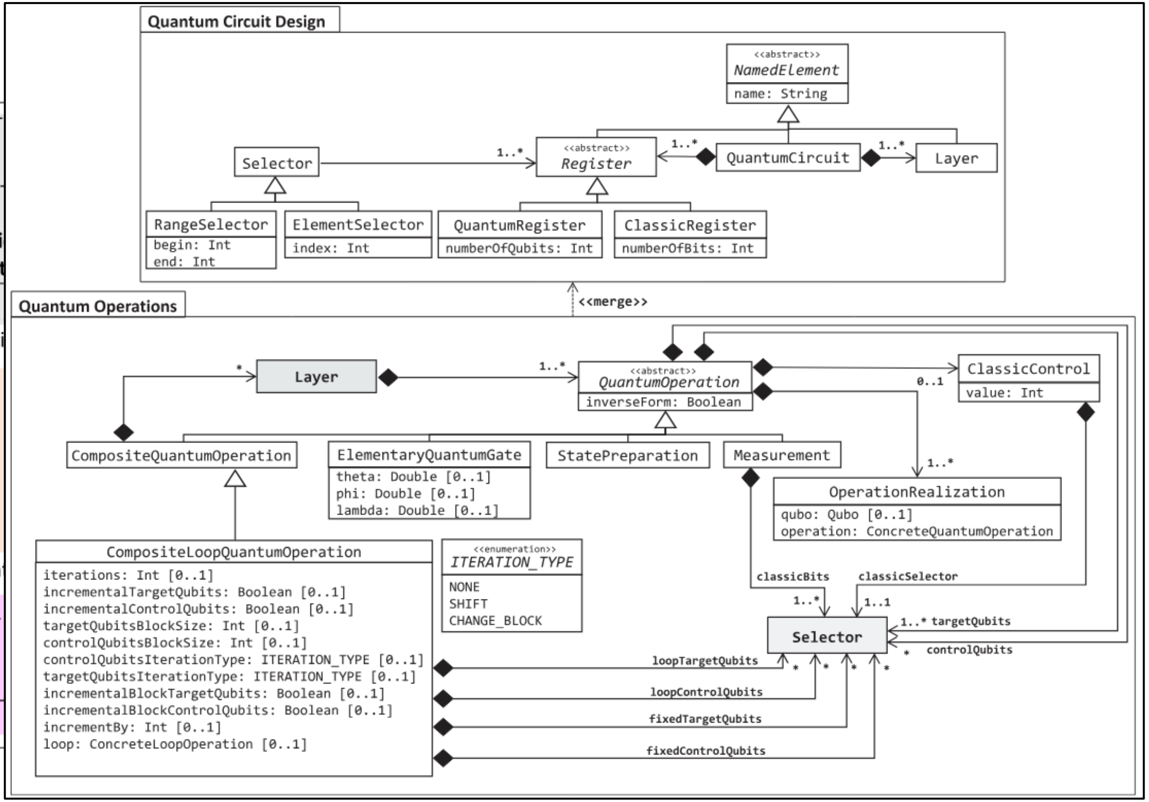
https://github.com/jku-win-se/composition-quantum-circuit

Area #1: MDSE 4 QSE

Methodology



Metamodel for Quantum Circuits

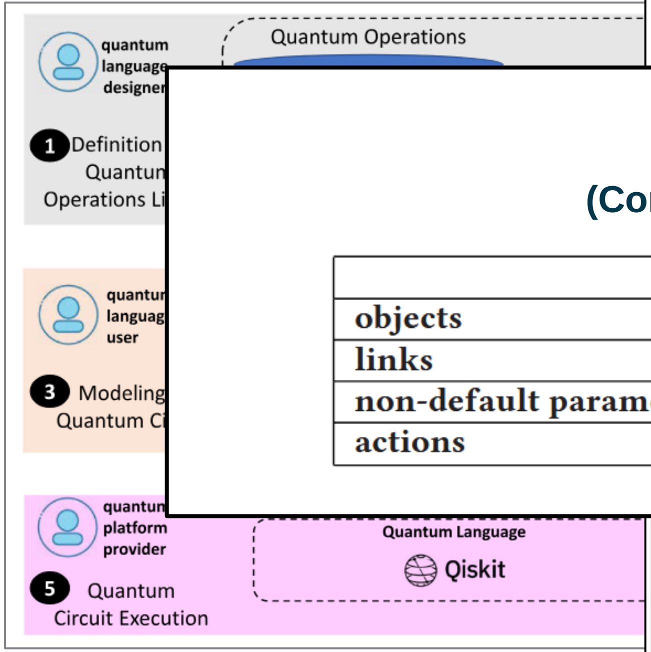


Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Robert Wille. A Model-Driven Framework for Composition-Based Quantum Circuit Design. ACM Transactions on Quantum Computing 5(4), 2024.

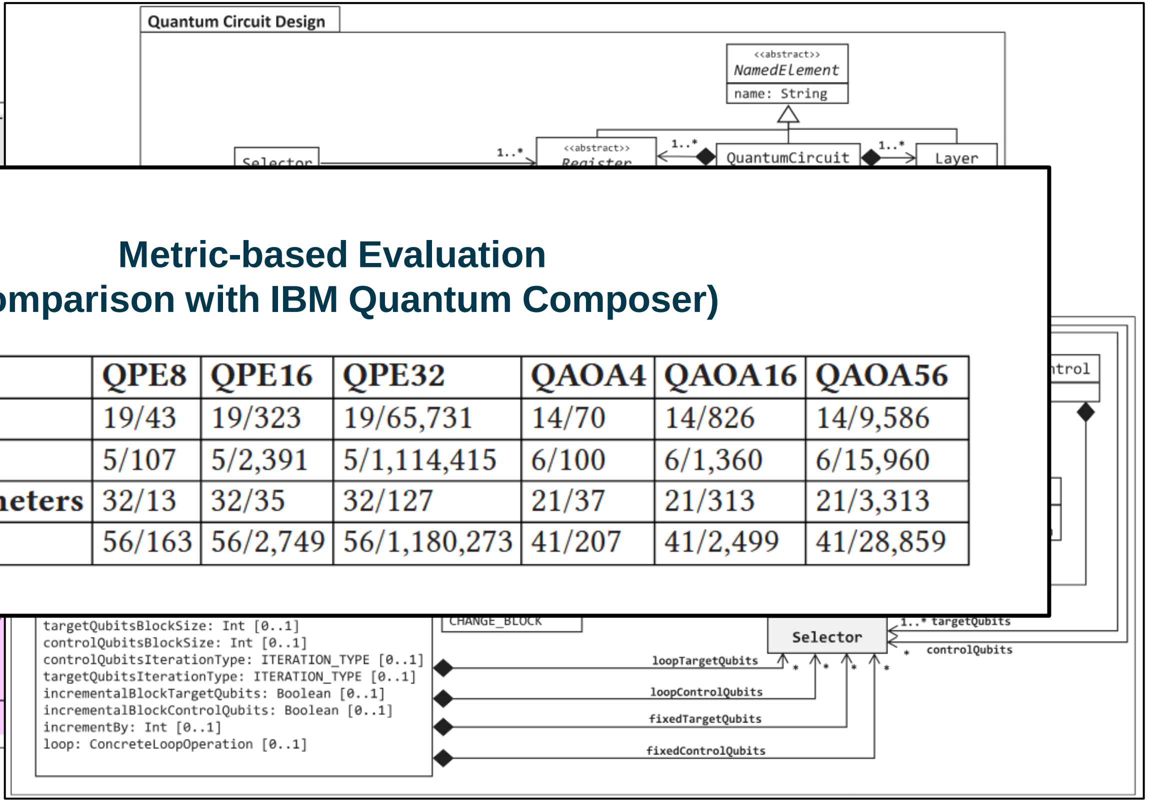
https://github.com/jku-win-se/composition-quantum-circuit

Area #1: MDSE 4 QSE

Methodology



Metamodel for Quantum Circuits



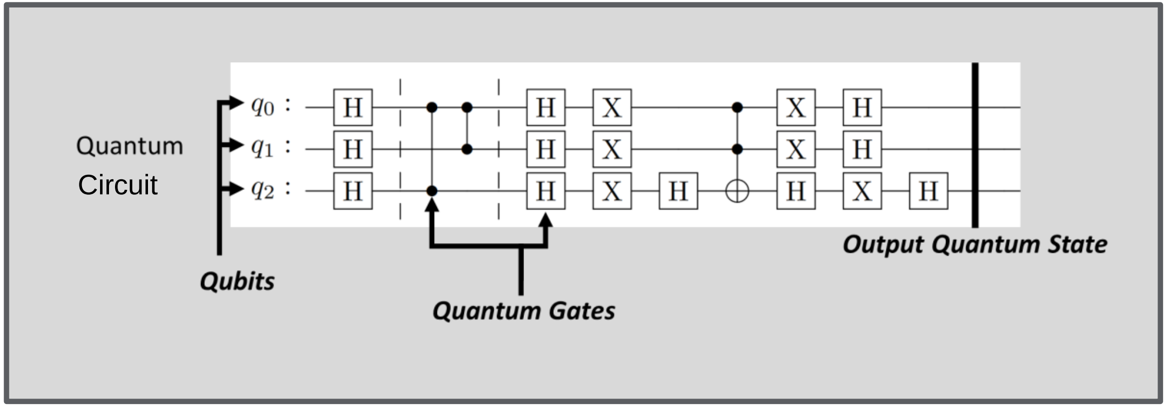
Metric-based Evaluation (Comparison with IBM Quantum Composer)

	QPE8	QPE16	QPE32	QAOA4	QAOA16	QAOA56
objects	19/43	19/323	19/65,731	14/70	14/826	14/9,586
links	5/107	5/2,391	5/1,114,415	6/100	6/1,360	6/15,960
non-default parameters	32/13	32/35	32/127	21/37	21/313	21/3,313
actions	56/163	56/2,749	56/1,180,273	41/207	41/2,499	41/28,859

Felix Gemeinhardt, Antonio Garmendia, Manuel Wimmer, Robert Wille. A Model-Driven Framework for Composition-Based Quantum Circuit Design. ACM Transactions on Quantum Computing 5(4), 2024.

Area #2: Quantum Circuit Synthesis (QCS)

Goal: Find a quantum circuit that implements a desired behavior.



QCS comes in many different flavors…

Compilation and Circuit Mapping

Abstract Algorithm → Concrete Circuit ⊢ hardware constraints

Quantum Architecture Search

Unitary → Variational QC (+ parameters)

Quantum Algorithm Discovery

Unitaries/Cases → Abstract Algorithm

State Preparation

Target State → Concrete Circuit

Challenges

– Vast design space
– Precision vs. computational cost
– **NISQ**: Errors, Probabilistic Nature, Hardware Specifics

→We treat QCS as a Search Problem!

QCS as a Search Problem

SBSE has been used for classical software for more than two decades [Harman, Jones 2001]

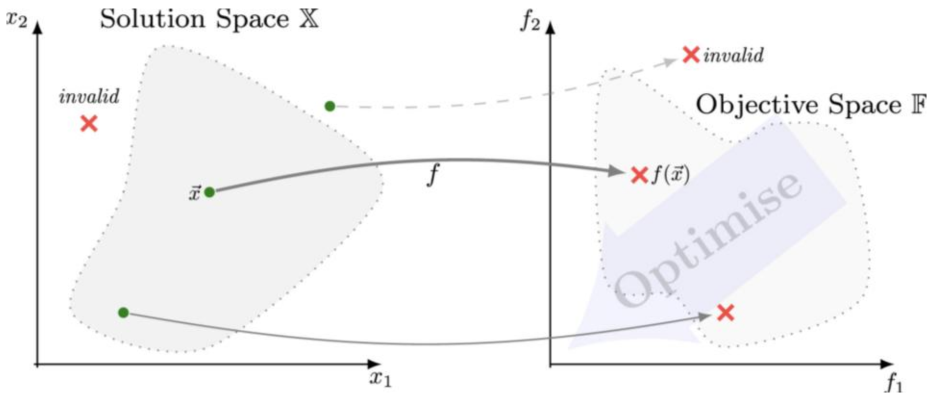
Goal: Find a software system / program that optimizes a given fitness function

Advantages:

Automated exploration by

Meta-heuristic searchers

- Genetic Programming (GP), …

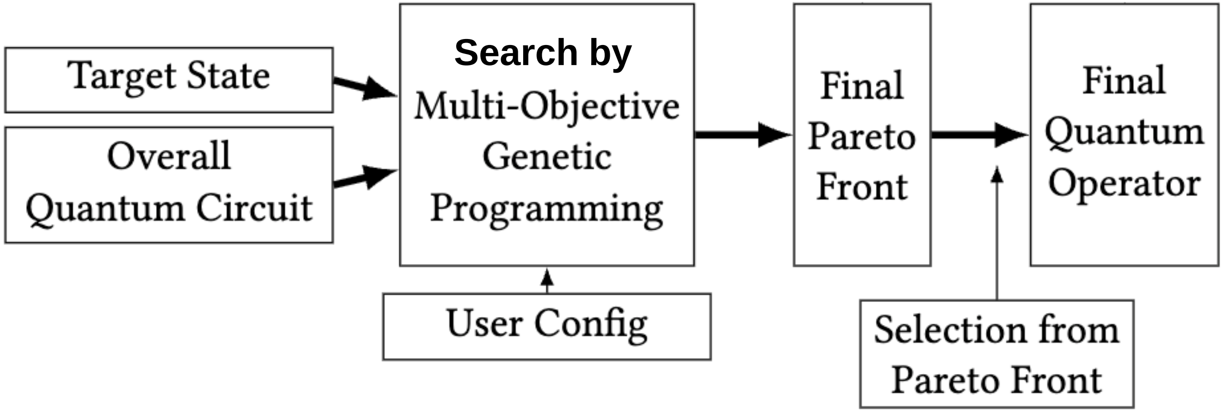


Challenges: Finding a good

1. Program encoding
2. Fitness function + evaluation capabilities

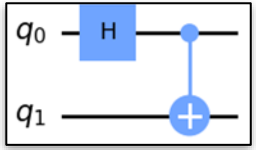
Our Approach: Search Scheme using GP

Multi-Objective Optimization allows to deal with **NISQ** characteristics



Typical GP Setup for Quantum Circuits

Encoding: gate vector



≅

[
  H(target=0),
  CNOT(target=1, control=0)
]

Fitness

behaviour: accuracy/fidelity/distance, and
structure: # gates, depth,
non-local gates, # parameters

Operators

mutation: add / delete / move / alter / swap gate
change qubits, …
crossover: one-point, two-point, …
selection: tournament, …

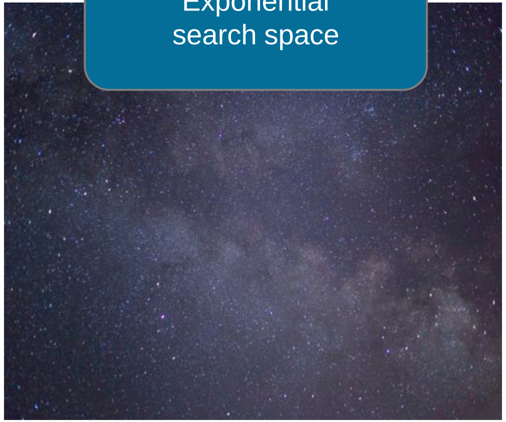
Algorithms

use of classical meta-heuristic algorithms:
GA, NSGA-II, NSGA-III, …

Is this all we need?

Dealing with several challenges

Exponential
search space



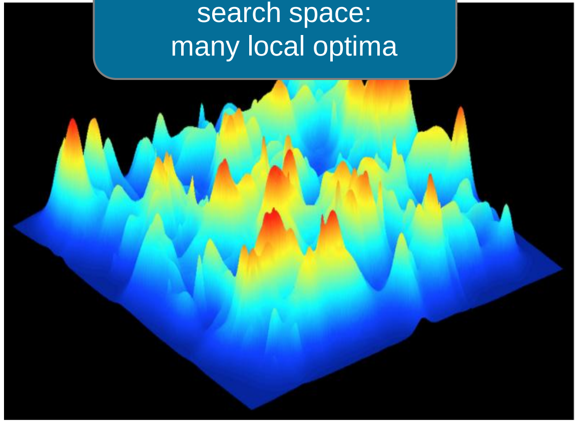
➔ Shrink search space

Expensive
Candidate evaluation
(Simulation)

(*photo* © Florian Hirzinger)

➔ Speed up or avoid simulation

Hard to navigate
search space:
many local optima



(*photo* © Bjørn Østman)

➔ Make search more efficient

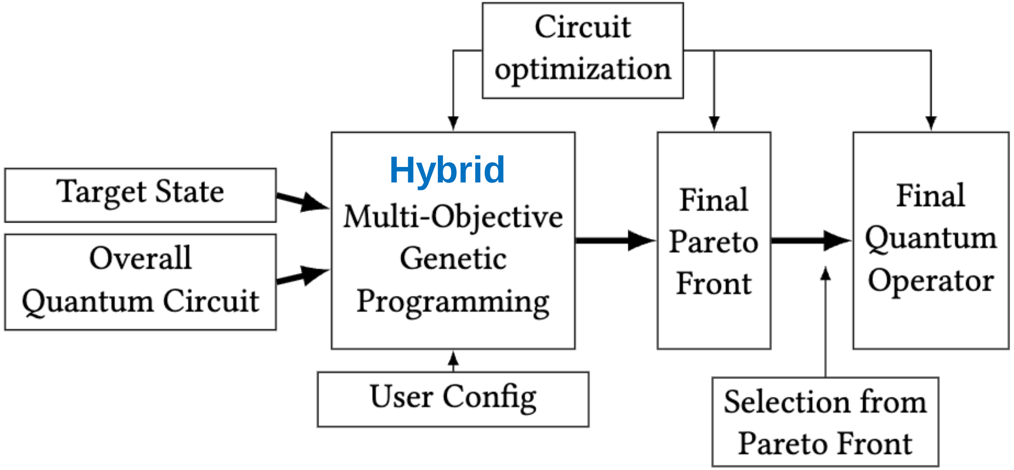
Challenge: Hard to navigate search space
Hybrid Quantum State Synthesis

Problem

- Some gates (e.g., RXθ, RYθ, RZθ) require parameters $\theta \in (0, 2\pi)$

Core Idea

- Embed parameter optimization (Nelder-Mead) within evolutionary search

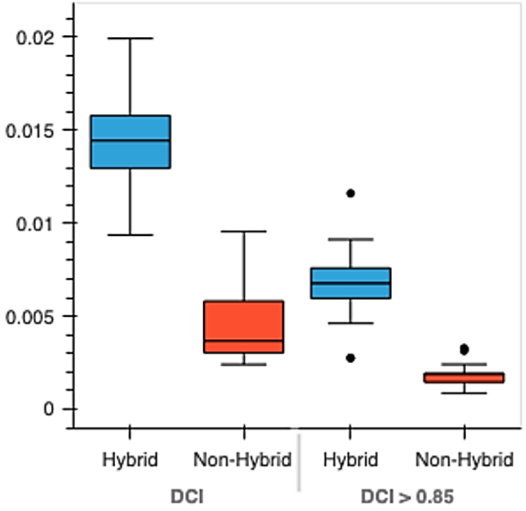


Results

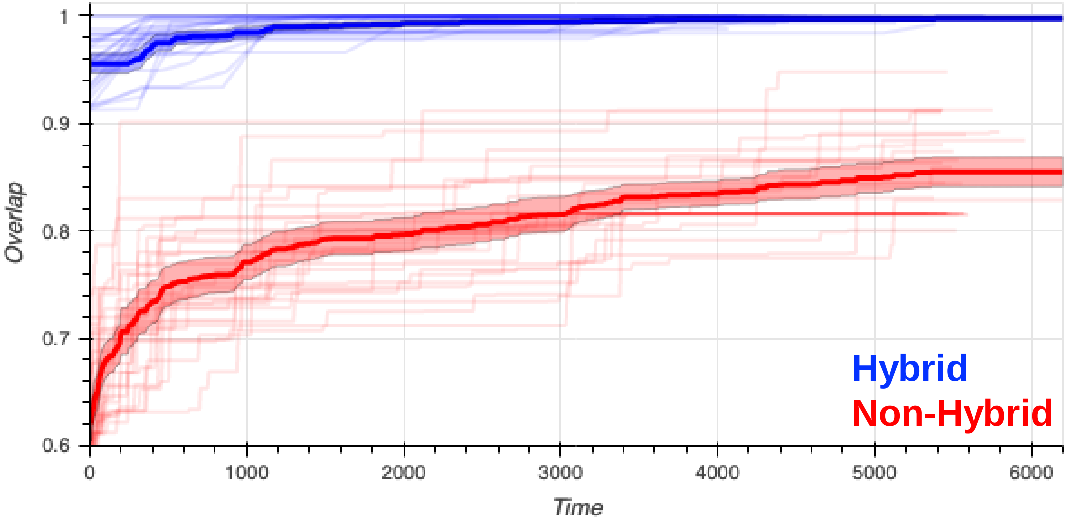
- Higher diversity
- Faster convergence

Felix Gemeinhardt, Stefan Klikovits, Manuel Wimmer: Hybrid Multi-Objective Genetic Programming for Parameterized Quantum Operator Discovery. GECCO Companion 2023: 795-798

Challenge: Hard to navigate search space
Hybrid Quantum State Synthesis

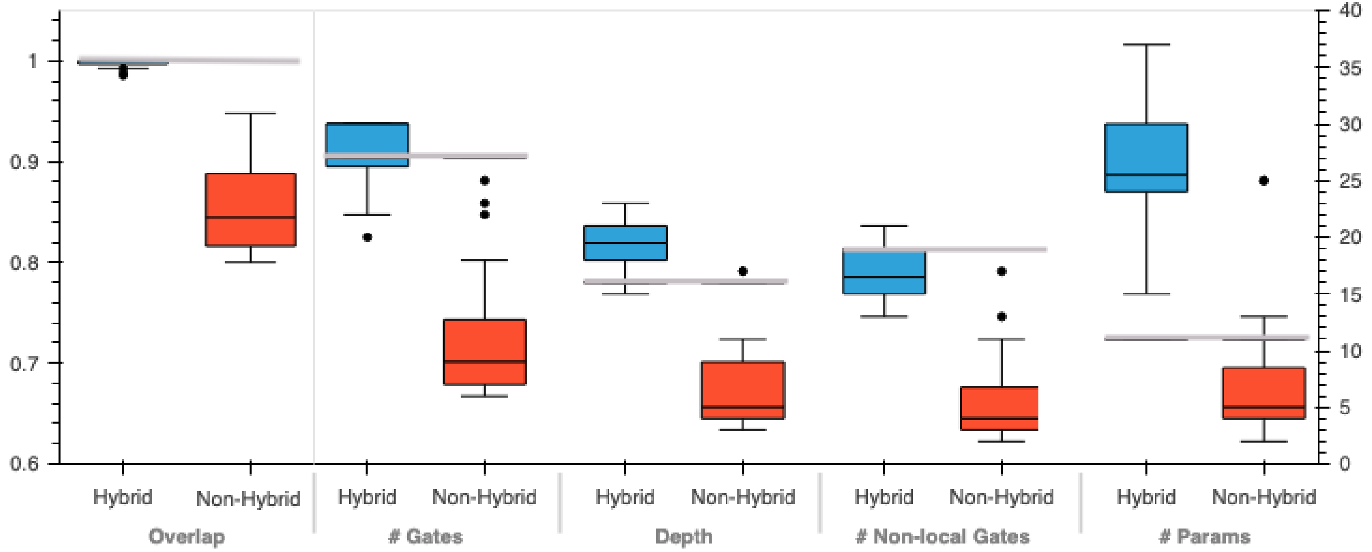


RQ1: better diversity



RQ2: Hybrid has higher overlap
+ faster convergence

Challenge: Hard to navigate search space
Hybrid Quantum State Synthesis



RQ3: higher overlap, but more costly circuits

Hybrid
Non-Hybrid

Challenge: Hard to navigate search space
Hybrid Quantum Debugging and Optimization

Problem

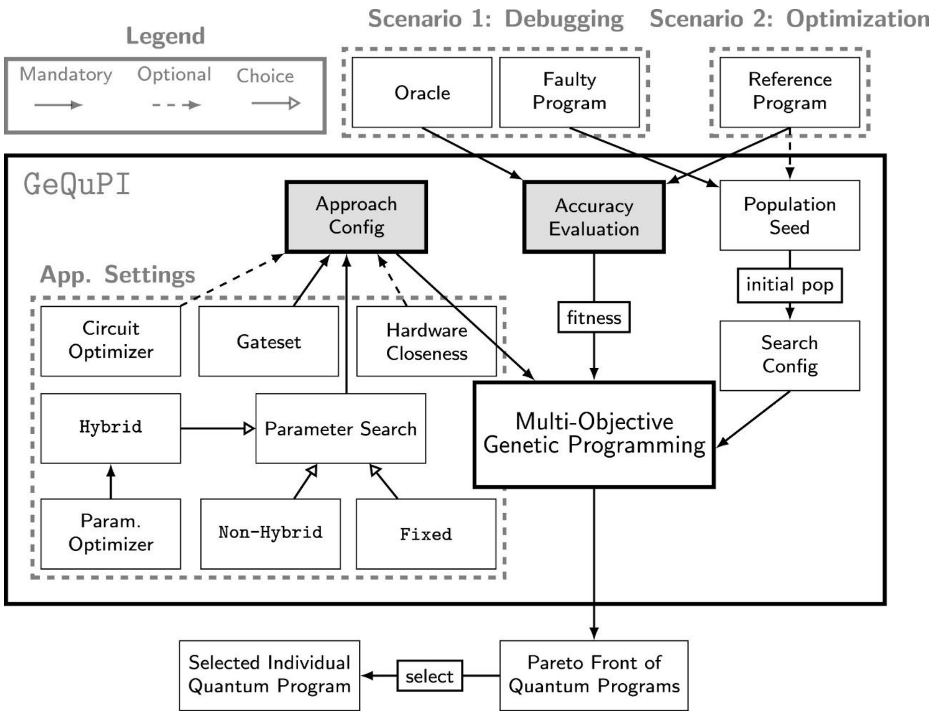
- Sometimes we already have a scaffold or almost correct quantum circuit

Core Idea

- Use **existing information** as “**seed**” for searching for better versions

Results

- Can fix buggy and optimize inefficient quantum circuits



JOHANNES KEPLER UNIVERSITY LINZ

Felix Gemeinhardt, Stefan Klikovits, Manuel Wimmer: GeQuPI: Quantum Program Improvement with Multi-Objective Genetic Programming. J. Syst. Softw. 219 (2025)

Debugging capabilities?

- Hybrid can optimize
  - for specific input states
  - with Non-Hybrid & Fixed sometimes only equally good solutions are found

- Hybrid can repair arbitrary input-state programs
  - Non-Hybrid & Fixed have problems

Debugging capabilities: number of runs per category and use case. (`Hybrid` / `Non-Hybrid` / `Fixed`).

(a) RQ1.1 (Perfect Accuracy)

Input state	Problem	Optimized	Pareto Equal	Worse	Faulty
Specific	QG_8 (2 qubits)	30/30/30	0/0/0	0/0/0	0/0/0
	QSO_6 (2 qubits)	26/0/0	4/30/30	0/0/0	0/0/0
	QSO_5 (3 qubits)	30/30/30	0/0/0	0/0/0	0/0/0
	QSE_15 (4 qubits)	30/30/30	0/0/0	0/0/0	0/0/0
	QSE_3 (5 qubits)	30/30/30	0/0/0	0/0/0	0/0/0
Arbitrary	QSE2_2 (2 qubits)	30/0/0	0/0/30	0/16/0	0/14/0
	QSE2_3 (3 qubits)	0/0/0	17/0/0	8/0/8	5/30/22
	QSE2_4 (4 qubits)	0/0/0	0/0/0	7/0/0	23/30/30
	QSE2_5 (5 qubits)	0/0/0	0/0/0	6/0/0	24/30/30

Optimization capabilities?

- All approaches can optimize
- Hybrid performs (almost) consistently better
- Hybrid improves by 35%

- Hybrid compared to “standard approach” (Qiskit built-in optimizer):
  - optimize in significantly more cases
  - and higher on average

Optimization capabilities (`Hybrid` / `Non-Hybrid` / `Fixed`).

(a) RQ2.1 (Perfect Accuracy)

	Optimized	Pareto Equal	Worse	Faulty
Total	541/135/105	143/84/64	50/10/14	406/911/957
Specific	274/63/45	89/84/64	23/10/14	124/353/387
Arbitrary	267/72/60	54/0/0	27/0/0	282/558/570
2 qubits	135/101/90	40/30/30	19/0/0	16/79/90
3 qubits	248/17/11	37/20/2	9/7/14	66/316/333
4 qubits	120/6/2	55/32/30	14/3/0	141/289/298
5 qubits	38/11/2	11/2/2	8/0/0	183/227/236

Lesson Learned:
Simulation of the candidate circuits consumes majority of the execution time of the discussed approaches!

Ongoing work: Simulation with Caching

Problem

- QCS (often) operates on **circuit populations** (e.g., when using GAs)
- Populations (often and on purpose) contain **redundancy**

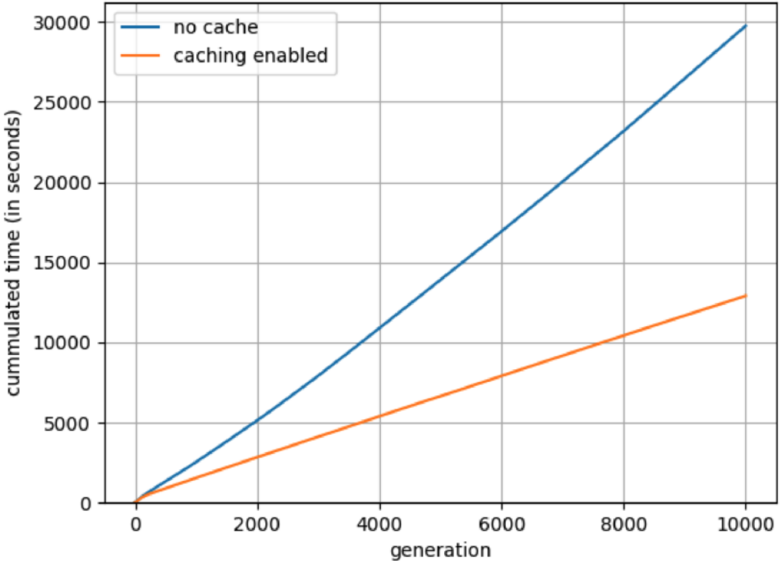
Core Idea

- Identify recurring gate patterns and aim for reuse of the results → reduce computations

Results

– Clear **reduction in simulation time** (depending on redundancy and qubits)

> 50% time savings in GAs



Ongoing work: Divide & Conquer Simulation

Problem

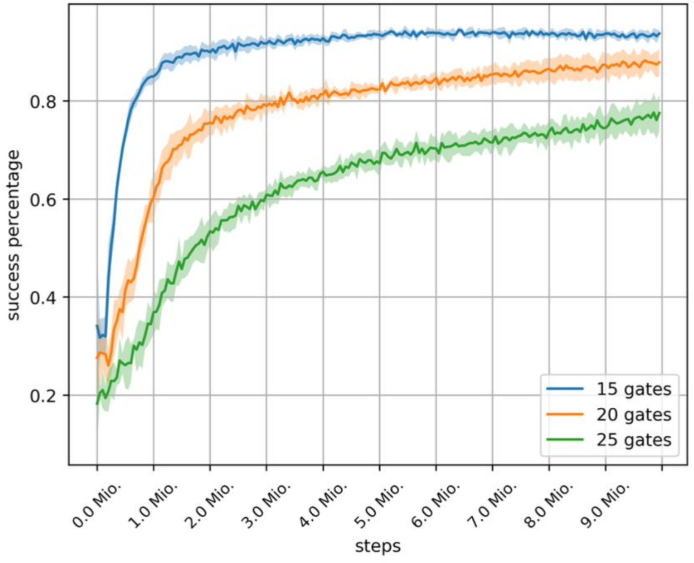
- Cost of simulation grows exponentially in number of qubits

Core Idea

- Use **Reinforcement Learning** to **split** synthesis problem and **solve** subproblems **independently**

Results

- Agent was able to divide in > 75% of all cases
- Much fewer gates needed for separation than for synthesis of the circuits



JOHANNES KEPLER UNIVERSITY LINZ

Summary – Lessons Learned

Technological Spaces

- Integrating QC into “your” technologies (we did this for the EMF ecosystem)
- Allows to reuse existing tooling (which is often defined on the meta-level)
- Model transformations, code generation, validation, …

Adaptation of Methodologies

- Possible, but requires completely novel treatments for particular subproblems
- **Requires interdisciplinary exchange and new skills for researchers**

Quantum Europe Strategy

https://digital-strategy.ec.europa.eu/en/policies/quantum

Area 5 - Quantum Skills: Building a diverse, world-class workforce through coordinated education, training, and talent mobility across the EU

„However, this is still insufficient to meet the projected demand from EU's startups and industry, which faces **major shortages of professionals** with relevant applied skills.“

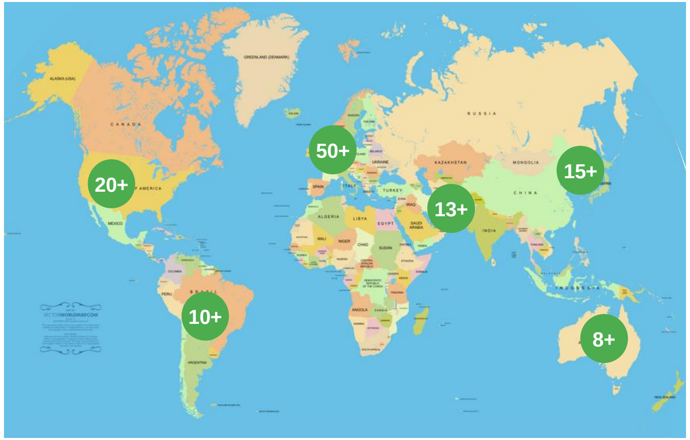
„Shortages are **most critical** in applied fields, including **quantum software engineering**, system integration, and quantum cybersecurity, slowing the commercialisation path for EU-based startups and scaleups.“

Taken from:

https://digital-strategy.ec.europa.eu/en/library/quantum-europe-strategy

Academic Study Programmes

100+ Programmes Worldwide



Characteristics

- Heterogeneous landscape of programmes
- **Focus on MSc** degree (typically building on CS or physics)
- Industry Partners: IBM, Microsoft, Google, AWS, …
- Include Q-Programming, more advanced software engineering topics rarely embedded within broader QC curricula

Notable Examples

- Quantum Master Barcelona (incl. Quantum Software track)
- Aalto: BSc → MSc → PhD in Quantum Technology
- Gdansk: International MSc in QIT

What is Missing

- Dedicated Programme for Quantum Software
(Software Specialisation only at KAIST, SDU, UTS)
- Shared understanding on “how to design such a programme”

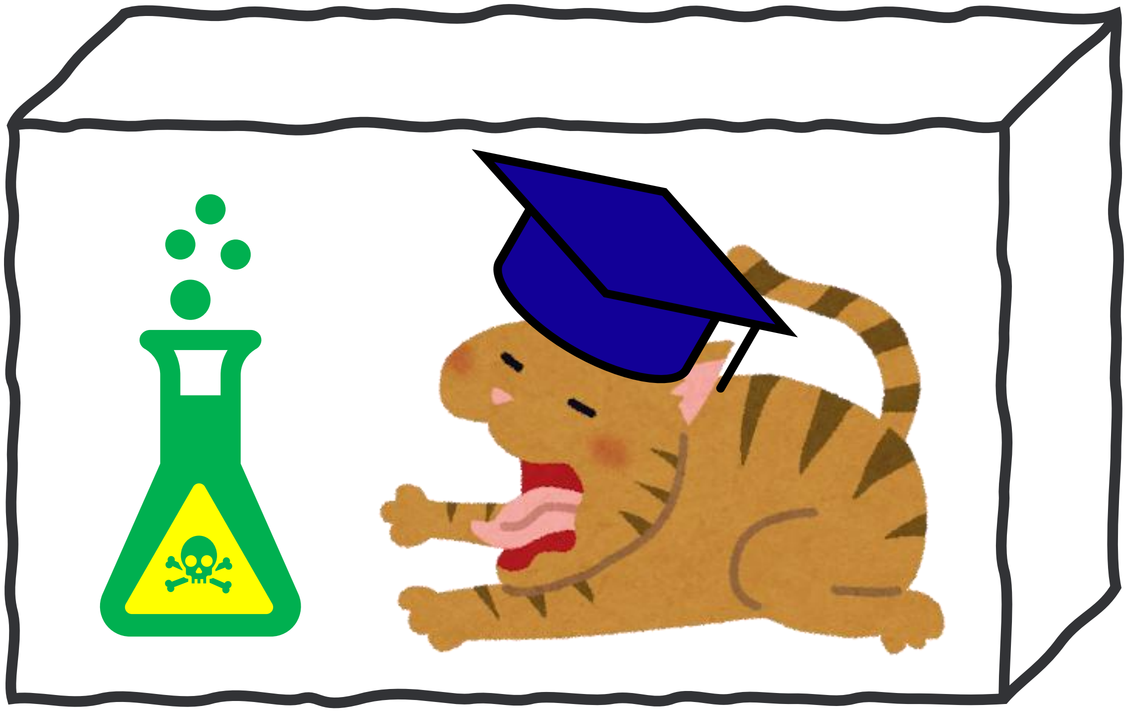
QSE Community Building

- **SE Field**
  - **IEEE International Conference on Quantum Software (QSW)**
    - Includes a symposium (especially for industry), workshop
  - **International Workshop on Quantum Software Engineering (Q-SE) @ ICSE**
  - + many more workshops
- **Quantum Field**
  - **IEEE Quantum Week** – Flagship Conference on Quantum Technologies
    - **Quantum System Software Track**
    - Quantum Science and Engineering Education Conference (QSEEC)
- **Journals**
  - **SE**: ACM TOSEM (Continuous Special Section on Quantum SE)
  - **Quantum**: ACM TQC, IEEE TQE
  - …

Wrap-Up

- **Quantum computers** made a **huge progress** in the last decade, roadmaps look ambitious
  - Quantum Utility still to be shown!
- **Innovations** of quantum computing will be **driven by** quantum **software**
  - What role will Computer Science play here?
- Current **practices of developing quantum software** are **improving**, but are still **limited**
  - What can be reused from classical SE?
  - How are quantum components integrated in larger software systems?
  - How can we scale up current practices supporting 10-100 qubit systems to 1k qubit systems and beyond?
- What type of **organizational change** is required here?
  - Dedicated quantum groups + software, software groups + quantum, software groups + quantum groups, …

We're hiring!



☐ Dead?

☐ Alive?

☒ PhD in Linz

From qubits to qudits ☺

Thank you!

Comments? Questions? Feedback?

Looking forward to discussions and collaborations!

Many THANKS to my
Quantum Software Co-Authors:

Felix Gemeinhardt, Christoph Stein, Stefan Klikovits, Antonio Garmendia,
Benjamin Weder, Frank Leymann, Martin Eisenberg, Daniel Lehner