Institute of Architecture of Application Systems

Quantum Workforce: Human Resources for the Future of Computing

Johanna Barzen, Frank Leymann

Institute of Architecture of Application Systems,
University of Stuttgart, Germany
Johanna.barzen@iaas.uni-stuttgart.de

Technical Report 2022/01

Presented at Transatlantic Quantum Forum 2022
(Munich, Germany, September 16 – 17, 2022)

BibTeX

```
@techreport{Barzen2022_QuantumWorkforce,
    Author = {Johanna Barzen and Frank Leymann},
    Title = {{Quantum Workforce: Human Resources for the Future of Computing}},
    Institution = {Universit{\"a}t Stuttgart, Fakult{\"a}t Informatik,
    Elektrotechnik und Informationstechnik, Germany},
    Pages = 6,
    Type = {Technischer Bericht Informatik},
    Number = {2022/01},
    Month = sep,
    Year = 2022,
}
```

Quantum Workforce:
Human Resources for the Future of Computing[1]

Johanna Barzen and Frank Leymann

University of Stuttgart, IAAS, Universitätsstr. 38, 70569 Stuttgart, Germany
{firstname.lastname}@iaas.uni-stuttgart.de

Introduction

Quantum computing has an enormous economic and social potential [1],[2]. The global quantum computing market is growing at an exponential pace [3]. In the past, the focus of quantum computing was on building hardware, but since a couple of years quantum algorithms and their applications is gaining more and more interest. This is reflected by estimations according to which the market share of quantum hardware is less than 10% of the share of quantum applications and services [4], [5]. Thus, the question arises who can develop these applications and who can provide these services [6].

Quantum algorithms — which are at the heart of quantum applications — are very different from classical algorithms, and the same is true for the implementation of quantum algorithms on quantum hardware. The experience of the authors through several years of teaching in this field shows that creating such algorithms and implementing them is impossible for people with todays computer science and software development skills.

Consequently, quantum algorithms are typically invented today by quantum physicists who also implement them. But "just" inventing and implementing a quantum algorithm is by far not a quantum application: developing an application requires a lot of classical software being integrated with the implementation of the quantum algorithm [7], [8]. This in turn requires deep software engineering skills [9] which goes far beyond the programming skills of quantum physicists.

Thus, quantum physicists and software engineers must work hand-in-hand in order to build quantum applications or to deliver corresponding services to realize the expected potential economic value of quantum computing. For this purpose, at least a joint language across these two disciplines must be established to furnish the required collaboration. Beyond this, the capability to communicate with domain experts is needed to explore the full spectrum of potential applications of quantum computing.

In practice, application projects involving such a heterogenous set of people and their different skills have high chances to fail. Thus, ideally, the relevant skills should be combined into a new single role. This implies a new course of study and a corresponding curriculum to educate the quantum workforce needed to harvest the potentials of quantum applications. Several universities have started this (see [10] for a first impression on 20 master and PhD degree programs in quantum computing) but the curricula are diverse reflecting the mix of existing foci of the collaborating departments of the universities. This must become more homogeneous in order to result in a new role that can be globally relied on — just like the role of a quantum physicist or a computer scientist itself is somehow uniform.

It might very well be that we witness the dawn of a new scientific field (like business informatics or digital humanities before): the field of *quantum software*. This would imply the need for new structures at universities, i.e. new faculties, new career opportunities for young researcher etc. The consequence is a quest to governments and public entities to shape the eduction system and career system in quantum informatics.

In this paper, we will sketch the main action items required to set up the field of quantum software in teaching, career development, and research funding.

Homogeneous Curriculum

Several universities offer master courses in the domain of quantum computing. But these courses show a broad spectrum of foci of the subjects taught (e.g. see [11], [12], [13], [14] to name just a few). These different foci typically result from the domain expertise already available in the departments collaborating in offering the course. As a consequence, graduates of the study programs of different universities are hard to compare in the sense that they have very different skill sets, i.e. it not clear from the outset what skills can be expected from such a graduate and, thus, how to properly deploy such a graduate in projects.

This is different from well-established study programs like computer science, or physics, or mathematics. For example, a graduate in computer science can be expected to understand compilers, distributed systems, algorithms, databases etc., and have a certain level of practice in at least one programming language. Taking a look at various curricula and course outlines of quantum computing programs, the common set of knowledge is far less homogeneous.

Action Item 1: In order to ensure a coherent workforce in quantum software, a homogeneous curriculum (like the one proposed here) should be agreed on and offered as a master program at universities worldwide.

In order to ease the deployment of future graduates for quantum software in projects, we propose such a homogeneous curriculum and sketch corresponding course outlines. This curriculum is intended to educate the workforce needed to build quantum software, i.e. to build applications making use of quantum computers - especially, the curriculum does not strive to educate the workforce for creating quantum computers themselves. Target participants of this master course are bachelor graduates of physics, mathematics, or computer science. The curriculum is based on experiences the authors gained in quantum software projects with several industry partners as well as teaching quantum computing for several years.

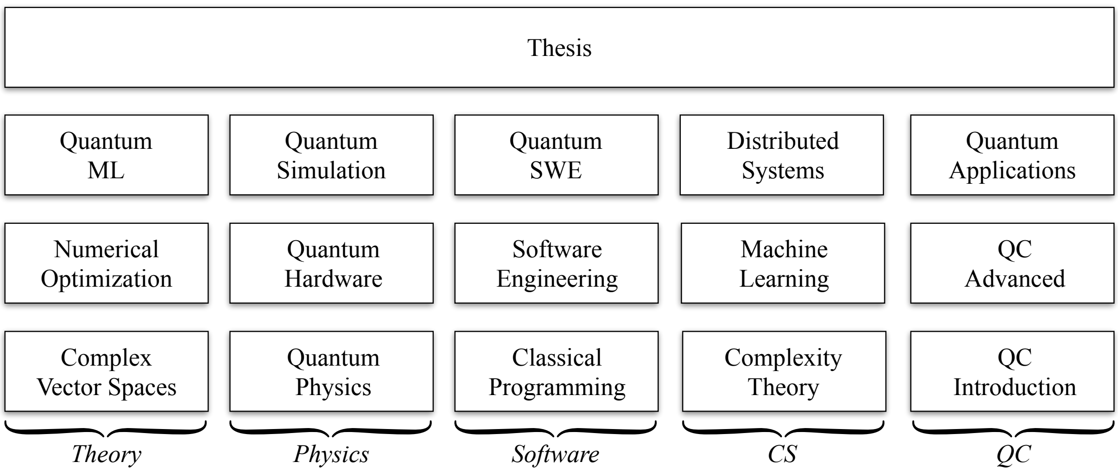

Figure 1: Sketch of a Homogeneous Curriculum for a Master Course in Quantum Software

The modules of the curriculum are sketched in Figure 1. For simplicity, each course is assumed to weigh 6 ECTS (i.e. 3 US credits). Each row corresponds to a semester. The first semester is represented by the row at the bottom, the fourth semester (completely devoted to the master thesis) is the row at the top. The columns represent the key foci of the study program:

- The *Theory* focus consists of the Complex Vector Space module, the Numerical Optimization Module, and the Quantum Machine Learning module (the outlines of the courses are sketched below).
- The modules Quantum Physics, Quantum Hardware, and Quantum Simulation make up the *Physics* focus.

- *Software* is focussed on Classical Programming, Software Engineering, and Quantum Software Engineering.
- *Computer Science* (CS) consists of Complexity Theory, Machine Learning, and Distributed Systems.
- Finally, *Quantum Computing* (QC) focusses on an Introduction to Quantum Computing, Advanced Quantum Computing, and Quantum Applications

Note again, that the goal of this curriculum is to teach the workforce for quantum software. By exchanging a few modules, a different emphasize can be achieved, thus, our curriculum is generic in this sense. For example, by exchanging Machine Learning by Security Basics, Quantum Simulation by Quantum Protocols, and Quantum ML by Post Quantum Security, an emphasize on quantum security maybe given.

The outline of each module is as follows (we list only the main subjects to teach for each module specific for the curriculum, and are brief on modules with "standard" content):

- *Complex Vector Spaces*: complex numbers, linear maps, matrix algebra, determinants, linear equations, eigenvalues and eigenvectors, spectrum and trace, complex scalar product, pre-hilbert spaces, norms and matrix norms, convergence, hilbert spaces, orthonormal basis, hermitian and unitary maps, spectral theorem, eigenbasis, real subspace of hermitian matrices, unitary group, tensor product, Schmidt decomposition
- *Quantum Physics*: photo effect, Compton effect, matter waves, wave-particle duality, double-slit experiments, Heisenberg's uncertainty principle, superposition, measurement, interference, entanglement, EPR paradox, hidden variables, Bell's inequality, pure states, ensembles and density matrices, reduced density matrices, Copenhagen interpretation, many worlds, de Brogli-Bohm theory
- *Classical Programming*: control flow constructs, data flow, introduction to procedural programming, introduction to object-oriented programming, Python overview (the choice of Python is based on its widespread use in frameworks for quantum computing), extensive labs in Python
- *Complexity Theory*: Classical problems, complexity measures, complexity classes, P & NP & NP complete & NP hard, hierarchies, reductions, approximations
- *Introduction to Quantum Computing*: qubits, Bloch sphere, quantum registers, tensor products, Dirac notation, entanglement, comparing entangled states, 1-qubit operations, Z-Y decomposition, 2-qubit operators, universality of operator sets, approximation of arbitrary unitary operators, preparation of entangled states, tooling for circuits, Deutsch-Jozsa algorithm, measurement, teleportation, dense encoding, Grover algorithm, quantum complexity, success amplification, Shor's algorithm, quantum cryptography, quantum key distribution, error correction, error mitigation, NISQ
- *Numerical Optimization*: unconstraint optimization, gradient methods, derivatives approximations, automatic differentiation, derivative-free optimization, Langrange calculus, quadratic programming
- *Quantum Hardware*: building blocks of a quantum computer, various qubit implementations (superconducting, ion trap, diamond, photons, spin, Rydberg, topological), gate implementations on these various qubits, readout, error mitigation
- *Software Engineering*: various process models, modeling (UML), requirements engineering, software architecture, patterns, modularization, APIs, reuse, quality assurance, project management, documentation, pattern languages
- *Machine Learning*: supervised/unsupervised learning, clustering, regression, perceptron, feature selection, feature engineering, principal component analysis, classification, support vector machines, kernels, neural networks
- *Advanced Quantum Computing*: density operators, pure states, ensembles, reduced density operators, quantum information, implementation of quantum Fourier transform, quantum phase estimation, amplitude amplification, state preparation, quantum

associative memory, oracle expansions, transpilation, readout errors, Hamiltonian simulation, HHL algorithms, quantum linear algebra, Pauli strings, measurement with Pauli operators, arbitrary measurements, variational algorithms, VQE, QAOA, warm starting, quantum gradients, hybrid algorithms, quantum annealing & adiabatic optimization

- *Quantum Machine Learning*: categorical data, quantum principal component projection, quantum neural networks, barren plateaus, quantum Boltzmann machines, quantum autoencoder, hybrid autoencoder, quantum no-free-lunch theorem, quantum support vector machines, quantum kernels, quantum classifier, variational classifier
- *Quantum Simulation*: exact diagonalization, quantum monte-carlo, renormalization, analog/digital quantum simulation, coherent dynamics, Suzuki-Trotter, quantum chemistry, ground state solver, excited states solver, potential energy surface
- *Quantum Software Engineering*: quantum circuit lifecycle (quantum-classical split, optimization, testing, hardware selection,…), quantum workflow lifecycle (quantum-classical split, rewrite,…), operations lifecycle (topology modeling, packaging, policies, deployment, observability), patterns
- *Distributed Systems*: N-tier architecture, APIs, OpenAPI, communication protocols, cloud computing, cloud native, topology modeling, deployment, provisioning, TOSCA, K8S, workflow management, quality of services, provenance
- *Quantum Applications*: services, quantum as a service, providers, API management, topology management, app stores, repositories, QPU plugability, hardware selection, runtime selection, tooling, interoperability

Academic Careers

Research in quantum computing in general, and in quantum software especially, requires a broad spectrum of knowledge and skills (as reflected by the various subjects of the modules of the curriculum above). At least two different types of researchers in this field can be observed: a "T-shaped" type and a "networked" type (as a variant of the "key-shaped type" [15]).

> **Action Item 2**: Research and teaching in quantum software requires a wholistic and cross-domain approach. Since corresponding researchers face hiring obstructions, dedicated departments or even a new faculty is needed at universities to exploit their potentials and advance their careers.

A T-shaped researcher has a good overview on many subjects that belong to the field and is highly specialized in a single subject. A networked researcher has a deep understanding of most subjects of the domain without being highly specialized in a single subject, but produces insights by achieving synergies across the subjects and by revealing application potentials.

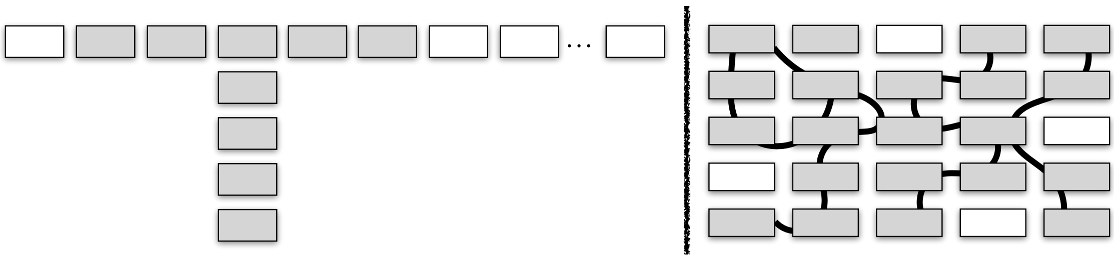

Figure 2: T-Shaped Researcher (left side) vs. Networked Researcher (right side)

T-shaped researchers fit into the current structures of todays universities. I.e. their specialization matches an existing faculty enabling an academic career for such a type of researcher.

A networked researcher cannot be really associated with a faculty of todays universities. Thus, these kind of researchers face a significant obstruction in advancing their academic career up to a tenured professor position. But their work is key in this domain, which is inherently based on integrating various knowledge areas.

The situation is that of business informatics some decades ago. Because of the economic value of business informatics, many universities created a separate faculty for business informatics, or a dedicated department with several tenured professor positions. The same must happen with quantum software to open a perspective for networked researchers in this domain. Without the views and contributions of this type of networked researchers, the domain of quantum software will not develop properly.

Research Funding

Research funding organizations tend to prefer funding projects with foci that fit the structure of the organization or the given predefined funding portfolio. Thus, projects that can not be associated with one of the corresponding units have lower chances to receive funding. For example, digital humanities projects had problems in the past to receive funding because their subjects span at least one domain from the humanities as well as computer science. After adding digital humanities to the portfolio, corresponding research proposals received funding without these organizational obstructions. As a consequence, the funding organizations must establish a mechanism to fund projects across units. Otherwise, one unit may argue why it should use its share of funds that benefits another unit also. Also, as reviews are being driven out of the units it may be difficult to find reviewers from the domain of other units qualified to assess a research proposal.

Action Item 3: Research in quantum software is crossing structural boundaries of funding organizations and their portfolios. To ensure equal opportunity for corresponding research proposals, “quantum software” must be added with a dedicated budget to these portfolios.

The latter is the current situation in quantum software. Corresponding research proposals often encompass subjects from computer science, physics, mathematics, and an application domain. Just like digital humanities research projects in the past, today research proposals in quantum software realizes quite some obstructions to receive funding. In order to reflect its economic value quantum software should be added to the portfolio of research organizations. The German expert council for quantum computing even recommends to found a separate funding organization called the German Quantum Foundation for the whole spectrum of quantum computing even separate from the German Research Foundation (DFG) [6].

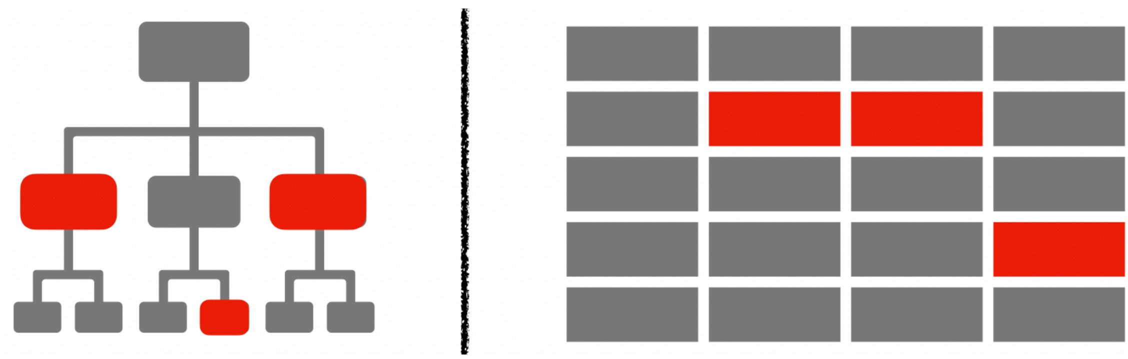

Figure 3: Quantum Software Research Proposals Consist of Subjects that Touch (in red) Different Organizational Boundaries of Funding Organizations (left side) or Their Portfolio Elements (right side)

References

[1] National Academies of Sciences, Engineering, and Medicine, 2019. Quantum Computing: Progress and Prospects. The National Academies Press, DOI: https://doi.org/10.17226/25196.

[2] Kagermann, Henning; Süssenguth, Florian; Körner, Jorg; Liepold, Annka: The Innovation Potential of Second-generation Quantum Technologies. acatech IMPULSE, 2020. https://www.acatech.de/publikation/innovationspotenziale-der-quantentechnologien/

[3] Fortune Business Insights. 2021. https://www.globenewswire.com/en/news-release/2022/03/04/2396846/0/en/Quantum-Computing-Market-to-Grow-Exponentially-Increasing-Product-Applications-to-Generate-Remunerative-Market-Opportunities-Fortune-Business-Insights.html

[4] Kirkpatrick, K.; Kaul, A.: Quantum Computing for Enterprise Markets. Tractica, 2019.

[5] Quantencomputing - Software for innovative and future-proof applications (in German). Federal Ministry for Economics and Climate Action, 2022. https://www.digitale-technologien.de/DT/Redaktion/DE/Downloads/Publikation/KI-Inno/2022/2022_03_22_Quantencomputing_3-2022.pdf

[6] Roadmap Quantencomputing. The Federal Government of Germany, 2021. https://www.quantentechnologien.de/fileadmin/public/Redaktion/Dokumente/PDF/Publikationen/Roadmap-Quantencomputing-bf-C1.pdf

[7] Leymann, Frank; Barzen, Johanna: The bitter truth about gate-based quantum algorithms in the NISQ era. In: Quantum Science and Technology, IOP Publishing Ltd, 2020.

[8] Weder, Benjamin; Barzen, Johanna; Leymann, Frank; Zimmermann, Michael: Hybrid Quantum Applications Need Two Orchestrations in Superposition: A Software Architecture Perspective. In: Proceedings of the 18th IEEE International Conference on Web Services (ICWS 2021), IEEE, 2021.

[9] Weder, Benjamin; Barzen, Johanna; Leymann, Frank; Salm, Marie; Vietz, Daniel: The Quantum Software Lifecycle. In: Proceedings of the 1st ACM SIGSOFT International Workshop on Architectures and Paradigms for Engineering Quantum Software (APEQS 2020), ACM, 2020.

[10] Hughes-Castleberry, Kenna: Top 20 Quantum Computing Masters & Ph.D. Degree Programs in 2022. https://thequantuminsider.com/2022/06/06/top-20-quantum-computing-masters-ph-d-programs-in-2022/

[11] Technical University of Munich (TUM): Master of Science (M.Sc.) “Quantum Science & Technology”. https://www.tum.de/en/studies/degree-programs/detail/quantum-science-technology-master-of-science-msc

[12] ETH Zürich: Master of Science (M.Sc.) “Master Quantum Engineering”. https://ethz.ch/en/studies/master/degree-programmes/engineering-sciences/quantum-engineering.html

[13] Deggendorf Institute of Technology: Master of Science (M.Sc.) “High Performance Computing / Quantum Computing”. https://www.th-deg.de/qc-m-en

[14] Democritus University of Thrace: Master of Science (M.Sc.) “International M.Sc. Program in Quantum Computing and Quantum Technologies” https://quantum.ee.duth.gr/curriculum/ https://www.th-deg.de/Fakultäten/Studiengangsflyer/flyer-qc-m-en.pdf

[15] David Wilkinson: New research: The shift between T-Shaped & Key-shaped employees. https://oxford-review.com/shift-between-t-key-shaped-people/

All Links have been followed August 7th, 2022.