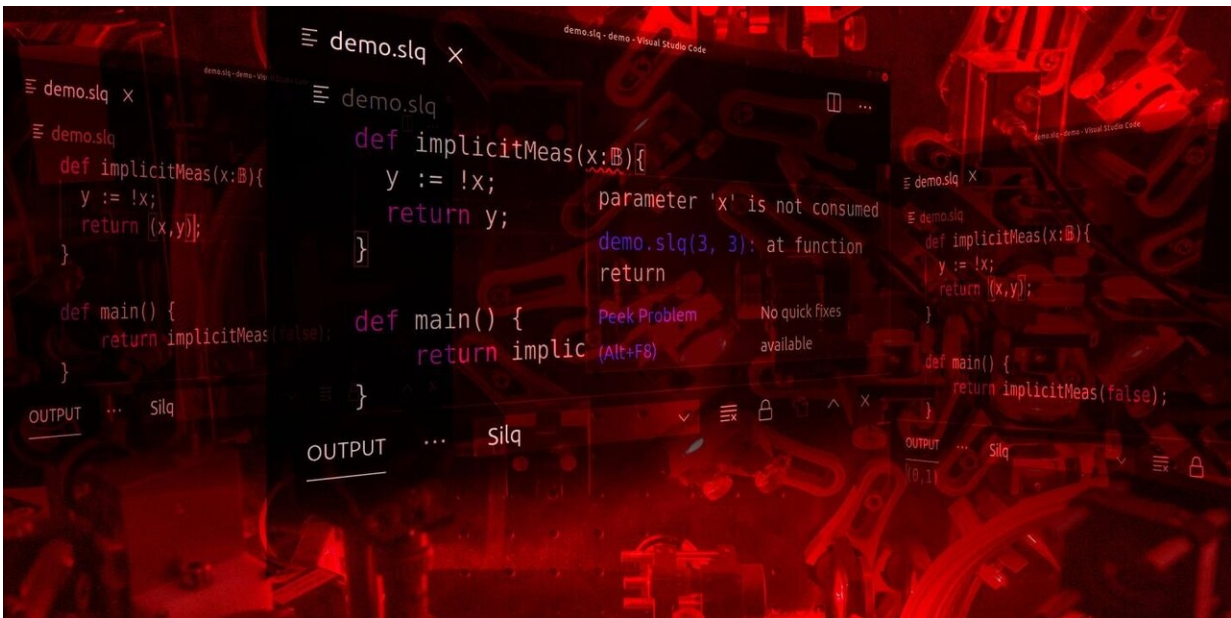


The first intuitive programming language for quantum computers

June 15 2020



Computer scientists at ETH Zurich have developed the first quantum programming language that enables solving complex computations elegantly, simply and safely. Credit: ETH Zurich

Programming quantum computers is becoming easier: computer scientists at ETH Zurich have designed the first programming language that can be used to program quantum computers as simply, reliably and safely as classical computers. "Programming quantum computers is still a challenge for researchers," says Martin Vechev, computer science

professor in ETH's Secure, Reliable and Intelligent Systems Lab (SRI), "which is why I'm so excited that we can now continue ETH Zurich's tradition in the development of quantum computers and programming languages."

He adds, "Our quantum programming language Silq allows programmers to utilize the potential of quantum computers better than with existing languages, because the code is more compact, faster, more intuitive and easier to understand for programmers." This week, Vechev will introduce Silq to other experts in the field at PLDI 2020, a conference for programming languages. To facilitate discussion, adoption and further development, he and his team have also released Silq on its own website (silq.ethz.ch).

Quantum computing has been seeing increased attention over the last decade, since these computers, which function according to the principles of quantum physics, have enormous potential. Today, most researchers believe that these computers will one day be able to solve certain problems faster than classical computers, since to perform their calculations they use entangled quantum states in which various bits of information overlap at a certain point in time. This means that in the future, quantum computers will be able to efficiently solve problems which classical computers cannot solve within a reasonable timeframe.

This quantum supremacy has still to be proven conclusively. However, some significant technical advances have been achieved recently. In late summer 2019, a quantum [computer](#) succeeded in solving a problem—albeit a very specific one—more quickly than the fastest classical computer.

For certain "quantum algorithms", i.e. computational strategies, it is also known that they are faster than classical algorithms, which do not exploit the potential of quantum computers. To date, however, these algorithms

still cannot be calculated on existing quantum hardware because quantum computers are currently still too error-prone.

Expressing the programmer's intent

Utilizing the potential of quantum computation not only requires the latest technology, but also a quantum programming language to describe quantum algorithms. In principle, an algorithm is a "recipe" for solving a problem; a programming language describes the algorithm so that a computer can perform the necessary calculations.

Today, quantum programming languages are tied closely to specific hardware; in other words, they describe precisely the behavior of the underlying circuits. For programmers, these "hardware description languages" are cumbersome and error-prone, since the individual programming instructions must be extremely detailed and thus explicitly describe the minutiae needed to implement quantum algorithms.

This is where Vechev and his group come in with their development of Silq. "Silq is the first quantum programming language that is not designed primarily around the construction and functionality of the hardware, but on the mindset of the programmers when they want to solve a problem—without requiring them to understand every detail of the computer architecture and implementation," says Benjamin Bichsel, a doctoral student in Vechev's group who is supervising the development of Silq.

Computer scientists refer to computer languages that abstract from the technical details of the specific type of computer as high-level programming languages. Silq is the very first high-level programming language for quantum computers. High-level programming languages are more expressive, meaning that they can describe even complex tasks and algorithms with less code. This makes them more comprehensible and

easier to use for programmers. They can also be used with different computer architectures.

Eliminating errors through automatic uncomputation

The greatest innovation and simplification that Silq brings to quantum programming languages concerns a source of errors that has plagued quantum programming until now. A computer calculates a task in several intermediate steps, which creates intermediate results or temporary values.

In order to relieve the memory, [classical computers](#) automatically erase these values. Computer scientists refer to this as "garbage collection", since the superfluous temporary values are disposed of.

In the case of quantum computers, this disposal is trickier due to quantum entanglement: the previously calculated values can interact with the current ones, interfering with the correct calculation. Accordingly, cleaning up such temporary values on quantum computers requires a more advanced technique of so-called uncomputation.

"Silq is the first [programming language](#) that automatically identifies and erases values that are no longer needed," explains Bichsel. The [computer scientists](#) achieved this by applying their knowledge of classical programming languages: their automatic uncomputation method uses only programming commands that are free of any special quantum operations—they are "qfree", as Vechev and Bichsel say.

"Silq is a major breakthrough in terms of optimizing the programming of quantum computers; it is not the final phase of development," says Vechev. There are still many open questions, but because Silq is easier to understand, Vechev and Bichsel hope to stimulate both the further development of quantum programming languages and the theory and

development of new quantum algorithms.

"Our team of four has made the breakthrough after two years of work thanks to the combination of different expertise in [language](#) design, quantum physics and implementation. If other research and development teams embrace our innovations, it will be a great success," says Bichsel.

More information: Benjamin Bichsel et al, Silq: a high-level quantum language with safe uncomputation and intuitive semantics, *Proceedings of the 41st ACM SIGPLAN Conference on Programming Language Design and Implementation* (2020). [DOI: 10.1145/3385412.3386007](https://doi.org/10.1145/3385412.3386007)

Provided by ETH Zurich

Citation: The first intuitive programming language for quantum computers (2020, June 15) retrieved 4 May 2024 from <https://techxplore.com/news/2020-06-intuitive-language-quantum.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.