

# SimulaQron

A simulator for developing quantum internet software

arXiv: 1712.08032

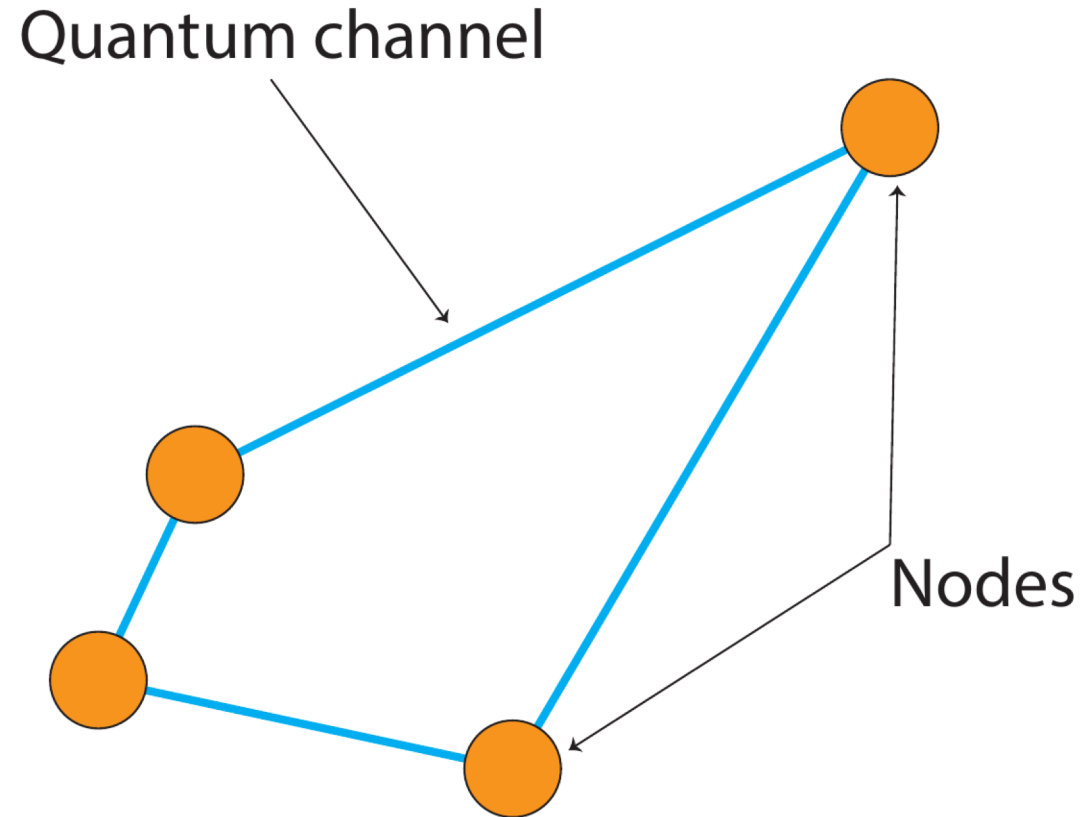
[www.simulaqron.org](http://www.simulaqron.org)



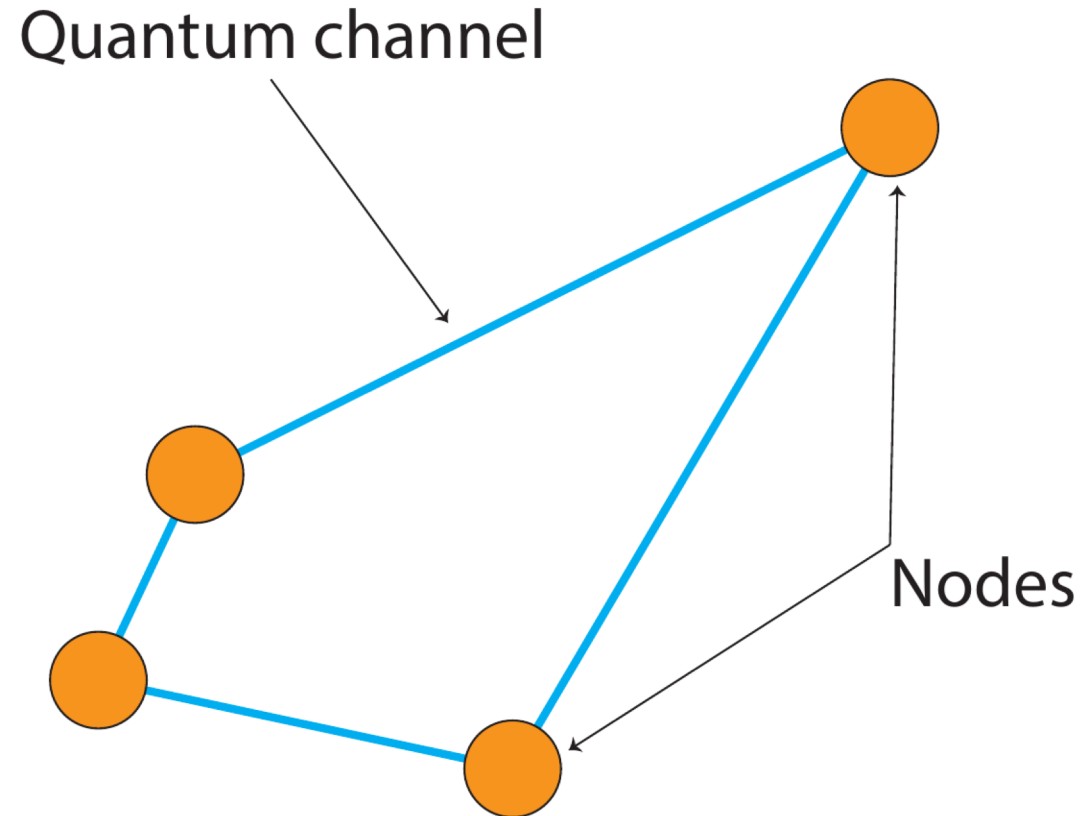
Axel Dahlberg  
Stephanie Wehner

# Quantum Network

# Quantum Network



# Quantum Network

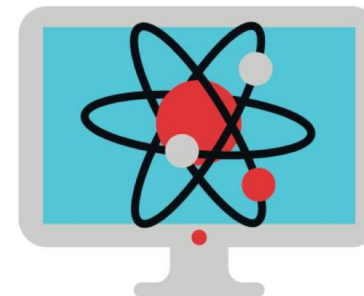
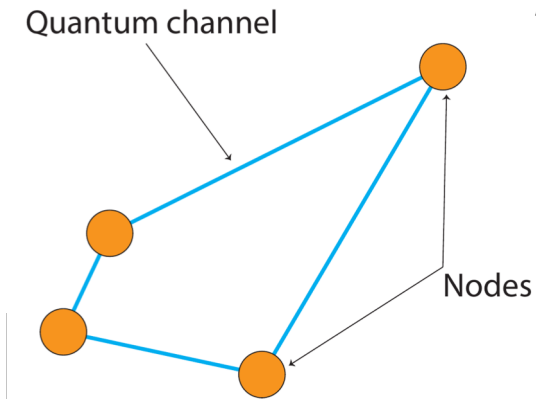




# Quantum Network, why?

## Applications:

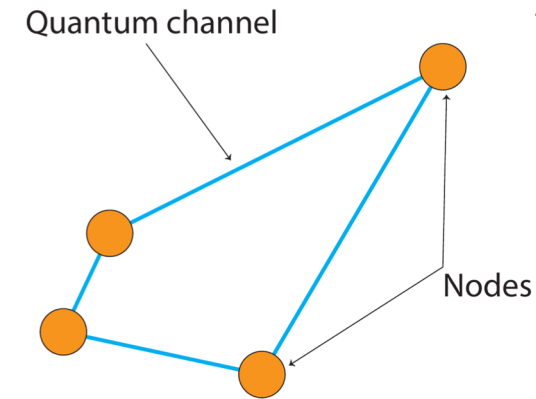
- Quantum key distribution
- Clock synchronization
- Blind quantum computation
- Secret sharing
- Anonymous transfer
- ...



# Quantum Network, why?

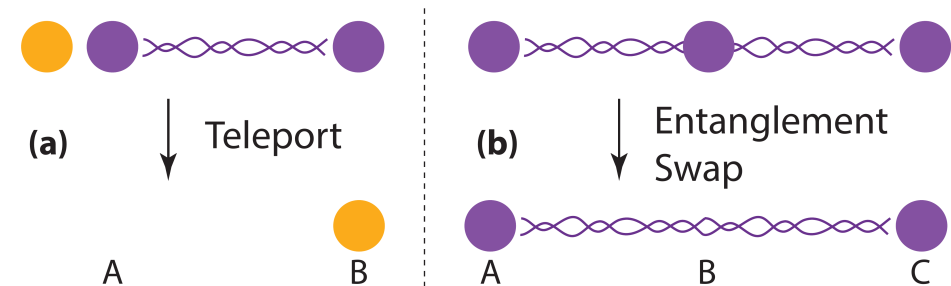
## Applications:

- Quantum key distribution
- Clock synchronization
- Blind quantum computation
- Secret sharing
- Anonymous transfer
- ...

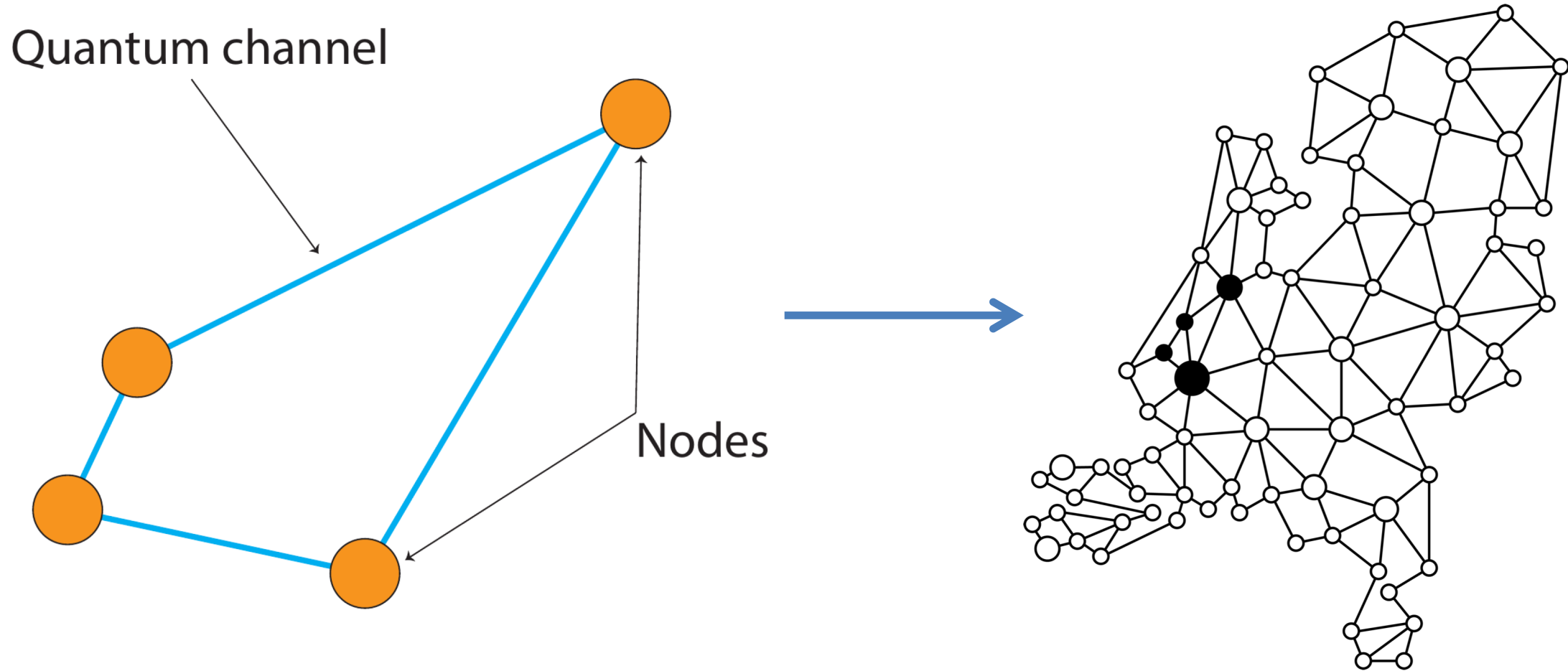


Requires entanglement!

Also for long-distance quantum communication.



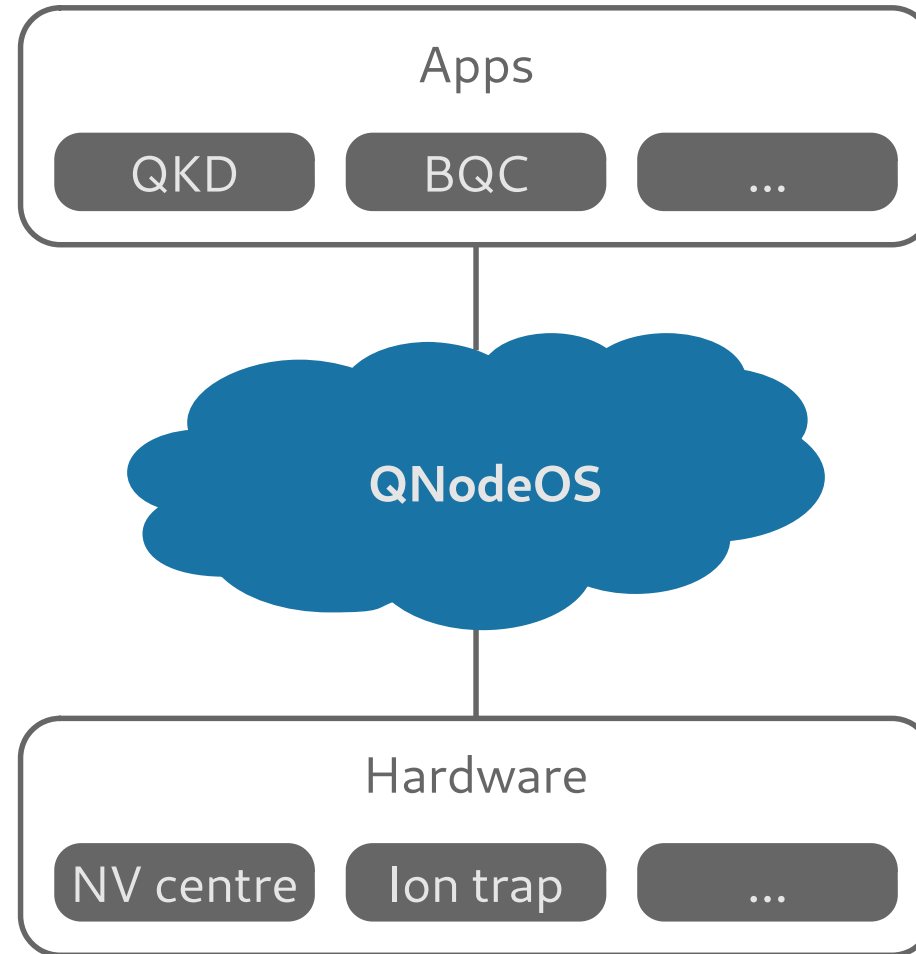
# Towards a Quantum Internet



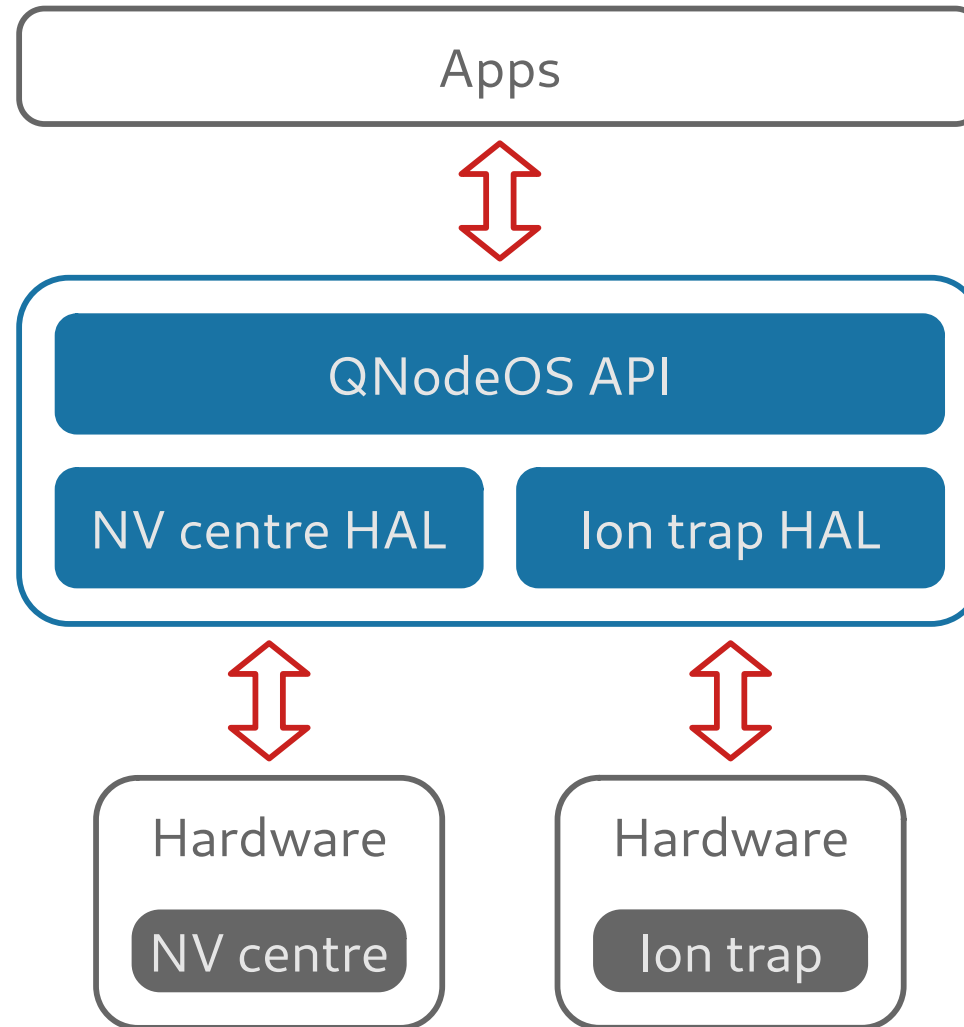
# Need for a quantum network stack

Application	
Transport	Qubit transmission
Network	Long distance entanglement
Link	Robust entanglement generation
Physical	Attempt entanglement generation

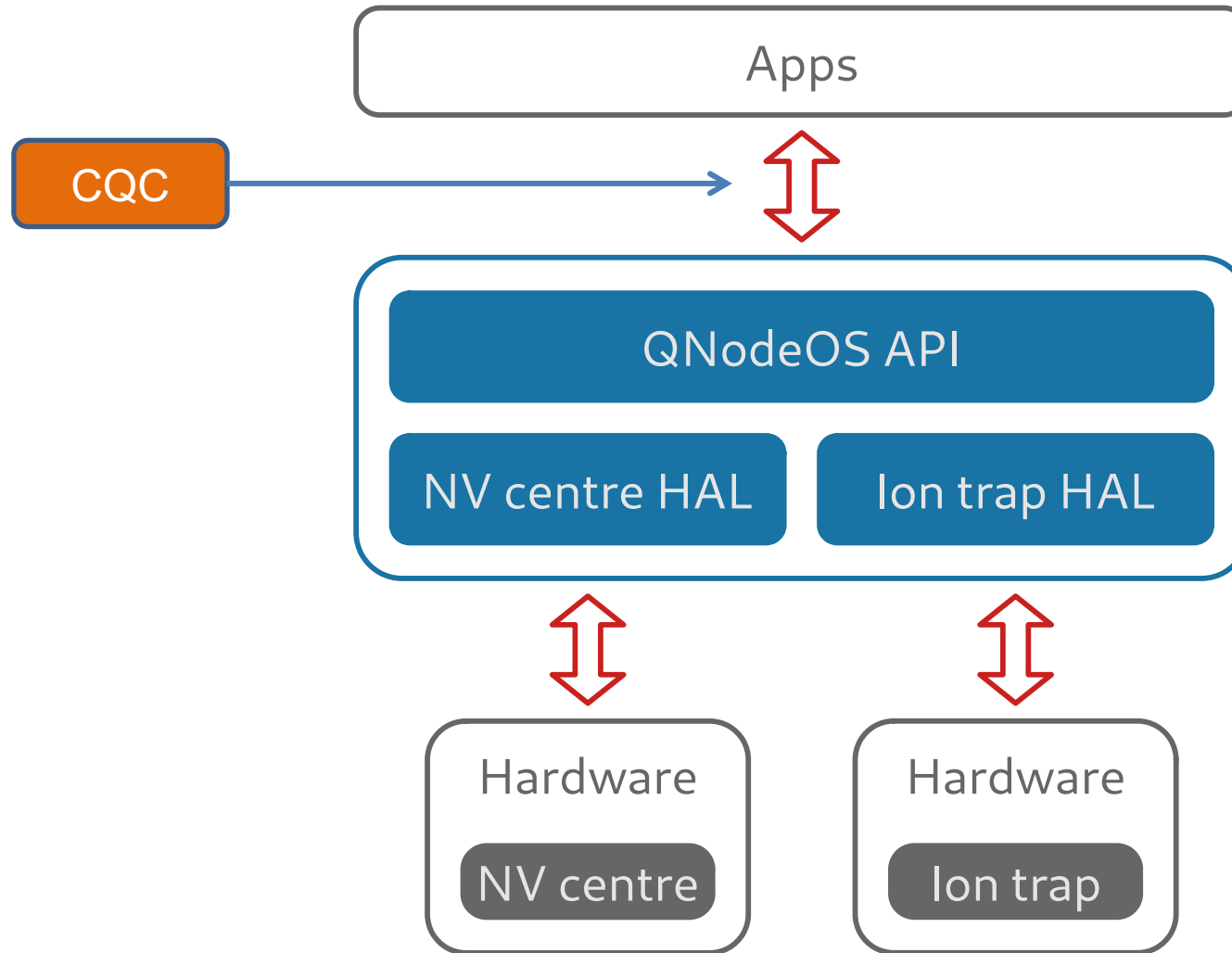
# Need for QNodeOS



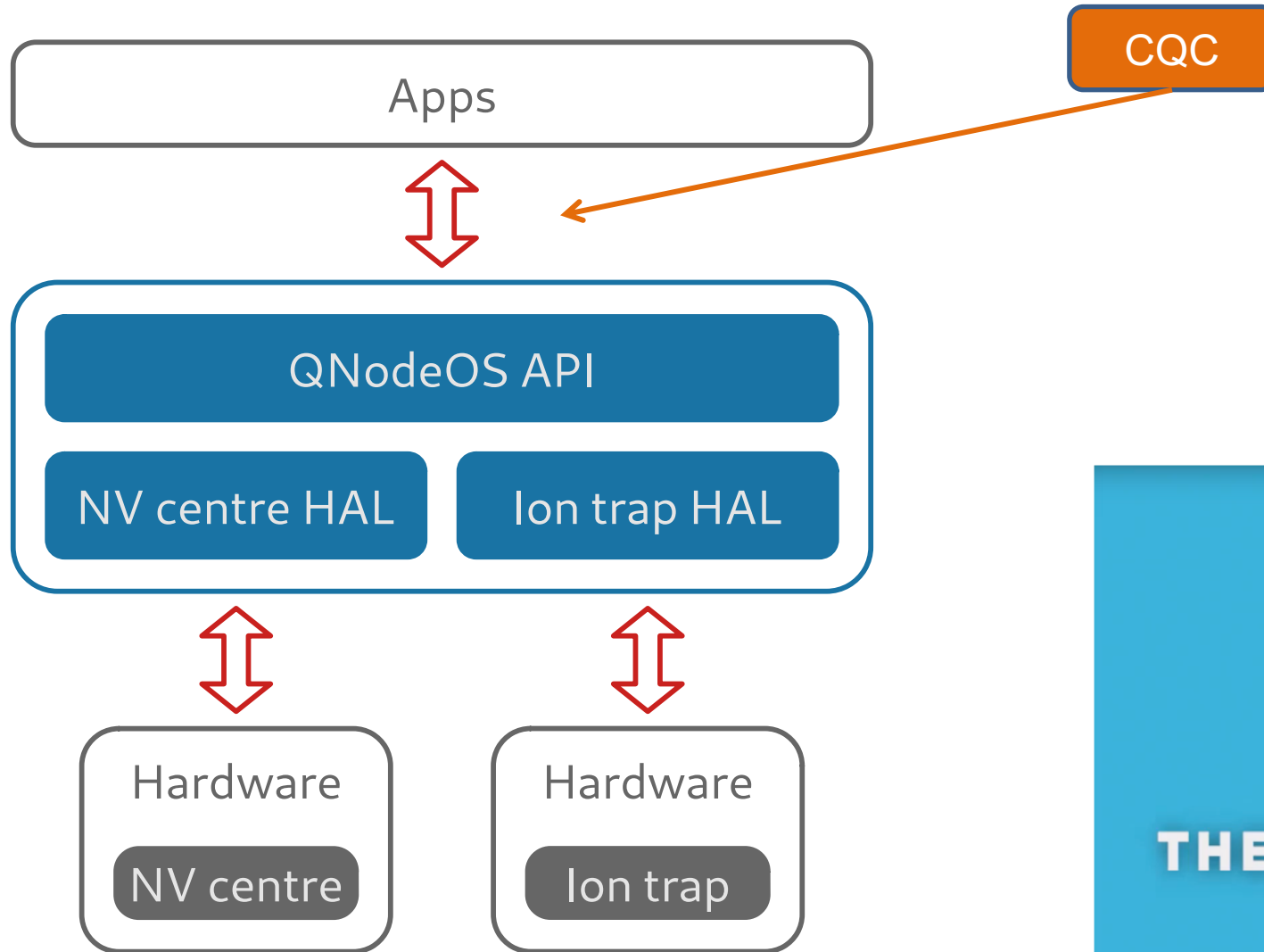
# Need for QNodeOS



# Need for QNodeOS

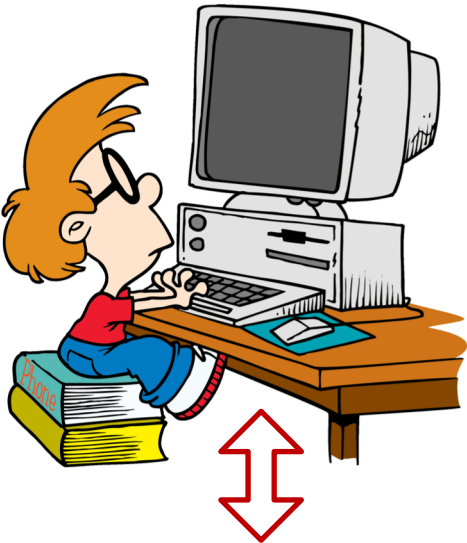
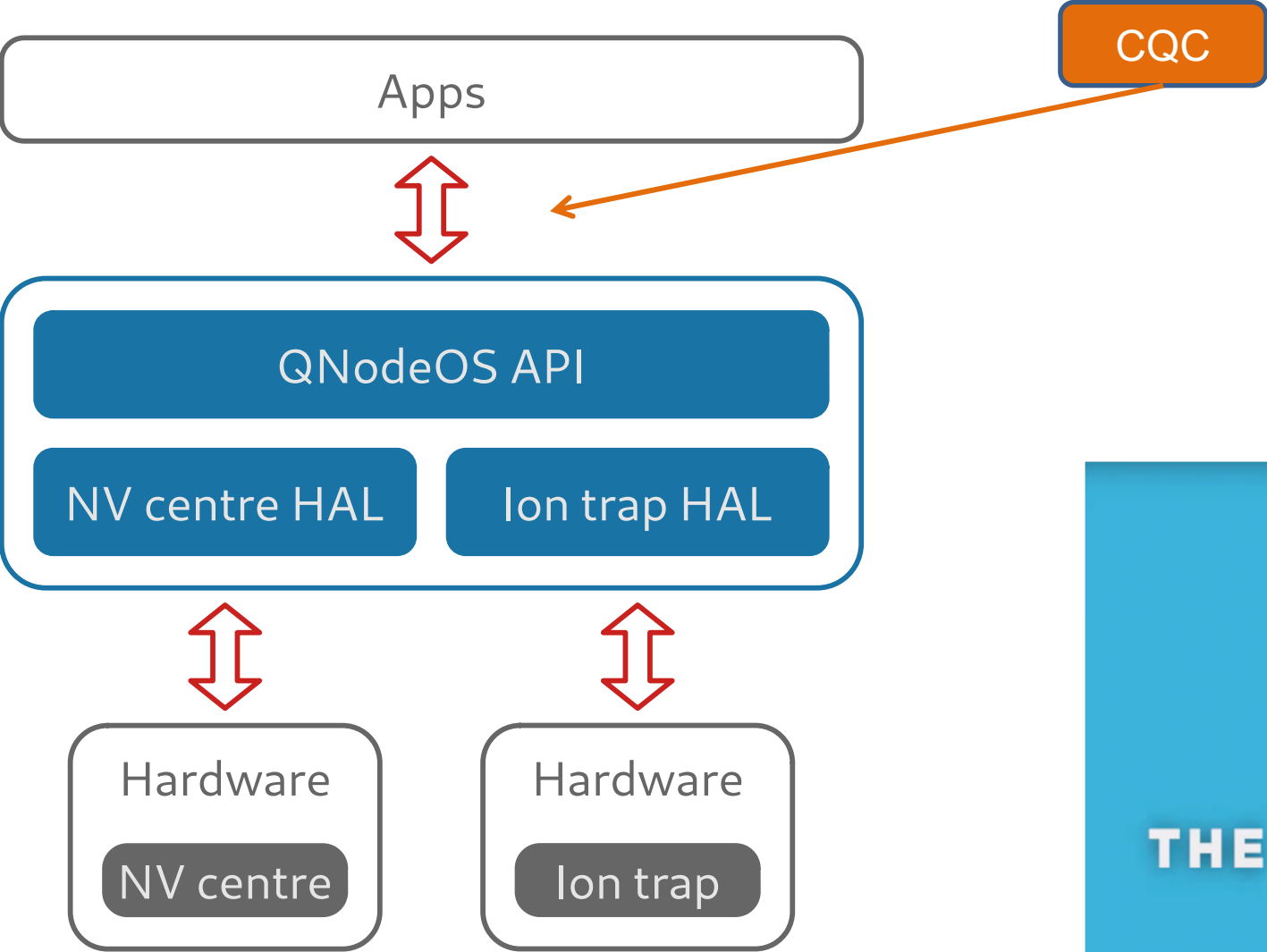


# Need for QNodeOS

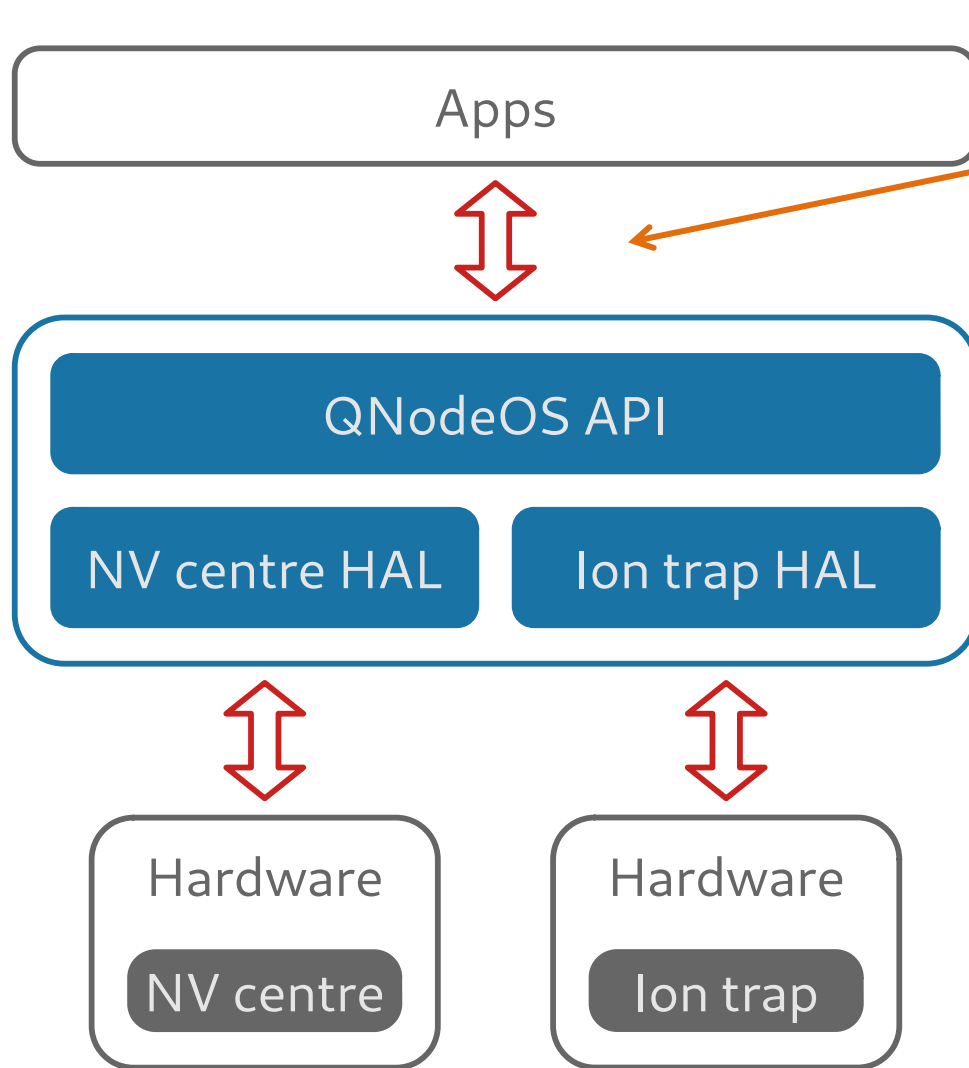




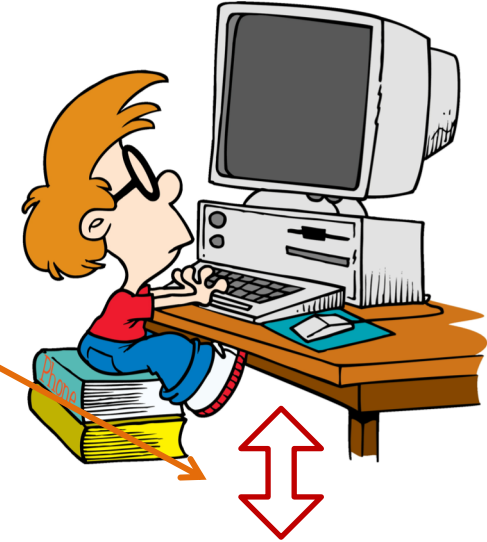
# Need for QNodeOS



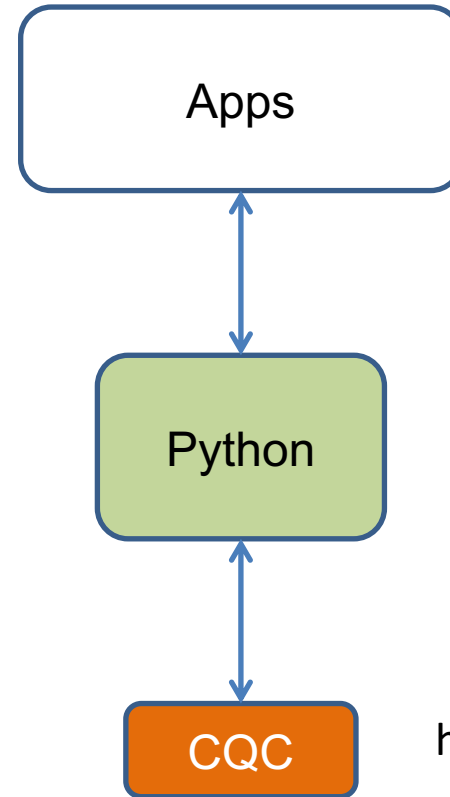
# Need for QNodeOS



CQC

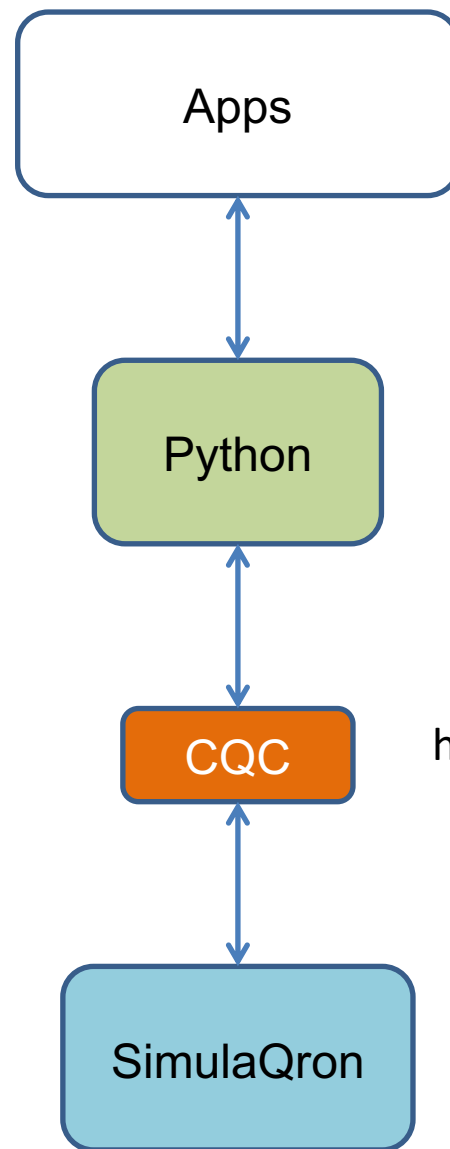


# No hardware?



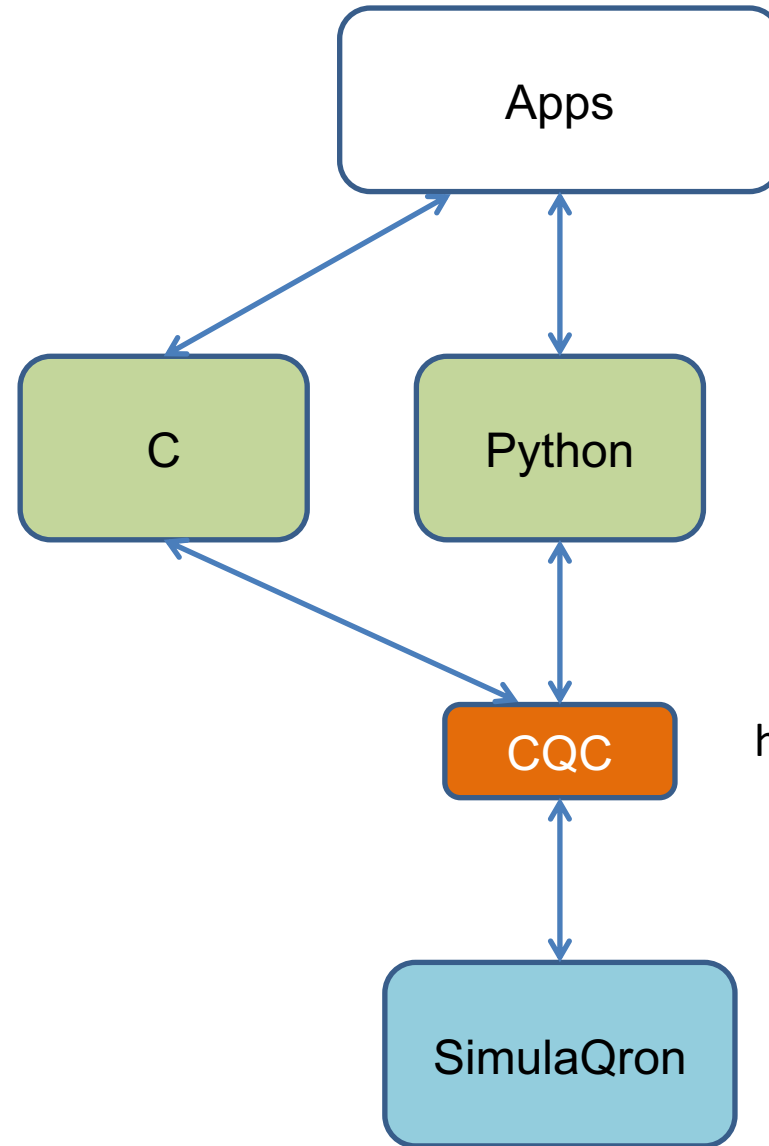
<https://github.com/SoftwareQuTech/CQC-Python>

# No hardware?



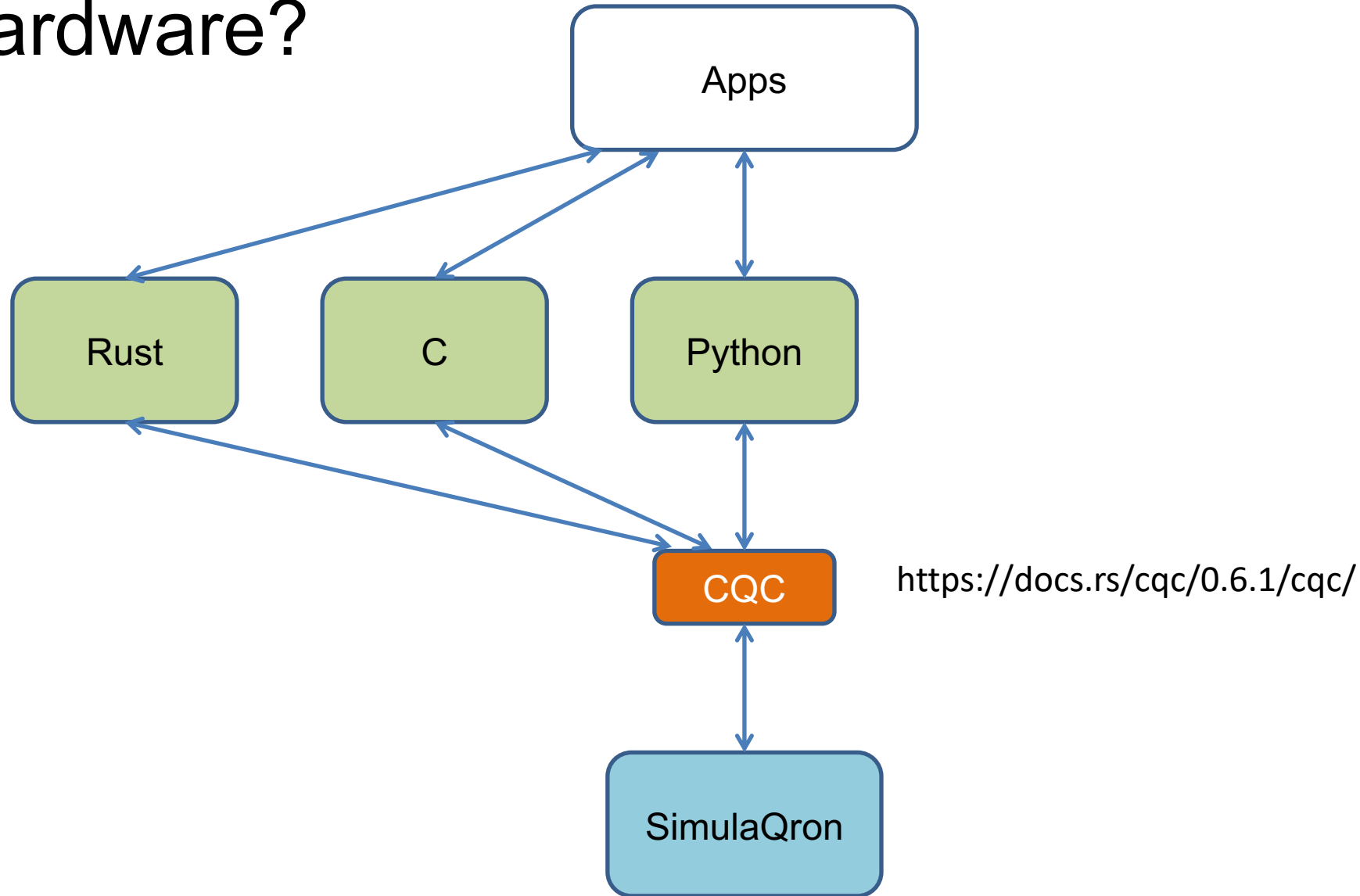
<https://github.com/SoftwareQuTech/CQC-Python>

# No hardware?

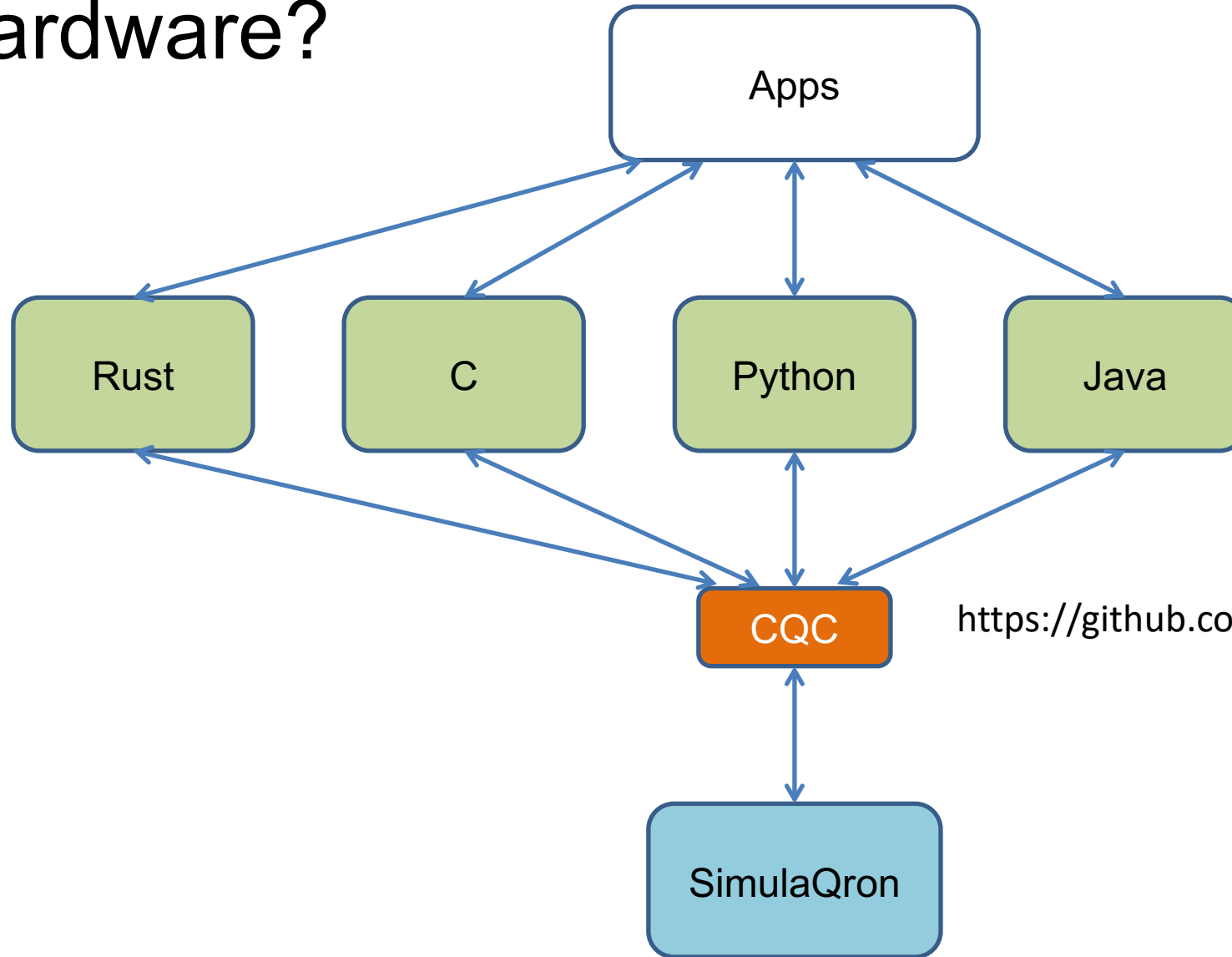


<https://github.com/SoftwareQuTech/CQC-C>

# No hardware?

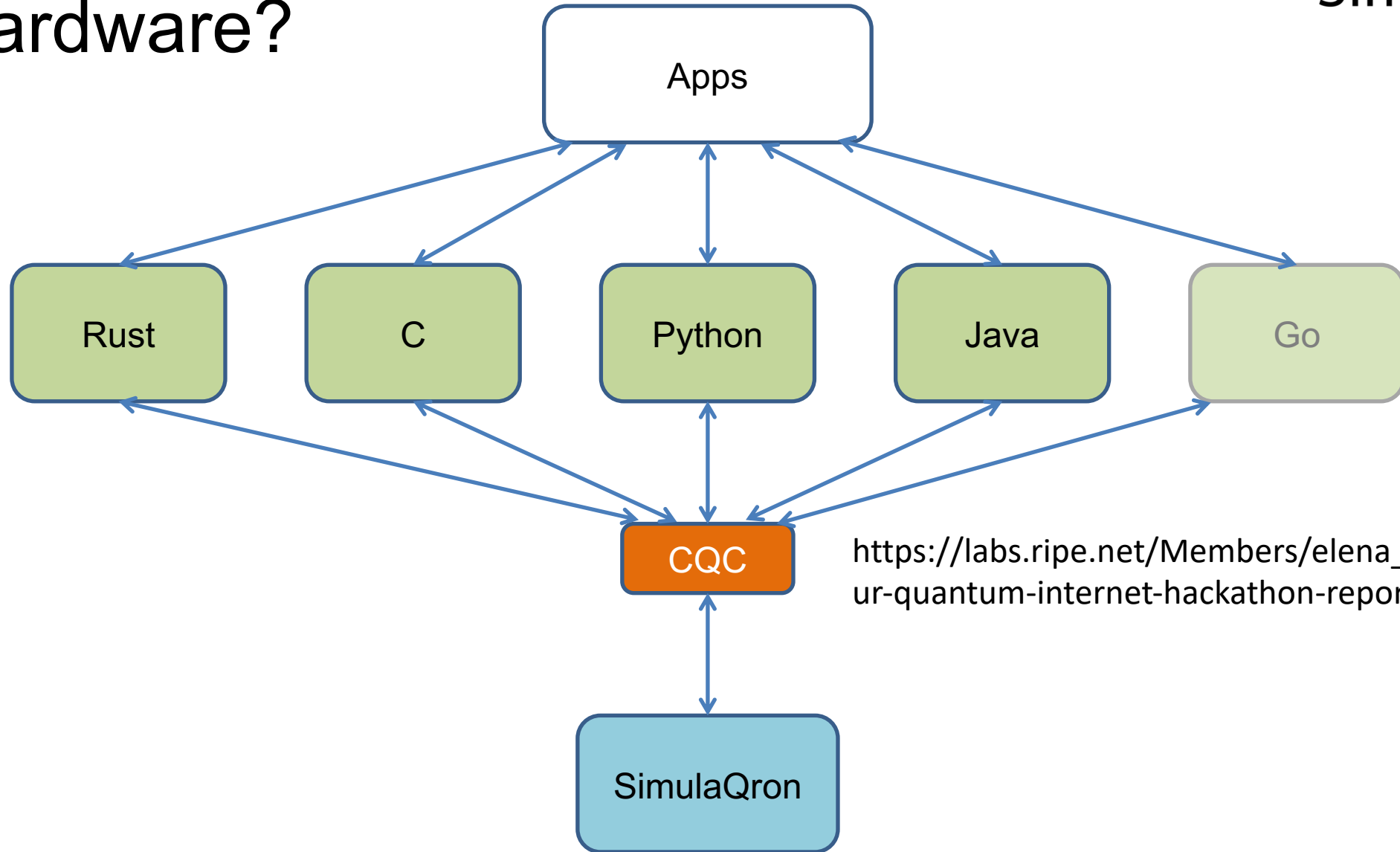


# No hardware?



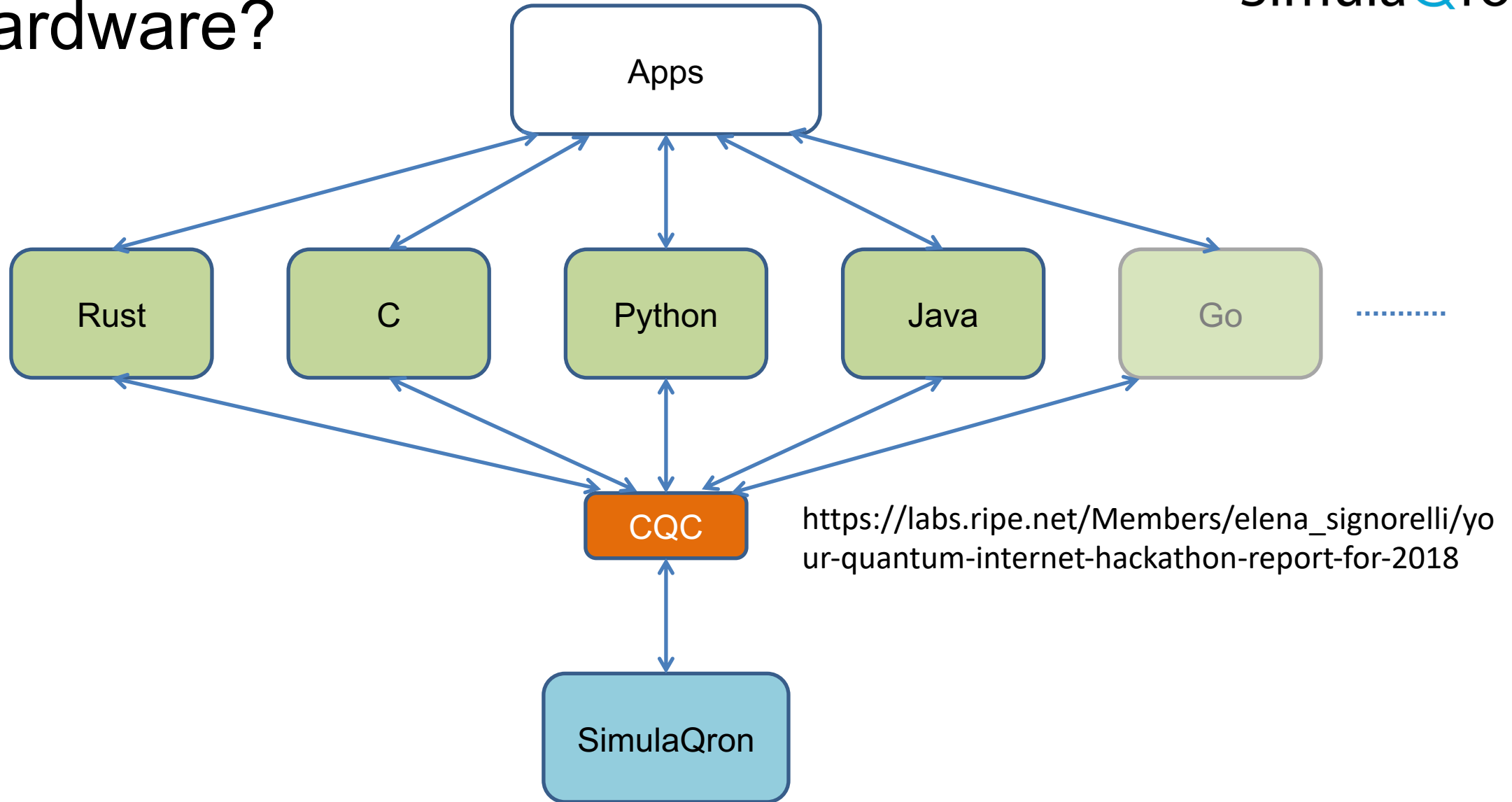
<https://github.com/gluonhq/strange/pull/19>

# No hardware?

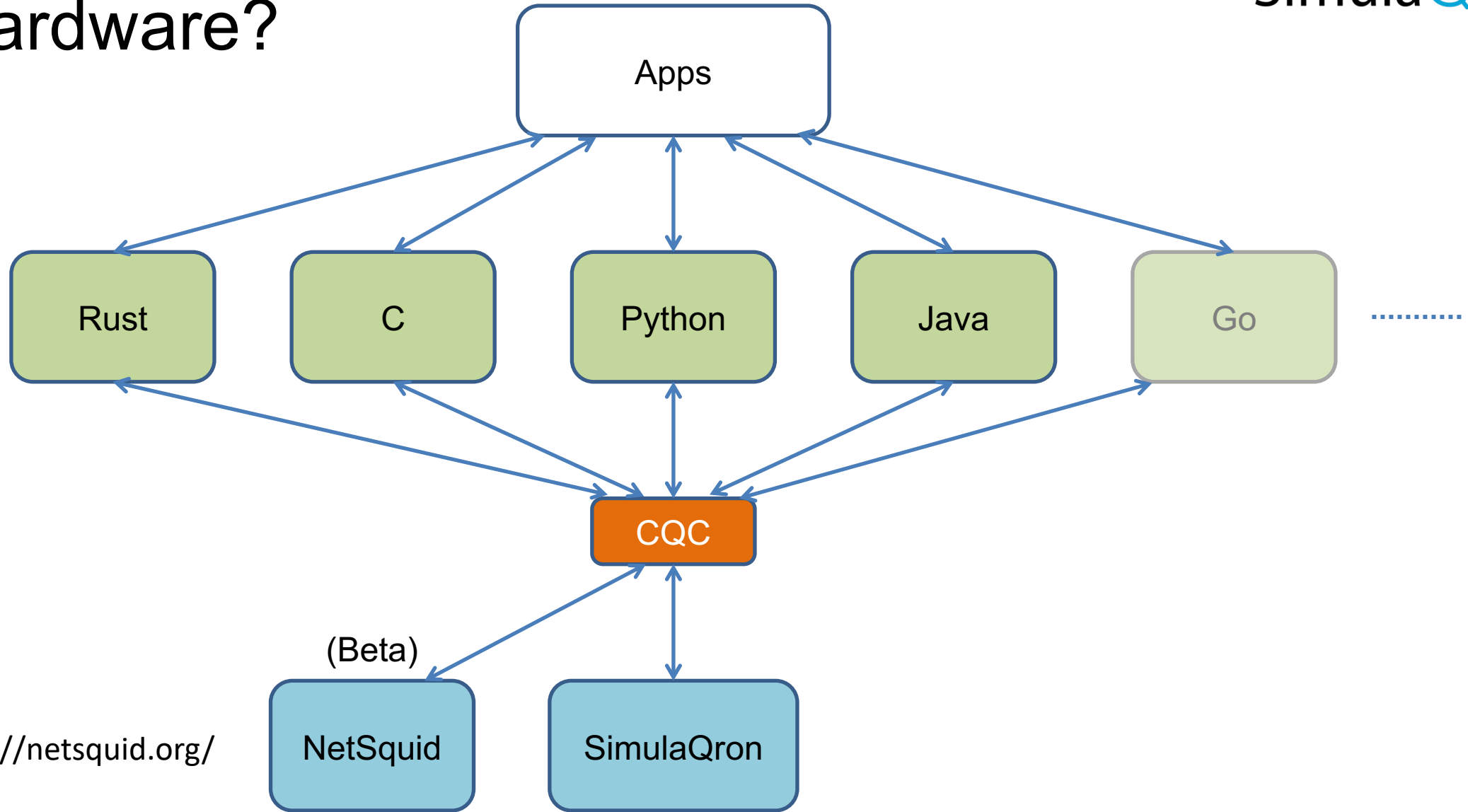




# No hardware?

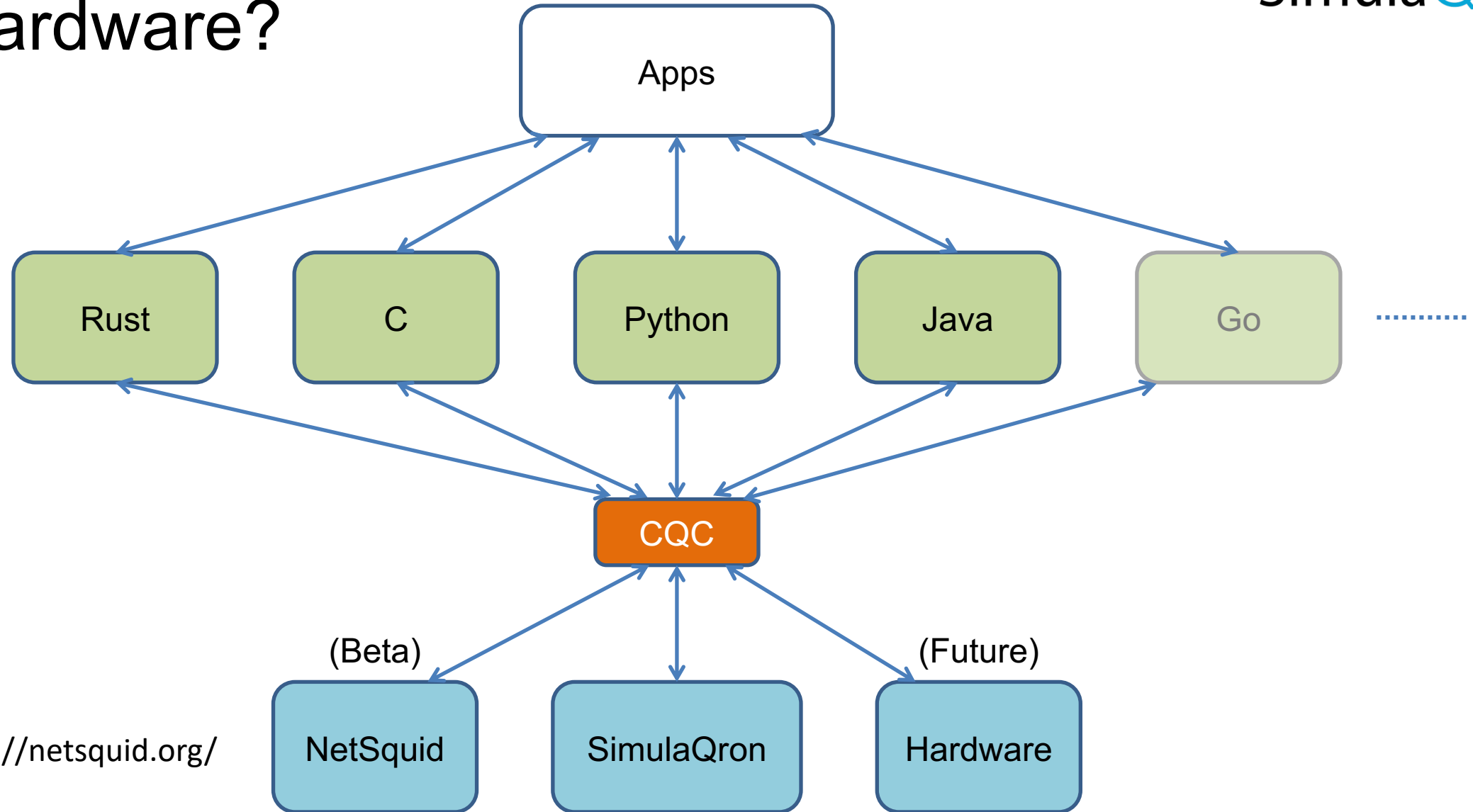


# No hardware?



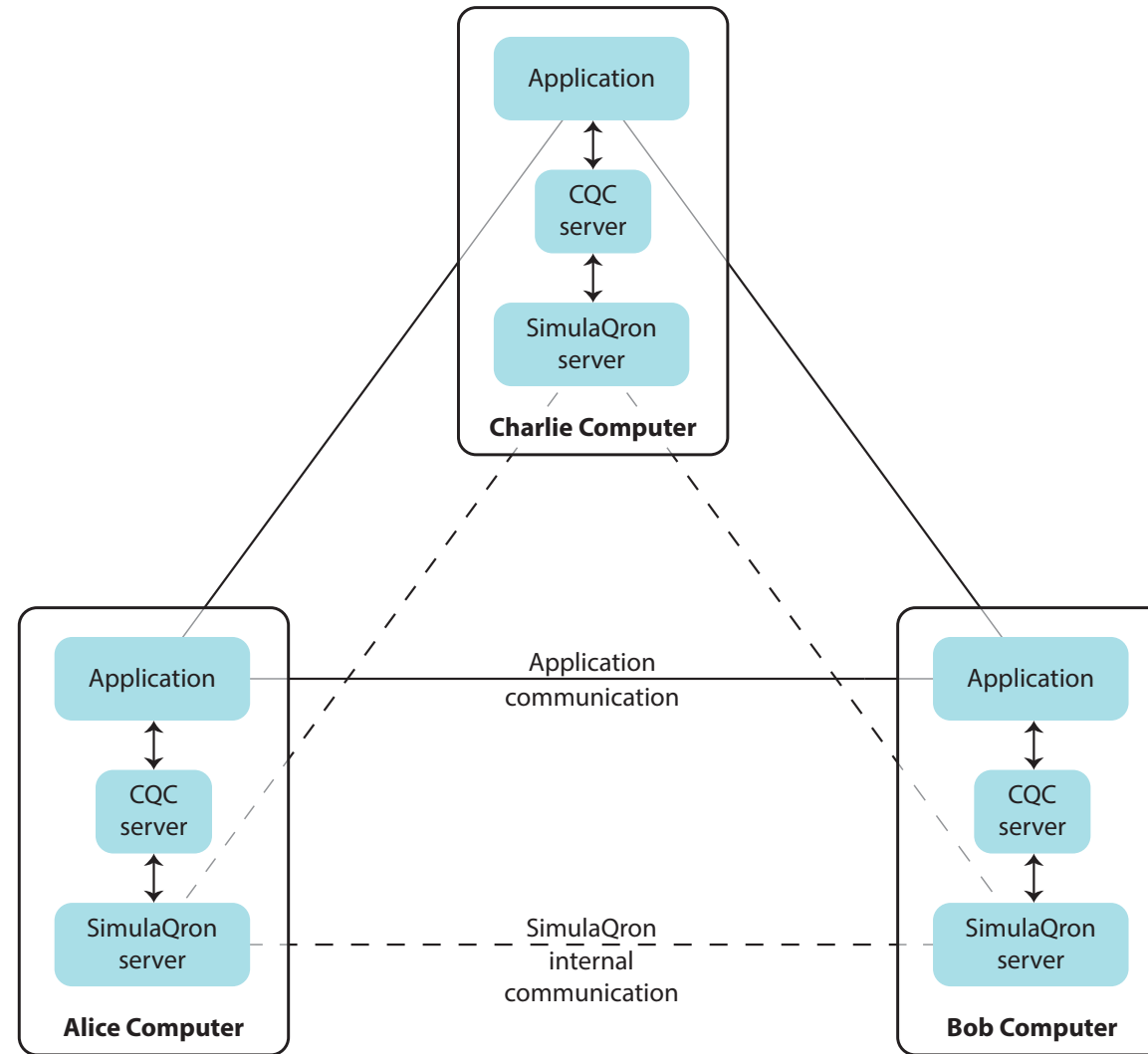
<https://netsquid.org/>

# No hardware?

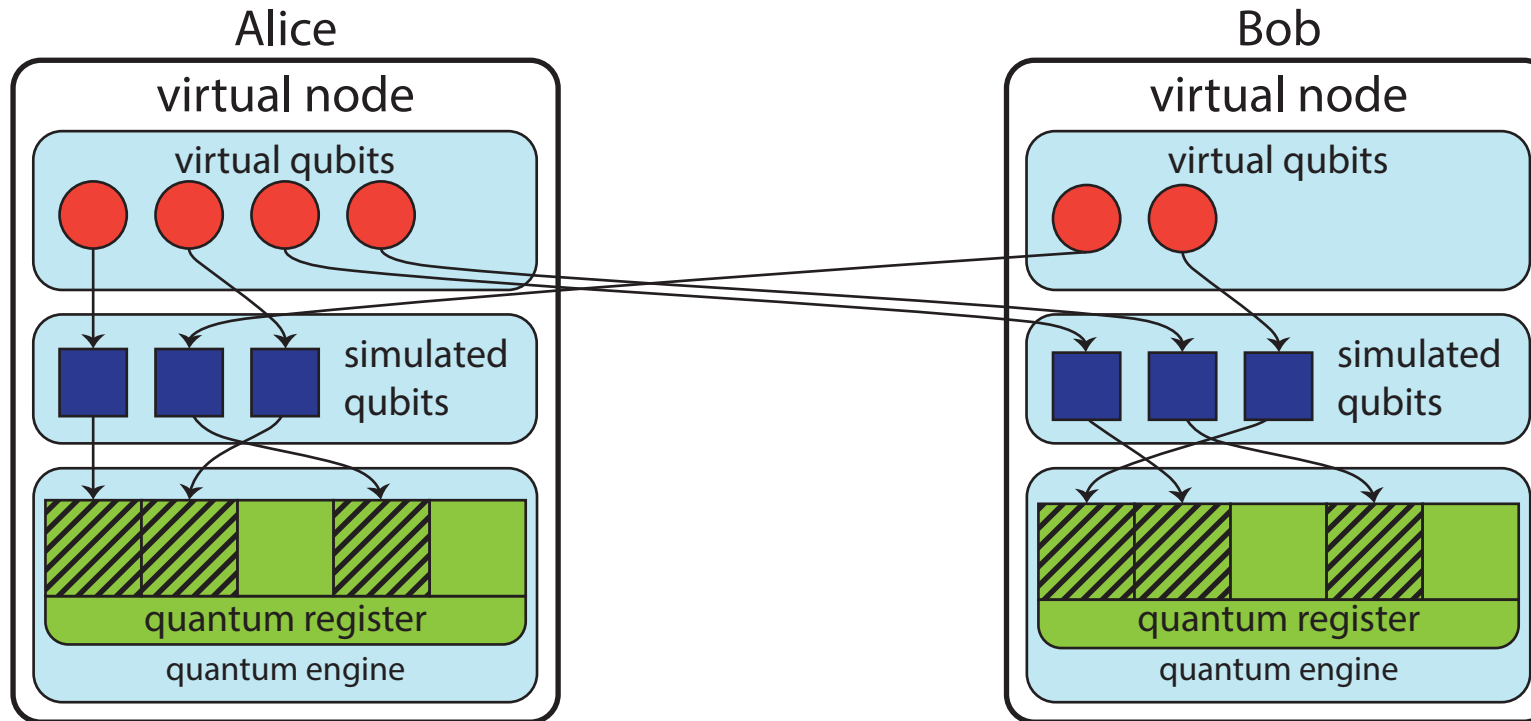


<https://netsquid.org/>

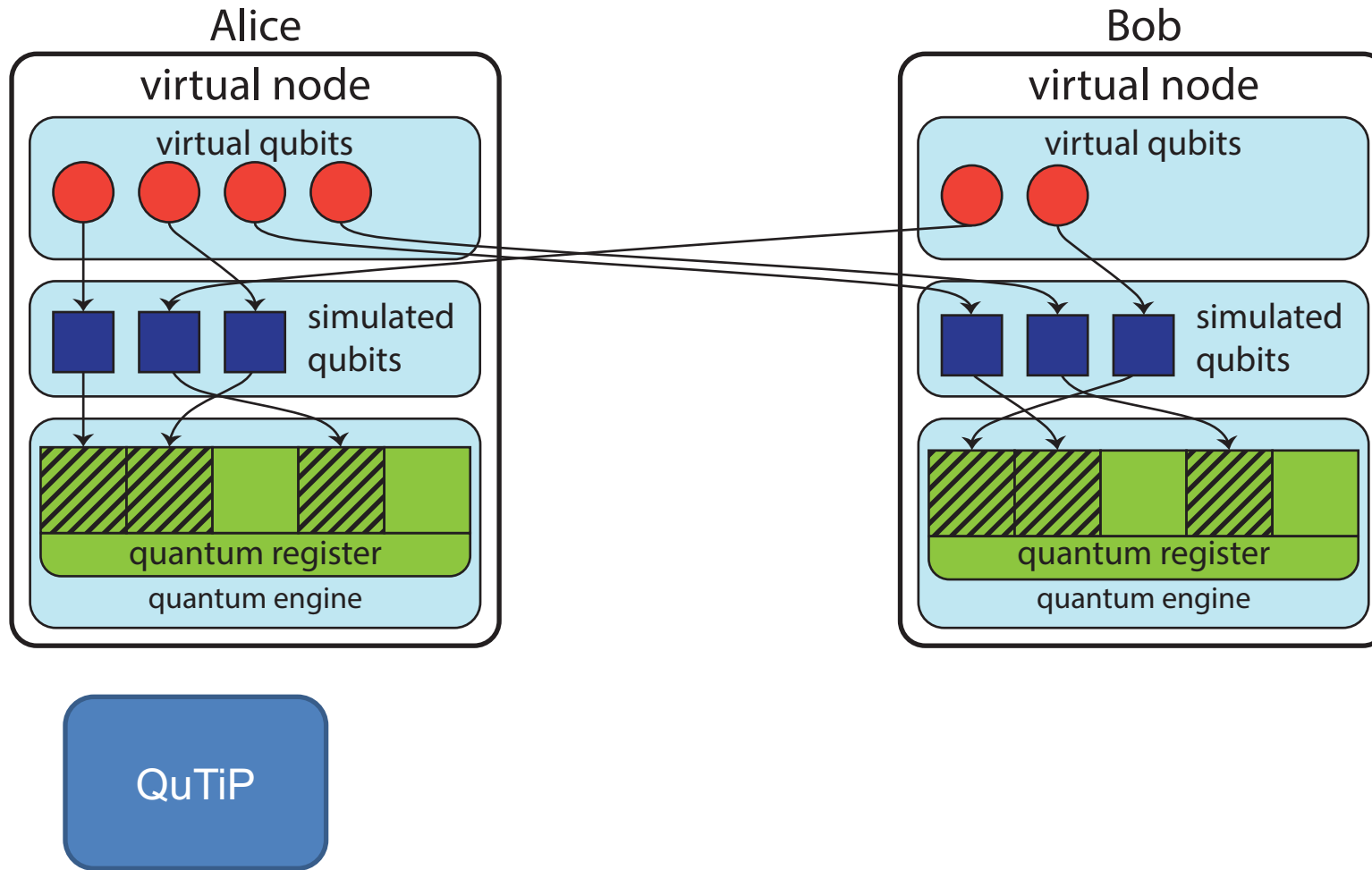
# Distributed simulation



# Nodes in SimulaQron

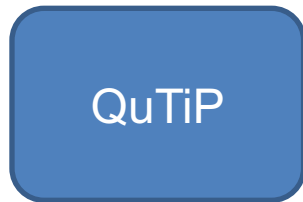
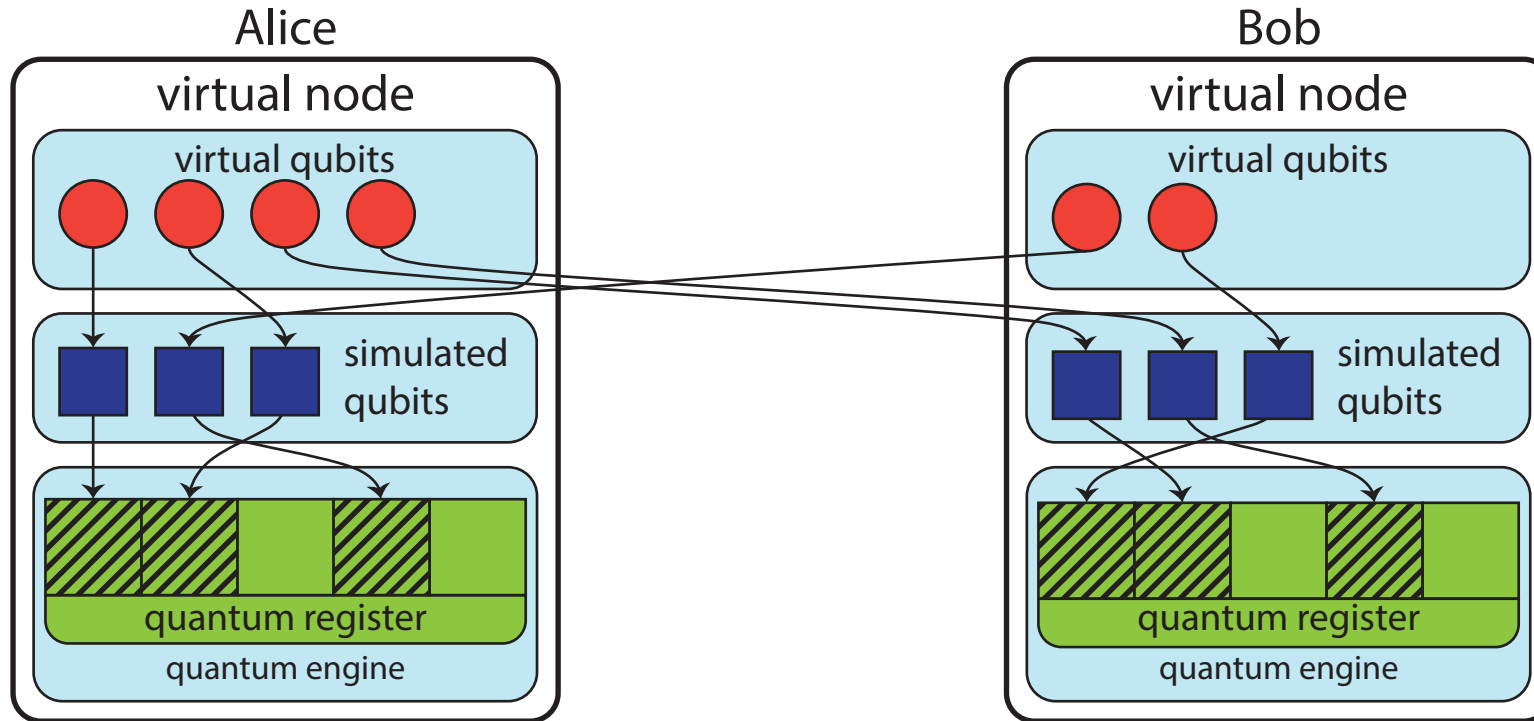


# Nodes in SimulaQron



Density matrices

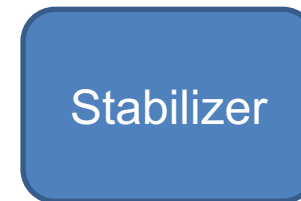
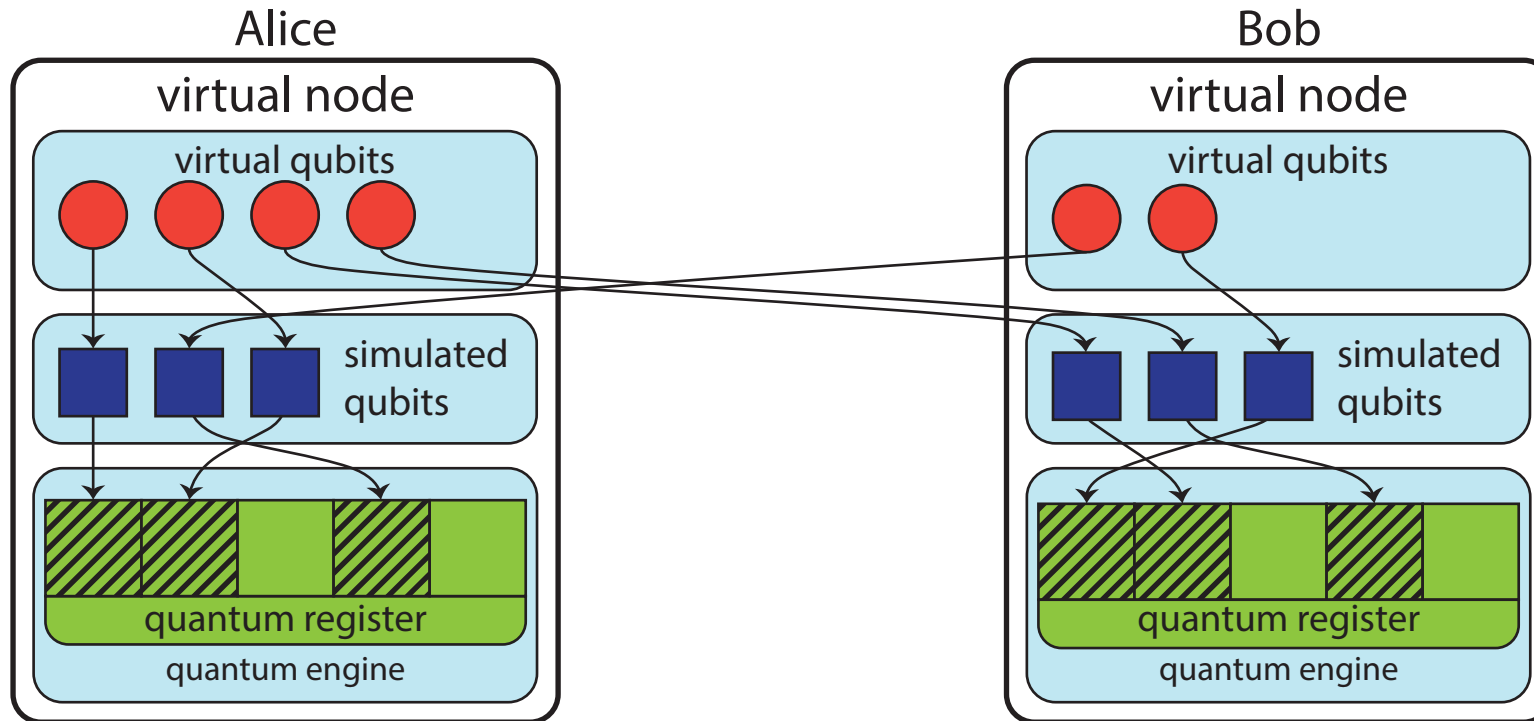
# Nodes in SimulaQron



Density matrices

Vectors

# Nodes in SimulaQron

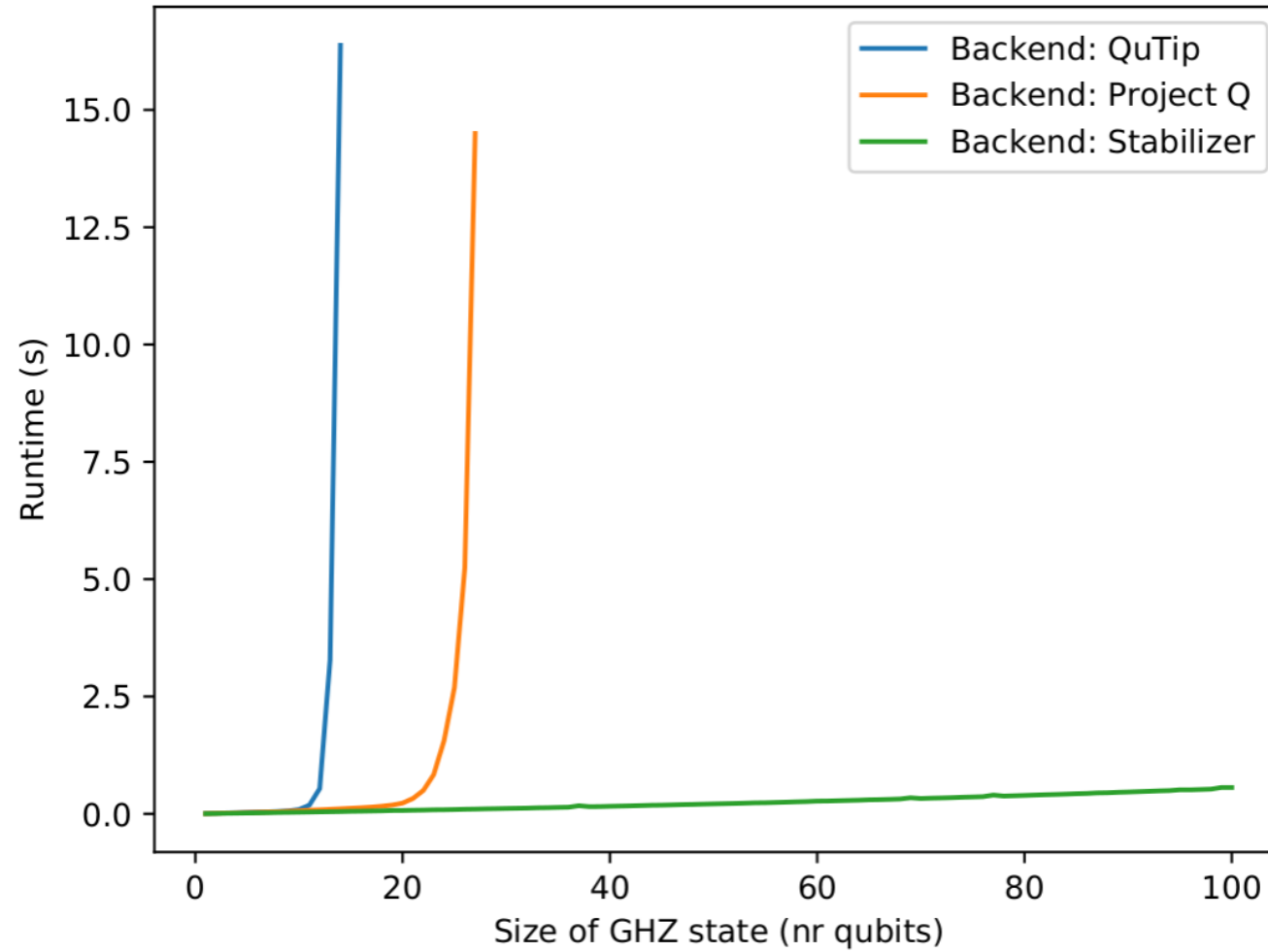


Density matrices

Vectors

Stabilizer formalism









## SimulaQron

SimulaQron is an application level simulator for a quantum internet that allows you to program your own quantum internet applications. Explore how to realize software for a quantum internet connecting local quantum processors by quantum communication, and develop your own libraries and software engineering concepts suitable for a quantum internet.

[Getting Started](#)

## What does SimulaQron do?

SimulaQron is a simulator written to provide an essential tool for software development for a quantum internet. Specifically, SimulaQron provides a distributed simulation of several quantum processors, connected by a simulated quantum communication channels. Each local quantum processor is accessible via a server running on classical computer. In the background, SimulaQron will connect these servers using classical communication to simulate the exchange of qubits and the creation of entanglement between distant processors. Each simulated processor may thereby run on a different classical computer, and supports the execution of local quantum gates and measurements, as well as commands for sending qubits to remote nodes.

Your quantum internet application can access the simulated local quantum hardware via a Python or C library, as well as a universal instruction set format (CQC). We aim to make a (possibly slightly advanced) version of CQC available also on the planned 2020 QuTech quantum internet demonstrator connecting four dutch cities. Here, we will connect quantum processors each having a few qubits to form a small quantum internet. SimulaQron allows you to develop your software already now!

SimulaQron can be used as a tool for software development in all areas ranging from the implementation of the actual applications, the development of application level abstractions and programming libraries, to exploring the implementation of a quantum network stack.

Checkout what is [under the hood](#), how to [get started](#) or our [paper](#).