Quantum Computing @ ELTE IK

Tamás Kozsik

National Research, Development and Innovation Office
Hungary

PROGRAM FINANCED FROM THE NRDI FUND

Post-quantum cryptography

Quantum communication infrastructure

Hardware security module for quantum cloud computing

Quantum Circuit Repository

Data-flow engine based acceleration

Piquasso: Photonic quantum computing

Quantum agnostic programming: Qubla

Hybrid Quantum Machine Learning

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE
HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

- Shor's algorithm: period-finding
  - Integer factoring
  - Discrete logarithm
- PKC needs PQC
- NIST calls
  - key exchange
  - digital signatures

New schemes are needed

- Finding short vectors in lattices
- Decoding linear codes
- Solving multivariate quadratic systems
- Security properties of hash functions
- Finding isogenies between supersingular elliptic curves

Post quantum cryptography

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

Our research focus is on:

- Isogeny-based constructions
- Cryptanalysis of isogeny-based and multivariate schemes
- HW implementations, side channel security
- Real-world applications, such as blockchains, positioning (Galileo), homom. enc., E2E messaging

Researchers

- Péter Kutas
- Péter Burcsi
- Péter Ligeti
- et al.

poqeth: Efficient, post-quantum signature verification on Ethereum (Proc. ACM ASIACCS, 2025)

Post quantum cryptography

National Research, Development and Innovation Office Hungary
PROGRAM FINANCED FROM THE NRDI FUND

- Quantum cryptography
- Symmetric / secret key cryptography
- Quantum Key Distribution (QKD)
- No-cloning theorem against eavesdropping
- BB84 etc.

QCI Hungary

- ELTE IK
- Pro-M Zrt.
- HUN-REN Wigner
- BME VIK

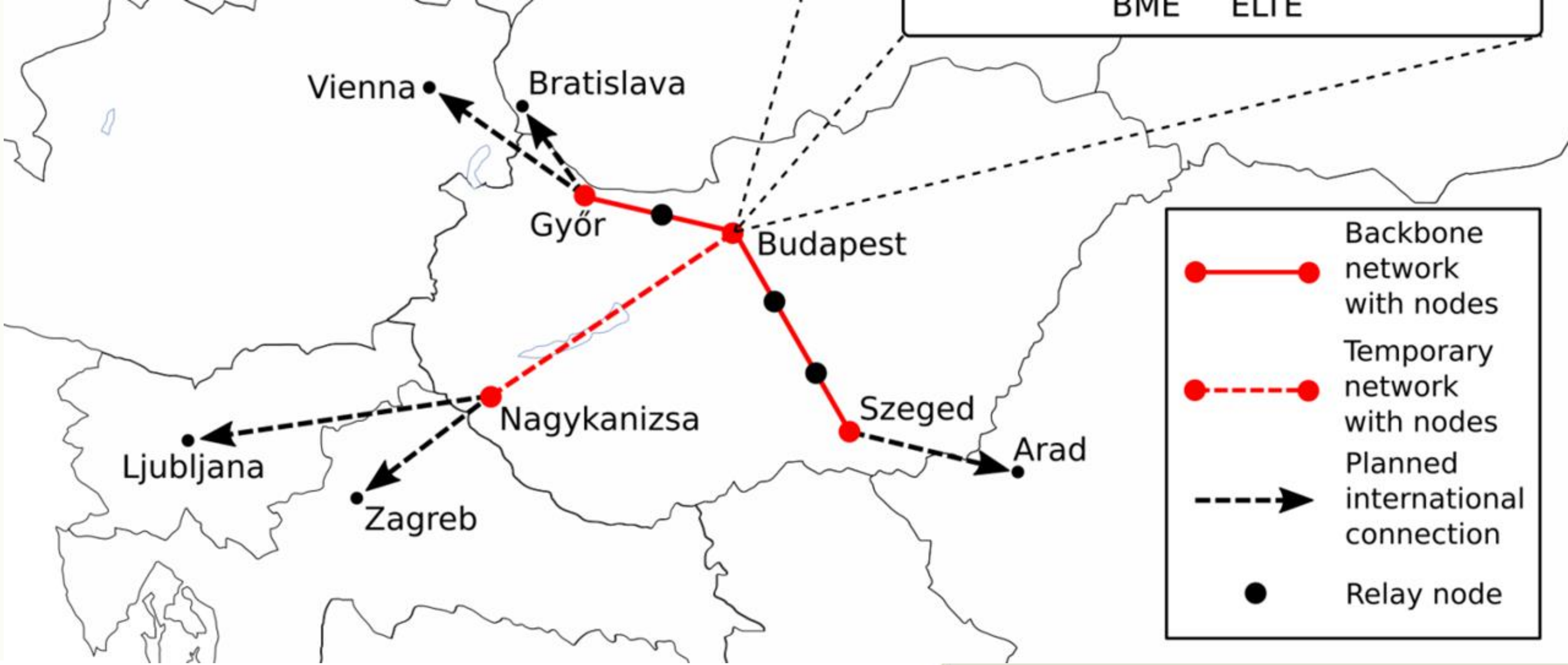
Europe is building a…

… CEQCI …

Quantum communication infrastructure

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

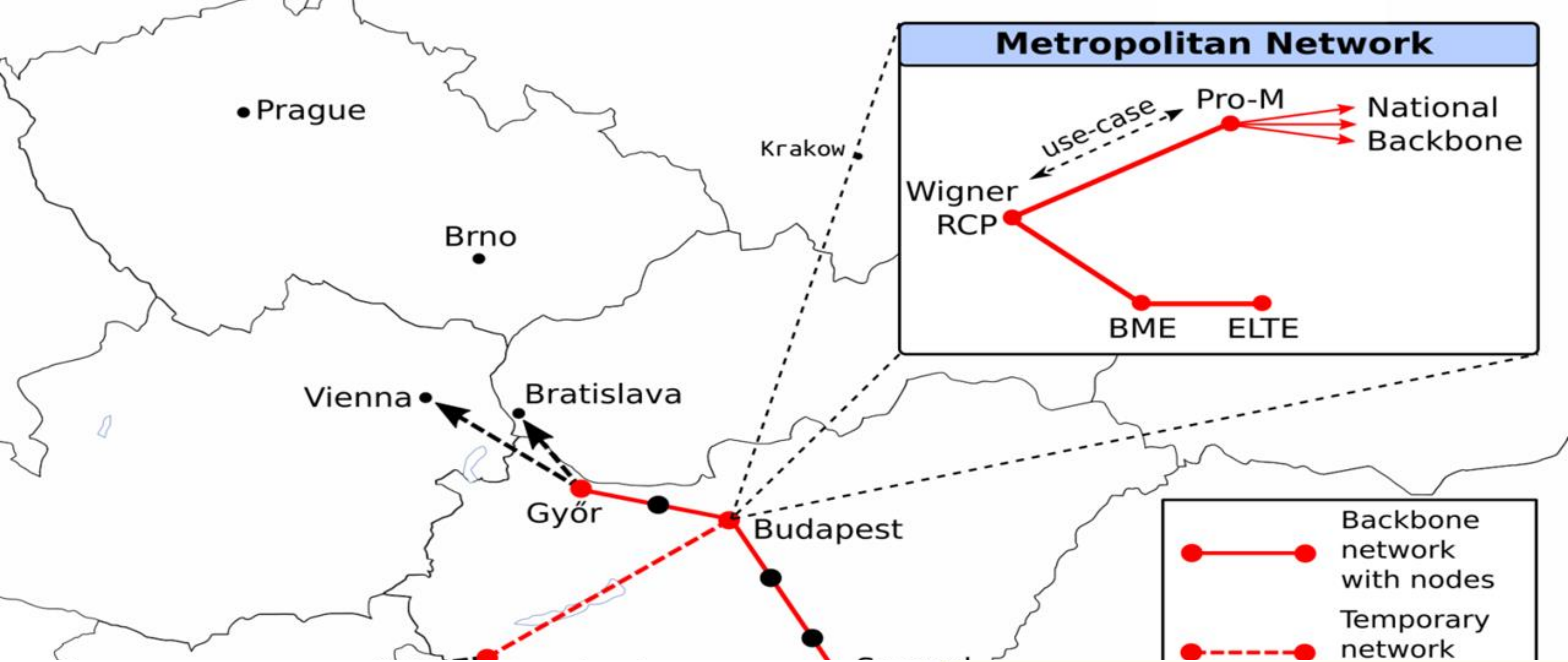
PROGRAM FINANCED FROM THE NRDI FUND



Quantum communication infrastructure

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE
HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

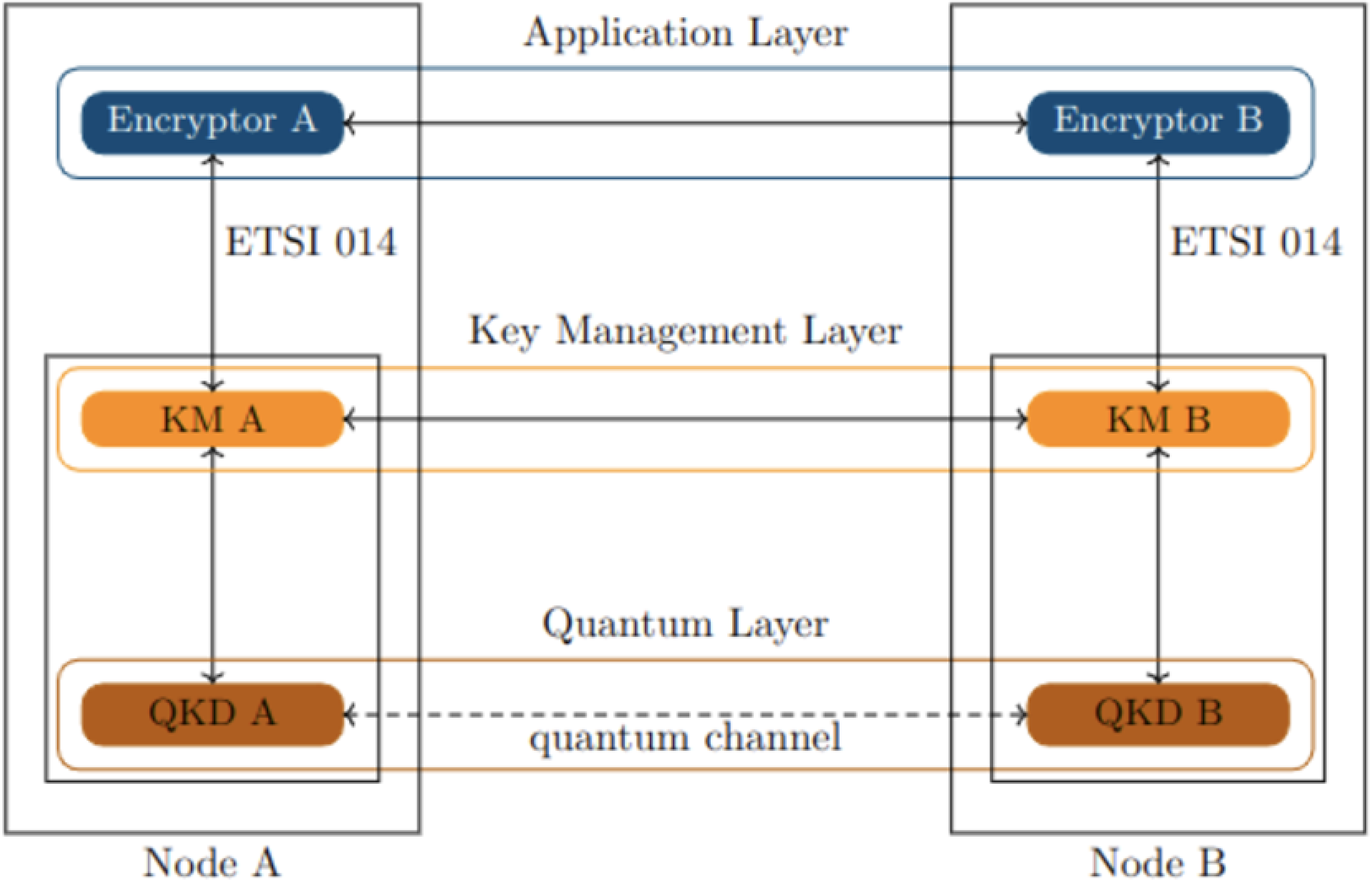


Quantum communication infrastructure

National Research, Development and Innovation Office
Hungary
PROGRAM FINANCED FROM THE NRDI FUND

Our R+D+I focus

- Software stack for QCI
- Hybrid: QKD and PQC
- Compatibility
- Open source



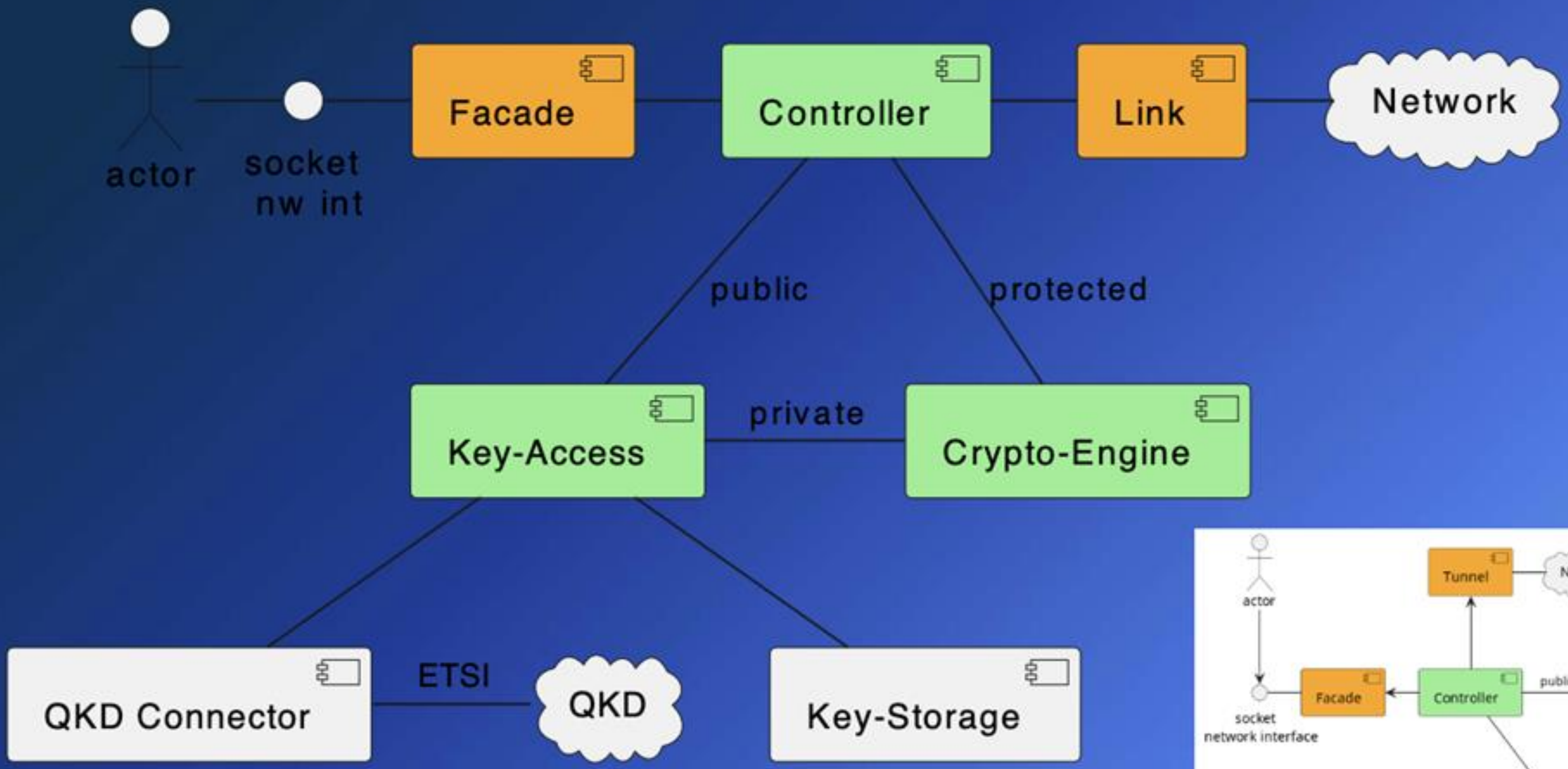
Quantum communication infrastructure

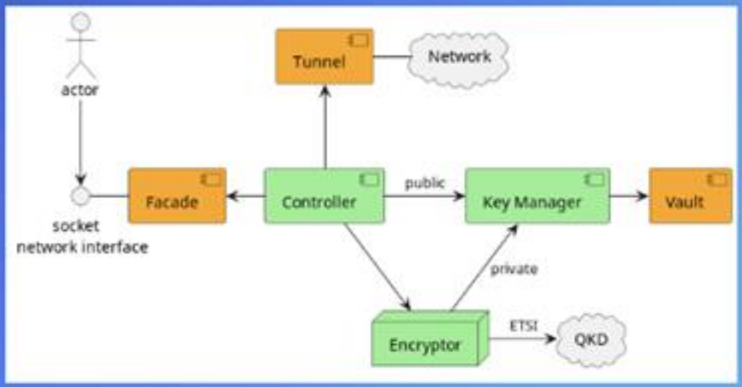
National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

QCIHungary

System architecture





QCIHungary Testing and software quality

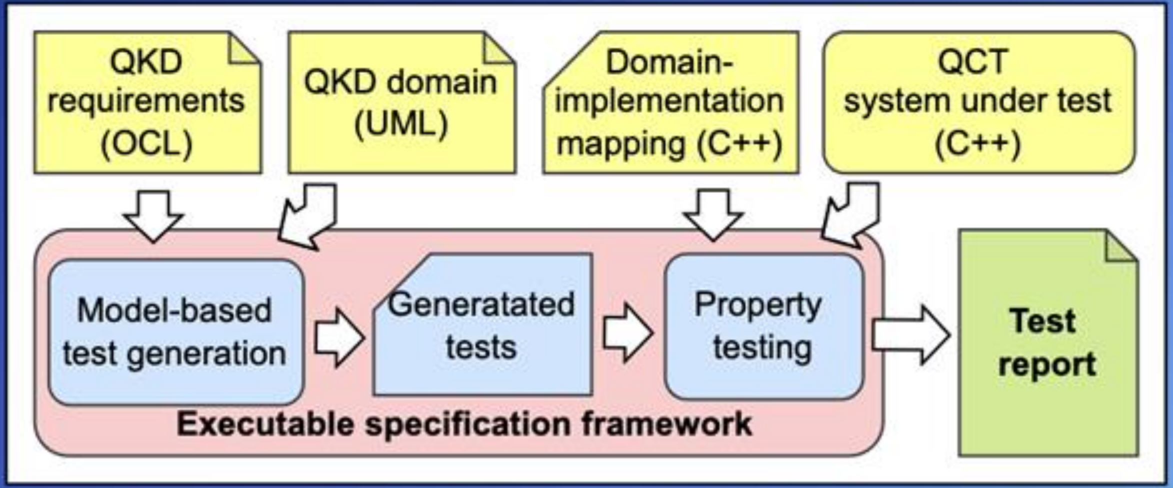
End-to-end testing: configuration generation

(Status: ready for integration)

- Property testing based on JSON Schema.

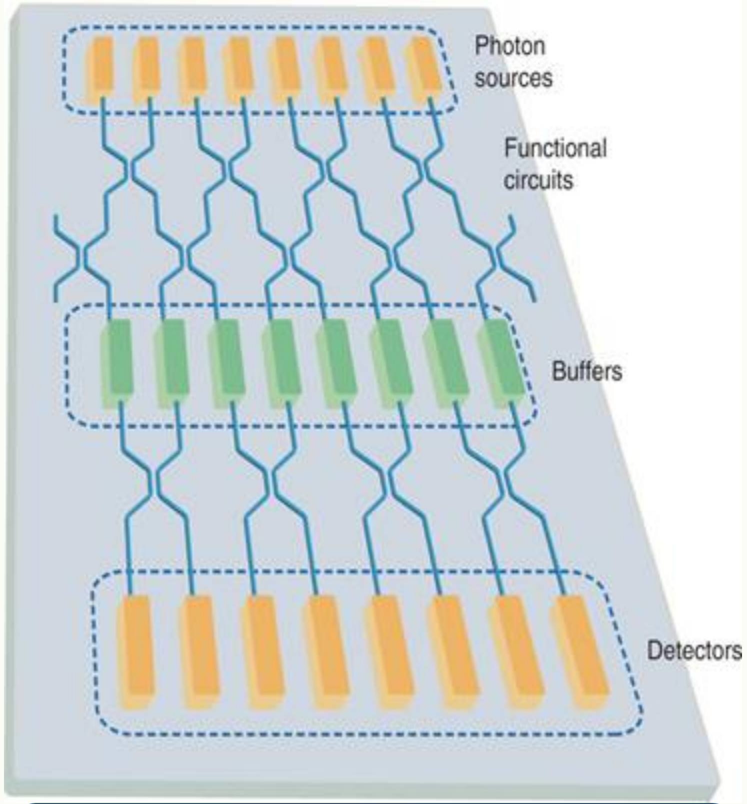
Test automation (Status: proof-of-concept)

- Goal: Executable formal specification
  - Based on ETSI 016, Common Criteria
  - UML, OCL
- Test generation
  - ModelTestRelax (ELTE IK)
  - RapidCheck, Z3



- photon based computational model
- qumodes instead of qubits
- beam splitter, phase shifter, squeezing, displacement
- non-linearity (cross-Kerr interaction)
- measurement-based quantum computing

- experimental demonstration of quantum advantage
- room temperature operation
- mature optical technologies



Photonic quantum processor, concept
From: N. Matsuda and H. Takesue
DOI:10.1515/nanoph-2015-0148

Piquasso: Photonic quantum computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY
PROGRAM FINANCED FROM THE NRDI FUND

Photonic Quantum Computer Simulation Platform

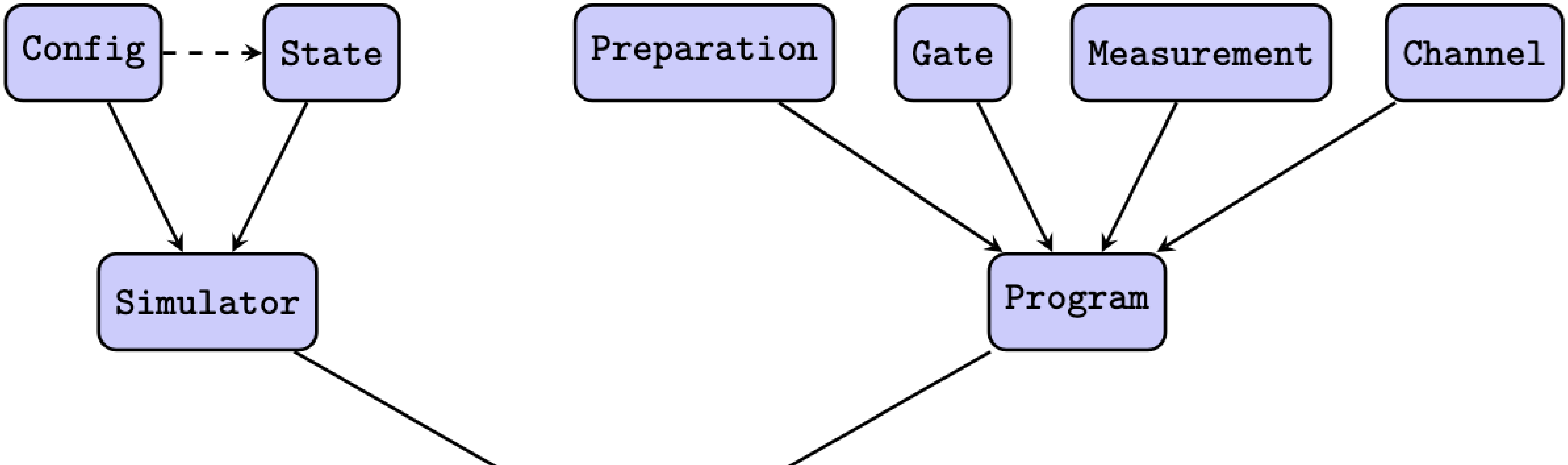
- Discrete and continuous-variable
- Simulation, analysis, differentiation
- Programming in Python (embedded DSL)
- Python and C++ backends
- Support for (Gaussian) Boson Sampling
- Support for photonic quantum machine learning
- Open source

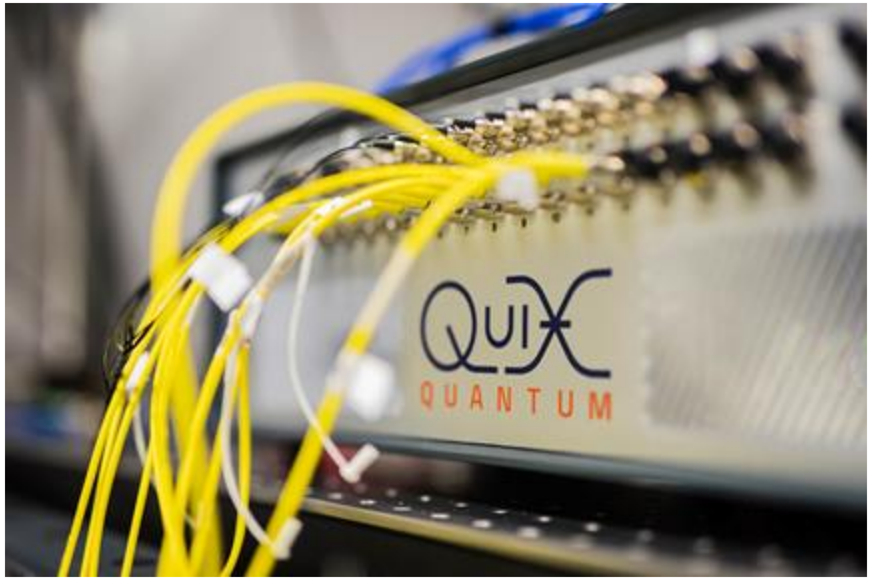
PIQUASSO

Piquasso: Photonic quantum computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND


Config
State
Preparation
Gate
Measurement
Channel
Simulator
Program
Simulator.execute
Result


QUANTUM

```python
import numpy as np
import piquasso as pq

# Beginning of program definition
with pq.Program() as program:
    # Initializing vacuum state
    pq.Q(all)  | pq.Vacuum()

    # Quantum Gates
    pq.Q(0)     | pq.Displacement(r=1.0)
    pq.Q(1)     | pq.Displacement(r=1.0)
    pq.Q(0)     | pq.Squeezing(r=0.1, phi=np.pi / 3)
    pq.Q(0, 1) | pq.Beamsplitter(theta=np.pi / 3, phi = np.pi / 4)

    # Measurement
    pq.Q(0, 1) | pq.ParticleNumberMeasurement()

# Choosing a simulator
simulator = pq.GaussianSimulator(d=2)

# Execution
result = simulator.execute(program, shots=1000)
```

PIQUASSO

Piquasso: Photonic quantum computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

a.) b.)

piquasso.com

Figure 14: An overview of the Piquasso web interface. The user should (a) choose a supported backend scheme before creating a circuit, and then (b) use the interactive drag-and-drop composer to create the circuit from the relevant components of the chosen scheme. The depicted example is a Gaussian Boson Sampling circuit, the simulation results are visualized in Fig. 15.

Piquasso: Photonic quantum computing

National Research, Development and Innovation Office
Hungary

PROGRAM
FINANCED FROM
THE NRDI FUND

Particle Number Detection Probabilities
Export Data
Export Image
0.4
0.3
0.2
0.1
0
0.09063918
000 001 010 100 002 011 020 101 110 200 003 012 021 030 102 111 120 201 210 300 004 013 022 031 040 103 112 121 130 202 211 220 301 310 400
Samples
Export Data
Export Image
Qumode 0
Qumode 1
Qumode 2
56 Qumode 2
55 Qumode 0
52 Qumode 1
Qumode 0
Qumode 2
Qumode 1
50
40
30
20
10
0
0 1 2 3 4
Wigner Function
Export Image
Qumode 0
Qumode 1
Qumode 2
x: 0.3237526
y: 0.8093815
z: 0.05601013
W(x,p)
0.05
0.08
0.06
0.04
0.02
0
p
x
−15 −10 −5 0 5 10
Python Code
Copy to clipboard
import numpy as np
import piquasso as pq
with pq.Program() as program:
pq.Q(0) | pq.Squeezing(r=0, phi=0)
pq.Q(1) | pq.Squeezing(r=1, phi=0)
pq.Q(2) | pq.Squeezing(r=1, phi=0)
pq.Q(0) | pq.Phaseshifter(phi=1.0472)
pq.Q(1) | pq.Phaseshifter(phi=1.0472)
pq.Q(2) | pq.Phaseshifter(phi=1.0472)
pq.Q(0, 1) | pq.Beamsplitter(theta=0.3927, phi=0.7854)
pq.Q(1, 2) | pq.Beamsplitter(theta=0.3927, phi=0.7854)
pq.Q(0, 1) | pq.Beamsplitter(theta=0.3927, phi=0.7854)
pq.Q(0, 1, 2) | pq.ParticleNumberMeasurement()
simulator = pq.GaussianSimulator(
d=3, config=pq.Config(cutoff=5)
)
result = simulator.execute(program, shots=100)

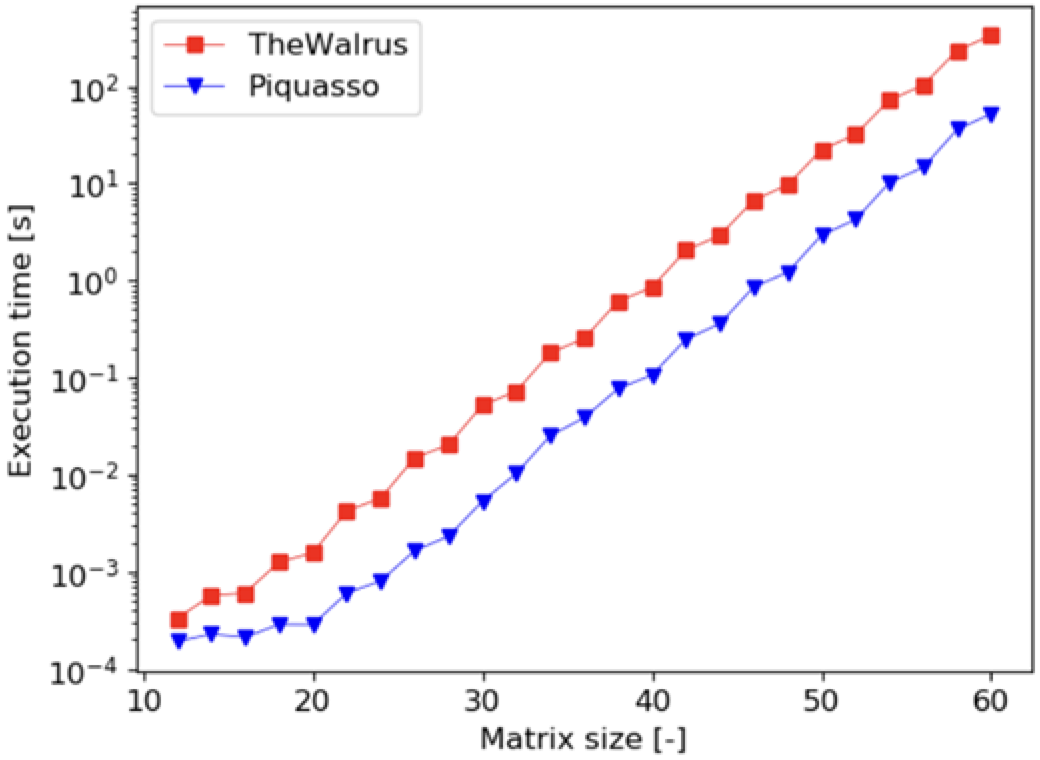


Figure 5: Comparing the average repeated loop hafnian execution times of Piquasso (version 5.0.0) and TheWalrus (version 0.21.0) with increasing input matrix sizes.

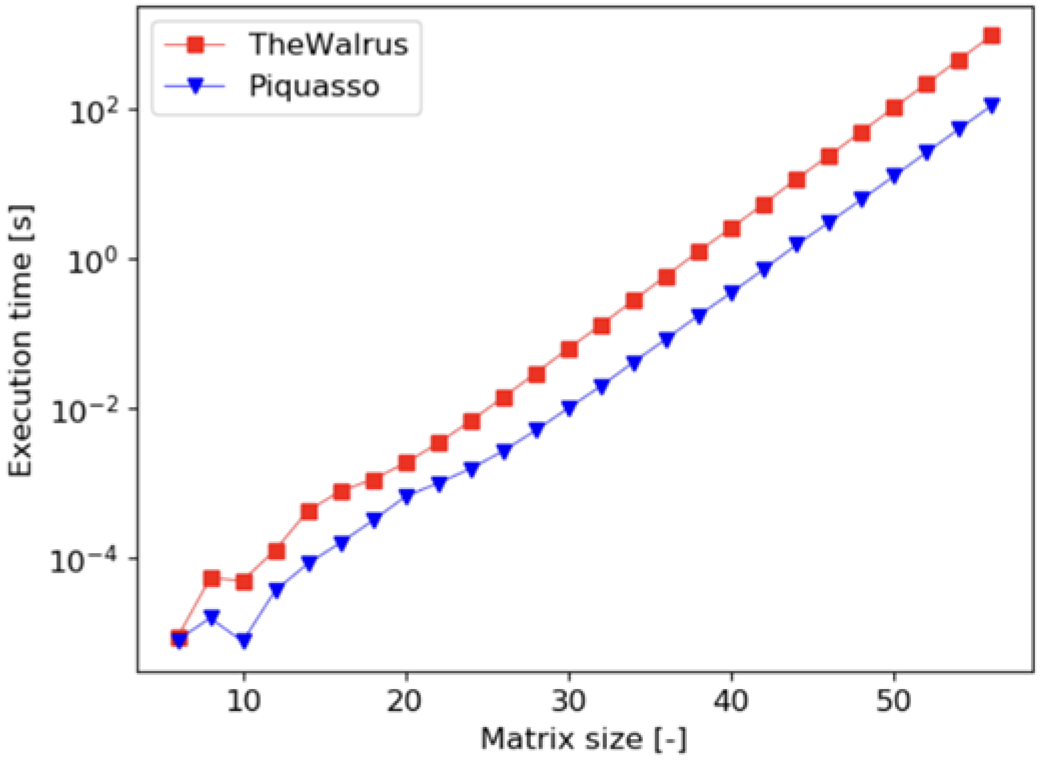


Figure 7: Comparing the average torontonian execution times of Piquasso (version 5.0.0) and TheWalrus (version 0.21.0) with increasing input matrix sizes. When

Piquasso: Photonic quantum computing

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

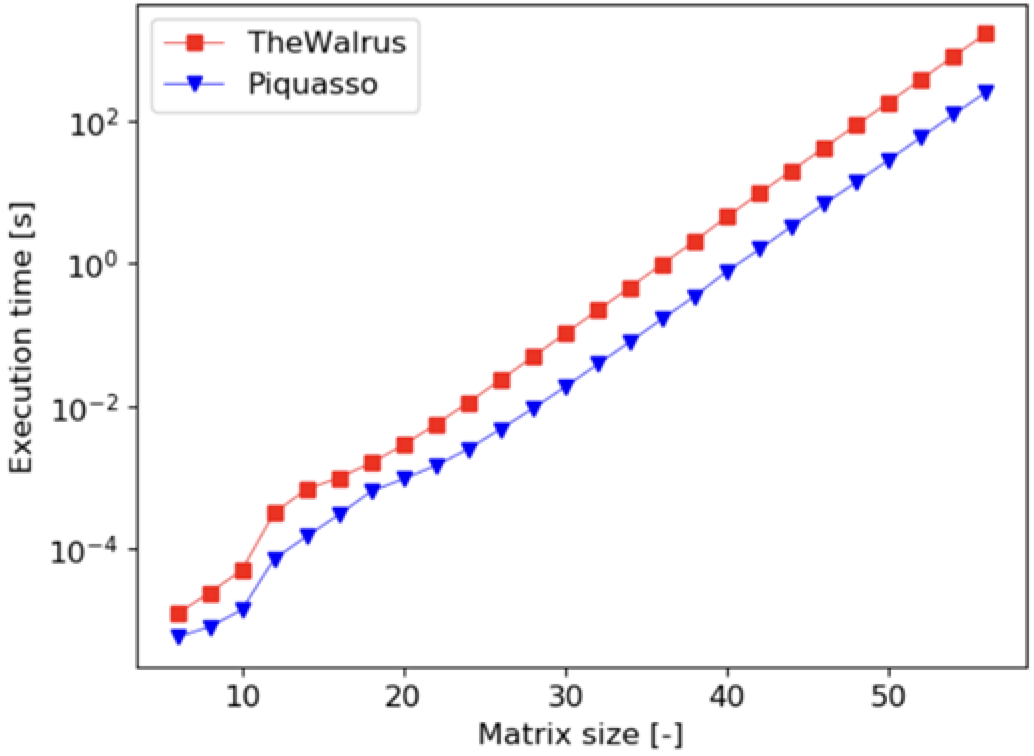


Figure 8: Comparing the average loop torontonian execution times of Piquasso (version 5.0.0) and TheWalrus (version 0.21.0) with increasing input matrix sizes.

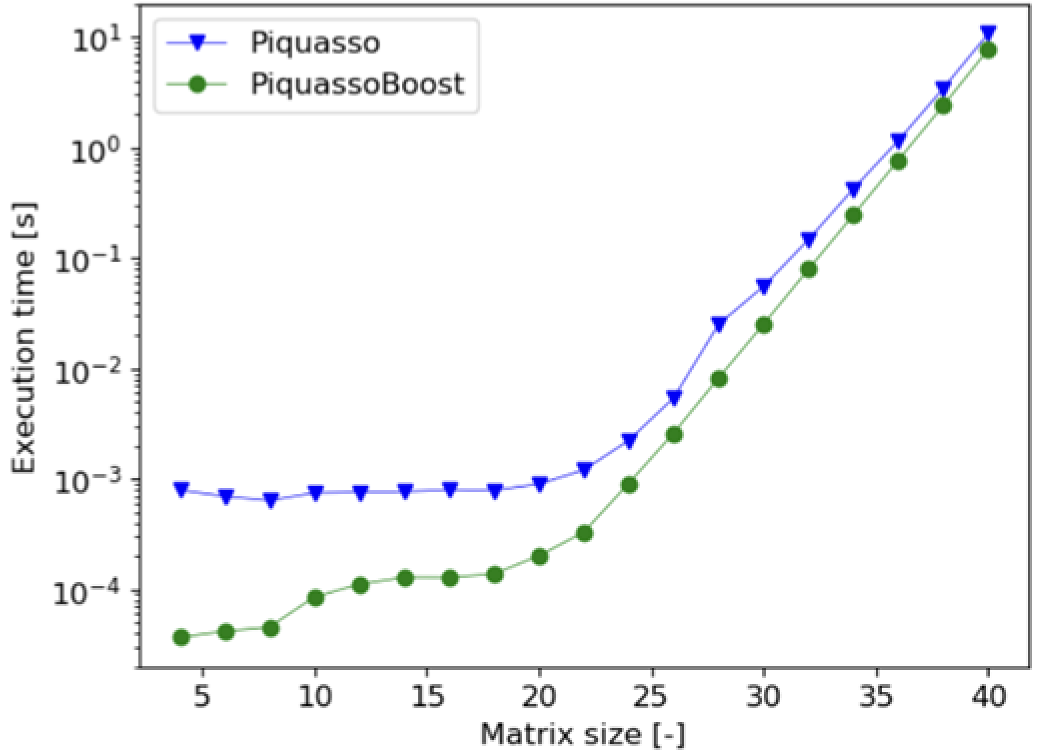
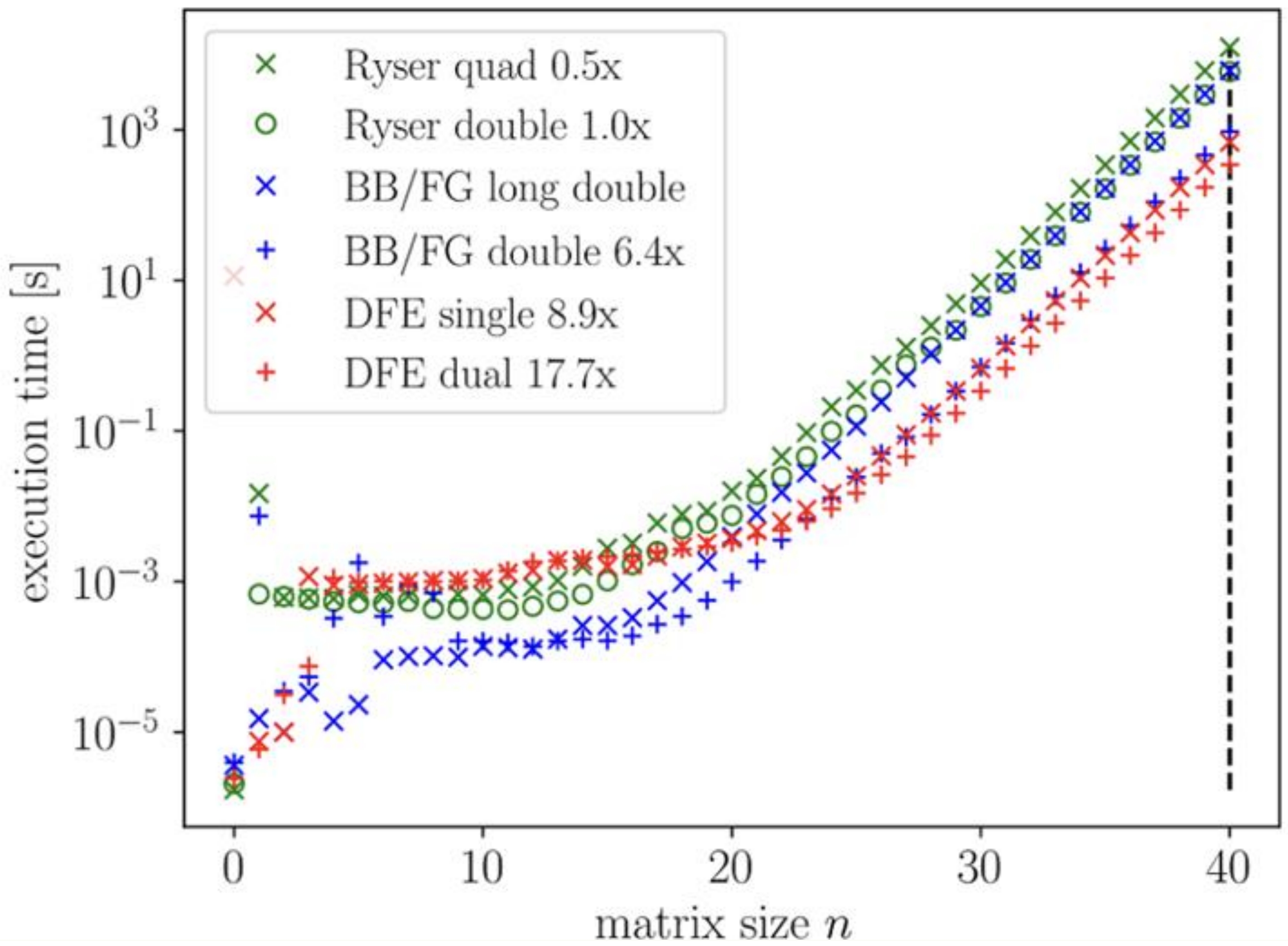
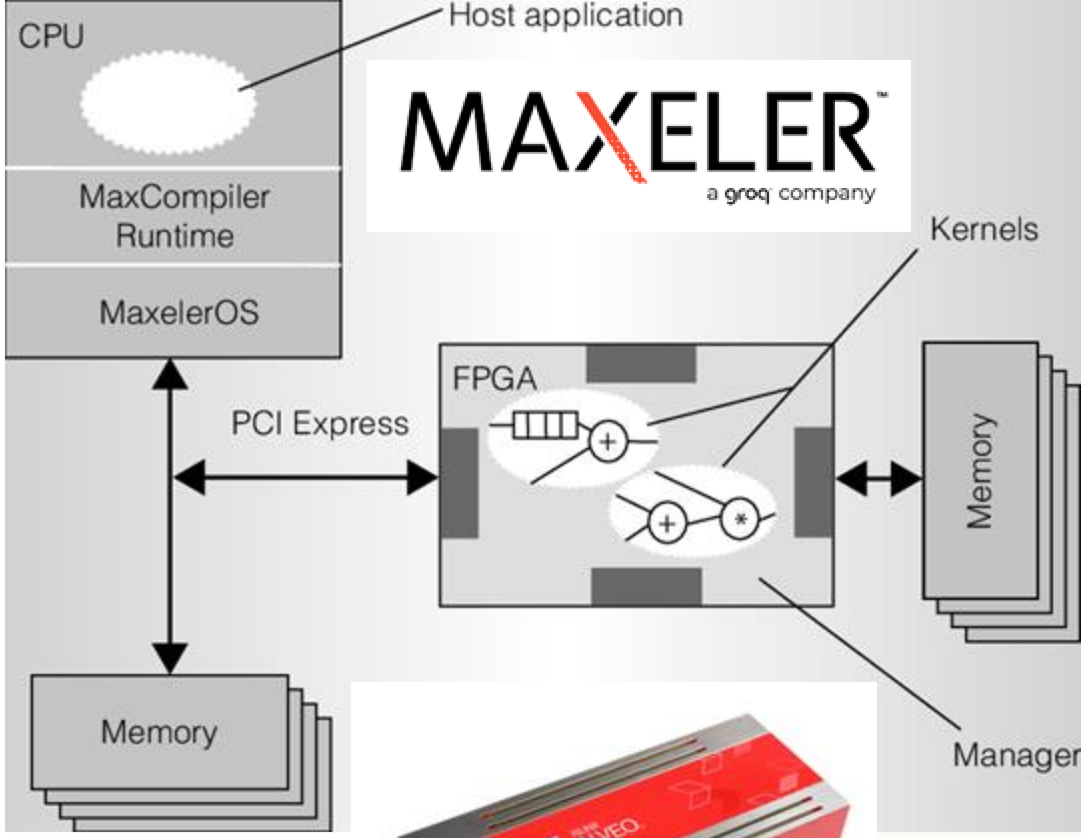


Figure 9: The average repeated permanent execution times of Piquasso (version 5.0.0) and PiquassoBoost with increasing input matrix sizes. The repetitions were

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY
PROGRAM FINANCED FROM THE NRDI FUND





Data-flow engine based acceleration

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE
HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

Motivation: delegated quantum computing

- Client: quantum device with limited power, no large quantum processor
- Server: powerful quantum computer, but not trusted

Security goals

- Privacy (blindness): server learns nothing about input, algorithm, output
- Integrity (verifiability): client can be confident the answer is correct

Hardware security module for quantum cloud computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE
HUNGARY

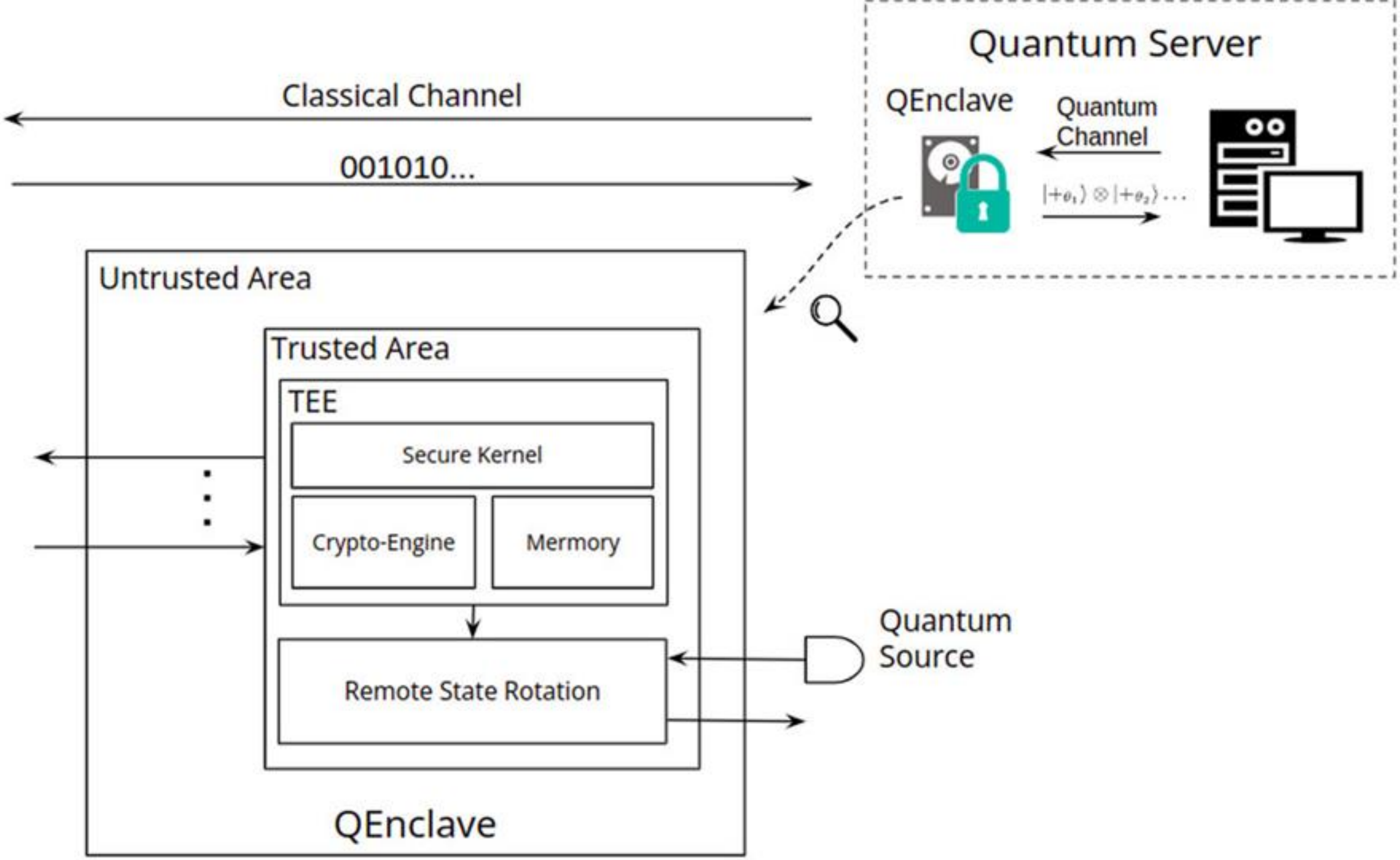
PROGRAM FINANCED FROM THE NRDI FUND

- Universal Blind Quantum Computation enables secure quantum computation by hiding client's classical input from the server that is encoding quantum computation

Broadbent, A., Fitzsimons, J., & Kashefi, E. (2009)
In: 50^{th} Annual IEEE Symp. Foundations of Computer Science

- Server performs computation but learns nothing useful about the input, algorithm, or output
- Originally designed for the qubit model, based on measurement-based quantum computation (MBQC) framework

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY
PROGRAM FINANCED FROM THE NRDI FUND



- Trusted Execution Environment
- HSM + UBQC
- 1-qubit rotations
- Classical client
- Untrusted server & qubit sources

Hardware security module for quantum cloud computing

National Research, Development and Innovation Office
Hungary
PROGRAM FINANCED FROM THE NRDI FUND

HSM-QCC: project by Quantera to produce a **hardware security module** for to the advancement of secure **quantum cloud computing**

- Quantum HSM Design and Security Requirements
- Randomisation Experiment
- Software Integration with Simulated HSM
- Proof of Concept with Quantum Cloud Computer

Website: https://hsm-qcc.eu/

Hardware security module for quantum cloud computing

National Research, Development and Innovation Office
Hungary
PROGRAM FINANCED FROM THE NRDI FUND

UNIVERSITY OF TWENTE.	Coordination, experimental developments
THE UNIVERSITY of EDINBURGH	Theoretical research, communication
SORBONNE UNIVERSITÉ	Quantum cryptography analysis
ELTE EÖTVÖS LORÁND UNIVERSITY	Quantum software dev. and simulation
THALES	HSM development
VeriQloud	Development of software components
QUIX QUANTUM	Provide photonic quantum processor and cloud services

Hardware security module for quantum cloud computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

Photonic qubits: dual-rail encoding

UBQC is often abstractly described with qubit operations, but it can be transferred also to photonic hardware:

- Dual-rail encoding to create photonic qubits (as in the KLM scheme)

Knill, E., Laflamme, R., & Milburn, G. J. (2001). A scheme for efficient quantum computation with linear optics. Nature, 409(6816)

- Using phaseshifters and beamsplitters to implement single-qubit gates and a probabilistic CZ gate (involving postselection)
- Measuring in the relevant basis using particle number measurements

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

Improvements to Piquasso for UBQC simulations

PIQUASSO

- Mid-circuit Measurements: Piquasso program can contain multiple measurements even in the middle of the circuit
- Conditional operations: condition instructions in a Piquasso program on boolean expressions that contain measurement outcomes
- Dual-rail encoding: users can now convert UBQC-like Qiskit QuantumCircuit objects to Piquasso (uses dual-rail encoding during conversion)

National Research, Development and Innovation Office Hungary
PROGRAM FINANCED FROM THE NRDI FUND

```python
with pq.Program() as program1:
    pq.Q() | pq.Program(instructions=prepare_plus_states())
    pq.Q(0) | pq.Program(instructions=hiding_with_theta(theta1))
    pq.Q(2) | pq.Program(instructions=hiding_with_theta(theta2))
    pq.Q(0, 2, 4, 5) | pq.Program(instructions=cz_on_two_bosonic_qubits())
    pq.Q() | post_select_for_cz(4, 5)

with pq.Program() as program2:
    pq.Program(instructions=rotate_into_ubqc_basis(theta1, theta2))
    pq.Q() | pq.ParticleNumberMeasurement()
```

```python
with pq.Program() as program:
    pq.Q() | pq.Program(instructions=prepare_plus_states())
    pq.Q(0) | pq.Program(instructions=hiding_with_theta(theta1))
    pq.Q(2) | pq.Program(instructions=hiding_with_theta(theta2))
    pq.Q(0, 2, 4, 5) | pq.Program(instructions=cz_on_two_bosonic_qubits())
    pq.Q() | post_select_for_cz(4, 5)
    pq.Q(0, 1, 2, 3) | pq.Program(
        instructions=rotate_into_ubqc_basis(theta1, theta2)
    )
    pq.Q() | pq.ParticleNumberMeasurement()
```

Mid-circuit measurements

Before: programs need to be split into two: post-selection and PNM in one circuit

Now: one program can contain multiple measurements even in the middle of the circuit

Hardware security module for quantum cloud computing

NATIONAL RESEARCH, DEVELOPMENT AND INNOVATION OFFICE HUNGARY

PROGRAM FINANCED FROM THE NRDI FUND

Conditional operations

```
pq.Program(
    instructions=[
        pq.StateVector([0, 2]) * np.sqrt(1 / 2),
        pq.StateVector([2, 0]) * np.sqrt(1 / 2),
        pq.ParticleNumberMeasurement().on_modes(1),
        pq.Squeezing(r=r).on_modes(0).when(lambda x: x[-1] == 2),
    ]
)
```

Condition instructions in a Piquasso program on boolean expressions containing measurement outcomes

Hardware security module for quantum cloud computing

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

```python
qc = qiskit.QuantumCircuit(2, 2)
qc.h(0)
qc.h(1)
qc.cz(0, 1)
qc.measure([0, 1], [0, 1])

prog = pq.dual_rail_encode_from_qiskit(qc)
```

Dual-rail encoding: Qiskit converter

We can now convert UBQC-like Qiskit QuantumCircuit objects to Piquasso (dual-rail encoding during conversion)

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

Dedicated quantum circuit repository addressing three problems

1 **No Standard for Sharing**

When publishing complex circuits, there's no standardized way to embed implementations in papers or supplements. Readers are left with incomplete descriptions.

2 **Incomplete Descriptions**

Papers show high-level concepts but omit gate parameters, dependencies, and implementation details necessary for reproduction.

3 **Dead Links**

References break. Repositories disappear. Two years later, the code is gone and reproducibility becomes impossible.

Gate Model

Qubit · Photonic · Continuous variable · Annealing · Quantum inspired

Libraries

qiskit · Piquasso · StrawberryFields · pyQuil · Q# · Cirq · Pennylane · Silq · Perceval

Tags

QFT · fourier transform · CV · grover · undirected · variational · directed-graph · topology · graph similarity · subgraph

https://qcrepository.org/

National Research, Development and Innovation Office Hungary

PROGRAM FINANCED FROM THE NRDI FUND

qcr™

Quantum Circuit Repository

An open-access platform for sharing, discovering, and reproducing quantum circuits

Open Access · Non-Profit Association · Community-Driven

Join Beta Waitlist · Access Beta

www.inf.elte.hu

National Research, Development and Innovation Office
Hungary

PROGRAM
FINANCED FROM
THE NRDI FUND