# The NEASQC Benchmark Suite (TNBS)

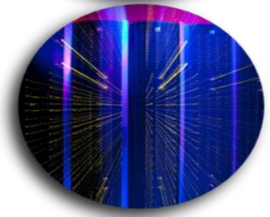
Gonzalo Ferro (CESGA)

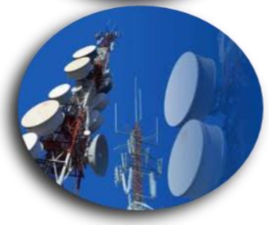
DECEMBER, 15-2023

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951821

# Galicia Supercomputing Centre (CESGA)

- CESGA is a Public Foundation and is member of the Spanish Research Infrastructure RES (Spanish Supercomputing Network).
- CESGA runs its own Quantum Computer (32 qbits) and participates in the major QC Spanish Initiative (Quantum Spain).





## Mission

**Contribute** to the advancement of Science and Technical Knowledge, by means of **research and application of high-performance computing and communications**, as well as other information technologies resources, **in collaboration** with other institutions, **for the profit of society**.

## Services

**Adapted** to technological evolution and **the needs of researchers and users of any area of knowledge** or productive sector.

# NEASQC Project

**NEASQC**: **N**ext **A**pplication**S** of **Q**uantum **C**omputing

> The NEASQC project brings together academic experts and industrial end-users to develop new Quantum-enabled applications that can take advantage of NISQ systems, in the near future, for solving industrial problems.

**Objectives**

> Develop 9 industrial use cases where practical quantum advantage is expected in NISQ machines.

> Develop open-source libraries from industrial uses cases.

> Build a user community dedicated to industrial NISQ applications.

> Develop software stacks and **benchmarks** for the Quantum Technology Flagship hardware platforms.

# The NEASQC Benchmark Suite (TNBS)

The NEASQC Benchmark suite (TNBS) gathers in one suite all the developed benchmarks in the different use cases of NEASQC project.

## OBJECTIVES

- **Objective 1**: The suite must help computer architects, programmers and researchers to design future quantum computers, taking into account the variability in their performance introduced by the different components of the **stack**.
- **Objective 2**: The suite must help to guide the **evolution** of quantum computers (more qubits, better topologies) or to **improve** the current platforms (reduction of noise, better compilers, etc.)

# The NEASQC Benchmark Suite (TNBS)

## OBJECTIVES

- **Objective 3**: The suite must allow a **comparison** of the performance of different platforms. Currently, there are many different proposals for a physical implementation that use different one- and two-qubit gates. The suite must allow users and researchers to compare the performance of different platforms and find their **bottlenecks**. For users, it should allow them to find the best platforms for their applications.
- **Objective 4**: The suite must consider **metrics** important to understand the quantum computing advantage (such as energy consumption, throughput or scalability)

# The NEASQC Benchmark Suite (TNBS)

## OBJECTIVES SUMMARY

- Characterize Quantum Architectures (**QA**) using a set of **heterogeneous benchmarks** based on **applications.**
- **Benchmarks** should be independent of the architecture.
- Not interested in Quantum Supremacy.
- Evaluation of all the stack: compilers, linkers, quantum technologies, etc.

# The NEASQC Benchmark Suite (TNBS)

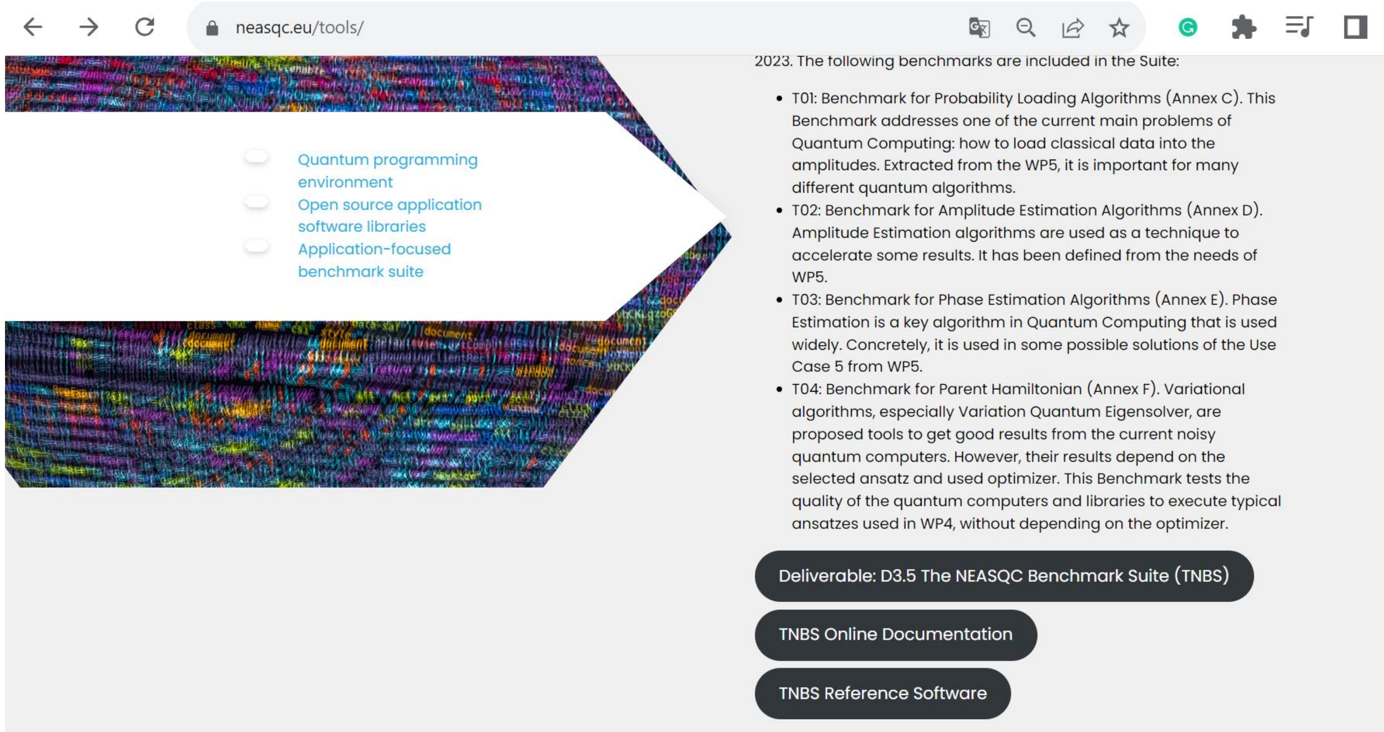
Is a Quantum Platform suitable for your problem or you application?

# TNBS Methodology

- Each benchmark of the suite **must be** documented following the TNBS guidelines.
  - Public project deliverable: **D3.5 The NEASQC benchmark suite (TNBS)** (https://www.neasqc.eu/tools/)
- The benchmarks should be extracted from the use cases of the NEASQC project.
  - Planning release this restriction in the future.
- Each benchmark should have a code implementation using Eviden myQLM.
  - Reference implementation: https://github.com/NEASQC/WP3_Benchmark
- A web platform where every user of the suite can submit their own results to cross-compare different platforms.

# TNBS Methodology

> Each benchmark of the suite **must be** documented following the TNBS guidelines.

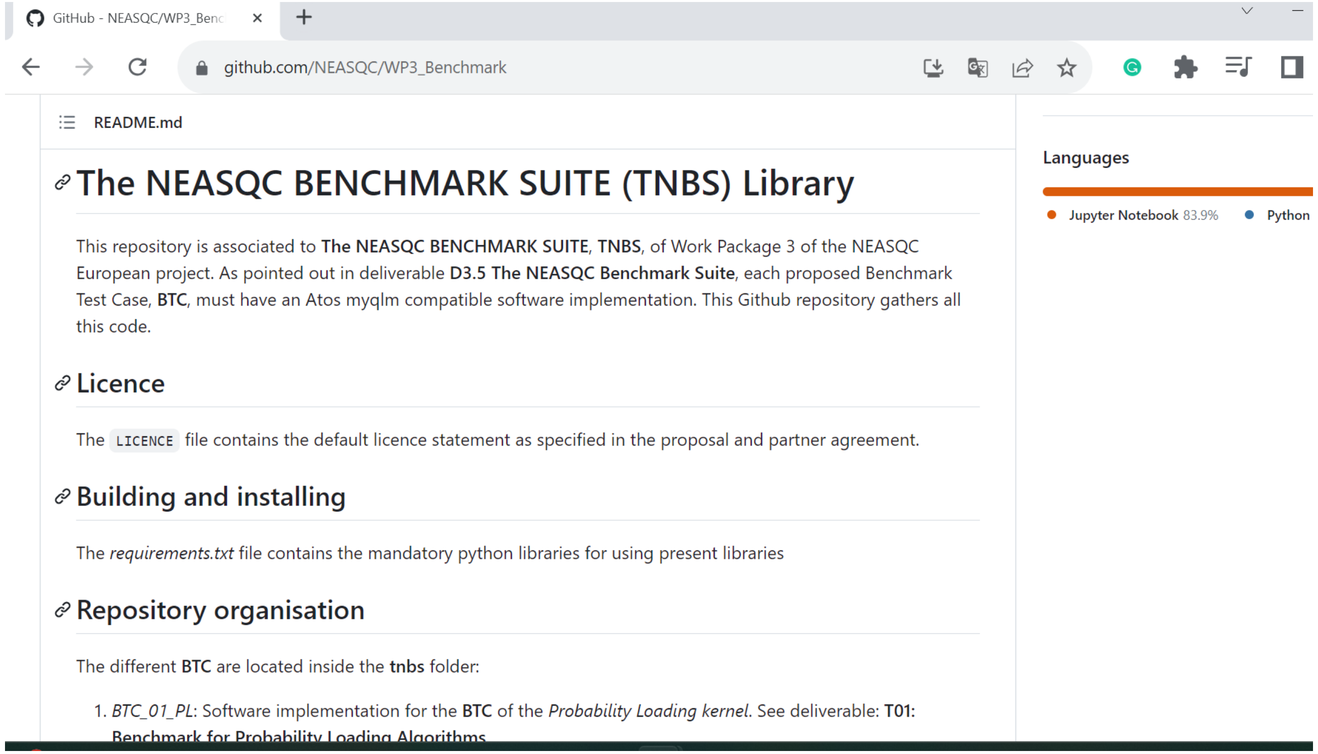


# TNBS Methodology

- Each benchmark of the suite **must be** documented following the TNBS guidelines.
  - Public project deliverable: **D3.5 The NEASQC benchmark suite (TNBS)** (https://www.neasqc.eu/tools/)
- The benchmarks should be extracted from the use cases of the NEASQC project.
  - Planning release this restriction in the future.
- Each benchmark should have a code implementation using Eviden myQLM.
  - Reference implementation: https://github.com/NEASQC/WP3_Benchmark
- A web platform where every user of the suite can submit their own results to cross-compare different platforms.

# TNBS Methodology

> Each benchmark should have a code implementation using Eviden myQLM.

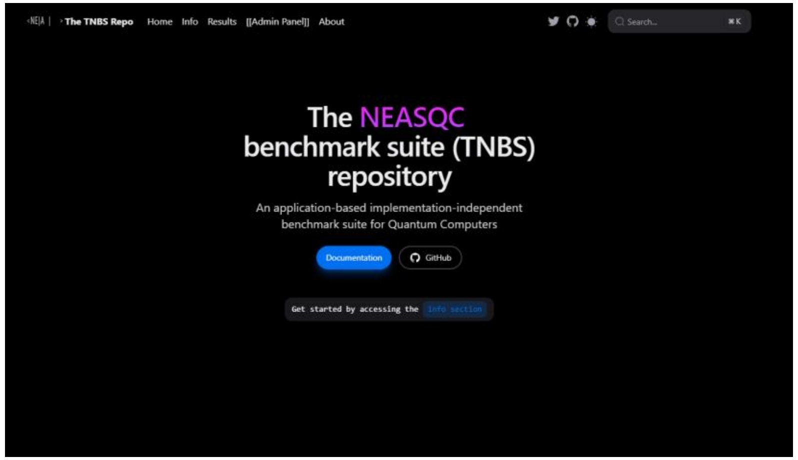


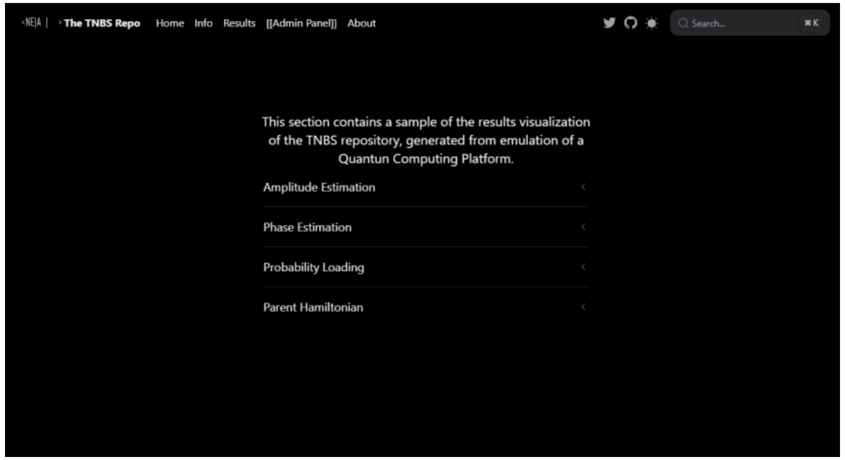
# TNBS Methodology

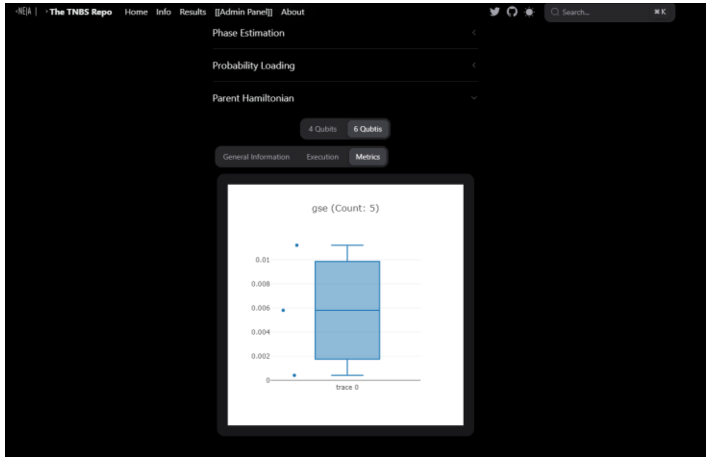
- Each benchmark of the suite **must be** documented following the TNBS guidelines.
  - Public project deliverable: **D3.5 The NEASQC benchmark suite (TNBS)** (https://www.neasqc.eu/tools/)
- The benchmarks should be extracted from the use cases of the NEASQC project.
  - Planning release this restriction in the future.
- Each benchmark should have a code implementation using Eviden myQLM.
  - Reference implementation: https://github.com/NEASQC/WP3_Benchmark
- A web platform where every user of the suite can submit their own results to cross-compare different platforms.

# TNBS Methodology

> A web platform where every user of the suite can submit their own results to cross-compare results from different platforms.



# TNBS Terminology

Each **benchmark** of the suite should be composed of:

- **Kernel**: core quantum subroutine high-level defined (mathematically or procedurally). Common to several algorithms. It could be implemented using different procedures.
- **Benchmark Test Case (BTC)**: problem whose solution involves the execution of the **Kernel**. Output verifiable analytically or through a classical simulation. Used for evaluating the performance of a Quantum Architecture.
- **Metric**: Parameter for evaluating the performance of a **QA** for solving the **BTC** (so it depends on it).
  - **Verification**: *Is the provided result by the QA correct?*
  - **Performance:** How fast *is the execution of the BTC in the QA?*

# TNBS Current microbenchmarks

- T01: Probability Loading microbenchmark.
- T02: Amplitude Estimation microbenchmark.
- T03: Quantum Phase Estimation microbenchmark.
- T04: Parent Hamiltonian microbenchmark.

# T01: Probability Loading microbenchmark

**Kernel:** Probability Loading. Build a unitary operator for loading a discrete probability distribution $P$ into the amplitude of a quantum state.

$$\mathbf{P} = \{p_0, p_1, \cdot, p_{2^n-1}\}, p_i \in [0, 1]$$

$$\sum_{i=0}^{2^n-1} |p_i|^2 = 1$$

$$\mathbf{U}|0\rangle_n = \sum_{i=0}^{2^n-1} \sqrt{p_i}|i\rangle_n$$

**BTC:** Load a known Gaussian probability distribution function

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

**Metrics (verification):** compare the measured (Q) and the theoretical distribution (P) using:

Kolgorov-Smirnov:

$$KS = \max\left(\left|\sum_{j=0}^{i} P(x_j) - \sum_{j=0}^{i} Q_j\right|, \forall i = 0, 1, \cdots, 2^n - 1\right)$$

Kullback-Leibler divergence :

$$KL(\mathbf{Q}/\mathbf{P}) = P(x_j) \ln \frac{P(x_j)}{\max(\epsilon, Q_k)}$$

**Applications:** QPCA, HHL, Quantum Amplitude Estimation…

# T02: Amplitude Estimation microbenchmark

**Kernel:** Amplitude Estimation. Estimate $a$ given a unitary operator $\hat{A}$

$$|\Psi\rangle = \hat{A}\,|0\rangle_n = \sqrt{a}\,|\Psi_0\rangle + \sqrt{1-a}\,|\Psi_1\rangle$$

**BTC:** Estimating the integral of a sine function encoded in $a$

$$F = \int_a^b \sin(x)dx = \cos(a) - \cos(b)$$

**Applications:** Finance, Chemistry, numeric integration.

**Metrics (verification):** Absolute error between Riemann sum of the integral and the estimation

$$\epsilon = \left| S^I_{[a,b]} - S_{[a,b]} \right| \qquad S^I_{[a,b]} = \frac{b-a}{2^n} \sum_{i=0}^{2^n-1} \sin(x_i) \sim F$$

**Several algorithms for AE can be used:**

- Quantum Phase Estimation
- Maximum Likelihood AE
- Iterative Quantum AE
- Real Quantum AE

# T03: Quantum Phase Estimation microbenchmark

**Kernel:** Quantum Phase Estimation. Given a unitary operator U estimate their eigenvalues

$$U|\psi_j\rangle = e^{2i\pi\lambda_j}|\psi_j\rangle$$

**BTC:** Estimate the eigenvalues of a $R_z^{\otimes n}(\theta)$

$$R_z(\theta) = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} = e^{-i\frac{\theta}{2}}|0\rangle\langle 0| + e^{i\frac{\theta}{2}}|1\rangle\langle 1|$$

2 different angle selection:

- Random
- Exact: $\theta_{i+1} = \theta_i + a\delta\theta.\ a = \{-1, 1\}.\ \delta\theta = \frac{4\pi}{2^m}$

**Metrics (verification):** Compare the theoretical ($P^{th}_{\lambda,m}$) and the obtained ($P^{QPE}_{\lambda,m}$) probability density of eigenvalues:

Random case: $$KS = \max\left(\left|\sum_{k=0}^{i} P^{th}_{\lambda,m}(\frac{k}{2^m}) - \sum_{k=0}^{i} P^{QPE}_{\lambda,m}(\frac{k}{2^m})\right|\right)$$

Exact case: $$fidelity = \frac{\sum_{k=0}^{n-1} P^{th}_{\lambda,m}(\frac{k}{2^m}) * P^{QPE}_{\lambda,m}(\frac{k}{2^m})}{|P^{th}_{\lambda}| * |P^{QPE}_{\lambda}|}$$

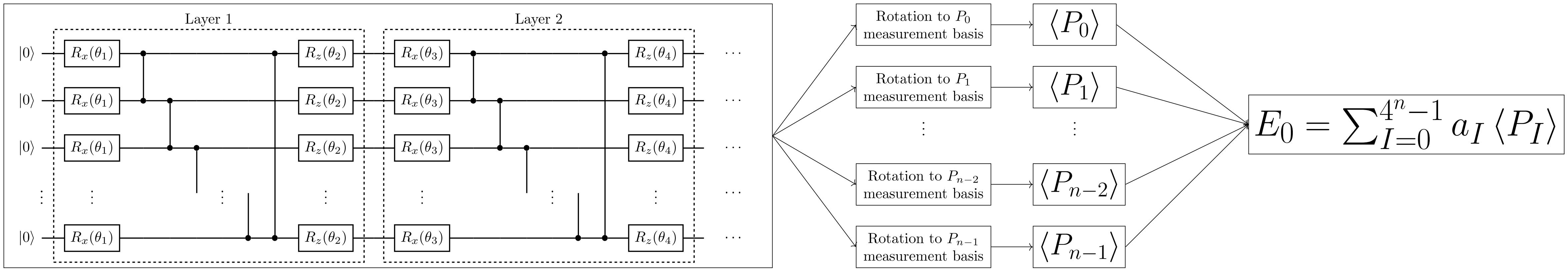
**Applications:** Shor's algorithm, quantum linear equation solver, VQE…

# T04: Parent Hamiltonian microbenchmark

**Kernel:** Parent Hamiltonian. Given an ansatz with state $|\Psi\,(\theta)\rangle$ its parent Hamiltonian, $H^{PH}$, is such that:

$$H^{PH}|\Psi\,(\theta)\rangle = E_0|\Psi\,(\theta)\rangle \;\; E_0 = 0$$

**BTC:** following Ansatz for different number of layers: $|\Psi\,(n, d, \theta_i)\rangle \rightarrow H^{PH} \rightarrow H^{PH} = \sum_{I=0}^{4^n-1} a_I P_I$




$$\theta_i = (i+1)\delta\theta \;\; i = 0, 1, \cdots 2n_l - 1 \text{ where } \delta\theta = \frac{\pi}{4 * (n_l + 1)}$$

# T04: Parent Hamiltonian microbenchmark

**Metrics (verification):** The computed ground state energy $E_0$should be compared with theoretical one: 0. So the metric is $E_0$.

**Applications:** Variational Quantum Algorithm (particularly VQE).

- The GitHub provided the Pauli decomposition from 3 to 30 qubits and from 1 to 4 layers (for each qubit) ansatzes.
- Additionally, a code for building the parent Hamiltonian and its Pauli decomposition for any ansatz is provided (using myQLM library).
- Implementation of BTC Parent Hamiltonian using MPS.

# Conclusions

- Developed  a benchmark suite for assessing quantum devices based on use cases of the NEASQC project.
- TNBS should help to improve the complete stack of a quantum hardware device.
- Microbenchmarks are composed of **Kernel** and a **Benchmark Test Case**.
- Four microbenchmarks have been developed:
  - Probability Loading
  - Amplitude Estimation
  - Quantum Phase Estimation
  - Parent Hamiltonian

# Thank you for your attention!

neasqc.eu

@neasqc

CESGA. SPAIN.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951821