

---

# IBM QisKit Performance & Quality Leadership

---

STEVE MCDOWELL, CHIEF ANALYST  
9/30/24

## CONTEXT

---

IBM [recently announced](#) that its Qiskit emerged as the highest-performing SDK in terms of both speed and quality in recent benchmarking tests to evaluate quantum software development kits. These tests measured how efficiently SDKs could build, manipulate, synthesize, and transpile quantum circuits. Qiskit passed more tests than any other SDK and demonstrated superior performance metrics, outperforming competitors like TKET.

## BACKGROUND: WHAT IS QISKIT?

---

Qiskit is an open-source quantum computing SDK developed by IBM Research, but isn't tied to IBM quantum computers. Qiskit allows developers and researchers to create, simulate, and run quantum algorithms on both simulators and IBM Quantum's actual quantum processors.

It supports a wide range of quantum applications, from research and education to industrial-scale quantum computing. Its recent advancements in speed and quality make it one of the most potent and efficient SDKs for quantum computing.

## NEW: INDUSTRY-LEADING PERFORMANCE & QUALITY

---

Recent benchmarking tests have confirmed that Qiskit, IBM's quantum SDK, outperforms its competitors in speed and quality, making it the top choice for quantum developers.

The performance and quality of a quantum SDK are essential for running quantum algorithms efficiently, as they directly affect how circuits are built, optimized, and executed on quantum hardware.

---

## KEY FINDINGS FROM THE BENCHMARKING TESTS

---

- **Speed:** Qiskit was found to be 13 times faster at transpiling quantum circuits than its closest competitor, TKET. Speed in transpilation is critical because it determines how quickly a quantum circuit can be mapped to the physical qubits on quantum hardware. Faster transpilation means reduced wait times and more efficient research and experimentation.
- **Quality:** Qiskit also produced circuits with 24% fewer two-qubit gates than TKET. Reducing the number of gates is crucial in quantum computing, as fewer gates mean lower error rates and more stable computations. This is particularly important for large and complex circuits prone to errors on current quantum hardware.

---

## KEY IMPROVEMENTS DRIVING QISKIT'S PERFORMANCE

---

### 1. **LightSABRE Algorithm:**

- Qiskit's performance leap is largely due to the introduction of the LightSABRE algorithm, an upgraded version of the widely used SABRE algorithm. This new algorithm reduces the use of SWAP gates by 18.9%, essential for minimizing circuit depth and improving overall execution.
- LightSABRE ensures more efficient routing and mapping of quantum circuits onto hardware, significantly lowering resource overhead, which is especially valuable for large-scale circuits.

### 2. **AI-Powered Transpiler:**

- Qiskit's AI-powered transpiler significantly boosts circuit optimization. For larger circuits, particularly those with over 100 qubits, the AI-enhanced transpiler reduces the circuit depth by 36% and the number of two-qubit gates by 24%.
- This AI-driven optimization leads to higher-quality circuits that are less error-prone, making it ideal for developers tackling complex quantum problems or working on quantum hardware with constrained resources.

### 3. **Rust Integration:**

- A substantial portion of Qiskit's codebase has been rewritten in Rust, a programming language known for its performance and memory safety. Rust's efficiency has made Qiskit faster and more

scalable, ensuring that it can handle larger circuits and more demanding quantum computing tasks without compromising on speed or reliability.

## NEW: LIGHTSABRE ALGORITHM

---

The LightSABRE algorithm is IBM Research's new advanced quantum circuit transpilation technique, optimized for mapping quantum circuits onto real hardware. It builds upon the original SABRE (Swap-based Adaptive Quantum Routing) algorithm, designed to efficiently route quantum circuits while minimizing the number of SWAP gates required.

SWAP gates are used to move quantum information between qubits that are not directly connected on a quantum chip. Still, they introduce additional complexity and potential for errors, making their reduction a key goal in optimization.

The LightSABRE algorithm arrives with several key features:

- **Reduction of SWAP Gates:** LightSABRE further reduces the number of costly SWAP gates in quantum circuits. This reduction is essential because fewer SWAP gates lead to shorter, less error-prone circuits, which improves the overall fidelity and execution speed of quantum computations.
- **Enhanced Scalability:** As quantum processors increase in size and complexity, the LightSABRE algorithm scales efficiently, allowing it to handle large quantum circuits involving hundreds of qubits. This scalability is crucial for running utility-scale quantum computations on future quantum hardware.
- **Improved Circuit Depth and Gate Optimization:** LightSABRE focuses on minimizing the number of SWAP gates and the circuit depth—the number of layers of quantum gates applied. A lower circuit depth is important because it reduces the time a quantum computation takes, limiting the impact of noise and errors, especially on today's noisy intermediate-scale quantum (NISQ) devices.

## NEW: AI-POWERED TRANSPILER

---

IBM's new AI-powered transpiler reduces the complexity of quantum circuit transpilation to enhance the performance of quantum algorithms on noisy

intermediate-scale quantum (NISQ) devices. The new transpiler delivers better circuit quality, improved error suppression, and shorter execution times.

## KEY FEATURES

---

Several key features define the transpiler:

- **Enhanced Circuit Optimization:** The transpiler uses machine learning algorithms to optimize the arrangement and execution of quantum gates, resulting in fewer two-qubit gates and a significant reduction in overall circuit depth – critical factors for improving the fidelity of quantum computations.
- **Error Suppression and Mitigation:** AI analyzes and predicts error patterns based on the hardware's characteristics. The transpiler can apply sophisticated error suppression techniques, minimizing the noise introduced during quantum gate operations, particularly on today's noisy quantum hardware.
- **Automated Adaptation to Hardware:** The AI-powered transpiler dynamically adapts to different quantum hardware configurations. It can automatically optimize circuits based on the target quantum processor's specific **connectivity** and **noise characteristics**, ensuring the best possible performance for each unique setup.
- **Scalability for Larger Circuits:** Circuit optimization becomes increasingly complex as quantum hardware scales to accommodate more qubits. The AI-powered transpiler handles large, utility-scale circuits, enabling efficient execution of quantum algorithms that involve hundreds or thousands of qubits.

## PERFORMANCE IMPROVEMENTS

---

- **Circuit Depth Reduction:** The benchmarks show that the AI-powered transpiler reduced circuit depth by an average of **36%**, allowing quantum algorithms to run faster and with fewer errors.
- **Gate Optimization:** The transpiler also decreased the number of two-qubit gates by an average of **24%**, which is particularly important for improving circuit stability on NISQ devices. Fewer gates result in fewer opportunities for errors, leading to higher-quality results.
- **Faster Execution:** By optimizing circuits more efficiently, the AI-powered transpiler can significantly reduce transpilation time, allowing developers to iterate faster and perform more experiments in less time.

---

## NEW: TRANSITION TO RUST

---

The transition of Qiskit from Python to Rust is a strategic technical move that dramatically enhances performance. Unlike Python, which is interpreted, Rust is compiled, leading to faster, more efficient code execution. For quantum developers, this shift means that building and optimizing quantum circuits can now be done in hours rather than weeks or months, significantly reducing development cycles.

---

## IMPLICATIONS FOR THE DEVELOPER COMMUNITY

---

Qiskit's advancements in speed and quality directly benefit the quantum developer community, offering powerful tools to create optimized circuits for real-world quantum hardware. These improvements come at a critical time as quantum computers scale up in qubit capacity and complexity. Here's why it matters:

- **Enhanced Developer Productivity:**
  - Developers can now transpile and optimize circuits more quickly, enabling them to run multiple experiments and reduce development bottlenecks.
- **Optimization for Large-Scale Quantum Projects:**
  - With Qiskit's AI-powered transpiler and circuit quality enhancements, developers can tackle more complex quantum problems, paving the way for breakthroughs in fields like quantum chemistry, machine learning, and optimization.
- **Open-Source Tools and Collaboration:**
  - IBM has also made its benchmarking tool, **Benchpress**, open-source, encouraging the quantum computing community to run their own tests and contribute to improving Qiskit's performance. This transparency fosters collaboration and competition, driving further innovation in the field.

---

## NEW: QISKIT FUNCTIONS CATALOG

---

IBM launched the Qiskit Functions Catalog to help simplify the development of quantum algorithms and applications. By abstracting the complexities of quantum hardware and software, the catalog provides pre-built functions that

allow developers and researchers to more easily leverage quantum computing for real-world use cases.

The catalog offers ready-to-use functions, removing the need for detailed knowledge of quantum circuits and error mitigation techniques.

It includes two categories of functions:

- **Circuit Functions:** Simplify tasks such as circuit transpilation, optimization, error suppression, and mitigation. These functions allow quantum computational scientists to focus on mapping problems to quantum circuits without having to construct circuits manually.
- **Application Functions:** Cater to higher-level, domain-specific tasks. Enterprise developers and data scientists without deep quantum expertise can integrate quantum computing into their workflows by inputting classical data and receiving quantum-optimized outputs.

In addition to IBM's functions, the catalog includes contributions from partners such as Q-CTRL, Algorithmiq, and QunaSys, offering solutions for error suppression, quantum optimization, and domain-specific problems like chemistry ground-state calculations.

IBM's Circuit Functions incorporate AI-driven features for circuit synthesis, optimization, and scheduling, enabling developers to perform better on IBM's quantum hardware.

The catalog is currently available in preview to Premium Plan users. They can explore functions built by IBM and its partners and purchase licenses for specific functions that align with their needs.

---

## ANALYSIS

---

Quantum computing research heavily relies on SDKs' performance to efficiently build and run circuits. The ability to transpile circuits quickly and with fewer gates can significantly shorten the time required for quantum experiments. Qiskit's advancements in speed and quality make it the go-to SDK for researchers, offering a more seamless experience from code development to circuit execution on real hardware.

Qiskit's continued focus on performance optimization, through innovations like the LightSABRE algorithm and AI-driven transpilation, positions it as the leading SDK for researchers and developers focused on quantum computing at scale. Combining SDK improvements with robust hardware metrics, IBM's

comprehensive approach creates a compelling proposition for those working in quantum research and enterprise applications.

One of the most significant elements of IBM's updates to Qiskit is the shift from Python to Rust, which gives Qiskit a competitive edge over other quantum SDKs that rely on slower, interpreted languages.

The company's quantum roadmap, which includes solving previously unsolvable problems in areas like artificial intelligence, chemistry, and energy, is dependent not just on advanced hardware but also on equally advanced software. Quantum hardware alone cannot deliver practical advantages without the related software tools to build and optimize quantum circuits at scale, making Qiskit a key part of its quantum computing strategy.

As quantum computing moves toward broader commercial applications, the need for high-performance, reliable quantum software solutions like Qiskit will continue to grow.

We expect Qiskit to maintain its leadership position in the quantum SDK space, driven by ongoing performance enhancements and the growing capabilities of IBM's quantum ecosystem. Other quantum SDK providers will need to address their current limitations to remain competitive, particularly in reducing bugs and improving execution time.



© Copyright NAND Research.

NAND Research is a registered trademark of NAND Research LLC, All Rights Reserved.

This document may not be reproduced, distributed, or modified, in physical or electronic form, without the express written consent of NAND Research. Questions about licensing or use of this document should be directed to [info@nand-research.com](mailto:info@nand-research.com).

The information contained within this document was believed by NAND Research to be reliable and is provided for informational purposes only. The content may contain technical inaccuracies, omissions, or typographical errors. This document reflects the opinions of NAND Research, which is subject to change. NAND Research does not warranty or otherwise guarantee the accuracy of the information contained within.

NAND Research is a technology-focused industry analyst firm providing research, customer content, market and competitive intelligence, and custom deliverables to technology vendors, investors, and end-customer IT organizations.

Contact NAND Research via email at [info@nand-research.com](mailto:info@nand-research.com) or visit our website at [nand-research.com](http://nand-research.com).