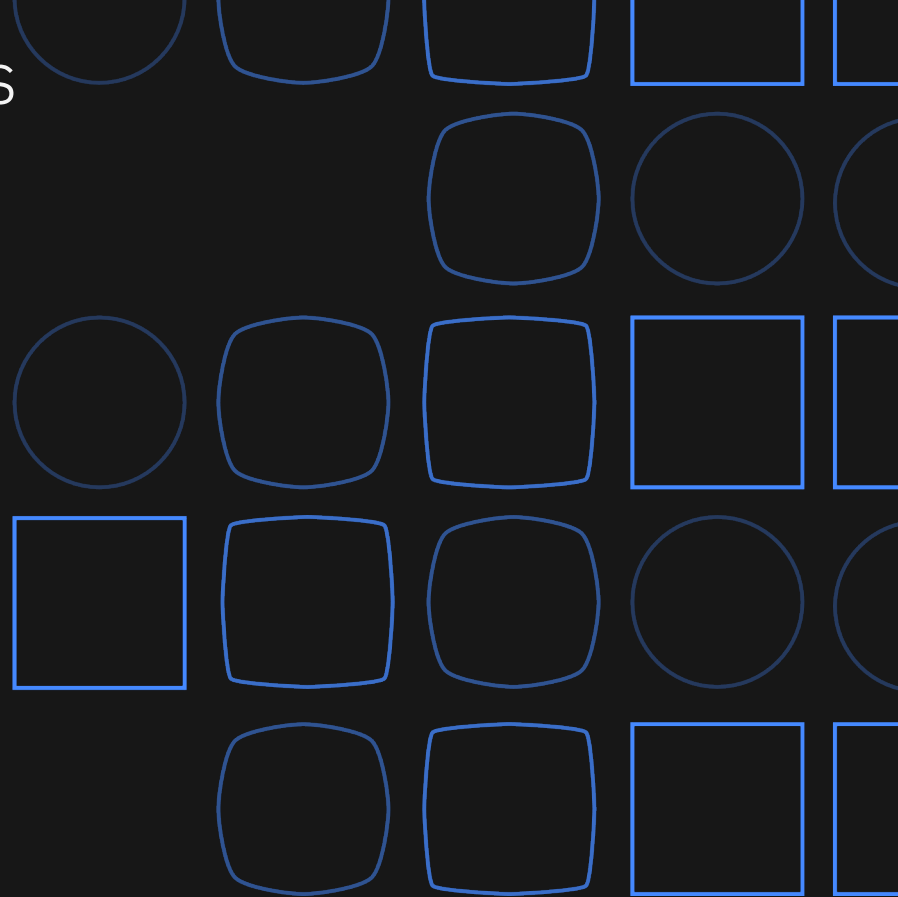


Qiskit Deep Dive: Experiments

Christopher J. Wood

Research Staff Member, IBM Quantum



- New Qiskit package for end-to-end running of experiments on quantum computers and storing of results in an online IBM experiment database.
- Focus on calibration and characterization experiments
- Provides a framework for developers to create their own experiments.
- Custom experiments can also be stored and retrieved from IBM experiment database.

Install

```
pip install qiskit-experiments
```

Documentation

```
qiskit.org/documentation/experiments/
```

Github

```
github.com/Qiskit/qiskit-experiments/
```

- New Qiskit package for end-to-end running of experiments on quantum computers and storing of results in an online IBM experiment database.
- Focus on calibration and characterization experiments
- Provides a framework for developers to create their own experiments.
- Custom experiments can also be stored and retrieved from IBM experiment database.

Qiskit Experiments API Reference 📖

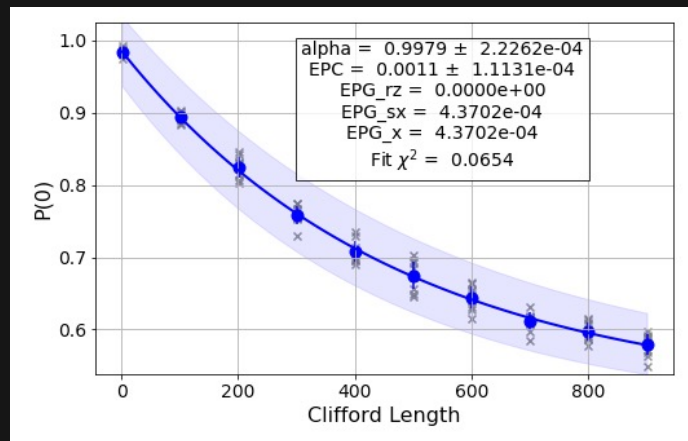
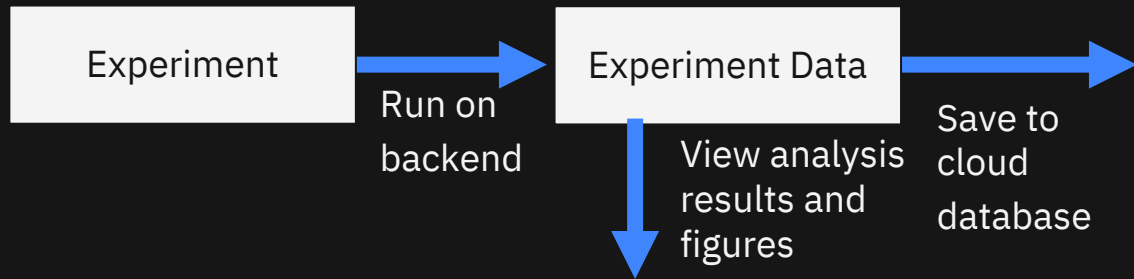
Package Modules 📖

- [Qiskit Experiments](#) (`qiskit_experiments`)
- [Experiment Framework](#) (`qiskit_experiments.framework`)
- [Experiment Library](#) (`qiskit_experiments.library`)
- [Data Processing](#) (`qiskit_experiments.data_processing`)
- [Curve Analysis](#) (`qiskit_experiments.curve_analysis`)
- [Calibration Management](#) (`qiskit_experiments.calibration_management`)
- [Database Service](#) (`qiskit_experiments.database_service`)

Experiment Modules 📖

- [Calibration Experiments](#) (`qiskit_experiments.library.calibration`)
- [Characterization Experiments](#) (`qiskit_experiments.library.characterization`)
- [Randomized Benchmarking Experiments](#)
(`qiskit_experiments.library.randomized_benchmarking`)
- [Tomography Experiments](#) (`qiskit_experiments.library.tomography`)
- [Quantum Volume Experiment](#) (`qiskit_experiments.library.quantum_volume`)

Framework Overview



```
DbAnalysisResultV1
- name: EPG_rz
- value: 0.0
-  $\chi^2$ : 0.06537136051261279
- device_components: ['Q0']
- verified: False
DbAnalysisResultV1
- name: EPG_sx
- value: 0.00043701967376250566
-  $\chi^2$ : 0.06537136051261279
- device_components: ['Q0']
- verified: False
DbAnalysisResultV1
- name: EPG_x
- value: 0.00043701967376250566
-  $\chi^2$ : 0.06537136051261279
- device_components: ['Q0']
- verified: False
```

The screenshot shows the IBM Quantum StandardRB experiment page. The experiment ID is 4adcbbf5-1b75-442b-8ba6-229af46c00ba. The page includes sections for Details, Related jobs (1), Artifacts (0), Metadata, and Plots (1). The Analysis Results section is expanded, showing a table of parameters and their values.

Type	Components	Value	Chisq	Quality	Verified
<input type="checkbox"/> EPG_x	Q0	1.1054e-3	1.6720	--	
<input type="checkbox"/> EPG_sx	Q0	1.1054e-3	1.6720	--	
<input type="checkbox"/> EPG_rz	Q0	0.0000e+0	1.6720	--	
<input type="checkbox"/> EPC	Q0	(2.1305 ± 0.2012) e-3	1.6720	--	
<input type="checkbox"/> alpha	Q0	(9.9574 ± 0.0040) e-1	1.6720	--	
<input type="checkbox"/> @Parameters_RBAnalysis	Q0	--	1.6720	--	

Experiment Class

- Generates and transpiles circuits to run on backend
- Contains an analysis class for processing result data

Analysis Class

- Processes raw result data (counts or IQ)
- Generates analysis results and figures from data

Experiment Data Class

- Stores job information for running circuits
- Stores result data
- Stores analysis results and figures
- Can be saved and loaded to online cloud database

Running Experiments

Let's try it out in a Jupyter Notebook...

For more documentation and tutorials see

<https://qiskit.org/documentation/experiments/>