

WHITEPAPER — QQDF-UL Standard (v1.0)
Open Quantum Data Format — Unlimited Layers

Author: Agustín Rodrigues De Barros
License: Apache 2.0
Status: Draft / Public Release Candidate
Date: 2025-11-08

1. Abstract

Quantum computing lacks a standardized, hardware-neutral method for representing classical datasets as quantum-ready structures. Existing standards—OpenQASM and QIR—define how circuits are executed, but not how data should be encoded.

QQDF-UL (Open Quantum Data Format — Unlimited Layers) introduces a high-level data description layer that specifies amplitudes, phases, layers, and encoding rules in a vendor-agnostic way. It defines what quantum state is intended, while allowing compilers to decide how to implement it.

QQDF-UL simplifies hybrid workflows and prepares classical data for future quantum-native storage formats.

2. Motivation

2.1 Lack of a unified data interface

Quantum libraries encode classical data inconsistently—Qiskit FeatureMaps, PennyLane templates, Cirq ops, Braket builders—creating fragmentation.

2.2 Difficulty scaling to large datasets

Current encodings require manual choices: qubit count, normalization, phase rules, and amplitude-phase interactions.

2.3 Absence of a portable data format

Quantum computing lacks an equivalent to:

JSON (general data)

ONNX (ML models)

Parquet/Arrow (high-volume storage)

2.4 Future quantum storage demands a standard

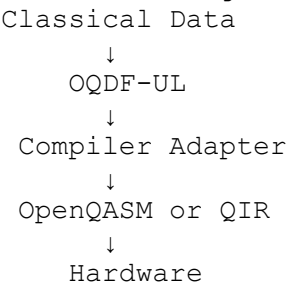
Emerging technologies (quantum RAM, photonic memories, NV center storage) need a format, not ad-hoc pipelines.

QQDF-UL addresses these gaps.

3. Design Goals

- Hardware-neutral (no gate sets, no backend assumptions)
- Unlimited scalability (qubits inferred from dataset size)
- Layered amplitude-phase model
- Interoperability with QASM/QIR
- Declarative, not procedural
- Human-readable, machine-validated (JSON Schema)

4. Positioning Within the Quantum Ecosystem



Comparison Table

Feature	OQDF-UL	OpenQASM	QIR
Purpose	Data Encoding	Circuit Description	Compiler IR
Format	JSON	Language/Text	LLVM IR
Hardware	Very High	Medium	High
Amplitude/Phase Encoding	YES	NO	NO
Qubit Auto-Inference	YES	NO	NO
Gate-Level Details	None	Required	Internal

5. The OQDF-UL Model

5.1 Metadata

Authorship, timestamps, dataset identity, tags, and licensing.

5.2 Classical Source

Path to the dataset (CSV/JSON/Parquet/etc.).

5.3 Quantum Space

Defines:

number of qubits

dimensions

normalization

layer count

5.4 Feature Map

Each classical variable maps to:

amplitude

phase

categorical_phase

layered

custom

Includes ranges, scale factors, phase rules, and categorical mappings.

5.5 Operations

Declarative instructions:

prepare state

attach layer

apply rule

custom ops

These describe intent, not circuits.

6. Encoding Examples

Amplitude

```
{ "name": "temperature", "encoding": "amplitude", "range": [-10, 45] }
```

Phase

```
{ "name": "day_of_week", "encoding": "phase", "phase_factor": 0.15 }
```

Categorical Phase

```
{  
  "name": "weather",  
  "encoding": "categorical_phase",  
}
```

```
"mapping": {  
  "sunny": 0.0,  
  "cloudy": 1.2,  
  "rain": 2.4  
}  
}
```

7. Unlimited Layer System (UL)

Each layer acts as an independent "quantum plane":

Layer 1 → amplitude

Layer 2 → phase

Layer 3 → domain rules

Layer 4 → gradient/frequency maps

Layer N → custom embeddings

This allows future compatibility with:

quantum RAM

photonic frequency multiplexing

multi-layer VQE/QAOA

quantum CNN architectures

8. Validation and Security

Validation uses:

JSON Schema

range checking

layer consistency

encoding-type verification

Security:

no executable code inside QQDF-UL

custom operations are declarative, not scripts

adapters must sandbox user-defined mappings

9. Roadmap to QQDF-UL v2.0

9.1 Compiler Adapters

QQDF-UL → OpenQASM

QQDF-UL → QIR

9.2 Binary Format (QQDF-UL-B)

Compressed version for large-scale datasets (>1M rows).

9.3 Quantum Task Layer

High-level operations, e.g.:

“optimize forecast”

“compute best schedule”

“minimize objective function”

9.4 Quantum-Ready Storage Containers

Similar to Parquet/Arrow but for:

amplitude vectors

phase layers

multi-layered encodings

10. Conclusion

QQDF-UL provides the first open, hardware-neutral, data-centric quantum format. It offers a high-level declarative structure that enables reproducible hybrid quantum pipelines, long-term quantum storage strategies, and cross-organization collaboration.

Version 1.0 establishes the foundation.

Version 2.0 will introduce the execution and compiler bridging layer.

11. Citation

Rodrigues De Barros, A. (2025).

QQDF-UL: Open Quantum Data Format – Unlimited Layers (v1.0).

Public Specification Draft.

https://github.com/<your_repo>