Quantum Error Correction Framework Comparison

Loom (Entropica Labs) vs Deltakit (Riverlane)

October 2025

**Executive Summary**

This document provides a comprehensive comparison of two leading quantum error correction (QEC) software frameworks: Loom by Entropica Labs and Deltakit by Riverlane. Both frameworks represent cutting-edge approaches to making quantum error correction accessible, but they serve different primary purposes and excel in different areas.

Loom focuses on visual design and lattice surgery operations, making it ideal for research, rapid prototyping, and educational environments. Deltakit emphasizes structured learning and production deployment, with a comprehensive textbook and integration with Riverlane's hardware stack.

# 1. Introduction

Quantum error correction is essential for building practical, fault-tolerant quantum computers. As the field matures, specialized software frameworks have emerged to simplify QEC implementation, design, and deployment.

This report compares two prominent frameworks that have launched in 2024-2025:

* **Loom by Entropica Labs** - An open-source toolkit emphasizing visual design and lattice surgery
* **Deltakit by Riverlane** - An SDK with comprehensive learning resources and hardware integration

Our analysis examines these frameworks across multiple dimensions including ease of use, capabilities, maturity, documentation quality, and ideal use cases.

# 2. Framework Overview

## 2.1 Loom (Entropica Labs)

Loom is an open-source Python library designed to build, simulate, and validate QEC code primitives. It is part of Entropica Labs' broader vision for Quilt, a Fault-Tolerance Operating System (FTOS).

**Key Components:**

* **Loom Core Library:** Python package for QEC primitives including state preparation, syndrome extraction, logical operations, and decoding routines
* **Entwine:** Browser-based visual IDE for designing lattice surgery operations with drag-and-drop interface
* **EKA Data Structure:** Sanskrit for 'one', provides a single source of truth for QEC state management throughout execution

**Organization:**

Entropica Labs is a Singapore-based company founded by CEO Tommaso Demarie. The company focuses on building the end-to-end software stack for fault-tolerant quantum computing.

## 2.2 Deltakit (Riverlane)

Deltakit is an open-source SDK launched in September 2025 as the software complement to Deltaflow, Riverlane's real-time QEC hardware technology. It provides both learning resources and practical tools for QEC implementation.

**Key Components:**

* **Deltakit SDK:** Python library for generating QEC circuits, adding noise models, simulating execution, and decoding
* **Deltakit Textbook:** Interactive, hands-on introduction to QEC concepts with code exercises and practical examples
* **Cloud Integration:** Access to proprietary decoders including the Local Clustering Decoder (LCD)

**Organization:**

Riverlane, based in Cambridge, UK, is led by CEO Steve Brierley. The company is the global leader in quantum error correction, building both hardware (Deltaflow) and software (Deltakit) solutions. Liz Durst, former lead developer for Qiskit at IBM Quantum, leads Deltakit development as VP of QEC Community.

# 3. Detailed Comparison

## 3.1 Ease of Use

### Loom

**Strengths:**

* Entwine's visual interface makes lattice surgery highly intuitive with drag-and-drop design
* No authentication or cloud setup required - fully local operation
* Browser-based Entwine requires no installation
* Standard Python workflow with pip/poetry installation

**Challenges:**

* Steeper learning curve for the programmatic API
* Less structured learning path compared to Deltakit
* Documentation is good but could benefit from more comprehensive tutorials

### Deltakit

**Strengths:**

* Structured textbook provides clear, progressive learning path from basics to advanced topics
* Simple pip install with straightforward token setup
* Interactive tutorials with code exercises reinforce concepts
* Well-documented SDK with practical examples

**Challenges:**

* Requires cloud token registration (though token is free)
* No visual design interface - purely code-based
* Cloud dependency for advanced decoder features

## 3.2 Capabilities and Features

### Loom Capabilities

* **Lattice Surgery:** Comprehensive visual and programmatic tools for designing lattice surgery operations
* **QEC Codes:** Pre-built implementations of surface codes, repetition codes, Steane code, Shor code, and 5-qubit code
* **Scalability:** Visual designer supports up to 60-70 patches; programmatic approach has no theoretical limit
* **Backend Integration:** Exports to Stim and OpenQASM 3.0; integrates with PennyLane/Catalyst for quantum ML
* **EKA Architecture:** Unique data structure provides single source of truth for QEC state management

### Deltakit Capabilities

* **Noise Modeling:** Comprehensive, realistic noise models based on QPU characteristics
* **Decoders:** Multiple decoder implementations plus cloud access to proprietary Local Clustering Decoder
* **QEC Codes:** Focus on repetition and surface codes with emphasis on practical implementation
* **Hardware Integration:** Designed for seamless integration with Deltaflow hardware stack
* **Real-time QEC:** Focus on real-time error correction execution for production environments

## 3.3 Feature Comparison Matrix

The following table provides a side-by-side comparison of key features:

| **Feature** | **Loom** | **Deltakit** |
| --- | --- | --- |
| **License** | Open-source | Open-source + proprietary cloud |
| **Primary Focus** | Circuit design & lattice surgery | Learning & deployment pipeline |
| **Visual Design Tool** | Yes (Entwine) | No |
| **Installation** | pip/poetry, local | pip + cloud token |
| **Lattice Surgery** | Yes - visual & code | No |
| **Learning Resources** | Examples, blog posts | Comprehensive textbook |
| **Cloud Integration** | No - fully local | Yes - proprietary decoders |
| **Hardware Focus** | Hardware-agnostic | Deltaflow integration |
| **Documentation** | Good (7/10) | Excellent (9/10) |

# 4. Maturity and Ecosystem

## 4.1 Loom Maturity

Loom launched in 2024-2025 as part of Entropica Labs' broader software stack vision. While relatively new, it benefits from:

* Active development by an experienced team in Singapore
* Public validation through the Entropica QEC Challenge (May-June 2025)
* Integration with established quantum frameworks like PennyLane
* Part of a longer-term vision for Quilt FTOS
* Growing community with open GitHub repository

## 4.2 Deltakit Maturity

Deltakit launched in September 2025, but benefits from Riverlane's established position as a QEC leader:

* Backed by Riverlane, a well-funded company with partnerships with leading quantum hardware manufacturers
* Leadership by Liz Durst, former IBM Qiskit lead developer
* Designed as complement to Deltaflow hardware (targeting MegaQuOp scale by 2026)
* Addresses documented industry need: 82% of QEC professionals cite lack of training as barrier
* Active development with planned expansions to documentation and community

## 4.3 Community and Support

**Loom:** Growing academic and research community, particularly in institutions focused on lattice surgery and topological codes. Active GitHub presence with example notebooks and technical papers.

**Deltakit:** Building a global community through its comprehensive textbook. Strong support from Riverlane's partnerships with quantum hardware manufacturers, national labs, and HPC centers.

# 5. Documentation Quality

## 5.1 Loom Documentation

**Rating: 7/10**

**Strengths:**

* Comprehensive API reference documentation available at loom-api-docs.entropicalabs.com
* Clear installation and setup guides
* Example notebooks demonstrating key features
* Technical blog posts explaining concepts and algorithms
* Entwine visual tool is self-explanatory and intuitive

**Areas for Improvement:**

* Could benefit from more comprehensive step-by-step tutorials
* Learning path less structured than Deltakit
* Some advanced features require deeper diving into source code

## 5.2 Deltakit Documentation

**Rating: 9/10**

**Strengths:**

* Comprehensive interactive textbook (textbook.riverlane.com) covering QEC from fundamentals to advanced
* Four structured modules with progressive difficulty
* Code exercises integrated throughout with explanations
* Practical examples demonstrating real-world scenarios
* Clear API documentation for SDK
* Written by Riverlane researchers with contributions from multiple experts
* Directly addresses the documented 82% barrier of lack of QEC training

**Minor Limitations:**

* Very new platform - community contributions and examples still growing
* Some advanced cloud decoder features less documented than core SDK

# 6. Use Cases and Recommendations

## 6.1 When to Choose Loom

Loom is the ideal choice for:

* **Visual design and rapid prototyping:** When you need to quickly sketch and test lattice surgery constructions with an intuitive interface
* **Lattice surgery research:** Deep focus on topological codes and lattice surgery operations
* **Academic environments:** Educational settings where visual tools enhance understanding
* **Local-only workflows:** When cloud integration is not desired or permitted
* **Quantum ML integration:** Projects combining QEC with PennyLane/Catalyst for fault-tolerant quantum machine learning
* **Hardware-agnostic research:** When exploring QEC concepts across different hardware platforms

## 6.2 When to Choose Deltakit

Deltakit is the ideal choice for:

* **Learning QEC from scratch:** Comprehensive textbook provides structured path from basics to advanced concepts
* **Production deployment:** Designed for real-world hardware implementation with Deltaflow integration
* **Advanced decoding:** Access to proprietary high-performance decoders through cloud
* **Realistic noise modeling:** Comprehensive tools for simulating QPU-specific error characteristics
* **Hardware preparation:** When planning for eventual deployment on quantum hardware
* **Team skill development:** Building QEC expertise across development teams with structured resources

## 6.3 Using Both Frameworks

Many practitioners may benefit from using both frameworks in complementary ways:

* **Learning Foundation:** Use Deltakit's textbook to build solid understanding of QEC fundamentals
* **Visual Design:** Leverage Loom's Entwine for intuitive lattice surgery design and rapid prototyping
* **Code Exploration:** Explore different QEC code implementations across both frameworks
* **Research to Production:** Prototype with Loom, then move to Deltakit for hardware-ready implementation

# 7. Technical Implementation Patterns

## 7.1 Loom Code Example Structure

Typical Loom workflow involves:

* Defining QEC codes using pre-built factories (RepetitionCode, RotatedSurfaceCode, etc.)
* Building syndrome extraction circuits
* Designing lattice surgery operations (visually or programmatically)
* Executing on backends (Stim, OpenQASM) through the Executor
* Using EKA data structure to maintain QEC context throughout execution

## 7.2 Deltakit Code Example Structure

Typical Deltakit workflow involves:

* Setting up client authentication with cloud token
* Defining QEC codes (SurfaceCode, RepetitionCode)
* Generating circuits with specified number of rounds
* Adding realistic noise models based on hardware parameters
* Simulating circuit execution
* Decoding syndrome measurements (local or cloud decoders)
* Analyzing logical error rates

# 8. Conclusion

Both Loom and Deltakit represent significant advances in making quantum error correction accessible and practical. While they share the common goal of democratizing QEC, they approach this challenge from different angles and excel in complementary areas.

## 8.1 Key Takeaways

**Loom's Distinctive Strengths:**

* Visual design through Entwine makes complex lattice surgery operations intuitive
* Hardware-agnostic approach suits research and exploration
* Fully local operation without cloud dependencies
* Strong integration with quantum ML frameworks

**Deltakit's Distinctive Strengths:**

* Comprehensive textbook provides structured learning path
* Production-ready approach with hardware integration
* Access to advanced proprietary decoders
* Backed by established QEC leader Riverlane

## 8.2 The Bigger Picture

The emergence of specialized QEC frameworks like Loom and Deltakit signals the maturation of the quantum computing field. Both tools address critical gaps identified in industry surveys, particularly the lack of accessible tools and training for QEC implementation.

Rather than competing directly, these frameworks serve complementary purposes in the QEC ecosystem. Loom excels at visual design and research workflows, while Deltakit focuses on learning and production deployment. Together, they provide a comprehensive toolkit for the quantum computing community.

## 8.3 Future Outlook

Both frameworks are under active development with promising roadmaps:

* Loom is evolving toward Quilt, a comprehensive Fault-Tolerance Operating System
* Deltakit will expand with more documentation, community examples, and tighter Deltaflow integration
* Both are likely to benefit from growing communities and ecosystem partnerships

As quantum error correction moves from theory to practice, tools like Loom and Deltakit will play crucial roles in enabling the next generation of fault-tolerant quantum applications.

# Appendix: Additional Resources

## Loom Resources

* **Documentation:** https://loom-docs.entropicalabs.com/
* **API Reference:** https://loom-api-docs.entropicalabs.com/
* **GitHub Repository:** https://github.com/entropicalabs/el-loom
* **Entwine Visual Tool:** https://entwine.entropicalabs.io/
* **Company Website:** https://entropicalabs.com/

## Deltakit Resources

* **Main Website:** https://deltakit.riverlane.com/
* **Textbook:** https://textbook.riverlane.com/
* **Installation:** pip install deltakit
* **Company Website:** https://www.riverlane.com/
* **Deltaflow Hardware:** https://www.riverlane.com/quantum-error-correction-stack

**Document Information**

**Version:** 1.0

**Date:** October 2025

**Author:** Comparative Analysis

***Note:*** *This comparison is based on publicly available information as of October 2025. Both frameworks are under active development and features may evolve.*