QuantaCrypt-Studio: Simulating Quantum Circuits, Quantum Key Distribution, and Post-Quantum Cryptography

Author: Bryan P. White (ORCID: 0000-0003-4191-3511) • **Affiliation:** CRV Science Labs • **Preprint ID:** APP-PREPRINT-16 (v2.2.0 Empirical Preprint) • **Categories:** quant-ph • cs.CR • cs.MS
Canonical DOI: 10.5281/zenodo.24244089 • **Apps Hub:** https://crvscience.com/apps • **Repository:** https://github.com/bpwhite/crvscience-quantacrypt-studio

Abstract — QuantaCrypt-Studio (v2.2.0) is an open-source 23-module passive stochastic quantum computing and cryptography workbench built in safe Rust. Quantum mechanics and post-quantum security often seem disconnected or intimidating: beginners struggle to picture wave-particle interference and entanglement, while security engineers need concrete ways to compare quantum key distribution (BB84, E91, B92) against lattice-based post-quantum cryptography (ML-KEM) and fault-tolerant surface codes. QuantaCrypt-Studio bridges that gap inside one self-contained offline desktop app. Users can step through classic quantum-intuition experiments (double-slit, Mach-Zehnder, Stern-Gerlach, quantum Zeno), build and export OpenQASM qubit circuits with realistic Kraus noise channels, simulate higher-dimensional qudits, test eavesdropping detection in photonic QKD, inspect lattice cryptography and Shor/Grover attacks, and explore surface-code error correction and variational algorithms (VQE, QAOA).

1. Real-World Context & Why This Problem Matters

Why Quantum Mechanics and Cryptography Need One Shared Workbench

Quantum technology touches two halves of modern computing at once: harnessing superposition, interference, and entanglement to solve specialized linear-algebra and physics problems, and protecting global communications against future quantum attacks. Yet most software tools focus on only one isolated slice of that story—either low-level circuit matrix libraries written for quantum-hardware specialists or single-purpose textbook applets illustrating one optical experiment at a time.

To truly grasp why large quantum computers threaten RSA integer factoring while photonic quantum key distribution (QKD)[1][2] and structured lattice cryptography resist them, learners need to see probability amplitudes, Kraus noise channels, eavesdropper disturbance, and error-correcting surface codes[3] operating under consistent mathematical rules. QuantaCrypt-Studio unites all 23 pedagogical and research modules inside a single deterministic Rust desktop studio.

Think of QuantaCrypt-Studio as a glass-walled quantum and security laboratory: every probability amplitude, noisy channel parameter, eavesdropper bit-error spike, and cryptographic key exchange sits right on screen, backed by reproducible NIST FIPS 180-4 SHA-256 random streams and portable workset envelopes.

From Double-Slit Intuition to Fault-Tolerant Surface Codes

QuantaCrypt-Studio (v `2.2.0`) is written in 100% safe Rust with zero external quantum SDK or C++ dependencies, combining exact complex statevector and density-matrix engines with interactive Bloch-sphere and Wigner plots, OpenQASM example circuits, and tamper-evident SQLite worksets. Learners can begin with intuitive single-photon interference and progress step by step to multi-qubit circuits, quantum teleportation, QKD security bounds, and post-quantum lattice encapsulation without switching tools.

Rather than treating quantum mechanics as an abstract collection of Dirac-bracket identities divorced from engineering reality, each module in the 23-part curriculum highlights how physical assumptions translate into measurable outputs. Users can inspect why entanglement entropy rises when tracing out one half of a Bell pair, how Weyl-Heisenberg operators generalize Pauli gates to higher-dimensional qudits, and why surface-code syndrome cycles require physical gate error rates below threshold before logical qubits outlive bare physical qubits. Because every simulation uses deterministic SHA-256 random streams, classroom exercises and cryptographic parameter sweeps can be shared and replayed identically across machines.

2. Conceptual Design & How QuantaCrypt-Studio Works

Statevectors, Density Matrices, Photonic QKD, and Lattice PQC

At the numerical core, `complex.rs`, `statevector.rs`, and `density.rs` implement exact complex linear algebra, multi-qubit unitary gate evolution, OpenQASM 2.0 round-tripping, partial traces, von Neumann entanglement entropy, and full Kraus decoherence channels (bit-flip, phase-flip, amplitude damping, and depolarizing noise). Higher-dimensional Weyl-Heisenberg qudits and entanglement-bit accounting live in `qudit.rs`, letting users move beyond two-level qubits to d-level quantum systems.

Cryptographic and algorithmic modules span `qkd.rs` (BB84[1], E91 Bell-CHSH[2], and B92 protocols with intercept-resend eavesdropping detection), `pqc_lattice.rs` (FIPS 203 ML-KEM key encapsulation and lattice signature rejection sampling), and `cryptanalysis.rs` (Shor period finding and Grover amplitude amplification).

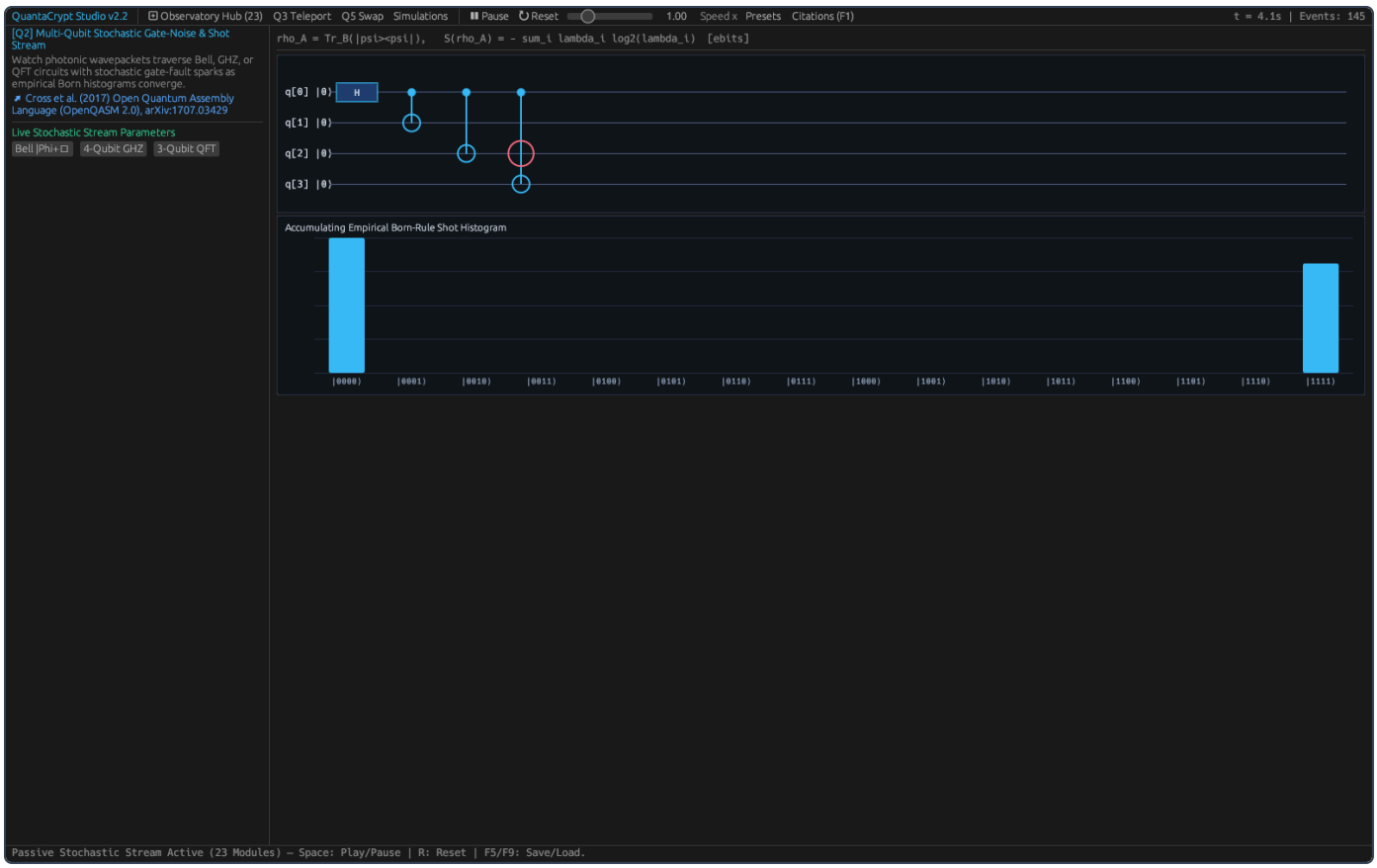
A unified 23-module passive stochastic arena and structured curriculum navigator live in `stochastic_sim.rs` and `curriculum.rs`, covering double-slit, Mach-Zehnder, Stern-Gerlach, quantum Zeno, three-qubit and surface-code error correction[3], variational quantum eigensolvers (VQE), QAOA MaxCut, and magic-state distillation. Deterministic SHA-256 seeded entropy, custom plotting, interactive UI, and SQLite persistence are handled by `rng_sha256.rs`, `plotter.rs`, `gui.rs`, and `workset.rs`, all verified by `tests/integration_suite.rs` (10/10 checks passing).

Every module inside the curriculum navigator pairs concise conceptual notes with live parameter sliders so learners never have to guess what a gate angle, decoherence rate, or lattice error bound actually changes. Adjusting a single slider immediately updates Bloch coordinates, probability histograms, Wigner quasi-probability contours, and QBER security readouts in real time, making subtle quantum-information phenomena—like why measurement basis mismatches leak zero information or why surface-code syndrome rounds fail above physical threshold—visually unmistakable.

Table 1. Software Modules, Descriptions, and Citations (QuantaCrypt-Studio)

Module / Crate	What It Does	Citation
`complex.rs`, `statevector.rs` & `density.rs`	Complex matrix math, multi-qubit OpenQASM circuit simulation, Kraus noise channels, and 3-qubit/surface-code error recovery.	Fowler et al. (2012) [3]
`qudit.rs`	High-dimensional d-level qudit Weyl-Heisenberg operators and entanglement accounting.	CRV Science Apps Network [4–20]
`qkd.rs`	Photonic BB84, E91 Bell-CHSH, and B92 quantum key distribution with eavesdropper QBER detection.	Bennett & Brassard (2014) [1]; Ekert (1991) [2]
`pqc_lattice.rs` & `cryptanalysis.rs`	FIPS 203 ML-KEM lattice encapsulation, rejection-sampling signatures, Shor factoring, and Grover search.	CRV Science Apps Network [4–20]
`stochastic_sim.rs` & `curriculum.rs`	23-module passive stochastic quantum arena spanning intuition labs, VQE, QAOA, and magic-state distillation.	CRV Science Apps Network [4–20]
`gui.rs`, `plotter.rs`, `rng_sha256.rs` & `workset.rs`	Interactive Bloch/circuit plotting, NIST FIPS 180-4 SHA-256 seeded PRNG, and tamper-evident SQLite worksets.	CRV Science Apps Network [4–20]

Figure 1. Sample Screenshot: QuantaCrypt-Studio Quantum Circuit & Statevector Workbench



Sample screenshot of the multi-qubit OpenQASM circuit editor, Bloch sphere readout, and exact statevector probabilities.

3. Walkthrough of Capabilities & Quantitative Findings

Noise Thresholds, QBER Security Regimes, and Error Correction

Across its 23 modules, QuantaCrypt-Studio lets users observe exact physical thresholds rather than hand-waving abstractions. In three-qubit repetition codes and planar surface-code error correction[3], toggling single-qubit Kraus error rates immediately reveals when stabilizer syndrome extraction recovers near-unit logical fidelity and where physical noise crosses threshold and overwhelms protection. Watching entanglement entropy and Bloch vectors contract under amplitude damping makes decoherence tangible.

In photonic QKD, users can toggle an active intercept-resend eavesdropper on the quantum channel and watch the quantum bit error rate (QBER) jump above the BB84 security bound while the E91 Bell-CHSH correlation parameter collapses below the classical limit[1][2]. Instead of merely reading that measurement disturbs quantum states, learners observe the statistical signature of eavesdropping directly in sifted key bits.

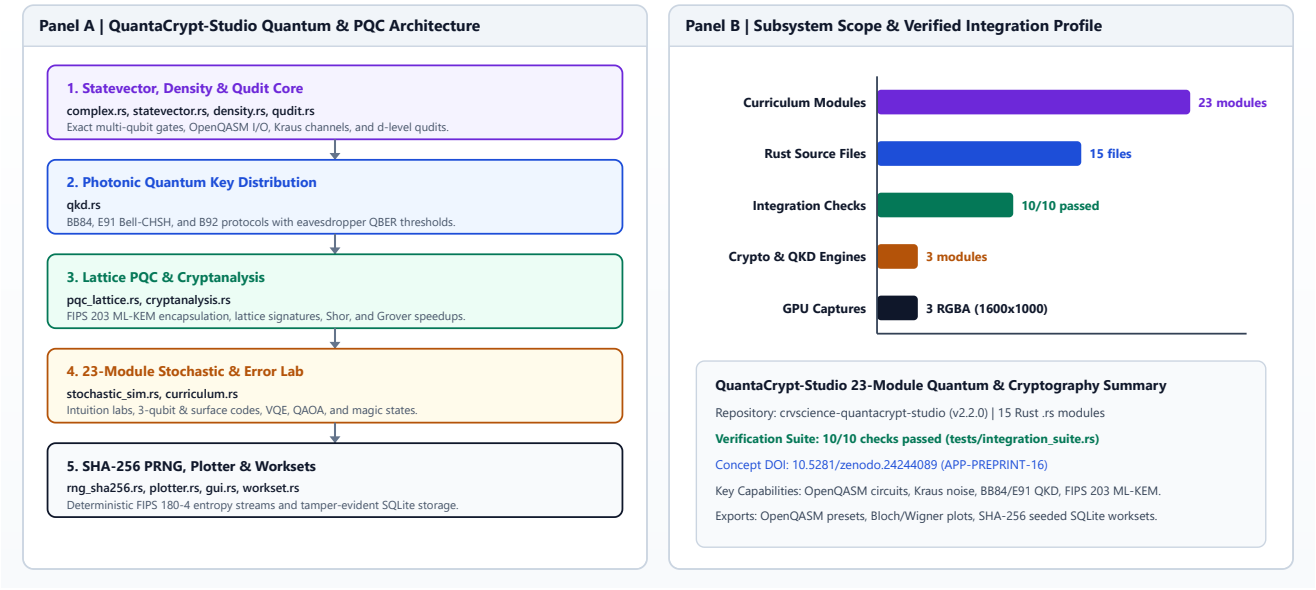
In post-quantum cryptography, users walk step by step through polynomial ring arithmetic and error-tolerance bounds in FIPS 203 ML-KEM alongside Grover quadratic search speedup and Shor modular-exponentiation period finding—seeing directly why symmetric key lengths double under Grover's algorithm while structured lattice problems remain hard. Built-in OpenQASM presets (Bell-CHSH, 4-qubit GHZ, 3-qubit Grover, and quantum teleportation) let users inspect and export standard circuit text immediately. Variational hybrid workflows such as VQE and QAOA round out the workbench by showing how parameterized ansätze optimize ground-state energies and combinatorial graph cuts interactively.

A typical guided session in QuantaCrypt-Studio shows how intuition builds cleanly from single-photon interference up to post-quantum security. Starting in the double-slit and Mach-Zehnder modules, a student adjusts relative optical phase shifts and watches constructive and destructive interference fringes emerge smoothly from complex probability amplitudes, then enables a which-path detector to see interference vanish instantly as entanglement transfers path information to the environment. Moving to the multi-qubit circuit studio, the student loads the Bell-CHSH OpenQASM preset, applies depolarizing or amplitude-damping Kraus noise, and observes both the Bloch-vector shrinkage and the rising mixed-state entropy in the density-matrix readout. Stepping over to the photonic QKD lab, toggling an active intercept-resend eavesdropper pushes the sifted BB84 error rate past the eleven-percent threshold while breaking E91 Bell-CHSH inequality violations[1][2]. Finally, opening the lattice PQC tab demonstrates how ML-KEM encapsulates shared keys using noisy ring equations that resist both Shor's factoring circuit and Grover's unstructured search, while the surface-code module[3] illustrates how physical qubits band together to keep logical error rates suppressed below fault-tolerance thresholds.

Table 2. Representative Capabilities and Quantitative Findings (APP-PREPRINT-16)

Subsystem	Verified Scope	Suite Result
Statevector & Density	OpenQASM I/O, Kraus noise, Weyl-Heisenberg qudits	Verified in `tests/integration_suite.rs`
QKD & Error Correction	BB84, E91 CHSH, B92 & 3-qubit/surface-code recovery	10/10 integration checks passed
Lattice PQC & Attacks	FIPS 203 ML-KEM, Shor/Grover & 23-module arena	15 Rust modules (v2.2.0)

Figure 2. QuantaCrypt-Studio (v2.2.0) 23-module quantum and cryptography architecture



QuantaCrypt-Studio (v2.2.0) 23-module quantum and cryptography architecture. Panel A links statevector/density-matrix physics to photonic QKD, ML-KEM lattice cryptography, and error correction. Panel B summarizes module scope and verified integration checks.

4. Practical Scenarios: Who Uses QuantaCrypt-Studio & How

Physics, Computer Science, and Cybersecurity Education

Physics, mathematics, and computer-science instructors use QuantaCrypt-Studio to guide students smoothly from freshman wave-particle interference intuition to senior-level density matrices, variational hybrid algorithms (VQE/QAOA), and fault-tolerant surface codes. Because every module shares consistent state-inspection plots and SHA-256 seeded repeatability, lab exercises remain predictable and easy to grade.

Cybersecurity teams, cryptography students, and software engineers can compare physical QKD security assumptions against NIST-standardized lattice PQC side by side without configuring complex Python, Qiskit, or C++ toolchains. Saved SQLite worksets and OpenQASM circuit files make it straightforward to archive classroom demonstrations or self-study experiments.

Just as importantly, QuantaCrypt-Studio demystifies the conceptual gulf between popular headlines about quantum computing and the actual linear algebra underneath. Many articles describe qubits vaguely as "being zero and one at the same time," leaving readers puzzled about why quantum computers speed up factoring and unstructured search while failing to solve every hard problem instantaneously. Inside the 23-module workbench, users see directly that quantum advantage relies on carefully choreographed constructive and destructive phase interference across complex amplitudes—and that environmental decoherence, modeled honestly through Kraus bit-flip, phase-flip, and amplitude-damping channels, constantly threatens to wash those delicate phases away unless error-correcting codes step in.

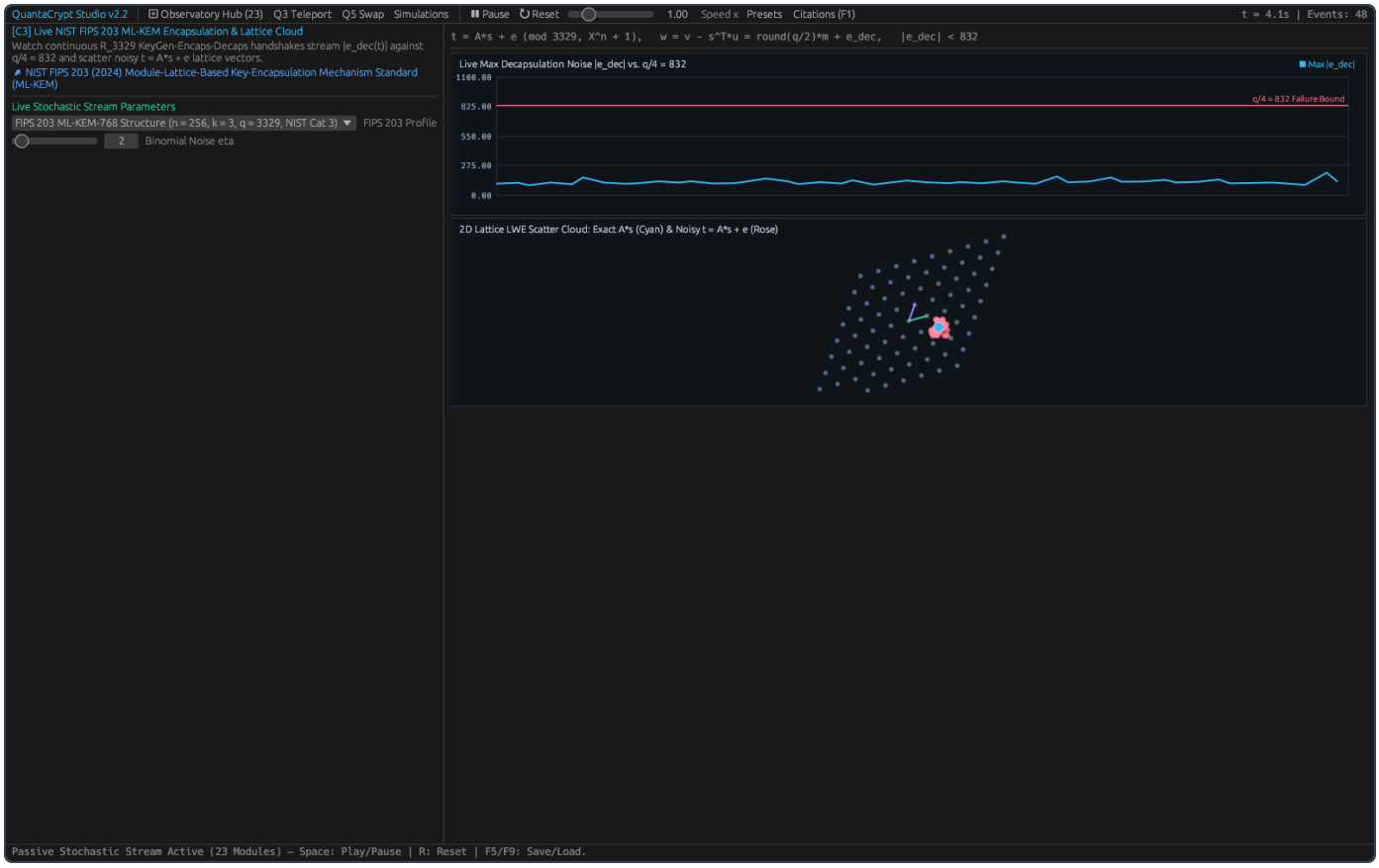
Figure 3. Sample Screenshot: Photonic QKD (BB84 / E91) & Eavesdropper QBER Security View



Sample screenshot showing BB84 and E91 Bell-CHSH quantum key distribution diagnostics with eavesdropper QBER threshold detection.

Equally illuminating for security practitioners is the side-by-side contrast between physical-layer quantum key distribution and mathematical post-quantum cryptography. While BB84 and E91 rely on quantum no-cloning and Bell inequalities over dedicated optical channels[1][2], FIPS 203 ML-KEM runs over ordinary classical internet packets by hiding secrets inside noisy high-dimensional lattice equations. Exploring both inside one offline Rust studio gives engineers, students, and educators a grounded, quantitative view of where quantum threats begin and how modern cryptography adapts. Combined with OpenQASM circuit export and tamper-evident SQLite worksets, every demonstration doubles as a reproducible teaching artifact.

Figure 4. Sample Screenshot: Lattice Post-Quantum Cryptography (ML-KEM) & Surface Code Error Correction



Sample screenshot displaying FIPS 203 ML-KEM lattice key encapsulation alongside noisy surface-code syndrome recovery.

5. Open Access & Connections Across the CRV Science App Network

Open Access and Connections Across the CRV Science App Network

QuantaCrypt-Studio is released under the MIT license with full Rust source code, continuous-integration checks, OpenQASM examples, and a detailed user guide PDF on GitHub. All 15 Rust source files compile offline into a single standalone desktop studio. Whether a reader is exploring double-slit interference for the very first time, testing how three-qubit syndrome extraction corrects bit-flip errors, or benchmarking how NIST FIPS 203 lattice parameters scale against quantum cryptanalysis, every calculation remains transparent, deterministic, and inspectable down to the source line.

Within the CRV Science Apps network[4–20], QuantaCrypt-Studio anchors the quantum-information and cryptography pillar alongside numerical differential-equation and linear-algebra solvers in Calc-Lab, differential-privacy epidemiology in EpiLedger, and network citation graphs in Cit-Atlas. By sharing clean offline design habits and mutual Zenodo Concept DOI citations across all seventeen applications, the suite turns specialized computational subjects into approachable desktop instruments.

References

[1] Bennett, C. H., & Brassard, G. (2014). Quantum cryptography: Public key distribution and coin tossing. Theoretical Computer Science. https://doi.org/10.1016/j.tcs.2014.05.025

[2] Ekert, A. K. (1991). Quantum cryptography based on Bell's theorem. Physical Review Letters. https://doi.org/10.1103/physrevlett.67.661

[3] Fowler, A. G., et al. (2012). Surface codes: Towards practical large-scale quantum computation. Physical Review A. https://doi.org/10.1103/physreva.86.032324

[4] White, B. P. (2026). The CRV Science App Network: Seventeen Open-Source Rust Programs for Science, Data, and Simulation. CRV Science App Preprint APP-PREPRINT-01 [cs.SE, cs.CE, q-bio.QM], v1.0.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222398

[5] White, B. P. (2026). AeroFlyway: Tracking Bird Migration with Weather Radar, Skyglow Maps, and Nighttime Flight Calls. CRV Science App Preprint APP-PREPRINT-13 [q-bio.PE, physics.ao-ph, cs.GR], v0.6.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222366

[6] White, B. P. (2026). BeoFold: Simulating How Proteins Fold Using Distributed Computing and State Networks. CRV Science App Preprint APP-PREPRINT-02 [q-bio.BM, physics.chem-ph, cs.DC], v1.4.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222370

[7] White, B. P. (2026). Calc-Lab: Interactive Calculus, Linear Algebra, and Differential Equation Solver Workbench. CRV Science App Preprint APP-PREPRINT-03 [math.NA, cs.MS, cs.CY], v0.5.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222372

[8] White, B. P. (2026). Causal-Atlas: Tracing How Genes and Drugs Turn Biological Pathways On and Off. CRV Science App Preprint APP-PREPRINT-04 [q-bio.MN, q-bio.GN, stat.ML], v0.2.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222376

[9] White, B. P. (2026). Cit-Atlas: Mapping Citation Networks and Research Impact Across Scientific Papers. CRV Science App Preprint APP-PREPRINT-05 [cs.DL, cs.SI, physics.soc-ph], v1.8.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222380

[10] White, B. P. (2026). EpiLedger: Privacy-Safe Disease Tracking, Clinical Text Screening, and Outbreak Monitoring. CRV Science App Preprint APP-PREPRINT-06 [cs.CR, q-bio.PE, stat.AP], v2.0.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222382

[11] White, B. P. (2026). ITME: Simulating Four Billion Years of Life, Oceans, and Atmosphere Evolution. CRV Science App Preprint APP-PREPRINT-07 [q-bio.PE, nlin.AO, physics.bio-ph], v2.3.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222386

[12] White, B. P. (2026). OSABT: All-in-One Toolkit for Plant Physiology, Survival Curves, Genetics, and Enzyme Kinetics. CRV Science App Preprint APP-PREPRINT-08 [q-bio.QM, stat.AP, cs.SE], v2.0.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222388

[13] White, B. P. (2026). SAR-Atlas: Drug Design, Body Distribution Modeling, and Candidate Screening in Rust. CRV Science App Preprint APP-PREPRINT-09 [physics.chem-ph, q-bio.BM, cs.LG], v3.8.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222390

[14] White, B. P. (2026). TerraTwin: Fast Climate Projections for Global Warming, Sea-Level Rise, and Arctic Change. CRV Science App Preprint APP-PREPRINT-10 [physics.ao-ph, cs.CE, stat.AP], v2.5.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222392

[15] White, B. P. (2026). Urban-Grid: Modeling City Traffic, Emergency Response Coverage, and Earthquake Evacuation. CRV Science App Preprint APP-PREPRINT-11 [physics.soc-ph, cs.MA, cs.CE], v5.2.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222394

[16] White, B. P. (2026). WildFront BattleGrounds: Turn-Based Hex Strategy Game with Voxel Graphics and Procedural Map Generation. CRV Science App Preprint APP-PREPRINT-12 [cs.GT, cs.GR, math.CO], v3.4.1. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24222396

[17] White, B. P. (2026). HelioProbe: Simulating Spacecraft Orbits, Deep-Space Telescopes, and Comet Tails in 3D. CRV Science App Preprint APP-PREPRINT-14 [astro-ph.EP, astro-ph.IM, cs.GR], v2.3.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24244083

[18] White, B. P. (2026). PopBio-Studio: Interactive Population Biology, Adaptive Landscapes, and Genetic Skyline Simulator. CRV Science App Preprint APP-PREPRINT-15 [q-bio.PE, q-bio.QM, stat.AP], v1.2.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24244085

[19] White, B. P. (2026). SonoWing: Analyzing Bat Echolocation and Bird Calls with Spectrograms and Harmonic Tracking. CRV Science App Preprint APP-PREPRINT-17 [q-bio.NC, q-bio.PE, cs.SD], v1.1.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24244091

[20] White, B. P. (2026). SwarmForge: 3D Isometric Automated Factory and Swarm Robotics Simulation Game. CRV Science App Preprint APP-PREPRINT-18 [cs.MA, cs.RO, cs.GT], v1.3.0. Hosted via CRV Science Apps Hub (https://crvscience.com/apps). https://doi.org/10.5281/zenodo.24244093