

Marcus Vance

Lead Quantitative Developer • Ultra-Low Latency Systems

📍 New York, NY • ✉️ marcus.vance@nexa-tech.io • 📞 +1 (555) 382-9901 • 🌐 github.com/marcusvance-quant • [linkedin.com/marcus-vance-quant](https://www.linkedin.com/in/marcus-vance-quant)

Technical Skills

Languages	C++ (C++20/17/14), Python (NumPy, Pandas, SciPy, PyTorch), Bash, SQL, q (KDB+), Assembly (x86_64)
Low-Latency & Systems	Multithreading (std::thread), Lock-free Programming, SIMD (AVX-512), Kernel Bypass (EF_VI/OpenOnload), Cache Optimization
Libraries & Tools	CMake, Git, GDB, Valgrind, Nexa Benchmark, Eigen, Boost, Docker, Linux Systems Programming
Quantitative Finance	Market Data (FIX, ITCH, OUCH), Tick-to-Trade Optimization, Stochastic Calculus, Mathematical Modeling, Statistics

Professional Experience

Lead Quantitative Developer • Apex Trading Group Jan 2023 – Present
Ultra-Low Latency Market Making & Algorithmic Execution New York, NY

- Designed and optimized **Nexa-Feed**, a zero-allocation C++20 market data feed handler for Meridex ITCH/OUCH, reducing p99 latency from 4.2 to 1.8 microseconds. - Implemented lock-free ring buffers and shared memory IPC, achieving sub-50 nanosecond messaging latency across adjacent CPU cores. - Developed a Python-C++ hybrid backtesting platform with **Vertex-Sim**, enabling historical replay of multi-gigabyte intraday order books with microsecond resolution. - Streamlined execution code paths utilizing AVX-512 SIMD vectorization, achieving a 35% speedup in real-time covariance and volatility estimators.

Senior Quantitative Developer • Vertex Capital Sep 2020 – Dec 2022
High-Frequency Futures Market Making Chicago, IL

- Developed and deployed **Synapse-Gate**, a low-latency execution gateway for the Meridian Futures Exchange, processing peak traffic of 200,000 orders/sec. - Implemented custom memory-mapped ring buffers and low-overhead loggers, reducing worst-case execution paths by 15% (saving 300 nanoseconds). - Designed high-performance Linux kernel-bypass socket handlers, reducing tick-to-trade latency under peak market volatility. - Created a Python-C++ hybrid statistical arbitrage library, integrating C++ vector kernels with Python backtesters.

Quantitative Software Engineer • Synapse Systems Jul 2018 – Aug 2020
Algorithmic Execution & Infrastructure Boston, MA

- Engineered an order-routing engine in C++17, utilizing custom pool allocators to achieve deterministic memory management and eliminate runtime heap allocations. - Maintained tick-level databases storing historical market data, designing schema structures that optimized query speeds for quant research backtesting. - Implemented multi-threaded execution models, dividing market feed reading, signal generation, and order sending across dedicated cores.

Projects

Meridian-Backtester • High-Performance Simulator C++, Python, Git

- Developed an event-driven backtesting engine in C++20 with a Python wrapper, simulating order book dynamics and matching engines at nanosecond granularity. - Leveraged zero-copy parsing of historical ITCH files to process over 20 GB of raw market data per minute.

Nimbus-MD • UDP Multicast Parser C++, Linux Systems Programming

- Built a high-speed UDP multicast parser using Linux epoll and socket buffer tuning, achieving zero packet loss at peak line rates.

Education

Meridian Institute of Technology Sep 2016 – Jun 2018
Master of Science in Financial Engineering GPA: 3.92/4.00

- Focus: Stochastic Calculus, Quantitative Risk Management, Numerical Methods, Time Series Analysis.

Nexa University Sep 2012 – Jun 2016
Bachelor of Science in Computer Science & Applied Mathematics Graduated Magna Cum Laude

- Focus: Data Structures & Algorithms, Real Analysis, Linear Algebra, Parallel Programming.