

ROY WILLIAM WARD

<https://www.orange-kiwi.com>

roy@orange-kiwi.com

SUMMARY

Uncommonly skilled at finding software performance bottlenecks and addressing them. Software architect and engineer with exceptional record of innovation and value delivery, seeking an individual contributor role where software performance and team collaboration are emphasized. Over 20 years of full-time experience building high performance and low memory solutions, resulting in 12 US and 7 international software patents. Expertise in optimization, algorithms, data structures, compilers, microprocessor architecture, memory management, assemblers, operating systems, number representation, concurrency/multithreading, compression, GPU rendering. <https://www.orange-kiwi.com> has some exemplars of my work including optimization blog posts.

PROFESSIONAL EXPERIENCE

2024-present: Senior Prolog Developer for LogicMoo

- Enhanced the MeTTalog language interpreter for the language MeTTa
- Redesigned the MeTTalog to Prolog Compiler
- Developed a Language Server Protocol Server for MeTTa

2010-2024: Lead Applications Engineer and Architect for Moonshadow Mobile Inc.

- Lead developer of MMZIP, a stand alone, fully streaming, compression/annotation/filtering ETL tool, written in Modern C++ and optimized for geospatial time series data
 - Wrote a data analyzer that adaptively determined the best compression method for each column
 - Wrote a custom compressor/decompressor that typically produced archives 35-50% the size of gzip archives
 - Built components that could match roads extracted from OSM or WKT data at over 1 million rows/second and region match at 5-20 million rows/second
- Lead developer and architect of the Ephemeris engine, a high performance, multithreaded, in memory, columnar database, written in C++ and optimized for geospatial and time series data. Ephemeris was used by the Hurricane Evacuation Traffic Project, where 3 second frequency/vehicle data from seven states was processed and made available live using just two servers, ingesting over 100,000 records per second at peak.
 - Wrote an interpreter that could process about 50 million records/second/core
 - Wrote a JIT compiler that converted queries to x86-64 machine code (including using AVX2) in about 1-2 milliseconds, generating code that could process 100-200 million records/second/core
 - Wrote custom JSON and CSV parsing libraries for ingestion of over 100,000 records/second
 - Added in-engine functionality for very fast data visualization by directly generating PNG tiles
 - Wrote a lockless LRU cache, and low latency locking code for managing concurrent queries
 - Collaborated with and supervised other developers

2002-2010: Lead Developer and Architect for BuyMusicHere Inc.

- Lead developer of the BuyMusicHere caching server, a multithreaded in memory store for metadata about CDs, DVDs and books, written in Java that provided caching, searching (using Lucene) and business logic between the PHP front end and the PostgreSQL backing store.

PROJECTS

- <https://www.orange-kiwi.com> A blog on software optimization and related topics
- <https://github.com/royward/pseudo-double> A relatively fast C and C++ 64 bit floating point library written using only integer operations
- <https://www.orange-kiwi.com/galaxy-generator/> Procedural generation and fast navigation in a 400 billion star galaxy on Linux, Android and Windows written using C++ and Vulkan

EDUCATION

B.Sc. (Hons) 1st Class in Mathematics from the University of Otago, Dunedin, New Zealand.

SKILLS

Languages: Modern C++, C, Java, Rust, SQL, Python, GLSL, Haskell, Prolog, Verilog, x86-64 (w/AVX), Bash
Platforms: Linux, Android (native), Windows. **Hardware:** Xilinx FPGAs/CPLDs, PIC Microcontrollers.
Tools and Libraries: PostgreSQL/libpq, POSIX, gdb, valgrind, Vulkan, git, Lucene, LaTeX, Dear ImGui.