

# Nim metaprogramming for performance, portability, and productivity

*Friday, 26 April 2019 16:00 (5 minutes)*

The Nim programming language has an extensive metaprogramming capability. With the help of Nim's AST-based metaprogramming, we are building a framework where the application code is as high-level as in Python, and the low-level architecture specific code is compartmentalized. This allows us to target OpenMP, CUDA, and OpenCL for supported architectures. Here we describe our ongoing process of converting our CPU only lattice field theory framework QEX (Quantum EXpressions) as an effort of the ECP application team, and compare the performance of key math kernels with multiple backends, including OpenMP with target device offloading, CUDA, and OpenCL. We show how a LISP-like macro system in Nim, combined with its feature rich type system, helps keep us mostly at a high level, while allowing us to dig in to the performance kernels and influence the C or C++ code generated by Nim.

## Summary

**Presenter:** JIN, Xiao-Yong (RIKEN Advanced Institute for Computational Science)

**Session Classification:** Poster Session w/ Coffee