XCOMPUTE:
Algorithms and Instruction Sequences for CFD/FEA Multi-Physics

G.J. Orr*, R.J. Kwan*, M. Doudar*

* Xplicit Computing, Inc. USA

Abstract

Despite continuous growth in CPU & GPU processing bandwidth, many CAE software tools cannot leverage heterogeneous hardware across teams and systems; a universal vectorized specification is required for: geometry preparation, physics definition, condition assignment, solver numerics, or other useful data-processing functions as part of CFD & FEA and future integrations. The objects and interfaces of said “xcompute” pipeline are detailed here; an executable instruction comprises an algorithm/verb bound to one or more argument(s)/noun(s) and sequenced into physical models and solvers to define the procedures of high-level workflows. Instructions and sequences are assembled from human-facing building blocks to meet a variety of customizations and optimizations. Algorithms can be defined by static host bytecode and/or implemented as a device kernel as part of a compiled OpenCL program, callable from a solver program. Property-key inputs, outputs, and constants permit algorithms to connect in an executable sequence or be invoked individually as part of a polymorphic compute framework. Multiple networked client and server sessions interact in real-time using xcompute protocol from the new libxccommon and libxcmessages libraries. Execution strategies include: interactive mode, command line shell, protocol buffer language bindings, and spreadsheet exchange.