

## Exascale: Can My Code Get from Here to There?"

Today's high-end scientific and engineering software is formulated to fit an execution model that we have evolved to over half a century. The primary features of this model that distinguish it from that used two decades ago are distributed memory and message passing. Additional features that have recently crept in include shared memory nodes, increasingly sophisticated virtual memory hierarchies, multicore CPUs, and SIMD ALUs. It is getting increasingly difficult for application developers to map their codes to today's petascale systems given a programming environment that is cobbled together from a mixture of programming languages, extensions, and libraries, almost all designed for the systems fielded in the last millenium. This talk will discuss features expected in exascale systems that are not well represented in today's programming model. The objective will be to bring these problems to the fore, not propose solutions to them.