

# Platforms for open-source scientific software

DAVE HALE, COLORADO SCHOOL OF MINES  
*dhale@mines.edu*

## Abstract

When developing open-source scientific software, what programming languages should we use? What software libraries should we build upon? What operating systems should we support? And what open-source license should we adopt? Our answers to these and other questions define the platform, and may well determine the success, of our software systems.

When extending an existing software system, these questions have already been answered, perhaps years ago when choices were fewer and simpler. Then, perhaps even our definition of scientific software was simpler. (Did that include parallel computing, or interactive graphics?) But for today's open-source software, we must choose from an increasingly bewildering variety of platforms and consider scientific computing beyond simple array processing.

Today, two platforms are most promising. The first (Java) platform is defined primarily by the Java programming language and the many class libraries that are part of the standard Java runtime environment. The second (C++) platform is defined by a combination of the programming languages C++ and Python, their standard libraries, plus the non-standard Qt or wxWidgets and other libraries.

Both platforms are freely available and widely used for open-source software. Both support multiple operating systems, including Linux, Windows, and Mac OS X. Both support the use of open-source software libraries written in other languages. And both facilitate a broad definition of scientific computing.

However, these two platforms are not equivalent. Furthermore, though both enable the use of software written in other languages, *the Java and C++ platforms are largely incompatible. We must choose one or the other.*

Before choosing, we should compare different aspects of the Java and C++ platforms. Numerical performance is one obvious aspect, but differences here are less significant than we might expect. Other aspects include portability to different operating systems, and the quality and licensing of software libraries. But *simplicity* is the aspect in which these two platforms differ most, and this aspect leads us to favor the Java platform.