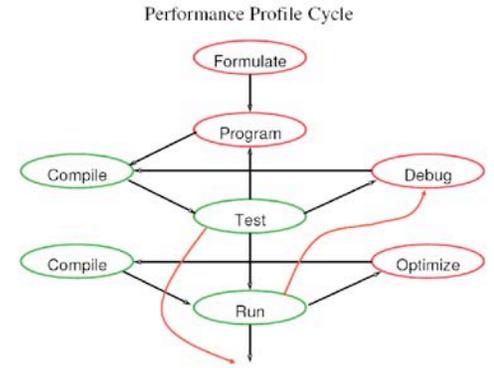


# Maximizing Application Performance on ASC Platforms

Effective utilization of ASC computing platforms requires that ASC codes be optimized to best use the available computing and communication resources. To perform this optimization, developers need insight into the details of program execution. ASC has supported a variety of tools for giving developers a detailed view of how their applications run on the platforms.

A portion of the ASC tool portfolio will be highlighted by a series of presentations at the ASC booth. Accompanying demonstrations will focus on investigative methodology and actual analysis using ASC performance measurement tools. We will step through the process of optimization for sample codes. We cover how these tools can be used to examine computation to communications balance, computational efficiency, and communication efficiency as well as code coverage.

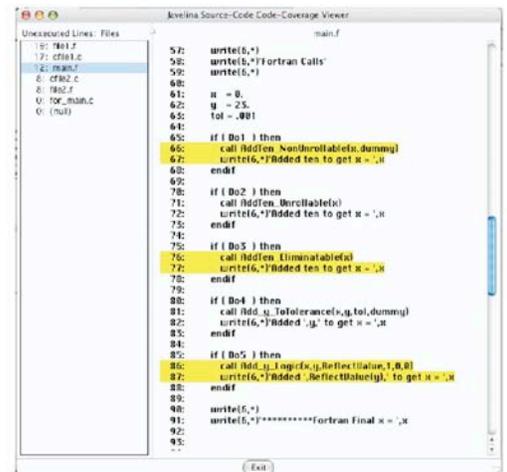
The tools presented range from the mature to those currently under



development. For an in-depth understanding of the performance optimization process as well as the tools, participants can request demonstrations at the performance tools demo station in which the cycle of analysis, optimization, and re-analysis is shown. Additionally, attendees can request guest accounts for the duration of the conference on a local cluster and use the tools themselves either on the code being demonstrated or code of their own.

The following is a brief description of the tools covered by these talks and demonstrations:

**Javelina** – An advanced code coverage tool that uses dynamic instrumentation. Dynamic instrumentation allows code coverage data to be acquired with a minimum overhead. Once a section of code has been executed, the instrumentation for that code is removed. Thus, there is a performance hit only the first time a section of code is executed. This is particularly beneficial for scientific applications where loops are repeatedly evaluated. Furthermore, Javelina allows one to apply advanced logical operations to the acquired coverage data. This has allowed LANL to optimize the testing of ASC applications by focusing the testing effort on the sections of code that are executed by end users but are not exercised by test cases.



ASC Performance Tools are critical to the optimization of code performance on large scale ASC platforms.

*Visualize the Difference*



