wGUI: A Web-Based Graphical User Interface for High-Performance Computing[*]

Sergey Avdeev[1] and B. A. Shadwick[2]

[1] Department of Computer Science and Engineering
University of Nebraska - Lincoln

[2] Department of Physics and Astronomy
University of Nebraska - Lincoln

It is becoming more common for scientific codes running on workstations to include a GUI to allow the user to launch, monitor and control calculations. Migrating such codes to a high-performance cluster platform invariably requires abandoning the GUI in favor of traditional input files and post-processing of results. The relative inconvenience of the user interface in a typical cluster environment leads some users to migrate to clusters as a last resort. To ease the transition from the workstation and GUI environment to a cluster environment, we have developed a platform independent architecture for adding a sophisticated GUI to applications running on clusters.

The system architecture consists of two elements: a custom web-server (middleware layer) that mediates interactions between the user and the computational kernel; and a lightweight XML-RPC server library embedded in the computational kernel. Typically the middleware will run on the publicly accessible master-node of a cluster. The XML-RPC server library is added to the scientific code. This server, running as a separate thread, processes requests from the middleware in response to user actions. The server adds little overhead to the computational processes running on the nodes as it is typically idle unless responding to user-initiated tasks. An important design consideration is to have all communication between the middleware and the computational code running on cluster nodes be "out-of-band." That is, all such communication does not use MPI so as to avoid contention with the scientific code. In its simplest form, this system allows graphical inspection of internal data-structures in the scientific code and gives real-time access to all log messages. Alternatively, by exposing functions to the XML-RPC server, complex GUI applications can be readily created.

A preliminary implementation of this system has been developed around a hydrodynamic laser-plasma interaction code. This code was originally developed with an OS-dependent GUI. At present, wGUI is able to reproduce most of the existing GUI features of the code as well as adding some new features. In this implementation, the middleware is based on the Python web-application framework `CherryPy`. The C library `xmlrpc-c` provides the XML-RPC server functions in the kernel. This server functions as a supervisor directing the actions of the code in a manner analogous to a typical GUI event handler. We will give a live demonstration of wGUI system, launching a new task as well as monitoring intermediate and final results.