Methods toward Automatic Configuration of Computing Environments for Application Execution

Distributed system infrastructures that exist today (e.g. TeraGrid, Open Science Grid, Unicore, Genesis II) make it much easier than it has been in the past to access a pool of assorted resources and direct them to solve a computational task. However, the process of preparing applications for execution on a varied collection of resources requires considerable preliminary user involvement. The issue is with preparing an application to run on one of a diverse set of resources. Computing systems have unique software and hardware characteristics. To run a particular application, each execution site may need to be configured with dependencies in addition to requiring a platform specific version of an application’s binary. Currently, users most often manually perform the preparatory steps to configure target environments for application execution. To make available resources more efficiently usable by computational scientists, techniques are needed to manage the configuration of target environments automatically. I propose to research this area and develop models that will enable the creation of an automated application configuration framework. I hypothesize that, with the support of limited user involvement, properties of application dependencies can be extracted and exploited to develop automated techniques for the configuration of dependencies on a variety of computing environments in preparation for application execution.

Computing sites can be heterogeneous in many ways. The system hardware can vary in the number of processors per core, the size of memory, or the type of network connection. Systems can use different instruction set architectures and operating systems. Even systems that have the same architecture and operating system can differ in the exact operating system version. Furthermore, systems with the same architecture and operating system version are often uniquely configured with different versions of libraries, packages, and available software. These are the environment details that contribute to determining whether an application will be able to execute at a given computing site.

The process of making heterogeneous environments ready for application execution can be complex and time consuming. It can take many labor-hours to become familiar with a new architecture and resolve application dependency issues. Normally, it is the responsibility of users to ensure that their applications will execute on their target computing environments. The process is most often managed manually, as there is no general purpose tool for determining if required dependencies are met. Performing configurations manually is not scalable. The potentially large activation energy involved in the configuration process becomes a barrier to the efficient utilization of otherwise easily accessible resources such as via distributed infrastructures.

The initial approach to this research has been to divide the problem space into two parts: the description of the application and the configuration of the target environments. The idea is to capture the dependencies without which an application will not execute within an application description. The process to configure these components at different target environments is being modeled. Then transformations will be developed to carry out the configuration process. The ultimate goal of this research is to automate the process of deriving procedures to configure application dependencies on heterogeneous computing environments. The main research contributions will focus around the composition of specialized configuration information for target environments and the transformation process to configure the target environments.