What could GridSolve do for you?

Jon Gibson
HPC Consultant, Manchester Computing, University of Manchester

The NetSolve/GridSolve project is an attempt by the Innovative Computing Laboratory at the University of Tennessee to enable a scientific/engineering community, using its standard programming interfaces and desktop computing environments, to easily and efficiently access distributed hardware and software resources. The GridSolve software, a development from NetSolve, is the middleware necessary to provide this bridge. Built using standard internet protocols, it is available for most variants of UNIX and parts of the system are available for Windows 2000 and XP. It supports a fully heterogeneous environment. Operating a client-agent-server system, it searches for computational resources on a network, chooses the best available, then using retry for fault-tolerance, solves the requested problem and returns the answer to the user. Available resources are load-balanced to ensure their efficient use. The system is versatile enough to allow the integration of any arbitrary piece of software as a resource. For user accessibility, interfaces are currently available in Fortran, C, Matlab, Mathematica and Octave.

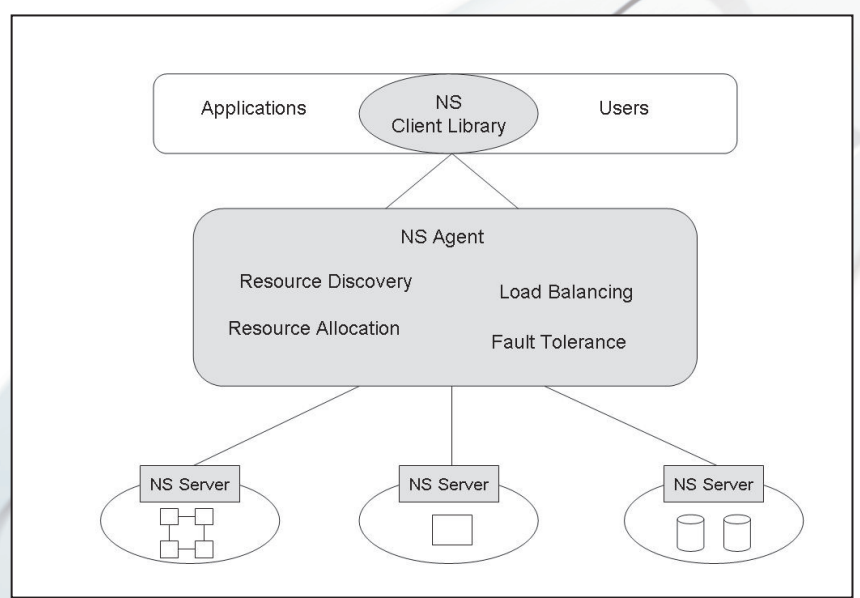

Figure 1: The GridSolve System.

Figure 1 illustrates the architecture of a GridSolve system (with the shaded parts being part of GridSolve). To access GridSolve, the user must link their application with the GridSolve library. This application will make calls to GridSolve's API requesting specific services, so gaining access to remote computing resources without the user having to know what resources or requiring specialist knowledge. The GridSolve agent maintains a database of servers, along with their capabilities (hardware performance and allocated software) and dynamic usage statistics, from which it allocates server resources to client requests, looking for the quickest solution time whilst load balancing the system. The agent keeps track of failed servers. The GridSolve server is a daemon process that awaits client requests. It could potentially be run on anything from a PC to a cluster to a large HPC machine, such as CSAR's Altix or Origins. An important part of the server is a source code generator, which parses a problem description file (PDF), essentially a standard wrapper around the functions being called. Use of these PDF's allows new functionality to be incorporated into the system. Hidden from the user, a given GridSolve request proceeds as follows: the client contacts the agent for a list of capable servers; the client contacts the designated server, sending the input parameters of the request; the server runs the requested routine and returns either the output parameters or an error status to the client.

Obviously, given the limited space, I have only really been able to provide a short introduction; more information about the GridSolve project can be found on their website: http://icl.cs.utk.edu/netsolve. It is likely that future developments in the GridSolve software, such as an interface to the LSF batch queuing system and GSI authentication, will potentially allow GridSolve to be run as a service on CSAR machines. Monitoring usage of resources will, of course, be integral to such as service. If anyone feels they would be particularly interested in using it or has any comments or questions, then please e-mail me at jon.gibson@manchester.ac.uk.