Condor Activities at Liverpool

We have a team of three people (myself - team leader, Ian Smith and Cliff Addison) working here on e-Science activities, of which Condor forms a part. Our interest in it comes from both an earlier (mid-90s) involvement in distributed computing (a JISC/NTI project called far) and a more recent, and pressing, need for increased computational resources.

More recently, we have put in place the software infrastructure necessary to give a single point of access to various high performance computing facilities across different University departments. These comprise several Beowulf clusters plus a pair of Sun V-880 SMP machines. We have followed the, now popular, grid computing paradigm and make heavy use of Globus Toolkit 2.4 as an enabling middleware. The result, dubbed ULGrid (University of Liverpool Grid), is really a demonstration model on which we can test out ideas with researchers and sysadmins from other departments before launching a full production service. ULGrid uses Sun’s Grid Engine software as distributed job manager combined with the Transfer-queue Over Globus middleware from Edinburgh Parallel Computing Centre to provide a uniform interface to users.

The Computing Services Department (CSD) Beowulf cluster has been heavily used almost since day one and its utilization (~ 90%) is now reaching saturation point. We have observed that several applications which run regularly on the cluster are embarrassingly parallel in nature (examples include Monte Carlo analysis and parameter space searches) and therefore have little need for the high performance interconnect provided by it. These applications would seem, however, to be ideally suited to Condor-style execution on the many (> 1000) publicly accessible PCs located around the University.

With this in mind, we looked briefly at products from Entropia and United Devices but they seemed too expensive for an exploratory project. When Condor became available for Windows, its free availability made it look very attractive and we decided to test it by installing it on a group of around forty PCs in a CSD-managed teaching centre.

The teaching centre PCs run Windows 2000 (although we are moving to XP soon) and are linked to Novell file servers which provide disk space for users and act as a central repository for application software. Applications can be installed locally on the hard disk or, more usually, the PC environment (e.g. registry settings) is set up to access the software on the file server. Both methods make extensive use of the Novell Application Launcher (NAL) which can be both difficult and painstaking to set up.

Condor was manually installed on the test machines because of this but it is an issue we will have to address when moving to a production service. The master node is a Sun blade server that can also be considered to be the master node in ULGrid.

The test machines are available to run jobs only outside of normal University hours (0830-1730 weekdays). Since check-pointing is not supported on Windows there is little point in trying to take advantage of the all available idle time (similar to SETI@home). There are also considerable political sensitivities which mean that use must be confined to “after hours”.
The test service seems to be running fairly smoothly at present with just a few select users who have a reasonable grasp of distributed computing. We have kept publicity deliberately “low key” but intend to make the service more widely known once we are confident that the teething problems have been ironed out. Potential users in the near future include chemists (molecular dynamics), materials scientists (molecular dynamics), civil engineers (modelling of hydrodynamics and sediment transport in coastal areas), bioscientists (BLAST searches) and physicists (Monte Carlo simulations).

Our main concerns tend to be fairly Windows specific and include :-

- The inaccessibility of network drives when using the vanilla universe. This means that configuration changes must be copied over to each individual machine and that all application software (DLLs etc) needs to be transferred to the execution hosts since it cannot be accessed centrally.
- Difficulty in running Windows packages (e.g. Spartan) which do not require any user interaction once started but do not have a command line (DOS style) interface.
- Lack of a check-pointing environment under Windows.
- Output files not being returned to the submission machine when Condor terminates a job at the end the “after hours” period meaning that compute time is wasted.
- Accessing condor execution hosts through firewalls (our network people are just about to install Campus firewalls)
- Easy management and deployment of Condor in a Computing Services environment (some issues would be dealt with by access to network drives but there are others).

If our initial experiment is successful we hope to expand the Condor service to include more and more classrooms. We are hoping that subsequent Condor implementations for Windows will increase the number of applications which can be run remotely using Condor. We would also like to incorporate the Condor pools into ULGrid at some stage by making use of its Globus interfaces. This would move us nearer to our long-term goal of providing uniform access to all of our high performance/throughput computing facilities through a web portal.

Steve Morgan & Ian Smith, Computing Services Department, University of Liverpool.