
Bits & Bytes

No. 173

RZG Computer Bulletin

January 2003

Computer Center of the Max Planck Society and the Institute for Plasma Physics*

Status IBM Regatta

The new IBM supercomputer at RZG which was rated the most powerful German system in the Nov. 2002 top 500 list (see also: www.top500.org) will be moved to the new machine hall in January/February. At present time, it is being reconfigured concerning I/O and SP switch connection. All regatta systems (cluster and stand alone nodes) will be integrated in a single "Colony" SP switch based system. The disk capacity of currently 9 TB will be expanded to 25 TB. Additional to the /u and /ptmp file systems a migrating file system will be implemented where files will be automatically migrated to tape after some time.

Inge Weidl

New Linux Clusters

Several Max Planck Institutes have procured Linux clusters to be operated by RZG. A rack-based cluster of the Fritz-Haber-Institute with 32 nodes of double processor systems (Pentium IV @ 2.4 GHz) has been in service since Nov. 2002. An even more compact technology is provided by Blade systems which are currently being installed for MPI for Astrophysics, MPI for Quantum Optics, MPI for Polymer Research, and for RZG. Four Blade centers will be put into operation. Each center consists of 14 nodes (with 2 Pentium IV processors @ 2.4 GHz and 2 GB main memory each) with a Gigabit Ethernet connection via backplane. A detailed hardware description will be given in www.rzg.mpg.de/computing. The overall peak performance of the 176 Pentium IV processors is rated at 845 GFlop/s.

Hermann Lederer

Compiler Suites for Linux

The following compiler suites are currently available for Linux:

1) GCC - GNU Compiler Collection

At RZG the C compiler gcc, the C++ Compiler g++, the Fortran77 compiler g77, and the JAVA compiler gcj are available.

*Max-Planck-Institut für Plasmaphysik, Boltzmannstraße 2, D-85748 Garching bei München, tel.: +49(89) 3299-01, e-mail: benutzerberatung@rzg.mpg.de, URL: <http://www.rzg.mpg.de/>
Editorial: Dr. Roman Hatzky, Tel. -1707

By default, the commands gcc and g77 call the compiler version 3.2.1. In the case the user is dependent on version 2.95.3, the usual commands have to be supplied with the extension -2.95.3.

With the commands mpicc and mpif77 source using the MPI parallel library can be compiled.

gcj calls the java compiler which can produce byte and native code. Swing and AWT support is very limited, though.

For the GNU Compiler Collection, see also: www/docs/languages/GCC.html
gcc.gnu.org

2) Intel Compiler

- Fortran 95

The Intel Fortran Compiler is producing code especially optimized for Intel processors. It can be called by executing f95i or mpif95i in the case of MPI. Both are scripts setting up the environment for the usage of the Intel Fortran suite and execute the compilers for you. These scripts are in your standard path, and so you do not need to bother about location of the compiler or its MPI version. See also: www.rzg.mpg.de/docs/languages/Intel_f95.html

- C/C++

The C/C++ compiler can be called by executing icc. Its MPI version is available with a compiled script named similar to the Fortran script: mpicci
See also: www/docs/languages/Intel_cc.html

3) Lahey/Fujitsu Compiler

The LF95 compiler is especially useful to find runtime errors due to the extensive runtime diagnostics available. The Lahey compiler can be called by executing f95f or mpif95f in the case of MPI. As in the case of the Intel Fortran compiler, scripts are executed which set up the environment and call the compiler. See also: www.rzg.mpg.de/docs/languages/LF_f95.html

4) Sun Java SDK

In order to be able to use Java you will have to expand your search path by adding
`/afs/ipp/@sys/soft/java/j2sdk/bin`

System dependent programs and tools concerning java (e.g. the j2sdk) are available from /afs/ipp/@sys/soft/java, while pure java tools and applications (like ant) can be found under /afs/ipp/@sys/soft/java.

See also: www.rzg.mpg.de/docs/languages/java.html
java.sun.com

Thomas Soddemann
Roman Hatzky

Debugging under Linux

At present, debuggers available on the RZG Linux machines are “gdb” (the line-oriented, general GNU debugger, version 5.0) sources.redhat.com/gdb, “ddd” (a graphical user interface to gdb, version 3.3) www.gnu.org/software/ddd. and “totalview” (5.0-4, 5.0-2, 5.0 and 4.1.0) www.rzg.mpg.de/docs/debugging/totalview.html.

The totalview debugger is the method of choice when it comes to debugging a parallel application. The recent version 6 will be available soon.

Werner Nagel

Batch system

1) Introduction

Since the IBM LoadLeveler (LL) is currently not available for Linux, another batch system had to be chosen. After a brief evaluation of various systems on the “Open Source Market”, we opted for the Sun Grid Engine (with NQS/DQS-like syntax). If you are familiar with the IBM Load Leveler, you may have to adapt to a slightly different definition of a queue:

Grid Engine Queues are entities which make use of the resources of a single computing host.

This definition makes it impossible to have parallel queues in the sense of IBM’s Load Leveler, which include more than one host. On the other hand by using SGE we are able to define so called *Parallel Environments* (PEs) which serve the purpose of a parallel queue quite well. PEs are entities which make use of a collection of local, host bound queues in order to execute their jobs in parallel. This has the advantage that a job can request any number of hosts up to the maximum supported by the PE. Unused resources are still available to others. Let us assume that user1 submits a job using 8 CPUs out of a PE configured with 32 CPUs. Once his job is executed, 24 CPU are still available to all other users, using the same Parallel Environment.

2) Quick Start

1. First, make sure that your password has been encrypted in the MIT format. If unsure, change your password on sp.rzg.mpg.de with

```
# kpasswd
```

2. You need to store your password in an encrypted form on the queue master host with:

```
# save-password
```

This allows the queuing system to obtain an AFS-token for your job, just in case it wants to write something to or read something from the AFS file system (e.g. to your home directory).

3. If you are using csh/tcsh as your shell issue:

```
# source \  
/opt/SGE/$Cluster/common/settings.csh
```

Or, in case you are using bash or ksh:

```
# . /opt/SGE/$Cluster/common/settings.sh
```

Here “\$Cluster” has to be replaced by the name of the cluster. If you use the RZG cluster, e.g., use “RZG” (without the quotes).

4. Write a new submit script, e.g. called submit.sh. Documented templates are available from www.rzg.mpg.de/docs/linux/queue.html

5. Use qsub to submit your job:

```
# qsub submit.sh
```

3) Basic commands

qmon is a graphical user interface for the SGE. It can assemble submit scripts, let’s you watch the queue and load of the machines in the cluster. The advanced user will in general be much quicker by typing the CLI commands.

qsub lets you submit jobs to the queuing system. See also man qsub on the according system for options.

qstat shows the currently queued and executed jobs. It has various options, e.g., **qstat -f** outputs a formatted view, **qstat -F** gives you the resource lists for each queue. Have a look at its man page for more information.

4) Remarks

Additional to the Linux clusters SGE is or will be used for a series of Solaris systems. Links to external resources:

- Grid Engine’s home: www.Sun.com/software/gridware
- Developer’s resources: gridengine.sunsource.net

Thomas Soddemann

Migration of Kerberos Server

On Tuesday, Jan 14th 2003, the migration from Kerberos 4 based authentication to Kerberos 5 took place. Kerberos 5 clients are still supported, although we intend to upgrade to Kerberos 5 clients in the future.

Karl Lehnberger