New utilities and tools for using and monitoring ENEA-GRID environment

Status monitoring and networking

While data are sent and received by means of internet transport protocols like UDP and TCP, a remote user or application can start its processes on a host of the ENEA GRID by using several authentication protocols like rsh, ssh etc. Therefore it is important to test both network performance and authentication procedures. We present some of the developed tools: the rsh test, the UDP test and the TCP/IP bandwidth test. The former is an example of a software which is able to test the efficiency of a command over the full grid, the second measures the rate at which data can be transmitted over the internet, the third measures the TCP/IP bandwidth.

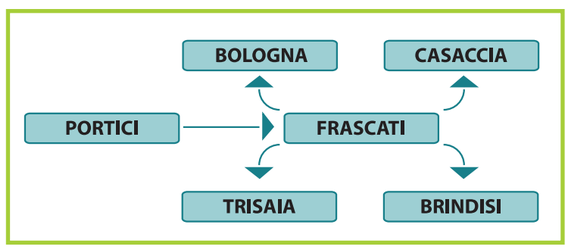

Figura 1.1. From each host it is possibile to test the status of the grid. The connectivity status of the host which one whishes to test with respect to the others can be tested by an application. This can be started independently of the host the user is logged on, by means of the ENEA submission tool. A program can be started on another remote host for which one wishes to test its status with respect to all the machines of the grid. This tool is important, for example, to discover failures of the authentication protocols (like rsh, ssh, etc...) used by applications based on parallel programming.

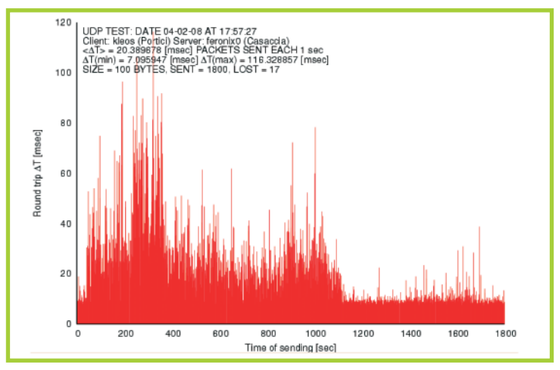

Fig. 1.2. The NRT (Network Response Time) tool has been developed in order to detect how the communication speed between two hosts of ENEA GRID changes over time. The network response time is defined as the round trip time it takes to send a packet of data and receive it back. The test uses the UDP protocol The average round trip time ‹ T› as well as other statistical information are reported.

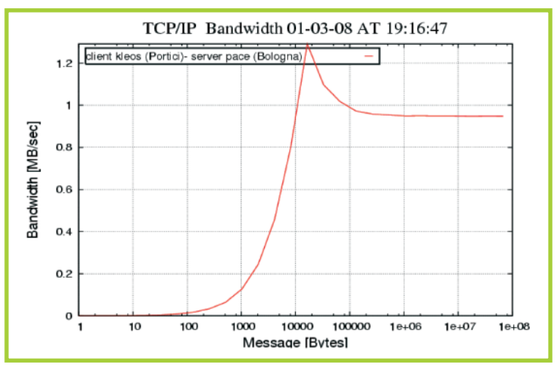

Fig. 1.3. The test measures the network TCP/IP bandwidth between two hosts (clent and server) of ENEA GRID. The user can specify the interval of the band to analyse along with the step size. This tool provides an efficient way to check whether the network is working accordingly to the aspected value of the band saturation point as well as to reveal data flow bottlenecks.

Serial and parallel launchers

Grid computing enables development of large scientific applications using computational resources not available at a single site. Designing and implementing grid application often require to pool together resources that belong to different platforms. ENEA-GRID is a collection of many cluster with distributed and shared memory architectures. Hosts with operating systems like Linux, AIX and SGI IRIX, share a distributed file system (AFS). This infrastructure places some problem due to its heterogeneous environment. Below (see Fig. 2.1 and 2.2) it is showed, as examples, the different kind of compilers (for FORTRAN 77/90/95, C,C++ source) and MPI implementations for parallel programming with Message Passing model.

The problem above outlined is a general issue. Running an application with many binaries or installations, one for each platform, require job lauchers or wrappers. Fig. 2.3 shows the scheme followed by us to hide the complex structure of a given application.

	IBM	PGI	GNU	Silicon	Intel
Fortran 77	xlf	pgf77	f77	f77	ifort
Fortran 90	xlf90, f90	pgf90	f90	f90	ifort
C	xlc	pgcc	cc, gcc	cc, gcc	icc
C++	xlC	pgCC	g++, gcc	g++, gcc	icpc

FiG. 2.1. Note that the same host may have installed more than one compiler.

Available MPI implementation on ENEA-GRID:

1. LAM/MPI:
open source implementation
2. MPICH:
support Linux, AIX, IRIX
3. IBM POE:
native implementation of AIX O.S.
4. OpenMPI:
represents the merger beetween FT-MPI, LAMPI and LAM/MPI and PACX-MPI

Fig. 2.2. These MPI implementations excell in one or more areas, while OpenMPI tries to collect them.

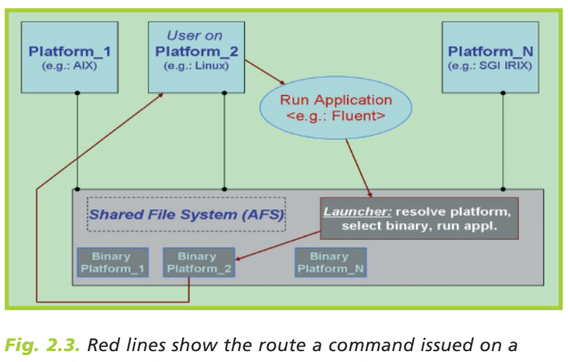

Fig. 2.3. Red lines show the route a command issued on a given platform does in a ENEA-GRID enviroment.

Some References: 1. UNIX Network Programming, Volume 1, Second Edition, Prentice Hall, 1998; UNIX Network Programming, Volume 2, Second Edition:Prentice Hall, 1999 2. http://www.open-mpi.org/; http://www.afs.enea.it/project/eneagrid/Resources/List.html 3. http://www.mathworks.com; http://www.ll.mit.edu/MatlabMPI/H. Kim, J. Mullen, Introduction to Parallel Programming and pMatlab, MIT Lincoln Laboratori. 4. http://www.top500.org; http://www.netlib.org/benchmark/hpl/

Since all ENEA GRID's hosts access to the same file system, we place a job launcher. Its task is to check the host's platform from which application was launched, select the proper binary and run it on the initial host. In a grid environment, jobs should be run under control of resource manager and thus job launchers give to programmers the choice to submit applications in interactive or batch mode. Fig. 2.4 show an example: regardless of serial or parallel jobs, we can integrate many different applications in a single interface to grid's hosts. Moreover such an interface can be implemented

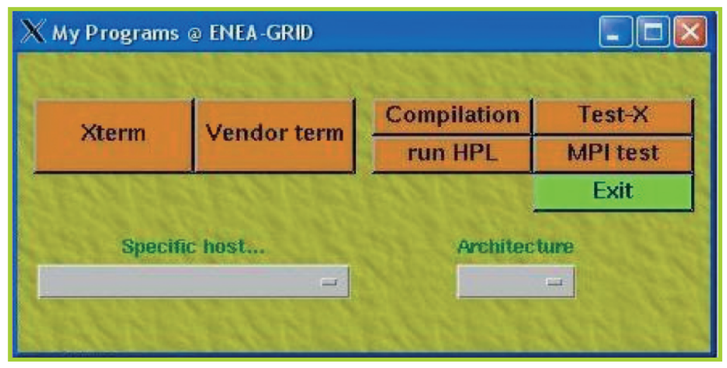

Fig. 2.4. Example of ENEA-GRID interface, Java classes allow to

Parallel programming in Matlab

Matlab (MATrix LABoratory) provides an environment for numeric elaboration and an interpreted programming language widely used in science and technology. The package is equipped with specialized libraries (toolboxes) which can solve problems in several fields. For some years the implementation of parallel programming in Matlab environment has been under study. Mathworks, the producer of Matlab, distributes toolboxes with this target, but there are also third party projects with the same aim. Among them are the "MatlabMPI" and "pMatlab" libraries developed by Lincoln Laboratories of the Massachusetts Institute of technology. ENEA GRID, by the introduction of CRESCO machines, will achieve a very high computational power. So, the possibility of parallel programming in Matlab on this structure becomes a meaningful aim. We have chosen to modify and install the "MatlabMPI" library for Linux machines of ENEA-GRID.

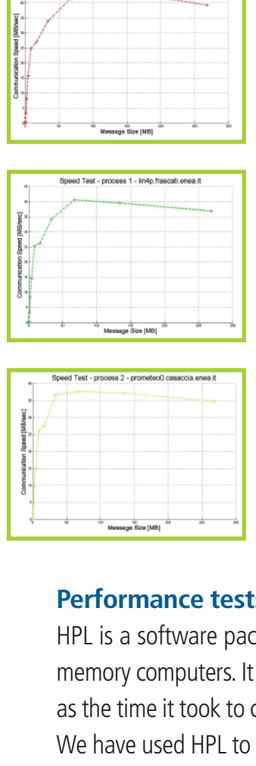

Figura 3.1. Plots show the results of a test made with MatlabMPI library on ENEA-GRID. The involved machines are lin4p, bw305-2 of Frascati and prometeo0 of Casaccia clusters. The test consists of measuring the elapsed time in each "send" and "receive" sequence employed in the exchange of increasing dimension messages among processes. This test does not measure the pure bandwidth because send and re-ceive operations involve file system accesses. We can de-fine it as a test for the communication speed among processes using Matlab-MPI.

MatlabMPI. This functions library, written in pure Matlab language, implements a subset of the functions defined in the MPI standard and allows to run, at the same time, several Matlab processes both on multicore machines and computer clusters. Some code changes were required to install MatlabMPI on ENEA GRID. To enable the launching of Matlab jobs on remote machines, allowing them to access the afs file system without losing reading and writing permissions, or to be more precise to ensure token preservation for the user, we have resorted to the ENEA-GRID submission tool. In particular "lsrun" is now the command by which Matlab processes can be launched on Linux machines of ENEA-GRID. Data exchange among the processes does not happen by real message sending, but via file system access operations. An example of this is the creation of directories and files, that all processes can see, in which are temporarily stored the data to be exchanged.

Performance tests

HPL is a software package that solves a dense linear system in double precision (64 bits) arithmetic on distributed-memory computers. It provides a testing and timing program to quantify the accuracy of the obtained solution as well as the time it took to compute it.

We have used HPL to measure the performance of two kind of system in ENEA-GRID: a Linux cluster 16 nodes with 2 Daul-Core AMD Opteron per node, interconnected via Infiniband (IB) and 1 SMP node Power 4 with 32 CPUs. Fig. 4.1 and Fig 4.2 show the collected data. We report the following main parameters:

1. Memory. It is related to the dimension N of the matrix. Growing values of N allow a better utilization of memory hierarchy, from local to level 1 chache memory. Moreover, since HPL uses BLAS library, the algorithm can use Level 3 BLAS due to the two-dimensional Block-Cyclic Distribution Data.

2. Peak. It is the most significant benchmark's result since it gives the number of floating point operation per second that the system can perform. We report it as a function of memory utilization (or N).

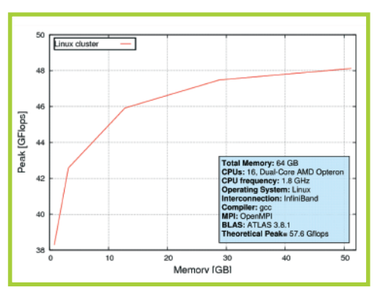

Fig. 4.1. Performance Peak (Gflops) of the Linux cluster. The Peak=48.12 Gflops produce a efficiency of 83.50%.

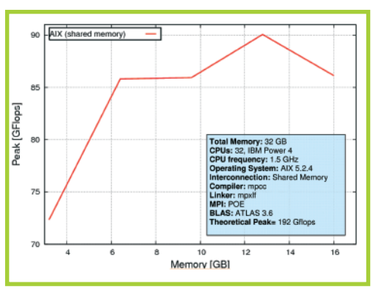

Fig. 4.2. Performance Peak (Gflops) for Aix. The Peak=90.06 Gflops produce a efficiency of about 50%.

Autori: A. Funel, G. Guarnieri, S. Raia, F. Ambrosino, S. Migliori, S. Podda
ENEA - Dipartimento Tecnologie Fisiche e nuovi Materiali
e-mail contact: agostino.funel@portici.enea.it, guido.guarnieri@portici.enea.it, salvatore.raia@portici.enea.it ➤ www.cresco.enea.it