The Sicilian Grid Infrastructure for High Performance Computing

Carmelo Marcello Iacono-Manno, Consorzio COMETA, Italy

Marco Fargetta, Consorzio COMETA, Italy

Roberto Barbera, Consorzio COMETA, Italy, and Università di Catania, Italy

Alberto Falzone, NICE srl, Italy

Giuseppe Andronico, Istituto Nazionale di Fisica Nucleare, Italy

Salvatore Monforte, Istituto Nazionale di Fisica Nucleare, Italy

Annamaria Muoio, Consorzio COMETA, Italy

Riccardo Bruno, Consorzio COMETA, Italy

Pietro Di Primo, Consorzio COMETA, Italy

Salvatore Orlando, Istituto Nazionale di Astro-Fisica, Palermo

Emanuele Leggio, Consorzio COMETA, Italy

Alessandro Lombardo, Consorzio COMETA, Italy

Gianluca Passaro, Consorzio COMETA, Italy

Gianmarco De Francisci-Morales, Consorzio COMETA, Italy, and Università degli Studi di Catania, Catania

Simona Blandino, Consorzio COMETA, Italy, and Università degli Studi di Catania, Catania

ABSTRACT

The conjugation of High Performance Computing (HPC) and Grid paradigm with applications based on commercial software is one among the major challenges of today e-Infrastructures. Several research communities from either industry or academia need to run high parallel applications based on licensed software over hundreds of CPU cores; a satisfactory fulfillment of such requests is one of the keys for the penetration of this computing paradigm into the industry world and sustainability of Grid infrastructures. This problem has been tackled in the context of the PI2S2 project that created a regional e-Infrastructure in Sicily, the first in Italy over a regional area. Present article will describe the features added in order to integrate an HPC facility into the PI2S2 Grid infrastructure, the adoption of the InifiniBand low-latency net connection, the gLite middleware extended to support MPI/MPI2 jobs, the newly developed license server and the specific scheduling policy adopted. Moreover, it will show the results of some relevant use cases belonging to Computer Fluid-Dynamics (Fluent, OpenFOAM), Chemistry (GAMESS), Astro-Physics (Flash) and Bio-Informatics (ClustalW)).

Keywords: *e-Infrastructure, gLite, Grid, MPI, PI2S2, Software Licences*

DOI: 10.4018/jdst.2010090803

INTRODUCTION

The growing demand for parallel programming and High Performance Computing (HPC) poses a question: Grids (Foster et al., 2001) try to maximize the overall infrastructure exploitation instead of the performance of each single application running on them; in fact, the quality policies address the performance of the whole infrastructure over long periods, instead of the performance of each single run. For instance, a typical quality parameter for Grids is the total number of jobs run over a month. Grid users usually have a different point of view and they decide among the various computing solutions (proprietary cluster, buying time on a supercomputer, etc.) having in mind the time performance of their own applications as the most relevant parameter to be evaluated and traded-off with the expensiveness of the candidate solution. Bridging the gap between Grid and HPC may result in a great advantage for Grids as the business of massive parallel applications can bring novel resources and foster infrastructures' sustainability. Some technical aspects of running HPC programs on Grids have been described in a recent article (Orlando et al., 2008). Obviously the hardware equipment is the basic factor driving the application performance. The usual choice during the building of a Grid infrastructure is to have more processors instead of the fastest ones, in order to run more jobs simultaneously. This is one of the differences (probably the most important one) between an HPC cluster dedicated (often tailed) on a single application and a general-purpose Grid infrastructure. Nevertheless, as it happens for many technologies, adaptability and procedure standardization can largely compensate for the use of commercial components and architectures instead of customized ones. For instance, sharing resources allows more processors compared to the average dedicated clusters and this feature may be exploited to enhance the performances for well-scalable applications.

Having in mind the above considerations, the strategic importance of such HPC applications comes from the growing demand about this specific computing paradigm arising from both academic institutions and private enterprises. Small/medium size companies may take advantage of the Grid infrastructures to run HPC programs otherwise too much expensive for either hardware costs or lack of human expertise. Acting as a reliable, standardized and reasonably fast HPC facility, a Grid infrastructure can sensitively enlarge the range of its users. This is easier to happen for a regional Grid whose Virtual Organization (VO) usually gathers all the institutions acting on the same area, resulting in a more versatile and multi-disciplinary community compared to the international VOs that are often devoted to a single discipline. The following sections describe the efforts that the Sicilian Grid infrastructure is producing to fully support to HPC applications. Section 2 briefly describes the Sicilian Grid infrastructure, its characteristics and purposes. Section 3 illustrates the adopted scheduling policy and the newly developed license server. Section 4 treats middleware modifications and general porting procedure. Section 5 outlines some use cases testifying the wideness and variety of impacted fields also reporting about the results of preliminary tests. Finally, section 6 draws some conclusions.

THE PI2S2 PROJECT AND THE SICILIAN GRID INFRASTRUCTURE

The PI2S2 project (Barbera, 2007) aims at providing Sicily with a Virtual Laboratory running a computational Grid for scientific and industrial applications. The COMETA Consortium (Falzone, 2007), is a partnership among the Sicilian Universities of Catania, Messina and Palermo, the National Research Institutes for Nuclear Physics (INFN), Astro-Physics (INAF), Geo-Physics and Volcanogy (INGV) and the SCIRE Consortium. The COMETA Consortium (Barbera, 2006) developed the PI2S2 project and currently manages the infrastructure. The adopted standards rank it at a very high technol-

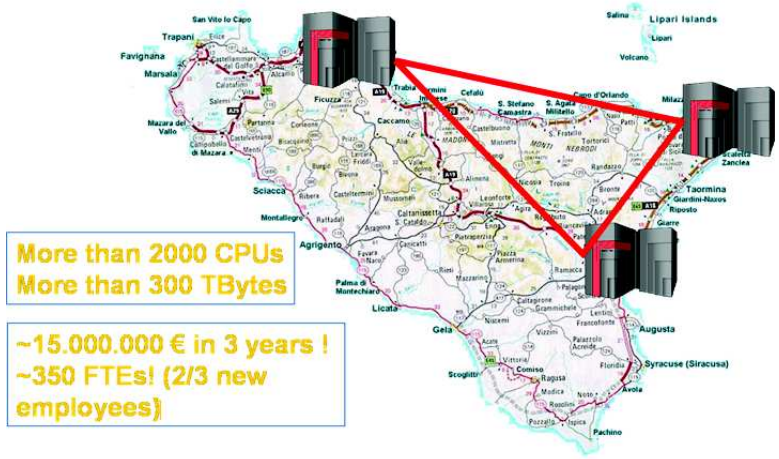
ogy level to become an open, general purpose, on-demand facility for distributed computation and massive data storage (the first one on a regional scale in Italy). The Sicilian infrastructure is connected to the international computational Grids (See Figure 1) in order to improve both the level of scientific collaboration between the Sicilian Universities and Research Institutes and their counterparts in the rest of the world, and enhance the competitiveness of local Small and Medium Enterprises.

The main sites of this infrastructure are located at the Universities of Catania, Messina and Palermo and the INAF and INFN sites in Catania. During 2008, the PI2S2 infrastructure reached the overall amount of 2000 cores and 300 TB of data storage capacity, so becoming one among the most important computing centers of Italy.

The adoption of the InfiniBand low-latency network layer since the very beginning of the project testifies about the consideration towards HPC applications. InfiniBand is a low latency connection designed to improve the communication efficiency over the local networks connecting the Worker Nodes of each site. The peculiarity of this connection is that the optimization focuses on the latency time of little data packets. In fact, in the usual nets, the overall data throughput is the most relevant parameter. This difference is connected with parallel programs that have to exchange very frequent little data burst instead of distanced large packages. Latency on the InfiniBand net drops to a few microseconds, from 10 to 100 times better than the usual value over the Ethernet. The applications need to be compiled and linked to the specific libraries in order to use the InfiniBand layer for their node-to-node communications. Thus a Grid expert usually assists the user of a parallel InfiniBand application in the porting procedure. The InfiniBand net layer is reserved for applications in the sense that no Grid services use it for its communications.

PI2S2 infrastructure currently runs many applications belonging to several disciplines ranging from Astro-Physics to Chemistry, from Nuclear Physics to Bioinformatics. Particularly significant applications are those related to Civil Defense, such as the simulations about volcanic ashes or lava flow paths. Although most applications belong to academia, some

Figure 1. The Sicilian grid infrastructure



Fluid-Dynamic applications aim at the development of industrial products. Thus, the PI2S2 infrastructure is a production facility opened to industry. The adopted middleware is gLite3.1. Many extensions have been added to the standard software in order to better support the user applications. This is particularly true for parallel HPC applications. The availability of a large data storage capacity is another key opportunity that is currently being explored as well as HPC. Genius web portal (Falzone, 2007) designed for non-expert users, gives access to the infrastructure from a wide variety of devices such as personal computers (either desktops or notebooks), palmtops, mobile phones and so on.

Some future projects concern the COMETA infrastructure beyond the end of the PI2S2 project scheduled for February 2009. Among them, a relevant one is about linking the 4 sites located in the Catania Campus by the InfiniBand net layer. The site-to-site distances are within the specifications and the overall amount of available cores will range towards one thousand. The challenging aspect of the project is how to manage this super-site, as each component site must preserve its operational and legal autonomy: a simple merging is not feasible as each site belongs to a different institution. On the other hand, with current middleware, a single parallel application cannot run over more than a site, so merging is necessary unless the middleware is modified. Such a modification addresses a very general issue and cannot be tackled by a single VO, requiring a common effort of all the Grid community at its higher levels, so the idea is to use this super-site only for a few special applications requiring such a large amount of cores. Dedicated connections among the 3 cities are also under consideration, but the target is not related to parallel applications. The dedicated connections are intended to keep faster and more secure the communications amongst the infrastructure's services.

gLite middleware adopts the Message Passing Interface (MPI) libraries (Pacheco, 2008) as the standard communication tool among the cooperating simultaneous processes. The only supported version is MPICH. The updated MPI2 code is supported as well. Next sections describe how MPI applications can run on the Grid infrastructure and some of them have been actually deployed on it. Scheduling, license server and middleware modifications will be the main subjects.

Scheduling Policy and License Server

The coexistence of several heterogeneous jobs running on the same infrastructure is a major difference between Grids and dedicated clusters. The advantage of having more resources is worthy only if they are effectively shared among the various users whose requirements may be very different. HPC jobs may last for several days, so a queuing time up to one or two days may be acceptable as it is far lower than execution time. On the other hand, short jobs lasting up to a few hours are usually privileged in order to start their execution as soon as possible. PI2S2 sites have different queues for jobs of different durations with increasing priority for shorter jobs, but the situation is more complex due to the variety of requirements to be satisfied. For instance, emergency jobs, related to volcanic surveillance and Civil Defence need absolute priority, so these jobs perform pre-emption on the running cores interrupting the execution of short ones. Pre-emption can be very negative for the emptied job in the case the incoming job is very long. Emergency jobs are not very long, so pre-emption is acceptable. HPC jobs need from tens to hundreds cores; the more resources needed the more they have to wait before they are available. According to the usual policy, they starts only if there are no shorter jobs queued and there are enough free resources. This situation can cause long queuing times for MPI jobs when many short jobs compete with them for resource allocation. As the duration of the MPI jobs can be very long, pre-emption is not feasible; the short pre-empted jobs may remain interrupted for days. Currently, the policy adopted for HPC jobs is resource reservation. When scheduled, an HPC

job begins to collect and reserve unloaded cores up to needed amount. This policy is reasonably effective as the number of HPC jobs is far lower compared to the number of short jobs, so the resource turn-over is fast enough to avoid the reserved cores being idle for a sensibly long period. An on-line automatic control, avoiding monopolistic resource allocation, completes the described basic mechanism.

Another important modification of the standard situation is the creation of queues dedicated to single applications. This is due to the need of authorizing a strictly restricted group of users to access the license server or a privileged scheduling policy. For instance, HPC users have a dedicated group into the COMETA VO and their jobs have a special scheduling as above described.

Many HPC programs require a software license. Together with many legal issues, the use of such programs on a distributed infrastructure raises the technical problem of making the license available to all the working nodes. Currently many licenses are either "username-locked", i.e. connected to the user name, or "node-locked", i.e. connected to the physical net address of the executing processor. None of these solutions fits the distributed environment of Grid infrastructures where a possibly unique license must be shared for all the executions that may run on different and geographically distant sites. Thus a license server providing the so-called "floating" licenses allows each user to contact the license server asking for the authorization that is granted up to the stated number of simultaneous users. On job completion, the client packet releases the used license items, keeping them available to another user. Comparing to a cluster sharing a private net, Grid infrastructures add a further difficulty, because licenses must be delivered on a public net to the remote execution sites.

FlexLM is a free software package allowing the management of such "floating" licenses from a single server, although the tool is replicated on other two servers for redundancy reasons. In the usual scheme, the user writes the address and port of the license server on a configuration file or an environment variable to be read by the executable that contacts the server to authorize the run. Up to this point, every user of the Virtual Organization who is able to submit jobs on the infrastructure, can use the licensed software simply by addressing the server license. Site administrators can arrange a first filter by setting the execution permissions on the executable file of the package, but this only works when the file is statically stored on the infrastructure. In fact, VO members may still run their unauthorized software asking for the license to the license server bringing their own executables. To avoid this, the Grid License Manager (GridLM) developed by the Catania Grid support group, associates license granting to the identity of each single user. More precisely, execution permission is granted if the user belongs to an authorized group inside the VO, so the user must be authenticated on the infrastructure also specifying the specific user group. For instance, in order to use a commercial SW ('example_sw') on the COMETA infrastructure, the authentication command becomes:

```
voms-proxy-init --voms cometa:/cometa/
example_sw.
```

Moreover, the user must add a line with the specific tag describing the commercial SW to the Job Description Language (JDL). Also thanks to the encrypted communications, this mechanism is both enough secure and flexible to be used over a public network between the distant sites of the COMETA infrastructure.

Currently the GridLM tool is undergoing a further development in order to associate the license also to the proxies that are delegated by the user credentials when a job is submitted. This will allow the starting of the production step for many applications that are currently being tested. The adoption of robot certificates, i.e. an authentication method linked to the application more than a single user, will also have a positive impact on these applications that are described later on.

MPI Modifications to Glite Middleware

At present time, the standard version of the adopted middleware gLite3.1, only supports the MPICH version of MPI. The parallel applications using other communication systems (LAM, OpenMPI) have to be modified and recompiled using the MPICH libraries. EGEE community has shown interest about the integration of other MPI versions into the middleware.

HPC requires the full exploitation of the worker nodes and communication capabilities. HPC applications are intrinsically more complicated than other job types and reach the highest performances only if all the involved nodes are dedicated to them, so the most important applications require to run on reserved executing nodes. Moreover, many other HPC applications are adapted from the shared memory paradigm, i.e. the execution is optimized for a few multi-core processors with a huge memory available on the same board. This standard is widely diffused in the industry world where workstations with multiprocessor motherboards are normally used to run licensed software for simulation and design purposes. Many of these applications have been adapted to a distributed environment but their communication efficiency drops when the core number is increased to more than a few tens. Even some applications originally designed to run on a distributed environment often show their best performances when they minimize net communication, using all the cores of the same processor. The concentration of job execution on the lowest number of physical processors is a winning strategy in order to enhance software performance. This consideration only fails in a few cases, for two basic reasons: 1) either the application requires more memory than the available amount on a single physical processor, or 2) the communication is so fast that it competes with local processing. The situation 1) is rather probable. In this case, the only advantage of using sparse cores is that the distribution over different processors averages the overall amount of required memory per core with the other jobs that are usually less pretentious. However, the advantage is actually vanished by the increased overload of net communication. The situation 2) is rather uneven. Although on our infrastructure, either of the two may occur, they have not been observed so far, so this "Granularity" issue has been the first major modification to the standard middleware. A new JDL tag has been added to select how many cores of the same physical processor have to be used by the job. Performance improvement resulted in a rough 3 factor for an application that has been built on shared memory systems (GROMACS). Lower figures were detected for other applications. Obviously, performance improvement must be evaluated for each single case, due to its dependency on software implementation especially about the communication over the net.

An MPI (Gropp, 2006) program is a current FORTRAN or C code including some calls to MPI library functions that allow information exchange among the cooperating nodes running the MPI program and the synchronization of the execution flow. Usually a master node starts the processes on the other (slave) nodes by establishing some remote connections. On the Sicilian Grid this procedure is based on the use of Script Secure Shell (SSH) and requires an initial setup for the necessary key exchange.

In order to provide instruments to satisfy the requirements of HPC applications, many patches have been developed for the LCG-2 Resource Broker (RB), Workload Management System (WMS), User Interface (UI) and Computing Element (CE). These patched components support new tag types for MPI flavours in addition to the existent MPICH one:

- MPICH2 for MPI2;
- MVAPICH for M PI with InfiniBand native libraries;
- MVAPICH2 for MPI2 with InfiniBand native libraries.

In the MPI implementations currently deployed on the PI2S2 infrastructure, it is

required that the cooperating nodes running a MPI program are tightly connected each other, in order to ensure enough low latency for the node-to-node communication. Based on this assumption, current middleware does not support geographic distribution of MPI jobs on nodes belonging to different Grid sites, although some studies have been targeting the problem (Turala, 2005). Despite the above limitation, the number of available processor ranges from several tens into the hundreds for many sites, so running a MPI application on a Grid infrastructure leads to a sensitive performance improvement.

Moreover, all PI2S2 Grid sites are equipped with the InfiniBand 4x low-latency network connection, so High Performance Computing (HPC) applications requiring fine grain parallelism run in a proper HW/SW environment.

The usual procedure in order to port a MPI application to the Grid is to recompile it using the Portland Group C/Fortran, gcc or Intel compiler, including the libraries of one among the several supported MPI flavors (MPICH, MPICH2, MVAPICH, MVAPICH2 corresponding to the combination of MPI and MPI2 codes over a GigaBit or InfiniBand communication layer). The extension to the gLite3.1 middleware version essentially consists in a wrapper, able to collect the needed information from the Local Job Scheduler (usually LSF). Two special scripts ("mpi.pre.sh" and "mpi.post.sh") have been introduced to be sourced before and after the execution of the MPI program in order to respectively prepare the execution environment and collect final results. The following code illustrates a modified JDL file:

```
Type = "Job";
JobType = "MPICH";
MPIType = "MPICH_pgi706";
NodeNumber = 12;
Executable = "mergesort-mpi1-pgi";
#Arguments = "12";
StdOutput = "mpi.out";
StdError = "mpi.err";
InputSandbox = {"mergesort-mpi1-pgi","mpi.
pre.sh","mpi.post.sh"};
OutputSandbox = {"mpi.err","mpi.out"};
```

Note the new tags and the pre- and post-processing scripts. It is the user that usually provides these two scripts as they have to match the specific needs of each application. Nevertheless, the middleware modifications greatly simplify the user's work, as the wrapper cares about file copying on the slave nodes and environment settings. An open issue is the execution of pre- and post-processing scripts on the slave nodes. In fact, the mpi.pre.sh and mpi.post.sh only run on the master node. The MPI session on the slave node is opened directly by the user code by the MPI_Init instruction, so it is only after this command that a script can be launched to act in the actual session where the MPI program will run on the slave node. This means that only by inserting a script into the user code, it can set the proper environment on the slave node. This approach is not recommendable because it is far better for debugging to keep application and middleware codes well separated. In fact, in case of error occurrence during job execution, having two clearly distinct software levels is very desirable in order to lead to a fast recovery. The adopted alternative approach is then to statically modify the .bashrc script executed at every start of the remote session which, on its turn, implies that a there must be a dedicated user profile tailed on the single application. Summarily, we create a new dedicated queue, together with a new group users enabled to access it. These new profiles include the configuration script, so when the MPI_Init will start a new session on a slave node for these novel users, it will run the configuration script. This technique is rather complicated, but it is feasible provided that only a very few applications do require it. A similar technique is adopted when the job requires a specific library that cannot be installed statically as it is incompatible with a part of the middleware. If the user still wants to run the job renouncing to part of the middleware, a new dedicate queue with a particular user profile allows a dynamic installation of the desired library without any permanent interferences with the installed middleware.

The described approach is the standard one when the application is totally open and modifiable. Nevertheless, other approaches are preferable for standard software packages where any modifications or even recompilation is difficult for technical and/or legal reasons. A common situation happens when the MPI package to be gridified, has its own wrapper or it is closed, in the sense that the source code is not available for modification. Portability is nevertheless possible, but the usual submission mechanism has to be by-passed. What we do then, is to submit a test MPI application that reserves the same number of cores required by the "actual" MPI application and proofs the status of the infrastructure. After the test code terminates its execution, the mpi.post.sh launches the "actual" MPI application. On one hand, running a test application immediately before the production run is very useful in case of failure in order to diagnose whether the error is due to the infrastructure or the application. On the other hand, by-passing the common submission mechanism requires a great caution by the user, as he/she might be tempted to use more resources than allowed, with no communication to the middleware. The result could be that a worker node runs more process than the middleware is aware of. In order to avoid such a security degradation, this submission mechanism is only allowed to a few selected users, qualified as HPC users. They use a dedicated queue with a peculiar scheduling policy as described before. Common users cannot reserve more than a core per single job, as opening new sessions on other cores is prohibited to them.

The complexity of MPI jobs pushed the development of other software tools that are not specifically connected to MPI, i.e. that can be adopted for any kinds of jobs. For instance, long durations of MPI jobs require a constant monitoring of the job evolution. The gLite3.1 middleware offers the Perusal job technique, i.e. during the job execution, some selected files are copied at regular intervals from the working directory to another one where the user can inspect them. As the request of running long duration and MPI jobs started well before the adoption of this middleware version, we developed a similar "watchdog" (Bruno, 2008) mechanism since the time when the PI2S2 infrastructure ran with the LCG middleware. A special version of the mpi.pre.sh has been introduced in order to include this important monitoring activity. By modifying the supplied open script code, the user can adapt the monitoring tool to the specific needs of his/her application, gaining a greater flexibility compared to the perusal file technique. This is the reason why this approach has survived to the adoption of gLite3.1. Another tool has been recently added to further increase the level of control over job execution. This "VisualGrid" tool (Iacono-Manno, 2009) allows the user to save the image files produced during the job execution and create a quasi-live video that can be used for either job execution monitoring or demonstration purpose. Finally, the lack of recursive commands in the standard middleware for file up/download to/from the catalogue, was detected during the development of the scripts for an MPI applications. This led to the extension of the standard middleware commands as described on the PI2S2 wiki (Bruno, 2007).

Similarly to all the applications to be "gridified", i.e. modified in order to run on a Grid infrastructure, MPI applications require the definition of a strategy ("computational schema") in order to fully exploit the Grid's opportunities. For instance, the way data are moved from/to their storage devices, greatly impacts the overall performance. This is particularly true for MPI programs that often have the more strict time constraints amongst Grid programs, being very similar to real-time applications as for some Civil Defence simulations. The usual choice for MPI applications, that are intrinsically more complex compared to the average of other Grid programs, is to install them statically the SW package, having them available on each node of the infrastructure as a shared part of the file system. Thus the execution time is reduced, as the code is immediately available for execution with no need for long data transfers; on the other hand, some work overhead is paid in case of SW

updating, due to security reasons that prevent common user to access sensitive areas of the file systems and require the intervention of a Software Manager.

The following use cases illustrate a wide range of different HPC situations and the related solutions.

Use Cases

FLASH (Fryxel et al., 2000) is a 3D astrophysical hydrodynamic code for supercomputers; it is widely used in current astrophysical research due to its modularity that allows to easily add more physics, more complex meshes and customized solvers. Production started during the last September and the number of the submitted jobs (each usually a few days long) have gone into the hundreds. Figure 2 shows performance enhancement vs. the number of cores and the diagram indicates a good behavior up to 64 cores.

FLUENT (ANSYS, 2008) is a commercial package currently used in Computer Fluid Dynamics (CFD) mainly for flow modeling and heat and mass transfer simulations. This application has been deployed and tested on the various sites of the infrastructure and it begun its production since late 2008. This application shows a good scalability with an increasing number of cores as reported by the following Figure 2.

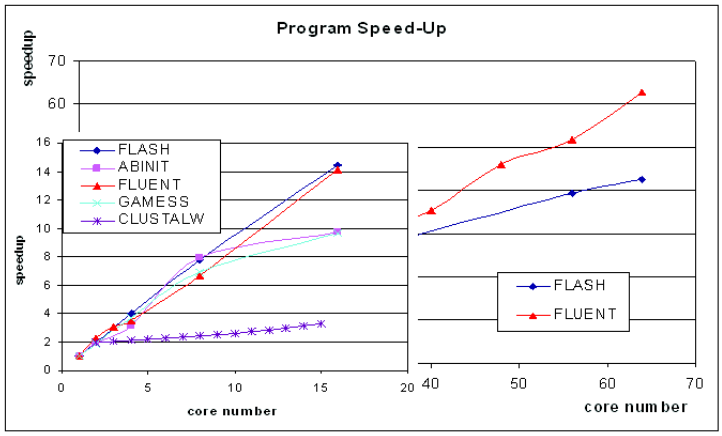
ABINIT (Gonze, 2009) is a package whose main program allows one to find the total energy, charge density and electronic structure of systems made of electrons and nuclei (molecules and periodic solids) within Density Functional Theory. It has been tested on the PI2S2 infrastructure on up to a few tens of cores.

GAMESS (Gordon, 2009) is a program for ab-initio molecular quantum chemistry. GAMESS can compute molecular wavefunctions including many effects and corrections. It has been running on the Sicilian Grid Infrastructure since 2007.

ClustalW-MPI (Lombardo, 2008) is a parallel implementation of ClustalW, a general purpose multiple sequence alignment program for DNA and proteins; it produces biologically meaningful multiple sequence alignments of divergent sequences. It has been implemented on the Sicilian Grid as a part of the ViralPack package (Lombardo, 2008), a Genius-integrated application for virology studies by the PI2S2 infrastructure.

The diagrams show a good speed-up improvement for all the tested applications. However only two of them (FLASH, FLUENT)

Figure 2. Speed-up vs. number of cores



have been extensively proofed up to a massive amount of cores as they are specifically built to run on many cores and have been tested on various infrastructures. Recent tests for FLUENT on the PI2S2 infrastructure have reached 176 cores. Other programs (GAMESS, ABINIT) have been deployed on a limited number of cores because they did not scale well on a high number of cores or simply because users were satisfied by the reached performance. The close-up of Figure 2 shows the behaviors of the above applications with an increasing number of cores.

The collected results have also been used to compare Grid to other clusters' efficiency. Figure 3 shows the comparison between the PI2S2 clusters and other HPC platforms. Each diagram referring to a program (FLASH or FLUENT) is directly comparable with the other similar ones. Results show that the performance of the PI2S2 infrastructure is at least comparable to other clusters (CLX, IBMSPC5, EROI) with a similar hardware and number of cores. In the FLUENT case, the availability of more processors is effective in compensating the performance gap between the top-level processors of dedicated cluster and shared Grid infrastructure facility.

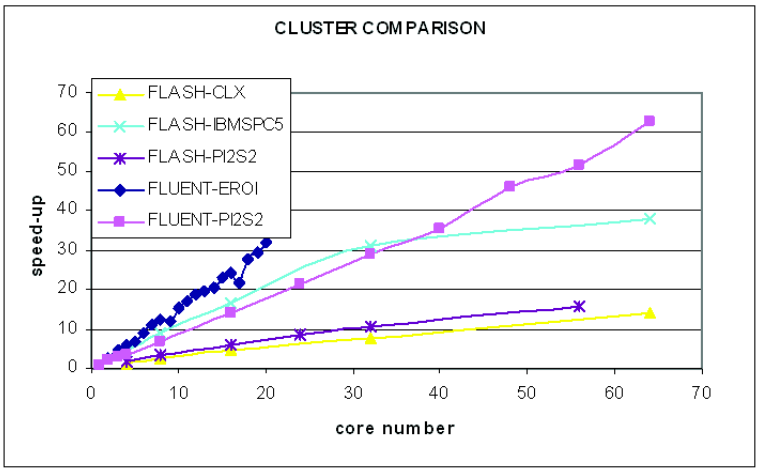
As a description of the work-in-progress we briefly report about OpenFOAM (OpenCFD, 2009), that is an open-source simulation environment widely used in CFD as an equation solver. Differently from other tools, the OpenFOAM user can manipulate the code and even write his/her own solvers. OpenFOAM has been the most difficult porting case ever, due to the peculiar directory structure and the need for appropriate script sourcing on all the nodes. Currently, the 64-bit OpenFOAM-1.4.1 version runs on the Gigabit net layer on up to 16 cores. Porting to InfiniBand is still under development.

CONCLUSION AND FUTURE DEVELOPMENTS

Grid is being used as an HPC environment, despite of initial difficulties, running several MPI programs on the Sicilian Grid infrastructure. Many applications are performing their advanced tests and some of them have started their production with encouraging results.

The increasing demand for MPI applications is pushing forward a reorganization of the middleware in order to offer a wider and tighter support to this important class of applications.

Figure 3. Workstation speed-up comparison



New wrappers have been added in order to match the combination of different network layers (TCP/IP, InfiniBand), compilers and MPI flavours. A specific scheduling policy has been chosen to fulfill the requirements of the heterogeneous jobs running on the infrastructure. A license server has been developed for the PI2S2 distributed environment. Other software tools, normally extension of the standard middleware commands, have been developed. They have a general impact on the use of the infrastructure in the sense that they can be used by all the jobs. The new approach for the integration of MPI jobs has been also developed in order to increase SW modularity and ease both maintenance and future developments giving a further enhancement to the infrastructure efficiency.

ACKNOWLEDGMENT

The PI2S2 Project has been funded by the Italian Minister of Research and University.

REFERENCES

ANSYS Co. (2008). *Fluent.* Retrieved on February, 4th, 2009 from http://www.fluent.com/

Barbera, R. (2006). *COnsorzio Multi-Ente per la promozione e l'adozione di Tecnologie di calcolo Avanzato.* Retrieved January 16th, 2009, from http://www.consorzio-cometa.it/

Barbera, R. (2007). *Progetto per l'Implementazione e lo Sviluppo di una e-Infrastruttura in Sicilia basata sul paradigma Grid.* Retrieved January 16th, 2009, from http://www.pi2s2.it/

Bruno, R. (2007). Recursive Catalog Interaction with lcg-rec-* Tools. *PI2S2 Wiki Pages*. Retrieved on January, 29th,2009, from https://grid.ct.infn.it/twiki/bin/view/PI2S2/CatalogUpDownload

Bruno, R. (2008). The WatchDog Utility. *PI2S2 Wiki Pages*. Retrieved on January, 29th,2009, from https://grid.ct.infn.it/twiki/bin/view/PI2S2/WatchdogUtility

Falzone, A. (2007). *Grid Enabled web eNvironment for site Independent User job Submission.* Retrieved January 16th, 2009, from https://genius.ct.infn.it

Foster, I., Kesselmann, C., & Tuecke, S. (2001). The Anatomy of the Grid . *The International Journal of Supercomputer Applications*, *15*(3), 200–222. doi:10.1177/109434200101500302

Fryxell, B., Olson, K., Ricker, P., Timmes, F. X., Zingale, M., & Lamb, D. Q. (2000). FLASH: An Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes. *The Astrophysical Journal. Supplement Series*, *131*(1), 273–334. doi:10.1086/317361

Gonze, X., et al., (2009). *Abinit*. Retrieved on February, 4th, 2009, from http://www.abinit.org/

Gordon, M. (2009). *GAMESS*. Retrieved on February, 4th, 2009, from http://www.msg.chem.iastate.edu/GAMESS/

Gropp, W. (2006). *The Message Passing Interface (MPI) standard*. Retrieved January 27th, 2009, from http://www.mcs.anl.gov/research/projects/mpi/

Iacono-Manno, C. M. (2009). The VisualGrid Tool. *PI2S2 Wiki Pages*. Retrieved on February 2nd, 2009 from https://grid.ct.infn.it/twiki/bin/view/PI2S2/VisualGrid

Li, K. B. (2003). Multiple ClustalW-MPI: ClustalW analysis using distributed and parallel computing. [Oxford: Oxford University Press.]. *Bioinformatics (Oxford, England)*, *19*(12), 1585–1586. doi:10.1093/bioinformatics/btg192

Lombardo, A., Muoio, A., Iacono-Manno, C. M., Lanzalone, G., & Barbera, R. (2008). *ViralPack*. Retrieved on February, 4th, 2009, from http://documents.ct.infn.it/record/170/files/posternapoli.pdf?version=1

OpenCFD. (2009). *The Open Source CFD Toolbox*. Retrieved on February, 4th, 2009, from http://www.opencfd.co.uk/openfoam/

Orlando, S. Peres., G., Reale, F., Bocchino, F., & Sacco, G. (2008), High Performance Computing on the COMETA Grid Infrastructure, In *Proceedings of the Grid Open Days at the University of Palermo* (pp.181-185), Catania: Consorzio COMETA

Pacheco, P. (2008). *Parallel Programming with MPI*. Retrieved January 27th, 2009, from http://www.cs.usfca.edu/mpi/

Turala, M. (2005). *The CrossGrid Project.* Retrieved on January, 28th, 2009, from http://www.eu-crossgrid.org/project.htm

Carmelo Marcello Iacono-Manno, after cum laude physics graduation during 1990, worked at the Laboratorio Nazionale del Sud, Istituto Nazionale di Fisica Nucleare (INFN), on Data Acquisition and Control Systems developing sequential and parallel C codes for the readout programs and electronic device control. He also developed an off-line application for off-line data analysis based on a neural network simulation. After PhD graduation in 2003, he joined the Grid team at the INFN Catania Section, collaborating to the Trigrid and PI2S2 projects as an application supporter focusing on parallel and massive calculus. He helped porting many applications onto the Grid, covering various fields such as Computer Fluid-Dynamics (Fluent, OpenFOAM), Chemistry (GROMACS, GAMESS), Nuclear Physics (CoMD-II, ISOSPIN). He also contributed to the development of tools for middleware extension (VisualGrid, watchdog, recursive catalogue command) and cared about some of the editorial activities of the Consorzio COMETA (Proceedings of the Grid Open Days at the University of Palermo).

Marco Fargetta graduated in Computer Engineering at the University of Catania in 2002 with a thesis on a new Java extension for Computational Reflection. Since 2006 he holds a Ph.D. in Computer Engineering from the same University with a thesis titled A Model for Automatically Supporting Advanced Reservation, Allocation and Pricing in a Grid Environment. *Afterwards he has worked in Grid projects at national (i.e. TriGrid VL and PI2S2 founded by the Sicily Region) and international (i.e. ICEAGE founded by the EU) level. Actually he is working at INFN (Istituto Nazionale di Fisica Nucleare) in the context of the EU founded EUAsiaGrid Project.*

Roberto Barbera was born in Catania (Italy) in October 1963. He graduated in physics cum laude at the University of Catania in 1986 and since 1990 he holds a PhD in physics from the same University. Since 2005 he is associate professor at the Department of Physics and Astronomy of the Catania University. Since his graduation his main research activity has been done in the realm of experimental nuclear and particle physics. He has been involved in many experiments in France, Russia, United States and Sweden to study nuclear matter properties in heavy ion collisions at intermediate energies. He is author of more than 100 scientific papers published on international journals and more than 150 proceedings of international conferences. He is also referee of prestigious reviews in the field of Grid computing such as Journal of Grid Computing *and* Future Generation Computer Systems. *Since 1997 he is involved in the NA57 Experiment at CERN SPS and in the ALICE Experiment at CERN LHC. Within ALICE, he has been the coordinator of the Off-line software of the Inner Tracking System detector and member of the Off-line Board. Since 1999 he is interested in Grid computing. He is a member of the Executive Board of the Italian INFN Grid Project (grid.infn.it). At Italian level, he is the Chief Technical Officer of the Consorzio COMETA and the Director of two big Grid Projects (TriGrid VL and PI2S2) funded by the Sicilian Regional Government and by the Ministry of University and Research, respectively. At European level, he is the Technical Coordinator of the EC funded EELA-2 Project and has several responsibilities in other EC Grid Projects such as EGEE-III, EUAsiaGrid, and EU-IndiaGrid. Since 2002 he is the responsible of the GENIUS grid portal project and, in 2004, he created the international GILDA grid infrastructure for training and dissemination that he coordinates since the beginning. Since the birth of GILDA he has been the organizer of/teacher in more than 350 Grid training events (tutorials, schools, and even university courses) around the world.*

Alberto Falzone is an IT consultant for NICE srl, Cortanze (AT) Italy, since June 2000. An exclusively knowledge base about distributed computing based on LSF tool, which NICE is vendor

since 1996, and JOB management, was built during the customers assistance activity in the first period. He takes care the scientific customers mainly, as INFN Istituto Nazionale di Fisica Nucleare, ENEA (the Italian Environment and Energy Agency), CRS4 in Cagliari and others, with support activities for their clusters in LAN or WAN. Teaching activities about LSF was held in SNS (Scuola Normale Superiore) in Pisa for customer's system administrators. Support in R&D about EnginFrame as web portal infrastructure, developed from NICE srl, he has trusted within the same scientific context. Since January 2001 he started to develop the GENIUS portal, as NICE partner, powered by EnginFrame, working in collaboration with Roberto Barbera, as INFN partner. He does maintain the supervision of GENIUS Portal development planned for HEP experiments, with significative results for CMS in INFN Padova site, Italy. Since october 2002 he started a mutual collaboration with Science Academy of Prague, Intitute of Physics, Czech Republic, where GENIUS has been installed. Since 2006 he supported the installation of all sites in COMETA Consortium, taking care about HPC layers on every cluster and specific customizations to enable HPC features within gLite environment on the PI2S2 insfrastructure. Other research interest areas, as IT consultant, are security infrastructures development, firewalling and internet routing, bioinformatics.

Giuseppe Andronico was born in Catania (Italy) in January 1965. He graduated in Physics "cum laude" at the University of Catania in 1991 and since 1995 he holds a Ph. D. in Physics from the same University. Since March 2001 he is technologist at the INFN Sezione di Catania. Since his graduation his main research activity has been done in the realm of theoretical physics. He has been involved in lattice field theory simulations. Since late 1999 he has been interested in grid computing participating to several initiatives: European DataGRID, InfnGrid, EGEE. In these initiatives he has been involved in developing code, in operations, in training and in dissemination activities. More recently he has been involved in some European funded projects: in EELA and EUMEDGRID he has been involved as WP manager, in EUChinaGRID he has been involved as Technical Manager, in EGEE-II and then in EGEE-III.

Salvatore Monforte graduated in computer sciences at the University of Catania in 1991 and got his PhD in electronic engineering, computer science and telecommunications in 2002. From 1999 to 2000 he has been working within the Ipp-hurray Research Group of the Polytechnic Institute of Porto School of Engineering (Isep-ipp) studying the Worst Case Response Time scheduling in R-Fieldbues real-time systems. Since 2000 has been collaborating within the National Institute of Nuclear Physics (INFN) as IT specialist for "DataGRID", EGEE, EGEE-II, EGEE-III EU projects as member of Workload Management System Package WP1/JRA1 group, developing gLite EGEE Grid middleware since 2000. He has a more than 5 years experience in programming and Web programming languages, grid computing, computers networks. He is author of more than 20 papers published on international peer-reviewed journals.

Annamaria Muoio was born in Cava de' Tirreni on 15 august of 1971. In 1998 she graduated in nuclear physics. She joined the TriGrid and PI2S2 projects in 2005, porting applications on the Sicilian Grid; in particular, she focused on multi-disciplinary scientific and industrial applications accessible via a web gateway (Genius). She implemented applications for the CHIMERA multidetector, the ISOSPIN experiment, the CoMD-II (Constrained Molecular Dynamics) model the ViralPack project, an industrial project carried out for the Clean Room, for pollution calculation. Dr. Muoio also taught during many tutorials both in Italy and abroad and participated as a Grid expert to various schools.

Riccardo Bruno was born in Catania the 3rd December 1969 and graduated in computer science in May 1999, he started to work as researcher for the University of Catania from April 1999. In April 2000 he joined the LHS Company as software consultant for the BSCS billing system sold to telephony mobile companies belonging to the EMEA countries. He joined the INFN in Catania the 1st January 2006 as responsible of the activity WP 5.2 (Dissemination of Advanced Knowledge Activities) of the EUMEDGRID project, finished on February 2008. From March 2008 he is involved as responsible of the activity NA2.3 (Training) of the EELA-2 project.

Pietro Di Primo was born in Catania during 1972. He scored a master's degree as a telecommunication engineer at the Catania University during 2004 and a higher level master on ICT during 2005. He worked as an ICT consultant for ACSE Co. in Milan. Since 2007 he joined the Grid team at the Catania Section of the National Institute for Nuclear Physics.

Salvatore Orlando is research astronomer at the INAF-Osservatorio Astronomico di Palermo since the end of 1999. He graduated in Physics "cum Laude" at the University of Palermo in 1993 and since 1997 he holds a PhD in physics from the same University. In 1995, he was research fellow at the Dept. of Astronomy and Astrophysics, the University of Chicago (USA), and in 1997-1999, research fellow at the European Space Agency (ESA) - Solar System Division (Noordwijk, The Netherlands). Since his graduation his main research activity has been performed in the realm of optically thin astrophysical plasmas, more specifically solar and stellar coronae, supernova remnants. He has developed and applied hydrodynamic and magnetohydrodynamic parallel numerical models on High Performance Computing (HPC) systems, and Grid/HPC infrastructures. He is author of about 50 scientific articles on refereed international journals (first author of about 20), and of several invited presentations at international meetings. He is scientific coordinator of PHOENIX, an European program for the transfer of knowledge on young stellar objects (2006-2010), and member of the scientific and technical board in the framework of the COMETA-CILEA agreement for HPC and Grid technologies.

Emanuele Leggio was born in Ragusa, Italy in 1978. He receveid the Master Degree in mechanical engineering from Catania University in 2007. He worked for Consortium SCIRE, Numidia s.r.l., Honor Center of Italian Universities (H2CU) in the field of Computational Fluid Dynamics (CFD) and GRID computing. On 2008 he joined the GRID team at INFN Catania section developing CFD modelling of Marmore Waterfalls and developed Fluent-GRID environment. He is from November 2008 a PhD student in innovative technologies for sustainable mobility at the University of Rome "Tor Vergata".

Alessandro Lombardo is a molecular biologist with a PhD in phytosanitary technologies on biotechnological applications in plant pathology at the University of Catania. He is member of the steering committee of the italian technological platform "IT-Plants for the future", member of the scientific committee, challenge conversion, of the italian technological platform "Biofuels Italy". Since 2005 he is responsible of the laboratory of biotechnology, genomic analysis and varietal correspondence (Science and Technology Park of Sicily) and since 2008 responsible of the laboratory of phytopathology (Science and Technology Park of Sicily). He was involved in several Interregional, national and regional projects. He is author of about 25 scientific papers published in international refereed journals and presented in national and internationalcongress and 1 patent. His main research areas are the development of expression vectors for plants,

yeasts and bacteria, bioinformatic applied on plant viruses, variety correspondence, molecular characterization of plants, yeasts and viruses.

Gianluca Passaro has been working as a technician for the Consorzio COMETA since 2005. He deals with the configuration and maintenance of the net and various services (mail, virtual machines). He also supports the deployment of the applications focusing on HPC.

Gianmarco De Francisci Morales was born in Caltagirone (CT), Italy on July 7th 1983. He received both his Bachelor's and Master's degree in Computer Engineering (cum laude) from the University of Catania respectively in 2004 and 2008. From 2007 he has been working for COMETA Consortium on Grid Computing. He is currently a PhD student in computer science and engineering at IMT Institute for Advanced Studies Lucca.

Simona Blandino was born in Ragusa (RG), Italy on July 15th 1982. She received both Bachelor's and Master's degree (cum laude) in Computer Engineering from the University of Catania respectively in 2004 and 2008. From 2007 she has been working for COMETA Consortium on Grid Computing.