Building a computational cluster managed by ARC NorduGrid

Alexey N. Makarov (*MakarovAlexey@gmail.com)*
Scientific adviser: Margarita M. Stepanova (*mstep@mms.nw.ru)*
Saint-Petersburg State University Department of Physics
March – 2008 - Saint-Petersburg - JASS2008

1. Introduction

In the modern world many problems demand complex calculations, processing of great data files, in many areas of sciences there are problems which demand increasing capacity of computers. For these purposes Grids in which units there are computing clusters, uniting resources of different types (computing, program, data storages) are created.

Coordinated distribution of computing resources geographically located in different places, allows solving raised complexity tasks for a wide range of users.

One of the main tasks of the Grid is maintenance simple access to resources. To reach that, users are united in the virtual organizations. Virtual organization is a group of people incorporated to some general attribute, for example there can be researchers in the field of high energy physics who need to count theoretical models and to process the data received during experiments.

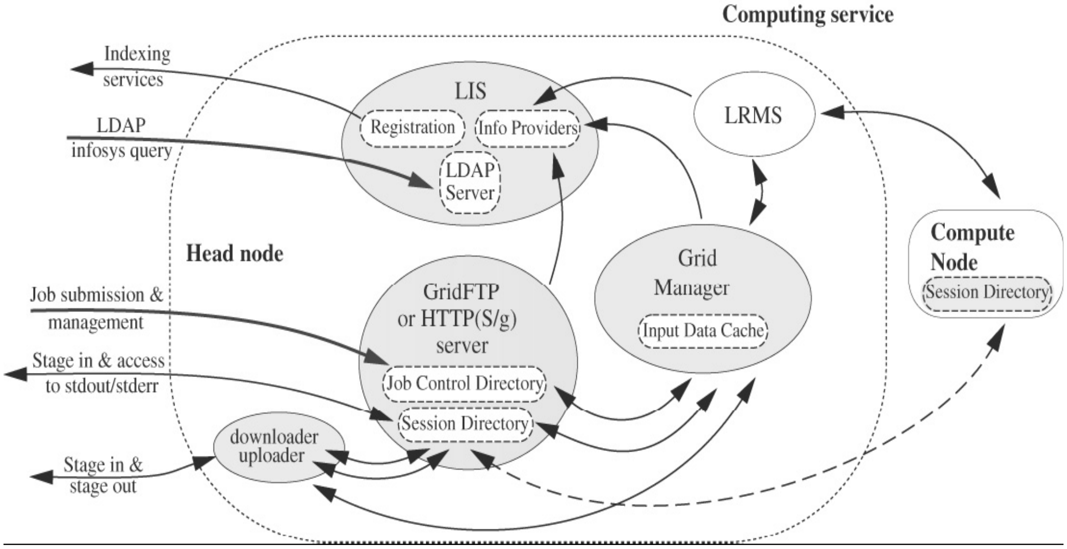
Realizations of Grid-systems can be wide different; nowadays there are tens of them. In the given work one of them, project named Advanced Resource Connector (ARC) NorduGrid [1] is considered. The choice of ARC is caused by high stability and support, simple integration with cluster systems and applications. Also, due to scaled dynamic information system and architecture of the broker, ARC NorduGrid constantly supports data in an actual condition and has a high resistance against overloads and failures.

To start working in NorduGrid it is necessary for user to have only the certificate, entitling to use those or other resources, and also to describe resources required for work, physical (quantity of processors, an empty place on a disk, etc.) and program - installed software (compilers, working environments). Other is carried out by the program - the broker, without intervention from the user.

An aim of this work is adjustment and start of the cluster system connected to ARC NorduGrid. Installation of tasks of users should be carried out through system of queues. Computing cluster should support technologies MPI2 [2], OpenMP [3].

2. Architecture ARC NorduGrid

Architecture of the ARC NorduGrid can be presented as combination of a server part which consists of three basic services (Grid Manager [4], GridFTP [4], Local Information Service [5]) and a client part [6] which is installed on the client side.



Pic.1. ARC NorduGrid services. Arrows show information and data transferred between components. Picture is downloaded from a site http://www.nordugrid.org/pictures.html

ARC gives two ways of work with data storages: standard - based on GridFTP and Smart Storage Element [7] based on Web services.

Computing resources and resources of data storages are connected to Grid through registration service which registers them in Information Index Service, thus, aggregating resources.

2.1. Grid Manager

Grid Manager (GM) is a central ARC service, it is responsible for starting and executing user tasks on cluster, saving of results and providing access to them.

In ARC task is defined as a set of input files, main executed file, requested resources (including installed software) and a set of output files. For each task during its preparation for executing, the separate catalogue - the working catalogue is created. Task will use this catalogue for all input, output, and temporary files as well.

GM processes the description of a task and provides reception of files necessary for job executing (input files). Every access to data during the work of the program, falling out from the range of predetermined input files, should be carried out by the program.

GM has no its own network protocol, for data transmission GridFTP is used. After all required files are loaded and a task is ready for executing, GM incorporates with the local resource manager to start the main executed file.

GM also supports Runtime Environments (RTE). RTE [8] provides access to installed software. The inquiry RTE in the description of a task means that the task will be carried out in an environment with required utilities, libraries, variable environments, etc.

2.2. GridFTP server

GridFTP gives users, and other elements of ARC NorduGrid an opportunity of downloading and uploading of data. In this case the modified version of protocol FTP - Grid Secure Infrastructure FTP (GSIFTP) is used.

The access through GridFTP to every part of the information is protected and flexible control of access the task data is possible. By default task information is accessible only to the client who has signed it on execution. Access rights can be changed through Grid Access Control List [9] (GACL).

GACL uses the language based on XML and intended for the description of access rights for files and catalogues. It allows defining access rights for files from the working catalogue and for the files in data storages. Each catalogue should contain a file with .gacl extension in which the rights of users (<read/> <list/>) are described. This file is created by default, but the owner (user) can change it. Files with .acl extension are used for the control of access to separate files. Addition of GACL rules to the description of a task of the user allows viewing data or even changing it to other users.

2.3. Smart Storage Element

GridFtp is not a unique opportunity of access to data. In ARC NorduGrid service SSE (Smart Storage Element) possessing more flexible functionality is considered.

SSE is a service of preservation data which gives a basic set of options for data management without user intervention. Its main purpose is formation of data storage infrastructure together with replica catalogues, such as Replica Location Service [10] (RLS).

RLS service allows users to load the files into the Grid, without any caring about a concrete place and an individual name of a file. The file will be loaded to one of data storage elements. During this process the unique name will be appropriated to the users file and a record in database RLS will be created. This database represents the list of conformity of a virtual file name with a physical one.

The physical file name is a name under which the file will be stored on resource, and the virtual one is a name under which the user loaded the file. Thus, the user should not know a physical file name, and it is enough to specify its virtual name.

2.4. Information system

ARC developers take care about creation of scaled dynamic system which reflects information about resources condition in real time (a problem of displaying the condition of several hundreds of tasks which are carried out in different parts of the world is more complicated). Information system is not an exclusive research, it is based on protocol LDAP [11] (Lightweight Data Access Protocol) and is realized by set of connected resources lists (services of indexation) and local databases LDAP.

The information system consists of three main components:

- Local Information Services (LIS)
- Index Services (IS)
- Registration Processes (RP)

LIS are responsible for description of specific system's resources, where resources are the description of available processors, data storages, information about tasks and their status. LIS is the specially created and filled database LDAP containing dynamically updated information on resource status.

IS are used for maintaining dynamic list of available resources and contain contact information of the registered objects (for example, URL address of database LDAP or URL of other indexation service).

Separate LIS and IS should be connected and merged in a certain topological structure to create a logical, connected information system. For this purpose registration processes are used. RP add records to database about resource, it is initiated by register information service and is updated on a regular basis to keep the IS records up to date.

IS are arranged in tree structure database containing references to other databases - services of other levels. I.e., there are corporate services registered in services at region levels and so on up to high level services. There are several high-level services which allows to avoid bottleneck places. Services do not have to be registered consistently at the higher levels, for example, any service can be registered immediately at a high level service, except for the highest. This is made to keep the loading of high-level server balanced, and realized by filtration.

Installation of a corporate GIIS which contains the information from all local sites and aggregates it for transfer to a higher level, allows to avoid problems in operation of corporate system and significally raises reliability of central servers. In addition to the local monitor the system maintains functionality even in case of autonomous work.

2.5. Grid Monitor

ARC Grid Monitor [12] is a program that realizes Web interface for ARC information system, allowing to look through the data published via information system. For the user this information is given in the form of a regularly updated web-page, with information about resources and a set of references, allowing to look through all accessible information from information system, including the status of tasks.

2.6. Authentication and authorization

Authentication in ARC is based on a system of certificates using asymmetric enciphering.

To have access to Grid resources the user should have a signed certificate. Having the certificate, user initiates the authentication process in the beginning of a session. A so-called "proxy" certificate which contains ciphered information of the owner is created and allows Grid services to work on behalf of the certificate owner. The "Proxy" certificate is created with limited life time and after its expiration a new one should be created.

2.7. Certificates

ARC safety model is based on Grid Security Infrastructure [13] (GSI). For authentication the Certificate Authorities (CA) registration policies are used.

In order to use a system in Grid it is necessary to receive and establish host certificate. This certificate is necessary for acknowledgement of resource identity and should be established on all systems, wishing to share resources in Grid. The host certificate provides safety of the user (user precisely knows, on what system its program will be executed) and other Grid-hosts cooperating with it.

It is necessary only on front-end server to receive host certificate, instead of on each computer.

User certificates are used for user authorization in Grid resources. User certificates and host certificates subscribe in the same authorization centers.

All certificates are based on asymmetric enciphering and make up for a pair of keys - private and public keys. The public key should be signed in the Certificate Authorities center.

Execution of a task on cluster is started on behalf of the local unix-user who is not having any common with the client who started a task. User certificates are mapped on local unix-users through the grid-mapfile mechanism.

The file grid-mapfile contains a list of certificates allowed on given system and names of unix-users to which certificates are mapped. This file is created and regularly updated by the nordugridmap utility. This utility is connected to databases of users of those virtual organizations which we trust and fills a file. Also individual users can be added, using a file local_users which is also used by the nordugridmap utility.

2.8. The client module

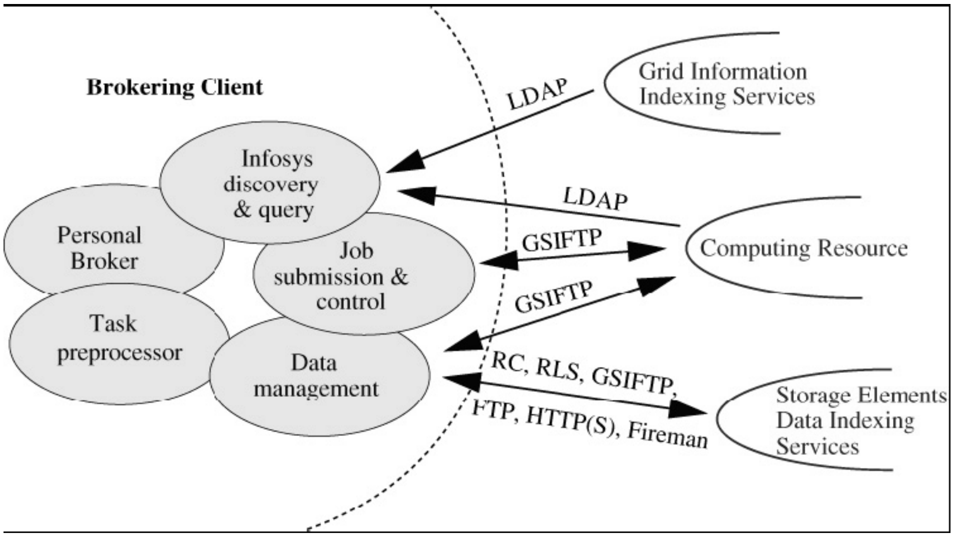
The Client module includes all necessary programs for submitting tasks in Grid.

For task description there is a special language - XRSL [14] (Extended Resource Specification Language). This language represents a set of parameters and attributes with which a task should be carried out.

To avoid necessity to choose suitable cluster for a task by user there is a program - broker. Broker enters into the client module and holds responsibility for a cluster choice, and transfers the task to this cluster.

Broker reads parameters (resources, software) from the description of a task after it is connected to information system and receives the list of resources available at present. Broker chooses from accessible resources suitable under requirements and sends a task on it.

Such realization of the broker (each client finds the resource he needs) allows to avoid the "bottlenecks" caused by starting tasks as there is no uniform broker who starts all tasks. Accordingly there is no possibility of its overload.



Pic.2.Interaction of the broker with Grid services.
Above the arrows used protocols are specified. Picture is downloaded from a site http://www.nordugrid.org/pictures.html

The process of starting a job could be also more correctly called ‘User signing for a job’, as the user only specifies resources which are necessary and signs (applies the proxy data to a task), and on clusters the job is started by services. On each system with ARC the allowed certificates are listed.

3. Installation and adjustment ARC NorduGrid

3.1. The equipment and adjustment of operation system

Installation ARC NorduGrid was made on the following server equipment:

- Front-end a server (ap8.gridzone.ru):

Server platform Supermicro 5015M-MT / 1U / Intel Pentium 4 Dual Core 3.2Ghz/2048/2 x 1024MB DDR2 ECC / 2xSATA Raid 0,1 / 2 x Seagate 200GB Barracuda, SATA-150, 7200rpm, cache 8MB / Lan 2x1Gb

- Computing units (w3, w4, w7, w8):

Server platform Supermicro X7DVL-E, 5000V / 1U " / 2 x Intel Xeon 3.0GHz/1333/4MB (Dual Core) EM64T / 2 x 2048MB DDR ECC REG/2 x Seagate 160GB Barracuda, SATA-150, 7200rpm, cache 8MB/Lan 2x1Gb

Local segment of network is executed on technology GigabitEhternet on the basis of switchboard CiscoCatalyst 2960 24G. For connection to the Internet the external fibre-optical channel 100Mbps is used.

On all servers operational system (OS) Scientific Linux-4.4 [15] (SL 4.4) is established. The choice of this OS is dictated by the following reasons. Scientific Linux - the distribution kit which is compiled from sources of Red Hat Enterprise Linux-4 [16] (RHEL-4), being for today the most developed and steady linux-distribution kit. But, unlike RHEL-4, SL-4.4 is accessible to free use. Software NorduGrid is completely compatible with RHEL 4.

For all servers installation of operational system utility KickStart which allows the complete automatizing of the process of installation and initial adjustment of system was carried out to receive absolutely identical copies of system.

The catalogue /scratch which is mounted by NFS on all cluster computers has been created on front-end server. For this purpose the rpc, portmapper, nfs services were started and adjusted on a server. For assembling the catalogue /scratch on other computers portmapper service have been adjusted. This catalogue is required for ARC services.

In adjustments of ap8.gridzone.ru firewall the certain ports which will be used by services ARC NorduGrid were resolved.

The user "grid" for which free access by ssh to all machines is adjusted was created on all servers.

3.2. Software installation

Before installing the ARC packages it is necessary to install some additional software and local recourse management system, in our case it is a Portable Batch System (PBS) TORQUE [17] which freely distributes as an archive with an initial code.

Additional software includes:

- Grid Packaging Tools [18] (GPT) is a packages control system which monitors for established packages and resolves dependences between them
- Some of Globus Toolkit [19] packages
- gSOAP [20] is intend for simplification of development SOAP/XML web-services and client applications on C/C++
- VOMS [21] - Virtual Organization Membership Service gives the information to users, according to their virtual organizations, groups, roles and opportunities
- Client MySQL [22]
- Libraries libxml2 [23]
- Python [24] development libraries

Next the ARC software which can be found on http://www.nordugrid.org installed.

After installation it is necessary to configure ARC services. For this purpose there is a file /etc/arc.conf which contains all configurations of parameters.

The file arc.conf consists of blocks; each block is related to some ARC service and responsible for its adjustment.

For work it is necessary to start Grid services which should work in the form of "daemons" on front-end server. The start of services is carried out by start/stop scripts; these start the next "daemons" - grid-manager, httpsd, gridftpd, grid-infosys. It should be noted that for successful start grid-manager - PBS system should be already started.

The MPICH2 [25] for supporting MPI2 and a set of compilers gcc-4.2 [26] supporting OpenMP technology have been installed on cluster.

3.3. Installation of the local authorization center

For work in Grid user should receive certificate signed in authorization center. Using of the global authorization centers is not always convenient. For example, during the education students have to practice with Grid on our resources. It is convenient to have the local center of authorization that will considerably simplify and accelerate the procedure of certificates receiving/canceling.

To creation the local authorization center it is necessary to establish package SimpleCA [27], which is a part of GlobusToolkit. As this package does not enter into a set of packages installed earlier, it has been separately compiled from an initial code. Now support of our LocalCA is added on ap8.gridzone.ru and nordic.nw.ru.

4. Testing ARC NorduGrid

4.1. Testing of services

Testing is necessary for many objective reasons. First of all, it is necessary to define working capacity of created system, to estimate that all services work correctly and adjustments are right. The definition of various factors affecting on productivity is also an important problem.

It is necessary to check up a possibility of loading files, launching tasks on cluster, the capacity of work of an information system.

An information system is one of the major services because any activity in Grid is impossible without it.

Testing of information system has been made by programs ldapsearch [28] and LdapBrowser [29], and also the Grid-monitor.

For testing file access services commands ngcp, ngls, ngrm which allow working with files in Grid were used.

Joint testing of all Grid services has been made with usage of the built-in ngtest tests which allow checking up the functionality of system completely. Besides, the real programs using MPI2 and OpenMP for calculation of matrixes have been started. Testing on these programs allowed setting and adjusting of runtime environment.

The major characteristic of computational cluster is its productivity. For an evaluation of the maximal productivity High Performance Linpack Benchmark [30] (HPL) has been used. The given program is used for testing of productivity of computational clusters and the results of testing maintain the "Top500" [31] - list of the most powerful systems in the world. The tests had shown that productivity of our cluster is 5,136e+01 Gflops, that is 51,36 billion operations per second. Each operation is an operation with a number of double accuracy (64 bit). The given value corresponds both to theoretical estimations, and to the practical results received from systems with the similar equipment.

The testing of all subsystems had shown full functionality of components, and the testing of whole system confirmed the correct working.

4.2. Monitoring resources

After service starts, the information system becomes accessible through the general Grid monitor [32] in which the systems connected to ARC NorduGrid are visible and also in local [33], displaying only clusters of our branch. Receiving information about the resource and being convinced in its working capacity is possible by using foreign software such as ldapsearch, LdapBrowser.

In the Grid monitor and the programs mentioned above the information about current cluster and tasks which are carried out at present time is accessible. Often there is necessity to learn the history of the tasks which had passed through the cluster. Due to absence of the specialized program at the moment of installation the program "Logger" which writes down the data about the last tasks in database MySQL has been developed for this purpose, the simple web-interface allowing looking through this information has also been created.

It is important to see not only data about current cluster and history of its functioning, but also to have the statistical information, to analyze it and operatively react to arising supernumerary situations. For these purposes there are many special software packages. There are two packages - Ganglia [34] and Nagios[35], installed for additional system monitoring on our cluster.

Nagios is focused on monitoring of availability of computers and services. It can be rather useful for tracking cluster workability and notifications in case of serious failures; however, it is not capable to trace an operative picture of loading and using without serious additional loadings on system.

Ganglia allows to receive the evident information about conditions of various subsystems, for example, about loading processors, using of operative memory and a disk space, network interfaces, etc.

5. Conclusion

The main purpose of this work was creation of the computational resource connected to Grid allowing to simplify and unify using of various resources and providing access to them for users and virtual organizations.

According to a purpose computational resource connected to Grid environment has been created.

It included:

- installation and adjustment
  - ✓ of server equipment (5 servers, OS Scientific Linux 4.4)
  - ✓ of cluster with PBS system
  - ✓ of software for support high-efficiency calculations (MPI2, OpenMP)
  - ✓ of ARC NorduGrid software
  - ✓ of local authorization center (localCA)
- detailed testing of each service
- complex testing of a system
- testing of cluster productivity

Since 2007 resource is maintained in an industrial mode. During this time a number of practical problems had been solved, in particular the group of employees of St.Petersburg State University had made calculations (with use of technology MPI2) on research of "Runet" structure and its dynamics within the project "Internet-mathematician 2007 ", supported by Yandex. Also resource was used by the Russian and European scientists (Paul Lihatov, Olav Syljuasen, Antti Hyvarinen) for computational tasks. Also the system was used for practical studies on a special "Grid-technology" course of the Computational Physics department.

6. Literature

1. ARC NorduGrid - http://www.nordugrid.org
2. MPI2 standards - http://www.mpi-forum.org/docs/docs.html
3. OpenMP standards - http://www.openmp.org/blog/specifications
4. A. Konstantinov. The NorduGrid Manager and GridFTP Server. Description and Administrator's Manual − NORDUGRID-TECH-2, 2007.
5. B. Kónya. The NorduGrid/ARC Information System − NORDUGRID-TECH-4, 2007.
6. ARC User Interface: User's Manual, NORDUGRID-MANUAL-1 http://www.nordugrid.org/documents/ui.pdf
7. A. Konstantinov. The NorduGrid "Smart" Storage Element − NORDUGRID-TECH-10, 2006.
8. M. Ellert, A.Konstantinov, B. Kónya, O.Smirnova, A.Wäänänen. Architecture Proposal, NORDUGRID-TECH-1, 2002.
9. GACL mini-howto NORDUGRID-MEMO-5 - http://www.gridpp.ac.uk/website/gacl.html
10. Replica Location Service − http://www.globus.org/toolkit/data/rls
11. Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map - http://tools.ietf.org/html/rfc4510
12. O. Smirnova. The Grid Monitor: Usage Manual NORDUGRID-MANUAL-5, 2003.
13. Overview of the Grid Security Infrastructure - http://www.globus.org/security/overview.html
14. O. Smirnova. Extended Resource Specification Language NORDUGRID-MANUAL-4, 2008.
15. Scientific Linux-4.4 - https://www.scientificlinux.org/distributions/4x/44
16. Red Hat Enterprise Linux-4 - http://www.redhat.com
17. TORQUE Resource Manager - http://www.clusterresources.com/pages/products/torque-resource-manager.php
18. Grid Packaging Tools - http://www.gridpackagingtools.com
19. Globus - http://www.globus.org
20. gSOAP - http://www.cs.fsu.edu/~engelen/soap.html
21. VOMS - http://vdt.cs.wisc.edu/VOMS-documentation.html

22. MySQL 3.23, 4.0, 4.1 Reference Manual - http://dev.mysql.com/doc/refman/4.1/en/index.html
23. The XML C parser - http://xmlsoft.org
24. Guido van Rossum. Python Library Reference. Python Software Foundation - http://docs.python.org/lib
25. MPICH2 - http://www.mcs.anl.gov/research/projects/mpich2
26. GNU Compiler Collection version-4.2.0 - http://gcc.gnu.org/gcc-4.2
27. SimpleCA - http://www.globus.org/toolkit/docs/4.0/admin/docbook/ch07.html
28. ldapsearch - http://docs.sun.com/source/816-6400-10/lsearch.html
29. Ldap Browser - http://www.ldapbrowser.com
30. A. Petitet, R. C. Whaley, J. Dongarra, A. Cleary. HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers - http://www.netlib.org/benchmark/hpl
31. «Top500» list - http://www.top500.org
32. Main Grid monitor- http://www.nordugrid.org/monitor
33. LocalGrid monitor - http://nordic.nw.ru/gridmonitor
34. Monitoring system for high-performance computing systems – «Ganglia» - http://ganglia.sourceforge.net
35. An enterprise-class monitoring solutions for hosts, services, and networks – «Nagios» - http://www.nagios.org