

# Telecommunication Network Modeling and Planning Tool on ASP clusters

P. Bacquet, O. Brun, J.M. Garcia, T. Monteil, P. Pascal, and S. Richard

DELTA Partners SA ANITE Group and LAAS-CNRS, Toulouse, France,  
monteil@laas.fr,  
WWW home page: <http://www.laas.fr/CASP>

**Abstract.** The development of Internet allows to consider new execution paradigm in industrial context: ASP model (Application Service Provider). An interesting execution support to implement this concept is clusters of PC. General principles of the ASP are described. The architecture for the resource manager and watcher is described. A set of existing or created tools to materialize a solution to this problem is explained. An industrial telecommunication modeling and planning software is used on the ASP. In particular, the analytical performance evaluation kernel of this software is parallelized on the cluster through the ASP. Several results on cluster are given.

## 1 Introduction

Many users around the world need a set of applications to administer the companies, to manage relations with customers or to solve scientific problems. Those kinds of software could be heavy, expensive and sometimes made with many specific components depending on utilization (plug in notion). They could require specific help during hard work time or for complex problems. They could also need powerful computers (processor, memory or data storage) and the purchase of a lot of licenses for punctual work.

For different reasons (financial, human resources, complexity of solution, ...) , the potential user of those software could have difficulties to efficiently use those applications on the machine in his office. Another aspect, is the response time requested for the application which can be very short.

A solution to those problems is to use remote applications on remote resources: ASP model (Application Service Provider) to save money and time. The most common way to offer all those services is to use Internet or Intranet. The machines that could run those remote applications should be multi-processors for parallel or sequential applications. The clusters of PC is, at this time, the best execution support due to its low price and its capacity of scalability, modularity and evolution.

Nevertheless, the utilization of clusters in ASP model requires specific software to administrate the clusters, to manage the resources, to communicate through the network and to insure security. An important aspect for industrial utilization is to insure high availability, different classes of quality of service and exact accounting.

This article deals with an ASP solution developed in the CASP project (Clusters for Application Service Providers)<sup>1</sup> and its validation on the NetQUAD telecommunication modeling software (DELTA Partners SA, a company of Anite group) [1]. The main contributions of this article are to propose a set of tools to create ASP on clusters and to present the experiments done on an industrial application. In section 2, an ASP model is proposed. Then, the NetQUAD software is described. A specific NetQUAD plug-in for ASP utilization and a new parallel kernel are detailed. Finally, some results are given on an ASP cluster.

## 2 ASP model proposed

### 2.1 Introduction

CASP (Clusters for Application Service Providers) project aims at creation of remote Internet services on a cluster for high performance applications. Two entities are defined: the customers and the providers. The providers own the resources (processors, disks, memory and applications servers) and the customers possess all the necessary tools to express their needs (quality of service, resources, software). The providers and the customers could be in the same network area, in this case the network used is an Intranet. The providers could also be outside the customers company. In this case the network between customers and providers could be dedicated or free like the Internet.

Cluster architecture is used because of its extension capacities and modularity capacities. A software infrastructure is needed to manage the cluster in order to make ASP. Resources management with quality of service and an Internet portal are also needed.

The goal of this project is to put on the market a generic product for ASP running on Linux with an end to end solution. It is composed of :

- Software managing reliable communications between a client and a remote application.
- Tools to guarantee a quality of service to make an industrial and commercial use (resources management, high availability).
- Software to manage the provider platform.

The project allows to use cluster in an industrial and commercial world, in the supercomputing and ASP domain.

---

<sup>1</sup> This project is a cooperation between Alinka SA, Delta Partners SA group Anite and LAAS-CNRS and is funded by the French Ministry of Industry and RNTL (Reseau National des Technologies Logicielles)

## 2.2 AROMA

AROMA (scAlable ResOurces Manager and wAtcher) is the part of CASP integrating all the necessary tools to manage the remote execution with a defined quality of service to produce an ASP. AROMA integrates a resource management system [3, 4], an application launcher and an accounting system. This tool is an evolution of works done on Network-Analyser system [5].

It must have an easy-to-use graphical interface which must adapt to different classes of users : resource customers, administrators. They have access to different dynamic services : visualization of dynamic and static information on resources (CPU, memory, network, libraries) on different times (last seconds, minutes, hours, days, weeks, months, years) and mapping services. The customers can use different services depending on their permissions. The user interface is dynamic and depends on permissions and evolution of the services. A client defines a contract with the provider. According to this contract, the provider guarantees a quality of service and the client pays according to the account.

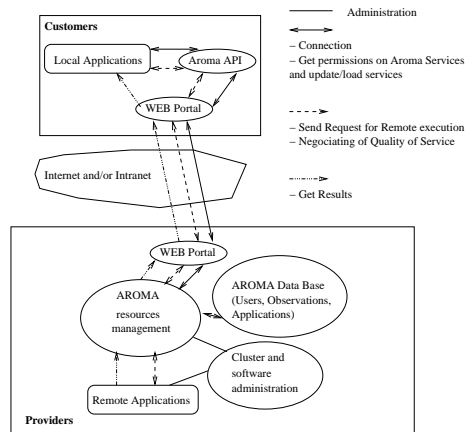
At first (figure 1), the customers build all the necessary information for the job to execute. The connection to the remote system is defined with a login, a group and a contract key. The contract will define utilization of resources and quality level for the remote execution. The first connection is used to define the different permissions on all the AROMA systems services and on the remote applications controlled by AROMA. The data base of AROMA is used to do this. It is also used to update or load services on the client host. It could be new services given to customers or services acquired with an evolution of their contract. It is dynamic and transparent for users.

A request is sent to AROMA to execute all or a part of the work. A negotiation is established between AROMA and the application depending on the quality of service. Finally the job is executed into AROMA environment.

AROMA is able to control the execution (real time resources consummation, end of application detection). This functionality is used by the accounting system for example to stop a job which has consumed too many resources. When the remote application is finished, customers are advertised and can manually or automatically get the results.

The AROMA resource manager has to face three problems: portability, dynamicity and efficiency.

- **Portability:** The Java programming language has been chosen to ensure portability. It guarantees code portability without re-writing.
- **Dynamicity:** Two levels of dynamicity can be distinguished. Dynamicity due to changes in the cluster topology and dynamicity due to users permissions. Both aspects are resolved using Jini technologies and Java Remote



**Fig. 1.** Communication between customers and providers

Method Invocation system <sup>2</sup>. Jini communication protocols are used to detect changes in the cluster architecture, while RMI dynamic class downloading is used to offer different services depending on the client permissions.

- **Efficiency:** Mapping with Quality of Service implies watching a lot of resources. The mapping decision has to be as quick as possible and must not create a system overload. C language is used to collect system dependent information in a very efficient way. Communication between C modules and Java servers is ensured using the Java Native Interface.

### 2.3 WEB portal

With the Internet portal an enterprise can have a personalized access to its applications. All dialogues between clients and providers use a WEB portal to insure security and transparent communications (figure 1).

### 2.4 Clusters and application administration

To insure a good quality of service of the whole system in a cluster utilization, the software Alinka raisin [7] is used. It manages quick installation and configuration of Linux kernel and useful software. Installation and management of clusters could be done using Alinka raisin. It helps providers to manage different clusters and operating system version. It could automatically create the architecture of AROMA. It is also used to link UNIX users on the provider clusters and remote users on the customer machine.

<sup>2</sup> Java, all Java-based marks, Jini and all Jini-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

## 3 Telecom Modeling Application

### 3.1 Principle

**The NetQUAD Network Planning Tool** NetQUAD is a software suite that provides a set of tools for the design, optimization and performance evaluation of large scale telecommunication networks [2]. Most network technologies are included in the various plug-in : Telephone networks, IP networks, IP-DiffServ-MPLS networks, ATM, Frame Relay, SDH. This software is used by many operators and manufacturers.

A key point in the NetQUAD kernel is the differential traffic theory that is used for the analytical modeling of stochastic traffic (link carried traffics, end to end carried traffics, blocking probabilities, waiting times, jitter ...) [11, 12]. These models are very accurate and far more faster than discrete event simulations for performance evaluation of large scale telecommunication systems with complex routing and complex traffic patterns [10].

Nevertheless solving the whole set of non-linear coupled differential equations can take several hours of computing time for real networks. Computing time reduction is a major issue when one is interested in testing several changes in the network or implementing iterative optimization algorithms.

In the following we describe the circuit-switched models used in the NetQUAD Telephony plug-in and how these models have been parallelized in order to drastically reduce the computing times for very large scale networks (millions of flows, thousands of links, etc.).

**Circuit-switched Network Modeling in NetQUAD** Using classical exponential inter arrival time and service time laws, a rigorous description of the state of a circuit-switched network is given by a Markov chain. Though such hypothesis can be practically verified, they are not suited for modeling because of the huge number of states that need to be considered in a real network.

It is the aim of "traffic modeling" (queuing theory) to give ways to evaluate network performances such as the mean and variance of carried traffics as well as nodes and trunks blocking probabilities [13, 14].

Theoretical studies have permitted to study the form of the mean differential equation associated with each network resource in the case of simple "cells" [8, 9]. These cells reflect classical "serial/parallel" structures that can be found within a network. This analysis has been done for call routing combining load-sharing and overflow. The first fundamental idea, developed within NetQUAD Telephony, is to model the traffic starting from the exact differential equation of each traffic flow. The second idea is to approximate blocking probabilities by means of fictitious offered traffics (equivalent traffics) linked to such set of

differential equations.

Finally, the model is described by a set of ordinary differential equations whose structure (and number of equations) depends on the call routing rules. Within this model, call routing is generic; this allows any network topology associated to any call routing mechanism (multiple overflows and cross-overflows, load-sharing on primary or overflow routes, etc.)

These studies have led to the definition of a new efficient and generic tool targeted to circuit-switched networks performance analysis. NetQUAD Telephony includes an algorithm named VTAM (Voice Traffic Analytical Modeling), the numerical translation of this model. The main advantages of VTAM are the following:

- good accuracy compared to other approaches,
- dynamical model avoiding heavy event-driven simulations when the need to study transient behavior arises,
- easy computation of the stationary solution due to an efficient fixed point algorithm (contraction property),
- differentiable with respect to network parameters (allowing gradients computation and parameter optimization).

In particular, the fixed point algorithm used to solve the nonlinear system of equations works as follows. At each iteration, once the blocking probabilities have been computed, new values of carried traffics are computed according to the traffic nonlinear equations. These equations are applied for each traffic flow independently. The blocking probabilities on trunks, set of trunks or path, are then updated by applying the Erlang-B formula to fictitious offered traffics. This process is repeated until the global convergence of the blocking probabilities.

**VTAM Parallel Algorithm** The most straightforward way to take advantage of the inherent parallelism of the VTAM algorithm is to use a synchronous data parallel approach. It consists in partitioning the set of all flows into several independent separate subsets. Each subset is assigned to one processor.

The assignment of the  $N$  flows to the  $K$  parallel tasks is of course a major issue in achieving a good load-balancing. Assigning nearly  $N/K$  flows to each task does not always lead to a good load-balancing. Indeed the computational load of a parallel task greatly depends on the number of routing commands  $r_i$  of the flows  $i$  assigned to that task.

The assignment policy we have used can be described as follows. Let  $l_1^j$  and  $l_2^j$  be respectively the first and last flows assigned to task  $j = 0, \dots, K - 1$ . Obviously, we have the following relations:

$$l_1^0 = 0, l_1^1 = l_2^0 + 1, \dots, l_1^K - 1 = l_2^{K-2} + 1 \quad \text{and} \quad l_2^{K-1} = N - 1$$

For tasks  $j = 0, \dots, K-2$ , the last flow  $l_2^j$  assigned to task  $j$  can be computed recursively using the following formula:

$$l_2^j = \min\{k > l_1^{j-1} \text{ such that } \sum_{i=l_1^j}^k r_i > \sum_{i=0}^N r_i / K\}$$

Experimental results have shown that this assignment policy allows to obtain a very good load-balancing of the parallel tasks.

Once flows have been assigned to tasks, the associated nonlinear system of differential equations is solved. To this end, the  $K$  parallel tasks iteratively performs a fixed point algorithm. This parallel algorithm is based on the message passing paradigm and uses the MPI communication library. Iteration  $k$  of this algorithm proceeds as follows:

1. **Computation of carried traffics.** Each task  $j = 0, \dots, K-1$  computes new values of carried traffics  $X_j^l(k)$  for each link  $l$  using its own subset of flows. This computation is done independently by the  $K$  tasks using the same algorithm than in the sequential algorithm.
2. **Global reduction operation** Each task  $j = 0, \dots, K-1$  exchanges its vector  $(X_j^0(k), \dots, X_j^L(k))$  of carried traffics with all other tasks and computes the global carried traffic  $X^l(k) = \sum_{j=0}^{K-1} X_j^l(k)$  for each link  $l$ .
3. **Update of blocking probabilities** Each task  $j = 0, \dots, K-1$  independently updates the blocking probabilities of each link, set of links or path using the carried traffics  $X^l(k)$ ,  $l = 0, \dots, L$ .

This process is repeated until the global convergence of the blocking probabilities.

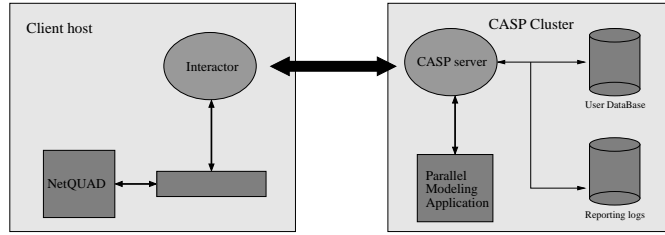
### 3.2 ASP utilization

Interactions between NetQUAD and CASP servers are illustrated by (figure 2).

NetQUAD connection to AROMA is made without any change to the NetQUAD initial code. All the work is performed by a special plug-in called the Interactor.

The aim of the Interactor is to handle the communication work between the local NetQUAD and the CASP servers. The Interactor uses both NetQUAD and AROMA Application Programming Interfaces to perform authentication, authorization, submission, reporting and negotiation.

Authorization and authentication is done in two steps. First, a local verification is made in order to know if the user owns a NetQUAD license, then a remote verification is done to test if the user is known as an AROMA client.



**Fig. 2.** Interaction between NetQUAD and CASP servers

This remote verification is performed with each later communication between the Interactor and the CASP servers in order to be sure that the user has right to make the associated operation.

Once the user has been identified as an AROMA client, he can submit its jobs. The Interactor then communicates with the CASP servers to make reporting or to collect results. Both operations can be done in-line or in batch mode.

An AROMA client is linked to a contract which specified the client limitations on resources utilization (globally : number of processors, maximum cpu time, memory size...). For every new submission, a verification is performed to test if the user respects its contract resource limitations. If it not the case, the Interactor is in charge of modifying and re-submitting the NetQUAD request so that it can suit the user contract limitations.

### 3.3 Validation on ASP cluster

The results given in the following have been obtained on a linux PC cluster. The main features of the cluster nodes are given in table 1. These nodes are interconnected by a 1 Gb/s myrinet network.

Number of Processors	2
Model Name	Pentium III (Katmai)
CPU MHz	451.032
RAM memory	128 MB
Cache Size	512 KB

**Table 1.** Main features of the cluster nodes

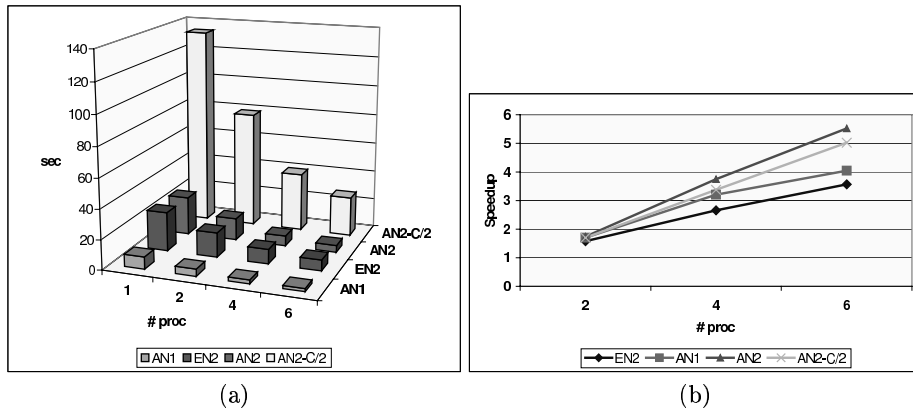
The testbed networks used to assess the performance of the parallel algorithm are described in table 2. The AN2 topology approaches real topologies with very large trunk groups. The AN2-C/2 topology has been obtained from the AN2 topology by reducing the link capacities by a factor 2.



	EN2	AN1	AN2	AN2-C/2
Number of Nodes	113	208	312	312
Number of Links	287	388	606	606
Average capacity per link	136.64	380.94	532.83	266.41
Number of flows	12321	11200	25200	25200
Total Traffic (Erlangs)	12321	11200	25200	25200
Max Number of Hops	3	4	4	4
Max number of OverFlows	1	1	1	1
Number of Iterations	5	3	3	40
Size of the input file (MB)	19.9	42	100	100

**Table 2.** Main features of the testbed networks

The parallel computing times obtained on the linux PC cluster with the 4 testbed networks are given in figure 3.a while the corresponding parallel speedups are shown in figure 3.b.



**Fig. 3.** VTAM computing times and parallel speedups.

The best speedups are obtained for the large size networks AN2 and AN2-C/2. For the AN2-C/2 topology, the computing time falls from 138 seconds on 1 processor to only 27.9 seconds on 6 processors. However, when the I/O processing time is a significant part of the computing time, worst speedups are obtained.

## 4 Conclusion

The ASP model developed in the CASP project will respond to industrial needs. Many high performance application will use remote Internet services on cluster with ASP. The Internet portal guarantees security and transparency to industrial customers. The client part is portable (use of Java Virtual Machine) and

the API allows to convert different applications to the ASP model.

The ASP concept developed within the CASP project has been validated using the NetQUAD telecommunication modeling software, an industrial application used by many operators and manufacturers around the world. The new NetQUAD parallel plug-in specifically developed for ASP execution on cluster demonstrates the power of this concept. As can be seen from the experimental results, such analytic simulations are efficiently parallelized on clusters and can also support remote execution by using input and output files.

New researches are undertaken in order to supply several grades of service to applications running on the parallel cluster according to contracts between the customers and the provider.

## References

1. <http://www.anite.com>
2. <http://www.delta.fr>
3. Sun Grid Engine. <http://www.gridengine.sunsource.net>
4. Karl Czajkowski, Ian Foster, Nick Karonis, Carl Kesselman and Stuart Martin. *A resource management architecture for metacomputing systems*. Proceedings 4th workshop on Job Scheduling Strategies for Parallel Processing, pp 62-82 Springer-Verlag LNCS 1459,1998.
5. T. Monteil, J.M. Garcia *Task Allocation Strategies on Workstations Using Processor Load Prediction*. PDPTA97, International Conference on Parallel and Distributed Processing Techniques and Applications, pp. 416-421, Las Vegas USA, 1997
6. <http://www.gridforum.org>
7. <http://www.alinka.com>
8. Garcia J. M., Le Gall F. et Bernussou J. *A model for telephone networks and its use for routing optimization purposes*, newblock IEEE Journal on selected areas in communication, Special issue on communication network performance evaluation, Vol. 4, No 6, september 1986.
9. Garcia J.M. *Problèmes liés à la modélisation du trafic et l'acheminement des appels dans un réseau téléphonique*. Thèse de doctorat, Université Paul Sabatier (Toulouse), 1980.
10. Misra J. *Distributed Discrete-Event Simulation*. ACM Computing Surveys, Vol.18, N° 1, 1986.
11. JM Garcia, D. Gauchard, O. Brun, P. Bacquet, J. Sexton and E. Lawless. *Modélisation Différentielle et Simulation Hybride Distribuée*, Réseaux et Systèmes Répartis, Volume 13, No 6, 2001.
12. Garcia J.M. *A new approach for analytical modelling of packet switched telecommunication networks*. LAAS Research Report N° 98443, 1998.
13. Roberts, J. et al Eds *Broadband Network Teletraffic, Final Report of Action Cost 242*. Springer Berlin, 1996.
14. Takacs, L. *Introduction to the Theory of Queues*. Oxford University Press, 1962.