INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team MESCAL

Middleware Efficiently SCALable

Rhône-Alpes

THEME NUM

Table of contents

1. Team

MESCAL project is a common project supported by CNRS, INPG, UJF and INRIA located in the ID-IMAG labs (UMR 5132).

Head of project team
Bruno Gaujal [DR, INRIA]

Administrative staff
Barbara Amouroux [INRIA Administrative Assistant, half-time]
Marion Ponsot [INRIA Administrative Assistant, half-time]

CNRS Staff
Arnaud Legrand [CR]

INPG Staff
Yves Denneulin [Assistant Professor]
Brigitte Plateau [Professor]

UJF Staff
Vania Martin [Assistant Professor]
Jean-François Méhaut [Professor]
Florence Perronnin [ATER 09/05-09/06]
Olivier Richard [Assistant Professor]
Jean-Marc Vincent [Assistant Professor]

Project technical staff
Nicolas Capit [RNTL IGI- 19/04-09/06]
Johann Peyrard [INRIA- 10/05-10/06]

PhD students
Carlos Barrios [2005, EGIDE, co-tutelle]
Anne Bouillard [2002, Normalien, MRNT scholarship]
Léonardo Brenner [2004, Brazilian CAPES scholarship]
Nicolas Bernard [2005, Luxembourg scholarship]
Georges DaCosta [2001, Normalien, MRNT scholarship]
Estelle Gabarron [2003, CIFRE BULL scholarship]
Adrien Lebre [2002, INRIA scholarship, Bull contract]
Maxime Martinasso [2003, CIFRE BULL scholarship]
Jérome Mazuy [2005, CIFRE “Pole Européen de Plasturgie” scholarship]
Duc Nguyen [2005, INRIA scholarship]
Jean-Michel NLong 2 [2002, INRIA scholarship , HP contract]
Afonso Sales [2005, Brazilian CAPES scholarship]
Ihab Sbeity [2003, MRNT scholarship]
Olivier Valentin [2003, MRNT scholarship]
Lucas Nussbaum [2005, BDI-CNRS MRNT scholarship]
Brice Videau [2005, MRNT scholarship]
Blaise Yenké [2004, Ngaundere University scholarship]

2. Overall Objectives

2.1. Presentation

MESCAL is a new INRIA team, created in 2005. The former APACHE project was closed in 2004 and gave birth to two new teams, MOAIS and MESCAL. These two projects still have strong collaborations, which are pointed out throughout this document.

2.2. Objectives

The recent evolutions in computer networks technology, as well as their diversification, yield a tremendous change in the use of these networks: applications and systems can now be designed at a much larger scale than before. This scaling evolution is dealing with the amount of data, the number of computers, the number of users, and the geographical diversity of theses users.

This race towards *large scale* computing questions many hypotheses underlying parallel and distributed algorithms and common management middlewares. Tools developed for average size systems cannot be run on large scale systems without a significant degradation of their performances.

The goal of the MESCAL project is to design and validate exploitation mechanisms (middleware and system services) for large distributed infrastructures.

MESCAL's target applications are intensive scientific computations such as cellular micro-physiology, protein conformations, particle detection, combinatorial optimization, Monte Carlo simulations, and others. Such applications are constituted of a large set of independent, equal-sized tasks and therefore may benefit from large-scale computing platforms. Initially executed on large dedicated clusters (CRAY, IBM, COMPAQ), they have been recently deployed on collections of homogeneous clusters aggregating a large number of commodity components. The experience showed that such clusters offer a huge computing power at a very reasonable price. MESCAL's target infrastructures are aggregations of commodity components and/or commodity clusters at metropolitan, national or international scale. Examples of target infrastructures are grids obtained through mutualization of available resources inside autonomous computing services, lightweight grids (such as the local CIMENT Grid) which are limited to trusted autonomous systems, clusters of intranet resources (Condor) or aggregation of Internet resources (SETI@home, Xtremweb).

MESCAL's methodology in order to ensure **efficiency** and **scalability** of proposed mechanisms is based on systematic modeling and performance evaluation of target architectures, software layers and applications.

3. Scientific Foundations

3.1. Large System Modeling and Analysis

Keywords: *Discrete event dynamic systems*, *Markov chains*, *Performance evaluation*, *Petri nets*, *Queueing networks*, *Simulation*.

Participants: Bruno Gaujal, Arnaud Legrand, Brigitte Plateau, Olivier Richard, Jean-Marc Vincent.

Understanding qualitative and quantitative properties of distributed systems and parallel applications is a major issue. The *a posteriori* analysis of the behavior of the system or the design of predictive models are notoriously challenging problems.

Indeed, large distributed systems contain many different features (processes, threads, jobs, messages, packets) with intricate interactions between them (communications, synchronizations). The analysis of the global behavior of the system requires to take into account large data sets.

As for *a priori* models, our current research focuses on capturing the distributed behavior of large dynamic architectures. Actually, both formal models and numerical tools are being used to get predictions on the behavior of large systems.

For large parallel systems, the non-determinism of parallel composition, the unpredictability of execution times and the influence of the outside world are usually expressed in the form of multidimensional stochastic processes which are continuous in time with a discrete state space. The state space is often infinite or very large and several specific techniques have been developed to deal with what is often termed as the "curse of dimensionality".

MESCAL deals with this problem using several complementary tracks:

- Behavior analysis of highly distributed systems,
- Simulation algorithms able to deal with very large systems,

- Fluid limits (used for simulation and analysis),
- Decomposition of the state space,
- Structural and qualitative analysis.

3.1.1. Behavior analysis of highly distributed systems

The development of highly distributed architectures running widely spread applications requires to elaborate new methodologies to analyze the behavior of systems. Indeed, runtime systems on such architectures are empirically tuned. Analysis of executions are generally manually performed on post-mortem traces that have been extracted with very specific tools. This tedious methodology is generally motivated by the difficulty to characterize the resources of such systems. For example, big clusters, grids or peer-to-peer (P2P) [1] networks present properties of size, heterogeneity, dynamicity that are usually not taken into account in classical system models. The asynchrony of the architecture also induces perturbations in the behavior of the application leading to significant slow-down that should be avoided. Therefore, when defining the workload of the system, the distributed nature of applications should be taken into account with a specific focus on problems related to synchronizations.

3.1.2. Simulation of distributed systems

Since the advent of distributed computer systems an active field of research has been the investigation of *scheduling* strategies for parallel applications. The common approach is to employ scheduling heuristics that approximate an optimal schedule. Unfortunately, it is often impossible to obtain analytical results to compare the efficiency of these heuristics. One possibility is to conduct large numbers of back-to-back experiments on real platforms. While this is possible on tightly-coupled platforms, it is infeasible on modern distributed platforms (i.e. Grids or peer-to-peer environments) as it is labor-intensive and does not enable repeatable results. The solution is to resort to *simulations*. Simulations not only enable repeatable results but also make it possible to explore wide ranges of platform and application scenarios.

The SIMGRID framework enables the simulation of distributed applications in distributed computing environments for the specific purpose of developing and evaluating scheduling algorithms. This software is the result of a long-time collaboration with Henri CASANOVA (University of California, San Diego).

3.1.3. Perfect Simulation

Using a constructive representation of a Markovian queueing network based on events (often called GSMPs), we have designed a perfect simulation tool computing samples distributed according to the stationary distribution of the Markov process with no bias. Two softwares have been developed. ψ analyzes a Markov chain using its transition matrix and provides perfect samples of cost functions of the stationary state. ψ^2 samples the stationary measure of Markov processes using directly the queueing network description. Some monotone networks with up to 10^{50} states can be handled within minutes over a regular PC.

3.1.4. Fluid models

Web caches as well as peer-to-peer systems must be able to serve a set of customers which is both large (several tens of thousands) and highly volatile (with short connection times). These features make analysis difficult when classical approaches (like Markovian Models or simulation) are used. We have designed simple fluid models to get rid of one dimension of the problem. This approach has been applied to several systems of web caches (such as Napster) and to peer-to-peer systems (such as BitTorrent). This helps to get a better understanding of the behavior of the system and to solve several optimization problems.

3.1.5. Markov Chain Decomposition

The first class of models we will be using is Continuous time Markov chains (CTMC). The usefulness of Markov models is undisputed, as attested by the large number of modeling tools implementing Markov solvers.

[1] Our definition of peer-to-peer is a network (mainly the internet) over which a large number of autonomous entities contribute to the execution of a single task

However their practical applications are limited by the *state-space explosion* problem, which puts excessive demands on memory and execution time when studying large real-life systems. Continuous-time Stochastic Automata Networks describe a system as a set of subsystems that interact. Each subsystem is modeled by a stochastic automaton, and some rules between the states of each automaton describe the interactions between subsystems. The main challenge is to come up with ways to compute the asymptotic (or transient) behavior of the system without ever generating the whole state space. Several techniques have been developed in our group based on bounds, lumpability, symmetry and properties of the Kronecker product. Most of them have been integrated in a software tool (PEPS) which is openly available.

3.1.6. Discrete Event Systems

As seen before, the interaction of several processes through synchronization, competition or superposition within a distributed system is the source of the main difficulties because it induces a state space explosion and a non-linear dynamic behavior. Here, the use of exotic algebras, such as (min,max,plus) can help. Highly synchronous systems become linear in this framework and therefore are amenable to formal solutions in simple cases. More complicated systems are neither linear in (max,plus) nor in the classical algebra. Several qualitative properties have been established for a large class of such systems called free-choice Petri nets (sub-additivity, monotonicity or convexity properties). Such qualitative properties are sometimes enough to assess the class of routing policies optimizing the global behavior of the system. They are also useful to design efficient numerical tools computing the asymptotic behavior.

3.2. Management of Large Architectures

Keywords: *Administration*, *Clusters*, *Deployment*, *Grids*, *Job scheduler*, *Peer-to-peer*.

Participants: Arnaud Legrand, Olivier Richard, Vania Marangozova.

As already mentioned, the race towards *large scale* computing questions many hypotheses underlying parallel and distributed algorithms and common management middlewares. Most distributed systems deployed nowadays are characterized by a high dynamism of their entities (participants can join and leave at will), a potential instability of the large scale networks (on which concurrent applications are running), and the increasing probability of failure. Therefore, as the size of the system increases, it becomes necessary that it adapts automatically to the changes of its components, requiring a self-organization of the system with respect to the arrival and departure of participants, data, or resources.

As a consequence, it becomes crucial to understand and model the behavior of large scale systems, to efficiently exploit these infrastructures. In particular it is essential to design dedicated algorithms and infrastructures handling a large amount of users and/or data.

MESCAL deals with this problem using several complementary tracks:

- Fairness in large-scale distributed systems,
- Deployment and management tools,
- Scalable batch scheduler for clusters and grids.

3.2.1. Fairness in large-scale distributed systems

Large-scale distributed platforms (Grid computing platforms, enterprise networks, peer-to-peer systems) result from the collaboration of many people. Thus, the scaling evolution we are facing is not only dealing with the amount of data and the number of computers but also with the number of users and the diversity of their behavior. In a high-performance computing framework, the rationale behind this joining of forces is that most users need a larger amount of resources than what they have on their own. Some only need these resources for a limited amount of time. On the opposite some others need as many resources as possible but do not have particular deadlines. Some may have mainly tightly-coupled applications while some others may have mostly embarrassingly parallel applications. The variety of user profiles makes resources sharing a challenge. However resources have to be *fairly* shared between users, otherwise users will leave the group and join another one. Large-scale systems therefore have a real need for fairness and this notion is missing from classical scheduling models.

3.2.2. Tools to operate clusters

The MESCAL project studies and develops a set of tools designed to help the installation and the use of a cluster of PCs. The first version had been developed for the icluster1 platform exploitation. The main tools are a scalable tool for cloning nodes (KA-DEPLOY) and a parallel launcher based on the TAKTUK project (now developed by the MOAIS project). Many interesting issues have been raised by the use of the first versions among which we can mention environment deployment, robustness and batch scheduler integration. A second generation of these tools is thus under development to meet these requirements.

The new KA-DEPLOY has been retained as the primary deployment tool for the experimental national grid GRID'5000.

3.2.3. OAR: simple and scalable batch scheduler for clusters and grids

This software is co-developed with MOAIS project.

Most known batch schedulers (PBS, LSF, Condor, ...) are of old-fashioned conception, built monolithically, with the purpose of fulfilling most of the exploitation needs. This results in systems of high software complexity (150000 lines of code for OpenPBS), offering a growing number of functions that are, most of the time, not used. In such a context, it becomes hard to control both the robustness and the scalability of the whole system.

OAR is an attempt to address these issues. Firstly, OAR is written in a very high level language (Perl) and makes intensive use of high level tools (MySql and TAKTUK), thereby resulting in a concise code (around 5000 lines of code) easy to maintain and extend. This small code as well as the choice of widespread tools (MySql) are essential elements that ensure a strong robustness of the system. Secondly, OAR makes use of Sql requests to perform most of its job management tasks thereby getting advantage of the strong scalability of most database management tools. Such scalability is further improved in OAR by making use of TAKTUK to manage nodes themselves.

Current development in OAR focuses on its extension to Grids and advanced scheduling techniques. The extension of OAR to Grids has already started by making it support best effort jobs. The integration of advanced scheduling techniques is in progress and aims at adding both state of the art batch scheduling algorithms and new task models to the system.

3.3. Migration and resilience

Keywords: *Fault tolerance, distributed algorithms, migration.*

Participants: Yves Denneulin, Jean-François Méhaut.

Making a distributed system reliable has been and remains an active research domain. Nonetheless this has not so far lead to results usable in an intranet or federal architecture for computing. Most propositions address only a given application or service. This may be due to the fact that until clusters and intranet architectures arose, it was obvious that client and server nodes were independent. So, a fault or a predictable disconnection

on most of the nodes didn't lead to a complete failure of the system. This is not the case in parallel scientific computing where a fault on a node can lead to a data loss on thousands of other nodes. The reliability of the system is hence a crucial point. MESCAL's work on this topic is based on the idea that each process in a parallel application will be executed by a group of nodes instead of a single node: when the node in charge of a process fails, another in the same group can replace it in a transparent way for the application.

There are two main problems to be solved in order to achieve this objective. The first one is the ability to migrate processes of a parallel, and thus communicating, application without enforcing modifications. The second one is the ability to maintain a group structure in a completely distributed way. The first one relies on a close interaction with the underlying operating systems and networks, since processes can be migrated in the middle of a communication. This can only be done by knowing how to save and to replay later all ongoing communications, independently of the communications. Freezing a process to restore it on another node is also an operation that requires collaboration of the operating system and a good knowledge of its internals. The other main problem (keeping a group structure) belongs to the distributed algorithms domain and is of a much higher level nature.

The resulting software of this research topic is called SAMORY and is able to keep a set of processes alive on a given set of nodes, even in the presence of faults, hardware or software. It has been used on seismic applications (wave propagation) as a part of the RNTL IGGI project (http://iggi.imag.fr). SAMORY is freely available at http://iggi.imag.fr. It is composed of a Linux kernel module and a daemon that monitors the processes and their communications and reacts when a fault is discovered or suspected.

Future work will concern the behavior analysis of checkpoint systems in order to predict precisely critical operations to optimize resource usage (network and disk bandwidth).

3.4. Large scale data management

Keywords: *Fault tolerance, distributed algorithms, migration.*

Participants: Yves Denneulin, Vania Marangozova.

In order to use large data, it is necessary (but not always sufficient, as seen later) to efficiently store and transfer them to a given site (a set of nodes) where it is going to be used. The first step toward this achievement is the construction of a file system that is an extension of NFS for the grid environment. The second step is an efficient transfer tool that provides throughputs close to optimal (*i.e.* the capacity of the underlying hardware).

3.4.1. Distributed storage over a cluster

3.4.1.1. Performances

NFSp is a distributed file system for clusters that enables one to store data over a set of nodes (instead of a single one). It was designed to permit the usage of a set of disks to optimize memory allocations. It is important for performance and simplicity that this new file system has little overhead for access and updates. From a user point of view, it is used just as a classical NFS. From the server point of view, however, the storage is distributed over several nodes (possibly including the users).

The mounting point is only in charge of the meta-data, name, owner, access permissions, size, inodes etc., of the files while their content is stored on separate nodes. Every read or write request is received by the meta-server, the mounting point, which sends them to the relevant storage nodes, called IOD for Input/Output Daemon which will serve the request and send the result to the client.

Two implementations were done, one at the user level and one at the kernel level. Performances are good for read operations, for example 150MBs/sec for 16 IODs connected through a 100Mb/s for 16 clients. For write operations performances are limited by the bandwidth available for the meta-server which is a significant bottleneck.

3.4.1.2. Reliability

Storage distribution on a large set of disks raises the reliability problem: more disks mean a higher fault rate. To address this problem we introduced in NFSp a redundancy on the IODs, the storage nodes by defining VIOD, Virtual IOD, which is a set of IODs that contain exactly the same data. So when an IOD fails another

one can serve the same data and continuity of service is insured though. This doesn't modify the way the file-system is used by the clients: distribution and replication remain transparent. Several consistency protocols are proposed with various levels of performance; they all enforce at least the NFS consistency which is expected by the client.

3.4.2. Efficient transfer on grids

To efficiently transfer files across a grid a "beowulf-like" solution consists in creating a set of point-to-point communications to parallelize the transfer of a file or a set of files. This approach was chosen, for instance, in gridftp [52]. It implies duplicating the data to transfer or distribute them on separate nodes before the transfer begins. We use the distributed storage property of NFSp to be able to do parallel transfer transparently. However, since a grid is heterogeneous from a hardware and a software point of view, we decided to build our own solution in a generic way, it can be used by any kind of data server: SAN, local file systems, NFS or NFSp. The component in charge of transfer across the grid is called GXFER, for Grid Transfer, its goal is to copy files between sites. A copy is done in a parallel way if both sender and receiver can handle it and have distributed storage capability. GXFER can be used as an external program, it will then behave like the classic `scp` command or can be used as a library inside an application.

GXFER performances are good, with a 1Gbytes file transferred in less than 10 seconds, 9.6s, between sites in Grenoble and Lyon connected with a 1Gbits/s link, with NFSp servers on both sides. Further experiments exhibited good scaling properties.

4. Application Domains

4.1. Introduction

Applications in the fields of numerical simulation, image synthesis, and processing are typical of the user demand for high performance computing. In order to confront our proposed solutions for parallel computing with real applications, the project is involved in collaborations with end-users to help them to parallelize their applications.

4.2. Bioinformatics

Keywords: *heterogeneous collection of databanks, protein comparison.*

Participant: Arnaud Legrand.

This joint work involves the GRAAL project.

The problem of searching large-scale genomic sequence databases is an increasingly important bioinformatics problem. We have obtained results on the deployment of such applications in heterogeneous parallel computing environments. These results are based on the analysis of the GriPPS [54], [53] protein comparison application. The GriPPS framework is based on large databases of information about proteins; each protein is represented by a string of characters denoting the sequence of amino acids of which it is composed. Biologists need to search such sequence databases for specific patterns that indicate biologically homologous structures. The GriPPS software enables such queries in grid environments, where the data may be replicated across a distributed heterogeneous computing platform.

In fact, this application is a part of a larger class of applications, in which each task in the application workload exhibits an "affinity" for particular nodes of the targeted computational platform. In the genomic sequence comparison scenario, the presence of the required data on a particular node is the sole factor that constrains task placement decisions. In this context, task affinities are determined by location and replication of the sequence databanks in the distributed platform.

Such biological sequence comparison algorithms are however typically computationally intensive, embarrassingly parallel workloads. In the scheduling literature, this computational model is effectively a *divisible workload scheduling* problem with negligible communication overheads. This framework has enabled us to

propose online scheduling algorithms whose output is *fair* and *efficient*: the slowdown experienced by every user due to the load incurred by the others is as uniform as possible.

4.3. On-demand Geographical Maps

Participant: Jean-Marc Vincent.

This joint work involves the UMR 8504 Géographie-Cité, LSR-IMAG, UMS RIATE and the Maisons de l'Homme et de la Société.

Improvements in the Web developments have opened new perspectives in interactive cartography. Nevertheless existing architectures have some problems to perform spatial analysis methods that require complex calculus over large data sets. Such a situation involves some limitations in the query capabilities and analysis methods proposed to users. The HyperCarte consortium with LSR-IMAG, Géographie-cité and UMR RIATE proposes innovative solutions to these problems. Our approach deals with various areas such as spatio-temporal modelling, parallel computing and cartographic visualization that related to spatial organizations of social phenomena.

Nowadays, analyses are done on huge heterogeneous data set. For example, demographic data sets at nuts 5 level, represent more than 100.000 territorial units with 40 social attributes. Many algorithms of spatial analysis, in particular potential analysis are quadratic in the size of the data set. Then adapted methods are needed to provide "user real time" analysis tools.

4.4. Seismic simulations

Participant: Jean-François Méhaut.

Numerical modelling of seismic wave propagation in complex three-dimensional media is an important research topic in seismology. Several approaches will be studied, and their suitability with respect to the specific constraints of NUMA architectures shall be evaluated. These modelling approaches will rely on modern numerical schemes such as spectral elements, high-order finite differences or finite elements applied to realistic 3D models. The NUMASIS project (see Section 8.2.8) will focus on issues related to parallel algorithms (distribution, scheduling) in order to optimize computations based on such numerical schemes by taking advantage of execution frameworks developed for NUMA architectures.

These approaches will be tested and validated on applications related to seismic risk assessment. Recent seismic events as those in Asia have evidenced the crucial research and development needs in this field. Some regions in France may as well be prone to such risks (French Riviera, Alps, French Antilles,...) and the experiments in the NUMASIS project will be carried out using some of the available data from these regions.

4.5. The CIMENT project

Participant: Olivier Richard.

The CIMENT project (Intensive Computing, Numerical Modeling and Technical Experiments, http://ciment.ujf-grenoble.fr/) gathers a wide scientific community involved in numerical modeling and computing (from numerical physics and chemistry to astrophysics, mechanics, biomodeling and imaging) and the distributed computer science teams from Grenoble. Among these various application domains, there is a huge demand to manage executions of large sets of independent jobs. These sets have between 10,000 to 100,000 jobs each. Providing a middleware able to steer such an amount of jobs is a challenge. The CiGri middleware project addresses this issue in a grid infrastructure (see Section 8.2.6).

5. Software

5.1. Tools for cluster management and software development

The large-sized clusters and grids show serious limitations in many basic system softwares. Indeed, the launching of a parallel application is a slow and significant operation in heterogeneous configurations. The

broadcast of data and executable files is widely under the control of users. Available tools do not scale because they are implemented in a sequential way. They are mainly based on a single sequence of commands applied over all the cluster nodes. In order to reach a high level of scalability, we propose a new design approach based on a parallel execution. We have implemented a parallelization technique based on spanning trees with a recursive starting of programs on nodes. Industrial collaborations were carried out with Mandrake, BULL, HP and Microsoft.

5.1.1. KA-Deploy: deployment tool for clusters and grids

KA-DEPLOY is an environment deployment toolkit that provides automated software installation and reconfiguration mechanisms for large clusters and light grids. The main contribution of KA-DEPLOY 2 toolkit is the introduction of a simple idea, aiming to be a new trend in cluster and grid exploitation: letting users concurrently deploy computing environments exactly fitted to their experiment needs on different sets of nodes. To reach this goal KA-DEPLOY must cooperate with batch schedulers, like OAR, and use a parallel launcher like TAKTUK (see below).

5.1.2. Taktuk: parallel launcher

TAKTUK is a tool to launch or deploy efficiently parallel applications on large clusters, and simple grids. Efficiency is obtained thanks to the overlap of all independent steps of the deployment. We have shown that this problem is equivalent to the well known problem of the single message broadcast. The performance gap between the cost of a network communication and of a remote execution call enables us to use a work stealing algorithm to realize a near-optimal schedule of remote execution calls. Currently, a complete rewriting based on a high level language (precisely Perl script language) is under progress. The aim is to provide a light and robust implementation. This development is lead by the MOAIS project.

5.1.3. NFSp and Gxfer: parallel file system

When deploying a cluster of PCs there is a lack of tools to give a global view of the available space on the drives. This leads to a suboptimal use of most of this space. To address this problem NFSP was developed, as an extension to NFS that divides file system handling in two components: one responsible for the data stored and the other for the metadata, like inodes, access permission.... They are handled by a server, fully NFS compliant, which will contact associated data servers to access information inside the files. This approach enables a full compatibility, for the client side, with the standard in distributed file systems, NFS, while permitting the use of the space available on the clusters nodes. Moreover efficient use of the bandwidth is done because several data servers can send data to the same client node, which is not possible with a usual NFS server. The prototype has now reached a mature state. Sources are available at http://www-id.imag.fr/Laboratoire/Membres/Lombard_Pierre/nfsp/.

5.1.4. aIOLi

As clusters use grows, lots of scientific applications (biology, climatology, nuclear physics ...) have been rewritten to fully exploit this extra CPU power and storage capacity. This kind of software uses and creates huge amounts of data with typical parallel I/O access patterns. Several issues, like *out-of-core limitation* or *efficient parallel input/output access* already known in a local context (on SMP nodes for example), have to be handled in a distributed environment such as a cluster. The effective local hardware facilities which reduced response time and access constraints on SMP could not provide optimal performances with respect to CPU and network power available in a cluster. Several solutions have been proposed by the scientific community to handle these issues, like Parallel File systems or Parallel I/O Libraries, but their specific API limits portability and requires good knowledge of their internal mechanisms.

We have designed AIOLI, an efficient I/O library for parallel access to remote storages in SMP clusters. Thanks to the SMP kernel features, our framework provides parallel I/O without inter-processes synchronization mechanisms as well as a simple interface based on the classic UNIX system calls (create/open/read/write/close). The AIOLI solution allows us to achieve performance close to the limits of the remote storage system. It is presented in more details in section 6.2.

5.1.5. SAMORY

Participants: Yves Denneulin, Jacques Briat.

SAMORY *is an architecture to provide resiliency to parallel applications running on top of virtual clusters, typically built from an intranet or an enterprise network.*

SAMORY is presented in more details in the new results chapter (see Section 6.3).

5.1.6. Generic trace and visualization: Paje

Participants: Arnaud Legrand, Jean-Marc Vincent, Jean-François Méhaut.

This software was formerly developed by members of the Apache project. Even if no real research effort is anymore done on this software, many members of the MESCAL project use it in their everyday research and promote its use. This software is now mainly maintained by Benhur Stein from Federal University Santa Monica (UFSM), Brazil.

PAJE allows applications programmers to define what is visualized and how new objects should be drawn. To achieve such flexibility, the hierarchy of events and the visualization commands may be defined by the programmers inside the applications. The visualization of parallel execution of ATHAPASCAN applications was achieved without any new addition into PAJE software. Inserting few events trace into the ATHAPASCAN runtime allows the visualization of different facets of the program: application computation time but also user task graph management and scheduling of these tasks. PAJE is also, among others, used to visualize Java program execution and large cluster monitoring. PAJE is actively used by the SIMGRID users' community and the NUMASIS project (see Section 8.2.8).

5.1.7. OAR: simple and scalable batch scheduler for clusters and grids

OAR is a batch scheduler that emphasizes simplicity, extensibility, modularity, efficiency, robustness and scalability. It is based on a high level conception that reduces drastically its software complexity. Its internal architecture is built on top of two main components: a generic and scalable tool for the administration of the cluster (launch, nodes administration, ...) and a database as the only way to share information between its internal modules. Completely written in Perl, OAR is also extremely modular and straightforward to extend. Thus, it constitutes a privileged platform to develop and evaluate several scheduling algorithms and new kinds of services.

Most known batch schedulers (PBS, LSF, Condor, ...) are of old-fashioned conception, built monolithically, with the purpose of fulfilling most of the exploitation needs. This results in systems of high software complexity (150000 lines of code for OpenPBS), offering a growing number of functions that are, most of the time, not used. In such a context, it becomes hard to control both the robustness and the scalability of the whole system.

The OAR project focuses on robust and highly scalable batch scheduling for clusters and grids. Its main objectives are the validation of grid administration tools such as TAKTUK, the development of new paradigms for grid scheduling and the experimentation of various scheduling algorithms and policies.

The grid development of OAR has already started with the integration of best effort jobs whose purpose is to take advantage of idle times of the resources. Managing such jobs requires a support of the whole system from the highest level (the scheduler has to know which tasks can be cancelled) down to the lowest level (the execution layer has to be able to cancel awkward jobs). The OAR architecture is perfectly suited to such developments thanks to its highly modular architecture. Moreover, this development is used for the CiGri grid middleware project (see 8.2.6).

The OAR system can also be viewed as a platform for the experimentation of new scheduling algorithms. Current developments focus on the integration of theoretical batch scheduling results into the system so that they can be validated experimentally.

5.2. Simulation tools

5.2.1. SimGrid: simulation of distributed applications

SIMGRID implements realistic fluid network models that enable very fast yet precise simulations. SIMGRID enables the simulation of distributed scheduling agents, which has become critical for current scheduling research in large-scale platforms.

Sources and documentations of SIMGRID are available at the following address http://simgrid.gforge.inria.fr/.

5.2.2. ψ and ψ^2: perfect simulation of Markov Chain stationary distribution

ψ and ψ^2 are two software implementing perfect simulation of Markov Chain stationary distributions using the coupling from the past technique. ψ starts from the transition kernel to derive the simulation program while ψ^2 uses a monotone constructive definition of a Markov chain. They are available at http://www-id.imag.fr/Logiciels/psi/.

5.2.3. PEPS

The main objective of PEPS is to facilitate the solution of large discrete event systems, in situations where classical methods fail. PEPS may be applied to the modelling of computer systems, telecommunication systems, road traffic, or manufacturing systems. The software is available at http://www-id.imag.fr/Logiciels/peps/.

5.3. HyperAtlas

The Hyperatlas software have been jointly developed with LSR-IMAG in the framework of the ESPON European project part 3.1 and 3.2. It includes visualization and analysis of socio-economical data in Europe at Nuts 1, Nuts 2 or Nuts 3 level providing analysis of dependence and spatial interaction. This software is available for European partners at http://www-lsr.imag.fr/HyperCarte/.

6. New Results

6.1. Modelling and Performance evaluation

Participants: Bruno Gaujal, Arnaud Legrand, Brigitte Plateau, Jean-Marc Vincent.

6.1.1. Discrete-Event Control of Stochastic Networks

Multimodularity is a discrete convex property enabling one to show that for a large class of stochastic discrete event systems and under rather natural assumptions on the behavior of the system as well as on the stochastic processes driving its evolution, the smoother the input process, the better the performances of the system.

Of course, the notion smoothness and the performance criteria have to be made precise. This requires several technicalities using several notions from convex analysis, stochastic processes and word combinatorics.

We have developed several tools for control problems with no state information. This is done both in a deterministic setting as well as in very a general stochastic framework. Two classes of problems are handled. In the first one, the control sequence is one dimensional; it covers admission control, service assignment and vacation control problems. Although the control is one dimensional, the systems to which it is applied may be quite general and complex. We focus on general discrete event models which can be described as linear in the max-plus algebra. This type of problems are fully solved, and an optimal policy is identified. A second type of problems involve multi-dimensional control. They cover routing as well as polling problems. These are more difficult to solve. Optimal policies can only be obtained in special cases: the case of symmetrical models and systems with dimension two. The deterministic case as well as the Markovian case have been solved in [49].

A very different kind of technique has been used to solve routing problems in homogeneous grids by using *index policies* “à la Gittins”. Although finding the optimal policy is believed to be exp-time, our approach finds index policies in polynomial time that perform extremely well (in most cases the ratio to the optimal policy is

less than 1 %). These policies are also very robust with respect to the traffic parameters. This work has been done in collaboration with Vandy Berten from Free University of Brussels.

6.1.2. Sampling the Stationary Distribution of Large Queueing Networks

Sampling the stationary distribution of a Markov Chain can be done using coupling from the past. We have shown that this technique can be adapted to the perfect simulation of large queueing networks using GSMP models. Modeling large classes of customer behaviors by index policies has allowed us to assert monotonicity properties by using a structural method rather than tedious ad-hoc proofs [44], [43], [45]. This increases the simulation speed by several orders of magnitude. This new approach has been implemented in ψ^2 and is currently used in the Free University of Brussels as well as the Free University of Amsterdam.

6.1.3. Stochastic Automata Networks

The use of Stochastic Automata Networks (SANs) is becoming increasingly important in performance modelling issues related to parallel and distributed computer systems. As such models become increasingly complex, so also does the complexity of the modelling process. Parallel and distributed systems are often viewed as collections of components that operate more or less independently, requiring only infrequent interaction such as synchronizing their actions, or operating at different rates depending on the state of parts of the overall system. This is exactly the viewpoint adopted by SANs. The components are modelled as individual stochastic automata that interact with each other. Furthermore, the state space explosion problem associated with Markov chain models is mitigated by the fact that the state transition matrix is not stored, nor even generated. Instead, it is represented by a number of much smaller matrices, one for each of the stochastic automata that constitute the system, and from these all relevant information may be determined without explicitly forming the global matrix. The implication is that a considerable saving in memory is effected by storing the matrix in this fashion. This saving is even enforced by the use of functions (of the current global state) as transition rates.

We have proposed an extension of the regular tensor (or Kronecker) operators for matrices [32]. This extension allows to use functions (of the current state of the SAN) as entries of the matrices. It has been shown that certain algebraic properties can be extended from the classical theory. These properties are used to design an efficient algorithm to compute the multiplication of a vector with a tensor-product structured matrix (with functions). The complexity of this operation is $O(N \log N)$ as compared to the complexity of the regular product N^2.

6.1.4. Fair scheduling in large-scale distributed systems

This is a collaborative work with the GRAAL project, the SCALAPPLIX project and the University of California San Diego.

We first have considered in [38] the problem of scheduling comparisons of motifs against biological databanks (see Section 4.2). This problem lies in the divisible load framework with negligible communication costs. Thus far, very few results had been proposed in this model. We have explored the relationship between this model and the preemptive uni-processor one. The requests originating from different users, there is a real need for proposing *fair* scheduling algorithms. After having selected a few relevant metrics (max-stretch and sum-stretch), we have been able to extend algorithms that have been proposed in the literature for the uni-processor model to our setting. We also have proposed original algorithms taking into account the databanks replication. Then we have extensively studied the performance of these algorithms in realistic scenarios. Our study clearly suggests an efficient (within a few percents from the offline optimal solution) online heuristic for each of the two metrics, though a combined optimization is in theory not possible in the general case.

In the previous study, all tasks are supposed to originate from different users. Therefore, all these tasks are in competition and our algorithms ensure that the slowdown incurred by others is as “uniform” as possible. We also have studied a situation where the number of users is small but the number of tasks is very large. Indeed, many applications (cellular micro-physiology, protein conformations, particle detection or others) are constituted of a very large set of independent, equal-sized tasks. All the tasks are generally held by a master who is in charge of distributing it to the different slaves. Some results have been proposed in the past for

optimizing the throughput of a single application on complex heterogeneous platforms. We have extended these results to the multiple application case [22]. In the single-application setting with a tree-shaped platform graph, i.e. when the underlying interconnection network is an oriented tree rooted at the master, it is possible to derive a closed-form formula that characterizes the optimal steady state, which can then be computed via a simple bottom-up traversal of the tree. In fact, this property enables to derive directly autonomous (i.e. where decisions are based only on local informations) task scheduling algorithms. In the multiple applications setting, deriving efficient local algorithms seems however to be much more difficult.

6.2. aIOLi

Participants: Adrien Lebre, Yves Denneulin.

The goal of the AIOLI project is to improve I/O performances on a cluster in a transparent way for parallel applications. This was done in several steps:

1. build a local framework that can do aggregation of requests at the application level. This is done by putting a layer between the application and the kernel in charge of delaying individual requests in order to merge them and thus improve performances. The key factor here is the delay that should be large enough to discover aggregation patterns with a limit to avoid leading to excessive delay. This is done by bounding this delay to a maximum and minimum value that the AIOLI layer has to respect.
2. schedule all I/O requests on a cluster in a global way in order to avoid congestion on a server that leads to bad performances. The idea was to set up a server that would do this work by knowing every I/O operations planned in the system and could thus schedule them. This solution didn't work because false predictions on the duration of a request, necessary to schedule them without overlaps, lead to a suboptimal use of the server and thus low performances.
3. schedule I/O requests locally on the server so that methods of aggregation and mixing of client requests can be used to improve performances. For that AIOLI had to be ported to the kernel and placed at both the VFS level and the lower file system one.

The results have been impressive, with AIOLI giving better performances than the best MPI/IO implementation without any modification of the applications [36] sometimes with a factor of 4. AIOLI can be downloaded from the address http://aioli.imag.fr, both the user library and the Linux kernel module versions.

6.3. SAMORY

Participant: Yves Denneulin.

SAMORY is a runtime aiming at providing resiliency to high performance computing applications running on a virtual cluster, typically hosts of an intranet. It is composed of a Linux kernel module that must be loaded at runtime and a distributed architecture for monitoring, checkpointing and restarting communicating processes. The size of the replication group for a process, the number of copies that will be done for a process, is a parameter that can be fixed, and modified, at runtime depending on the availability of the hosts. The communications between processes are also taken into account by SAMORY and, when a process fails, all pending communications will be transferred transparently to the site where a backup of it will resume.

The main advantage of SAMORY is its total transparency with respect to the applications: starting the monitoring of an application is solely telling the runtime to do so and the applications isn't changed in any way. This can be done at runtime. Introduction of the checkpointing hinders performances but in a reasonable way, the cost of checkpointing a process is directly proportional to the amount of memory it uses with, for example, a checkpoint time of 400ms for a 100Mbytes process on a PC with a Pentium 4 at 2Ghz and 512Mbs of RAM. Virtual memory management, and the appearance of faults, increases these values for large processes, 10s for a 400Mbytes process. By saving only the state related part of the process, excluding code and shared library for example, this cost can be reduced by 75%. The time necessary to restart a process is low, typically milliseconds, since most data will be loaded when necessary and so the execution can resume soon but will

generate page faults later. Since the final time will heavily depend on the behavior of the applications it is not possible to give generic performance results for this step. The overhead on the communications is negligible for small amounts of data and becomes significant for messages of size 8Mbytes.

6.4. Tools for performance evaluation

Participants: Arnaud Legrand, Jean-François Méhaut, Brigitte Plateau, Jean-Marc Vincent.

6.4.1. SimGrid

This is a collaborative work with Martin Quinson from the ALGORILLE project and Henri Casanova from the University of Hawai'i.

The SIMGRID project has been very active and has released a new stable version (3.0). This new version benefits from two major improvements.

- The core of the SIMGRID simulator has completely been rewritten. Using fluid models, ad-hoc data structures and avoiding as much as possible packet-based simulations, we have been able to improve the speed of common simulations by a factor 10. This major modification has also enabled us to easily introduce parallel tasks. On simulations using parallel tasks, the new kernel induces a speedup of more than 300.
- The main drawback of building simulations is that the code is generally a prototype that is hard to reuse. But sometimes, algorithms designed and evaluated with the help of the simulator are finally implemented in a real environment. However some parameters that had been omitted in early simulations may reveal crucial after a real-life implementation. As a matter of fact many back and forth may have to be done before getting to a good modeling and simulation of an application.
  To tackle this issue, the SIMGRID project has just been enriched with a new branch: Grid Reality And Simulation. GRAS provides a complete API to implement distributed application on top of heterogeneous platforms. In addition to the SimGrid implementation of this interface (allowing to work on the application within the comfort of the simulator), an implementation suited to real platforms is also provided (allowing to really use the application once the development is over).

Last, even if the SIMGRID project was already a free software, it has gone even more open than it already was thanks to the use of the INRIA forge (http://simgrid.gforge.inria.fr/). This forge has enabled the growing SIMGRID users community to participate more actively to the development.

6.4.2. Stochastic Automata Networks

New results obtained this year about stochastic automata networks (SANs) concern two issues: The first one is a method to obtain bounds on very large models, when storing the matrix is not possible even on disks. The method consists in computing a bound on an aggregated version of the corresponding Markov chain [17]. These techniques can be applied to computing availability for very large distributed systems. The second result concerns the integration into SANs of a non exponential distribution, namely the PH distribution in order to adjust more precisely to actual parameters. The technique allows to automatically introduce these distributions into a SAN and to keep the tensor formulation to cope with the state space explosion. This is submitted to publication.

6.4.3. Cluster communication models

Clusters of multiprocessor computers (SMP, NUMA,...) are powerful instruments for high-performance applications. However, network contention phenomena have significant impact on applications performance. This impact is increased when cluster nodes are multiprocessors where concurrent communications can income or outcome from the same node. Our interest is to understand the impact of complex communication patterns and to characterize/model the behaviors of concurrent conflicts (over physical links, NIC, DMA, etc...) in term of communication delays. We have proposed degradation functions of communication performance

based on fairness models of network resource sharing [41]. These degradation functions are useful to predict performance and to understand the behavior of applications.

6.5. Dynamic maps on demand

Participant: Jean-Marc Vincent.

This is a collaboration with people from the MOAIS project: Jérémy Allard, Thierry Gautier, Clément Ménier, Denis Trystram, and Jean-Louis Roch.

Geographic researchers create and manipulate maps with various data, changing data type, data correlation, smoothing function, etc. Building a map with acceptable resolution requires around one hour on a standard PC. Interactivity is thus impossible. By parallelizing the application with MPI and then ATHAPASCAN, optimizing the code and selecting suitable libraries, this time was reduced to a few seconds on a parallel cluster. A first prototype was built. One of the scientific problems with this architecture is to take into account previous map generations, in the map request flow. This requires a caching policy which is one of the main speedup factors of the application.

7. Contracts and Grants with Industry

7.1. Collaboration INRIA-BULL: action Dyade LIPS, 03-06

In the context of a global partnership between BULL and INRIA, BULL and the MESCAL project collaborate to develop clustering software solutions aimed at very large computing infrastructures. These clusters feature a complete software environment including management tools, efficient storage solutions and resource management. The partnership promotes the cluster architectures based on the Intel Itanium 2 processor which has established new records for floating point processing. This processor provides the 64-bit wide addressing scheme needed by large data sets of scientific applications and has up to 6 MB of on-chip cache to give the processor superfast access to data. BULL has developed FAME (Flexible Architecture for Multiple Environment) by using standard component assemblies as the building block of larger systems.

7.2. RNTL project IGGI, 04-06

The RNTL project IGGI associates the MESCAL project, the companies BRGM and Mandrake. The goal is to develop middleware for Intranet clustering as an extension to the CLIC RNTL project.

7.3. CIFRE with BULL, 04-06

Adrien Lebre is doing his PhD thesis in a CIFRE contract with the BULL company. His work started in march 2003 and will finish in march 2006 and address the topic of high performance I/O for clusters. His main result is the AIOLI framework presented in section 6.2.

7.4. CIFRE with BULL, 04-06

Maxime Martinasso has started a PhD thesis in January 2004 involving MESCAL and BULL under the terms of a Cifre contract. The topic of this thesis deals with the behavior analysis of Parallel Applications on SMP/NUMA clusters and more specifically on performance modeling of communication contentions and memory accesses.

7.5. CIFRE with BULL, 04-06

Estelle Gabarron has started a PhD thesis in January 2004 involving MESCAL and BULL under the terms of a Cifre contract. The subject addresses issues of the resources management in grids with the presence of large amounts of job in the system. Main issues studied are scalability, error recovery and data management.

8. Other Grants and Activities

8.1. Regional initiatives

8.1.1. CIMENT

The CIMENT project (Intensive Computing, Numerical Modeling and Technical Experiments, http://ciment.ujf-grenoble.fr/) gathers a wide scientific community involved in numerical modeling and computing (from numerical physics and chemistry to astrophysics, mechanics, biomodeling and imaging) and the distributed computer science teams from Grenoble. Several heterogeneous distributed computing platforms were set up (from PC clusters to IBM SP or alpha workstations) each being originally dedicated to a scientific domain. More than 600 processors are available for scientific computation. The MESCAL project provides expert skills in high performance computing infrastructures.

8.1.2. Grappe200 project

MENRT-UJF-INPG (800KF), Rhône-Alpes Region (1.2MF), INRIA (2.5MF), ENS-Lyon (300KF) have funded a 4.8 MF cluster composed of 110 bi-processors Itanium2 connected with a Myrinet (donation of MyriCom) high performance network. This project is lead by MESCAL, MOAIS, ReMaP and SARDES. It is part of the CIMENT project which aims at building high performance distributed grids between several research labs (see above).

8.1.3. Cluster Région

The MESCAL project is member of the regional “cluster” project on computer science and applied mathematics, the focus of its participation is on handling large amount of data large scale architecture. Other members of this subproject are the INRIA GRAAL project, the LSR-IMAG and IN2P3-LAPP laboratories.

8.2. National initiatives

8.2.1. Sure Path, 03-06, ACI SECURITY

Partners (INRIA-Apache, IRISA-Armor, PRISM-Epri).

In the area of distributed systems and networking, the objective of the project is to apply an expertise in mathematical tools, techniques, algorithms and software packages for performance, reliability or dependability studies.

8.2.2. Hypercarte, 02-05, ACI Masse de Données

Partners (Géographie-Cité, LSR-IMAG, UMS RIATE).

The aim of the Hypercarte project is to develop new methods of spatial analysis of socio-economical data and provide the corresponding software tools. These methods take into account either the administrative structure of spatial areas (decomposition at the different nuts levels) or the spatial interaction given by neighborhood (distance on a communication network,...)

Finally, the goal is to build an interactive environment that allows map manipulations by the user. Moreover the environment should depend on the user capacities and the purpose of the study (research, political analysis, territorial management,...). Because of performance, the software architecture is based on a distributed client-server scheme. The server is the front-end of a cluster that realizes computations on the fly and produces directly maps to the user interface. Cache policies to increase parallel code have been implemented and are under test.

8.2.3. Data Grid eXplorer, 03-06, ACI GRID

Partners (LRI, LIP).

The goal of Data Grid Explorer is to build an emulation environment to study large scale configurations. Today, it is difficult to evaluate new models for data placement and caching, network content distribution, peer-to-peer systems, etc. Options include writing simulation environments from scratch, employing detailed

packet-level simulation environments such as NS, local testing within a controlled cluster setting, or deploying live code across the Internet or a Testbed. Each approach has a number of limitations. Custom simulation environments typically simplify network and failure characteristics. Packet-level simulators add more realism but limit system scalability to a few hundred of simultaneous nodes. Cluster-based deployment adds another level of realism by allowing the evaluation of real code, but unfortunately the network is highly over-provisioned and uniform in its performance characteristics. Finally, live Internet and Testbed deployments provide the most realistic evaluation environment for wide-area distributed services. Unfortunately, there are significant challenges to deploying and evaluating real code running at a significant number of Internet sites. The main benefit of emulation is the ability to reproduce experimental conditions and results.

The project is structured horizontally into transverse working groups: Infrastructure, Emulation, Network, and Applications. The Regal team is leader for the Emulation working group.

8.2.4. GEDEON, 04-06, ACI Masse de Données

Partners (IMAG-LSR).

File systems (FS) are commonly used to store data. Especially, they are intensively used in the community of large scientific computing (astronomy, biology, weather prediction) which needs the storage of large amounts of data in a distributed manner. In a GRID context (cluster of clusters), traditional distributed file systems have been adapted to manage a large number of hosts (like the Andrew File System). However, such file systems remain inadequate to manage huge data. They are suited for traditional Unix (small) files. Thus, the grain of distribution is typically an entire file and not a piece of file which is essential for large files. Furthermore, the tools for managing data (e.g, interrogation, duplication, consistency) are unsuited for large sizes.

Database Management Systems (DBMS) provides different abstraction layers, high level languages for data interrogation and manipulation etc. However, the imposed data structure, the low distribution, and the usually monolithic architecture of DBMSs limit their utilization in the scientific computing context.

The main idea of the Gedeon project is to merge the functions of file systems and DBMS, focusing on structuration of meta-data, duplication and coherency control. Our goal is NOT to build a DBMS describing a set of files. We will study how database management services can be used to improve the efficiency of file access and to increase the functionality provided to scientific programmers.

8.2.5. GRID 5000, 04-07, ACI GRID

Partners (INRIA FUTURS, INRIA Sophia, IRISA, LORIA, IRIT, LABRI, LIP, LIFL).

The foundations of Grid'5000 have emerged from a thorough analysis and numerous discussions about methodologies used for scientific research in the Grid domain. A report presents the rationale for Grid'5000. In addition to theory, simulators and emulators, there is a strong need for large scale testbeds where real life experimental conditions hold. The size of Grid'5000, in terms of number of sites and number of CPUs per site, was established according to the scale of the experiments and the number of researchers involved in the project.

8.2.6. Cigri, 02-04, ACI GRID

Partners (CIMENT).

The aim of the CiGri project is to gather the unused computing resource from intranet infrastructure and to make it available for large scale applications. This grid is based on two software tools. The CiGri server software is based on a database and offers a user interface for launching grid computations (scripts and web tools). It interacts with the computing clusters through a batch scheduler software. CiGri is compatible with classical batch systems like PBS but an efficient batch software (OAR, http://oar.imag.fr/) has been developed by MESCAL and MOAIS Project for easy integrations and experimentations of scheduling tools.

8.2.7. DSLLab, 2005-2007, ANR Jeunes Chercheurs

Partners (INRA-FUTURS).

DSLlab is a research project aiming at building and using an experimental platform about distributed systems running on DSL Internet. The objective is twofold:

- provide accurate and customized measures of availability, activity and performances in order to characterize and tune the models of the ADSL resources;
- provide a validation and experimental tool for new protocols, services and simulators and emulators for these systems.

DSLlab consists of a set of low power, low noise computers spread over the ADSL. These computers are used simultaneously as active probes to capture the behavior traces, and as operational nodes to launch experiments. We expect from this experiment a better knowledge of the behavior of the ADSL and the design of accurate models for emulation and simulation of these systems which represents now a significant capability in terms of storage and computing power.

8.2.8. NUMASIS, 2005-2008, ANR Calcul Intensif et Grilles de Calcul

Future generations of multiprocessors machines will rely on a NUMA architecture featuring multiple memory levels as well as nested computing units (multi-core chips, multithreaded processors, multi-modules NUMA, etc.). To achieve most of the hardware's performance, parallel applications need powerful software to carefully distribute processes and data so as to limit non-local memory accesses. The ANR NUMASIS[2] project aims at evaluating the functionalities provided by current operating systems and middleware in order to point out their limitations. It also aims at designing new methods and mechanisms for an efficient scheduling of processes and a clever data distribution on such platforms. These mechanisms will be implemented within operating systems and middleware. The target application domain is seismology, which is very representative of the needs of computer-intensive scientific applications.

8.2.9. ALPAGE, 2005-2008, ARA Masses de Données

The new algorithmic challenges associated with large-scale platforms have been approached from two different directions. On the one hand, the parallel algorithms community has largely concentrated on the problems associated with heterogeneity and large amounts of data. Algorithms have been based on a centralized single-node, responsible for calculating the optimal solution; this approach induces significant computing times on the organizing node, and requires centralizing all the information about the platform. Therefore, these solutions clearly suffer from scalability and fault tolerance problems.

On the other hand, the distributed systems community has focused on scalability and fault-tolerance issues. The success of file sharing applications demonstrates the capacity of the resulting algorithms to manage huge volumes of data and users on large unstable platforms. Algorithms developed within this context are completely distributed and based on peer-to-peer communications. They are well adapted to very irregular applications, for which the communication pattern is unpredictable. But in the case of more regular applications, they lead to a significant waste of resources.

The goal of the ALPAGE project is to establish a link between these directions, by gathering researchers (ID, LIP, LORIA, LaBRI, LIX, LRI) from the distributed systems and parallel algorithms communities. More precisely, the objective is to develop efficient and robust algorithms for some elementary applications, such as broadcast and multicast, distribution of tasks that may or may not share files, resource discovery. These fundamental applications correspond well to the spectrum of the applications that can be considered on large scale, distributed platforms.

8.2.10. SMS, 2005-2008, ANR

The ACI SMS, "Simulation et Monotonie Stochastique en évaluation de performances", is composed by two teams: Performance Evaluation team from PRiSM Laboratory (ACI Leader) and the MESCAL project. The main objective is to study monotonicity properties of computer systems models in order to speed up the simulations and estimate performance indexes more accurately.

The composition formalisms we have contributed to develop during the recent years allow to build large Markov chains associated to complex systems in order to analyze their performance. However, it is often

[2]NUMASIS: Adapting and Optimizing Applicative Performance on NUMA Architectures: Design and Implementation with Applications in Seismology

impossible to solve the stationary or transient distributions. Analytical methods and simulations fail for different reasons.

However brute performances are not really useful. We need the proof that the system is better than an objective. Therefore it is natural to use comparison of random variables and sample-paths. Two important concepts appear: stochastic ordering and stochastic monotony. We chose to develop these two important concepts and apply them to perfect simulation, distributed simulation and product form queuing network. These concepts seem to appear frequently in various techniques in performance evaluation. Using the monotony property, one can reduce the computation time for perfect simulation with coupling from the past. Coupling from the past allows to sample the steady-state distribution in a finite time. Thus we do not encounter the same stopping problem that hold for ordinary simulations. Furthermore, some results show that the monotony property is often present in queuing network even if they do not have product form. We simply have to renormalize them to let the property appear. Using both properties it is also possible to derive distributed simulations which will be more efficient. We will develop two ideas: sample-path transformations to avoid rollback in optimistic simulations (and we compute a bound) and regenerative simulations.

Finally, these concepts can be used for product form queuing network to explain why some transformation applied on customer synchronization can provide product form solution and also how we can compute a solution of the traffic equation when they are unstable.

8.3. International initiatives

8.3.1. Europe

CoreGrid: The project MESCAL participates to the Network Of Excellence CoreGrid.

EuroNGI : The project MESCAL participates to the Network Of Excellence EuroNGI (Next Generation Internet).

ESPON : The MESCAL project participates to the ESPON (European Spatial Planning Observation Network) http://www.espon.lu/ It is involved in the action 3.1 on tools for analysis of socio-economical data. This work is done in the consortium hypercarte including the laboratories LSR-IMAG (UMR 5526), Géographie-cité (UMR 8504) and RIATE (UMS 2414). The Hyperatlas tools have been applied to the European context in order to study spatial deviation indexes on demographic and sociologic data at nuts 3 level.

PAI Van Gogh : The project is involved in a PAI Van-Gogh with Leiden University (2004-2005). This work is build around three research axis (generalizations and extensions of our previous results, optimization of monotonous and homogeneous discrete event systems and applications to communication networks).

PAI Germaine de Stael : The project is involved in a PAI Germaine de Stael with EPFL (2004-2005). This collaboration was on the use of network calculus techniques to assess real-time guaranties in embedded systems.

8.3.2. Africa

Cameroon : MESCAL takes part in the SARIMA[3] project an more precisely with the University of Yaoundé 1. Two Cameroon students (Jean-Michel NLong 2 and Blaise Yenké) are preparing their PhD in cotutelle (joint and remote supervision) with Professor Maurice Tchuenté. SARIMA also funded Adamou Hamza to prepare his Master Thesis during three months in the MESCAL project. SARIMA proposed J-F Méhaut to give a course on Operating System and Networks at Master Research Students.

[3]Soutien aux Activités de Recherche Informatique et Mathématiques en Afrique http://www-direction.inria.fr/international/AFRIQUE/sarima.html

8.3.3. North America

- University of California, San Diego: Arnaud Legrand has been working for a few months with Henri Casanova, Larry Carter and Jeanne Ferrante at UCSD. This collaboration was following a long-term collaboration that had been officialized by a NSF-INRIA grant between the GRAIL laboratory (UCSD) and the GRAAL project. The result of this visit is presented in Section 6.4.1 and 6.1.4.
- NSF Project with W. Stewart (NC State University), G. Ciardo (College William and Mary), S. Donatelli (U. de Turin), 2002-2006. The purpose of the project is to study structured methods for Markov chains in order to evaluate the performances of distributed systems.

8.3.4. South America

- PICS (2005-2007) CADIGE funded by the CNRS with the universities of Rio Grande do Sul, Brazil (UFRGS, UFSM, PUC, UNISINOS), around PC clusters, grid and performance evaluation tools.
- CAPES/COFECUB grant (2006-2008) with the UFRGS, Porto Alegre, Brazil around grid and PC clusters.
- Colombia: collaboration with the universities of Los Andes, Bogota, and UIS, Bucaramanga, on the topic of grids for computation and data management.

8.4. High Performance Computing Center

8.4.1. The ICluster2 and IDPot Platforms

The MESCAL project manages a cluster computing center on the Grenoble campus. The center manages different architectures: a 48 bi-processors PC (ID-POT), and the center is involved with a cluster based on 110 bi-processors Itanium2 (ICluster-2) located at INRIA.

More than 60 research projects in France have used the architectures, especially the 204 processors Icluster-2. Half of them have run typical numerical applications on this machine, the remainder has worked on middleware and new technology for cluster and grid computing.

The ICluster2 and IDPot platforms are now integrated the Grid'5000 grid platform.

8.4.2. The BULL Machine

In the context of our collaboration with BULL (LIPS, NUMASIS), the MESCAL project acquired a Novascale NUMA machine. The configuration is based on 8 Itanium II processors at 1.5 Ghz and 16 GB of RAM. This platform is mainly used by the BULL PhD students. This machine is also connected to the CIMENT Grid.

8.4.3. GRID 5000 and CIMENT

The MESCAL project is involved in development and management of Grid'5000 platform. The ICluster2 and IDPot clusters are integrated in Grid'5000. Moreover, these two clusters take part in CIMENT Grid. More precisely, their unused resources may exploited to executed job form partners of CIMENT project (see Section 8.1.1).

9. Dissemination

9.1. Leadership within scientific community

9.1.1. Program committees

Researchers of the MESCAL project have been members of the following program committees:

- Grid 2005 - 6th IEEE/ACM International Workshop on Grid Computing;

- CDUR'2005 - Journées Francophones sur la Cohérence des Données en Univers Réparti (IEEE/ACM Sigops France);
- GAN'2005 and 2006 - Third and Fourth Workshop on Grids and Advanced Networks GAN'05 (workshops of IEEE/ACM International Symposium on Cluster Computing and the Grid);
- VECPAR'06 - Seventh International Meeting on High Performance Computing for Computational Science;
- VLDB Workshop proposal "Data Management in Grids", 05
- 2nd European Performance Evaluation Workshop, 05, 06

9.1.2. PAGE: Probabilities and Applications in Grenoble and its surroundings

This seminar on probabilities and applications is targeted toward computer scientists as well as mathematicians. One of the goals is to encourage collaborations between people from different laboratories with varied backgrounds. More informations are available at http://www-lmc.imag.fr/lmc-sms/Bernard.Ycart/page/.

9.1.3. Grenoble's Seminar on performance evaluation

This seminar is organized by Jean-Marc Vincent and Bruno Gaujal. It is tightly coupled with the PAGE group and its main goal is to organize meetings between the various researchers of Grenoble using the same kind of mathematical tools (stochastic models, queuing networks, Petri networks, stochastic automata, Markovian process and chains, (max,+) algebra, fluid systems, ...). On the long term, this seminar should lead to inter-laboratory working groups on precise themes. More informations are available at http://www-id.imag.fr/Laboratoire/Membres/Vincent_Jean-Marc/EPG/.

9.2. Teaching

Members of the MESCAL team are actively involved in teaching. Their activities are balanced between graduate students and post-graduate students. Here are a few examples of their responsibilities:

- **2nd year of Research Master (Grenoble): Operating Systems and Software** head of the SAP track (operating systems, parallel and distributed applications, networks and multimedia). Here is a list of courses taught by researchers of the MESCAL project:
  - Cluster architectures for high-performance computing and high throughput data management.
  - Data measurement and analysis for network and operating systems performance evaluation.
  - Modeling and simulation for network and operating systems performance evaluation.
  - Building parallel and distributed applications (contributor).
  - Algorithms and basic techniques for parallel computing (contributor).
- **2nd year of Research Master (Paris): MPRI** Network algorithms
- **2nd year of Research Master (Yaoundé)** Operating systems and networks.
- **Magistère d'informatique Licence (Université Joseph Fourier)**

10. Bibliography

Major publications by the team in recent years

[1] E. ALTMAN, B. GAUJAL, A. HORDIJK. *Discrete-Event Control of Stochastic Networks: Multimodularity and Regularity*, LNM, nº 1829, Springer-Verlag, 2003.

[2] K. ATIF, B. PLATEAU. *Stochastic Automata Network for modeling parallel systems*, in "IEEE Transactions on Software Engineering", vol. 17, nº 10, October 1991.

[3] B. GAUJAL, S. HAAR, J. MAIRESSE. *Blocking a Transition in a Free Choice Net, and what it tells about its throughput*, in "Journal of Computer and System Sciences", vol. 66, nº 3, 2003, p. 515-548.

[4] J.-M. VINCENT. *Some Ergodic Results on Stochastic Iterative Discrete Event Systems*, in "Discrete Event Dynamic Systems", vol. 7, nº 2, 1997, p. 209-232.

Doctoral dissertations and Habilitation theses

[5] F. CLÉVENOT-PERRONNIN. *Fluid Models for Content Distribution Systems*, Ph. D. Thesis, University of Nice-Sophia Antipolis, 2005.

[6] E. EDI. *Caches de calculs pour des serveurs webs de requêtes dynamiques*, Ph. D. Thesis, INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE - INPG, 12 2005.

[7] R. B. ÁVILA. *Un modèle de distribution de serveur de fichiers pour grappes*, Ph. D. Thesis, Institut National Polytechnique de Grenoble et Université Fédérale du Rio Grande do Sul, June 2005.

Articles in refereed journals and book chapters

[8] O. BEAUMONT, H. CASANOVA, A. LEGRAND, Y. ROBERT, Y. YANG. *Scheduling divisible loads on star and tree networks: results and open problems*, in "IEEE Trans. Parallel Distributed Systems", vol. 16, nº 3, 2005, p. 207-218.

[9] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Pipelining broadcasts on heterogeneous platforms*, in "IEEE Trans. Parallel Distributed Systems", vol. 16, nº 4, 4 2005.

[10] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Steady-state scheduling on heterogeneous clusters*, in "Int. J. of Foundations of Computer Science", vol. 16, nº 2, 2005.

[11] A. BENOIT, B. PLATEAU, W. STEWART. *Réseaux d'automates stochastiques à temps discret*, in "TSI", 2005.

[12] N. BERNARD, Y. DENNEULIN, S. VARETTE. *La sécurité et le multimédia*, chap. Sécurité Réseau, Hermes, 2005.

[13] N. BERNARD, Y. DENNEULIN, S. VARETTE. *La sécurité et le multimédia*, chap. Sécurité Unix, Hermes, 2005.

[14] A. BOUILLARD, B. GAUJAL, J. MAIRESSE. *Extremal throughputs in free-choice nets*, in "Journal of Discrete Event Dynamic Systems", Accepted for publication, 2005.

[15] F. CAPPELLO, Y. DENNEULIN, J.-F. MÉHAUT, G. FEDAK, O. LODYGENSKY, G. ANTONIU, L. BOUGÉ, M. JAN, T. PRIOL, E. CARON, F. DESPREZ, N. EMAD, S. PETITON. *Informatique répartie*, D. TRYSTRAM, Y. SLIMANI, M. JEMNI (editors). , chap. Les grilles, Hermes, 2005, p. 120-160.

[16] F. CLÉVENOT-PERRONNIN, P. NAIN, K. W. ROSS. *Stochastic Fluid Models for Cache Clusters*, in "Performance Evaluation", vol. 59, n^o 1, January 2005, p. 1-18.

[17] J.-M. FOURNEAU, B. PLATEAU, I. SBEITY, W. STEWART. *SANs and Lumpable Stochastic Bounds: Bounding Availability*, in "Computer System, Network Performance and Quality of Service, Editeur Imperial College Press", 2005.

[18] B. GAUJAL, E. HYON, A. JEAN-MARIE. *Optimal Routing in two parallel queues with exponential service times*, in "Journal of Discrete Event Dynamic Systems", To appear in the special issue of selected papers from WODES, 2005.

[19] B. GAUJAL, N. NAVET. *Fault Confinement Mechanisms on CAN (Controller Area Network): Analysis and Improvements*, in "IEEE Transactions on Vehicular Technology", vol. 54, n^o 3, 2005.

[20] B. GAUJAL, N. NAVET. *Maximizing the robustness of TDMA networks with applications to TTP/C*, in "Real Time Systems", to appear, 2005.

[21] B. GAUJAL, N. NAVET, C. WALSH. *Shortest Path Algorithms for Real-Time Scheduling of FIFO Tasks with Minimal Energy Use*, in "ACM Transactions on Embedded Computing Systems", Accepted for publication, 2005.

[22] A. LEGRAND, L. MARCHAL, Y. ROBERT. *Optimizing the steady-state throughput of scatter and reduce operations on heterogeneous platforms*, in "J. Parallel and Distributed Computing", to appear, 2005.

[23] C. MARTIN, O. RICHARD, G. HUARD. *Déploiement adaptatif d'applications parallèles*, in "Technique et Science Informatiques (TSI)", 2005.

Publications in Conferences and Workshops

[24] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Independent and divisible tasks scheduling on heterogeneous star-schaped platforms with limited memory*, in "PDP'2005, 13th Euromicro Workshop on Parallel, Distributed and Network-based Processing", IEEE Computer Society Press, 2005, p. 179-186.

[25] A. BOUILLARD, B. GAUJAL. *Throughput in stochastic free-choice nets*, in "44th IEEE Conference on Decision and Control and European Control Conference ECC, Sevilla, Spain", 2005.

[26] A. BOUILLARD, B. GAUJAL, J. MAIRESSE. *Extremal throughputs in free-choice nets*, in "26th International Conference On Application and Theory of Petri Nets and Other Models of Concurrency, Miami", LNCS, Springer-Verlag, 2005.

[27] J. BOULIER, J.-M. VINCENT. *Neighborhood, Interaction and Continuous Spatial Structures*, in "Technical meeting on multiscales analysis of environmental landscape analysis, Copenhagen", European Environment Agency, May 2005.

[28] N. CAPIT, G. D. COSTA, Y. GEORGIOU, G. HUARD, C. MARTIN, G. MOUNIÉ, P. NEYRON, O. RICHARD. *A batch scheduler with high level components*, in "Cluster computing and Grid 2005 (CCGrid05)", 2005.

[29] F. CAPPELLO, F. DESPREZ, M. DAYDE, E. JEANNOT, Y. JEGOU, S. LANTERI, N. MELAB, R. NAMYST, P. PRIMET, O. RICHARD, E. CARON, J. LEDUC, G. MORNET. *Grid'5000: A Large Scale, Reconfigurable, Controlable and Monitorable Grid Platform*, in "Grid2005 6th IEEE/ACM International Workshop on Grid Computing", 2005.

[30] F. CLÉVENOT-PERRONNIN, P. NAIN. *Stochastic Fluid Model for P2P Caching Evaluation*, in "Proc. WCW 2005, Sophia Antipolis, France", September 2005, p. 104-111.

[31] F. CLÉVENOT-PERRONNIN, P. NAIN, K. W. ROSS. *Multiclass P2P Networks: Static Resource Allocation for Service Differentiation and Bandwidth Diversity*, in "Proc. PERFORMANCE 2005, Juan-les-Pins, France", October 2005, p. 32-49.

[32] J.-M. FOURNEAU, B. PLATEAU, I. SBEITY, W. STEWART. *Tensor products and bounds for stochastic automata networks.*, in "SIAM Conference on Computational Science and Engineering, Orlando, Florida, USA", February 2005.

[33] B. GAUJAL, A. HORDIJK. *On dropping sequences for RED*, in "1st Euro New Generation Internet Conference, Rome, Italy", 2005.

[34] C. GRASLAND, H. MARTIN, J.-M. VINCENT, J. GENSEL, H. MATHIAN, S. OULAHAL, O. CUENOT, E. EDI, L. LIZZI. *Le projet Hypercarte : analyse spatiale et cartographie interactive.*, in "SAGEO, Avignon", June 2005.

[35] R. KASSICK, C. MACHADO, E. HERMANN, R. B. ÁVILA, P. O. NAVAUX, Y. DENNEULIN. *Evaluation the performance of the dNFSP file system*, in "Proceedings of CCGrid 2005, Cardiff, Pays de Galles", 2005.

[36] A. LEBRE, Y. DENNEULIN. *aIOLi: An Input/Output LIbrary for cluster of SMP*, in "Proceedings of CCGrid 2005, Cardiff, Pays de Galles", 2005.

[37] A. LEGRAND. *Scheduling competing regular applications on a heterogeneous master-worker platforms*, in "NSF/INRIA Workshop: Scheduling for Large-Scale Distributed Platforms, La Jolla, California", 11 2005.

[38] A. LEGRAND, A. SU, F. VIVIEN. *Off-line scheduling of divisible requests on an heterogeneous collection of databanks*, in "Proceedings of the 14th Heterogeneous Computing Workshop, Denver, Colorado, USA", IEEE Computer Society Press, April 2005.

[39] C. MARCHAND, J.-M. VINCENT. *Observations d'exécutions d'algorithmes de consensus sur réseaux sans fil en mode ad-hoc*, in "CFSE, Le Croisic", 2005.

[40] C. MARCHAND, J.-M. VINCENT. *Performance Tuning of Failure Detectors in Wireless Ad-hoc Networks : modeling and experiments*, in "Formal Techniques for Computer Systems and Business Process, Epew 2005, Versailles", LNCS (editor). , vol. 3670, August 2005, p. 139-154.

[41] M. MARTINASSO, J.-F. MÉHAUT. *Prediction of Communication Latency over Complex Network Behaviors on SMP Clusters*, in "European Performance Engineering Workshop, Versailles", Lecture Notes in Computer Science, nº 3670, Springer Verlag, September 2005.

[42] P. POULLET, P. NUIRO, J.-F. MÉHAUT. *Parallélisation de méthodes itératives utilisant un préconditionneur multiniveaux*, in "2ème Congrès National de Mathématiques Appliquées et Industrielles SMAI'2005, Evian", Société de Mathématiques Appliquées et Industrielles, May 2005.

[43] J.-M. VINCENT, J. VIENNE. *Perfect simulation of index based routing queueing networks*, in "Performance, Antibes", poster, October 2005.

[44] J.-M. VINCENT. *Perfect simulation of monotone queueing networks*, in "IFIP WG 7.3, Antibes", October 2005.

[45] J.-M. VINCENT. *Perfect Simulation of Monotone Systems for Rare Event Probability Estimation*, in "Winter Simulation Conference, Orlando", 2005.

[46] J.-M. VINCENT. *Perfect simulation of queueing networks with blocking and rejection*, in "Saint IEEE conference, Trento", 2005, p. 268-271.

Internal Reports

[47] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Scheduling multiple bags of tasks on heterogeneous master-worker platforms: centralized versus distributed solutions*, Technical report, nº 2005-45, LIP, 2005, http://www.ens-lyon.fr/LIP/Pub/Rapports/RR/RR2005/RR2005-45.pdf.

[48] A. BOUILLARD, B. GAUJAL. *Exact Simulation of Fork-Join networks*, Technical report, nº 2005-12, LIP ENS-Lyon, 2005.

[49] B. GAUJAL, A. HORIDIJK, D. VAN DER LAAN. *On the optimal policy for deterministic and exponential polling systems*, Technical report, Free University of Amsterdam, 2005.

[50] A. LEBRE, Y. DENNEULIN. *aIOLi: gestion des Entrées/Sorties Parallèles dans les grappes SMP*, Technical report, INRIA, March 2005, http://www.inria.fr/rrrt/rr-5522.html.

[51] A. LEGRAND, Y. YANG, H. CASANOVA. *NP-Completeness of the Divisible Load Scheduling Problem on Heterogeneous Star Platforms with Affine Costs*, Technical report, nº CS2005-0818, UCSD CSE, March 2005.

Bibliography in notes

[52] *The GridFTP Protocol and Software*, 2002, http://www.globus.org/.

[53] *GriPPS webpage at , http://gripps.ibcp.fr/*, 2005.

[54] C. BLANCHET, C. COMBET, C. GEOURJON, G. DELÉAGE. *MPSA: Integrated System for Multiple Protein Sequence Analysis with client/server capabilities*, in "Bioinformatics", vol. 16, nº 3, 2000, p. 286-287.