INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team GRAAL

Algorithms and Scheduling for Distributed Heterogeneous Platforms

Grenoble - Rhône-Alpes

Theme : Distributed and High Performance Computing

Table of contents

The GRAAL *project-team is common to CNRS, ENS Lyon, and INRIA. This team is part of the Laboratoire de l'Informatique du Parallélisme (LIP), UMR ENS Lyon/CNRS/INRIA/UCBL 5668. The team is located in part at the École normale supérieure de Lyon and in part at the Université Claude Bernard – Lyon 1.*

1. Team

Research Scientists

Frédéric Desprez [Senior Researcher (DR), HdR]
Gilles Fedak [Junior Researcher (CR)]
Jean-Yves L'Excellent [Junior Researcher (CR)]
Loris Marchal [Junior Researcher (CR)]
Christian Pérez [Junior Researcher (CR), HdR]
Bora Uçar [Junior Researcher (CR)]
Frédéric Vivien [Team Leader, Senior Researcher (DR), HdR]

Faculty Members

Anne Benoît [Associate Professor (MCF), HdR]
Yves Caniou [Associate Professor (MCF)]
Eddy Caron [Associate Professor (MCF), HdR]
Bernard Tourancheau [Professor, HdR]
Yves Robert [Professor, HdR]

External Collaborators

Alexandru Dobrila [PhD student, MENRT grant]
Jean-Marc Nicod [Associate Professor, HdR]
Laurent Philippe [Professor, HdR]
Lamiel Toch [PhD student, MENRT grant]

Technical Staff

Nicolas Bard
Florent Chuffart
Benjamin Depardon [From October 15 to November 14, 2010]
Haiwu He
Benjamin Isnard [Until February 28, 2010]
Guillaume Joslin
Gaël Le Mahec [Until March 31, 2010]
José Francisco Saray Villamizar [Since October 15, 2010]
Daouda Traore [Until March 31, 2010]

PhD Students

Leila Ben Saad [MENRT grant]
Julien Bigot [MENRT grant until September 30, 2010 - INRIA from October 1 to December 31, 2010]
Ghislain Charrier [INRIA Cordi-S grant until October 31, 2010]
Benjamin Depardon [MENRT grant until September 30, 2010]
Fanny Dufossé [ENS grant]
Matthieu Gallet [ENS grant, until September 30, 2010]
Sylvain Gault [INRIA, since November 1, 2010]
Cristian Klein [INRIA grant]
Mathias Jacquelin [MENRT grant]
George Markomanolis [INRIA Cordi-S grant]
Vincent Pichon [CIFRE EDF R&D grant]
Georges Markomanolis [INRIA Cordi-S grant]
Adrian Muresan [MENRT grant]
Paul Renaud-Goud [MENRT grant]

Clément Rezvoy [MENRT grant]

Post-Doctoral Fellows

Marin Bougeret [Since October 1, 2010]
Hinde Bouziane [Until August 31, 2010]
Laurent Bobelin [Until March 4, 2010]
Indranil Chowdhury [Until March 5, 2010]
Simon Delamare [Since October 15, 2010]
Luis Rodero-Merino [From January 11 to December 31, 2010]
Mark Stillwell [Since December 1, 2010]

Visiting Scientists

Domingo Jimenez [From May 27 to June 22, 2010]
Mohamed Labidi [From June 20 to July 20, 2010]
Melhem Rami [From May 9 to June 6, 2010]
Mircea Moca [From April 1 au June 30, 2010]
Lu Lu [From September 01, 2010 to March 1, 2011]

Administrative Assistant

Evelyne Blesle [INRIA, 50% on the project]

2. Overall Objectives

2.1. Introduction

Parallel computing has spread into all fields of applications, from classical simulation of mechanical systems or weather forecast to databases, video-on-demand servers or search tools like Google. From the architectural point of view, parallel machines have evolved from large homogeneous machines to clusters of PCs (with sometimes boards of several processors sharing a common memory, these boards being connected by high speed networks like Myrinet). However, the need of computing or storage resources has continued to grow leading to the need of resource aggregation through Local Area Networks (LAN) or even Wide Area Networks (WAN). The recent progress of network technology has enabled the use of highly distributed platforms as a single parallel resource. This has been called Metacomputing or more recently Grid Computing [85]. An enormous amount of financing has recently been put into this important subject, leading to an exponential growth of the number of projects, most of them focusing on low level software detail. We believe that many of these projects failed to study fundamental issues such as the computational complexity of problems and algorithms and heuristics for scheduling problems. Also they usually have not validated their theoretical results on available software platforms.

From the architectural point of view, Grid Computing has different scales but is always highly heterogeneous and hierarchical. At a very large scale, tens of thousands of PCs connected through the Internet are aggregated to solve very large applications. This form of the Grid, usually called a Peer-to-Peer (P2P) system, has several incarnations, such as SETI@home, Gnutella or XtremWeb [94]. It is already used to solve large problems (or to share files) on PCs across the world. However, as today's network capacity is still low, the applications supported by such systems are usually embarrassingly parallel. Another large-scale example is TeraGRID which connects several supercomputing centers in the USA and reaches a peak performance of over 100 Teraflops. At a smaller scale but with a high bandwidth, one can mention the Grid'5000 project, which connects PC clusters spread in nine French university research centers. Many such projects exist over the world that connect a small set of machines through a fast network. Finally, at a research laboratory level, one can build an heterogeneous platform by connecting several clusters using a fast network such as Myrinet.

The common problem of all these platforms is not the hardware (these machines are already connected to the Internet) but the software (from the operating system to the algorithmic design). Indeed, the computers connected are usually highly heterogeneous (from clusters of SMPs to the Grid).

There are two main challenges for the widespread use of Grid platforms: the development of environments that will ease the use of the Grid (in a seamless way) and the design and evaluation of new algorithmic approaches for applications using such platforms. Environments used on the Grid include operating systems, languages, libraries, and middlewares [83], [85], [87]. Today's environments are based either on the adaptation of "classical" parallel environments or on the development of toolboxes based on Web Services.

Aims of the GRAAL project.

In the GRAAL project we work on the following research topics:

- algorithms and scheduling strategies for heterogeneous and distributed platforms,
- environments and tools for the deployment of applications over service oriented platforms.

The main keywords of the GRAAL project:
Algorithmic Design + Middleware/Libraries + Applications
over Heterogeneous and Distributed Architectures

2.2. Highlights of the year

- Frédéric Vivien was promoted Senior researcher.

3. Scientific Foundations

3.1. Scheduling Strategies and Algorithm Design for Heterogeneous Platforms

Participants: Anne Benoît, Marin Bougeret, Hinde Bouziane, Alexandru Dobrila, Fanny Dufossé, Matthieu Gallet, Mathias Jacquelin, Loris Marchal, Jean-Marc Nicod, Laurent Philippe, Paul Renaud-Goud, Clément Rezvoy, Yves Robert, Mark Stillwell, Bora Uçar, Frédéric Vivien.

Scheduling sets of computational tasks on distributed platforms is a key issue but a difficult problem. Although a large number of scheduling techniques and heuristics have been presented in the literature, most of them target only homogeneous resources. However, future computing systems, such as the computational Grid, are most likely to be widely distributed and strongly heterogeneous. Therefore, we consider the impact of heterogeneity on the design and analysis of scheduling techniques: how to enhance these techniques to efficiently address heterogeneous distributed platforms?

The traditional objective of scheduling algorithms is the following: given a task graph and a set of computing resources, or *processors*, map the tasks onto the processors, and order the execution of the tasks so that: (i) the task precedence constraints are satisfied; (ii) the resource constraints are satisfied; and (iii) a minimum schedule length is achieved. Task graph scheduling is usually studied using the so-called *macro-dataflow* model, which is widely used in the scheduling literature: see the survey papers [84], [93], [96], [97] and the references therein. This model was introduced for homogeneous processors, and has been (straightforwardly) extended to heterogeneous computing resources. In a word, there is a limited number of computing resources, or processors, to execute the tasks. Communication delays are taken into account as follows: let task T be a predecessor of task T' in the task graph; if both tasks are assigned to the same processor, no communication overhead is incurred, the execution of T' can start immediately at the end of the execution of T; on the contrary, if T and T' are assigned to two different processors P_i and P_j, a communication delay is incurred. More precisely, if P_i completes the execution of T at time-step t, then P_j cannot start the execution of T' before time-step $t + \text{comm}(T, T', P_i, P_j)$, where $\text{comm}(T, T', P_i, P_j)$ is the communication delay, which depends upon both tasks T and T', and both processors P_i and P_j. Because memory accesses are typically several orders of magnitude cheaper than inter-processor communications, it is sensible to neglect them when T and T' are assigned to the same processor.

The major flaw of the macro-dataflow model is that communication resources are not limited in this model. Firstly, a processor can send (or receive) any number of messages in parallel, hence an unlimited number of communication ports is assumed (this explains the name *macro-dataflow* for the model). Secondly, the number of messages that can simultaneously circulate between processors is not bounded, hence an unlimited number of communications can simultaneously occur on a given link. In other words, the communication network is assumed to be contention-free, which of course is not realistic as soon as the number of processors exceeds a few units.

The general scheduling problem is far more complex than the traditional objective in the *macro-dataflow* model. Indeed, the nature of the scheduling problem depends on the type of tasks to be scheduled, on the platform architecture, and on the aim of the scheduling policy. The tasks may be independent (e.g., they represent jobs submitted by different users to a same system, or they represent occurrences of the same program run on independent inputs), or the tasks may be dependent (e.g., they represent the different phases of a same processing and they form a task graph). The platform may or may not have a hierarchical architecture (clusters of clusters vs. a single cluster), it may or may not be dedicated. Resources may be added to or may disappear from the platform at any time, or the platform may have a stable composition. The processing units may have the same characteristics (e.g., computational power, amount of memory, multi-port or only single-port communications support, etc.) or not. The communication links may have the same characteristics (e.g., bandwidths, latency, routing policy, etc.) or not. The aim of the scheduling policy can be to minimize the overall execution time (makespan minimization), the throughput of processed tasks, etc. Finally, the set of all tasks to be scheduled may be known from the beginning, or new tasks may arrive all along the execution of the system (on-line scheduling).

In the GRAAL project, we investigate scheduling problems that are of practical interest in the context of large-scale distributed platforms. We assess the impact of the heterogeneity and volatility of the resources onto the scheduling strategies.

3.2. Scheduling for Parallel Sparse Direct Solvers

Participants: Guillaume Joslin, Maurice Brémond, Indranil Chowdhury, Jean-Yves L'Excellent, Bora Uçar.

The solution of sparse systems of linear equations (symmetric or unsymmetric, most often with an irregular structure) is at the heart of many scientific applications arising in various domains such as geophysics, chemistry, electromagnetism, structural optimization, and computational fluid dynamics. The importance and diversity of the fields of applications are our main motivation to pursue research on sparse linear solvers. Furthermore, in order to solve hard problems that result from ever-increasing demand for accuracy in simulations, special attention must be paid to both memory usage and execution time on the most powerful parallel platforms (whose usage is necessary because of the volume of data and amount of computation required). This is done by specific algorithmic choices and scheduling techniques. From a complementary point of view, it is also necessary to be aware of the functionality requirements from the applications and from the users, so that robust solutions can be proposed for a large range of problems.

Because of their efficiency and robustness, direct methods (based on Gaussian elimination) are methods of choice to solve these types of problems. In this context, we are particularly interested in the multifrontal method [91], [92] for symmetric positive definite, general symmetric or unsymmetric problems, with numerical pivoting in order to ensure numerical accuracy. The existence of numerical pivoting induces dynamic updates in the data structures where the updates are not predictable with a static or symbolic analysis approach.

The multifrontal method is based on an elimination tree [95] which results (i) from the graph structure corresponding to the nonzero pattern of the problem to be solved, and (ii) from the order in which variables are eliminated. This tree provides the dependency graph of the computations and is exploited to define tasks that may be executed in parallel. In the multifrontal method, each node of the tree corresponds to a task (itself can be potentially parallel) that consists in the partial factorization of a dense matrix. This approach allows for a good locality and hence efficient use of cache memories.

We are especially interested in approaches that are intrinsically dynamic and asynchronous [1], [86], as these approaches can encapsulate numerical pivoting and can be adopted to various computer architectures. In addition to their numerical robustness, the algorithms are based on a dynamic and distributed management of the computational tasks, not so far from today's peer-to-peer approaches: each process is responsible for providing work to some other processes and at the same time it acts as a worker for others. These algorithms are very interesting from the point of view of parallelism and in particular for the study of mapping and scheduling strategies for the following reasons:

- the associated task graphs are very irregular and can vary dynamically,
- they are currently used inside industrial applications, and
- the evolution of high performance platforms, to the more heterogeneous and less predictable ones, requires that applications adapt themselves, using a mixture of dynamic and static approaches, as our approach allows.

Our research in this field is strongly linked to the software package MUMPS (see Section 5.2) which is our main platform to experiment and validate new ideas and pursue new research directions. We are facing new challenges for very large problems (tens to hundreds of millions of equations) that occur nowadays in various application fields: in that case, either parallel out-of-core approaches are required, or direct solvers should be combined with iterative schemes, leading to hybrid direct-iterative methods.

3.3. Algorithms and Software Architectures for Service Oriented Platforms

Participants: Nicolas Bard, Julien Bigot, Laurent Bobelin, Yves Caniou, Eddy Caron, Ghislain Charrier, Florent Chuffart, Simon Delamare, Benjamin Depardon, Frédéric Desprez, Gilles Fedak, Sylvain Gault, Haiwu He, Benjamin Isnard, Cristian Klein, Mohamed Labidi, Gaël Le Mahec, George Markomanolis, Adrian Muresan, Christian Pérez, Vincent Pichon, Luis Rodero-Merino, José Francisco Saray Villamizar, Daouda Traore.

The fast evolution of hardware capabilities in terms of wide area communication as well as of machine virtualization leads to the requirement of another step in the abstraction of resources with respect to applications. Those large scale platforms based on the aggregation of large clusters (Grids), huge datacenters (Clouds) or collections of volunteer PCs (Desktop computing platforms) are now available for researchers of different fields of science as well as private companies. This variety of platforms and the way they are accessed have also an important impact on how applications are designed (i.e., the programming model used) as well as how applications are executed (i.e., the runtime/middleware system used). The access to these platforms is driven through the use of different services providing mandatory features such as security, resource discovery, virtualization, load-balancing, etc. Software as a Service (SaaS) has thus to play an important role in the future development of large scale applications. The overall idea is to consider the whole system, ranging from the resources to the application, as a set of services. Hence, a user application is an ordered set of instructions requiring and making uses of some services like for example an execution service. Such a service is also an application—but at the middleware level—that is proposing some services (here used by the user application) and potentially using other services like for example a scheduling service. This model based on services provided and/or offered is generalized within software component models which deal with composition issues as well as with deployment issues.

Our goal is to contribute to the design of programming models supporting a wide range of architectures and to their implementation by mastering the various algorithmic issues involved and by studying the impact on application-level algorithms. Ideally, an application should be written once; the complexity is to determine the adequate level of abstraction to provide a simple programming model to the developer while enabling efficient execution on a wide range of architectures. To achieve such a goal, the team plans to contribute at different level including programming models, distributed algorithms, deployment of services, services discovery, service composition and orchestration, large scale data management, etc.

4. Application Domains

4.1. Applications of Sparse Direct Solvers

In the context of our activity on sparse direct (multifrontal) solvers in distributed environments, we develop, distribute, maintain and support competitive software. Our methods have a wide range of applications, and they are at the heart of many numerical methods in simulation: whether a model uses finite elements or finite differences, or requires the optimization of a complex linear or nonlinear function, one almost always ends up solving a linear system of equations involving sparse matrices. There are therefore a number of application fields, among which we list some cited by the users of our sparse direct solver MUMPS (see Section 5.2): structural mechanical engineering (e.g., stress analysis, structural optimization, car bodies, ships, crankshaft segment, offshore platforms, computer assisted design, computer assisted engineering, rigidity of sphere packings); heat transfer analysis; thermomechanics in casting simulation; fracture mechanics; biomechanics; medical image processing; tomography; plasma physics (e.g., Maxwell's equations), critical physical phenomena, geophysics (e.g., seismic wave propagation, earthquake related problems); ad-hoc networking modeling (e.g., Markovian processes); modeling of the magnetic field inside machines; econometric models; soil-structure interaction problems; oil reservoir simulation; computational fluid dynamics (e.g., Navier-Stokes, ocean/atmospheric modeling with mixed finite elements methods, fluvial hydrodynamics, viscoelastic flows); electromagnetics; magneto-hydro-dynamics; modeling the structure of the optic nerve head and of cancellous bone; modeling of the heart valve; modeling and simulation of crystal growth processes; chemistry (e.g., chemical process modeling); vibro-acoustics; aero-acoustics; aero-elasticity; optical fiber modal analysis; blast furnace modeling; glaciology (e.g., modeling of ice flow); optimization; optimal control theory; astrophysics (e.g., supernova, thermonuclear reaction networks, neutron diffusion equation, quantum chaos, quantum transport); research on domain decomposition (e.g., MUMPS is used on subdomains in an iterative solver framework); and circuit simulations.

4.2. Molecular Dynamics

LAMMPS is a classical molecular dynamics (MD) code created for simulating molecular and atomic systems such as proteins in solution, liquid-crystals, polymers, or zeolites. It was designed for distributed-memory parallel computers and runs on any parallel platform that supports the MPI message-passing library or on single-processor workstations. LAMMPS is mainly written in F90.

LAMMPS was originally developed as part of a 5-way DoE-sponsored CRADA collaboration between 3 industrial partners (Cray Research, Bristol-Myers Squibb, and Dupont) and 2 DoE laboratories (Sandia and Livermore). The code is freely available under the terms of a simple license agreement that allows you to use it for your own purposes, but not to distribute it further.

The integration of LAMMPS into our Problem Solving Environment DIET is in progress. Discussions are still taking place in order to make the LAMMPS service available through a web portal, on at least one cluster managed by the Sun Grid Engine batch scheduler.

4.3. Biochemistry

Current progress in different areas of chemistry such as organic chemistry, physical chemistry or biochemistry allows the construction of complex molecular assemblies with predetermined properties. In all these fields, theoretical chemistry plays a major role by helping to build various models which can greatly differ in terms of theoretical and computational complexity, and which allow the understanding and the prediction of chemical properties.

Among the various theoretical approaches available, quantum chemistry is at a central position as all modern chemistry relies on it. This scientific domain is quite complex and involves heavy computations. In order to fully apprehend a model, it is necessary to explore the whole potential energy surface described by the independent variation of all its degrees of freedom. This involves the computation of many points on this surface.

Our project is to couple DIET with a relational database in order to explore the potential energy surface of molecular systems using quantum chemistry: all molecular configurations to compute are stored in a database, the latter is queried, and all configurations that have not been computed yet are passed through DIET to computer servers which run quantum calculations, all results are then sent back to the database through DIET. At the end, the database will store a whole potential energy surface which can then be analyzed using proper quantum chemical analysis tools.

4.4. Bioinformatics

Genomics acquiring programs, such as full genomes sequencing projects, are producing larger and larger amounts of data. The analysis of these raw biological data require very large computing resources. In some cases, due to the lack of sufficient computing and storage resources, skilled staff or technical abilities, laboratories cannot afford such huge analyses. Grid computing may be a viable solution to the needs of the genomics research field: it can provide scientists with a transparent access to large computational and data management resources. In this application domain, we are currently addressing two different problems outlined below.

In the first problem, we tackle the problem of clustering the sequences contained in international databanks into domain protein families. Our aim is to ensure, through the use of grids, the capacity of timely and automatically building of databases (such as ProDom) when such databases are built from exponentially-fast growing protein databases.

In the second problem, we consider protein functional sites. Functional sites and signatures of proteins are very useful for analyzing raw biological data or for correlating different kinds of existing biological data. These methods are applied, for example, to the identification and characterization of the potential functions of new sequenced proteins. The sites and signatures of proteins can be expressed by using the syntax defined by the PROSITE databank, and written as a "protein regular expression". Searching one such site in a sequence can be done with the criterion of the identity between the searched and the found patterns. Most of the time, this kind of analysis is quite fast. However, in order to identify non perfectly matching but biologically relevant sites, the user can accept a certain level of error between the searched and the matching patterns. Such an analysis can be very resource consuming.

4.5. Cosmological Simulations

Ramses [1] is a typical computational intensive application used by astrophysicists to study the formation of galaxies. *Ramses* is used, among other things, to simulate the evolution of a collisionless, self-gravitating fluid called "dark matter" through cosmic time. Individual trajectories of macro-particles are integrated using a state-of-the-art "N body solver", coupled to a finite volume Euler solver, based on the Adaptive Mesh Refinement technique. The computational space is decomposed among the available processors using a *mesh partitioning* strategy based on the Peano-Hilbert cell ordering.

Cosmological simulations are usually divided into two main categories. Large scale periodic boxes requiring massively parallel computers are performed on a very long elapsed time (usually several months). The second category stands for much faster small scale "zoom simulations". One of the particularity of the HORIZON project is that it allows the re-simulation of some areas of interest for astronomers.

We designed a Grid version of *Ramses* through the DIET middleware. From Grid'5000 experiments we proved that DIET is capable of handling long cosmological parallel simulations: mapping them on parallel resources of a Grid, executing and processing communication transfers. The overhead induced by the use of DIET is negligible compared to the execution time of the services. Thus DIET permits to explore new research axes in cosmological simulations (on various low resolutions initial conditions), with transparent access to the services and the data.

[1] http://www.projet-horizon.fr/

4.6. Ocean-Atmosphere Simulations

Climatologists have recourse to numerical simulation and particularly coupled models in several occasions: for example, to estimate natural variability (thousand of simulated years), for seasonal forecasting (only a few simulated months) or to study global warming characteristics (some simulated decades).

To take advantage of the Grid'5000 platform, we choose to launch parallel simulations (ensemble) on several nodes, approximatively 10 or more, according to the load of the platform. Scenario simulations that simulate from present climate to the next century require huge computing power. Indeed, each simulation will differ from each other in physical parameterization of atmospheric model. Comparing them, we expect to better estimate global warming prediction sensibility in order to model parameterization.

Practically, a 150 year long scenario combines 1800 simulations of one month each, launched one after the other. This partitioning eases workflow and implements checkpointing because the final state of the simulation of one month is used as the initial state of the next month.

Our goal regarding the climate forecasting application is to thoroughly analyze it in order to model its needs in terms of execution model, data access pattern, and computing needs. Once a proper model of the application has been derived, appropriate scheduling heuristics can be proposed, tested, and compared. We plan to extend this work to provide generic scheduling schemes for applications with similar dependence graphs.

4.7. Décrypthon

The Décrypthon project is built over a collaboration between CNRS, AFM (*Association Française contre les Myopathies*), and IBM. Its goal is to make computational and storage resources available to bioinformatic research teams in France. These resources, connected as a Grid through the Renater network, are installed in six universities and schools in France (Bordeaux, Jussieu, Lille, Lyon, Orsay, and Rouen). The Décrypthon project offers means necessary to use the Grid through financing of research teams and postdoc, and assistance on computer science problems (such as modeling, application development, and data management). The GRAAL research team is involved in this project as an expert for application gridification. The Grid middleware used at the beginning of the project was GridMP from United Devices. In 2007, DIET was chosen to be the Grid middleware of the Décrypthon Grid. It ensures the load-balancing of jobs over the six computation centers through the Renater network. This transfer of our middleware, first built for large scale experimentations of scheduling heuristics, in a production Grid is a real victory for our research team.

4.8. Micro-Factories

Micro-factories are automated units designed to produce pieces composed of micro-metric elements. Today's micro-factories are composed of elementary modules or robots able to carry out basic operations. To perform more complex operations, few elementary modules may be grouped in a cell. The realization of one of these cells is still a scientific challenge but several research projects have already got significant results in this domain. These results show very promising functionalities like the ability to configure or reconfigure a cell, by changing a robot tool for instance. However, the set of operations carried out by a cell is still limited. The next generation of micro-factories will put several cells together and make them cooperate to produce complex assembled pieces, as we do for macroscopic productions. In this context, the cell control will evolve to become more cooperative and distributed.

Micro-factories may be modeled in a way that allows reusing the results obtained in scheduling on heterogeneous platforms as Grids, in particular the results on steady-state scheduling. We develop scheduling strategies and algorithms adapted to this context and we optimize the deployment of cells based on the micro-product and the production specification. We are currently working on the evaluation and the adaptation of several scheduling algorithms in this context, taking small-to-medium batch of jobs into account.

At the micro-metric scale, the manipulation of the elements cannot be considered the same way as at macro-metric scale because the equilibrium of forces is modified. For instance, the electrostatic force becomes predominant on the gravity. This lead to uncontrolled behaviors and frequently generates faults. We are working on taking these faults into account into scheduling models and evaluating their performance depending on the fault characteristics.

5. Software

5.1. DIET

Participants: Nicolas Bard, Yves Caniou, Eddy Caron [correspondent], Ghislain Charrier, Frédéric Desprez, Adrian Muresan, Vincent Pichon.

Huge problems can now be processed over the Internet thanks to Grid middleware systems. The use of on-the-shelf applications is needed by scientists of other disciplines. Moreover, the computational power and memory needs of such applications may of course not be met by every workstation. Thus, the RPC paradigm seems to be a good candidate to build Problem Solving Environments on the Grid. The aim of the DIET project (http://graal.ens-lyon.fr/DIET) is to develop a set of tools to build computational servers accessible through a GridRPC API.

Moreover, the aim of a middleware system such as DIET is to provide a transparent access to a pool of computational servers. DIET focuses on offering such a service at a very large scale. A client which has a problem to solve should be able to obtain a reference to the server that is best suited for it. DIET is designed to take into account the data location when scheduling jobs. Data are kept as long as possible on (or near to) the computational servers in order to minimize transfer times. This kind of optimization is mandatory when performing job scheduling on a wide-area network. DIET is built upon *Server Daemons*. The scheduler is scattered across a hierarchy of *Local Agents* and *Master Agents*.

Applications targeted for the DIET platform are now able to exert a degree of control over the scheduling subsystem via *plug-in schedulers* [88]. As the applications that are to be deployed on the Grid vary greatly in terms of performance demands, the DIET plug-in scheduler facility permits the application designer to express application needs and features in order that they be taken into account when application tasks are scheduled. These features are invoked at runtime after a user has submitted a service request to the MA, which broadcasts the request to its agent hierarchy.

DIET has been validated on several applications. Some of them have been described in Sections 4.2 through 4.7.

5.1.1. Workflow support

Workflow-based applications are scientific, data intensive applications that consist of a set of tasks that need to be executed in a certain partial order. These applications are an important class of Grid applications and are used in various scientific domains like astronomy or bioinformatics. We have developed a workflow engine in DIET to manage such applications and propose to the end-user and the developer a simple way either to use provided scheduling algorithms or to develop their own scheduling algorithm.

In our implementation, workflows are described using the XML language. Since no standard exists for scientific workflows, we have proposed our formalism. The DIET agent hierarchy has been extended with a new special agent, the *MA_DAG*. To be flexible we can execute workflows even if this special agent is not present in the platform. The use of the *MA_DAG* centralizes the scheduling decisions and thus can provide a better scheduling when the platform is shared by multiple clients. On the other hand, if the client bypasses the *MA_DAG*, a new scheduling algorithm can be used without affecting the DIET platform. The current implementation of DIET provides several schedulers (Round Robin, HEFT, random, Fairness on finish Time, etc.).

The DIET workflow runtime also includes a rescheduling mechanism. Most workflow scheduling algorithms are based on performance predictions that are not always accurate (erroneous prediction tool or resource load wrongly estimated). The rescheduling mechanism can trigger the application rescheduling when some conditions specified by the client are filled.

We also continued our work on schedulers for DIET workflow engine concerning multi-workflows based applications, and graphical tools for workflows within the DIET DashBoard project. Within the Gwendia project, we worked on the implementation of the language defined in the project and around the Cardiac application. Experiments were done over the Grid'5000 platform.

5.1.2. Diet Data Management

DAGDA, designed during the PhD of Gaël Le Mahec, is a new data manager for the DIET middleware which allows data explicit or implicit replications and advanced data management on the grid. It was designed to be backward compatible with previously developed applications for DIET which benefit transparently of the data replications. It allows explicit or implicit data replications, file sharing between the nodes which can access to the same disk partition, the choice of a data replacement algorithm, and a high level configuration about the memory and disk space DIET should use for the data storage and transfers. To transfer a data, DAGDA uses the pull model instead of the push model used by DTM. The data are not sent into the profile from the source to the destination, but they are downloaded by the destination from the source. DAGDA also chooses the best source for a given data. DAGDA has also been used for the validation of our join replication and scheduling algorithms over DIET.

5.1.3. GridRPC Data Management API

The GridRPC paradigm is now an OGF standard. The GridRPC community has interests in the Data Management within the GridRPC paradigm. Because of previous works performed in the DIET middleware concerning Data Management, Eddy Caron is now co-chair of the GridRPC working group in order to lead the project to propose a powerful Grid Data Management API which will extend the GridRPC paradigm.

Data Management is a challenging issue inside the OGF GridRPC standard, for feasability and performance reasons. Indeed some temporarily data do not need to be transfered once computed and can reside on servers for example. We can also imagine that data can be directly transferred from one server to another one, without being transferred to the client in accordance to the GridRPC paradigm behavior.

In consequence, we work on a Data Management API which has been presented to almost all OGF sessions since OGF'21. Since december 2009, the proposal is available for public comment and may be reached at: http://www.ogf.org/gf/docs/?public_comment under the title "Proposal for a Data Management API within the GridRPC. Y. Caniou and others, via GRIDRPC-WG". Today public comment is closed and all remarks are included in the current document. This document is finished and will be standardized in 2011.

5.1.4. Middleware Interoperability

For the requirements of the GridTLSE project, DIET has been extended with a specialized version of a server daemon. It is able to provide access to the AEGIS middleware services developped in the JAEA. A demo has been presented in the JAEA booth at SuperComputing'10.

5.1.5. Diet as a Cloud System

Cloud computing is currently drawing more and more attention. This is due to multiple reasons, the most important of which are the on-demand way of provisioning resources and the pay-as-you-go pricing. In order to study and take advantage of these features, we extended the DIET middleware with Cloud support. DIET Cloud is a DIET module able of harnessing the extensibility of Cloud platforms in a seamless manner. We have targeted the Eucalyptus open-source Cloud because it implements the Amazon EC2 Cloud interface. Recently we also confirmed DIET Cloud's compatibility to Amazon EC2 by building a proof-of-concept demo which was shown at SuperComputing'10.

5.1.6. Diet Green

We have designed a new metric called (GreenPerf) to allow to DIET to provide a scheduler that takes into account the energy information. We designed a heuristic to find the best server with the good rate between performance and electric consumption. In collaboration with Laurent Lefevre from the RESO research team, we designed the architecture to deal with energy sensors. More developments and experiments are required to validate the integration into the current release.

5.1.7. MapReduce over Diet

The MapReduce programming model (re-)introduced by Google is a promising model to deploy data processing application services over large scale platforms such as Grids and Clouds. We developed a version of MapReduce over DIET. In particular, we automatized the creation of MapReduce-type workflows. Some large-scale experiments over the Grid'5000 platform were conducted to validate the concepts and algorithms developed.

For each input key/value pair, the DIET workflow engine generates one map task. Each map calculates intermediate key/value pairs and returns a container with all intermediate pairs. Thanks to the DIET workflow engine, the results are merged and all containers are sent to the sort service. This service sorts the pairs by combining one key and all its values in a container. This container is itself added to a container that is returned by the service. The DIET workflow engine then explodes the main container, and it creates a reducing task for each element. Reducing tasks calculate and return final key/value pairs. All final pairs are then merged and returned to the client.

We implemented a prototype with the sorting service and a prototype with tree reduction. These prototypes allowed us to validate the feasibility of two solutions and the constraints imposed by the DIET midlleware.

5.1.8. DIET and EDF R&D

We worked on the DIET integration into the EDF infrastructure in the context of the INRIA Grenoble-Rhône-Alpes GRAAL, EDF R&D SINETICS OSIS partnership. The first work was to provide a set of new functionalities for users to submit a large amount of tasks on different remote LRMS (*Local Resources Manager Systems*), to manage and tune these tasks and to finally retrieve the results. The solution is based on DIET modules that can be called in C/C++, or called directly from the command line.

5.1.9. Latest Releases

- May 26th, DIET 2.4 release.
- December 1st, DIET 2.5 release.

Moreover, since May, special developments around the File and Batch management of a HPC infrastructure for EDF R&D are available in open source.

5.2. MUMPS

Participants: Maurice Brémond, Indranil Chowdhury, Guillaume Joslin, Jean-Yves L'Excellent [correspondent], Bora Uçar.

MUMPS (for *MUltifrontal Massively Parallel Solver*, see http://graal.ens-lyon.fr/MUMPS) is a software package for the solution of large sparse systems of linear equations. The development of MUMPS was initiated by the European project PARASOL (Esprit 4, LTR project 20160, 1996-1999), whose results and developments were public domain. Since then, mainly in collaboration with ENSEEIHT-IRIT (Toulouse, France), lots of developments have been done, to enhance the software with more functionalities and integrate recent research work. Recent developments also involve the former INRIA project ScAlAppliX since the recruitment of Abdou Guermouche as an assistant professor at *LaBRI*, while CERFACS contributes to some research work.

MUMPS implements a direct method, the multifrontal method, and is a parallel code for distributed memory computers; it is unique by the performance obtained and the number of functionalities available, among which we can cite:

- various types of systems: symmetric positive definite, general symmetric, or unsymmetric,
- several matrix input formats: assembled or expressed as a sum of elemental matrices, centralized on one processor or pre-distributed on the processors,
- detection of null pivots and null space estimate,
- parallel analysis, parallel scaling algorithms,
- out-of-core execution to solve larger problems,
- partial factorization and Schur complement matrix,
- dense or sparse right-hand sides, centralized or distributed solution,
- real or complex arithmetic, single or double precision,
- partial threshold pivoting,
- fully asynchronous approach with overlap of computation and communication,
- distributed dynamic scheduling of the computational tasks to allow for a good load balance in case of numerical pivoting at runtime.

MUMPS is currently used by thousands of academic and industrial organizations, for a wide range of application fields (see Section 4.1). MUMPS users include:

- students and academic users from all over the world;
- various developers of finite element or optimization software;
- companies such as Boeing, EADS, EDF, Free Field Technologies, and Samtech.

The latest release is MUMPS 4.9.2, available since November 2009 (see http://graal.ens-lyon.fr/MUMPS/). Most of the work in 2010 has concerned development work related to projects and contracts, and software and validation issues, in the context of a MUMPS Action of Technological Development called "ADT-MUMPS".

5.3. HLCMi

Participants: Julien Bigot, Cristian Klein, Christian Pérez [correspondent], Vincent Pichon.

HLCMi is an implementation of the HLCM component model defined during the PhD of Julien Bigot. HLCM is a generic extensible component model with respect to component implementations and interaction concerns. Moreover, HLCM is abstract; it is its specialization—such as HLCM/CCM—that define the primitive elements of the model, such as the primitive components and the primitive interactions.

HLCMi is making use of Model-driven Engineering (MDE) methodology to generate a concrete assembly from an high level description. It is based on the Eclipse Modeling Framework (EMF). HLCMi contains 700 Emfatic lines to describe its models and 7000 JAVA lines for utility and model transformation purposes. HLCMi is a general framework that supports several HLCM specialization: HLCM/CCM, HLCM/JAVA and HLCM/C++.

5.4. BitDew

Participants: Gilles Fedak [correspondent], Haiwu He, José Francisco Saray Villamizar, Mircea Moca, Lu Lu.

BITDEW is an open source middleware implementing a set of distributed services for large scale data management on Desktop Grids and Clouds. BITDEW relies on five abstractions to manage the data : i) replication indicates how many occurrences of a data should be available at the same time on the network, ii) fault-tolerance controls the policy in presence of hardware failures, iii) lifetime is an attribute absolute or relative to the existence of other data, which decides the life cycle of a data in the system, iv) affinity drives movement of data according to dependency rules, v) protocol gives the runtime environment hints about the protocol to distribute the data (http, ftp or bittorrent). Programmers define for every data these simple criteria, and let the BITDEW runtime environment manage operations of data creation, deletion, movement, replication, and fault-tolerance operation.

The current status of the software is the following : BITDEW is open source under the GPLv3 or Cecill licence at the user's choice, 10 releases were produced in the last two years, and it has been downloaded approximativelly 6000 times on the INRIA forge. Known users are Université Paris-XI, Université Paris-XIII, University of Florida, Cardiff University and University of Sfax. Recently, we have implemented a first prototype of the MapReduce programming model for Desktop Grids on top of BitDew.

In term of support, the development of BitDew is partly funded by the INRIA ADT BitDew and by the ANR MapReduce projects.

5.5. XtremWeb

Participants: Gilles Fedak [correspondent], Haiwu He, Bing Tang, Simon Delamare.

XTREMWEB is an open source software for Desktop Grid computing, jointly developed by INRIA and IN2P3.

XTREMWEB allows to build lightweight Desktop Grid by gathering the unused resources of Desktop Computers (CPU, storage, network). Its primary features permit multi-users, multi-applications and cross-domains deployments. XTREMWEB turns a set of volatile resources spread over LAN or Internet into a runtime environment executing high througput applications.

XTREMWEB is a highly programmable and customizable middleware which supports a wide range of applications (bag-of tasks, master/worker), computing requirements (data/CPU/network-intensive) and computing infrastructures (clusters, Desktop PCs, multi-Lan) in a manageable, scalable and secure fasion. Known users include LIFL, LIP, LIG, LRI (CS), LAL (physics Orsay), IBBMC (biology), Université Paris-XIII, Université de Guadeloupe, IFP (petroleum), EADS, CEA, University of Wisconsin Madison, University of Tsukuba (Japan), AIST (Australia), UCSD (USA), Université de Tunis, AlmerGrid (NL), Fundecyt (Spain), Hobai (China), HUST (China).

There are two branches of XTREMWEB: XTREMWEB-HEP is a production version developed by IN2P3. It features many security improvements such as X509 support which allows its usage within the EGEE context. XTREMWEB-CH is a research version developed by HES-SO, Geneva, which aims at building an effective Peer-To-Peer system for CPU time consuming applications.

XTREMWEB has been supported by national grants (ACI CGP2P) and by major European grants around Grid and Desktop Grid such as FP6 CoreGrid: European Network of Excellence, FP6 Grid4all, and more recently FP7 EDGeS : Enabling Desktop Grid for E-Science and FP7 EDGI: European Desktop Grid Initiative.

On going developments include : providing Quality-of-Service for Desktop Grids (SpeQuloS), data-intensive processing on Desktop Grid as well as the portage of XtremWeb to the Google App Engine Cloud platform.

6. New Results

6.1. Scheduling Strategies and Algorithm Design for Heterogeneous Platforms

Participants: Anne Benoît, Marin Bougeret, Hinde Bouziane, Alexandru Dobrila, Fanny Dufossé, Matthieu Gallet, Mathias Jacquelin, Loris Marchal, Jean-Marc Nicod, Laurent Philippe, Paul Renaud-Goud, Clément Rezvoy, Yves Robert, Mark Stillwell, Bora Uçar, Frédéric Vivien.

6.1.1. Mapping simple workflow graphs

Mapping workflow applications onto parallel platforms is a challenging problem that becomes even more difficult when platforms are heterogeneous—nowadays a standard. A high-level approach to parallel programming not only eases the application developer's task, but it also provides additional information which can help realize an efficient mapping of the application. We focused on simple application graphs such as linear chains and fork patterns. Workflow applications are executed in a pipeline manner: a large set of data needs to be processed by all the different tasks of the application graph, thus inducing parallelism between the processing of different data sets. For such applications, several antagonist criteria should be optimized, such as throughput, latency, failure probability and energy minimization.

We have considered the mapping of workflow applications onto different types of platforms: *fully homogeneous* platforms with identical processors and interconnection links; *communication homogeneous* platforms, with identical links but processors of different speeds; and finally, *fully heterogeneous* platforms.

This year, we have pursued the work involving the energy minimization criteria, and we studied the impact of sharing resources for concurrent streaming applications. For interval mappings, a processor is assigned a set of consecutive stages of the same application, so there is no resource sharing across applications. On the contrary, the assignment is fully arbitrary for general mappings, hence a processor can be reused for several applications. On the theoretical side, we establish complexity results for this tri-criteria mapping problem (energy, period, latency), classifying polynomial versus NP-complete instances. Furthermore, we derive an integer linear program that provides the optimal solution in the most general case. On the experimental side, we design polynomial-time heuristics, and assess their absolute performance thanks to the linear program. One main goal is to assess the impact of processor sharing on the quality of the solution.

6.1.2. Throughput of probabilistic and replicated streaming applications

We have pursued the investigation of timed Petri nets to model the mapping of workflows with stage replication, that we had started in 2009. In particular, we have provided bounds for the throughput when stage parameters are arbitrary I.I.D. (Independent and Identically-Distributed) and N.B.U.E. (New Better than Used in Expectation) variables: the throughput is bounded from below by the exponential case and bounded from above by the deterministic case. This work was conducted in collaboration with Bruno Gaujal (LIG Grenoble).

6.1.3. Multi-criteria algorithms and heuristics

We have investigated several multi-criteria algorithms and heuristics for the problem of mapping pipelined applications, consisting of a linear chain of stages executed in a pipelined way, onto heterogeneous platforms. The objective was to optimize the reliability under a performance constraint, i.e., while guaranteeing a threshold throughput. In order to increase reliability, we replicate the execution of stages on multiple processors. On the theoretical side, we prove that this bi-criteria optimization problem is NP-hard. We propose some heuristics both for interval and for general mappings, and present extensive experiments evaluating their performance.

The first paper published on this work, "A. Benoit, H. L. Bouziane, Y. Robert. Optimizing the reliability of pipelined applications under throughput constraints. In ISPDC'2010, Istanbul, Turkey, July 2010" received the best paper award.

6.1.4. The impact of cache misses on the performance of matrix product algorithms on multicore platforms

The multicore revolution is underway, bringing new chips introducing more complex memory architectures. Classical algorithms must be revisited in order to take the hierarchical memory layout into account. The goal is this study is to design cache-aware algorithms that minimize the number of cache misses paid during the execution of the matrix product kernel on a multicore processor. We have analytically studied how to achieve the best possible tradeoff between shared and distributed caches. We have also implemented and evaluated several algorithms on two multicore platforms, one equipped with one Xeon quadcore, and the second one

enriched with a GPU. It turns out that the impact of cache misses is very different across both platforms, and we have identified what are the main design parameters that lead to peak performance for each target hardware configuration.

6.1.5. Tree traversals with minimum memory usage

In this study, we focus on the complexity of traversing tree-shaped workflows whose tasks require large I/O files. Such workflows typically arise in the multifrontal method of sparse matrix factorization. We target a classical two-level memory system, where the main memory is faster but smaller than the secondary memory. A task in the workflow can be processed if all its predecessors have been processed, and if its input and output files fit in the currently available main memory. The amount of available memory at a given time depends upon the ordering in which the tasks are executed. We focus on the problem of finding the minimum amount of main memory, over all postorder schemes, or over all possible traversals, that is needed for an in-core execution. We have established several complexity results that answer these questions. We have proposed a new, polynomial time, exact algorithm which runs faster than a reference algorithm. We have also addressed the setting where the required memory renders a pure in-core solution unfeasible. In this setting, we ask the following question: what is the minimum amount of I/O that must be performed between the main memory and the secondary memory? We have shown that this latter problem is NP-hard, and proposed efficient heuristics. All algorithms and heuristics were thoroughly evaluated on assembly trees arising in the context of sparse matrix factorizations.

6.1.6. Comparing archival policies for BlueWaters

In this work, we focus on the archive system which will be used in the BlueWaters supercomputer. We have introduced two archival policies tailored for the large tape storage system that will be available on BlueWaters. We have also shown how to adapt the well known RAIT strategy (the counterpart of RAID policy for tapes). We have provided an analytical model of the tape storage platform of BlueWaters, and we used it to asses and analyze the performance of the three policies through simulations. Storage requests were generated using random workloads whose characteristics model various realistic scenarios. The throughput of the system, as well as the average (weighted) response time for each user, are the main objectives.

6.1.7. Resource allocation using virtual clusters

We proposed a novel job scheduling approach for sharing a homogeneous cluster computing platform among competing jobs. Its key feature is the use of virtual machine technology for sharing resources in a precise and controlled manner. We followed up on our work on this subject by addressing the problem of resource utilization. We proposed a new measure for this utilization and we demonstrated how, following our approach, one can improve over batch scheduling by orders of magnitude in term of job stretch, while leading to comparable or better resource utilization.

6.1.8. Checkpointing policies for post-petascale supercomputers

An alternative to classical fault-tolerant approaches for large-scale clusters is failure avoidance, by which the occurrence of a fault is predicted and a preventive measure is taken. We developed analytical performance models for two types of such a measure: preventive checkpointing and preventive migration. We also developed an analytical model of the performance of a standard periodic checkpoint fault-tolerant approach. We instantiated these models for platform scenarios that are representative of the current and future technology trends. We found that preventive migration is the better approach in the short term, but that both approaches have comparable merit in the longer term. We also found that standard non-prediction-based fault tolerance achieves poor scaling when compared to prediction-based failure avoidance, thereby demonstrating the importance of failure prediction capabilities. Our results also showed that achieving good utilization of truly large-scale machines (e.g., 2^{20} nodes) for parallel workloads will require more than the failure avoidance techniques evaluated in this work.

In the previous work, we have assumed that checkpoints were occurring periodically. Indeed, it is usually claimed that such a policy is optimal. However, most of the existing proofs rely on approximations. One such assumption is that the probability that a fault occurs during the execution of an application is very small, an assumption that is no longer valid in the context of exascale platforms. We have begun studying this problem in a fully general context. We have established that, when failures follow a Poisson law, the periodic checkpointing policy is optimal. We have also showed an unexpected result: in some cases, when the platform is sufficiently large, the checkpointing costs are sufficiently expensive, or the failures are frequent enough, one should limit the application parallelism and duplicate tasks, rather than fully parallelize the application on the whole platform. In other words, the expectation of the job duration is smaller with fewer processors! To establish this result we derived and analyzed several scheduling heuristics.

6.1.9. Scheduling parallel iterative applications on volatile resources

In this work we study the efficient execution of iterative applications onto volatile ressources. We studied a master-worker scheduling scheme that trades-off between the speed and the (expected) reliability and availability of enrolled workers. A key feature of this approach is that it uses a realistic communication model that bounds the capacity of the master to serve the workers, which requires the design of sophisticated resource selection strategies. The contribution of this work is twofold. On the theoretical side, we assess the complexity of the problem in its off-line version, i.e., when processor availability behaviors are known in advance. Even with this knowledge, the problem is NP-hard. On the pragmatic side, we proposed several on-line heuristics that were evaluated in simulation while a Markovian model of processor availabilities.

6.1.10. Parallelizing the construction of the ProDom database

ProDom is a protein domain family database automatically built from a comprehensive analysis of all known protein sequences. ProDom development is headed by Daniel Kahn (INRIA project-team BAMBOO, formerly HELIX). With the protein sequence databases increasing in size at an exponential pace, the parallelization of MkDom2, the algorithm used to build ProDom, has become mandatory (the original sequential version of MkDom2 took 15 months to build the 2006 version of ProDom).

The parallelization of MkDom2 is not a trivial task. The sequential MkDom2 algorithm is an iterative process, and parallelizing it involves forecasting which of these iterations can be run in parallel and detecting and handling dependency breaks when they arise. We have moved forward to be able to efficiently handle larger databases. Such databases are prone to exhibit far larger variations in the processing time of query-sequences than was previously imagined. The collaboration with BAMBOO on ProDom continues today both on the computational aspects of the constructing of ProDom on distributed platforms, as well as on the biological aspects of evaluating the quality of the domains families defined by MkDom2, as well as the qualitative enhancement of ProDom.

This past year was devoted to the full scale validation of the the new parallel MPI_MkDom2 algorithm and code. We proposed a new methodology to compare two clusterings of sub-sequences in domains. We used this methodology to assess that the parallelization using MPI_MkDom2 do not significantly impact the quality of the clustering produced, when compared to the one produced by MkDom2. We successfully processed all the sequences included in the April 2010 version of the UniProt database, namely 6 118 869 sequences and 2 194 382 846 amino-acids. The whole computation would have taken 12 years and 97 days in sequential and was completed in parallel for a wall-clock time of 19 days and 12 hours. After a post-processing phase, this will lead to a new release of ProDom in the upcoming months after a four year hiatus.

6.2. Algorithms and Software Architectures for Service Oriented Platforms

Participants: Nicolas Bard, Julien Bigot, Laurent Bobelin, Yves Caniou, Eddy Caron, Ghislain Charrier, Florent Chuffart, Benjamin Depardon, Frédéric Desprez, Gilles Fedak, Haiwu He, Benjamin Isnard, Cristian Klein, Gaël Le Mahec, Mohamed Labidi, Georges Markomanolis, Adrian Muresan, Christian Pérez, Vincent Pichon, Daouda Traore.

6.2.1. Cluster Resource Allocation for Multiple Parallel Task Graphs

Many scientific applications can be structured as Parallel Task Graphs (PTGs), that is, graphs of data-parallel tasks. Adding data-parallelism to a task-parallel application provides opportunities for higher performance and scalability, but poses additional scheduling challenges. We studied the off-line scheduling of multiple PTGs on a single, homogeneous cluster. The objective was to optimize performance without compromising fairness among the PTGs. Many scheduling algorithms, both from the applied and the theoretical literature, are applicable to this problem, and we propose minor improvements when possible. Our main contribution is an extensive evaluation of these algorithms in simulation, using both synthetic and real-world application configurations, using two different metrics for performance and one metric for fairness. We identify a handful of algorithms that provide good trade-offs when considering all these metrics. The best algorithm overall is one that structures the schedule as a sequence of phases of increasing duration based on a makespan guarantee produced by an approximation algorithm.

6.2.2. Re-scheduling over the Grid

Each job submitted to a LRMS (Local Resources Manager System) must provide mandatory information like the number of requested computing resources and the requested duration of the resource usage, called *walltime*. Because the application is killed if not finished by the end of the reservation, the walltime is an over-estimation of the duration of the application launched by the job.

In the context of a Grid composed of several clusters managed by a Grid middleware which is able to tune, submit, and cancel LRMS jobs, such over-estimations have an impact on the local scheduling and performance. Consequently, previous grid scheduling, optimized at that moment, may not be relevant anymore. Thus, we have designed and studied non-intrusive mechanisms for a middleware to be able to migrate jobs still in the waiting files of the different LRMS in the Grid platform. We also proposed different scheduling heuristics integrated to the mechanims, which decide of the migration of jobs. We performed an exhaustive set of simulation experiments, in which parameters such as the load of each simulated parallel resource, the type of applications (rigid and moldable), the dedication of the platform resources, have been varied. We analyzed the performance of our propositions on different metrics which showed some counter-intuitive results.

6.2.3. Parallel constraint-based local search

Constraint Programming emerged in the late 1980's as a successful paradigm to tackle complex combinatorial problems in a declarative manner. It is somehow at the crossroads of combinatorial optimization, constraint satisfaction problems (CSP), declarative programming language and SAT problems (boolean constraint solvers and verification tools). Up to now, the only parallel method to solve optimization problems being deployed at large scale is the classical branch and bound, because it does not require much information to be communicated between parallel processes (basically: the current bound).

Adaptive Search was proposed by [89], [90] as a generic, domain-independent constraint-based local search method. This meta-heuristic takes advantage of the structure of the problem in terms of constraints and variables and can guide the search more precisely than a single global cost function to optimize, such as for instance the number of violated constraints. A parallelization of this algorithm based on threads realized on IBM BladeCenter with 16 Cell/BE cores show nearly ideal linear speed-ups for a variety of classical CSP benchmarks (magic squares, all-interval series, perfect square packing, etc.).

We parallelized the algorithm using the multi-start approach and realized experiments on the HA8000 machine, an Hitachi supercomputer with a maximum of nearly 16000 cores installed at University of Tokyo, and on the Grid'5000 infrastructure, the French national Grid for the research, which contains 5934 cores deployed on 9 sites distributed in France. Results show that speedups may surprisingly be architecture dependant, but that if they continue to grow with the number of processors, the increase tends to stabilize for some problems after 128 processes. Work in progress considers communications between each computing resource.

6.2.4. Service Discovery in Peer-to-Peer environments

Service discovery becomes a challenge in a large scale and distributed context. Heterogeneity and dynamicitu are the two main constraints that have to be taken into account in order to ensure reliability and system efficiency. Thereby, in a heterogeneous context, it is needed to equilibrate service discovery system load to get performance. Moreover, QoS in such an uncertain and dynamic environment has to be ensured by fail-safe mechanisms (self-stabilization and replication). First, Self-stabilisation ensures a consistent configuration in a convergence time. Second replication injects redundancy when the system becomes consistent. All those mechanisms will be validated and implemented. Furthermore, the service discovery system will interact with system with schedulers, batch submission systems, and storage Resource Broker. So, these component's exchange protocols have to be formally defined.

We decided to develop a new implementation, called Spades Based Middleware (SBAM) that includes all the concepts described above. This implementation, written in Java, relies on an efficient communication bus and has been developed according to advanced software engineering methods. The communication layer is based on the Ibis Portability Layer (IPL). SBAM has been evaluated with regard to service research request response time. Our experiments demonstrate the efficiency and scalability of the proposed middleware system. It was demonstrated at SuperComputing 2010.

6.2.5. On-Line Optimization of Publish/Subscribe Overlays

We continued the collaboration with the University of Nevada Las Vegas. We studied the benefit of Publish/subscribe overlays for the SPADES project. Loosely coupled applications can take advantage of the publish/subscribe communication paradigm. In this paradigm, subscribers declare which events, or which range of events, they wish to monitor, and are asynchronously informed whenever a publishers throws an event. In such a system, when a publication occurs, all peers whose subscriptions contain the publication must be informed. In our approach, the subscriptions are represented by a DR-tree, which is an R-tree where each minimum bounding rectangle is supervised by a peer. Instead of attempting to statically optimize the DR-tree, we give an on-line algorithm, the work function algorithm, which continually changes the DR-tree in response to the sequence of publications, in an attempt to dynamically optimize the structure. The competitiveness of this algorithm is computed to be at most 5 for any example where there are at most three subscriptions and the R-tree has height 2. The benefit of the on-line approach is that no prior knowledge of the distribution of publications in the attribute space is needed.

6.2.6. Décrypthon

In 2010, we added new features to, and fixed bugs of, the DIET WebBoard (a web interface for managing the Décrypthon Grid through DIET): support for multiple users on a same application, improved the database dumping method, statistics and charts, and storage space management. We deployed the newest version of DIET and the DIET WebBoard on the Décrypthon grid.

The MaxDO "Help cure muscular dystrophy, phase 2" was ported on the World Community Grid. To determine the size of the work-units sent to the World Community Grid users we ran benchmarks on Grid'5000. Finally on May 14th 2009 the project was launched and it is running since then. On December 10th 2010 a total of 30,000,549 work-unit results had been sent back by the World Community Grid volunteers, this is 64,972,205,369 positions out of 137,652,178,995 (47.2% of the project, each work-unit contains hundreds of "positions" for two proteins: the result is an energy value for this configuration). We are also checking and sorting the result files, reducing their size, and making statistics for the volunteers (cf http://graal.ens-lyon.fr/~nbard/WCGStats/). The estimated end of the project is for the end of 2011. The most recent update of the MaxDO program on our university Décrypthon grid was to add a new interface to enable researchers to easily submit batches of workunits made from results missing or skipped by the World Community Grid's desktop grid.

6.2.7. Scheduling Applications with Complex Structure

As resources become more powerful but heteregeneous, applications' structures are also becoming more complex, not only for harnessing the available power but also for more accurate modeling of physical phenomena. Efficient mapping and scheduling of applications to resources are thus becoming more challenging. However,

this is not possible with current resource management systems (RMS) that are assuming simple application models.

Therefore, we have done an initial, theoretical study of the gains one can obtain if RMS could support rigid, fully-predictable *evolving* applications. We have proposed an offline scheduling algorithm, with optional stretching capabilities. Experiments show that taking into account resource requirement evolvement leads to significant improvements in all measured metrics—such as resource utilization and completion time. However, considered stretching strategies do not appear very valuable.

Next, we have started revisiting RMS to enable efficient complex application resource selection. In 2010, we have focused on *moldable* applications. We have proposed CooRM, an RMS architecture which delegates the mapping and scheduling responsibility to the applications themselves. Simulations as well as a proof-of-concept implementation of CooRM show that the approach is feasible and performs well in terms of scalability and fairness.

As future work, we plan to extend CooRM to support evolving and malleable applications. With respect to its applicability to existing systems, we will study its integration into XtreemOS and Salome.

6.2.8. High Level Component Model

Most software component models focus on the reuse of existing pieces of code called primitive components. There are however many other elements that can be reused in component-based applications. Partial assemblies of components, well defined interactions between components and existing composition patterns (a.k.a. software skeletons) are examples of such reusable elements. It turns out that such elements of reuse are important for parallel and distributed applications.

Therefore, we have designed *High Level Component Model* (HLCM), a software component model that supports the reuse of these elements thanks to the concepts of hierarchy, genericity and connectors—and in particular the novel concepts of *open connection*. Moreover, HLCM supports multiple implementations for its elements so as to allow the optimization of applications for various hardware resources. HLCMi, an implementation of HLCM, has enabled us to validate the approach: algorithmic skeletons as well as parallel interactions such as data sharing, collective communications, and parallel method invocations have been successfully implemented.

Ongoing work includes further evaluations of HLCM with the OpenAtom application—in collaboration with Prof. Kale's team at the University of Illinois at Urbana-Champaign. Furthermore, the model will be used for the development of applications based on the MapReduce paradigm and for their efficient execution on Clouds and desktop grids in the context of the MapReduce ANR project.

6.2.9. Adaptive Mesh Refinement and Component Models

In 2010, we have studied whether component models can be useful to deal with complex application structure such as those found in adaptive mesh refinement applications (AMR). This kind of applications relies on dynamic and recursive data structures to adapt the computation grain to the simulation requirements. Though very relevant to decrease the computation load, AMR is seldom used as it is complex to implement.

Therefore, we have evaluated the feasibility of designing and implementing an AMR application—based on the heat equation—on two component models: ULCM and SALOME. Those models provide enough features but more are needed. Composite and dynamic management—such as found in ULCM—are very important to ease conception but user-defined skeletons and a mechanism to deal with domain decomposition are also welcome. HLCM enables to define user-defined skeletons but the issue of handling domain decomposition is left open.

We are investigating this problem targeting an application made of the coupling of several instances of Code_Aster, a thermomechanical calculation code from EDF R&D.

6.2.10. Cloud Resource Provisioning

Cloud client applications are able to dynamically scale based on their usage. This leads to a more efficient resource usage and, as a consequence, to expense saving. The problem is non-trivial as virtual resources have a setup time that cannot be neglected. In order to make accurate decisions when the Cloud client application needs to scale there are several valid approaches. We have focused our attention on identifying an approach that allows a Cloud client to scale his platform and compensate for the virtual resource setup time. Our approach uses self-similarities in Cloud client platform usage to predict resource usage in advance. In doing so, our approach identifies patterns in the Cloud client's past platform usage. This allows us to make usage predictions with considerable accuracy. We also shown that the prediction accuracy of our approach can be increased by increasing the size of the historic database that we use for matching.

Infrastructure as a Service clouds are a flexible and fast way to obtain (virtual) resources as demand varies. Grids, on the other hand, are middleware platforms able to combine resources from different administrative domains for tasks execution. Clouds can be used as providers of devices such as virtual machines by grids so they only use the resources they need at every moment, but this requires grids to be able to decide when to allocate and release those resources. We analyzed by simulation an economic approach to set resource prices and find when to scale resources depending on the users' demand. The results show how the proposed system can successfully adapt the to the demand, while at the same time ensuring that resources are fairly shared among users.

6.2.11. Towards Data Desktop Grid

Desktop Grids use the computing, network and storage resources from idle desktop PC's distributed over multiple-LAN's or the Internet to compute a large variety of resource-demanding distributed applications. While these applications need to access, compute, store and circulate large volumes of data, little attention has been paid to data management in such large-scale, dynamic, heterogeneous, volatile and highly distributed Grids. In most cases, data management relies on ad-hoc solutions, and providing a general approach is still a challenging issue.

We have proposed the BitDew framework which addresses the issue of how to design a programmable environment for automatic and transparent data management on computational Desktop Grids. BitDew relies on a specific set of meta-data to drive key data management operations, namely life cycle, distribution, placement, replication and fault-tolerance with a high level of abstraction.

Since July 2010, in collaboration with the University of Sfax, we are developing a data-aware and parallel version of Magik, an application for arabic writing recognition using the BitDew middleware. We are targeting digital libraries, which require distributed computing infrastructure to store the large number of digitalized books as raw images and at the same time to perform automatic processing of these documents such as OCR, translation, indexing, searching, etc.

In collaboration with the G.V.Kurdyumov Institute for Metal Physics and the LAL/IN2P3, we have developed a Desktop Grid version of the SLinCA (Scaling Laws in Cluster Aggregation) application. SLinCa simulates the several general scenarios of monomer aggregation in clusters with many initial configurations of monomers (random, regular, etc.), different kinetics law (arbitrary, diffusive, ballistic, etc.), various interaction laws (arbitrary, elastic, non-elastic, etc.). The typical simulation of one cluster aggregation process with 10 monomers takes approximately 1-7 days on a single modern processor, depending on the number of Monte Carlo steps (MCS). However, thousands of scenarios have to be simulated with different initial configurations to get statistically reliable results. To calculate the parameters of evolving aggregates (moments of probability density distributions, cumulative density distributions, scaling exponents, etc.) with appropriate accuracy (up to 2-4 significant digits), we need the better statistics (10^4 - 10^8 runs of many different statistical realizations of aggregating ensembles), which will be comparable with the same accuracy statistics of available experimental data. These separate runs of simulation for different physical parameters, initial configurations, and statistical realizations, are completely independent and can be easily split among available CPUs in a "parameter sweeping" manner of parallelism. A large number of runs, needed to reduce the standard deviation in Monte

Carlo simulations, are distributed equally among available workers and are combined at the end to calculate the final result.

6.2.12. MapReduce programing model for Desktop Grid

MapReduce is an emerging programming model for data-intense application proposed by Google, which has recently attracted a lot of attention. MapReduce borrows from functional programming, where programmer defines Map and Reduce tasks executed on large sets of distributed data. In 2010, we have developed an implementation of the MapReduce programming model based on the BitDew middleware. Our prototype features several optimizations which make our approach suitable for large scale and loosely connected Internet Desktop Grid: massive fault tolerance, replica management, barriers-free execution, latency-hiding optimization as well as distributed result checking. We have presented performance evaluations of the prototype both against micro-benchmarks and real MapReduce applications. The scalability test shows that we achieve linear speedup on the classical WordCount benchmark. Several scenarios involving lagger hosts and host crashes demonstrate that the prototype is able to cope with an experimental context similar to real-world Internet.

6.2.13. SpeQuloS: Providing Quality-of-Service to Desktop Grids using Cloud resources

EDGI is an FP7 European project, following the successful FP7 EDGeS project, whose goal is to build a Grid infrastructure composed of "Desktop Grids", such as BOINC or XtremWeb, where computing resources are provided by Internet volunteers, and "Service Grids", where computing resources are provided by institutional Grid such as EGEE, gLite, Unicore and "Clouds systems" such as OpenNebula and Eucalyptus, where resources are provided on-demand. The goal of the EDGI project is to provide an infrastructure where Service Grids are extended with public and institutional Desktop Grids and Clouds.

The main problem with the current infrastructure is that it cannot give any QoS support for running their applications in the Desktop Grid (DG) part of the infrastructure. For example, a public DG system enables clients to return work-unit results in the range of weeks. Although there are EGEE applications (e.g. the fusion community's applications) that can tolerate such a long latency most of the user communities want much smaller latencies.

In 2010, we have started the development and deployment of the SpeQuloS middleware to solve this critical problem.

We define QoS concretely as a probabilistic guarantee of job makespan or throughput. Providing QoS features even in Service Grids is hard and not solved yet satisfactorily. It is even more difficult in an environment where there are no guaranteed resources. In DG systems, resources can leave the system at any time for a long time or forever even after taking several work-units with the promise of computing them. Our approach is based on the extension of DG systems with Cloud resources. For such critical work-units the SpeQuloS system is able to dynamically deploy fast and trustable clients from some Clouds that are available to support the EDGI DG systems. It takes the right decision about assigning the necessary number of trusted clients and Cloud clients for the QoS applications. At this stage, the prototype is functional and the first version is planned to be delivered to the EDGI production infrastructure during spring 2011.

6.2.14. Performance evaluation and modeling

Simulation is a popular approach to obtain objective performance indicators of platforms that are not at one's disposal. It may for example help the dimensioning of compute clusters in large computing centers. In many cases, the execution of a distributed application does not behave as expected, it is thus necessary to understand what causes this strange behavior. Simulation provides the possibility to reproduce experiments under similar conditions. This is a suitable method for experimental validation of a parallel or distributed application.

The tracing instrumentation of a profiling tool is the ability to save all the information about the execution of an application at run-time. Every scientific application executed computes floating point operations (flops). The originality of our approach is that we measure the flops of the application and not its execution time. This means that if a distributed application is executed on N cores and we execute it again by mapping two processes

per core then we need N/2 cores and more time for the execution time of the application. An execution trace of an instrumented application can be transformed into a corresponding list of actions. These actions can then be simulated by SimGrid. Moreover the SimGrid execution traces will contain almost the same data because the only change is the use of half cores but the same number of processes. This does not affect the number of the flops so the simulation time does not get increased because of the overhead. The Grid'5000 platform is used for this work and the NAS Parallel Benchmarks are used to measure the performance of the clusters.

6.3. Parallel Sparse Direct Solvers and Combinatioral Scientific Computing

Participants: Maurice Brémond, Indranil Chowdhury, Guillaume Joslin, Jean-Yves L'Excellent, Bora Uçar.

6.3.1. Some Experiments and Issues to Exploit Multicore Parallelism in a Distributed-Memory Parallel Sparse Direct Solver

MUMPS (see Section 5.2) is a parallel sparse direct solver, using message passing (MPI) for parallelism. In this work we have experimented how thread parallelism can help taking advantage of recent multicore architectures. The work done consists in testing multithreaded BLAS libraries and inserting OpenMP directives in the routines revealed to be costly by profiling, with the objective to avoid any deep restructuring or rewriting of the code. In INRIA report RR-7411 (October 2010), we have reported on various aspects of this work, presented some of the benefits and difficulties, and showed that 4 to 8 threads per MPI process is generally a good compromise for performance, while increasing the number of threads is always interesting in terms of memory usage. We also considered and discussed several issues that appear to be critical with a mixed MPI-OpenMP approach in a multicore environment. In the future we plan to pursue this work on larger numbers of cores.

6.3.2. Design, Implementation, and Analysis of Maximum Transversal Algorithms

We have investigated seven maximum traversal algorithms. We report on their careful implementations. The algorithms are analyzed and design choices are discussed. To the best of our knowledge, this is the most comprehensive comparison of maximum transversal algorithms based on augmenting paths. Previous papers with the same objective either do not have all the algorithms discussed in this paper or they use non-uniform implementations from different researchers. We use a common base to implement all of the algorithms and compare their relative performance on a wide range of graphs and matrices. We systematize, develop and use several ideas for enhancing performance. One of these ideas improves the performance of one of the existing algorithms in most cases, sometimes significantly. So much so that we use this as the eighth algorithm in comparisons.

6.3.3. On computing inverse entries of a sparse matrix in an out-of-core environment

The inverse of an irreducible sparse matrix is structurally full, so that it is impractical to think of computing or storing it. However, there are several applications where a subset of the entries of the inverse is required. Given a factorization of the sparse matrix held in out-of-core storage, we show how to compute such a subset efficiently, by accessing only parts of the factors. When there are many inverse entries to compute, we need to guarantee that the overall computation scheme has reasonable memory requirements, while minimizing the cost of loading the factors. This leads to a partitioning problem that we prove is NP-complete. We also show that we cannot get a close approximation to the optimal solution in polynomial time. We thus need to develop heuristic algorithms, and we propose: (i) a lower bound on the cost of an optimum solution; (ii) an exact algorithm for a particular case; (iii) two other heuristics for a more general case; and (iv) hypergraph partitioning models for the most general setting. We illustrate the performance of our algorithms in practice using the MUMPS software package on a set of real-life problems as well as some standard test matrices. We show that our techniques can improve the execution time by a factor of 50.

6.3.4. The minimum degree ordering with dynamical constraints

We propose a modification of the minimum degree ordering algorithm in which some variables are constrained to be ordered only after some other nodes are ordered. The constrained variables are initially specified, and their constraints are removed during the course of the algorithm. This is close to the minimum degree ordering with constraints algorithm. The difference is that during the course of our algorithm we remove some of the constraints, whereas the constraints are static in the current constrained ordering algorithms. Such an algorithm can have different applications; we target the ordering problem for saddle point matrices.

6.3.5. On finding dense submatrices of a sparse matrix

We consider a family of problems exemplified with the following one: Given an $m \times n$ matrix A and an integer $k \leq \min\{m, n\}$, find a set of row indices $\mathcal{R} = \{r_1, r_2, ..., r_k\}$ and a set of column indices $\mathcal{C} = \{c_1, c_2, ..., c_k\}$ such that the number of nonzeros in the submatrix indexed by $\mathcal{R}$ and $\mathcal{C}$, i.e., $A(\mathcal{R}, \mathcal{C})$ in Matlab notation, is maximized. This is equivalent to finding a $k \times k$ submatrix S of A with entries $S_{ij} = A_{r_i,c_j}$ such that it contains the maximum number of nonzeros among all $k \times k$ submatrices of A. We show that this problem is NP-complete, and then propose and analyze heuristic approaches to the problem. The problems of this nature arises in a family of hybrid solvers for sparse linear systems.

7. Contracts and Grants with Industry

7.1. Contract with SAMTECH, 2008-2010

INRIA and INPT-IRIT have signed a new contract with the company Samtech S.A. (Belgium). Samtech develops the finite element software package SAMCEF, which uses our parallel sparse direct solver MUMPS as one of the internal solvers. The goal of this contract is to improve the memory usage of MUMPS, and to offer the possibility to address a larger amount of memory. We will also study how to use memory already allocated by SAMCEF instead of having the solver allocate its own memory. Finally we also plan to study how performance can be improved on Samtech problems by allowing the forward substitution step to be performed simultaneously with the matrix factorization. This last point is particularly interesting in the case of out-of-core executions.

The contract is 24-month long, and the new functionalities developed in MUMPS for this contract will be made available in a future public release of the package.

J.-Y. L'Excellent is the principal investigator for the LIP; M. Brémond, G. Joslin, and B. Uçar participate to this contract.

8. Other Grants and Activities

8.1. Regional Projects

8.1.1. Pôle Scientifique de Modélisation Numérique (PSMN), Fédération Lyonnaise de Modélistaion et Sciences Numériques

PSMN is a federation of laboratories that aims at sharing the parallel machines from ENS Lyon/PSMN and experiences of parallelization of applications. FLMSN is a wider structure, replacing the FLCHP (Fédération Lyonnaise de Calcul Hautes Performances).

J.-Y. L'Excellent is the correspondent of the LIP in these two structures.

8.1.2. Projet "Calcul Hautes Performances et Informatique Distribuée"

E. Caron leads (with C. Prudhomme from LJK, Grenoble) the "Calcul Hautes Performances et Informatique Distribuée" project of the cluster "Informatique, Signal, Logiciels Embarqués". Together with several research laboratories from the Rhône-Alpes region, we initiate collaborations between application researchers and distributed computing experts.

Y. Caniou, E. Caron, F. Desprez, J.-Y. L'Excellent, and F. Vivien participate to this project.

8.2. National Contracts and Projects

8.2.1. ANR Blanche: Stochagrid (Scheduling algorithms and stochastic performance models for workflow applications on dynamic Grid platforms), 3 years, ANR-06-BLAN60192-01, 2007-2010

In the third and final year of the project (2010), we have pursued the investigation of timed Petri nets to model the mapping of workflows with stage replication, in collaboration with Bruno Gaujal (LIG Grenoble).

Also, we have investigated several multi-criteria algorithms and heuristics for the problem of mapping pipelined applications, consisting of a linear chain of stages executed in a pipeline way, onto heterogeneous platforms. This work was conducted by the post-doctoral student hired on the project, Hinde Bouziane, and the first paper published on this work received the best paper award.

The project is entirely conducted within the GRAAL team by A. Benoit and Y. Robert.

8.2.2. ANR grant Gwendia ANR-06-MDCA-009 (Grid Workflow Efficient Enactment for Data Intensive Applications), 3 years, 2007-2010

The objective of the Gwendia[2] project is to design and develop workflow management systems for applications involving large amounts of data. It is a multidisciplinary project involving researchers in computer science (including GRAAL) and in life science (medical imaging and drug discovery). Our work consists in designing algorithms for the management of several workflows in distributed and heterogeneous platforms and to validate them within DIET over the Grid'5000 platform.

8.2.3. ANR grant SPADES, 3 years, 08-ANR-SEGI-025, 2009-2012

Today's emergence of Petascale architectures and evolutions of both research grids and computational grids increase a lot the number of potential resources. However, existing infrastructures and access rules do not allow to fully take advantage of these resources. One key idea of the SPADES project is to propose a non-intrusive but highly dynamic environment able to take advantage of the available resources without disturbing their native use. In other words, the SPADES vision is to adapt the desktop grid paradigm by replacing users at the edge of the Internet by volatile resources. These volatile resources are in fact submitted via batch schedulers to reservation mechanisms which are limited in time or susceptible to preemption (best-effort mode).

One of the priorities of SPADES is to support platforms at a very large scale. Petascale environments are therefore particularly considered. Nevertheless, these next-generation architectures still suffer from a lack of expertise for an accurate and relevant use. One of the SPADES goal is to show how to take advantage of the power of such architectures. Another challenge of SPADES is to provide a software solution for a service discovery system able to face a highly dynamic platform. This system will be deployed over volatile nodes and thus must tolerate failures. SPADES will propose solutions for the management of distributed schedulers in Desktop Computing environments, coping with a co-scheduling framework.

[2] http://gwendia.polytech.unice.fr/doku.php

8.2.4. ANR grant: COOP (Multi Level Cooperative Resource Management), 3 years, ANR-09-COSI-001-01, 2009-2012

The main goals of this project are to set up such a cooperation as general as possible with respect to programming models and resource management systems and to develop algorithms for efficient resource selection. In particular, the project targets the SALOME platform and GRID-TLSE expert-site (http://gridtlse.org/) as an example of programming models, and Marcel/PadicoTM, DIET and XtreemOS as examples of multithread scheduler/communication manager, grid middleware and distributed operating systems.

The project is led by Christian Pérez.

8.2.5. ANR JCJC: Clouds@Home (Cloud Computing over Unreliable, Shared Resources), 4 years, ANR-09-JCJC-0056-01, 2009-2012

Recently, a new vision of cloud computing has emerged where the complexity of an IT infrastructure is completely hidden from its users. At the same time, cloud computing platforms provide massive scalability, 99.999% reliability, and speedy performance at relatively low costs for complex applications and services. This project, lead by D. Kondo from INRIA MESCAL investigates the use of cloud computing for large-scale and demanding applications and services over unreliable resources. In particular, we target volunteered resources distributed over the Internet. In this project, G. Fedak leads the Data management task (WP3).

8.2.6. ANR ARPEGE MapReduce (Scalable data management for Map-Reduce-based data-intensive applications on cloud and hybrid infras tructures), 4 years, ANR-09-JCJC-0056-01, 2010-2013

MapReduce is a parallel programming paradigm successfully used by large Internet service providers to perform computations on massive amounts of data. After being strongly promoted by Google, it has also been implemented by the open source community through the Hadoop project, maintained by the Apache Foundation and supported by Yahoo! and even by Google itself. This model is currently getting more and more popular as a solution for rapid implementation of distributed data-intensive applications. The key strength of the Map-Reduce model is its inherently high degree of potential parallelism.

In this project, the GRAAL team participates to several work packages which address key issues such as efficient scheduling of several MR applications, integration using components on large infrastructures, security and dependability, MapReduce for Desktop Grid.

8.2.7. ANR Blanche: RESCUE (Resilience for exascale scientific computing), 4 years, ANR-2010-BLAN-0301-01, 2010-2014

The emergence of exascale computers will enable to solve new scientific challenges. However, the scientific applications deployed on such machines comprising up to millions of cores will have to cope with numerous failures: it is forecasted that with the current techniques, the mean time between two consecutive failures will be shorter than the time needed to checkpoint an application using the whole platform. The main objective of the RESCUE project is to develop new algorithm techniques and new software to solve the fault-tolerance problem on exascale machines.

The RESCUE project is led by Y. Robert and involves three INRIA teams: Grand-Large, HiePACS and GRAAL (A. Benoit, L. Marchal, F. Vivien).

8.2.8. ADT-MUMPS, 3 years, 2009-2012

ADT-MUMPS is an action of technological development funded by INRIA. This project gives support for 24 men x months of young engineer (“ingénieur jeune diplômé”). A permanent engineer from INRIA/SED also works on the project (Maurice Brémond, 30 % on the project). One goal of the project is to improve daily work of MUMPS developers by improving the software engineering aspects, by developing non-regression tests and drivers to experiment the package. This project is in collaboration with ENSEEIHT-IRIT.

8.2.9. ADT ALADDIN

ALADDIN is an INRIA action of technological development for "A LArge-scale DIstributed and Deployable INfrastructure" which aim is to manage the Grid'5000 experimental platform. Frédéric Desprez is leading this project (with David Margery from Rennes as the Technical Director).

8.2.10. ADT BitDew, 2 years, 2010-2012

ADT BitDew is an INRIA support action of technological development for the BitDew middleware. Objectives are several fold : i/ provide documentation and education material for end-users, ii/ improve software quality and support, iii/ develop new features allowing the management of Cloud and Grid resources. The ADT BitDew, leaded by G. Fedak, allows to recruit a young engineer for 24 months.

8.2.11. HEMERA Large Wingspan Inria Project

Hemera deals with the scientific animation of the Grid'5000 community. It aims at making progress in the understanding and management of large scale infrastructure by leveraging competences distributed in various French teams. Hemera contains several scientific challenges and working groups. Christian Pérez is leading the project that involves more than 20 teams located in 9 cities of France.

8.2.12. Action Interfaces Recherche en grille – Grilles de production. Institut des Grilles du CNRS – Action Aladdin INRIA

This action addresses economical issues concerning green-ness in scientific and production grids. Different issues are addressed like the confrontation of energy models in place in experimental grids versus the operational realities in production grids, the study of new energy prediction models related to real measures of energy consumption in production grids, and the design of energy aware scheduling heuristics.

Y. Caniou participates to this action.

8.2.13. SmartGame: Regional Grant

The SmartGame start'up asked to take benefit of the knowledge of the GRAAL research team on distributed systems and middleware systems. The aim of this company is to create games of new generation using a new distributed architecture. E. Caron and F. Desprez participate to this action.

8.3. European Contracts and Projects

8.3.1. ERCIM WG CoreGRID (2009-2011)

Following the success of the NoE CoreGRID, an ERCIM WG was started in 2009, leaded by F. Desprez. This working group gathers 31 research teams from all over Europe working on Grids, service oriented architectures and Clouds.

A workshop on Grids, Clouds, and P2P Computing was organized in conjunction with EuroPAR 2010, Ischia, August, 2010.

8.3.2. EU FP7 project EDGeS: Enabling Desktop Grids for e-Science (2008-2010)

This project is lead by P. Kacsuk, and involves the following partners : SZTAKI, INRIA, CIEMAT, Fundecyt, University of Westminster, Cardiff University, University of Coimbra. Grid systems are currently being used and adopted by a growing number of user groups and diverse application domains. However, there still exist many scientific communities whose applications require much more computing resources than existing Grids like EGEE can provide. The main objective of this project is to interconnect the existing EGEE Grid infrastructure with existing Desktop Grid (DG) systems like BOINC or XTREMWEB in a strong partnership with EGEE. The interconnection of these two types of Grid systems will enable more advanced applications and provide extended compute capabilities to more researchers. In this collaboration G. Fedak represents the GRAAL team and is responsible for JRA1: Service Grids-Desktop Grids Bridges Technologies and is involved in JRA3 : Data Management, as well as NA3 : Standardization within the OGF group.

8.3.3. EU FP7 project EDGI : European Desktop Grid Initiative (2010-2012)

The project EDGI will develop middleware that consolidates the results achieved in the EDGeS project concerning the extension of Service Grids with Desktop Grids in order to support EGI and NGI user communities that are heavy users of DCIs and require extremely large number of CPUs and cores. EDGI will go beyond existing DCIs that are typically cluster Grids and supercomputer Grids, and will extend them with public and institutional Desktop Grids and Clouds. EDGI will integrate software components of ARC, gLite, Unicore, BOINC, XWHEP, 3G Bridge, and Cloud middleware such as OpenNebula and Eucalyptus into SG→DG→Cloud platforms for service provision and as a result EDGI will extend ARC, gLite and Unicore Grids with volunteer and institutional DG systems. Our partners in EDGI are : SZTAKI, INRIA, CIEMAT, Fundecyt, University of Westminster, Cardiff University, University of Coimbra. In this project, G. Fedak is the INRIA representative and lead the JRA2 work package which is responsible for providing QoS to Desktop Grids.

8.4. International Contracts and Projects

8.4.1. French-Israeli project "Multicomputing" (2009-2010)

This project aims at improving the scalability of state-of-the-art computational fluid dynamics calculations by the use of state-of-the-art numerical linear algebra approaches. It mainly involves Tel Aviv University and ENSEEIHT-IRIT (Toulouse), where Alfredo Buttari is coordinator for the French side. In GRAAL, I. Chowdhury, J.-Y. L'Excellent, and B. Uçar participate to this project.

8.4.2. Associated-team MetagenoGrid (2008-2010)

The collaboration is done with the Concurrency Research Group (CoRG) of Henri Casanova, and the Bioinformatics Laboratory (BiL) of Guylaine Poisson of the Information and Computer Sciences Department, of the University of Hawai'i at Manoā, USA.

The associated-team targets the efficient scheduling of large-scale scientific applications on clusters and Grids. To provide context for this research, we focus on applications from the domain of bioinformatics, in particular comparative genomics and metagenomics applications, which are of interest to a large user community today. So far, applications (in bioinformatics or other fields) that have been successfully deployed at a large scale fall under the "independent task model": they consist of a large number of tasks that do not share data and that can be executed in any order. Furthermore, many of these application deployments rely on the fact that the application data for each task is "small", meaning that the cost of sending data over the network can be ignored in the face of long computation time. However, both previous assumptions are not valid for all applications, and in fact many crucial applications, such as the aforementioned bioinformatics applications, require computationally dependent tasks sharing very large data sets.

In our previous collaborations, we have tackled the issue of non-negligible network communication overheads and have made significant contributions. For instance, we have designed strategies that rely on the notions of steady-state scheduling (i.e., attempting to maximize the number of tasks that complete per time unit, in the long run) and/or divisible load scheduling (i.e., approximate the discrete workload that consists of individual tasks as a continuous workload). These strategies provide powerful means for rethinking the deployment and the scheduling of independent task applications when network communication can be a bottleneck. However, the target applications in this project cannot benefit from these strategies directly and will require fundamental advances. This project aims to build upon and go beyond our past collaborations, with two main research thrusts:

- Scheduling of applications with data requirements. We consider applications that require possibly multiple data files that need to be shared by multiple application tasks. These files may be extremely large (e.g., millions of genomic sequences) and may need to be updated frequently (e.g., when new sequences are identified). We must then ensure that file access is not a bottleneck.

- Scheduling of multiple concurrent applications. We also plan to study the scheduling for multiple applications, i.e., launched by different (most likely competing) users. We then aim to orchestrate computation and communication in order to have the best aggregate performance. This is a difficult problem, first in order to define a good performance metric, and then to maximize this performance metric in a tractable way.

A. Benoit, E. Caron, F. Desprez, Y. Robert and F. Vivien participate to this project.

8.4.3. French-Japanese ANR-JST FP3C project

This project federates INRIA Saclay, CNRS IRIT, CEA Saclay, INRIA Bordeaux, CNRS Prism, INRIA Rennes on the French side and the University of Tokyo, The University of Tsukuba, Titech, Kyoto University on the Japanese side. The main goal of the project is to develop a programming chain and associated runtime systems which will allow scientific end-users to efficiently execute their applications on post-petascale, highly hierarchical computing platforms making use of multi-core processors and accelerators.

Y. Caniou and J.-Y. L'Excellent participate to this project.

8.4.4. CNRS délégation of Yves Caniou (2010-2011)

Yves Caniou obtained a CNRS delegation for the scholar year 2009-2010, and this delegation has been prolongated for the scholar year 2010-2011. He is working at the CNRS Japan-French Laboratory in Informatics (JFLI) supervised by Philippe Codognet. The JFLI is located in Tokyo, Japan, and is composed of the Tokyo University, Université Pierre et Marie-Curie (UPMC), the Keio University, the CNRS, the NII partnership.

9. Dissemination

9.1. Scientific Missions

MUMPS User Group Meeting The MUMPS team organized a MUMPS User Group Meeting on April 15th and 16th 2010 at ENSEEIHT, Toulouse. This was the second edition of a series of meetings started with the 2006 MUMPS User Group Meeting. The aim of this event was to bring together experts both from academia and industry. The general theme of the meeting was sparse direct solvers and related issues, ranging from applications to experiences with MUMPS and other direct solvers, and combinatorial ingredients of sparse direct solvers.

Scheduling in Aussois, III The GRAAL project at École normale supérieure de Lyon organized a workshop in Aussois, France on June 2–4, 2010. The workshop focused on scheduling for large-scale systems and on scientific computing. This was the fifth edition of this workshop series, after Aussois in August 2004, San Diego in November 2005, Aussois in May 2008, and Knoxville in May 2009.

"Des grilles aux Clouds, nouveaux problèmes et nouvelles solutions" day at ENS The GRAAL project organized a day around Cloud platforms and research issues at ENS Lyon on December 13. This event that gathered more than 180 attendees allowed to share experiences and solutions both from academia and industry for the management of large scale virtualized resources.

9.2. Edition and Program Committees

Anne Benoit was the Program Chair of the 19th International Heterogeneity in Computing Workshop, HCW 2010, held in Atlanta, USA, April 2010, in conjunction with IPDPS 2010, and she is the General Chair of HCW 2011 in Anchorage, USA, May 2011 (in conjuction with IPDPS 2011). She co-organized the 7th International Workshop on Practical Aspects of high-level Parallel Programming (PAPP 2010) in Amsterdam, The Netherlands, May 2010.

A. Benoit was a member of the program committee of IPDPS 2010, HiPC 2010, HLPP 2010, APDCM 2010, ICCS 2010. She is a member of the program committee of IPDPS 2011 and SPAA 2011.

Yves Caniou is a member of the program committee of Heterogeneous Computing Workshop 2010 and 2011, and of the ICCSA 2010 and 2011 conferences.

Eddy Caron was a member of the program committee of PDP 2010, ISPA'2010, HCW'10, MapRed'2010, and CloudCom 2010.

He is co-chair of Grid-RPC group in the OGF (Open Grid Forum). He is a co-funder of the SysFera startup company and continue to be involved as a scientific consultant.

Frédéric Desprez is a member of the EuroPar Advisory board and the editorial board of "Scalable Computing: Practice and Experience" (SCPE).

F. Desprez participated to the program committees of DEPEND'2010, CCGRID 2010, EuroMPI'2010, VECPAR'10, CCGrid-Health 2010, workshop Grids meet Autonomic Computing, InterCloud2010. He was the vice-chair of the scheduling topic EuroPar'2010, LaSCoG 2010, the vice-chair of the "Tools/Software/Middleware" topic of Grid'2010, the program chair of the VTDC workshop in conjunction with HPDC'10, Cluster and Cloud Computing Track for IEEE ICPADS2010, CloudCom 2010, HeteroPar'2010, CloudComp,10, MobiCloud'2010, Cloud and Grid Computing track of AICCSA'2010.

Gilles Fedak co-chaired 2 workshops PCGRID'10 and MAPREDUCE'10 associated respectively with CCGRID (Melbourne Australia, 2010) and HPDC (Chicago, USA, 2010). He was the track co-chair for the High-speed Distributed Systems and Grids (HDSG) track in the 19th IEEE International Conference on Computer Communications and Networks (ICCCN), Zurich, Switzerland, August 2010. He was a member of the program committees of the following conferences and workshops : CloudCom 2010, (Indianapolis, USA, 2010 CoreGrid'10 , associated with Europar, (Ischia - Naples, Italy, 2010), MapRed'10, associated with CloudCom'2010, (Indiannapolis, USA, 2010)

He co-chairs 2 workshops PCGRID'11 and MAPREDUCE'11 associated respectively with IPDPS (Anchorage, Alaska 2011) and HPDC (San José, CA, 2011). He is a member of the program committee of HPDC'2011 (San Jose, California, 2011), CCGRID 2011, (Newport Beach, CA, 2011), ScalCom-11 (Cyprus, 2011), RenPar'20, (Saint-Malo, France, 2011), DICTAP 2011, (Dijon, France, 2011), 3DAPAS, in conjunction with HPDC 2011, (San Jose, CA, USA, 2011), MSOP2P'11, in conjunction with EuroMicro PDP 2011, (Ayia Napa, Cyprus, 2011)

Jean-Yves L'Excellent was a member of the program committee of VECPAR'10 (Berkeley, California).

Loris Marchal was a member of the program committee of ICNC 2010, LaSCoG 2010, IPDPS 2011 and HCW 2011.

Christian Pérez was a member of the program committee of VECPAR'10 (Berkeley, CA, USA, June 22-25, 2010), CBHPC (Brussels, Belgium, October 26, 2010), HPCC (Melbourne, Australia, September 1-3, 2010), and FMMC (Heidelberg Academy of Sciences, Germany, March 17-19, 2010)

He is a local chair of Euro-Par 2011 (Bordeaux, France, August 29-September 2011). He is a member of the program committee of ParCo (Ghent, Belgium, August 30-September 2, 2011), HipHaC (San Antonio, Texas, USA, February 12, 2011), MapReduce (San Jose, California, USA, June 2011), Renpar'20 (St Malo, France, May 10-13, 2011). He is a member of the Steering Committee of CBHPC.

C. Pérez serves as expert for evaluating proposal to the 2010 "White" call of ANR.

Yves Robert is a member of the editorial board of the *International Journal of High Performance Computing Applications* (Sage Press), of the *Journal of Computational Science* (Elsevier), and of the *International Journal of Grid and Utility Computing*.

Y. Robert was Program Chair of HiPC'2010, track Algorithms and Applications. He will be program vice-chair of ICPP'2011, track Algorithms and Applications.

Yves Robert is a member of the Steering Committee of HCW (IEEE Workshop on Heterogeneity in Computing) of IPDPS (IEEE Int. Parallel and Distributed Symposium), and of HeteroPar (Int. Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Platforms).

Bora Uçar was a member of the program committee of Algorithms and Applications track of ICNC'10, the First International Conference on Networking and Computing, Higashi Hiroshima, Japan, November 17–19, 2010. He was also a member of the program committee of IPDPS 2010, TCPP PhD forum.

B. Uçar organized a mini-symposium entitled "Parallel sparse matrix computations and enabling algorithms"as a part of SIAM Conference on Parallel Processing for Scientific Computing (PP10), February 24–266, 2010, Seattle, Washington, USA.

Frédéric Vivien is an associate editor of *Parallel Computing*.

F. Vivien was a member of the program committee of EuroPDP 2010, Pisa, Italy, February 2010, HiPC'2010, Goa, India, December 19-22, 2010. and Cluster 2010, Heraklion, Greece, September 20-24, 2010. He was Program chair of HeteroPar 2010 (the 8th International Workshop on Algorithms, Models, and Tools for Parallel Computing on Heterogeneous Platforms), Ischia-Naples, Italy, August 31, 2010.

9.3. Administrative and Teaching Responsibilities

9.3.1. Teaching Responsibilities

Licence d'Informatique Fondamentale at ENS Lyon. Anne Benoit was responsible of the 3rd year students on fundamental computer science at ENS Lyon until August 2010. She gave a course on algorithms to the 3rd year students.

Eddy Caron and Christian Pérez offered CR11-Grid and Cloud Computing lecture series in the Master d'Informatique at ENS Lyon.

Jean-Yves L'Excellent and Bora Uçar offered CR07-Sparse matrix computations lecture series in the Master d'Informatique Fondamentale at ENS Lyon.

Loris Marchal offered CR08-Scheduling lecture series in the Master d'Informatique Fondamentale at ENS Lyon.

Yves Robert is a member (in fact, the only European member) of the NSF/TCPP initiative on the parallel and distributed computing (PDC) curriculum. A working group from IEEE TCPP, NSF, and the sisters communities has taken up the task of proposing a curriculum for computer science (CS) and computer engineering (CE) undergraduates on parallel and distributed computing. The goal of this committee has been to propose a core curriculum for CS/CE undergraduates, with the premise that every such undergraduate should achieve a specified skill level regarding PDC-related topics as a result of required coursework. Over the last months, the working group has deliberated upon various topics and subtopics, agreed upon their learning outcomes and level of coverage, has identified where in current core courses these could be introduced, and has provided examples of how they might be taught. Limited reviews have been carried out by selected stakeholders. Early adopters in Fall-10 and Spring-11 will be employing and evaluating the proposed curriculum. See http://www.cs.gsu.edu/~tcpp/curriculum/index.php for more details.

Frédéric Vivien gave a course on Parallel Algorithms to 2nd year students.

10. Bibliography

Major publications by the team in recent years

[1] P. R. AMESTOY, I. S. DUFF, J. KOSTER, J.-Y. L'EXCELLENT. *A Fully Asynchronous Multifrontal Solver Using Distributed Dynamic Scheduling*, in "SIAM Journal on Matrix Analysis and Applications", 2001, vol. 23, n^{O} 1, p. 15-41.

[2] C. BANINO, O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, Y. ROBERT. *Scheduling strategies for master-slave tasking on heterogeneous processor platforms*, in "IEEE Trans. Parallel Distributed Systems", 2004, vol. 15, n^{o} 4, p. 319-330.

[3] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Centralized versus distributed schedulers for multiple bag-of-task applications*, in "IEEE Trans. Parallel Distributed Systems", 2008, vol. 19, n^{o} 5, p. 698-709.

[4] 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", 2005, vol. 16, n^{o} 3, p. 207-218.

[5] A. BENOIT, V. REHN-SONIGO, Y. ROBERT. *Replica placement and access policies in tree networks*, in "IEEE Trans. Parallel Distributed Systems", 2008, vol. 19, n^{o} 12, p. 1614-1627.

[6] E. CARON, F. DESPREZ. *DIET: A Scalable Toolbox to Build Network Enabled Servers on the Grid*, in "International Journal of High Performance Computing Applications", 2006, vol. 20, n^{o} 3, p. 335-352.

[7] F. DESPREZ, J. DONGARRA, A. PETITET, C. RANDRIAMARO, Y. ROBERT. *Scheduling block-cyclic array redistribution*, in "IEEE Trans. Parallel Distributed Systems", 1998, vol. 9, n^{o} 2, p. 192-205.

[8] F. DESPREZ, F. SUTER. *Impact of Mixed-Parallelism on Parallel Implementations of Strassen and Winograd Matrix Multiplication Algorithms*, in "Concurrency and Computation: Practice and Experience", July 2004, vol. 16, n^{o} 8, p. 771–797.

[9] A. GUERMOUCHE, J.-Y. L'EXCELLENT. *Constructing Memory-minimizing Schedules for Multifrontal Methods*, in "ACM Transactions on Mathematical Software", 2006, vol. 32, n^{o} 1, p. 17–32.

[10] A. LEGRAND, A. SU, F. VIVIEN. *Minimizing the stretch when scheduling flows of divisible requests*, in "Journal of Scheduling", 2008, vol. 11, n^{o} 5, p. 381-404.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[11] J. BIGOT. *Du support générique d'opérateurs de composition dans les modèles de composants logiciels, application au calcul scientifique*, INSA de Rennes, December 2010.

[12] E. CARON. *Contribution to the management of large scale platforms: the DIET experience*, École Normale Supérieure de Lyon, October6 2010, HDR (Habilitation à Diriger les Recherches).

[13] B. DEPARDON. *Contribution to the Deployment of a Distributed and Hierarchical Middleware Applied to Cosmological Simulations*, Ecole Normale Supérieure de Lyon, October 6 2010.

Articles in International Peer-Reviewed Journal

[14] K. AGRAWAL, A. BENOIT, F. DUFOSSÉ, Y. ROBERT. *Mapping filtering streaming applications*, in "Algorithmica", 2010, To appear.

[15] E. AGULLO, A. GUERMOUCHE, J.-Y. L'EXCELLENT. *Reducing the I/O Volume in Sparse Out-of-core Multifrontal Methods*, in "SIAM Journal on Scientific Computing", 2010, vol. 31, n^{o} 6, p. 4774-4794.

[16] A. BENOIT, H. CASANOVA, V. REHN-SONIGO, Y. ROBERT. *Resource allocation strategies for constructive in-network stream processing*, in "International Journal of Foundations of Computer Science", 2010, To appear.

[17] A. BENOIT, H. CASANOVA, V. REHN-SONIGO, Y. ROBERT. *Resource allocation strategies for multiple concurrent in-network stream processing applications*, in "Parallel Computing", 2011, To appear.

[18] A. BENOIT, M. HAKEM, Y. ROBERT. *Multi-criteria scheduling of precedence task graphs on heterogeneous platforms*, in "The Computer Journal", 2010, vol. 53, n^{o} 6, p. 772-785.

[19] A. BENOIT, L. MARCHAL, J.-F. PINEAU, Y. ROBERT, F. VIVIEN. *Scheduling concurrent bag-of-tasks applications on heterogeneous platforms*, in "IEEE Transactions on Computers", 2010, vol. 59, n^{o} 2, p. 202-217.

[20] A. BENOIT, Y. ROBERT. *Complexity results for throughput and latency optimization of replicated and data-parallel workflows*, in "Algorithmica", 2010, vol. 57, n^{o} 4, p. 689-724.

[21] A. BENOIT, Y. ROBERT, A. ROSENBERG, F. VIVIEN. *Static worksharing strategies for heterogeneous computers with unrecoverable interruptions*, in "Parallel Computing", 2010, To appear.

[22] E. CARON, A. K. DATTA, B. DEPARDON, L. L. LARMORE. *A Self-Stabilizing K-clustering algorithm for weighted graphs*, in "Journal of Parallel and Distributed Computing", Nov 2010, vol. 70, n^{o} 11, p. 1159-1173.

[23] E. CARON, F. DESPREZ, A. MURESAN. *Forecasting for Cloud Computing On-Demand Resources Based on Pattern Matching*, in "Journal of Grid Computing", 2011, To appear.

[24] E. CARON, F. DESPREZ, A. MURESAN, L. RODERO-MERINO. *Recent development in DIET: from Grid to Cloud*, in "ERCIM News. Special Theme: "Cloud Computing Platforms, Software, and Applications"", October 2010, vol. No. 83, To appear.

[25] E. CARON, F. DESPREZ, F. PETIT, C. TEDESCHI. *Snap-Stabilizing Prefix Tree for Peer-to-Peer Systems*, in "Parallel Processing Letters", March 2010, vol. 20, n^{o} 1, p. 15-30.

[26] H. CASANOVA, F. DESPREZ, F. SUTER. *On Cluster Resource Allocation for Multiple Parallel Task Graphs*, in "Journal of Parallel and Distributed Computing", December 2010, vol. 70, n^{o} 12, p. 1193–1203.

[27] I. S. DUFF, B. UÇAR. *On the block triangular form of symmetric matrices*, in "SIAM Review", 2010, vol. 52, n^{o} 3, p. 455–470.

[28] J.-F. PINEAU, Y. ROBERT, F. VIVIEN. *Energy-aware scheduling of bag-of-tasks applications on master-worker platforms*, in "Concurrency and Computation: Practice and Experience", 2010, To appear.

[29] M. STILLWELL, D. SCHANZENBACH, F. VIVIEN, H. CASANOVA. *Resource allocation algorithms for virtualized service hosting platforms*, in "Journal of Parallel and Distributed Computing",

2010, vol. 70, n^o 9, p. 962-974, http://www.sciencedirect.com/science/article/B6WKJ-504STDX-1/2/e86c5de5a775410a9edf62309fdc55a7.

[30] B. UÇAR, Ü. V. ÇATALYÜREK, C. AYKANAT. *A Matrix Partitioning Interface to PaToH in MATLAB*, in "Parallel Computing", June 2010, vol. 36, n^o 5–6, p. 254–272.

[31] Ü. V. ÇATALYÜREK, C. AYKANAT, B. UÇAR. *On two-dimensional sparse matrix partitioning: Models, methods, and a recipe*, in "SIAM Journal on Scientific Computing", 2010, vol. 32, n^o 2, p. 656–683.

International Peer-Reviewed Conference/Proceedings

[32] V. ACRETOAIE, E. CARON, C. TEDESCHI. *A Practical Study of Self-Stabilization for Prefix-Tree Based Overlay Networks*, in "MOSPAS 2010. Workshop on MOdeling and Simulation of Peer-to-Peer Architectures and Systems. As part of The International Conference on High Performance Computing and Simulation (HPCS 2010)", Caen, France, IEEE, June 28-July 2 2010, p. 341-347.

[33] K. AGRAWAL, A. BENOIT, L. MAGNAN, Y. ROBERT. *Scheduling algorithms for linear workflow optimization*, in "IPDPS'2010, the 24th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2010.

[34] N. BARD, R. BOLZE, E. CARON, F. DESPREZ, M. HEYMANN, A. FRIEDRICH, L. MOULINIER, N.-H. NGUYEN, O. POCH, T. TOURSEL. *Décrypthon Grid - Grid Resources Dedicated to Neuromuscular Disorders*, in "The 8th HealthGrid conference", Paris, France, June 2010, To appear.

[35] A. BENOIT, H. L. BOUZIANE, Y. ROBERT. *General vs. interval mappings for streaming applications*, in "ICPADS'2010, the 16th International Conference on Parallel and Distributed Systems", IEEE Computer Society Press, 2010.

[36] A. BENOIT, H. L. BOUZIANE, Y. ROBERT. *Optimizing the reliability of pipelined applications under throughput constraints*, in "ISPDC'2010, the 9th International Symposium on Parallel and Distributed Computing", IEEE Computer Society Press, 2010.

[37] A. BENOIT, H. CASANOVA, V. REHN-SONIGO, Y. ROBERT. *Resource allocation for multiple concurrent in-network stream-processing applications*, in "HeteroPar'2009: Seventh Int. Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Platforms, jointly held with Euro-Par 2009", LNCS 6043, Springer Verlag, 2010, p. 81-90, Received the Best Paper Award..

[38] A. BENOIT, A. DOBRILA, J.-M. NICOD, L. PHILIPPE. *Throughput optimization for micro-factories subject to task and machine failures*, in "APDCM'2010, 12th Workshop on Advances on Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2010, To appear.

[39] A. BENOIT, F. DUFOSSÉ, A. GIRAULT, Y. ROBERT. *Computing the throughput of replicated workflows on heterogeneous platforms*, in "ICPP'2010, the 39th International Conference on Parallel Processing", IEEE Computer Society Press, 2010.

[40] A. BENOIT, B. GAUJAL, F. DUFOSSÉ, M. GALLET, Y. ROBERT. *Computing the throughput of probabilistic and replicated streaming applications*, in "22nd ACM Symposium on Parallelism in Algorithms and Architectures SPAA 2010", ACM Press, 2010.

[41] A. BENOIT, L. MARCHAL, O. SINNEN, Y. ROBERT. *Mapping pipelined applications with replication to increase throughput and reliability*, in "SBAC-PAD'2010, the 22nd International Symposium on Parallel and Distributed Computing", IEEE Computer Society Press, 2010.

[42] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *Performance and energy optimization of concurrent pipelined applications*, in "IPDPS'2010, the 24th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2010.

[43] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *Sharing resources for performance and energy optimization of concurrent streaming applications*, in "SBAC-PAD'2010, the 22nd International Symposium on Parallel and Distributed Computing", IEEE Computer Society Press, 2010.

[44] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *Power-aware replica placement and update strategies in tree networks*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011, To appear.

[45] A. BENOIT, Y. ROBERT, A. ROSENBERG, F. VIVIEN. *Static worksharing strategies for heterogeneous computers with unrecoverable failures*, in "HeteroPar'2009: Seventh Int. Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Platforms, jointly published with Euro-Par 2009", LNCS 6043, Springer Verlag, 2010, p. 71-80.

[46] Y. CANIOU, G. CHARRIER, F. DESPREZ. *Analysis of Tasks Reallocation in a Dedicated Grid Environment*, in "IEEE International Conference on Cluster Computing 2010 (Cluster 2010)", Heraklion, Crete, Greece, September 20-24 2010, p. 284–291.

[47] Y. CANIOU, G. CHARRIER, F. DESPREZ. *Evaluation of Reallocation Heuristics for Moldable Tasks in Computational Grids*, in "9th Australasian Symposium on Parallel and Distributed Computing (AusPDC 2011)", Perth, Australia, January 2011, 10p.

[48] F. CAPPELLO, H. CASANOVA, Y. ROBERT. *Checkpointing vs. migration for post-petascale supercomputers*, in "ICPP'2010, the 39th International Conference on Parallel Processing", IEEE Computer Society Press, 2010.

[49] E. CARON, B. DEPARDON, F. DESPREZ. *Deployment of a hierarchical middleware*, in "Euro-Par 2010", Ischia - Naples, Italy, LNCS, August 31 to September 3 2010, vol. 6271 Part I, p. 343-354.

[50] E. CARON, B. DEPARDON, F. DESPREZ. *Modelization and Performance Evaluation of the DIET Middleware*, in "ICPP 2010, 39th International Conference on Parallel Processing", San Diego, CA, September 13-16 2010, p. 375-384.

[51] E. CARON, F. DESPREZ, T. GLATARD, M. KETAN, J. MONTAGNAT, D. REIMERT. *Workflow-based comparison of Distributed Computing Infrastructures*, in "Workflows in Support of Large-Scale Science (WORKS10)", New Orleans, IEEE, November 14 2010, To appear.

[52] E. CARON, F. DESPREZ, A. MURESAN. *Forecasting for Grid and Cloud Computing On-Demand Resources Based on Pattern Matching*, in "IEEE CloudCom 2010", Indianapolis, Indiana, USA, 456-463, IEEE, Nov 2010.

[53] H. CASANOVA, F. DUFOSSÉ, Y. ROBERT, F. VIVIEN. *Scheduling parallel iterative applications on volatile resources*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011, To appear.

[54] H. CASANOVA, M. GALLET, F. VIVIEN. *Non-clairvoyant Scheduling of Multiple Bag-of-tasks Applications*, in "Proceedings of the 16th International Euro-Par Conference (Euro-Par'10)", LNCS, Springer, September 2010, vol. 6271, p. 168-179.

[55] J. CELAYA, L. MARCHAL. *A Fair Decentralized Scheduler for Bag-of-tasks Applications on Desktop Grids*, in "Proceedings of CCGrid 2010: 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing", 2010, p. 538-541.

[56] P. CODOGNET, Y. CANIOU, D. DIAZ, S. ABREU. *Experiments in Parallel Constraint-based Local Search*, in "ACM 26th Symposium On Applied Computing (SAC 2011)", TaiChung, Taiwan, March 2011, 2.

[57] F. DESPREZ, F. SUTER. *A Bi-Criteria Algorithm for Scheduling Parallel Task Graphs on Clusters*, in "The 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)", Melbourne, Aus, IEEE/ACM, May 2010.

[58] G. FEDAK, J.-P. GELAS, T. HÉRAULT, V. INIESTA, D. KONDO, L. LEFÈVRE, P. MALÉCOT, L. NUSSBAUM, A. REZMERITA, O. RICHARD. *DSL-Lab: a Platform to Experiment on Domestic Broadband Internet*, in "Proceedings of the The 9th IEEE International Symposium on Parallel and Distributed Computing (ISPDC'10)", Istanbul, Turquey, July 2010, p. 141–148.

[59] M. GALLET, M. JACQUELIN, L. MARCHAL. *Scheduling complex streaming applications on the Cell processor*, in "Proceedings of MTAAP 2010: Workshop on Multithreaded Architectures and Applications", 2010.

[60] M. GALLET, N. YIGITBASI, B. JAVADI, D. KONDO, A. IOSUP, D. EPEMA. *A Model for Space-Correlated Failures in Large-Scale Distributed Systems*, in "Proceedings of the 16th International Euro-Par Conference (Euro-Par 2010)", LNCS, Springer, September 2010, vol. 6271, p. 168-179.

[61] H. HE, G. FEDAK, P. KACSUK, Z. FARKAS, Z. BALATON, O. LODYGENSKY, E. URBAH, G. CAILLAT, F. ARAUJO. *Extending the EGEE Grid with XtremWeb-HEP Desktop Grids*, in "Proceedings of CCGRID'10, 4th Workshop on Desktop Grids and Volunteer Computing Systems (PCGrid 2010)", Melbourne, Australia, May 2010, p. 685–690.

[62] M. JACQUELIN, L. MARCHAL, Y. ROBERT, B. UÇAR. *On optimal tree traversals for sparse matrix factorization*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011, To appear.

[63] A. C. MAROSI, P. KACSUK, G. FEDAK, O. LODYGENSKY. *Sandboxing for Desktop Grids using virtualization*, in "Proceedings of the 18th Euromicro International Conference on Parallel, Distributed and Network-Based Computing PDP 2010", Pisa, Italy, February 2010, p. 559–566.

[64] M. MOCA, G. C. SILAGHI, G. FEDAK. *Distributed Results Checking for MapReduce on Volunteer Computing*, in "Proceedings of IPDPS'2011, 4th Workshop on Desktop Grids and Volunteer Computing Systems (PCGrid 2010)", Anchorage, Alaska, May 2011, to appear.

[65] A. RIBES, C. PÉREZ, V. PICHON. *On the Design of Adaptive Mesh Refinement Applications based on Software Components*, in "2010 Workshop on Component-Based High Performance Computing (CBHPC 2010)", Brussels, Belgium, October 2010.

[66] M. STILLWELL, F. VIVIEN, H. CASANOVA. *Dynamic Fractional Resource Scheduling for HPC Workloads*, in "24th IEEE International Parallel and Distributed Processing Symposium (IPDPS)", IEEE CS Press, 2010.

[67] B. TANG, M. MOCA, S. CHEVALIER, H. HE, G. FEDAK. *Towards MapReduce for Desktop Grid Computing*, in "Fifth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'10)", Fukuoka, Japan, IEEE, November 2010, p. 193–200.

[68] B. UÇAR, Ü. V. ÇATALYÜREK. *On scalability of hypergraph models for sparse matrix partitioning*, in "Proceedings of PDP 2010: 18th Euromicro International Conference on Parallel, Distributed and Network-Based Computing", M. DANELUTTO, J. BOURGEOIS, T. GROSS (editors), IEEE Computer Society, Conference Publishing Services, 2010, p. 593–600.

[69] N. YIGITBASI, M. GALLET, D. KONDO, A. IOSUP, D. EPEMA. *Analysis and Modeling of Time-Correlated Failures in Large-Scale Distributed Systems*, in "Proceedings of the 11th ACM/IEEE International Conference on Grid Computing (GRID 2010)", ACM / IEEE Computer Society Press, October 2010.

Scientific Books (or Scientific Book chapters)

[70] P. R. AMESTOY, A. BUTTARI, I. S. DUFF, A. GUERMOUCHE, J.-Y. L'EXCELLENT, B. UÇAR. *MUMPS*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011, To appear.

[71] P. R. AMESTOY, A. BUTTARI, I. S. DUFF, A. GUERMOUCHE, J.-Y. L'EXCELLENT, B. UÇAR. *The Multifrontal Method*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011, To appear.

[72] J. BIGOT, C. PÉREZ. *On High Performance Composition Opertators in Component Models*, in "High Performance Scientific Computing with special emphasis on Current Capabilities and Future Perspectives", Advances in Parallel Computing, IOS Press, 2011, to appear.

[73] Y. ROBERT. *Task graph scheduling*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011, To appear.

[74] Ü. V. ÇATALYÜREK, B. UÇAR, C. AYKANAT. *Hypergraph Partitioning*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011, To appear.

Books or Proceedings Editing

[75] M. ALEXANDER, F. VIVIEN (editors). *Euro-Par 2010 - Parallel Processing Workshops, VHPC, HeteroPar, HPPC, GECON, HiBB, CoreGrid, UCHPC, HPCF, XtreemOS, PROPER, CCPI, Ischia, Italy, August 30-31, 2010, Revised Selected Papers*, Lecture Notes in Computer Science, Springer, 2010, To appear.

[76] Y. ROBERT, L. SOUSA, D. TRYSTRAM (editors). *Special issue on ISPDC'2009 and HeteroPar'2009*, Parallel Computing, 2011, To appear.

Other Publications

[77] Y. CANIOU. *Standardisation de la gestion de données dans le GridRPC*, November 2010, Poster at JFR'10, Journée Francophone de la Recherche.

[78] O. GATSENKO, O. BASKOVA, G. FEDAK, O. LODYGENSKY, Y. GORDIENKO. *Kinetics of Defect Aggregation in Materials Science Simulated in Desktop Grid Computing Environment Installed in Ordinary Material Science Lab*, in "Proceedings of 3rd Grid Experience workshop - Desktop Grid Applications for eScience and eBusiness", Alemere, Nederlands, EnterTheGrid, Alemere, Nederlands, March 2010.

[79] O. GATSENKO, O. BASKOVA, G. FEDAK, O. LODYGENSKY, Y. GORDIENKO. *Porting Multiparametric MATLAB Application for Image and Video Processing to Desktop Grid for High-Performance Distributed Computing*, in "Proceedings of 3rd Grid Experience workshop - Desktop Grid Applications for eScience and eBusiness", Alemere, Nederlands, EnterTheGrid, Alemere, Nederlands, March 2010.

[80] O. GATSENKO, O. BASKOVA, O. LODYGENSKY, G. FEDAK, Y. GORDIENKO. *Statistical Properties of Deformed Single-Crystal Surface under Real- Time Video Monitoring and Processing in the Desktop Grid Distributed Computing Environment*, in "Proceedings of the Sixth International Conference on Materials Struncture and Micromechanics of Fracture (MSMF6)", Brno, Czech Republic, June 2010.

[81] Y. SUZUKI, N. KUSHIDA, T. TATEKAWA, N. TESHIMA, Y. CANIOU, R. GUIVARCH, M. DAYDÉ, P. RAMET. *Development of an International Matrix-Solver Prediction System on a French-Japanese International Grid Computing Environment*, in "Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2010 (SNA + MC2010)", Hitotsubashi Memorial Hall, Tokyo, Japan, October 17-21 2010.

[82] B. UÇAR, Ü. V. ÇATALYÜREK. *Partitioning regular meshes for minimizing the total communication volume*, February 2010, Presentation at SIAM Conference on Parallel Processing for Scientific Computing (PP10), Seattle, Washington, USA.

References in notes

[83] R. BUYYA (editor). *High Performance Cluster Computing*, Prentice Hall, 1999, vol. 2: Programming and Applications, ISBN 0-13-013784-7.

[84] P. CHRÉTIENNE, E. G. COFFMAN JR., J. K. LENSTRA, Z. LIU (editors). *Scheduling Theory and its Applications*, John Wiley and Sons, 1995.

[85] I. FOSTER, C. KESSELMAN (editors). *The Grid: Blueprint for a New Computing Infrastructure*, Morgan-Kaufmann, 1998.

[86] P. R. AMESTOY, I. S. DUFF, J.-Y. L'EXCELLENT. *Multifrontal Parallel Distributed Symmetric and Unsymmetric Solvers*, in "Comput. Methods Appl. Mech. Eng.", 2000, vol. 184, p. 501–520.

[87] M. BAKER. *Cluster Computing White Paper*, 2000.

[88] E. CARON, A. CHIS, F. DESPREZ, A. SU. *Plug-in Scheduler Design for a Distributed Grid Environment*, in "4th International Workshop on Middleware for Grid Computing - MGC 2006", Melbourne, Australia, November 27th 2006, In conjunction with ACM/IFIP/USENIX 7th International Middleware Conference 2006.

[89] P. CODOGNET, D. DIAZ. *Yet Another Local Search Method for Constraint Solving*, in "proceedings of SAGA'01", Springer Verlag, 2001, p. 73-90.

[90] P. CODOGNET, D. DIAZ. *An Efficient Library for Solving CSP with Local Search*, in "MIC'03, 5th International Conference on Metaheuristics", T. IBARAKI (editor), 2003.

[91] I. S. DUFF, J. K. REID. *The Multifrontal Solution of Indefinite Sparse Symmetric Linear Systems*, in "ACM Transactions on Mathematical Software", 1983, vol. 9, p. 302-325.

[92] I. S. DUFF, J. K. REID. *The Multifrontal Solution of Unsymmetric Sets of Linear Systems*, in "SIAM Journal on Scientific and Statistical Computing", 1984, vol. 5, p. 633-641.

[93] H. EL-REWINI, H. H. ALI, T. G. LEWIS. *Task Scheduling in Multiprocessing Systems*, in "Computer", 1995, vol. 28, n^{o} 12, p. 27–37.

[94] G. FEDAK, C. GERMAIN, V. NÉRI, F. CAPPELLO. *XtremWeb : A Generic Global Computing System*, in "CCGRID2001, workshop on Global Computing on Personal Devices", IEEE Press, May 2001.

[95] J. W. H. LIU. *The Role of Elimination Trees in Sparse Factorization*, in "SIAM Journal on Matrix Analysis and Applications", 1990, vol. 11, p. 134–172.

[96] M. G. NORMAN, P. THANISCH. *Models of Machines and Computation for Mapping in Multicomputers*, in "ACM Computing Surveys", 1993, vol. 25, n^{o} 3, p. 103–117.

[97] B. A. SHIRAZI, A. R. HURSON, K. M. KAVI. *Scheduling and Load Balancing in Parallel and Distributed Systems*, IEEE Computer Science Press, 1995.