INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team oasis

Objets Actifs, Sémantique, Internet et Sécurité

Sophia Antipolis

THEME COM

Activity Report

2006

Table of contents

1. Team

OASIS *is an INRIA joint project with CNRS I3S and University of Nice Sophia Antipolis.*

Team Leader
Denis CAROMEL [UNSA, HdR]

Team Vice-Leader
Eric MADELAINE [INRIA]

Administrative Assistant
Claire SENICA [INRIA]

Research Scientist
Françoise BAUDE [UNSA, HdR]
Ludovic HENRIO [CNRS]
Fabrice HUET [UNSA]
Bernard SERPETTE [INRIA]

Technical staff
Virginie (LEGRAND) CONTES [PISE, from 01/02/05]
Cédric DALMASSO [GridComp, from 06/11/06]
Christian DELBE [GridCOMP, from 01/12/06]
Yu FENG [ETSI, from 01/10/06]
Stéphane MARIANI [CoreGrid, to 31/07/06]
Clément MATHIEU [INRIA, from 01/09/06]
Marc OZONNE [S4ALL, to 30/06/06]
Hejer REJEB [Fiacre, from 01/02/06]
Igor ROSENBERG [GridCoord, to 30/06/06]
Fabien VIALE [S4ALL, from 16/10/06]

Ph.D. Student
Javier BUSTOS [Co-tutelle UNSA/Univ. Chili]
Antonio CANSADO [UNSA]
Guillaume CHAZARAIN [UNSA]
Christian DELBE [UNSA, to 30/11/06]
Alexandre DI COSTANZO [UNSA]
Viet Dung DOAN [UNSA, from 01/10/06]
Mario LEYTON [UNSA]
Matthieu MOREL [UNSA]
Paul NOUAMEMKO [UNSA, from 01/10/06]
Marcela RIVERA [Conicyt Chili, from 01/12/06]

Post-doctoral Fellow
Ian STOCKES-REED [INRIA, from 01/12/06]
Nikolaus PARLAVANTZAS [ERCIM, to 28/02/06]

Visiting scientist
Blanca ABRAHAM [Univ. Mérida, from 27/06/06 to 15/07/06]
José AGUILAR [Univ. Mérida, from 09/07/06 to 15/07/06]
Tomas BARROS [Université du Chili, from 23/01/06 to 10/0206]
Arnaud CONTES [Université de Cardiff, from 04/09/06 to 30/11/06]
José PIQUER [Université du Chili, from 27/11/06 to 30/11/06]
Florian KAMMULLER [Université Technique de Berlin, from 11/09/06 to 29/09/06]

Student intern
Solange AHUMADA [Conicyt Chili, from 6/12/06]
Wasseim AL-ZOUABI [ERCIM, from 01/10/06]

Anissa ALLAL [UNSA, from 16/01/06 to 30/09/06]
Brian AMEDRO [UNSA, from 15/05/06 to 30/09/06]
Michèle BARRE [UNSA, from 15/05/06 to 30/09/06]
Vladimir BODNARTCHOUK [UNSA, from 15/05/06 to 30/09/06]
Sébastien BOUKHALFA [UNSA, from 15/05/06 to 30/09/06]
Géraldine CABANNES [UNSA, from 16/01/06 to 30/09/06]
Vincent CAVE [UNSA, from 15/03/06 to 31/07/06]
Viet Dung DOAN [UNSA, from 16/01/06 to 29/09/06]
Christian JARJOUHI [ESI Beyrouth, from 25/06/06 to 09/09/06]
Jean-Michael LEGAIT [UNSA, from 15/05/06 to 30/09/06]
Arnaud MANGIN [Univ. Pierre et Marie CURIE, from 03/04/06 to 30/09/06]
Elton MATHIAS [Univ. Santa Maria Brésil, from 16/08/06]
Clément MATHIEU [EPU de Nice-Sophia Antipolis, from 01/03/06 to 30/09/06]
Paul NAOUMENKO [EPU de Nice-Sophia Antipolis, from 20/03/06 to 29/09/06]
Christelle NDEYE [UNSA, from 16/01/06 to 30/09/06]
Samir OUIFI [UNSA, to 10/03/06]
Mohamed OUL AHMED [UNSA, to 10/03/06]
Michèle REYNIER [UNSA, from 15/05/06 to 30/09/06]
Marcela RIVERA [Conicyt Chili, to 31/07/06]
Christophe ROGER [UNSA, from 15/05/06 to 30/09/06]
Cristian RUZ [Conicyt Chili, from 15/11/06]
Emil SALAGEANU [UNSA, from 15/05/06]

Other personnel

Vladimir BODNARTCHOUK [UNSA, apprentice, from 01/10/06]
Jean-Michael LEGAIT [UNSA, apprentice, from 01/10/06]

2. Overall Objectives

2.1. Overall Objectives

The team focuses its activities on distributed (Grid) computing and more specifically on the development of secure and reliable distributed systems using distributed asynchronous object systems (active objects - OA of OASIS). From this central point of focus, other research fields are considered in the project:

- Semantics (first S of OASIS): formal specification of active objects with the definition of ASP (Asynchronous Sequential Processes) and the study of preconditions where this calculus becomes deterministic.
- Internet (I of OASIS): Grid computing with distributed and hierarchical components.
- Security (last S of OASIS): analysis and verification of programs written in such asynchronous models.

With these objectives, our approach is:

- theoretical: we study and define models and object-oriented languages (semantical definitions, equivalences, analysis);
- applied: we start from concrete and current problems, for which we propose technical solutions;
- pragmatic: we validate the models and solutions with full-scale experiments.

Internet clearly changed the sense of notions like mobility and security. We believe that we have the skills to be significantly fruitful in this major application domain; more specifically, we aim at producing interesting results for embedded applications for mobile users, Grid computing, peer-to-peer intranet, electronic trade and collaborative applications.

3. Scientific Foundations

3.1. Object Distributed Computation

The paradigm of object-oriented programming, although not very recent, got a second youth with the Java language. The concept of object, despite its universal denotation, is clearly not properly defined and implemented: notions like inheritance, sub-typing or overloading have as many definitions as there are underlying languages. The introduction of concurrency into objects also increases the complexity. It appeared that standard Java constituents such as RMI (Remote Method Invocation) do not help building, in a transparent way, sequential, multi-threaded, or distributed applications. Indeed, allowing, as RMI, the execution of the same application on a shared-memory multiprocessors architecture as well as on a network of workstations (intranet, Internet), or on any hierarchical combination of both is not sufficient for providing easy programming to the developer.

The question is thus: how to ease the construction, deployment and evolution of distributed applications ?

We have developed competencies in both theoretical and pragmatic fields, such as automatic distribution of activities using static analysis, and the building of a Java library for parallel, distributed, and concurrent computing.

3.2. Formal models for distributed objects

In distributed object systems where concurrent processes co-exist, being able to prove properties such as confluence and determinism is a step towards establishing correctness of those systems.

For this, calculi for distributed objects, and in particular, the ASP calculus we designed a few years ago, constitute one of our major scientific foundations. ASP is a calculus for distributed objects interacting using asynchronous method calls with generalized futures, i.e., wait-by-necessity – a must in large-scale systems, providing both high structuring and low coupling, and thus scalability. Our work on ASP provides very generic results on expressiveness and determinism, and the potential of this approach has been further demonstrated by its capacity to cope with advanced issues such as mobility, groups, and components [8].

ASP is thus a means to provide confluence and determinism properties in concurrent processes. Such results should allow one to program parallel and distributed applications that behave in a deterministic manner, even if they are distributed over local or wide area networks.

The ASP calculus is a model for the ProActive library. An extension of ASP is being designed to model distributed asynchronous components.

3.3. Static Analysis and Verification

Programming distributed objects, even with the help of high-level libraries, also increases the difficulty of analyzing their behaviors, and ensuring safety, security, or liveness properties of these applications.

More generally, the formal verification of software systems is an issue more recent and more difficult than verification of hardware and circuits. This is true both at a theoretical and at a pragmatic level, from the definition of adequate models representing programs, the mastering of state complexity through abstraction techniques or through new algorithmic approaches, to the design of software tools that will hide to the final user the complexity of the underlying theory.

In the context of distributed component systems, we get not only better descriptions of the structure of the system, that make the analysis more tractable, but also new interesting problems, such as the interplay between the functional definition of a component and its possible runtime transformations, expressed by the various management controllers of the component system.

Our approach is to use techniques of static analysis and abstract interpretation to extract models from the code of distributed applications [3]. We will use then generic tools for checking properties of this model. We concentrate on behavioural properties, expressed in terms of temporal logics (safety, liveness), of security, of adequacy of an implementation to its specification and of correct composition of software components.

4. Application Domains

4.1. Electronic Business

Keywords: *formal methods*, *mobility*, *program analysis*, *proofs*, *security*, *telecommunications*.

By electronic business, we mean distributed applications over the Internet that require safety and security otherwise they would not exist at all, due to highest risks (confidentiality, privacy, integrity, authentication and availability should be guaranteed).

We give examples of such applications:

- Secure electronic commerce: programming such applications distributed over networks may uncover very complex behaviors, that may lead to deadlocks, starvation, and many other kinds of reachability or liveness problems. It is then necessary to propose methods for specifying the application behavior (requirements), and tools to check the implementation against those requirements. On the other hand, protection of communications and data is a requirement for the development of commercial applications. These security requirements have to be expressed in a security policy agreed by all partners, including customers.
- Secure collaborative applications: a multi-site enterprise may want to use Internet for the communication between different services and the collaborative building of a particular task, leading to specific problems of election, synchronization, load balancing, etc.
- Mobility for enterprise applications: a mobile worker should be able to run enterprise applications from anywhere, using heterogeneous network, and any device (desktop, laptop, PDA, board computer) in a transparent and a secure manner.

4.2. Grid Computing

Keywords: *Grid*, *Telecommunications*, *distribution*, *fault tolerance*, *group communication*, *mobile object systems*, *peer-to-peer*, *security*, *synchronization*.

As distributed systems are becoming ubiquitous, Grid computing is emerging as one of the major challenge for computer science: seamless access and use of large-scale computing resources, world-wide. The word "Grid" is chosen by analogy with the electric power grid, which provides pervasive access to power and has had a dramatic impact on human capabilities and society. It is believed that by providing pervasive, dependable, consistent and inexpensive access to advanced computational capabilities, computational grids will have a similar transforming effect, allowing new classes of applications to emerge.

Another challenge is to use, for a specific computation, unused CPU cycles of desktop computers in a Local Area Network. This is intranet Computational Peer-To-Peer.

There is a need for models and infrastructures for grid and peer-to-peer computing, and we promote a programming model based on communicating and mobile objects and components.

Another related domain of application is to use mobile objects for system and network management, for instance in the setting of OSGi (www.osgi.org) services and platforms (e.g. in industrial or home automation, vehicles, smart phones, etc).

5. Software

5.1. ProActive

Participants: F. Baude, D. Caromel, G. Chazarain, A. di Costanzo, C. Delbé, F. Huet, V. Legrand, M. Leyton, S. Mariani, M. Morel, M. Ozonne, I. Rosenberg.

ProActive is a Java library (Source code under LGPL license) for parallel, distributed, and concurrent computing, also featuring mobility and security in a uniform framework. With a reduced set of simple primitives, ProActive provides a comprehensive API allowing to simplify the programming of applications that are distributed on a Local Area Network (LAN), on cluster of workstations, or on Internet Grids.

The library is based on an Active Object pattern that is a uniform way to encapsulate:

- a remotely accessible object,
- a thread as an asynchronous activity,
- an actor with its own script,
- a server of incoming requests,
- a mobile and potentially secure agent.

and has an architecture to inter-operate with (de facto) standards such as:

- Web Service exportation,
- HTTP transport,
- ssh, rsh, RMI/ssh tunneling,
- Globus: GT2, GT3, GT4, gsi, gLite, Unicore, ARC (NorduGrid)
- LSF, PBS, Sun Grid Engine, OAS.

ProActive is only made of standard Java classes, and requires **no changes to the Java Virtual Machine**, no preprocessing or compiler modification; programmers write standard Java code. Based on a simple Meta-Object Protocol, the library is itself extensible, making the system open for adaptations and optimizations. ProActive currently uses the RMI Java standard library as default portable transport layer, but others such as Ibis or HTTP can be used instead, in an adaptive way.

ProActive is particularly well-adapted for the development of applications distributed over the Internet, thanks to reuse of sequential code, through polymorphism, automatic future-based synchronizations, migration of activities from one virtual machine to another. The underlying programming model is thus innovative compared to, for instance, the well established MPI programming model.

Many features have been incorporated into ProActive in order to cope with the requirements of the Grid such as

- The deployment infrastructure that supports almost all Grid/cluster protocols: LSF, PBS, SGE, ssh, Globus, Unicore ...;
- The communication layer that can rely on RMI or HTTP or IBIS, or SOAP or RMI/ssh. This last protocol allows one to cross firewalls in many cases;
- The component framework which implements the ObjectWeb Fractal hierarchical component model is now mature, and is being extended with collective interfaces for targeting (parallel) grid components;
- The graphical user interface IC2D offers many other views of an application, for instance the Job monitor view, that allows better control and monitoring;
- The ability to exploit the migration capability of active objects, in network and system management;
- A computational P2P infrastructure;
- Object Oriented SPMD programming model with its API;
- Distributed and Non-Functional Exceptions handling;
- Fault-Tolerance and Checkpointing mechanisms;
- File Transfer capabilities over the Grid.

We have demonstrated on a set of applications the advantages of the ProActive library, and among other we are particularly proud of the following results, showing that portable and transparent Java code can compete with specific optimized approaches:

- NQueen challenge, where we equaled the world record n=24 (227 514 171 973 736 solutions) in 17 days based on ProActive's P2P infrastructure (300 machines).
- NQueen challenge, where we get the world record n=25 (2 207 893 435 808 352 solutions) in 6 month based on ProActive's P2P infrastructure using free cycles of 260 PCs.

Still based on ProActive, we organized in October 2006 the third Grid Plugtests in collaboration with ETSI, where we received 240 participants from many different countries.

At last ProActive tutorials were given in several occasions (Plugtests, CoreGRID school, ObjectWeb conference ...) and we expect to give many others for the next year.

ProActive is a project of the ObjectWeb Consortium. ObjectWeb is an international consortium fostering the development of open-source middleware for cutting-edge applications: EAI, e-business, clustering, grid computing, managed services and more. For more information, refer to [10] and to the web pages http://www.objectweb.org and http://www.inria.fr/oasis/proactive which list a lot of white papers.

In the RNRT PISE project, we designed and implemented a remote management tool for OSGi, based on ProActive and on the JMX (Java Management eXtension) management standard [41], [22]. As it relies on ProActive typed groups, it features scalability in the remote management of a possibly very large set of OSGi gateways. The remote management includes the provisioning of applications formed by OSGi bundles: our tool is interfaced with an external one dedicated to compute a global deployment plan given the targetted set of gateways and for each, the set of bundles already installed (developed by our partner LSR-IMAG, in this project); the execution of the deployment plan is run in parallel from the management tool. Conducted benchmarks on Grid'5000 proved both the effectiveness and efficiency of the remote launching of a possibly large and hierarchically organized set of OSGi gateways, the provisioning of bundle-based applications, and the related monitoring operations. The ProActive connectors for remote JMX-based operations and an OSGi compliant version of the ProActive library are part of the public ProActive distribution. Our perspective is of course to demonstrate their reusability in other contexts. This is notably what we are pursuing within the ITEA S4ALL project.

In the S4ALL project, our objective is to go further in the "bundlelisation" of the ProActive environment: besides the capability to host a ProActive runtime on an OSGi gateway and run classic distributed active-object oriented applications, the focus is to provision, run and monitor component-oriented ProActive/GCM applications.

5.2. Vercors platform

Participants: A. Cansado, H. Rejeb, E. Madelaine.

Vercors is a component verification platform which provides means to analyse the behaviour properties of applications built from distributed components. It comprises several tools for assisting the whole process of verification. Rather than creating a new model-checker, we implement our model-generation methods in a way that will be efficiently integrated with existing state-of-art tools for checking component specifications. The platform was presented in [20].

The construction of the model begins by giving the Architecture Description Language (ADL) file to `ADL2N`, which is a Java tool we developed for the ADL analysis of Fractal components. `ADL2N` obtains the possible actions on the component's interfaces by introspection of the Java interfaces provided as the signature of the component's interfaces in the ADL file. `ADL2N` searches the classes of such Java interfaces in its classpath, and creates a parameterized model of such.

The user of ADL2N will use the tool GUI to specify at the same time the methods that will be visible, the parameters that are significant, and the finite instantiations of those parameters. ADL2N then creates two files: a pNet in hierarchical Parameterized FC2 format expressing the synchronisations between components and an instantiation file defining the domain abstractions.

The functional behaviour of primitive components is specified in a Labelled Transition System language, either given by the user, or found by static analysis of Java source code. Hejer Rejeb is developping a prototype software for this matter which analyses ProActive code and creates a pNet model encoding the behaviour of the distributed component.

Finally, Emil Salageanu developped an extension to the Turtle tool [1] tomodel distributed components in UML 2.0 diagrams [51], more suitable to ProActive designers.

6. New Results

6.1. Programming the Grid

6.1.1. Component-oriented Grid Programming

Participants: F. Baude, D. Caromel, L. Henrio, S. Mariani, M. Morel.

Here, our objective is to simplify the design, programming and evolution (adaptation) of distributed Grid applications. In particular, we define parallel and hierarchical distributed components [15] starting from the Fractal component model developed by INRIA and France-Telecom. We are involved in the design of a Grid Component Model (GCM) [49], [47], that should be one of the major result of the CoreGrid European Network of Excellence. The GCM is intended to become a standard for Grid components, and most of our research on component models can be related to it. The GCM is a component model that is an extension of the Fractal model. On the practical side, ProActive/GCM is a prototype implementation of the GCM above the ProActive library; not all the GCM features are implemented yet in ProActive.

First, our research has focused on the design of collective actions that would allow for an easier building of Grid parallel component-based applications. Our proposal in this research topic is on multi-way communications between components through the specification of collective interfaces. Collective interfaces are specified as a *collective* cardinality in the type of the component interfaces, and their behavior is customizable through a dedicated controller. Multicast and gathercast are the two kinds of collective interfaces of this proposal. We also experimented such multicast and gathercast interfaces as an alternative (and still efficient) way to perform OO-SPMD (Object-oriented Single Program, Multiple Data) programming. OO-SPMD is in fact an extension of the typed group communication mechanism, previously designed and added to the ProActive programming model.

In the context of the DiscoGRID ANR funded project, we have started to use those collective actions capabilities in order to design a MPI-like hierarchical SPMD programming model [56]. Indeed, when programming a Grid, it appears that, for good performance purposes, it would be useful to take into consideration the physical topology, which is in fact a hierarchy of multi-processessors, multi-clusters on multi-sites and possibly on multiple grids. This work is done in collaboration with applied mathematicians (namely the CAIMAN and SMASH teams, partners of the DiscoGRID project). Indeed, they represent a community where programmers are used to the standard SPMD message-passing based model, and that is quite reluctant to adopt another model. Nonetheless, they are ready to design and program their parallel algorithms in a way that takes the physical hierarchy into consideration. The solution we are exploring within DiscoGRID is to rely on a GCM-based support in order to organize the MPI application in a hierarchical way, but in a transparent way for programmers.

[1] from Ludovic Aprvile, LabSoc, ENST Sophia-Antipolis, http://labsoc.comelec.enst.fr/turtle/

Second, as part of the design of the GCM, we started research on the componentization of the components' membranes [55], [46]. This consists in adopting a component view of the non-functional and control aspects of a component model. This contribution should result in a powerful model for the design and adaptation of components control. We expect to apply such a model to the conception of autonomic components, thus providing a structured programming model for autonomic computation and communication. One of our objectives is to experiment such autonomic capabilities in the context of the EU funded IP BIONETS project we are involved in.

On a formal aspect, we extended ASP by building hierarchical and asynchronous distributed components [33]. ASP components are hierarchical - a composite can be built from other components, and distributed - a composite can span over several machines. We also showed how the asynchronous component model can be used to statically assert component determinism.

We have also conducted an important effort for interconnecting CCA [2] and ProActive/GCM components, this work resulted in a collaboration with the Academic Computer Centre CYFRONET AGH, and a couple of visits of Maciek Malawski in our team. In practice, we obtained very good interoperability results between MOCCA and ProActive components. These results should lead to a publication in the next months.

6.1.2. Skeleton-based parallel programming

Participants: D. Caromel, M. Leyton.

The structured parallelism approach (skeletons) takes advantage of common patterns used in parallel and distributed applications. The goal of the skeleton paradigm consists in separation of concerns between the distribution aspects, and the functional aspects of an application.

Problems that present structured patterns can benefit from libraries or programming languages that support skeletons. The goal is that some day, the skeleton libraries will be able to handle the complex attributes of Grid programming such as: heterogeneity, dynamicity, adaptability, etc.

Therefore, we continued with our research from last year (Branch-and-Bound). Our main results in 2006 correspond to the design and implementation of a structured programming framework called Calcium.

Calcium is a more general framework capable of handling multiple patterns of structured computation. Calcium has been implemented in Java, and provides skeleton programming as a Java library. The patterns currently researched and implemented in Calcium are:

- **farm** Also known as master-slave is used for task replication.
- **pipe** Is used for staged computation, where different stages of computation must be executed one after the other.
- **if** Represents dynamic conditional evaluation.
- **while** Represents iteration computation combined with conditional evaluation.
- **for**, Represents iteration computation.
- **d&c**, The divide and conquer patterns corresponds to data parallelism, where a task is subdivided into smaller problems, the problems are solved, and then conquered to achieve the results.
- **map**. Also corresponds to data parallelism, and represents a particular case of d&c, where the division and conquer is only performed once.

An important feature of the Calcium framework is that these patterns can be nested to solve more complex applications.

For achieving distributed computation the Calcium framework is built on top of the ProActive middleware. To achieve distributed computation, we have combined our research on structured programming with our previous research on ProActive middleware. Therefore, Calcium takes advantage of ProActive's deployment framework for performing resource acquisition on the Grid, and uses ProActive's active object model for communication.

[2] Common Component Architecture: http://www.cca-forum.org/

6.1.3. Distributed Garbage Collector

Participants: D. Caromel, G. Chazarain.

With the increasing use of active objects, manual termination of these objects is becoming a burden. Distributed garbage collectors are a solution to this problem. Currently existing garbage collectors have at least one of these limitations:

- need of a central server (not peer to peer)
- unbounded message size
- only acyclic garbage collection

Guillaume Chazarain designed and implemented a distributed garbage collector solving all of these limitations. It is based on a heartbeat between an active object and its children in the object references graph. The heartbeat message is O(1) in size and the locally computed reply to this message is used in the algorithm.

Acyclic garbage collection is simple, an active object knows it will receive a heartbeat message at a specified frequency from its referencers. So, if no message is received during a certain amount of time, the active object considers itself to be garbage and terminates.

Cyclic garbage collection is where the complexity is. Unlike local garbage collectors and most distributed garbage collectors, we don't attempt to prove that a cycle is unreachable. Instead we try to find a strongly connected component in the active objects graph where every active object is waiting for a request. This component is clearly unreachable garbage although we didn't use its unreachability property to find it. Once such a component is found, one of its member terminates itself. The outcome is either some easily collectable acyclic garbage, or a smaller strongly connected component where the algorithm can restart.

Strongly connected components are found by making a consensus on the last activity in the component. The last activity is maintained as a Lamport clock with the owner associated. It is incremented on a:

- State change (`Busy` → `Idle`) ⇒ it may be the last activity
- Loss of a child ⇒ it may be the parent in the spanning tree
- Loss of a parent ⇒ it may be the owner of the last activity

The last activity is propagated using the heartbeat messages from the parents to their children. The children use the reply to the heartbeat message to make a consensus with their parents about the last activity in the component. When such a consensus is made, the owner of the activity terminates itself. The reply to the heartbeat message is in fact a traversal of a spanning tree covering the strongly connected component.

The distributed garbage collector has been implemented and successfully tested on small scale configurations.

As part of this development, some insight was needed on the topologies of object references in applications. So a plugin for IC2D was made to extract the object references graph from active objects and graphically represent it.

6.2. Deployment and Usage of Grid Systems

6.2.1. Peer-to-Peer Computing Infrastructure

Participants: D. Caromel, A. di Costanzo.

Peer-to-Peer (P2P) Computing is becoming a key execution environment for widely distributed or highly intensive applications. The potential of 100,000 of nodes interconnected to execute a single application is rather appealing, especially for Grid computing. Mimicking data P2P, one could start a computation that no failure would ever be able to stop (and maybe nobody).

We designed a P2P infrastructure implemented on top of the ProActive Java library, allowing the provision of computational nodes for distributed applications. Computational nodes are indeed Java Virtual Machines (JVM), which are located on LANs and clusters. This infrastructure is totally self-organized and fully configurable. The creation and the maintenance of this network of JVMs is based on exchange of messages between peers. Each peer of the infrastructure is an active object, which has a list of nodes. These nodes are acquired by the applications that use the P2P network. According to the power of the machine, a peer can provide one or several nodes.

We show, in [31], how with this infrastructure, we have improved and adapted the deployment of a numerical Hydrodynamic Simulation application, the TELEMAC parallel System, for desktop grid.

In addition, we used this infrastructure to mix desktop machines and cluster machines in order to run large scale grid experimentation [18].

6.2.2. Technical Services

Participants: A. di Costanzo, C. Delbé, M. Leyton, M. Morel.

For effective components, non-functional aspects must be added to the application functional code. Like enterprise middleware and component platforms, in the context of Grids, services must be deployed at execution in the component containers in order to implement those aspects.

The work described in [57] proposes an architecture for defining, configuring, and deploying such Technical Services in a Grid platform. A technical service is a non-functional requirement that may be dynamically fulfilled at runtime by adapting the configuration of selected resources.

This is completed by another work from [32], which proposes a mechanism for Grid computing frameworks, for specifying environmental requirements that may be set and be optimized by deployers. Specified by designers by parameterizing deployment abstractions, the constraints can be dynamically mapped onto the infrastructure. This work is integrated in the ProActive middleware with the concept of technical services.

6.2.3. Dynamic Load Balancing

Participants: J. Bustos, D. Caromel.

In [28], [29] we develop a contribution on dynamic load balancing for distributed and parallel object-oriented applications. We specially target on peer-to-peer systems and their ability to distribute parallel computations, transfering a large amount of data (called intensive- communicating applications) among a large number of processors. We analyse the relation between active objects and processor load. Using this relation, and defining an order relation on processors power, we describe an active object balance algorithm as a dynamic load balancing algorithm, focusing on minimizing the time when active objects are waiting for the completion of remote calls. We benchmark a Jacobi parallel application with several load balancing algorithms. Finally, we study results from these experimentations in order to show that a peer-to-peer load balancing features good performance in terms of migration decisions and scalability. We also improved our model to consider object communication and synchronisation, discovering the usefulness of environment-awareness in load-balancing algorithms [26]. This work is the topics of Javier Bustos-Jimenez PhD Thesis [14].

6.2.4. File Transfer on the Grid

Participants: D. Caromel, M. Leyton.

During 2006 we continued with our research line concerning File Transfer on the Grid. We identified that the file transfer requirements and environment conditions depend on the stage of Grid usage. In this sense we have now identified that file transfer can take place at three different stages: deployment, execution and retrieval.

Each stage provides different environmental conditions and therefore different approaches must be used. These results were presented at the GADA 2006 symposium [24].

The *deployment* stage was researched last year, where we implemented a solution for integrating file transfer tools in the generic deployment model that ProActive offers for the submission of jobs, and acquisition of resources. These allowed us to introduce the concept of on-the-fly deployment: installation of the Grid middleware and performing the resource acquisition on the fly (without previous installation). It is our belief that on-the-fly deployment greatly reduces the Grid maintenance and deployment effort.

This year, our main results consist in researching the *execution* phase. During the execution phase we introduced a file transfer based on our previous research on active objects. The goal of this research is to be able to transfer files between nodes using active objects. We showed that using active objects, file transfer benefits from asynchronism present in active objects, reduces the grid configuration effort, and allows better performance when sharing files between peers, as opposed to more traditional approaches.

Finally, the file transfer during the *retrieval* phase represents a mixture between the approaches used during deployment and execution. From the deployment we took the concept of specifying the file transfer with the deployment descriptor model, but as a file transfer mechanism we used the file transfer between active objects used during the file transfer execution.

6.3. Reliability and Security

6.3.1. Reliable distributed components

Participants: A. Cansado, L. Henrio, E. Madelaine, M. Morel.

We have continued our work on the specification and verification of distributed components, both at the formal and methodological level, that we describe here, and at the practical level, that is described in the Software section 5.2.

Component programming helps the development and maintenance of complex systems. A component is a self-contained entity that interacts with its environment through well-defined interfaces. The component model we used is the GCM (Grid Component Model) defined by the CoreGrid NoE, and based on the distributed implementation of the Fractal component model.

For specification and verification activities, there is a strong need to integrate the architecture and behaviour descriptions of a component. We have proposed in [30] an extension of the Fractal ADL (Architecture Description Langage) that links the component interface descriptions with the communication actions occuring in the behaviour description. This extension has also been included in the definition of the CoreGrid GCM [49]. Based on this model, Antonio Cansado is defining a component specification language that includes both architecture and behaviour aspects.

We have refined our previous work giving the semantics of Fractal non-functional controlers (life-cycle, binding controlers), and proposed a method for generating those controlers, and a synchronisation networks connecting them to the functional part of the component, based on the information contained in the ADL. This has been presented in [20], together with its implementation in the ADL2N tool, and with first experimental results on a significantly large case-study.

Additionaly, we have started research on UML diagrams that could provide a way for non-specialist users to express the architectural and behavioural specification of components [51]. This should lead the way to a dedicated profile that will contained higher level diagram constructs, for expressing the structure and the communication mechanisms of grid applications.

6.3.2. Fault Tolerance

Participants: F. Baude, D. Caromel, C. Delbé, L. Henrio.

Providing a fault-tolerance service to applications based on communicating activities deployed in heterogeneous and volatile contexts such as Grids is mandatory. This subject is explored in the PhD work of Christian Delbé. A fault-tolerance method has been designed [16], which is totally transparent concerning the code, but adaptive with respect to the architecture of the underlying Grid onto which the application has been deployed.

Our contribution this year focused on two aspects: a theoretical model for proving correctness of the fault-tolerance protocols that recover from inconsistent global states, and an implementation allow us to benchmark and experiment around a Grid oriented protocol.

First, from the theoretical point of view, we developed a framework that allows to take into account the semantics of the system in a event-based model [42]. This framework allowed us to prove the correctness of the recovery from an inconsistent global state, under given conditions. We used this model to prove the correctness of our previously implemented protocol [16].

In the context of Grid programming, the protocol that we proposed relies on the notion of recovery group of communicating activities. Such a group encompasses activities that are deployed, for instance, on the same cluster of PCs. Inter-group communications are subject to a pessimistic message-logging (PML) fault-tolerance protocol. On the contrary, intra-group communications are subject to a Communication Induced Checkpointing (CIC) fault-tolerance method. The implementation of this protocol has been finalized, and intensively tested and evaluated during this year.

7. Contracts and Grants with Industry

7.1. ITEA project S4ALL

S4ALL (Services for All) is an ITEA project, leaded by Alcatel CIT Research and Innovation. INRIA is also member of this project, through the involvement of several of its teams or hosted teams: ObjectWeb, SARDES, JACQUARD and OASIS. The global aim of the project is to explore the technical solutions, and consolidate the existing ones that may appear suitable for building the following vision: *A world of user-centric services that are easy to create, share and use*. Our contribution in this project is mainly on the usage of the ProActive platform, that should be helpful for programming and deploying distributed services, and publish them on the service bus. Moreover, as services may be deployed everywhere, including on scarce-resource devices, we target not only standard JVMs, but also OSGi platforms.

The project is built around three sets of partners: 6 large industrial companies (Alcatel CIT Research and Innovation, Bull, Nokia, Schneider Electric, Thales and Vodafone), 3 SMEs (Capricode, mCentric, Xquark), and 6 academic partners (Fraunhofer Fokus, Helsinki Institute for Information Technology, INRIA, INT, Univ. Joseph Fourrier IMAG, Univ. Politecnica de Madrid).

Our involvment within this ITEA project will contribute to the definition of a service oriented solution supporting the composition, deployment, execution and management of OSGi applications. Those applications may be distributed ones, so we focus on the ProActive/GCM programming framework. We specifically conduct research on how to package, deploy, run and monitor GCM applications in the context of networked OSGi platforms.

This project started in July 2005, for 24 months, for a total of 129 kEuros.

7.2. RNRT PISE

Pise is a RNRT project (http://www.telecom.gouv.fr/rnrt/rnrt/projets/PISE.htm) involving Schneider Electric SA, Trialog, Universite Joseph Fourier, IMAG/LSR, INRIA and France Telecom.

The aim of this RNRT project is to define an integrated solution for the programming, deployment, management of OSGi applications, to be deployed on possibly large sets of OSGi gateways. OASIS has been specifically involved in the design and development of a remote management tool, capable of provisioning OSGi applications on hierarchically organised sets of OSGi gateways, and further manage those gateways. The use of the ProActive technology has been a key input for the scalability of remote management in the OSGi world.

This project started in September 2004, for 24 months, for a total of 170 kEuros.

7.3. Extending the Fractal model For Grid Computing - France Telecom

This is a CRE (Contrat de Recherche Externalisée), supported by France Telecom RD, starting in October 2004, for two years, for an amount of 108 kEuros.

We study how the Fractal Model for components is appropriate to the programming of Grids. We wish to define parallel and hierarchical components, collective actions that would allow for an easier building of Grid applications. We also want to experiment on large-scale applications (one thousand nodes). Altogether, this research should provide tools for programming and deploying Grid applications, including the packaging of legacy code.

In October 2005 we made the second deliverable where we described how to handle multiway communications through multiple components, through collective interfaces.

Collective interfaces are proposed as an extension of the Fractal component model, and they allow to expose collective behaviours in the definition of distributed components.

This collaboration with France Telecom has been finalized with the thesis of Matthieu Morel [15], entitled "Components for Grid Computing". It gathers the work of this collaboration into a proposal for a component mode for Grid computing. The approach relies on the Fractal component model, extended with virtualized deployment and collective interfaces. The thesis also presents the framework that was implemented, and the applications of collective interfaces, notably for SPMD-based component programming.

7.4. Soc. du Canal de Provence

Canal de Provence is a company that manages hydrolique nnetwork, and also does simulations of flooding. They are using software (i.e. Telemac, designed by EDF, and distributed by SOGREHA) in order to model and simulate large floodings, for instance due to dam breaks. The contract aimed at using ObjectWeb ProActive in order to speed up simulations that can last several days. A prototype was experimented, running on desktop machines.

This contract started in march 2006 for 4 month, and had a budget of 4 kEuros.

7.5. ETSI 2006

The general aim is to provide ETSI with a scientific and technical assistance in the framework of the Grid event entitled Grids@work: Middlewares, Users, Contest and Plugtest. In particular, this assistance was required for the scientific and technical support of the second grid PlugtestsTM and contest. After the success of the first and second Grid Plugtests and Contest organized by the ETSI PlugtestsTM service and INRIA in October 2004, and October 2005 this third edition took place from 27 November to 1 December 2006, at ETSI headquarters, Sophia-Antipolis, France. The Grid@work 2006 week consisted of the following events:

- EU project meetings: GRIDCOMP, CoreGRID Institute on Resource Management and Scheduling, CoreGRID Institute on System Architecture, CoreGRID Institute on Programming Model, and CoreGRID Institute on Grid Systems, Tools and Environments;
- 1st CoreGRID Industrial Conference
- BIGG : Bridging Global Computing with Grid – Project Consultation meeting;
- Ibis Tutorial;
- User Groups: ProActive and GCM Tutorial, and 3rd ProActive and GCM User Group;
- 3rd GRID Plugtests: N-Queens Contest and FlowShop Contest, and Free Interop Post contests;
- EGEE VO Management and the 3rd Grid Plugtests
- g-Eclipse Information meeting and BOF
- ETSI TC GRID#2 Standardization Meeting

This contract started in august 2006, for 6 months, involving 45 kEuros.

8. Other Grants and Activities

8.1. National Collaborations

8.1.1. ACI Data Grid Explorer

Data Grid Explorer is a project of the French *Action Concertée Incitative* (ACI) *Masse de données* of the Ministry of Research. This project started in 2003, for 3 years, involving 7 kEuros.

The project Data Grid Explorer aims at experimenting on large scale distributed systems on different features such as : fault tolerance, localization and performance.

Members of this project are: IMAG, LaRIA, LRI, LASSI, LORIA, LIP ENS Lyon, LIFL, LIP6, LABRI, IBCP, CEA and IRISA.

This was the last year of a 3 years project, involving a total budget of 7,2 kEuros.

8.1.2. ACI Fiacre

Fiacre stands for "FIabilité des Assemblages de Composants REpartis: Modèles et outils pour l'analyse de propriétés de sécurité et de sureté"; Fiacre is a software project of the French *Action Concertée Incitative* (ACI) *Sécurité Informatique* of the Ministry of Research. This project started in September 2004, for 3 years, involving 97 kEuros.

Gathering teams specialized in behavioural specifications of components, languages and models for distributed, mobile, and the programming of communicating application, and compositional verification, the goal of FIACRE is to design methods and tools for specification, model extraction, and verification of distributed, hierarchical, and communicating components. We would like the collaboration to result in a software prototype applicable to real applications.

Members of this project are: INRIA Oasis (coordinator) and Vasy teams, Feria/SVF, and ENST/ILR.

Our contribution for this year is on methods and tools for building parameterized models of distributed components [20] [30].

8.1.3. ANR DiscoGrid

This ANR funded project gather partners that are applied mathematicians (OMEGA and SMASH teams) and computer scientists researching in distributed and grid programming environments (OASIS, PARIS, LaBRI SoD, MOASIS).

The DiscoGRID project aims at defining a new SPMD programming model, suited to High Performance Computing on Computational Grids. Grids are hierarchical in nature (multi-CPU machines, interconnected within clusters, themselves interconnected as grids), so the incurred latency for inter processes communication can vary greatly depending on the effective location of the processes. The challenge is to define a programming model that should allow programmers to exploit this hierarchy, as easily and efficiently as possible. As the MPI SPMD message-passing model is very popular in High Performance Computing we are defining a hierarchical extension of MPI. To address Grid hierarchy and high dynamicity, this new MPI implementation will rely on GCM components, organized hierarchically as composite components in order to reflect the effective deployment of the MPI application on the grid [56].

This project started in january 2006, for 36 months, involving 110 kEuros.

8.1.4. ANR GCPMF : Grille de Calcul pour les Mathematiques Financieres

We collaborate with the OMEGA team within the ANR project entitled “GCPMF” funded by the ANR Research Program "Calcul Intensif et Grilles de Calcul 2005".

Financial applications require to solve large size computations, that are so huge that they can not be tackled by conventional PCs. A typical example addresses risk analysis evaluation as periodically run by financial institutions (like VaR – Value at Risk –, and also market risks: greeks, duration, beta, ...). Parallelism is already applied in this financial context, but its usage in Computing Grids is far from being mastered.

The aim of this ANR program is to highlight the potential of parallel techniques applied to mathematical finance computing on Grid infrastructures. The consortium that conducts this project includes ten participants from academic laboratories in computer science and mathematics, banks, and IT companies.

This year, in collaboration with Mireille Bossy from the OMEGA team, and with Stéphane Vialle and his team from Supélec, we have designed and implemented a Grid software architecture based on ProActive, and in particular on typed groups. We have applied this architecture for the pricing of European options by Monte Carlo Methods. The specification and implementation for the pricing of American options using the Longstaff-Schwartz algorithm is part of our current work, and has very different characteristics, with a lot more communication between parallel tasks. A part of this collaborative work is presented in [25].

This project started in january 2006, for 36 months, and a total budget of 44 kEuros.

8.1.5. ARC Automan

The AutoMan project aims at devising a new breed of autonomic protocols to attain high scalable and available multi-tier J2EE enterprise server replication deployed on grid systems. Grid technology will provide the infrastructure to exploit a pool of available servers to attain self-provisioning. Autonomic management will continuously tune the system to maximize performance and availability without involving human administrators . The members of this project are: INRIA Rhône-Alpes – SARDES research group, INRIA Sophia Antipolis – OASIS research group, and Technical University of Madrid – LSD research group.

The AutoMan project started in March 2006 for a length of 2 years, with a budget managed by the Sardes team.

The objective of the Oasis team in the Automan project is to explore the possibility to apply the ProActive technology to the Gridification of duplicated and self-administrated J2EE enterprise servers. The JADE tool, developed at INRIA SARDES already addresses the autonomic management of clustered J2EE application servers. Our aim is to go further by being able to recruit dynamically grid nodes in order to run replica of the server.

8.2. European Collaborations

8.2.1. SSA GridCoord

GridCoord (*Era Pilot on a Coordinated Europe-wide Initiative in Grid Research*) is a Specific Support Action (SSA) of the Sixth Framework Programme of the European Community.

Our objectives are to (1) overcome fragmentation and dispersion across EU to reinforce impact of national and Community research and (2) strengthen Europe's position on Grid Research and its exploitation.

We are particularly involved in the workpackages dedicated to enhance collaboration among researchers and users of Grids.

This project started in July 2004, for 24 months, involving 94 kEuros.

8.2.2. NOE CoreGRID

CoreGRID is an European Research Network on Foundations, Software Infrastructures and Applications for large scale distributed, GRID and Peer-to-Peer Technologies.

The CoreGRID Network of Excellence (NoE) aims at strengthening and advancing scientific and technological excellence in the area of Grid and Peer-to-Peer technologies. To achieve this objective, the Network brings together a critical mass of well-established researchers (119 permanent researchers and 165 PhD students) from forty-two institutions who have constructed an ambitious joint programme of activities. This joint programme of activity is structured around six complementary research areas that have been selected on the basis of their strategic importance, their research challenges and the recognized European expertise to develop next generation Grid middleware.

Besides the involvement of OASIS in the management and dissemination activities, the team is involved in three virtual institutes of the NoE.

- Programming Model: we are leading the Task dedicated to *Components and Hierarchical Composition*; our involvement here is to guide the design of a component model for the Grid (named GCM: Grid Component Model) at the European level. Besides, we are also involved in the Task dedicated to the study of *Basic Programming Models* for which we promote our approach of distributed and active object programming, extended with group communications and an innovative OO-SPMD approach, and in the Task dedicated to *Advanced Programming Models* in which we promote our tools for ensuring and verifying the correct behaviour of components.
- System Architecture: thanks to our experience in transparent checkpointing and recovery, we contribute to the research and integration work around Dependability in GRIDs.
- Problem Solving Environments, tools and GRID systems: thanks to our practical experience in developing the ProActive platform, we contribute in the collective study and effort to yield a generic, interoperable, portable, high-level grid toolkit, platform and environment.

Our involvement in CoreGRID led us to contribute in numerous deliverables, mainly to [49], [47], [50] and [48].

This project started in September 2004, for 48 months, involving 62 kEuros in 2006.

8.2.3. IP Bionets

The OASIS team is involved in the European project called BIONETS (BIOlogically-inspired autonomic NETworks and Services)

The motivation for BIONETS comes from emerging trends towards pervasive computing and communication environments, where myriads of networked devices with very different features will enhance our five senses, our communication and tool manipulation capabilities. The complexity of such environments will not be far from that of biological organisms, ecosystems, and socio-economic communities. Traditional communication approaches are ineffective in this context, since they fail to address several new features: a huge number of nodes including low-cost sensing/identifying devices, a wide heterogeneity in node capabilities, high node mobility, the management complexity, and the possibility of exploiting spare node resources. BIONETS aims at a novel approach able to address these challenges. BIONETS overcomes device heterogeneity and achieves scalability via an autonomic and localized peer-to-peer communication paradigm. Services in BIONETS are also autonomic, and evolve to adapt to the surrounding environment, like living organisms evolve by natural selection. Biologically-inspired concepts permeate the network and its services, blending them together, so that the network moulds itself to the services it runs, and services, in turn, become a mirror image of the social networks of users they serve.

The team is involded in work packages 3.1 (Requirement Analysis and Architecture) and 3.2 (Autonomic Service Life-Cycle and Service Ecosystems). We contributed to the deliverables: [40], [39]

The project started in 2006, for 48 months, for a total budget of 127 kEuros.

8.2.4. Strep GridComp

GridCOMP is a Strep project under leadership of ERCIM. Denis Caromel is the scientific coordinator. The european partners are university of Pisa and CNR in Pisa, university of Westminster on the academic side, and GridSystems (Spain), IBM Zurich (Switzerland), ATOS Origin (Spain). Additionaly there are 3 partners outside europe, namely from universities of Tsinghua (China), Melbourne (Australia) and University of Chili (Santiago, Chili).

GridCOMP main goal is the design and implementation of a component based framework suitable to support the development of efficient grid applications. The framework will implement the "invisible grid" concept: abstract away grid related implementation details (hardware, OS, authorization and security, load, failure, etc.) that usually require high programming efforts to be dealt with.

The project has started in july 2006, for a duration of 30 months, with a budget of 24 kEuros for 2006.

8.3. International Collaborations

8.3.1. Associated Team OSCAR

Oscar (*Objets et Sémantique, Concurrence, Aspects et Reflexion*) is a bilateral collaboration (équipe associée) between University of Chili in Santiago and the Oasis team at INRIA-Sophia. (http://www.inria.fr/oasis/oscar) We aim at gathering expertise on meta-object protocols, concurrency, transparent distributed programming, and verification of distributed systems. Contributions are related to modeling and verifying distributed software and safe concurrency. The project was started in January 2003 for a duration of 3 years, facilitating exchanges of students, visits of researchers, and organization of common workshops.

This year we were particularly involved in the following research tracks:

- new models for concurrency management (parallel SOM)
- load-balancing for migrating active objects
- Behaviour specifications and verification
- Computing Grids and Meta object protocols

2006 was the third and last year of the associated team, and was concluded by a final workshop held in Chili, includeing the defense of the cotutelle PhD of Javier Bustos [14].

The associated team started in january 2004, for 3 years, and a total budget of 42 kEuros.

8.3.2. Stic-Amsud ReSeCo

ReSeCo (*Reliability and Security of Distributed Software Components*) is a collaboration of INRIA with partners of the south american CONESUD, namely Un. of Cordoba (Argentina), Un. of Montevideo (Urugay), Un. Diego Portales and Un. de Chili (Chili). The two complementary themes of this project are the Specification and Verification of Component Systems on one side, and Security through Verifiable Evidence (proof Carrying Code) on the other hand. It started in november 2006 for a duration of 3 years, and will fund researcher visits and organisation of workshops.

9. Dissemination

9.1. Seminars and conferences

- A large part of the team took part in Grids@work (December, Sophia Antipolis, France).
- **Francoise Baude** was invited speaker at the EU Project consultation meeting *BIGG* (Bridging Global Computing with Grid) workshop, 28-29 Nov. 2006,

  organized the EU BIONETS working days in Sophia-Antipolis, 13-14 november 2006,

  took part with Virginie Legrand to the final review of the RNRT PISE Project, 24 November 2006, Schneider Electrics, presented OASIS team activites resp. in the context of the S4ALL project at the 2nd 2006 ObjectWeb architecture meeting, INRIA Lille, 12-13 June 2006, and in the context of Grid interoperability at the 2nd ETSI GRID-WORKSHOP Standardisation meeting, Sophia-Antipolis, 24 May 2006,

contributed to the Service Architecture of BIONETS given at TUB Berlin/FOKUS, 14-15 august 2006 and at INRIA Sophia-Antipolis 13-14 november 2006.

- **Antonio Cansado** presented the paper *Model Checking Distributed Components : The Vercors Platform* at the FACS'06 Int. Workshop (september, Prague),

  presented his research work at the meetings of Fiacre ACI (february Paris, november, Toulouse), at the ReSeCo kickoff meeting (decembre, Cordoba, Argentina) and at the last Oscar meeting (december, Valparaiso, Chili),

  participated and represented the team in the CoCoMe meeting (october, Karlsruhe).

- **Denis Caromel** has given invited ProActive courses at American University of Beirut (january, Lebanon), Guadalajara (january, Mexico), Tsinghua university (march, Biejing, China),

  gave invited talks on *Grid Programming with Components: An Advanced Component Platform for an Effective invisible Grid* at the 9th HLRS Metacomputing and Grid Workshop (july, Stuttgart), *From Theory to Practice in Distributed Component Systems* at the Int. Formal Aspects of Component Software workshop (FACS'06, september, Prague), *Programming Concurrent and GRID Applications with an Active Object Model: ObjectWeb ProActive* at JAOO (october, Aarhus, Danmark), *GCM Programming Model for the Grid* in the Component BOF, SC 2006,

  gave a presentation on *Distributed and Parallel Objects* Distributed Supercomputing, SOS 10 (march, Maui, USA),

  organized the Int. Workshop on Java for Parallel and Distributed Systems, (april, IPDPS 2006, Rhodes Island, Greece),

  co-organised the Grid@work week (GridCoord conference, GridCOMP EU meeting, ProActive User Group, etc.) together with ETSI (november, Sophia-Antipolis).

- **Alexandre di Costanzo** was invited at the Data Grid eXplorer Days (Orsay, France) to present his work on large scale experiments with Grid'5000,

  presented the workshop article [57] at the second CoreGrid Workshop on Grid and Peer to Peer Systems Architecture (Paris, France).

- **Viet Dung Doan** presented the article [25] at the second IEEE International Conference on e-Science and Grid Computing, dec. 2006.

- **Ludovic Henrio** was invited to present the team works on distributed components at the University of Chile (Santiago du Chili), the university Diego Portales (Santiago du Chili), and at the meeting of Fiacre ACI,

  took part at TCS'06 and presented the paper *Asynchonous Distributed Components: Concurrency and Determinacy* [33].

- **Fabrice Huet** participated at the ProActive booth and presentation at SuperComputing'06,

  presented the work of the group on peer-to-peer computing at the University of Sannio (Benevento, Italy),

  gave a two days lecture on ProActive to Master students of the University of Sannio (Benevento, Italy).

- **Mario Leyton** took part at EuroPar 2006 (August, Dresden, Germany), and presented the paper *Coupling Contracts for Deployment on Alien Grids* [27] in the Europar 2006 CoreGrid workshop and took part at OTM 2006 (November, Montpellier, France), and presented the paper *Grid File Transfer During Deployment Execution and Retrieval* [24] in the GADA 2006 symposium.

- **Eric Madelaine** presented the paper *Towards Real Case Component Model-Checking* at the Fractal workshop (july, Nantes, France),

  gave a presentation on *Evaluation of ProActive on Grid'5000* at the Paristic meeting (november, Nancy),

  was invited to present the research of the team at University Diego Portales (decembre, Chili),

  presented the verification activities of the team and at the meetings of Fiacre ACI (february Paris, november, Toulouse), at the ReSeCo kickoff meeting (decembre, Cordoba, Argentina) and at the last Oscar meeting (december, Valparaiso, Chili).
- **Paul Naoumenko** participated in Bionetics conference, dec. 2006.

9.2. Animation

- Oasis received the visit of Maciek Malawski, PhD student at Cyfronet (Poland) between 6 April and 12 April, and during the month of September 2006.
- Oasis received the visit of Florian Kammueller, teaching assistant at "Technische University", Berlin during the month of September.
- Oasis received the visit of Blanca ABRAHAM and José AGUILAR (Univ. Mérida) between 27 of june and 15 of july.
- Oasis received the visit of Pr. Tomàs Barros (Un. Diego Portales, Santiago de Chili) from 23 january to 10 ferbuary, and of Pr. José Piquer (Un. de Chili, Santiago) from 27/11 to 1/12.
- Oasis received the visit of Arnaud CONTES, post-doc at Université de Cardiff, from 04/09 to 30/11.
- **Françoise Baude** was publicity chair for Europe regarding HPDC 2006 (15th IEEE International Symposium on High Performance Distributed Computing),

  program commitee member of the following international conferences or workshops: IASTED PDCN *Parallel and Distributed Computing and Networks* 2007, 20th ECOOP *European Conference on Object Oriented Programming*, 2006, Workshop on Middleware for Grid Computing - MGG joint with ACM/IFIP/USENIX Middleware 2006, workshop CompFrame at HPDC 2006.
- **Denis Caromel** was member of the HPDC 2006 *workshop on Grid Programming Environments and Components*, member of the steering committee of ECOOP-AITO; was member of the PhD jury of S. Lacour (IRISA).
- **Alexandre di Costanzo** was member of the program committee of GECO-Compframe 2006 and participated to SC2006 as coordinator and exhibitor for the INRIA Sophia Antipolis booth.
- **Fabrice Huet** was member of the program committee of HPDC 2006, was member of the program committee of HIPS 2006, is serving as a technical committee member of the Workshop on Large-Scale, Volatile Desktop Grids (PCGrid 2007).
- **Eric Madelaine** is member of the steering committee and program committee of the FACS workshop (Formal Aspects of Component Software); was referee in the PhD thesis of Jiri Adamek (Charles University, Prague).
- **Bernard Serpette** was member of the Jury of the PhD thesis of Julien Cohen (University of Evry); was member of program committee of JFLA'06.

9.3. Teaching

- **Françoise Baude** is member of the *commission de spécialistes 27ème section CNU* at UNSA, gives courses on *Distributed Algorithms and Parallel Functional Programming* in the Master research RSD at Unsa, gives courses on *Parallel Functional Programming* in the master research RSD and master research PLMT at UNSA.

- **Denis Caromel** coordinates the *Distributed Systems* track of the Master research RSD (Réseaux et Systèmes Distribués) at UNSA, in collaboration with CMA, CNET, Eurécom, INRIA Sophia Antipolis; is in charge of coordinating the Master professional Télécommunications, within the département d'Informatique from UNSA; coordinates the course on *Concurrent, Parallel and Distributed Programming Languages* in the Master research RSD and Master research PLMT at UNSA; coordinates and is in charge of the courses on *Distributed Programming* in the Master Informatique at UNSA.
- **Fabrice Huet** gives courses on *Distributed Systems and RMI* in the Master Professional Télécommunications at Unsa, on *Web Programming* in the Licence Miage at Unsa, on *Tools for Software Engineering* in the Licence Professionnelle at Unsa, on *Network Game Programming* in the Licence Professionnelle at Unsa, on *Algorithms and Data Structures* in the Licence MI at Unsa.
- **Eric Madelaine** gives courses on *Component behaviour verification* in the Master research RSD at Unsa.

10. Bibliography

Major publications by the team in recent years

[1] I. ATTALI, D. CAROMEL, S. O. EHMETY. *A Natural Semantics for Eiffel Dynamic Binding*, in "ACM Transactions on Programming Languages and Systems (TOPLAS)", vol. 18, n^{o} 5, November 1996.

[2] I. ATTALI, D. CAROMEL, S. O. EHMETY, S. LIPPI. *Semantic-based visualization for parallel object-oriented programming*, in "Proceedings of OOPSLA'96, San Jose, CA", ACM Sigplan Notices, vol. 31, 10, ACM Press, October 1996.

[3] T. BARROS, R. BOULIFA, E. MADELAINE. *Parameterized Models for Distributed Java Objects*, in "Forte'04 conference, Madrid", LNCS 3235, Springer Verlag, 2004, http://hal.inria.fr/inria-00087222.

[4] F. BAUDE, D. CAROMEL, F. HUET, L. MESTRE, J. VAYSSIÈRE. *Interactive and Descriptor-based Deployment of Object-Oriented Grid Applications*, in "11th IEEE International Symposium on High Performance Distributed Computing", 2002, p. 93–102.

[5] F. BAUDE, D. CAROMEL, M. MOREL. *From Distributed Objects to Hierarchical Grid Components*, in "International Symposium on Distributed Objects and Applications (DOA), Catania, Sicily, Italy, 3-7 November, Springer Verlag", vol. 2888, Lecture Notes in Computer Science, LNCS, 2003, p. 1226–1242.

[6] F. BAUDE, D. CAROMEL, D. SAGNOL. *Distributed Objects for Parallel Numerical Applications*, in "Mathematical Modelling and Numerical Analysis Modelisation, special issue on Programming tools for Numerical Analysis, EDP Sciences, SMAI", vol. 36, n^{o} 5, 2002, p. 837–861.

[7] D. CAROMEL. *Towards a Method of Object-Oriented Concurrent Programming*, in "Communications of the ACM", vol. 36, n^{o} 9, September 1993, p. 90–102.

[8] D. CAROMEL, L. HENRIO. *A Theory of Distributed Object*, Springer-Verlag, 2005.

[9] D. CAROMEL, L. HENRIO, B. SERPETTE. *Asynchronous and Deterministic Objects*, in "Proceedings of the 31st ACM Symposium on Principles of Programming Languages", ACM Press, 2004, p. 123–134.

[10] D. CAROMEL, W. KLAUSER, J. VAYSSIERE. *Towards Seamless Computing and Metacomputing in Java*, in "Concurrency Practice and Experience", vol. 10, n^{o} 11–13, November 1998, p. 1043–1061.

[11] D. CAROMEL, L. MATEU, E. TANTER. *Sequential Object Monitors*, in "Proceedings of the 18th European Conference on Object-Oriented Programming (ECOOP 2004), Oslo, Norway", LNCS, n^{o} 3086, jun 2004, p. 316-340.

[12] F. HUET, D. CAROMEL, H. E. BAL. *A High Performance Java Middleware with a Real Application.*, in "Proceedings of the ACM/IEEE SC2004 Conference on High Performance Networking and Computing, 6-12 November 2004, Pittsburgh, PA, USA", 2004.

Year Publications

Doctoral dissertations and Habilitation theses

[13] F. BAUDE. *De la programmation parallèle structurée à la programmation pour la grille*, Habilitation à diriger des Recherches, Ph. D. Thesis, Université de Nice Sophia Antipolis, Sep 2006.

[14] J. BUSTOS-JIMENEZ. *Dynamic Load Balancing of Active Objects on Grids*, Ph. D. Thesis, Université de Nice-Sophia Antipolis, Universidad de Chili, December 2006.

[15] M. MOREL. *Components for Grid Computing*, Ph. D. Thesis, Université de Nice-Sophia Antipolis, November 2006.

Articles in refereed journals and book chapters

[16] F. BAUDE, D. CAROMEL, C. DELBÉ, L. HENRIO. *Un protocole de tolérance aux pannes pour objets actifs non préemptifs*, in "Technique et Science Informatiques", 2006.

[17] F. BAUDE, D. CAROMEL, F. HUET, T. KIELMANN, A. MERZKY, H. BAL. *Future Generation Grids*, Core-GRID series, ISBN: 0-387-27935-0. Also as the CoreGrid TR003 report, chap. Grid Application Programming Environments, Springer, jan 2006, http://www.coregrid.net/mambo/images/stories/TechnicalReports/tr-0003.pdf.

[18] D. CAROMEL, C. DELBÉ, A. DI COSTANZO, M. LEYTON. *ProActive: an Integrated platform for programming and running applications on Grids and P2P systems*, in "Computational Methods in Science and Technology", vol. 12, 2006.

[19] J. DÜNNWEBER, F. BAUDE, V. LEGRAND, N. PARLAVANTZAS, S. GORLATCH. *Towards Automatic Creation of Web Services for Grid Component Composition*, in "Integrated Research in Grid Computing", V. GETOV, D. LAFORENZA, A. REINEFELD (editors). , volume 4 of Core-GRID series, ISBN: 0-387-27935-0. Also as the CoreGrid TR003 report, Springer, jan 2006, http://www.coregrid.net/mambo/images/stories/TechnicalReports/tr-0003.pdf.

Publications in Conferences and Workshops

[20] T. BARROS, A. CANSADO, E. MADELAINE, M. RIVERA. *Model Checking Distributed Components : The Vercors Platform*, in "3rd workshop on Formal Aspects of Component Systems, Prague, Tcheque Republic", ENTCS, Sep 2006, http://hal.inria.fr/inria-00091569/en/.

[21] A. BASSO, A. BOLOTOV, A. BASUKOSKI, V. GETOV, L. HENRIO, M. URBANSKI. *Specification and Verification of Reconfiguration Protocols in Grid Component Systems*, in "Proceedings of the 3rd IEEE Conference On Intelligent Systems IS-2006", long version published as a CoreGRID Technical Report, TR-0042, IEEE, 2006.

[22] F. BAUDE, A. BOTTARO, J. BRUN, A. CHAZALET, A. CONSTANCIN, D. DONSEZ, L. GURGEN, P. LALANDA, V. LEGRAND, V. LESTIDEAU, S. MARIÉ, C. MARIN, A. MOREAU, V. OLIVE.. *Extension de passerelles OSGi pour la grande échelle : Modèles et outils*, in "Atelier OSGi, 5 Septembre 2006, co-localisé avec UbiMob'06, 3ème Journées Francophones Mobilité et Ubiquité, Paris", 2006.

[23] F. BAUDE, D. CAROMEL, C. DELBÉ, L. HENRIO. *Promised Messages: Recovering from Inconsistent Global States*, in "Proceedings of the ACM SIGPLAN 2007 Symposium on Principles and Practice of Parallel Programming", Poster, ACM Press, March 2006.

[24] F. BAUDE, D. CAROMEL, M. LEYTON, R. QUILICI. *Grid File Transfer during Deployment, Execution, and Retrieval*, in "Proceedings of the International Symposium on Grid computing, high-performAnce and Distributed Applications (GADA'06), part of OnTheMove Federated Conferences (OTM'06)", Lecture Notes in Computer Science, To appear, Springer-Verlag, November 2006.

[25] S. BEZINNE, V. GALTIER, S. VIALLE, F. BAUDE, M. BOSSY, V. D. DOAN, L. HENRIO. *A Fault Tolerant and Multi-Paradigm Grid Architecture for Time Constrained Problems. Application to Option Pricing in Finance*, in "2nd IEEE International Conference on e-Science and Grid Computing", IEEE, December 2006.

[26] J. BUSTOS-JIMENEZ, D. CAROMEL, A. IOSUP. *The Rocking Chair and the Grid: Balancing load across project grids*, in "Proceedings of the 2nd CoreGrid Integration Workshop, Cracow, Poland", Springer-Verlag, October 2006.

[27] J. BUSTOS-JIMENEZ, D. CAROMEL, M. LEYTON, J. M. PIQUER. *Coupling Contracts for Deployment on Alien Grids*, in "Proceedings of the International Euro-Par Workshops, Dresden, Germany", Lecture Notes in Computer Science, To appear, Springer-Verlag, August 2006.

[28] J. BUSTOS-JIMENEZ, D. CAROMEL, M. LEYTON, J. M. PIQUER. *Load Information Sharing Policies in Communication-Intensive Parallel Applications*, in "Proceedings of Advanced Distributed Systems: 6th International School and Symposium, ISSADS 2006, Guadalajara, Mexico", Lecture Notes in Computer Science, Springer-Verlag, January 2006.

[29] J. BUSTOS-JIMENEZ, D. CAROMEL, J. M. PIQUER. *"Load-Balancing: Towards the Infinite Network and Beyond"*, in "Proceedings of the 12th Workshop on Job Scheduling Strategies for Parallel Processing, in conjunction with SIGMETRICS 2006, Saint-Malo, France", Only interim proceedings available yet, at www.cs.huji.ac.il/~feit/parsched/p-06-9.pdf, July 2006.

[30] A. CANSADO, L. HENRIO, E. MADELAINE. *Towards Real Case Component Model-Checking*, in "5th Fractal Workshop, Nantes, France", July 2006.

[31] D. CAROMEL, V. CAVÉ, A. DI COSTANZO, C. BRIGNOLLES, B. GRAWITZ, Y. VIALA. *Executing Hydrodynamic Simulation on Desktop Grid with ObjectWeb ProActive*, in "HIC2006: Proceedings of the 7th International Conference on HydroInformatics, Nice, France", September 2006, http://hal.inria.fr/inria-00068176/en/.

[32] D. CAROMEL, A. DI COSTANZO, C. DELBÉ, M. MOREL. *Dynamically-Fulfilled Application Constraints through Technical Services - Towards Flexible Component Deployments*, in "Proceedings of CompFrame 2006, Component and Framework Technology in High-Performance and Scientific Computing, Paris, France", IEEE, June 2006, http://hal.inria.fr/inria-00068595/en/.

[33] D. CAROMEL, L. HENRIO. *Asynchonous Distributed Components: Concurrency and Determinacy*, in "Proceedings of the IFIP International Conference on Theoretical Computer Science 2006 (IFIP TCS'06), Santiago, Chile,", 19th IFIP World Computer Congress, Springer Science, August 2006.

[34] N. PARLAVANTZAS, V. GETOV, M. MOREL, F. BAUDE, F. HUET, D. CAROMEL. *Componentising a Scientific Application for the Grid*, in "Proceedings of the 2nd CoreGRID integration workshop, Cracow, Poland", also CoreGRID Technical Report, TR-0031, Oct. 2006, http://www.coregrid.net/mambo/images/stories/TechnicalReports/tr-0031.pdf.

[35] J. THIYAGALINGAM, N. PARLAVANTZAS, S. ISAIADIS, L. HENRIO, D. CAROMEL, V. GETOV. *Proposal for a Lightweight Generic Grid Platform Architecture*, in "Proceedings of CompFrame 2006, Component and Framework Technology in High-Performance and Scientific Computing, Paris, France", IEEE, June 2006.

[36] R. TOLEDO, E. TANTER, J. PIQUER, D. CAROMEL, M. LEYTON. *Using ReflexD for a Grid Solution to the N-Queens Problem: A Case Study*, in "Proceedings of the 2nd CoreGrid Integration Workshop, Cracow, Poland", Springer-Verlag, October 2006.

Internal Reports

[37] R. M. BADIA, O. BECKMANN, S. PANAGIOTIDI, D. CAROMEL, L. HENRIO, M. BUBAK, M. MALAWSKI, V. GETOV, S. ISAIADIS, J. THIYAGALINGAM, V. LAZAROV. *Lightweight Grid Platform: Design Methodology*, CoreGRID Technical Report, TR-0020, Technical report, Institute on Grid Systems, Tools and Environments, Jan 2006.

[38] A. BASSO, A. BOLOTOV, A. BASUKOSKI, V. GETOV, L. HENRIO, M. URBANSKI. *Specification and Verification of Reconfiguration Protocols in Grid Component Systems*, CoreGRID Technical Report, TR-0042, Technical report, Institute on Programming Model (WP3), May 2006.

[39] F. BAUDE, V. D. DOAN, L. HENRIO, P. NAOUMENKO, PARTNERS VTT, TUB, UBASEL, CREATENET, SUN, NOKIA OF THE BIONETS CONSORTIUM. *Specification of Service Life-Cycle*, Technical report, n^{o} D.3.2.1, BIONETS IP Project Delivrable from the Requirement and Analysis workpackage (3.2), Dec. 2006.

[40] F. BAUDE, L. HENRIO, PARTNERS VTT, TUB, UNIHH OF THE BIONETS CONSORTIUM. *Service Architecture Requirement Specification*, Technical report, n^{o} D.3.1.1, BIONETS IP Project Delivrable from the Requirement and Analysis workpackage (3.1), July 2006, http://www.bionets.eu/docs/BIONETS_D3_1_1.pdf.

[41] F. BAUDE, V. LEGRAND. *PISE: Outil et API d'administration et déploiement à grande échelle*, Technical report, July 2006.

[42] D. CAROMEL, C. DELBÉ, L. HENRIO. *Promised Consistency for Rollback Recovery*, Technical report n RR-5902, Technical report, 2006, http://hal.inria.fr/inria-00071365/en/.

[43] V. CAVÉ. *Immediate Services For The Grid*, Master recherche STIC spécialité RSD, UNSA, Technical report, 2006.

[44] V. D. DOAN. *Prototypage d'une valorisation d'options européenes et américaines par la méthode de Monte Carlo sur grille de calcul*, Master professionnel STIC spécialité IMAFA, UNSA, Technical report, 2006.

[45] C. MATHIEU. *Déploiement Hiérarchique pour la Grille.*, Master recherche STIC spécialité RSD, UNSA, Technical report, 2006.

[46] P. NAOUMENKO. *Une approche orientée compoants pour la programmation adaptative et autonome : application pour des communications autonomes.*, Master recherche STIC spécialité RSD, et 3ème année ingénieur EPU, UNSA, Technical report, 2006.

[47] OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Basic Features of the Grid Component Model (assessed)*, Deliverable D.PM.04, CoreGRID, Programming Model Institute, Technical report, December 2006.

[48] OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Programming models for the single GCM component: a survey*, Deliverable D.PM.06, CoreGRID, Programming Model Institute, Technical report, Sep. 2006.

[49] OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Proposals for a Grid Component Model*, Responsible for the delivrable, Technical report, n^{o} D.PM.02, CoreGRID, Programming Model Virtual Institute, Feb 2006.

[50] OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Survey of advanced component programming models*, Deliverable D.PM.05, CoreGRID, Programming Model Institute, Technical report, Oct. 2006.

[51] E. SALAGEANU. *An UML Profile for the Specification of Distributed Components Systems*, Rapport de stage de Master 1, Technical report, 2006.

Miscellaneous

[52] F. BAUDE. *Autonomous services – Feedback from some on-going EU research projects*, Invited talk at the Project consultation meeting "Bridging Global Computing with Grid (BIGG)", organized by the EU IST, Grids@work II conference, Nov. 2006.

[53] F. BAUDE. *INRIA OASIS's interests in Grids: programming Grids*, May 2006, 2nd ETSI GRID-WORKSHOP Standardisation meeting, Sophia-Antipolis, 24 May 2006.

[54] F. BAUDE. *ProActive: Grid programming in the Services for All (S4ALL) perspective*, June 2006, http://www.objectweb.org/phorum/read.php?f=16&i=327&t=327, 2nd 2006 ObjectWeb architecture meeting, INRIA Lille.

[55] F. BAUDE, D. CAROMEL, L. HENRIO, M. MOREL, P. NAOUMENKO. *Fractalising Fractal Controller for a Componentisation of the Non-Functional Aspects*, 5th Fractal Workshop in conjunction with ECOOP'20 – poster, July 2006.

[56] F. BAUDE, D. CAROMEL, N. MAILLARD, E. MATHIAS.. *Hierarchical MPI-like Programming using GCM Components as Implementation Support*, Selected and presented at the CoreGRID workshop: Grid Systems, Tools and Environments, Grids@work II conference, Dec. 2006.

[57] D. CAROMEL, C. DELBÉ, A. DI COSTANZO. *Peer-to-Peer and Fault-Tolerance: Towards Deployment Based Technical Services*, January 2006, http://hal.inria.fr/inria-00001238/en/.