INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team oasis

Objets Actifs, Sémantique, Internet et Sécurité

Sophia Antipolis

THEME COM

Activity Report

2005

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]

Team Vice-Leader
Eric MADELAINE [INRIA]

Administrative Assistant
Claire SENICA [INRIA]

Research Scientist
Françoise BAUDE [UNSA]
Ludovic HENRIO [CNRS, from 01/10/05]
Fabrice HUET [UNSA]
Bernard SERPETTE [INRIA]

Technical staff
Philippe HENRI [ETSI, from 01/09/05 to 31/10/05]
Virginie LEGRAND [PISE, from 01/02/05]
Stéphane MARIANI [CoreGrid, from 17/10/05 to 16/12/05]
Matthieu MOREL [France Telecom]
Marc OZONNE [S4ALL, from 01/11/05]
Romain QUILICI [GridCoord, to 15/11/05]
Igor ROSENBERG [GridCoord]

Lecturer
Laurent BADUEL [UNSA, from 01/07/05 to 31/08/05]
Rabéa BOULIFA [UNSA, to 31/08/05]

Ph.D. Student
Laurent BADUEL [UNSA, to 30/06/05]
Tomas BARROS [UNSA, to 30/11/05]
Javier BUSTOS [Co-tutelle UNSA/Univ. Chili, to 15/07/05]
Antonio CANSADO [UNSA, from 01/10/05]
Guillaume CHAZARAIN [UNSA, from 01/10/05]
Arnaud CONTES [UNSA, to 31/05/05]
Christian DELBE [UNSA]
Alexandre DI COSTANZO [UNSA]
Mario LEYTON [UNSA, from 01/11/05]
Felipe LUNA DEL AQUILA [UNSA, to 31/10/05]
Matthieu MOREL [UNSA]

Post-doctoral Fellow
Nikolaus PARLAVANTZAS [ERCIM]

Visiting scientist
Blanca ABRAHAM [Univ. Mérida, from 11/06/05 to 11/07/05]
José AGUILAR [Univ. Mérida, from 27/06/05 to 07/07/05]
Niels DROST [Univ.Amsterdam, from 27/06/05 to 08/07/05]
Jan DUNNWEBER [Univ.Munich, from 28/08/05 to 10/09/05]
Mathias ELTON [Univ. Brésil, from 03/10/05 to 23/10/05]
Nathalie FURMENTO [London college, to 31/01/05]
Ludovic HENRIO [Univ.Westminster, from 03/03/05 to 08/03/05]
Luis MATEU [Univ. Chili, from 18/02/05 to 13/03/05]

José PIQUER [Univ. Chili, from 02/10/05 to 14/10/05]
Diégo PUPPIN [Univ. PISE, from 15/06/05 to 07/07/05]
Eric TANTER [Univ. Chili, from 18/02/05 to 08/03/05 and from 01/07/05 to 15/07/05]
Andrew WENDELBORN [Univ. Adélaide, from 25/09/05 to 18/10/05]

Student intern

Santosh ANAND [UNSA, to 31/01/05]
Fabrice ANDRIANTAFIKA [ESSI 2, from 27/06/05 to 31/08/05]
Walid BELKHIR [Univ. Aix-Marseille, from 15/03/05 to 30/06/05]
Vincent CAVE [UNSA, from 24/10/05]
Vincent CAVE [ESSI 2, from 25/07/05 to 14/08/05]
Guillaume CHAZARAIN [UNSA, from 07/03/05 to 30/09/05]
Cédric DALMASSO [UNSA, from 01/07/05 to 30/09/05]
Didier DALMASSO [UNSA, from 01/07/05 to 30/09/05]
Nicolas DALMASSO [UNSA, from 01/07/05 to 30/09/05]
Viet Dung DOAN [UNSA, from 15/11/05]
Osman FARAH [ESSI 2, from 01/07/05 to 15/09/05]
Térence FERUT [UNSA, from 01/07/05 to 30/09/05]
Mario LEYTON [Univ. Chili, from 19/05/05 to 31/10/05]
Florian MARTIN [UNSA, from 04/04/05 to 31/10/05]
Clément MATHIEU [UNSA, from 01/07/05 to 30/09/05]
Samir OUIFI [UNSA, from 15/11/05]
Mohamed OUL AHMED [UNSA, from 21/10/05]
Marc OZONNE [ESSI, from 12/05/05 to 10/06/05]
Damien PRACA [ISEN Toulon, from 01/03/05 to 31/08/05]
Marcela RIVERA [Conicyt Chili, from 01/10/05]
Frédéric SASSOLAS [SUPELEC, from 29/06/05 to 09/09/05]

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.

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 correctness of those systems.

For this, calculuses 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 [17].

ASP is thus a mean 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 area 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 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 [33]. We will use then generic tools for checking properties of this model. We concentrate on behavioral 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 commercial trade: 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, like the computer and a small number of other advances, 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

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:

- 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 2005 the second 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 [61] ...) 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 (see http://www.objectweb.org). For more information, refer to [11] and to the web page http://www.inria.fr/oasis/proactive which lists a lot of white papers.

5.2. Bigloo's back-ends

Our work on Bigloo's back-ends, in collaboration with Manuel Serrano of the Mimosa project, is integrated in the Bigloo compiler (http://www-sop.inria.fr/mimosa/fp/Bigloo/).

6. New Results

6.1. Programming the Grid

6.1.1. Component-oriented Grid Programming

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 [39] starting from the Fractal component model developed by INRIA and France-Telecom. Our recent research has focused on the design of collective actions that would allow for an easier building of Grid parallel component-based applications (such as code coupling applications [31]). Our proposal in this research topic is on multi-way communications between Fractal components through the extension of the Fractal model by the specification of collective interfaces [48] [55] [44]. We propose to express multi-way bindings not only through intermediate components, but simply by exposing the collective nature of component 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, gathercast and gather-multicast are the three kinds of collective interfaces of this proposal.

Moreover, we extended ASP by building hierarchical and asynchronous distributed components [17]. 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 also conduct an important effort for wrapping legacy MPI codes within components, and further deploy them on the Grid using the configurable ProActive deployment model. We use for our current experiments a code coupling application in vibro acoustic (courtesy of EADS CCR), and wrap it using several ProActive/Fractal Components [64].

We have contributed some work about the features needed by a component-oriented framework, in order to implement a tool for component search. In particular, we addressed the Fractal framework: we describe its main features and what is needed to perform effective component search in this framework. This study was concretized by the article [45].

A promising approach for the development of grid software consists in combining high-level components, which simplify application programming, with Web Services, which enable remote access to a component-oriented system by clients developed in a totally different paradigm (like a web client for instance). Such an access from any sort of client to a parallel grid application seen as a skeleton is available in the HOCs system developed by the Univ. of Munster, one of our partners in the CoreGRID network of excellence. The purpose of this joint work was to combine HOCs and the ProActive/Fractal component model. This raised a problem which we addressed in [40]: how does one support code-carrying parameters in component interfaces exposed as web services, for configuring the ProActive/Fractal component system representing the skeleton ? We solve this problem by combining the ProActive/Fractal mechanism for automatic web service exposition with the HOC mechanism for code provisioning.

6.1.2. Typed group communications and Object Oriented SPMD

In [18], [30], and [59] we propose an evolution of the classical SPMD (Single Program, Multiple Data) parallel programming paradigm, for clusters and grids. We name it *Object-Oriented SPMD* as it is based on remote method invocation. More precisely, it is based on an active object pattern, extended as a typed group of active objects, to which SPMD specificities are added. The proposed programming model is more flexible than the low-level message-passing based approach (e.g., of MPI): indeed, techniques to postpone barrier and to remove any explicit loop make it possible to privilege reactivity and reuse. The resulting OO-SPMD API has been implemented in ProActive.

OO-SPMD is in fact an extension of the typed group communication mechanism, previously designed and added to the ProActive programming model [23] [31] [32]. Our current work is to prove that the combined approach of both object orientation and component programming models both enables efficiency (through the usage of highly efficient transport layer such as RMI/IBIS) and enhances modifiability by permitting a flexible combination of the various pieces of codes, possibly incorporated in a hierarchical manner to exhibit better scalability. This process is being experimented, in particular on the experimental Grid: Grid'5000, within the context of Jem3D, an object-oriented time domain finite volume solver on unstructured meshes for the 3D Maxwell equations modeling the propagation of electromagnetic waves [26] (in collaboration with CAIMAN).

6.1.3. Skeleton-based parallel programming

Structuring parallelism is of uttermost importance as soon as the parallelism degree and the complexity of the problems to be solved increase. Also, the heterogeneity and dynamicity of the underlying computing infrastructure, such as Grids and peer-to-peer systems, put a lot of burden on programmers. So, we started to conduct research in designing Grid-aware structured parallelism approaches, such as the ones known as algorithmic skeletons; the aim is both to hide the complexity of using the Grids, and to provide room for adaptivity and autonomicity, thanks to the usage of the component-oriented approach presented above.

Our first result in this domain is the design and implementation of a Branch-and-Bound framework, on top of the ProActive platform.

The Branch-and-Bound (BnB) method consists in an algorithmic technique for exploring a solution tree from which the optimal solution will be returned. The main goal of the BnB API, suited to grid platforms and implemented in ProActive, is to provide a set of tools for helping the developers to parallelize his BnB problem implementation.

The main features are:

- Hiding computation distribution,
- Dynamic task splitting,
- Automatic solution gathering,

- Task communications for broadcasting current best solution,
- Different strategies for task allocation, provided by the API or by the user,
- Open API for extensions.

This work also focuses on optimizing the deployment of the computational activities and their interactions through communications built upon the hierarchical group communication model.

In the context of the 2nd Grid Plugtests and contest [70], we employed this API for solving a scheduling Flowshop problem. During this event, we had the opportunity to deploy a real case on about 1600 CPUs. We continue to experiment the API with different Flowshop implementations using Grid'5000 at a large scale: each experimentation runs on at least 500 machines and takes 1 hour long as minimal time.

6.1.4. Exceptions and Asynchronism

From March to September, Guillaume Chazarain made his DEA internship on *Exceptions and Asynchronism* [63]. The achievement was to provide the well-known way of handling exceptions using the try/catch construction to programs written using asynchronous method calls. Unfortunately, the usage is not totally transparent because of Java limitations. The main part of the approach is to build an exception mask stack following the Java one, this work was published in [42].

Since October, Guillaume is a Ph.D. student and the general objective of this research will be to:

- propose, define, and experiment flexible mechanisms to deal with Functional and Non-Functional Exceptions in distributed, mobile, and asynchronous object systems,
- propose, define and experiment flexible mechanisms to deal with Error and Termination.

6.2. Deployment and Usage of Grid Systems

6.2.1. Peer-to-Peer Computing Infrastructure

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. 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.

A long-term infrastructure within INRIA Sophia Antipolis desktop computers was deployed, and we have experimented massive parallel applications for one year. In our experiments, we are the first ones in the world to have computed the number of ways of placing 25 queens on a 25 x 25 chessboard so that they do not check each other [57]. This number is equal to 2,207,893,435,808,352. The previous world record - the number of placements for 24 queens on a 24 x 24 chessboard - was held by a Japanese team. By performing the computation in six months (4444h 54mn 52s 854), the computing Grid solely composed of 260 desktop PCs working when they were not or very little used, made it possible to bypass a computation of over 50 years (53 years, 2 days, 16 hours, 27 minutes, and 1 second).

6.2.2. Dynamic Load Balancing

In [41] we present 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 explain 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 b balance 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 experimentation in order to show that a peer-to-peer load balancing features good performance in terms of migration decisions and scalability.

6.2.3. File Transfer on the Grid

We have defined and implemented a solution for integrating file-transfer tools in the generic deployment model that ProActive offers for the submission of jobs, their registration and retrieve. In [38] we present this mechanism and how it is possible to execute pre- and post-staging of application data and results, even the application code itself, the ProActive library, and the JVM at job submission time. Such a mechanism proves to be very useful for automatic code provisioning in Grid environments (like Unicore based ones) where the user is only allocated a temporary workspace.

It is worth noticing the way this mechanism compares with other approaches for Grid middleware [25] [37]: unlike others, like GAT which features an API (even if it is claimed to be a simple API) for accessing the Grid in a portable way, we promote a solution for deploying jobs, transferring files, and monitoring activities that is almost transparent for the user code. Indeed, thanks to the use of very flexible XML-based descriptors, the programming effort (and corresponding code) to operate the Grid is kept to a minimum and is stringly evolutive. Overall the approach we use in the ProActive Grid platform is a lightweight one [29].

6.2.4. Communicating through Firewalls

In [43] we test the suitability of Java to implement a scalable Web Service that solves a set of problems related to peer-to-peer interactions between Web Services that are behind firewalls or not generally accessible. In particular we describe how to enable reliable and long-running conversations through firewalls between Web Service peers that have no accessible network endpoints. Our solution is to implement in Java a Web Services Dispatcher (WSD) that is an intermediary service that forwards messages and can facilitate message exchanges by supporting SOAP RPC over HTTP and WS-Addressing for asynchronous messaging. We describe how Web Service clients that have no network endpoints, such as applets, can become Web Service peers by using an additional message store-and-forward service ("mailbox"). We also conducted a set of experiments to evaluate performance of Java implementation in realistic Web Service scenarios, involving intercontinental tests between France and the US.

In collaboration with an industrial partner, Mobile Distillery, we have also sketched a solution for moving ProActive mobile objects on web servers and further interact with them through SMS, such as to emulate the mobility of applications from PCs to mobile phones (currently only supporting the J2ME standard) [67].

The design and development of an innovative and integrated mechanism for hierarchical deployment and communication forwarding is drafted, and incorporated, as a beta version, into the new version of the ProActive middleware released in November 2005. This mechanism has been successfully used during the 2nd Grid Plugtests and contest. We had to set up a Grid as a federation of very heterogeneous grids, clusters, machines, some based on private IP addresses, some only accessible through front-ends protected by firewalls, behind which the Grid nodes were interconnected by a Virtual Private Network (as it is the case for Grid'5000). First conclusions from the usage on this mechanism on a wide scale are already available [70].

We have also started a collaboration with the OMEGA team and SUPELEC, on using ProActive for financial distributed computations on the Grid [68].

6.3. Reliability and Security

6.3.1. Security and the Grid

Arnaud Contes presented his thesis [20] on *An Hierarchical, Versatile and Dynamic Security Architecture for the Grid*. Whereas security is a key notion in the world of distributed applications, its numerous concepts are difficult to grasp when constructing such applications. Current middlewares provide many major security-related technologies. However developers still have to select the most accurate one and handle all its underlying fundations which is particularly difficult with dynamic, grid-enabled applications.

To facilitate the use of security concepts in applications, this research promotes a decentralized security model which takes care of security requirements expressed by all actors (resource providers, administrators, users) involved in a computation [27]. The model is implemented outside the application source code. Its configuration is achieved from external policy files allowing the adaptation of the application's security according to its deployment. It has been conceived to handle specific behaviors which could happen during a distributed application life-cycle (use of newly discovered resources, remote object creation). Moreover, the implantation within the ProActive library has validates the approach and has demonstrated its advantages. Indeed, thanks to its transparency, it has been seamlessly integrated with the other features of the library (migration, group communications, components, peer-to-peer). Benchmarks have consolidated the validity of the approach.

6.3.2. Reliable distributed components

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.

ProActive includes an implementation and extension of the Fractal component model [6]. This implementation provides high-level primitives and semantics for programming Java applications with distributed, asynchronous and hierarchical components. It also provides a separation between functional and non-functional aspects, the latter allows the control of a component at execution and its dynamic evolution.

We provided a formal framework to ensure that the applications built from ProActive's components are safe. Safe, in the sense that each component must be adequate to its assigned role within the system, and ensuring that the update or replacement of a component should not cause deadlocks or failures to the system.

We introduced a new intermediate format extending the networks of communicating automata, by adding parameters to their communication events and processes (pLTS). Then we use this intermediate format to give behavioral specifications of component applications [47] [34] [35]. We assume the models of the primitive components are known (given by the user or via static analysis). Using the component description, we build a controller describing the component's non-functional behavior. The semantics of a component is then generated as the synchronization product of: its LTSs sub-components and the controller. The resulting system can be checked against requirements expressed as a set of temporal logic formulas.

The results of this work are described in Tomas Barros' thesis [19]. We have started developing the tools supporting our approach, producing finite parametrized models that represent the behavior of communicating components, and their non-functional features (life-cycle management, introspection, architectural transformations). The models are built from the architectural description (ADL) of the components (Nicolas Dalmasso, Marcela Rivera). We then provide tools for building a finite abstraction of the models, and to use it as input of model-checking tools.

Antonio Cansado, starting his PhD in October 2005, will use this same model to generate parts of the implementation of components directly from the specification, in a way that will guarantee correctness properties w.r.t. the specification. Typically, (Java) interfaces can be derived automatically from the components ADL that will express the static aspects of the composite component structure and bindings. Then, the behavioral specification can be automatically implemented by code generation in a way that ensures correctness versus the component specification, and ultimately correctness of component composition. Alternatively, assertions can be generated that will check at runtime the correctness of a developer's implementation code.

6.3.3. Fault Tolerance

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 Delbe. A fault-tolerance method has been designed: it is totally transparent concerning the code, but adaptive with respect to the architecture of the underlying Grid onto which the application has been deployed. The underlying architecture relies on the notion of groups of communicating activities; 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) based fault-tolerance protocol. On the contrary, intra-group communications are subject to a Communication Induced Checkpointing (CIC) based and rollback-recovery fault-tolerance method [36] and [24]. Such a mixed protocol has the advantage that the fault of a given activity does not imply the rollback of the whole set of activities, but only of those activities that belong to the same group as the failed activity. Moreover, the usage of an inter-group protocol that is only based on the receiver implies the total independence of the checkpoint and recovery servers that are associated to each group.

6.3.4. Information Flow Security

Felipe Luna Del Aguila presented his thesis [21] on *Information Flow Security for Asynchronous, Distributed, and Mobile Applications*. The objective was to propose a security solution to regulate information flows, specifically through an access flow control mechanism, targeted to distributed applications using active objects with asynchronous communications. The work includes a security policy and the mechanism that will enforce the rules existing in such policies.

The security model [28] is founded on a strong theoretical background, the Asynchronous Sequential Processes (ASP) calculus, related to well-known formalisms. The formal semantics of ASP is extended with predicate conditions providing a formal basis to this security framework and, at the same time, making it possible to formalize the dynamic checks for unauthorized accesses. In order to prove the correctness of the security model, an intuitive secure information flow property is defined and proved to be ensured by the application of the access control model.

7. Contracts and Grants with Industry

7.1. OSMOSE

OSMOSE (http://www.itea-osmose.org) is an acronym for *Open Source Middleware for Open Systems in Europe*. It is an ITEA project, started in 2003, for 2 years, involving 60 kEuros.

The overall technical goal of the OSMOSE project is focused on the development, enhancement, and validation in defining test-beds of a comprehensive adaptable Open Source middleware to be hosted by the ObjectWeb consortium http://www.ObjectWeb.org.

The members of the project belong to 8 different European countries: Belgium, Czech Republic, France, Greece, Ireland, Netherlands, Spain and Switzerland. The project is built around three sets of partners: 6 large industrial companies (Bull, France Télécom, Philips, Telefonica, Telvent and Thales), 6 SMEs (Bantry Technologies, iTEL, Kelua, Lynx, VICORE and Whitestein Technologies), and 7 academic partners (CharlesUniversity, EPFL, INRIA, INT, LIFL, LSR and Universidad Politécnica of Madrid).

7.2. 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). This project started in July 2005, for 24 months, involving 129 kEuros.

7.3. 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 objective of the PISE project is to design, develop and validate a software infrastructure for secure Internet gateways, that can provide dynamic hosting for service and user applications, as well as a remote administration tool to manage an installed base of several thousands of gateways.

This infrastructure, fully compliant with the OSGi standard, will be based on a component model enabling a modular approach to service development. Most importantly, it will integrate in a transparent and flexible way all technical aspects related to resource management, security, distribution, and logging.

This project started in September 2004, for 24 months, involving 170 kEuros.

7.4. Extending the Fractal model For Grid Computing

This is a CRE (Contrat de Recherche Externalisée), supported by France Telecom RD, starting in October 2004, for two years, for an amount of 128 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.

On April 2005 we made the first deliverable [50] where we argued in details why the Fractal model is adequate for Grid computing.

Ongoing work focuses on the addition of collective interfaces to the Fractal model, in order to better address the coupling of (maybe parallel) components deployed on the grid. This forms the core of the second deliverable [48] due in November 2005.

7.5. Texas Instrument

During January to August, Texas Instruments and Inria had a contract of 50 kEuros. The work was done inside the Mimosa and Oasis teams. The objective of this contract was to study the JSM architecture and its potential for compilation and to propose optimization techniques. The JSM is a machine developed by Texas accepting both the Java Virtual Machine (JVM) instruction set and a general register oriented (C-ISA) instruction set. To achieve the objective of the contract, we had incorporated two new back-ends to the Bigloo Scheme compiler. The first one uses the C-ISA as target, the second makes transformations on some JVM instructions in order to generate specific C-ISA instructions. One of the optimization techniques for compilation was a register allocation policy that uses the micro-stack of JSM and the registers of the C-ISA, but other optimizations had been proposed.

7.6. ETSI

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 Grid Plugtests and Contest organized by the ETSI PlugtestsTM service and INRIA in October 2004 [69], this second edition took place from 10 to 12 October 2005, at ETSI headquarters, Sophia-Antipolis, France (250 participants). The report from this event is now available [70].

This contract started in April 2005, for 9 months, involving 25 kEuros.

8. Other Grants and Activities

8.1. National Collaborations

8.1.1. 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.

8.1.2. 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 behavioral 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 ensuring behavioral properties of distributed components [34] [35], on software integration between Oasis and Vasy tools, and on behavioral contracts for components [62].

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. Besides a strong involvement in analysis and dissemination (e.g. [60], [53], [54], [51]) we also (co)-organized two workshops: Grids@large [52] and Grids@work [49] , respectively in September and October 2005.

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.
- 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 [56] and [55], besides several oral presentations like [65], [66], [59] or [58].

This project started in September 2004, for 18 months, involving 65 kEuros in 2005.

8.3. International Collaborations

8.3.1. 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. 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 (http://www.inria.fr/oasis/oscar) [34] [35] [41]. 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)
- distributed garbage collector
- Behaviour specifications and verification
- Computing Grids and Meta object protocols

8.3.2. ECOS-Nord

ECOS-Nord with the University of Andes, Merida, Venezuela: 2 weeks visit (from 04/08 to 04/24) of Matthieu Morel in Merida, Venezuela, working with Jose Aguilar and Blanca Abraham, 2 weeks visit of Jose Aguilar (from 06/25 to 07/09), and 1 month visit of Blanca Abraham (from 06/11 to 07/10)in the Oasis team.

The topic of this collaboration is *distributed and intelligent components*, in which the Oasis team is putting its expertise in components and distributed systems to provide an environment for experimentations on intelligent agents and components, notably for a collective management of agents.

During his stay, Matthieu presented recent work carried in the Oasis team, especially contribution related to the ProActive library. During the three presentations he made, the focus was put on reflexion, components and agents, as well as Grid deployment.

This work led to the development of an architecture and a prototype of based on intelligent agents and using the Fractal component model implemented in the ProActive library.

9. Dissemination

9.1. Seminars and conferences

- A large part of the team took part in Grids@work (October, Sophia Antipolis, France). Christian Delbé, Virginie Legrand, Matthieu Morel and Alexandre di Costanzo presented the tutorial *ProActive Tutorial and Hands-On Grid Programming*; Denis Caromel, Matthieu Morel, Romain Quilici, Christian Delbé, Alexandre di Costanzo, Diego Puppin, Vincent Cavé, Andrew L. Wendelborn, Arnaud Contes, Mario Leyton, Virginie Legrand, Eric Madelaine, Fabrice Huet and Nikolaos Parlavantzas made presentations at the *ProActive User Group*.
- **Laurent Baduel** took part to CCGrid 2005 (May, Cardiff, UK) and presented the paper *Object-Oriented SPMD* [30].
- **Tomás Barros** made a presentation on *an Overview of Formal Methods* (Santiago de Chile, July 2005), took part in SPIN'05 (August, San Francisco, USA) and presented the paper *Behavioural Models for Hierarchical Components* [34]; he also presented the paper *Secured Information Flow for Asynchronous Sequential Processes* [28] at SecCo'05 (August, San Francisco, USA).
- **Francoise Baude** took part in to the European Grid Technology Days 2005, organized by the IST Grid Unit (May, Brussels, Belgium)
- **Denis Caromel** took part in IPDPS'05 (April, Denver, USA) as co-organizer of a workshop, made an invited seminar *Asynchronous Object-Oriented Middleware* at Master on Software Technologies MUTS, organized by the University of Sannio and RCOST (Research Centre on Software Technology), (April, Benevento, Napoly, Italy); made a talk *Grid Components Techniques: Composing, Gathering, and Scattering* at Coupled Problems 2005 (May, Santorini, Greece); took part in Coupled Problems'05 (May, Santorini, Greece) and presented the paper *Grid Components Techniques: Composing, Gathering, and Scattering* [31], made an invited seminar *ProActive Environment for the Grid* at the Core Grid WP 7, (June, Barcelone, Spain); made an invited seminar *Open Source Middleware for the Grid: ObjectWeb ProActive* at the Grid@Asia workshop, Beihang University, (June, Beijing, China) and the French Ministry of Industry, OrientWare meeting, on July; made an invited seminar *Programming Concurrent and GRID Applications with an Active Object Model: ObjectWeb ProActive*, at IBM Watson Research Center, (September, New York, USA); made a tutorial *Middleware issues: ObjectWeb ProActive* in CoreGrid Tutorial on Grids, CoreGRID Summer School, (September, Lausanne, Switzerland); made an invited seminar *Distributed Objects and the Grid: from Practice to Theory* at the Software PhD Program Short Course, Generalitat de Catalunya, Universitats Publiques Catalanes, (September, Barcelona, Spain).

- **Arnaud Contes** took part to ICCS 2005 (May, Atlanta, USA) and presented the paper *Deployment-Based Security for Grid Applications* [27].
- **Christian Delbe** took part to EuroPar 2005, and presented the paper *A Hybrid Message Logging-CIC Protocol for Constrained Checkpointability* [36]
- **Fabrice Huet** took part in the IMACS Special Session "New Trends in Computational Electromagnetics" (July, Paris, France), and presented the paper *Advances in Information Technologies for Electromagnetics* [26].
- **Mario Leyton** took part at the CoreGRID Integration Workshop (November, Pisa, Italia) and presented the paper *Integrating Deployment and File Transfer Tools for the Grid* [38].
- **Eric Madelaine** took part in the "Programming models and components for the Grid" workshop (October, Sophia-Antipolis, France) and gave a talk on *Behavioral Specification and Verification of Compositional Grid Components*; presented the paper *Verification of Distributed Components* at the FACS'05 workshop (October, Macau, China) and at the Fractal workshop (November, Grenoble, France)
- **Matthieu Morel** took part in ObjectWebCon'05 (January, Lyon, France), gave a talk *ProActive: an open source middleware for the Grid*, organized and ran a tutorial *Open source middleware for the Grid : distributed objects and components in ProActive*, took part in CoreGRID WP3 Programming model 2nd workshop (June, Barcelona, Spain), made a presentation *Fractal, Models, Components for the GRID* [65], took part at Middleware'05 (November, Grenoble, France) and presented the paper *Fractal ProActive and a natural extension for The Grid: multicast and gathercast interfaces* at the Fractal workshop [44].
- **Nikolaos Parlavantzas** took part at the CoreGRID integration Workshop (November, Pisa, Italia) and presented the paper *Towards Automatic Creation of Web Services for Grid Component* [40].
- **Igor Rosenberg** took part in to the European Grid Technology Days 2005, organized by the IST Grid Unit (May, Brussels, Belgium)
- **Bernard Serpette** took part in JFLA'05 (March, Obernai, France) and presented an article *Coq à la conquète des moulins* with Laurence Rideau of the Marelle project [46].
- Demonstrations of ProActive at the INRIA booth, Supercomputing 2005, by Denis Caromel and Christian Delbe (November, Seattle, USA).

9.2. Animation

- Oasis received the visit of Ludovic Henrio, from 3 to 8 March 2005, from University of Westminster in the context of CoreGrid NoE and the visit of José Piquer, professor at University of Chile, from 02 to 14 of October
- **Francoise Baude** edited [16], was publicity chair for Europe for the 15th IEEE International Symposium on High Performance Distributed Computing, was member of the program committee for the 20th ECOOP conference, June 2006; was member of the program committee of the 3rd Workshop on Middleware for Grid Computing - MGG 2005 in conjunction with ACM/IFIP/USENIX 6th International Middleware Conference; was program committee member of the IASTED Parallel and Distributed Computing and Networks conference, 2005 and 2006; was member of the HPDC 2006 workshop on Grid Programming Environments and Components; was program committee member of RenPar'06.

- **Denis Caromel** edited [22], was member of the HPDC 2006 workshop on Grid Programming Environments and Components; was co-chair of the workshop *Java for Parallel and Distributed Computing* in IPDPS'2005; was a referee in the HDR (Habilitation à Diriger des Recherches) of Nouredine Melab (University of Lille); was a referee in the PhD Thesis of Stéphane Fechter (University of Paris 6); was a referee in the PhD Thesis of Pascal Grange (University of Bordeaux I), was member of the jury of the PhD Thesis of Sébastion Lacour (University of Rennes I); was chair in the panel *Industrial Views on Existing and Future Grid Middlewares* (October, Sophia Antipolis); presented Proactive at SuperComputing conference, INRIA stand, Seattle.
- **Arnaud Contes** made a presentation on the team's objectives at EDF/INRIA reunion (Middleware and Grid), January.
- **Ludovic Henrio** visited the University of Westminster, from the 6 to 9 December, in the context of CoreGrid NoE.
- **Fabrice Huet** was member of the program committee of HPDC 2006, was member of the program committee of HIPS 2006.
- **Eric Madelaine** was member of the program comittee of LDTA 2005; was member of the jury of the PhD thesis of Christophe Joubert (Institut National Polytechnique de Grenoble).
- **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* at UNSA, gives courses on *Distributed Algorithms* 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] M. ODERSKY (editor). *Proceedings of the 18th European Conference on Object-Oriented Programming (ECOOP 2004)*, LNCS, n° 3086, SV, Oslo, Norway, jun 2004.

[2] *Proceedings of the ACM/IEEE SC2004 Conference on High Performance Networking and Computing, 6-12 November 2004, Pittsburgh, PA, USA, CD-Rom*, IEEE Computer Society, 2004.

[3] 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° 5, November 1996.

[4] 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.

[5] 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.

[6] 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.

[7] 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° 5, 2002, p. 837–861.

[8] D. CAROMEL, F. BELLONCLE, Y. ROUDIER. *The C++// System*, in "Parallel Programming Using C++", ISBN 0-262-73118-5, MIT Press, 1996.

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

[10] 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.

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

[12] D. CAROMEL, L. MATEU, E. TANTER. *Sequential Object Monitors*, in "Proceedings of the 18th European Conference on Object-Oriented Programming (ECOOP 2004), Oslo, Norway", M. ODERSKY (editor). , LNCS, n° 3086, SV, jun 2004, p. 316-340.

[13] F. HUET, D. CAROMEL, H. E. BAL. *A High Performance Java Middleware with a Real Application.*, in "SC", IEEE Computer Society, 2004, 2.

[14] E. MADELAINE. *Verification Tools from the Concur project*, in "EATCS Bulletin", n° 47, 1992.

[15] B. SERPETTE. *Approximations d'évaluateurs fonctionnels*, in "Proceedings of WSA (Workshop on Static Analysis)", Bigre, 1992, p. 79–90.

Books and Monographs

[16] F. BAUDE (editor). *Parallèlisme: Algorithmique, systèmes, applications*, Techniques et Sciences Informatiques, Versions longues de communications sélectionnées lors de RenPar'15, vol. 24, n° 5, Hermès Lavoisier, 2005.

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

Doctoral dissertations and Habilitation theses

[18] L. BADUEL. *Typed Groups for the Grid*, Ph. D. Thesis, University of Nice Sophia-Antipolis, Jul 2005.

[19] T. BARROS. *Formal specification and verification of distributed component systems*, Ph. D. Thesis, Université de Nice - INRIA Sophia Antipolis, November 2005.

[20] A. CONTES. *Une Architecture de Sécurité Hiérarchique, Adaptable et Dynamique pour la Grille*, Ph. D. Thesis, University of Nice Sophia-Antipolis, Sept 2005.

[21] F. LUNA DEL AGUILA. *Information Flow Security for Asynchronous, Distributed, and Mobile Applications*, Ph. D. Thesis, Université de Nice-Sophia Antipolis, October 2005.

Articles in refereed journals and book chapters

[22] *Parallel Programming: Models, Methods and Languages, topic in the 11th International Euro-Par Conference*M. DANELUTTO, D. CAROMEL, D. SZAFRON, F. SILVA (editors). , LNCS, Springer-Verlag, 2005.

[23] L. BADUEL, F. BAUDE, D. CAROMEL, A. CONTES, F. HUET, M. MOREL, R. QUILICI. *Grid Computing: Software Environments and Tools*, To appear, chap. Programming, Composing, Deploying, for the Grid, Springer Verlag, 2005.

[24] F. BAUDE, D. CAROMEL, C. DELBÉ, L. HENRIO. *Un protocole de tolérance aux pannes pour objets actifs non préemptifs*, in "Technique et science informatiques", A paraître, 2005.

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

[26] D. CAROMEL, F. HUET, S. LANTERI, N. PARLAVANTZAS. *Advances in Information Technologies for Electromagnetics*, L. TARRICONE, A. ESPOSITO (editors). , chap. Distributed and Object-Oriented CEM

on the Grid, Springer, 2005.

Publications in Conferences and Workshops

[27] I. ATTALI, D. CAROMEL, A. CONTES. *Deployment-Based Security for Grid Applications*, in "The International Conference on Computational Science (ICCS 2005), Atlanta, USA, May 22-25", LNCS, Springer Verlag, 2005.

[28] I. ATTALI, D. CAROMEL, L. HENRIO, F. LUNA DEL AGUILA. *Secured Information Flow for Asynchronous Sequential Processes*, in "3rd International Workshop on Security Issues in Concurrency (SecCo'05), San Francisco, USA", Electronic Notes in Theoretical Computer Science, Elsevier, aug 2005.

[29] R. BADIA, O. BECKMANN, M. BUBAK, D. CAROMEL, V. GETOV, S. ISAIADIS, V. LAZAROV, M. MALAWSKI, S. PANAGIOTIDI, J. THIYAGALINGAM. *Leightweight Grid Platform: Design Methodology*, in "1st CoreGRID integration workshop", "Also selected and presented at the CoreGrid workshop: Grid Systems, Tools and Environments, Oct. 12th 2005, Grids@work conference", Nov 2005.

[30] L. BADUEL, F. BAUDE, D. CAROMEL. *Object-Oriented SPMD*, in "CCGrid 2005: IEEE International Symposium on Cluster Computing and the Grid, Cardiff, United Kingdom", May 2005.

[31] L. BADUEL, F. BAUDE, D. CAROMEL, F. HUET, L. HENRIO, S. LANTERI, M. MOREL. *Grid Components Techniques: Composing, Gathering, and Scattering*, in "Coupled Problems 2005, Computational Methods for Coupled Problems in Science and Engineeringan, an ECCOMAS Thematic Conference", European Congress on Computational Methods in Applied Sciences and Engineering, may 2005.

[32] L. BADUEL, F. BAUDE, N. RANALDO, E. ZIMEO. *Effective and Efficient Communication in Grid Computing with an Extension of ProActive Groups*, in "JPDC, 7th International Worshop on Java for Parallel and Distributed Computing at IPDPS, Denver, Colorado, USA", apr 2005.

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

[34] T. BARROS, L. HENRIO, E. MADELAINE. *Behavioural Models for Hierarchical Components*, in "Model Checking Software, 12th International SPIN Workshop, San Francisco, CA, USA", P. GODEFROID (editor). , vol. LNCS 3639, Springer, August 2005, p. 154-168.

[35] T. BARROS, L. HENRIO, E. MADELAINE. *Verification of Distributed Hierarchical Components*, in "International Workshop on Formal Aspects of Component Software (FACS'05), Macao", Electronic Notes in Theoretical Computer Science (ENTCS), October 2005.

[36] F. BAUDE, D. CAROMEL, C. DELBÉ, L. HENRIO. *A Hybrid Message Logging-CIC Protocol for Constrained Checkpointability*, in "Proc. of the 11th International Euro-Par Conference", LNCS, vol. 3648, Springer-Verlag, 2005.

[37] F. BAUDE, D. CAROMEL, F. HUET, T. KIELMANN, A. MERZKY, H. BAL. *Grid Application Programming Environments: Comparing ProActive, Ibis, and GAT*, in "Workshop on Grid Applications: from Early Adopters to Mainstream Users. In conjunction with GGF14, (June 27, Chicago, USA)", June 2005,

http://www.cs.vu.nl/ggf/apps-rg/meetings/ggf14.html.

[38] F. BAUDE, D. CAROMEL, M. LEYTON, R. QUILICI. *Integrating Deployment and File Transfer Tools for the Grid*, in "1st CoreGRID integration workshop", "Also selected and presented at the CoreGrid workshop: Grid Systems, Tools and Environments, Oct. 12th 2005, Grids@work conference", Nov 2005.

[39] F. BAUDE, D. CAROMEL, M. MOREL. *On Hierarchical, Parallel and Distributed Components for Grid Programming*, in "Component Models and Systems for Grid Applications", V. GETOV, T. KIELMANN (editors). , Proceedings of the Workshop on Component Models and Systems for Grid Applications held June 26, 2004 in conjunction with ACM ICS 2004, Springer, June 2005, p. 97–108.

[40] F. BAUDE, V. LEGRAND, N. PARLAVANTZAS, J. DÜNNWEBER, S. GORLATCH. *Towards Automatic Creation of Web Services for Grid Component Composition*, in "1st CoreGRID integration workshop", "Also selected and presented at the CoreGrid workshop: Grid Systems, Tools and Environments, Oct. 12th 2005, Grids@work conference", Nov 2005.

[41] J. BUSTOS-JIMENEZ, D. CAROMEL, A. DI COSTANZO, M. LEYTON, J. M. PIQUER. *Balancing Active Objects on a Peer to Peer Infrastructure*, in "Proceedings of the XXV International Conference of the Chilean Computer Science Society (SCCC 2005), Valdivia, Chile", IEEE, November 2005.

[42] D. CAROMEL, G. CHAZARAIN. *Robust Exception Handling in an Asynchronous Environment*, in "Developing Systems that Handle Exceptions. Proceedings of ECOOP 2005 Workshop on Exception Handling in Object Oriented Systems. Technical Report No 05-050", A. ROMANOVSKY, C. DONY, J. KNUDSEN, A. TRIPATHI (editors). , Department of Computer Science. LIRMM. Montpellier-II University. 2005. July. France, 2005.

[43] D. CAROMEL, A. DI COSTANZO, D. GANNON, A. SLOMINSKI. *Asynchronous Peer-to-Peer Web Services and Firewalls*, in "IPDPS '05: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 5, Washington, DC, USA", IEEE Computer Society, 2005, 183.1.

[44] M. MOREL. *Fractal ProActive and a natural extension for The Grid: multicast and gathercast interfaces*, in "Fractal workshop, Grenoble, France", November 2005.

[45] D. PUPPIN, D. MOREL, D. CAROMEL, D. LAFORENZA, F. BAUDE. *GRIDLE Search for the Fractal Component Model*, in "CoreGRID Integration Workshop, Pisa, Italia", November 2005.

[46] L. RIDEAU, B. P. SERPETTE. *Coq à la conquète des moulins*, in "Actes des journées JFLA, Obernai, France", mars 2005.

Internal Reports

[47] T. BARROS, L. HENRIO, E. MADELAINE. *Behavioural Models for Hierarchical Components*, Technical report, n° RR-5591, INRIA, June 2005.

[48] F. BAUDE, D. CAROMEL, L. HENRIO, M. MOREL. *Collective interfaces for a Grid component model, Proposed extensions to the Fractal Component Model*, Technical report, Internal technical report, CRE France Telecom R&D/INRIA, Nov 2005.

[49] F. BAUDE, D. CAROMEL, I. ROSENBERG, R. QUILICI. *GridCoord Deliverable D.4.2.1 The Use of Open Middleware for the Grids*, "Report on the GridCoord workshop organised at Grids@work", Technical report, GridCoord SSA EU project, Dec 2005.

[50] F. BAUDE, M. MOREL. *A study of the adequacy of the Fractal model and tools for Grid computing*, Technical report, Internal technical report, CRE France Telecom R&D/INRIA, Apr 2005.

[51] L. BOUGÉ, I. ROSENBERG. *GridCoord Deliverable 4.3.1 Knowledge Transfer and Education*, Technical report, GridCoord SSA EU project, Nov 2005.

[52] L. BOUGÉ, I. ROSENBERG. *GridCoord Deliverable D.4.2.2 Really large-scale Grids*, Report on the GridCoord workshop co-localized with EuroPar2005, Lisbon, Technical report, GridCoord SSA EU project, Dec 2005.

[53] GRIDCOORD CONSORTIUM. *GridCoord Deliverable 3.1.1 Survey of Funding Bodies*, Technical report, GridCoord SSA EU project, Nov 2005.

[54] GRIDCOORD CONSORTIUM. *GridCoord Deliverable 3.1.2 Survey of Universities and Research Labs*, Technical report, GridCoord SSA EU project, Nov 2005.

[55] THE OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Proposals for a Common Grid Component Model for GRID*, Responsible for the delivrable, Technical report, n° D.PM.02, CoreGRID, Programming Model Virtual Institute, Nov 2005.

[56] THE OASIS TEAM AND OTHER PARTNERS IN THE COREGRID PROGRAMMING MODEL VIRTUAL INSTITUTE. *Roadmap version 1 on Programming Model*, Technical report, n° D.PM.01, CoreGRID, Programming Model Virtual Institute, Feb 2005.

Miscellaneous

[57] *Grid Computing: France holds a new computing world record*, June 2005, http://www.inria.fr/presse/pre141.en.html, Press Release.

[58] T. BARROS, E. MADELAINE. *Behavioral Specification and Verification of Compositional Grid ComponentsCoreGRID workshop on Programming models and components for the Grid, Sophia Antipolis, France*, October 2005, http://www.coregrid.net/mambo/content/view/188/30/.

[59] F. BAUDE, L. BADUEL, D. CAROMEL. *OO SPMD for the Grid: an alternative to MPI and the road to ComponentCoreGRID workshop on Programming models and components for the Grid, Sophia Antipolis, France*, October 2005, http://www.coregrid.net/mambo/content/view/188/30/.

[60] F. BAUDE. *INRIA OASIS's Interests in Grid Standardization issues*, Sept. 2005, 1st ETSI GRID standardisation meeting.

[61] F. BAUDE, D. CAROMEL, M. MOREL. *ProActive: An Open Source Middleware for Grid*, http://wiki.objectweb.org/ObjectWebCon05/, 4th annual ObjectWeb conference, Lyon, January 2005.

[62] W. BELHKIR. *Behavioural Typing for Proactice Distributed Components*, Technical report, Master recherche Univ. d'Aix-Marseille, juin 2005.

[63] G. CHAZARAIN. *Exceptions et Asynchronisme*, Technical report, Master recherche STIC spécialité RSD, UNSA, 2005.

[64] S. MARIANI. *Méthode de couplage multi-physiques à base de composants ProActive*, En partenariat avec CCR EADS et INRIA CAIMAN, Technical report, Master professionel STIC spécialité TIM, UNSA, 2005.

[65] M. MOREL, D. CAROMEL. *Fractal, Models, Components for the GRID*, June 2005, CoreGRID WP3 Programming model 2nd workshop.

[66] M. MOREL, D. CAROMEL, N. PARLAVANTZAS. *Multicast and Gathercast interfaces for the GCMCoreGRID workshop on Programming models and components for the Grid, Sophia Antipolis, France*, October 2005, http://www.coregrid.net/mambo/content/view/188/30/.

[67] D. PRACA. *Solution de transfert de données entre PC et téléphones mobiles pour les applications Java*, en partenariat avec V. Bergé, Mobile Distillery, Technical report, Ecole Supérieure d'Electronique et du Numérique de Lille, 2005.

[68] S. VIALLE, F. SASSOLAS, F. BAUDE, V. LEGRAND. *Distributed Financial Computations with ProActive*, October 2005, http://www.inria.fr/oasis/plugtest2005/ProActiveUserGroup.html, 2nd ProActive User Group.

[69] THE OASIS TEAM. *GRID PLUGTEST: INTEROPERABILITY ON THE GRID*, http://www.gridtoday.com, Vol 4, Issue 4, Jan. 2005 of GRIDtoday.

[70] THE OASIS TEAM. *Report from the 2nd Grid Plugtests and contest*, Dec 2005, http://www.inria.fr/oasis/plugtest2005/2ndGridPlug 02.pdf.