



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Projet oasis

*Objets Actifs, Sémantique, Internet et
Sécurité*

Sophia Antipolis

THÈME 2A

*R*apport
d'Activité

2002

Table des matières

1. Composition de l'équipe	1
2. Présentation et objectifs généraux	2
3. Fondements scientifiques	2
3.1. Spécifications, environnements, analyses et transformations	2
3.2. Programmation objet, concurrence et répartition	3
4. Domaines d'application	4
4.1. Panorama	4
4.2. Maintenance et manipulation de programmes	4
4.3. Logiciels sécurisés pour le Commerce Electronique	4
4.4. Programmation répartie, collaborative et sécurisée pour Internet	4
5. Logiciels	4
5.1. Répartition, mobilité et sécurité : ProActive	4
5.2. Outils Interactifs Génériques : SmartTools	5
5.3. La bibliothèque C++//	6
6. Résultats nouveaux	6
6.1. Environnement d'analyse et de vérification pour Java	6
6.2. Analyses statiques et transformations de programmes	7
6.3. Composants, spécifications sémantiques et vérifications	7
6.4. Implémentation des langages à objets	8
6.5. Bibliothèques pour la répartition	8
6.6. Etude formelle des modèles à objets distribués	10
6.7. Sécurisation des applications à objets distribuées	10
7. Contrats industriels	11
7.1. ARCAD - Architecture Répartie extensible pour Composants ADaptables	11
7.2. GRID RMI - Objets distribués haute performance pour la grille de calcul	11
7.3. Question-How	12
7.4. ProActive, projet du consortium ObjectWeb	12
7.5. Contrat SUN Microsystems	12
7.6. Projet Intellisurf	12
8. Actions régionales, nationales et internationales	12
8.1. Actions régionales	12
8.1.1. Programmation répartie et collaborative pour Internet	12
8.2. Actions nationales	12
8.2.1. Action de Recherche Coopérative Modocop	12
8.3. Actions européennes	13
8.3.1. Verificard	13
8.4. Actions internationales	13
8.4.1. Concurrence et applications	13
8.4.2. Proximos - Programmation Répartie Objet, refleXion, Sémantique	13
8.5. Visites, Participations à des conférences, et invitations de chercheurs	13
9. Diffusion des résultats	15
9.1. Animation de la Communauté scientifique	15
9.2. Enseignement	17
10. Bibliographie	18

1. Composition de l'équipe

OASIS est un projet commun à l'INRIA, au CNRS et à l'université de Nice-Sophia Antipolis.

Responsable scientifique

Isabelle Attali [DR]

Responsable permanent

Bernard Serpette [CR]

Assistance administrative

Philippe Dereymez [MOO Septembre 2001 - Août 2002]

Claire Senica [depuis le 1 Septembre 2002]

Personnel Inria

Eric Madelaine [CR]

Francis Montagnac [IR, à temps partiel]

Didier Parigot [CR]

Personnel UNSA

Françoise Baude [Maitre de Conférence, UNSA]

Denis Caromel [Professeur, UNSA, membre IUF]

Ingénieurs experts

Pascal Degenne [W3C, Question-How, SmartTools]

Alexandre Fau [W3C, Question-How, SmartTools]

Joël Fillon [SmartTools, jusque mars 2002]

Lionel Mestre [ACI GRID RMI, du 01/01/02 au 30/06/2002]

Romain Quilici [ODL Proactive depuis le 01/06/2002]

Matthieu Morel [ProActive, Arcad, depuis le 01/07/2002]

Chercheurs doctorants

Laurent Baduel [Allocataire MESR + Moniteur UNSA, deuxième année]

Tomas Barros [Bourse Conicyt Chili, depuis le 01/10/2002]

Rabéa Boulifa [Bourse INRIA, deuxième année]

Arnaud Contes [Boursier DGA, deuxième année]

Carine Courbis [ATER, puis INRIA, quatrième année, soutenance prévue le 10 décembre 2002]

Alexandre Genoud [Allocation MESR, depuis le 01/10/2002]

Ludovic Henrio [Boursier DGA, troisième année]

Fabrice Huet [Allocataire MESR + Moniteur UNSA, puis ATER, quatrième année, soutenance prévue le 11 décembre 2002]

Felipe Luna [Bourse Conacit Mexique, depuis le 02/11/2002]

Emmanuel Reuter [ingénieur IUFM Nice, troisième année]

Julien Vayssière [Boursier Région, UNSA, quatrième année, soutenance le 25 Novembre, puis post-doc industriel chez SAP]

Stagiaires

Tomas Barros [Pre-Doc, Université du Chili du 01/02/2002 au 30/03/2002]

Roland Bertuli [Maîtrise Informatique, UNSA, du 01/07/2002 au 30/09/2002]

Olivier Chabrol [DESS Informatique, Marseille, du 01/04/2002 au 30/09/2002]

Damien Ciabrini [DEA Informatique, UNSA, du 01/03/2002 au 30/05/2002, en collaboration avec Mimosa]

Christian Delbe [Maîtrise Informatique, UNSA, du 01/07/2002 au 30/09/2002]

Patrice Farrugia [DESS Informatique, Marseille, du 01/04/2002 au 30/09/2002]

Alexandre Genoud [DEA Informatique, UNSA du 01/03/2002 au 30/09/2002]

Felipe Luna [Mastère ENST Paris, du 01/07/2002 au 30/10/2002]

Toufik Maarouk [DEA Informatique, Orléans du 01/04/2002 au 30/09/2002]

Diego Niewbourg [DESS Telecom, UNSA, du 01/04/2002 au 30/09/2002]
Haris Saybasili [DEA RSD, UNSA du 01/04/2002 au 30/09/2002]
Szabolcs Szentes [Ecole des Mines de Nantes, du 02/05/2002 au 31/08/2002]
Joseph George Variamparambil [Bachelor IIT Kanpur du 09/05/2002 au 31/07/2002]
Dao Anh Viet [DEA Systèmes Informatiques Répartis, Paris 6 du 01/04/2002 au 30/09/2002]
Saida Ziane [DEA MISI, Versailles du 01/04/2002 au 30/09/2002]

Professeurs invités

Andrew Wendelborn [Univ. Adelaide, en Juin-Juillet 2002]
Paul Roe [Univ. Brisbane, Août-Décembre 2002]

2. Présentation et objectifs généraux

Dans le cadre des applications réparties (réseaux Internet et intranets, cartes à puce et terminaux), l'objectif du projet est de proposer des principes fondamentaux, des techniques et des outils pour la construction, l'analyse, la validation, la vérification et la maintenance de systèmes fiables.

Le projet rassemble deux domaines d'expertise clairement identifiés :

- sémantique, environnements, compilation et analyse statique,
- programmation à objets répartie.

A partir de ces compétences, l'objectif du projet est de créer une synergie et d'obtenir des résultats sur les thématiques suivantes :

- environnement fondé sur la sémantique pour le développement, l'analyse et la vérification d'applications réparties et communicantes liées à l'Internet (par exemple Java, Java Card) ;
- construction de bibliothèques facilitant la programmation et la maintenance d'applications multi-threadées, distribuées, sécurisées, en particulier pour les applications collaboratives et le commerce électronique.

D'autre part, l'avènement d'Internet a créé la nécessité d'étendre, de manière fondamentale, les conceptions existantes de mobilité et sécurité. Nous disposons de compétences, d'outils et de méthodes qu'il nous paraît intéressant de mettre à profit dans un domaine d'application devenu majeur. Plus précisément, nous visons les domaines d'applications suivants : applications embarquées, carte à puce, commerce électronique, télécommunications, téléphonie mobile.

3. Fondements scientifiques

3.1. Spécifications, environnements, analyses et transformations

Mots clés : *sémantique, environnements, méthodes formelles et preuves, analyses de programmes, fiabilité du logiciel, sécurité.*

Depuis les débuts de l'informatisation des tâches, l'utilisateur (du plus novice au plus expert) a toujours eu besoin d'aide pour mettre au point ses programmes, manipuler une banque de données, construire un circuit imprimé ou encore concevoir un plan d'architecte. L'existence des termes comme CAO (Conception Assistée par Ordinateur), EAO (Enseignement Assisté par Ordinateur), PAO (Publication Assistée par Ordinateur), ou dans un domaine plus proche, CASE (Computer-Aided Software Engineering), montre bien l'état des besoins.

Les domaines d'utilisation d'outils d'aide sont nombreux et variés : les interfaces homme-machine, la programmation, les aspects distribution et configuration de systèmes, les environnements de développement de preuves, la gestion de bibliothèques, la simulation et l'évaluation de performance, etc.

Nous sommes donc convaincus de l'utilité d'environnements de développement dans lesquels le programmeur sera guidé, dans la mise au point de ses applications, par des outils interactifs, graphiques, utilisant

la sémantique du langage. A ce titre, Java Card est un bon exemple par la taille réduite des applications, et permet d'envisager le traitement de problèmes insolubles dans le cas plus général d'applications contenant des millions de lignes de code.

Nous nous appuyons sur la Sémantique Naturelle (formalisme issu de la sémantique opérationnelle structurale et de la déduction naturelle). Une spécification est un ensemble de règles d'inférence qui décrivent comment déduire une conclusion à partir de prémisses. Les règles décrivent le comportement des constructeurs du langage manipulé. L'approche structurale de la méthode apparaît dans la forme des séquents, qui ont généralement un *sujet*, terme d'une syntaxe abstraite. Nous avons étudié plusieurs classes de langages de programmation, en nous spécialisant sur les langages à objets (Eiffel, Java) [1][2].

Ces environnements gagnent à être associés à des outils d'analyses statiques et de transformations qui assurent une certaine aide à la programmation et à l'optimisation de programmes [10].

Enfin le contexte applicatif (Internet, commerce électronique, cartes à puce et sécurité) nous pousse à nous intéresser à des outils de vérification qui seront particulièrement utiles aux programmeurs pour garantir qu'une politique de sécurité est respectée (identification, intégrité, confidentialité), par exemple, que tel programme gardera des données intègres en cas de retrait inopiné de la carte.

Nous nous intéressons au cas où nous pouvons, par des techniques d'analyse statique et d'interprétation abstraite, extraire un modèle fini d'une application. Nous utiliserons alors des outils de vérification génériques [8], issus des travaux du projet MEIJE et que nous adapterons aux problèmes de preuve de propriétés de sécurité, ou aux problèmes de comportement d'applications distribuées sur des réseaux.

Notre travail sur les transformations de programmes comme outils de généralité nous a amené à comparer différentes techniques de transformations issues de divers styles de programmation : fonctionnelle pour la technique de déforestation et la programmation polytypique, à objets pour les *visitor patterns* ou les *tree traversals* et enfin les grammaires attribuées [9][6].

3.2. Programmation objet, concurrence et répartition

Mots clés : *programmation à objets répartie, analyses de programmes, code mobile, collecticiels, communication de groupe, concurrence, distribution, synchronisation, sécurité.*

Le paradigme objet, même s'il date des années 70, reste un des aspects des langages de programmation les plus étudiés de nos jours. Ces dernières années, avec l'apparition du langage Java, on a pu observer une recrudescence de l'activité autour de la méthodologie objet. Autant le concept se veut universel, autant les variations des modèles et leurs implémentations possèdent des propriétés spécifiques souvent mal définies : sous la même terminologie objet, se retrouvent des thèmes particuliers, comme l'héritage (simple ou multiple), le sous-typage, la surcharge, etc.

D'autre part, les aspects programmation concurrente (en particulier, multi-threading, accès concurrents) viennent apporter un degré supplémentaire de complexité. Le mélange de l'ensemble de ces traits peut faire apparaître des cas où la spécification du langage reste floue et pour lesquels la construction d'applications et leur mise au point restent délicates.

Le langage Java peut être également abordé comme un langage très prometteur pour la programmation distribuée sur un réseau ; l'arrivée de Java a laissé espérer que l'on pourrait distribuer des applications haute performance sur le réseau Internet, premier pas vers le méta-computing. Malheureusement, les composants Java standards tels que RMI (Remote Method Invocation) n'aident en fait pas à construire de manière transparente des applications séquentielles, multi-threadées, ou distribuées, en permettant l'exécution d'une même application sur une architecture multi-processeurs à mémoire partagée aussi bien que sur un réseau de stations de travail (intranet, Internet), ou encore sur n'importe quelle combinaison hiérarchique des deux.

La question est donc : comment construire, à partir des outils standards (par exemple threads, RMI, etc), des modèles et bibliothèques facilitant la programmation distribuée ?

Nous avons développé des compétences en programmation concurrente, répartie et parallèle dans le cadre des langages à objets ; ces recherches sont menées aussi bien sur des aspects théoriques comme les sémantiques formelles, ou les analyses statiques et transformations pour la répartition automatique ou semi-automatique,

que sur des aspects pragmatiques comme la conception de langages et de méthodes de programmation [4], ou la construction de bibliothèques pour le parallélisme [3][5].

4. Domaines d'application

4.1. Panorama

Les domaines d'application du projet Oasis couvrent tous les aspects du logiciel embarqué (cartes à puces, commerce électronique, téléphonie mobile) ainsi que les aspects liés aux applications réparties et communicantes sur intranets et Internet (par exemple applications collaboratives).

4.2. Maintenance et manipulation de programmes

La maintenance des systèmes logiciels est l'un des points critiques du cycle de vie d'un logiciel. La durée de vie de ces logiciels a augmenté ainsi que leur complexité due à des mises à jour successives, soit pour corriger des bugs, soit pour ajouter de nouvelles fonctionnalités. Les problèmes de l'an 2000 et du passage à l'Euro sont deux exemples concrets de maintenance de logiciel. La taille et la complexité de ces systèmes logiciels rendent leur maintenance coûteuse et hasardeuse si elle n'est pas automatisée. Les systèmes de manipulation et de transformation de programmes, cantonnés jusqu'ici dans des applications de taille réduite, semblent désormais adaptés pour répondre à ces nouveaux besoins.

4.3. Logiciels sécurisés pour le Commerce Electronique

Plusieurs domaines d'applications du réseau Internet et des cartes à puces, dont le commerce électronique, demandent la protection de données précieuses (numéros de compte en banque, codes confidentiels, etc) qui, en un moment déterminé, seront disponibles sur des plates-formes reliées physiquement à un vaste réseau. Ici le besoin de sécurité maximale est réel et partagé :

- les sociétés de services (banques, administration etc.) doivent avoir la garantie qu'une application vérifie des propriétés de sécurité, définies dans une politique de sécurité globale (par exemple identification, intégrité, confidentialité, ou non-répudiation) et pourra ainsi résister à des attaques systématiques ;
- leurs clients doivent être assurés que les informations qu'ils fournissent lors d'une demande de services ne seront pas utilisées à mauvais escient ou détournées vers un tiers.

4.4. Programmation répartie, collaborative et sécurisée pour Internet

Nos champs d'application comprennent les systèmes collaboratifs (par exemple des systèmes d'entreprise intranet et Internet), mettant ainsi l'accent sur les aspects liés à la sécurité, les systèmes transactionnels commerciaux, bancaires, etc.

Un des domaines d'application particulièrement représentatif est la construction et l'évolution de collectifs (plusieurs utilisateurs distants travaillent de manière coordonnée à une même tâche) dans lesquels se posent des problèmes d'élection, de synchronisation, de répartition des calculs, etc.

5. Logiciels

5.1. Répartition, mobilité et sécurité : ProActive

Participants : Laurent Baduel, Françoise Baude, Denis Caromel [correspondant], Alexandre Genoud, Fabrice Huet, Lionel Mestre, Matthieu Morel, Romain Quilici, Emmanuel Reuter, Julien Vayssière.

Mots clés : *programmation objet, parallélisme et répartition, représentation Meta-Objet, Mobilité, Sécurité, Metacomputing.*

Notre objectif est de permettre l'exécution d'une même application sur une architecture multi-processeurs à mémoire partagée, sur un réseau de stations de travail, sur Internet ou encore sur n'importe quelle combinaison hiérarchique.

Pour attaquer ce problème, nous avons développé une bibliothèque 100% Java, qui fournit des threads transparents, des objets distants et mobiles, des appels (bi-point et multipoint) asynchrones avec futurs transparents, et des mécanismes de synchronisation de haut niveau. Cette bibliothèque permet très facilement de répartir et rendre collaborative toute application écrite en Java.

Afin de démontrer la puissance de cette bibliothèque, nous avons par exemple développé des applications collaboratives parallèles et distribuées qui permettent à plusieurs utilisateurs de travailler ensemble sur une scène 3D avec des mécanismes d'élection et de synchronisation : l'image de la scène est calculée par un ensemble dynamique de moteurs de rendu utilisant un algorithme de lancer de rayon. Nous avons également développé une plateforme d'administration système et réseau qui permet, par le biais d'objets mobiles, d'administrer des équipements distants en utilisant le protocole standard SNMP. Une collaboration démarrée (Color 2002) avec Stéphane Lanteri de l'équipe CAIMAN de l'UR de Sophia-Antipolis a permis de mettre au point une version entièrement Java d'une application de résolution numérique d'un système d'équations de Maxwell 3D. A partir d'une première version Java purement séquentielle, il s'est avéré aisé d'en dériver une version répartie par l'utilisation de ProActive. Bien que présentant des performances non encore satisfaisantes comparé à la version écrite précédemment en Fortran et parallélisée grâce à la bibliothèque d'échange de messages MPI, cette expérience a permis d'illustrer de nombreuses caractéristiques offertes par ProActive - comme par exemple la communication multi-point - et de démontrer l'intérêt d'une approche totalement portable et extensible pour ce type d'applications.

Un environnement de mise au point (IC2D : Interactive Control & Debug for Distribution) de contrôle et inspection, pour les applications développées en ProActive est disponible. La gestion du support d'exécution est rendue transparente et est externalisée par rapport au code de l'utilisateur. Cette gestion offre notamment du chargement dynamique de code, l'interfaçage avec des machines hôtes accessibles via Globus, le déploiement de JVMs automatisé avec l'aide de fichiers de déploiement au format XML.

ProActive est ainsi particulièrement adaptée au développement d'applications réparties sur l'Internet grâce notamment à la réutilisation de code initialement non réparti, à une synchronisation automatique et à la possibilité de faire migrer des activités d'une machine à l'autre.

Depuis avril 2002, ProActive est soutenue par le consortium ObjectWeb <http://www.objectweb.org>. Pour plus d'information, consulter [5] et la page <http://www.inria.fr/oasis/proactive>.

5.2. Outils Interactifs Génériques : SmartTools

Participants : Isabelle Attali, Carine Courbis, Pascal Degenne, Alexandre Fau, Joël Fillon, Didier Parigot [correspondant].

SmartTools, générateur d'environnements de développement interactif, permet de générer, à partir de spécifications formelles associées à un langage, un environnement interactif de développement pour ce langage.

L'intérêt et l'originalité de notre approche est de rendre accessible et facilement utilisable des techniques avancées de programmation initialement développées pour les langages de programmation. En particulier, les nouvelles approches de programmation adaptative, par aspects et par composants que nous utilisons largement, nous semblent être un atout très important pour ce type d'applications en termes de modularité et de réutilisation de composant.

Un challenge supplémentaire est de proposer ces nouveaux mécanismes de programmation à des non-spécialistes en les intégrant dans un environnement uniforme et interactif et sur une architecture modulaire largement ouverte aux technologies XML. Les retombées industrielles de ce type de plateforme sont à la hauteur des enjeux des applications du commerce électronique et des cartes à puces. En d'autres termes, l'ouverture vers les langages métiers offre à notre outil un vaste champ d'application et justifie pleinement notre approche générique.

L'originalité et l'innovation de notre approche peuvent se synthétiser en quatre points importants :

1. Accepter en entrée des formalismes W3C (DTD et schéma), ce qui nous permet du même coup de profiter des nombreux développements autour des standards W3C. Notre innovation consiste à proposer pour des documents XML, une méthodologie de programmation (pour décrire les traitements sémantiques) fondée sur les travaux autour des « design patterns », issus de la programmation à objets.
2. Fournir une interface utilisateur conviviale est aussi un impératif ; l'innovation de notre approche est de traiter tous ces aspects d'affichage (y compris l'interface utilisateur) sur un même modèle. Cela permet de proposer une approche homogène et uniforme avec un fort potentiel de réutilisation tant pour la plate-forme SmartTools que pour les environnements produits.
3. Pour assurer une bonne évolution de l'outil, il était vital de concevoir une architecture logicielle modulaire (par composants indépendants) et extensible vers d'autres technologies de composant [32].
4. Enfin, nous proposons une approche originale de programmation par aspect au-dessus de la technique des visiteurs qui ne requiert pas de transformation des sources. Cette approche dynamique a l'intérêt d'être beaucoup plus simple dans sa mise en œuvre.

Une nouvelle version, version 4, qui devrait être disponible en fin d'année, intègre un ensemble de travaux décrits dans la section 6.3. Pour plus d'information, voir <http://www-sop.inria.fr/oasis/SmartTools>.

5.3. La bibliothèque C++//

Mots clés : *programmation objet, parallélisme et répartition, représentation Meta-Objet, Metacomputing.*

Participants : Françoise Baude, Denis Caromel [correspondant], David Sagnol.

C++// est une bibliothèque pour C++ permettant de répartir une application en réutilisant le maximum du code séquentiel. Son implémentation au dessus du système de metacomputing Nexus/Globus permet à une application C++// d'utiliser des ordinateurs répartis sur l'Internet, Globus se chargeant du contrôle d'accès et de l'authentification de l'utilisateur sur ces machines.

Depuis la fin du travail de thèse de David Sagnol, C++// n'est plus vraiment maintenue, mais reste publiquement accessible : en effet, elle nous sert de vitrine auprès d'équipes de calcul scientifique, numérique, qui voudraient utiliser un modèle parallèle, réparti, de programmation orientée objet, mais qui, de par leur souci important de recherche de performance, sont attirés en premier lieu par C++// et non pas par ProActive. Cependant, ces deux bibliothèques offrant le même modèle de programmation, il est assez facile de les diriger vers ProActive, car nous arrivons à les convaincre que, quitte à utiliser un modèle objet, autant utiliser Java. Ainsi, les résultats et l'expérience acquis durant les recherches menées autour de C++// sont reportés sur celles autour de ProActive. Mais notre souci de recherche de performance et d'application de notre modèle de programmation répartie au calcul scientifique parallèle reste d'actualité. Concrètement, nous avons entamé et poursuivons une collaboration (color 2002) avec l'équipe CAIMAN de l'UR de Sophia-Antipolis, ciblée vers le développement d'une bibliothèque orienté objets (en Java et ProActive) pour le calcul réparti intensif en électromagnétisme.

Pour plus d'information sur C++//, consulter la page <http://www-sop.inria.fr/oasis/c++//>.

6. Résultats nouveaux

6.1. Environnement d'analyse et de vérification pour Java

Participants : Dao Anh Viet, Isabelle Attali, Tomas Barros, Rabéa Boulifa, Denis Caromel, Ludovic Henrio, Toufik Maarouk, Eric Madelaine.

Dans une première étape en direction d'un environnement d'analyse et de vérification de propriétés pour des applications (Java) distribuées, nous développons des méthodes et des outils pour la vérification automatique de propriétés comportementales de programmes ProActive. Les propriétés qui nous intéressent sont des

propriétés temporelles liées aux échanges de messages entre objets distribués, en particulier des propriétés de sûreté (absence de blocage, ordonnancement) ou de vivacité, ou plus globalement de conformité à une spécification. Certains types de propriétés de sécurité (gestion des droits d'accès à des ressources, confidentialité) peuvent aussi utiliser des méthodes similaires.

Nous développons des méthodes de génération de modèles finis d'applications ProActive, basées sur des techniques d'analyse statique. Ces modèles sont ensuite utilisés dans des outils de *Model Checking* existants.

En amont, nous avons amorcé avec Dao Anh Viet [45] l'étude des relations entre la sémantique opérationnelle fondée sur le calcul d'objets ASP ([35]) et une sémantique comportementale décrivant les aspects dynamiques d'une application ProActive (création d'objets actifs et échanges de messages entre objets actifs). La définition et la mise en oeuvre de cette sémantique comportementale pour la génération de modèles finis et compositionnels pour ProActive est au coeur des travaux de thèse de Rabéa Boulifa [30], qui réalise aussi, en collaboration avec l'équipe Lande de l'IRISA, un prototype implémentant les analyses statiques du code ProActive et la génération des modèles comportementaux.

Toufik Maarouk a étudié deux questions relatives au lien entre notre plate-forme d'analyse et les outils de vérification automatiques [41], à savoir la comparaison, en terme de formats intermédiaires et d'expressivité de la logique, de deux outils de vérification issus respectivement des travaux des équipes Meije à Sophia (Fc2Tools), et Vasy à l'Inria Rhone-Alpes (CADP - Evaluator). Toufik Maarouk a ensuite défini une extension du format intermédiaire Fc2 commun à ces deux outils permettant le codage direct de système d'automates communicants paramétrés.

Enfin, Tomas Barros a étudié la spécification d'un exemple de système distribué pour le traitement électronique des déclarations de TVA au Chili, sous forme de systèmes d'automates communicants, mettant en évidence la nécessité d'utiliser des méthodes compositionnelles pour espérer faire passer nos outils à l'échelle d'applications réalistes. Ce travail se poursuit en thèse.

6.2. Analyses statiques et transformations de programmes

Participants : Isabelle Attali, Denis Caromel, Ludovic Henrio, Didier Parigot, Bernard Serpette.

Les travaux, en collaboration avec Manuel Serrano du projet Mimosa, sur la compilation de Scheme dans la machine virtuelle Java (JVM), par extension du compilateur existant Bigloo, ont abouti à la publication d'un article [29]. Dans le même cadre de collaboration, nous travaillons sur la génération de code pour la machine virtuelle de .NET. Notre premier prototype fournit des résultats comparables aux premières versions du générateur JVM : ce travail apparait donc comme prometteur.

Les travaux sur la vérification de bytecode ont abouti à l'acceptation d'un article [25].

Dans le cadre du stage de Szabolcs Szentes, nous avons étudié une série de quatre algorithmes et leurs complexités associées résolvant la plus petite des réductions statiques plus communément connue sous le nom de « 0th order control flow analysis ». Même si cette analyse est généralement déclarée comme calculable en temps cubique, l'algorithme complet atteignant cette complexité n'est pas direct. Le dernier de ces algorithmes (et le meilleur) que nous proposons n'a, à notre connaissance, jamais été publié et s'exécute, dans les pires cas, en un temps cubique par rapport à la taille du programme [31].

6.3. Composants, spécifications sémantiques et vérifications

Participants : Isabelle Attali, Carine Courbis, Pascal Degenne, Alexandre Fau, Didier Parigot, Joseph George Variamparambil, Olivier Chabrol, Patrice Farrugia, Saida Ziane.

Dans le cadre de l'outil SmartTools, l'un des résultats importants de cette année est certainement la construction d'une vraie architecture à base de composants. Cette évolution était nécessaire dans le cadre de notre participation au projet européen QUESTION-HOW <http://www.w3.org/2001/qh/> avec le W3C.

SmartTools est constitué d'outils de base et de composants spécifiques pour chaque langage traité. Ces outils et composants offrent et requièrent des services qu'il faut connecter et rendre accessibles à travers une interface utilisateur. Par exemple, chaque manipulation au niveau d'une vue graphique (sélection, remplacement,

insertion ou appel d'un traitement) va se traduire par l'invocation des services au niveau du document associé. L'aspect générique de SmartTools, un méta-outil, impose que son interface utilisateur et son architecture soient les plus configurables et extensibles possibles pour pouvoir les adapter aux services spécifiques de chaque nouveau langage traité. Il est impératif que toutes les entités logicielles produites soient autonomes et puissent être exécutées sans nécessiter la présence de l'interface, par exemple en mode commande (*batch*) ou dans d'autres environnements. Il doit aussi être facile d'ajouter des composants externes en ne modifiant ni la plateforme, ni ces composants.

Dans les premières versions de l'outil, l'absence d'une vraie approche par composants était un handicap pour son évolution. Nos composants étaient indissociables d'un bus logiciel qui assurait toutes les communications. Toute intégration et toute exportation de composants impliquaient la présence de ce bus. L'approche orientée composant s'est donc imposée pour la réalisation de notre outil. Aucun des modèles existants, tels que les *Web Services*, les composants CORBA (*CCM - CORBA Component Model*) ou les EJB, ne semblait correspondre parfaitement à nos besoins spécifiques et cette dépendance vis-à-vis d'une technologie nous paraissait préjudiciable à l'évolution de l'outil. Nous avons donc conçu notre propre modèle (indépendamment de ces technologies) adapté et restreint à nos besoins, mais surtout transposable vers ces technologies de composants.

Ces travaux sur un modèle à composant ont abouti à plusieurs publications [24][26][20].

Joseph George Variamparambil [32] a étudié et implanté la transformation de notre modèle de composant vers les trois technologies suivantes : *Web Services*, CORBA et les EJB.

Saida Ziane [46] a étudié une traduction possible des Schemas (le nouveau formalisme du W3C pour décrire des fichiers XML) vers notre formalisme de syntaxe abstraite.

Olivier Chabrol [36] a étudié les relations entre les formalismes de modélisation proposés par l'OMG, en particulier UML, et les formalismes de description de syntaxe abstraite (AST). Ce travail s'inscrivait dans le cadre de notre soumission de projet RNTL, XLUC (<http://www-sop.inria.fr/oasis/SmartTools/XLUC.html>). XLUC est un acronyme pour XML - Langages - UML - Contrats.

Patrice Farrugia [38] a travaillé sur un nouveau langage de description (de transformation) de syntaxe de surface et d'affichage graphique. Ce travail a abouti à la conception d'un nouveau langage Cosynt qui est actuellement disponible dans la version 4 de SmartTools.

6.4. Implémentation des langages à objets

Participants : Denis Caromel, Ludovic Henrio, Bernard Serpette.

La première version de notre interpréteur de bytecode JVM et JCVM utilisait une structure de donnée spécifique pour la représentation des objets. Cette technique a l'avantage de pouvoir étudier divers types d'implémentation, mais se heurte au problème des fonctions natives : celles-ci doivent être réécrites afin de prendre en compte la nouvelle structure des objets, soit environ 200 fonctions dans les packages `java.*`.

Pour pallier ce problème, nous avons développé une seconde version de l'interprète où la structure des objets coïncide avec celle de l'interprète. Ainsi, les fonctions natives peuvent être appelées par l'interprète de bas-niveau. Néanmoins, par une granularité trop forte de l'API de réflexion (`java.lang.reflect.Constructor.newInstance`), l'interprète de haut-niveau perd le contrôle de l'évaluation durant l'appel au constructeur.

Le premier interprète reste utilisable dans le cadre de bibliothèques restreintes (i.e. la JCVM), tandis que le deuxième est préconisé dans le cadre d'applicatifs plus lourds sachant que les résultats obtenus (graphe d'appel, consommation des ressources,...) seront une approximation des résultats réels.

6.5. Bibliothèques pour la répartition

Participants : Laurent Baduel, Françoise Baude, Roland Bertuli, Denis Caromel, Christian Delbe, Alexandre Genoud, Fabrice Huet, Lionel Mestre, Matthieu Morel, Diego Niewbourg, Romain Quilici, Emmanuel Reuter, Paul Roe, Haris Saybasili, Julien Vayssière.

Nous développons un modèle de programmation concurrente et répartie à objets ainsi qu'une implémentation sous la forme d'une bibliothèque appelée ProActive, 100 % Java (<http://www-sop.inria.fr/oasis/proactive>), faisant suite aux travaux autour d'Eiffel// et de C++// (<http://www-sop.inria.fr/oasis/c++//>).

Les qualités essentielles du modèle proposé sont :

- de rendre transparent à l'utilisateur le fait que :
 - les objets de son application sont ou non distants,
 - et sont supportés ou non par des threads,permettant ainsi la réutilisation de code, par exemple pour un déploiement sur Internet (un de nos objectifs étant de parvenir à une conservation de la sémantique) ;
- de fournir un modèle haut niveau de gestion de la concurrence entre objets actifs.

Le jeu de primitives que fournit le MOP (Meta Object Protocol) est utilisable pour la définition de mécanismes plus élaborés que le « simple » appel distant de méthode. En particulier, nous avons introduit un mécanisme de communication de groupe qui étend le principe de l'appel de méthode asynchrone et distante entre deux objets actifs, à l'invocation asynchrone et distante de méthode sur un groupe d'objets actifs [22]. Ce mécanisme a été notamment utilisé pour simplifier la programmation d'une application de résolution d'équations de Maxwell 3D à partir de méthodes numériques mises au point dans l'équipe CAIMAN de l'UR [34]. Dans ce travail, il s'agit de proposer un modèle et une bibliothèque à objets Java répartis qui outre son potentiel d'extensibilité soit compétitive face à des versions Fortran-MPI.

Nous avons également défini un mécanisme de migration d'objets utilisable par le programmeur grâce à un ensemble minimal de primitives, mais dont la gestion est totalement transparente [18]. Il devient dès lors possible de construire des API d'agents mobiles grâce à la construction d'itinéraires d'un plus haut niveau d'abstraction. Nous avons démontré qu'une classe d'applications pouvant parfaitement être supportée par le modèle offert par ProActive, et en particulier, la construction d'itinéraires de haut niveau, consiste en de l'administration de systèmes en réseau à l'aide d'agents mobiles [28][27].

Nous avons entamé une étude portant sur la définition, l'interception et la gestion d'exceptions dans un contexte réparti à base d'objets mobiles [39]. Ce travail se poursuit actuellement en thèse.

L'implémentation du MOP est aussi le lieu idéal pour injecter des optimisations d'exécution, comme par exemple le recouvrement du transfert de certains paramètres d'un appel distant de méthode, avec l'exécution de cette méthode distante [17]. Même si ce résultat a été obtenu en utilisant la bibliothèque C++// [19], il s'agit d'un concept réutilisable notamment dans le cadre de ProActive.

Nous avons conçu et développé un environnement de déploiement, mise au point, contrôle, inspection, pour les applications développées en ProActive : IC2D (Interactive Control & Debug for Distribution) [23].

Les caractéristiques principales d'IC2D sont, sans introduire aucune modification dans le code de l'utilisateur :

- interactivité forte (visualisation graphique de la topologie d'une application répartie, nombreux contrôles à la souris) ;
- possibilité pour l'utilisateur d'influer sur le comportement de l'application répartie (par exemple, choix dynamique de nœuds où créer de nouveaux objets actifs comme par exemple sur des nœuds d'une grille de calcul accessible via Globus [42], déplacement d'un objet actif d'un nœud à un autre pendant l'exécution de l'application, etc),

Le principe consiste à rajouter un moniteur qui va être informé de façon transparente d'événements jugés importants pour suivre le déroulement d'une application à objets actifs répartis (tels par exemple les envois et réceptions de requêtes de service et de leurs réponses). Couplé à ces possibilités interactives pour influer sur le déploiement d'une application en cours d'exécution, nous avons introduit l'utilisation de fichiers de déploiement écrits au format XML. A l'aide de tels fichiers, on indique toute information utile afin de savoir, à

l'instanciation de l'application, sur quels hôtes démarrer les JVMs et déployer les objets actifs, précisant ainsi les paramètres de déploiement de l'application sans avoir besoin de la recompiler.

Cette externalisation de la configuration de certains aspects d'une application ProActive, est poussée plus avant par l'introduction d'un modèle de composants pour ProActive [43]. L'objectif est de réussir à encapsuler des objets actifs dans des composants pour permettre de construire des applications réparties à base de composants interconnectés. En utilisant un modèle de composants, tel Fractal, permettant la *reconnexion dynamique* de composants, on pourra obtenir des applications écrites en ProActive dynamiquement reconfigurables. Notre choix d'utiliser le modèle de composants Fractal, supporté aussi par le consortium ObjectWeb et par le projet RNTL Arcad, se justifie également par le fait que Fractal offre un modèle de composants *hiérarchiques*. Ceci permettra d'emboîter des composants ProActive et de leur appliquer ainsi récursivement des opérations de contrôle, comme par exemple la migration de tout un groupe de composants vers un autre cluster de PCs. Cette définition d'un modèle de composants pour ProActive se poursuit en thèse.

Paul Roe a étudié comment exposer les objets ProActive comme des services Web, sur la base d'un exemple (application C3D) ; plus généralement, il serait utile de combiner les concepts de ProActive (objets distribués) et les technologies du Web (XML, Service Web, XQuery) afin de pallier deux problèmes constatés : (1) la dualité statique vs dynamique (pour donner de nouveaux services à runtime) ainsi que (2) des communications plus lourdes à mettre en place et moins efficaces. Des expérimentations sont en cours.

6.6. Etude formelle des modèles à objets distribués

Participants : Denis Caromel, Ludovic Henrio, Fabrice Huet, Bernard Serpette.

En collaboration avec Sara Alouf et Philippe Nain (Mistral), nous avons étudié la modélisation de différentes méthodes de localisation d'agents mobiles. Pour assurer la communication avec un agent mobile, deux approches principales sont possibles, l'une centralisée (requête de localisation d'un agent mobile, auprès d'un serveur de noms) et l'autre distribuée (utilisation d'une chaîne de répéteurs afin de parvenir jusqu'à l'agent mobile). Nous utilisons la modélisation par chaînes de Markov pour comparer les performances de ces deux approches. L'objectif est de trouver un modèle qui soit assez réaliste dans chaque cas et de définir les valeurs des paramètres du système qui assurent la stabilité. De plus, nous souhaitons parvenir à des recommandations sur l'approche à choisir pour une application donnée.

Dans l'approche centralisée, une chaîne de Markov à 27 états a été étudiée et des expérimentations sur le système réel dans un contexte de LAN et de MAN ont permis de conclure que la modélisation reflète correctement le système réel [21][16]. Nous avons par ailleurs complexifié la modélisation dans le cas d'un nombre variable d'agents mobiles dans le système, et mis en place un mécanisme mixte qui permettra le choix dynamique de la meilleure politique de localisation [14].

Par ailleurs, dans le cadre de la thèse de Ludovic Henrio, nous étudions les caractéristiques que devrait avoir un calcul d'objets pour nous permettre de démontrer un certain nombre de nouvelles propriétés sur le modèle de répartition utilisé pour la bibliothèque ProActive [35]. Ces travaux devraient aboutir à la démonstration d'un certain nombre de conjectures formulées sur ce calcul, notamment sur son déterminisme.

6.7. Sécurisation des applications à objets distribués

Participants : Denis Caromel, Arnaud Contes, Fabrice Huet, Felipe Luna, Bernard Serpette, Julien Vayssière.

Nous étudions, d'une part, des critères qui garantissent certaines propriétés de sécurité, et, d'autre part, les outils qui pourraient permettre de formaliser ces propriétés. Le cadre de ces recherches s'applique à Java et JavaCard, et naturellement aux objets distribués ProActive.

Par ailleurs, dans [15], nous avons étudié les problèmes de sécurité posés par l'utilisation de protocoles à métaobjets (MOP) dans le cadre d'application à composants développées dans la langage Java. Les contributions de ce travail se répartissent selon deux axes.

Nous avons tout d'abord étudié les problèmes de sécurité soulevés par l'utilisation des différents types de MOP pour Java, et comment l'architecture de sécurité existante permet, ou non, d'y répondre. Nous en avons

conclu que le modèle de sécurité de Java doit être étendu afin de pouvoir appliquer des contrôles d'accès aux opérations effectuées par le code de méta niveau sur le code du niveau de base.

Nous avons ensuite conçu et implémenté un MOP à objets d'interception pour Java avec le double objectif d'être non-intrusif et de sécuriser les interactions entre les métaobjets et les objets de base. Après avoir montré comment la présence du méta niveau peut être transparente pour le niveau de base du point de vue de la sécurité, nous avons défini un ensemble de permissions destinées à contrôler les opérations que les métaobjets peuvent faire sur, à la fois, les objets du niveau de base et les éléments de l'exécution du programme de base réifiés.

Nous avons enfin implémenté un certain nombre de comportement de niveau méta afin de mener des expériences avec notre MOP et son architecture de sécurité. Ces résultats prouvent que la granularité des permissions choisies est correcte, et que le surcoût introduit par le mécanisme de sécurité intégré au MOP est tout à fait acceptable. Nous terminons en expliquant comment notre architecture de sécurité pourrait être appliquée à d'autres types de MOP pour Java.

D'autre part, et dans le cadre de la bibliothèque ProActive, nous souhaitons proposer une méthodologie et les outils associés afin de faciliter le développement d'applications réparties, collaboratives sur Internet : possibilité de définir des domaines virtuels, dans lesquels les objets actifs s'exécutent, avec description de contraintes d'authentification, intégrité, confidentialité de toute communication entre objets actifs inter-domaines [33]. Ceci est à rapprocher des techniques de mise en œuvre de Virtual Private Networks.

Enfin, Felipe Luna, dans le cadre de son stage de Mastère ENST Paris a étudié différents modèles existants pour définir ces politiques de sécurité qui spécifient les conditions, règles et procédures à suivre pour bien garantir et atteindre les objectifs de sécurité. Il a proposé un modèle propre aux systèmes distribués basés sur des objets répartis, avec un contrôle de flux de l'information transmise, et avec des opérations sous environnements multi-niveaux [40]. Ce travail se poursuit en thèse.

7. Contrats industriels

7.1. ARCAD - Architecture Répartie extensible pour Composants ADaptables

Participants : Françoise Baude, Denis Caromel, Fabrice Huet, Matthieu Morel, Julien Vayssière.

Il s'agit d'un contrat RNTL démarré en 2001, d'une durée de 3 ans, et d'un montant de 1 020 kF.

Les partenaires sont les équipes Rainbow (I3S CNRS UNSA), DTL/ASR (France Télécom R&D), SIRAC (INRIA Rhône-Alpes), et OCM (Ecole des Mines de Nantes).

Le travail effectué par l'équipe dans le cadre de ce projet est développé dans 6.5.

7.2. GRID RMI - Objets distribués haute performance pour la grille de calcul

Participants : Laurent Baduel, Françoise Baude, Denis Caromel, Fabrice Huet, Matthieu Morel, Lionel Mestre, Romain Quilici.

Dans le cadre de l'ACTION CONCERTÉE INCITATIVE 2001, nommée « Globalisation des Ressources Informatiques et des Données », le projet « GRID RMI : Objets distribués haute performance pour la grille de calcul », auquel participe l'équipe Oasis, vise globalement à simplifier la programmation répartie, le debugging, le contrôle et le déploiement d'applications de metacomputing, pour un montant de 37 keuros pour deux ans. Plus précisément, les deux objectifs principaux sont :

- promouvoir un modèle de programmation pour les grilles de calcul combinant à la fois des modèles du calcul parallèle et du calcul distribué,
- concevoir une plate-forme d'objets distribués qui prend en compte des ressources réseaux variées afin d'être capable d'exploiter, de la façon la plus transparente possible, les performances des réseaux sous-jacents.

Les partenaires du projet sont l'IRISA, le LIP, LIFL, et EADS (European Aeronautic Defence and Space Company). Le travail effectué par l'équipe dans le cadre de ce projet est développé dans 6.5.

7.3. Question-How

Participants : Isabelle Attali, Carine Courbis, Pascal Degenne, Alexandre Fau, Didier Parigot.

Le générateur d'environnement de programmation XML SmartTools est partiellement supporté par le projet IST (Information Society Technologies), QUESTION-HOW (2002-2004) piloté par le W3C/Inria (financement de deux ingénieurs sur un an). La contribution de l'équipe consiste à adapter la plate-forme SmartTools afin d'utiliser complètement les standards W3C (XML, XML Schema, DOM, Web Services, etc).

Les activités de recherche sont décrites en 6.3.

7.4. ProActive, projet du consortium ObjectWeb

Participants : Françoise Baude, Denis Caromel, Fabrice Huet, Matthieu Morel, Romain Quilici.

Depuis Avril 2002, la bibliothèque ProActive est un projet ObjectWeb, et est désormais disponible sous licence LGPL (Gnu Lesser General Public License).

7.5. Contrat SUN Microsystems

Mots clés : *Composants, parallélisme et distribution, Java.*

Participants : Françoise Baude, Denis Caromel, Alexandre Genoud, Matthieu Morel, Romain Quilici.

Sun Microsystems supporte (25 000 \$) les travaux sur la bibliothèque ProActive, en particulier, la conception et l'implantation d'un modèle de composant standard pour des applications parallèles et distribuées en Java.

Les résultats scientifiques sont développés en 6.5.

7.6. Projet Intellisurf

Participants : Pascal Degenne, Alexandre Fau, Didier Parigot.

Le logiciel SmartTools a été mis à disposition en 2001 de la startup Dynam-IT (pour une durée de 6 mois) à des fins d'évaluation. En effet, Dynam-IT a des activités de développement et commercialisation de logiciels d'accélération des livraisons de contenus web. Les clients visés sont d'une part les sites internet professionnels et d'autre part les fournisseurs d'accès. Les terminaux visés sont les PC. Ensemble, Dynam-IT et Oasis ont déposé un projet RNTL « Intellisurf » qui a été labellisé en 2002, et devrait pouvoir démarrer dès que la jeune pousse Dynam-IT aura pu lever les fonds. La première étape est son incubation au sein de la structure INRIA-Transfert.

L'objectif du projet IntelliSurf est de développer, à partir de l'outil SmartTools, une plate-forme d'accélération des livraisons de contenu web, fondée sur un système de méta-données XML et de cache réparti à trois niveaux : serveur, personal proxies coopérants, et méta-réseau de type CDN.

8. Actions régionales, nationales et internationales

8.1. Actions régionales

8.1.1. Programmation répartie et collaborative pour Internet

Participants : Denis Caromel, Julien Vayssière.

Cette action, soutenue partiellement par la Région PACA, permet de financer la thèse de doctorat de Julien Vayssière.

8.2. Actions nationales

8.2.1. Action de Recherche Coopérative Modocop

Participants : Rabea Boulifa, Toufik Maarouk, Eric Madelaine.

L'action de recherche coopérative Modocop : Model checking Of Concurrent Object-oriented Programs. Le but de cette action est la spécification, la vérification et le test symbolique de programmes à objets concurrents. Ceci inclut la vérification de programmes JavaCard single-thread, à partir d'outils d'aide à la preuve, de techniques symboliques basées sur des modèles finis ou infinis. Les participants sont les équipes Lande, Lemme (coordinateur), Oasis, Vasy, Vertecs, Verimag. <http://www-sop.inria.fr/lemme/modocop/>.

Les activités de recherche sont décrites en http://www.inria.fr/rapportsactivite/RA2001/oasis/resul_verifs_mn.html.

8.3. Actions européennes

8.3.1. Verificard

Participants : Eric Madelaine.

Eric Madelaine participe aux travaux du projet Verificard sur la modélisation et vérification de la plate-forme et des programmes Javacard. Les partenaires sont, en dehors de l'INRIA : GemPlus, Bull, Universités de Nimegue, Munich, Hagen, Sics (voir <http://www.verificard.com/>).

8.4. Actions internationales

8.4.1. Concurrence et applications

Denis Caromel est co-responsable du groupe de travail "Concurrence et applications" (The Concurrency, Applications, and Benchmarks Group) du Java Grande Forum (voir <http://www.javagrande.org>). Dennis Gannon (Indiana University and NASA Ames, USA) est co-responsable de ce groupe.

L'objectif du Java Grande Forum est de constituer un groupe de conseil et de pression pour faciliter l'utilisation de la plate-forme Java dans les applications hautes performances.

8.4.2. Proximos - Programmation Répartie Objet, reflexion, Sémantique

Cette collaboration INRIA-Conicyt a démarré en 2002 par la visite de Tomas Barros (février-avril 2002) et José Piquer et Eric Tanter (juin 2002). Le thème de cette collaboration est d'étudier et d'intégrer des techniques et des logiciels à base de protocoles meta-objets. Deux visites au Chili sont prévues : la première en novembre 2002 à l'occasion de la conférence SCCC'02 (XXII International Conference of the Chilean Computer Science Society), Copiapo ; la deuxième en décembre 2002, à Santiago.

8.5. Visites, Participations à des conférences, et invitations de chercheurs

- Visites et présentations invitées :
 - Isabelle Attali, Denis Caromel et Julien Vayssière ont participé aux enseignements de l'Ecole Cimpa « Objets, Distribution et Internet », 7-18 janvier 2002 à Mérida, Venezuela.
 - Denis Caromel :
 - * a présenté la bibliothèque ProActive lors de la journée GRID INRIA, le 31 Janvier à l'ENS-Lyon.
 - * a donné une journée de cours sur le thème « Objets, concurrence, répartition et mobilité » à l'École Jeunes Chercheurs en Programmation, 20-31 mai 2002, Rennes.
 - * a présenté la bibliothèque ProActive et l'interface de contrôle IC2D à Edinburgh, IEEE International Symposium On High Performance Distributed Computing, HPDC-11, 24-26 juillet, 2002
 - * a présenté les principes et la mise en œuvre des technologies à agents mobiles lors d'un séminaire invité à EDF, 8 octobre 2002, Clamart.

- * a donné une conférence invitée à SCCC'02 (conf. IEEE) à Copiapo, Chili (voir <http://www.sccc.cl/jcc2002/>) dont le titre était "Towards software components for the Grid" ; par ailleurs, ce voyage a permis de former 25 étudiants de dernière année école d'ingénieur/master/doctorants à la bibliothèque ProActive.
- Didier Parigot :
 - * a présenté la plate-forme SmartTools à France Telecom R&D Lannion le 13 septembre 2002.
 - * a participé à plusieurs réunions de travail préparatoire au montage de projets RNTL (Paris, 12 juillet, 23 juillet, 15 novembre, 22 novembre).
- Participations écoles :
 - Laurent Baduel, Rabéa Boulifa et Arnaud Contes ont participé à l'Ecole Jeunes Chercheurs en Programmation, Rennes, mai 2002.
 - Julien Vayssière a participé à l'Ecole d'été « Ubiquitous and pervasive computing » à Dagstuhl en Allemagne, du 6 au 14 août 2002.
 - Arnaud Contes et Felipe Luna ont participé au séminaire de formation Microsoft Research sur la sécurité, 4-6 novembre 2002 à Cambridge.
 - Laurent Baduel et Alexandre Genoud participent à l'Ecole GRID Calcul Distribué, Méta-Computing, Globalisation des Ressources, 2-6 décembre 2002 à Aussois.
- Participations conférences :
 - Plusieurs membres du projet ont participé aux réunions de travail régulières des projets MODOCOP, ARCAD, GRID-RMI, et ObjectWeb.
 - Une grande partie de l'équipe a participé à ETAPS 2002, Grenoble, 13-20 avril 2002.
 - Isabelle Attali et Denis Caromel ont participé à la conférence IPDPS 2002, à Fort Lauderdale, avril 2002, USA. Isabelle Attali a présidé la session industrielle et est en charge (avec Kiran Bondalapati) de la session industrielle pour IPDPS 2003, Nice, avril 2003.
 - Isabelle Attali et Denis Caromel ont participé à la réunion du groupe de travail JRAF (Java Research Activities in France), 11-12 juillet 2002, Bordeaux.
 - Laurent Baduel a participé à la conférence conjointe Java Grande-ISCOPE 2002, où il a présenté ses travaux, novembre 2002. Il a également participé à OOPSLA 2002, co-localisée avec Java Grande-ISCOPE.
 - Françoise Baude
 - * a participé à l'animation du club des utilisateurs du Calcul Parallèle sur le site de Sophia Antipolis et dans sa région (une réunion par trimestre environ) et dans ce contexte, a donné des séminaires "Un exemple d'outil de programmation parallèle et distribuée orientée objets : ProActive", en avril et en décembre.
 - * a organisé les réunions du projet ARCAD et de l'ACI Grid RMI qui ont eu lieu à Sophia Antipolis durant l'année 2002.
 - Denis Caromel a participé à la Conférence Ecoop (European Conference on Object-Oriented Programming), 11-14 juin 2002, Malaga.
 - Denis Caromel et Romain Quilici ont participé à la conférence SuperComputing 2002, 25-29 novembre 2002, Baltimore, pendant laquelle ils ont présenté des démonstra-

- tions de la bibliothèque ProActive.
- Fabrice Huet a participé a la conférence ACM Sigmetrics 2002, 15-19 juin 2002, Marina Del Rey, et a présenté ses travaux sous la forme d'un poster, puis à la Conférence Performance 2002, 23-27 septembre 2002, Rome.
 - Eric Madelaine a participé :
 - * avec Rabéa Boulifa au troisième workshop international sur « Verification, Model checking and abstract interpretation », 20-22 janvier 2002, Venise.
 - * au quatrième workshop sur « Rigorous Object-Oriented Methods », 21-22 mars 2002, Londres.
 - * à la Conférence FLOC (Federated Logic Conference) 20 juillet-1er août 2002, Copenhagen.
 - * à la conférence FMCO'02 (First International Symposium on Formal Methods for Components and Objects), 5-8 novembre 2002, Amsterdam.
 - Matthieu Morel a participé à la conférence Java Scope, 2-4 octobre 2002, Versailles, pendant laquelle il a présenté des démonstrations de la bibliothèque ProActive.
 - Didier Parigot a participé :
 - * à LMO (Langages et Modèles Objets), 23-25 janvier 2002, Montpellier.
 - * aux journées composants des 17 et 18 octobre 2002, Grenoble.
 - Emmanuel Reuter a présenté ses travaux lors des conférences MATA (4th International Workshop on Mobile Agents for Telecommunications Applications), 23-24 octobre 2002, Barcelone et IPOM 2002 (IEEE Workshop on IP Operations and Management), 29-31 octobre 2002, Dallas.
 - Bernard Serpette a participé aux journées JFLA, Journées Francophones des Langues Applicatifs, 27-29 janvier 2002, Anglet.
- Invitations de chercheurs :
 - Andrew Wendelborn (University of Adelaide), juin 2002.
 - Paul Roe (QUT Brisbane), août-décembre 2002.

9. Diffusion des résultats

9.1. Animation de la Communauté scientifique

- Isabelle Attali
 - est membre nommé de la Commission d'Evaluation (depuis Septembre).
 - a participé au Comité de Programme de la conférence CARDIS'02, Fifth Smart Card Research and Advanced Application Conference, San Jose, Novembre 2002.
 - organise avec Refik Molva (Eurecom) et Joachim Posegga (SAP) des groupes de travail « SecurAntipolis » sur le thème de la sécurité. La première réunion a eu lieu à SAP le 15 Mai (33 participants). Le programme est disponible à l'adresse <http://www-sop.inria.fr/oasis/SecurAntipolis.html>. La deuxième réunion a eu lieu le 18 septembre à l'INRIA (29 participants). Le programme est disponible à l'adresse <http://www-sop.inria.fr/oasis/SecurAntipolis2.html>. La troisième réunion est prévue le 10 Décembre à Eurecom.

- a été la responsable scientifique de la première session InTech Sophia Antipolis sur les « Web Services » le 22 Mars (programme disponible à l'adresse <http://www-sop.inria.fr/oasis/Intech.html>), et fait partie du comité de pilotage de InTech Sophia.
- Françoise Baude
 - est membre de la commission des spécialistes, 27^e section de l'UNSA.
 - a participé au comité de pilotage de RenPar en mai, étant en charge de l'organisation de ces Rencontres du Parallélisme (RenPar) en octobre 2003 dans la région niçoise.
 - participe au comité d'édition de la revue Calculateurs Parallèles, éditions Hermès (depuis Mars 96).
 - a dirigé et édité un ouvrage sur le calcul réparti à grande échelle, paru en mai 2002 chez Hermès Sciences - Lavoisier [11].
 - a participé au comité de programme de LMO 2003, des journées composants des 17 et 18 octobre 2002, ainsi qu'à celui d'un numéro spécial de TSI sur le thème "Systèmes à composants adaptables et extensibles" (à paraître en 2003).
 - a été *examineur* dans le cadre du Jury de Doctorat de Eric Gascard, 3 Juillet 2002, Université de Aix-Marseille I.
- Denis Caromel
 - a participé aux Comités de Programmes :
 - * CCGrid, IEEE International Symposium on Cluster Computing and the Grid, May 21-24, 2002, Berlin, Germany, (<http://www.ccgriid.org>)
 - * HPDC, IEEE International Symposium on High Performance Distributed Computing, HPDC-11 Edinburgh, Scotland, July 24-26, 2002 (voir <http://www.hpdc.org/>)
 - * ASIAN, Asian Computing Science Conference, Hanoi, Vietnam, December 4-6, 2002, (voir <http://www.lirmm.fr/asian02/>)
 - * JAVA PDC, Workshop on Java for Parallel and Distributed Computing, an IPDPS'2002 workshop, (voir <http://www.ipdps.org/>) [12].
 - est président de comité de sélection des tutoriels de la conférence IPDPS 2002 IEEE Computer Society and ACM SIGARCH, Fort Lauderdale, Florida, Avril 15-19 2002.
 - fait partie des Steering Committee :
 - * ACM SIGPLAN Java Grande ISCOPE Conference (voir <http://www.javagrande.org/>).
 - * ECOOP (AITO, Association Internationale pour les Technologies Objets).
 - a été *examineur* dans le cadre du Jury de Doctorat de Martha Gonzalez, 9 Septembre 2002, UPMC - LIP6 (Laboratoire d'Informatique de Paris 6).
- Didier Parigot a participé au comité d'organisation du 2ieme Workshop sur « Language Descriptions, Tools and Applications » (LDTA'02), le 13 avril 2002 Grenoble, dans le cadre de ETAPS'2002 (voir <http://www.cwi.nl/~markvdb/LDTA2002/>).
- Bernard Serpette a participé au comité de programme de JFLA 2002, Journées Francophones des Langages Applicatifs, Janvier 2002.

Plusieurs membres du projet ont évalué des articles soumis aux :

- conférences suivantes :

Cardis 2002, RenPar'02, SuperComputing'02, LDTA'02, HPDC 2002, ISCOPE/Java Grande 2002, SIROCCO 2002, Journées Composants 2002, CCGrid 2002, Asian02, JFLA 2002, CC'03, LMO 2003.

- et revues suivantes : M2AN (Mathematical Modelling and Numerical Analysis), Software Practice and Experience.

9.2. Enseignement

- Isabelle Attali est responsable de la commission Formation Emploi de l'association Télécom-Valley.
- Françoise Baude
 - est coordinateur au département d'informatique de la Licence d'informatique, et pour 2002-2003 également de la Maitrise d'informatique.
 - est responsable du module « Algorithmique et environnements de programmation parallèle et répartie », de la filière SAR de l'ESSI 3ieme année,
 - est responsable du module de mise à niveau en Systèmes, du DESS Télécommunications de l'UNSA,
 - est responsable de l'enseignement de Licence d'informatique de l'UNSA "Concepts des systèmes d'exploitation et gestion de la concurrence",
 - est en charge du cours "Systèmes Distribués" en maitrise MIAGE,
 - participe à l'enseignement de Java en Licence Informatique,
 - participe au module de tronc commun du DEA RSD "Programmation des systèmes distribués".
- Denis Caromel
 - est responsable du module de Maitrise d'informatique « Programmation distribuée et Administration Système »,
 - est responsable de la filière « Systèmes Distribués » du DEA RSD (Réseaux et Systèmes Distribués) de l'UNSA (en collaboration avec CMA, CNET, Eurécom, INRIA Sophia Antipolis), depuis septembre 1995,
 - est responsable depuis 1995 du Module « Langages de Programmation Concurrente, Parallèle, Distribuée » commun aux DEAs Informatique et RSD de l'UNSA,
 - est coordinateur au Département Informatique du DESS Télécommunications, Université de Nice - Sophia Antipolis.
- Eric Madelaine intervient au DEA Informatique UNSA, dans le cours de tronc commun « Méthodes formelles et fiabilité du logiciel » organisé par Robert de Simone (module de 36h).
- Laurent Baduel :
 - Systèmes Informatiques, Deug Maths Info (39h TP).
 - Programmation Applicative (Scheme), Deug Maths Info (26h TP).
 - Programmation C et outils, Licence Informatique (26h TD).
- Rabéa Boulifa a assuré des TPs de Java en Deug Maths-info (40h).
- Carine Courbis :
 - Technologie Web, Licence MIAGE, TD sur machines (2*6h).
 - Compilation, Licence MIAGE, TD sur machines (21h).
 - Programmation système, Licence Informatique, TD sur machines (26h).

- Encadrement, Licence Informatique (2h).
- Julien Vayssière :
 - Programmation distribuée, DESS Telecom (6h cours + 6h TD)
 - Sécurité, DESS Telecom (6h cours + 6h TD)
- Ludovic Henrio a assuré 2h de cours, 4h de TD en DESS Telecom sur la sécurité JavaCard.

10. Bibliographie

Bibliographie de référence

- [1] I. ATTALI, D. CAROMEL, S. O. EHMETY. *A Natural Semantics for Eiffel Dynamic Binding*. in « ACM Transactions on Programming Languages and Systems (TOPLAS) », numéro 5, volume 18, novembre, 1996.
- [2] I. ATTALI, D. CAROMEL, S. O. EHMETY, S. LIPPI. *Semantic-based visualization for parallel object-oriented programming*. in « Proceedings of OOPSLA'96 », série ACM Sigplan Notices, volume 31, 10, ACM Press, San Jose, CA, octobre, 1996.
- [3] D. CAROMEL, F. BELLONCLE, Y. ROUDIER. *The C++// System*. in « Parallel Programming Using C++ », MIT Press, 1996, ISBN 0-262-73118-5.
- [4] D. CAROMEL. *Towards a Method of Object-Oriented Concurrent Programming*. in « Communications of the ACM », numéro 9, volume 36, septembre, 1993, pages 90-102.
- [5] D. CAROMEL, W. KLAUSER, J. VAYSSIÈRE. *Towards Seamless Computing and Metacomputing in Java*. in « Concurrency Practice and Experience », numéro 11-13, volume 10, novembre, 1998, pages 1043-1061.
- [6] L. CORRENSON, E. DURIS, D. PARIGOT, G. ROUSSEL. *Generic Programming by Program Composition (position paper)*. in « Workshop on Generic Programming », Marstrand, Sweden, juin, 1998, <ftp://ftp-sop.inria.fr/oasis/Didier.Parigot/publications/Correnson98a.ps.gz>, conjunction with MPC'98.
- [7] N. FURMENTO, F. BAUDE. *Schooner : An Object-Oriented Run-time Support for Distributed Applications*. in « In K. Yetongnon and S. Hariri, editors , Proceedings of Parallel and Distributed Computing Systems (PDCS'96), Dijon », volume 1, International Society for Computers and their Applications (ISCA), pages 31-36, septembre, 1996.
- [8] E. MADELAINE. *Verification Tools from the Concur project*. in « EATCS Bulletin », numéro 47, 1992.
- [9] D. PARIGOT, G. ROUSSEL, M. JOURDAN, E. DURIS. *Dynamic Attribute Grammars*. in « Int. Symp. on Progr. Languages, Implementations, Logics and Programs (PLILP'96) », série Lecture Notes in Computer Science, volume 1140, Springer-Verlag, éditeurs H. KUCHEN, S. D. SWIERSTRA., pages 122-136, Aachen, septembre, 1996, <ftp://ftp-sop.inria.fr/oasis/Didier.Parigot/publications/plilp96.ps.gz>.
- [10] B. SERPETTE. *Approximations d'évaluateurs fonctionnels*. in « Proceedings of WSA (Workshop on Static Analysis) », Bigre, pages 79-90, 1992.

Livres et monographies

- [11] éditeurs F. BAUDE., *Calcul réparti à grande échelle - Meta-computing*. Hermès Science - Lavoisier, mai, 2002, ISBN 2-7462-0472-X..
- [12] *Java for Parallel and Distributed Computing*. éditeurs D. CAROMEL, S. CHAUMETTE, G. FOX, P. GRAHAM., IPDPSP'02, Workshop in IEEE IPDPS'2002 Proceedings, IEEE CS Press, Fort Lauderdale, avril, 2002, ISBN 0-7695-1573-8.

Thèses et habilitations à diriger des recherche

- [13] C. COURBIS. *Contribution à la programmation générative - Applications dans le générateur SmartTools : technologies XML, programmation par aspects et composants*. thèse de doctorat, Université de Nice - Sophia Antipolis, décembre, 2002.
- [14] F. HUET. *Objets Mobiles : conception d'un middleware et évaluation de la communication*. thèse de doctorat, Université de Nice - Sophia Antipolis, décembre, 2002.
- [15] J. VAYSSIÈRE. *Une architecture de sécurité pour les applications réflexives - Application à Java*. thèse de doctorat, Université de Nice - Sophia Antipolis, novembre, 2002.

Articles et chapitres de livre

- [16] S. ALOUF, F. HUET, P. NAIN. *Forwarders vs. centralized server : An evaluation of two approaches for locating mobile agents*. in « Performance Evaluation », volume 49, septembre, 2002, pages 299-319, ISSN 0166-5316.
- [17] F. BAUDE, D. CAROMEL, N. FURMENTO, D. SAGNOL. *Optimizing remote method invocation with communication-computation overlap*. in « Future Generation Computer Systems, Elsevier », volume 18, 2002, pages 769-778, Selected article from PaCT 2001.
- [18] F. BAUDE, D. CAROMEL, F. HUET, J. VAYSSIÈRE. *Objets actifs mobiles et communicants*. in « Technique et science informatiques », numéro 6, volume 21, 2002, pages 1-36.
- [19] 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 », numéro 5, volume 36, 2002, pages 837-861.
- [20] C. COURBIS, P. DEGENNE, A. FAU, D. PARIGOT. *L'apport des technologies XML et Objets pour un générateur d'environnements : SmartTools*. in « revue L'Objet, numéro spécial XML et les Objets », 2002, À paraître, <ftp://ftp-sop.inria.fr/oasis/publications/2002/smartobjet02.pdf>.

Communications à des congrès, colloques, etc.

- [21] S. ALOUF, F. HUET, P. NAIN. *Forwarders vs. centralized server : an evaluation of two approaches for locating mobile agents*. in « SIGMETRICS 2002, International Conference on Measurements and Modeling of Computer Systems », ACM, pages 278-279, Marina Del Rey, California, 2002, Poster.

- [22] L. BADUEL, F. BAUDE, D. CAROMEL. *Efficient, Flexible, and Typed Group Communications in Java*. in « Joint ACM Java Grande - ISCOPE 2002 Conference », ACM Press, pages 28-36, Seattle, 2002, ISBN 1-58113-559-8.
- [23] 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 HPDC-11 », IEEE Computer Society, pages 93-102, Edimburgh, 2002, ISBN 0-7695-1686-6.
- [24] C. COURBIS, P. DEGENNE, A. FAU, D. PARIGOT. *Un modèle de composants pour l'atelier de développement SmartTools*. in « Journée systèmes à composants adaptables et extensibles », Octobre, 2002, <ftp://ftp-sop.inria.fr/oasis/publications/2002/smartcompo02.pdf>, ISBN 2-7261-1229-3.
- [25] L. HENRIO, B. P. SERPETTE. *A Parametrized Polyvariant Bytecode Verifier*. in « Actes des journées JFLA », Chamrousse, France, janvier, 2003.
- [26] D. PARIGOT, C. COURBIS, P. DEGENNE, A. FAU, C. PASQUIER, J. FILLON, C. HELD, I. ATTALI. *Aspect and XML-oriented Semantic Framework Generator : SmartTools*. in « ETAPS'2002, LDFA workshop », Electronic Notes in Theoretical Computer Science (ENTCS), Grenoble, France, avril, 2002, <ftp://ftp-sop.inria.fr/oasis/publications/2002/smartldfa02.pdf>.
- [27] E. REUTER, F. BAUDE. *A mobile-agent and SNMP based management platform built with the Java ProActive library*. in « IEEE Workshop on IP Operations and Management (IPOM 2002) », pages 140-145, Dallas, 2002, ISBN 0-7803-7658-7.
- [28] E. REUTER, F. BAUDE. *System and Network Management Itineraries for Mobile Agents*. in « 4th International Workshop on Mobile Agents for Telecommunications Applications, MATA », Lecture Notes in Computer Science vol. 2521, Springer-Verlag, pages 227-238, Barcelona, 2002.
- [29] B. SERPETTE, M. SERRANO. *Compiling Scheme to JVM Bytecode : a Performance Study*. in « ICFP'02 - International Conference on Functional Programming », October 4-6, 2002, Pittsburgh, PA, USA, 2002.

Rapports de recherche et publications internes

- [30] R. BOULIFA, E. MADELAINE. *Preuves de propriétés de comportement de programmes ProActive*. rapport technique, numéro RR4460, INRIA, mai, 2002, <http://www.inria.fr/rrrt/rr-4460.html>.
- [31] B. P. SERPETTE, S. SZENTES. *Implementation and Complexity of the Lowest Static Reduction*. Rapport de Recherche, 2002.
- [32] J. G. VARIAMPARAMBIL. *Enabling SmartTools components with component technologies : WebServices, CORBA and EJBs*. rapport technique, INRIA, 2002, <ftp://ftp-sop.inria.fr/oasis/Didier.Parigot/publications/Rapport/Variamparambil>

Divers

- [33] I. ATTALI, D. CAROMEL, A. CONTES. *Security for Distributed and Mobile Active Objects with the ProActive Library*. ErcimNews, numéro 49, Avril, 2002, http://www.ercim.org/publication/Ercim_News/enw49/.

-
- [34] F. BAUDE, R. BERTULI, D. CAROMEL, C. DELBE, S. E. KASMI, S. LANTERI. *An Object-Oriented Application for 3D Electromagnetism*. 2002, Soumis à publication.
- [35] D. CAROMEL, L. HENRIO, B. SERPETTE. *Towards Asynchronous and Deterministic Distributed Objects*. 2002, Soumis à publication.
- [36] O. CHABROL. *Passerelle entre les méta-modèles (UML) et les formalismes de Syntaxe abstraite à l'aide des formalismes du W3C (XMI)*. rapport technique, Univ. Marseille, juin, 2002, Stage DESS Informatique.
- [37] D. CIABRINI. *Scheme et JVM : un environnement de programmation*. rapport technique, Université de Nice - Sophia Antipolis, juin, 2002, Stage DEA Informatique.
- [38] P. FARRUGIA. *Etude d'un langage de transformation, Cosynt*. rapport technique, Marseille, juin, 2002, Stage DESS Informatique.
- [39] A. GENOUD. *Exceptions fonctionnelles et non fonctionnelles pour objets mobiles et asynchrones*. rapport technique, Université de Nice - Sophia Antipolis, juin, 2002, Stage DEA Informatique.
- [40] F. LUNA. *Sécurité par Contrôle de Flux et Multi-niveaux des Systèmes Distribués avec Objets Mobiles et Communicants*. rapport technique, ENST Paris, octobre, 2002, Stage Mastère Sécurité.
- [41] T. MAAROUK. *Outils pour le model-checking d'applications Java distribuées*. rapport technique, Université d'Orléans, septembre, 2002, Stage de DEA en Informatique.
- [42] D. NIEWBOURG. *Grid et Metacomputing dans ProActive*. rapport technique, Université de Nice Sophia-Antipolis, septembre, 2002, Stage DESS Télécommunications.
- [43] H. SAYBASILI. *Un Modèle de Composant Hiérarchique pour ProActive et les principes de son implémentation avec Fractal*. rapport technique, Université de Nice - Sophia Antipolis, septembre, 2002, Stage DEA Réseaux et Systèmes Distribués.
- [44] S. SZENTES. *Implémentations efficaces de la Réduction Statique*. rapport technique, Ecole des Mines de Nantes, août, 2002, Stage ingénieur troisième année.
- [45] D. A. VIET. *Modèle Comportemental pour Calculs d'Objets Répartis et Mobiles*. rapport technique, Université Pierre et Marie Curie, LIP6, septembre, 2002, Stage de DEA de Systèmes Informatiques Répartis.
- [46] S. ZIANE. *Passerelle entre les formalismes (Schema) du W3C et la programmation par Objets*. rapport technique, Univ. Versailles, juin, 2002, Stage DEA MISI.