Project Number:	IST-2001-33100
Project Acronym:	PROFUNDIS
Title :	Proofs of Functionality for Mobile Distributed Systems

Final Report

Preparation date:	June 29, 2005
Classification:	Public
Contract start date	1 January 2002
Duration:	3 years
Project co-ordinator:	Joachim Parrow
Partners:	Univ. Uppsala, Sweden Univ. Pisa, Italy INRIA, France Univ. FFCT, Portugal

Contents

PROFUNDIS:
Final Report

June 29, 2005

1 Executive Summary

PROFUNDIS is a FET GC project with the main goal to advance the state of the art of formal modelling and verification techniques to the point where key issues in mobile distributed systems can be treated rigorously and with considerable automatic support. The partners are Uppsala (co-ordinator), Lisbon, INRIA and Pisa. The project started 1/1/2002 and terminated 30/4/2005.

We have developed comprehensive automata-like models that supports effective techniques to specify and verify properties of network applications. We have designed, implemented and tested several toolkits to perform such analyses automatically or with considerable automatic support. The toolkits are integrated in the Profundis Verification Environment, thereby making semantic-based verification available as Web services, using standards such as WSDL and SOAP. We have developed a spatial logic to describe and verify properties of concurrent systems, and explored its expressiveness and model checking algorithms. We have developed new type systems for specific areas and applied them to advanced programming constructs and realistic languages.

All work has been carried out in the context of FET GC. Collaboration with other GC projects has been pronounced. A distinguishing feature of PROFUNDIS is the emphasis on the experimental development of verification techniques.

PROFUNDIS conducts research of a foundational nature with potentially far reaching implications, though none that will immediately give rise to a new invention or start-up company. Our effort, directed at making mobile open systems more trustworthy, contributes to meet the social objectives of the Community of improving the quality of life and safety. It has also had a lasting impact on the PhD training programmes.

During the whole project the main principle has been to publish all results in scientific peer-reviewed journals or conferences. In this respect the project has been a tremendous success with 82 peer-reviewed published articles, 4 invited articles and a further 43 papers many of which are submitted and considered for publication, thus greatly exceeding our original goals for dissemination.

2 Project Objective

The objective of PROFUNDIS was to advance the state of the art of formal modelling and verification techniques to the point where key issues in mobile distributed systems, such as security protocols, authentication, access rights and resource management can be treated rigorously and with considerable automatic support. In particular we aimed to verify properties typical in so called open systems, where the behaviour of some parts (like intruders or adversaries) is unknowable, in extensible systems, where parts may be added or removed as the system executes, and in mobile systems where physical and logical connectivity between parts may change. We also aimed at implementing automatic and partly automatic analysis methods for ascertaining correct behaviour of such systems.

PROFUNDIS consisted of three technical Work Packages (WPs) outlined below. All sites contributed to all work packages to varying degrees.

WP1: Models

The overall goal of WP1 is to develop a comprehensive automata-like model that supports effective techniques to specify and verify properties of network applications. The research activities are centered around three tasks: The emphasis of tasks on automata with operations and substitutions (Task 1.1) and on Proof Techniques (Task 1.2) is on the development of automata-like model which support the handling of names in its full generality together with effective proof verification techniques. The task on prototype and case studies (Task 1.3) has been concerned with the prototype implementation of the verification toolkits suggested by the theoretical investigations, and with assessing the applicability of the verification methodologies.

WP2: Specifications

The goal of WP2 is to develop logics to support the specification and verification of spatial and behavioural properties of mobile concurrent systems and to develop verification tools for those logics. There are three tasks: one where spatial logics are proposed for several purposes, including logics for different models of mobile computation (Task 2.1); another dedicated to the study of the expressiveness of those logics (Task 2.2); and a third one to build verification tools, integrating theorem-proving and model-checking techniques (Task 2.3).

WP3: Types

The purpose of WP3 is to explore verification techniques for mobility based on types. More precisely, WP3 is divided into four tasks, whose objectives are: to develop new type systems to control interferences among processes and the resources used by the processes (Task 3.1); to integrate the type techniques with operational and logic techniques (Task 3.2); to investigate the robustness of the type techniques and their algorithmic definitions (Task 3.3); to assess the applicability of the techniques by means of case studies, and to implement some of the type algorithms and proof techniques (Task 3.4).

3 Methodologies

3.1 Models

Nominal calculi, process calculi with primitive mechanisms for local name generation, name exchange and scoping rules, have been successfully applied to specify and verify properties of global computing systems. Names provide a suitable abstraction to describe a variety of different computational phenomena such as mobility, localities, distributed object systems, security keys, session identifiers and son on. Several properties of security protocols have been naturally expressed through nominal calculi enriched with cryptographic primitives. Finally, nominal calculi also provide a basic programming model that has been incorporated in novel programming languages and workflow languages for Web Service coordination. Verification via semantic equivalence provides a well established framework to reason about the behaviour of systems specified using nominal calculi. In this approach, checking behavioural properties is reduced to the problem of contrasting two system abstractions in order to determine whether their behaviours coincide with respect to a suitable notion of semantic equivalence. However, in the case of nominal calculi verification via semantic equivalence is intrinsically difficult. Indeed, when an unbound number of new names can be generated during execution, models of nominal calculi (e.g. labelled transition systems) tend to be infinite even in the simplest cases unless explicit mechanisms are introduced to deal with names.

History Dependent automata (HD-automata in brief) have been proposed as a basic syntax-free approach for nominal calculi. Since states of HD-automata are just items characterized by their properties regardless of their syntactic details they can be used to handle a variety of calculi and semantics. HD-automata have been modelled by exploiting co-algebraic techniques. The theoretical results guarantee the existence of the minimal realization. The minimal automata are computed using a partition refinement algorithm and they have a very important practical fall-out: for instance, the problem of deciding bisimilarity is reduced to the problem of computing the minimal transition automaton. Moreover, the minimal automaton is indistinguishable from the original one with respect to many behavioural properties (e.g., bisimilarity) and properties expressed in most modal or temporal logics. Theoretical results have been established about HD automata modelled as coalgebras. Also coalgebraic models (not necessarily finite state) of mobile calculi have been defined, equipped with operations of parallel composition and restriction.

A substantial effort has been devoted to develop effective verification techniques for security protocols. The specification and validation of security protocols often require the explicit handling of data (cryptographic keys). In the project symbolic verification techniques have been developed and applied to security protocols.

The theoretical results have provided the firm foundations needed for the experimental development, and they have driven the design and the prototype implementation. Tool prototypes have been developed for HD automata minimization wrt. bisimilarity. Verification toolkits exploiting symbolic techniques have been designed, implemented and tested. The distinguished and innovative feature of the Profundis Verification environment, called *Profundis WEB*, or *PWeb*, is the idea of viewing the environment as a distributed infrastructure exploited as a *service distributor*. In the Profundis WEB each verification toolkit has an interface which is Internet accessible through standard network protocols and which describes the interaction capabilities of the verification toolkit.

Worldwide research activities in the field has been active during the past few years. Quite relevant to the foundations of HD automata are the the work of Gabbay-Pitts, and Cattani-Sewell. The main advantage of History-Dependent automata with respect to similar approaches reside of the fact they that they provide a notion of minimal realization. Indeed, the distinguished contribution of PROFUNDIS consists on the emphasis on experimentation: verification services exploiting the theories developed inside the project have been designed, implemented and tested through several case studies.

3.2 Specifications

Formal reasoning about spatial aspects of distributed systems have been somewhat neglected by the computer science community until very recently. Examples of spatial properties of distributed systems include such diverse aspects as connectivity, unique handling, invariants of the communication topology and routing, location-dependent access rights to resources, dynamically created objects and references, and security and secrecy features, among others. Other useful applications of spatial reasoning are in the areas of semi-structured data and of pointer structures in imperative languages. In the project we concentrated mainly on concurrency, distribution and mobility, but comparisons and results with impact on other areas were also obtained. Our approach to address spatial and behavioural issues in an integrated way consisted in providing logics where such properties could be stated, systems specified and proofs of correctness given. New logics require a study of their strengths and limitations. One of the goals of the project was to extend known results from the ambient logic and from logics for semi-structured data to the logics proposed in the project. Examples of important expressivity questions are the decidability of the validity and satisfiability problems, on which model-checking depends, and the study of the relative expressive power of subclasses of spatial operators. Many expressivity results were obtained, requiring new techniques specially introduced for that purpose. A major goal of the project was the construction of tools to assist in the reasoning process: a theorem-proving tool and a model-checker (no such tools existed for combined spatial/behavioural reasoning before the project started). We expected to produce the theoretical framework, develop reasoning tool prototypes and check the tools against case studies. A sequent calculus-based theorem-proving framework was developed, some proof techniques proposed and a formal proof of correctness of a nontrivial peer-to-peer algorithm was given. A spatial logic model-checking tool was defined and implemented.

3.3 Types

New type systems have been introduced that, we think, significantly enlarge the collection of properties of mobile processes that can be handled with types. These include types for for static control of migrating processes and of resource consumption, and types for specifying the state transitions of a protocol (session types).

For other properties in which types alone seemed to be insufficient, we have developed techniques and models that combine ideas from types with ideas from other fields, in particular logical relations and term rewriting. We have also studied the integration of types with algebraic techniques.

A substantial effort has also gone into the transfer to types developed in foundational calculi to other languages, closer to realistic programming languages. This includes, for instance, information flow type systems that enforce non-interference for realistic assembly languages (JVM-like), and session types for multi-threaded functional languages.

Finally, to study of the impact of type systems on implementations, we have designed abstract machines for the execution of calculi of distributed processes (Safe Ambients), in which types are exploited to improve the efficiency.

Worldwide, research in the field of type systems for mobile processes has been active during the past few years, and a substantial part of it has been carried out within the Global Computing initiative. Due to our connections with other projects, we have been in close contact with the researchers who are active in the area. Some of the work we have done in PROFUNDIS is similar to work carried out outside it. The aspects in which PROFUNDIS has been more original are to integrate the type techniques with other techniques (in particular logic and algebraic), to experiment with our type techniques in real programming languages (as opposed to foundational calculi), and to put emphasis on techniques that can be implemented and possibly included in automated tools.

4 Project Results

4.1 Models

The formulation of History-Dependent (HD) automata as the basic model for nominal calculi has significantly progressed. HD-automata have been shown to provide an effective model suitable for finite state verification of finite control processes. Different versions of HD automata have been defined. The simplest version can be easily translated to ordinary automata, but possibly with a larger number of states. In a second version, the states are equipped with name symmetries which further reduce the size of the automata. A third version handles the symbolic semantics of nominal calculi, where inputs are represented as variables which are instantiated only when needed. Furthermore, a theory based on coalgebras has been developed for these classes of HD-automata. Moreover, the coalgebraic formulation of the partition-refinement minimization algorithm has directly suggested the software architecture of a verification toolkit. These results have extended the applicability of the HD-automata approach to nominal calculi and guarantee the existence of the minimal automaton within the same bisimilarity class.

A substantial effort has also gone into the development of abstract models for nominal calculi by considering suitable functors categories (e.g. presheaves over subcategories of *Rel*). These models of nominal calculi clarify in an abstract setting the meaning of names and, moreover, and provide general constructions to relate the different abstract models of nominal calculi.

4.2 Tools

We designed, implemented and tested the following toolkits.

MIHDA performs minimisation of History-Dependent (HD) automata. MIHDA has been exploited to perform finite state verification of π-calculus specifications.

ASPASYA relies on a symbolic technique to model check properties of cryptographic protocols. Security properties are expressed via a logic that predicates over data exchanged in the protocol and observed by an intruder in the execution environment, and also over the "presumed" identities of the protocol principals. ASPASYA allows varying the intruder's knowledge, the portion of the state space to be explored, and the specification of implicit assumptions that are very frequent in security. The user can opportunely mix those three ingredients for checking the correctness of the protocol without modifying neither the protocol specification nor the specification of the desired properties.

TRUST relies on an exact symbolic reduction method, combined with several symbolic techniques aiming at reducing the number of interleaving that have to be considered. Authentication and secrecy properties are specified using the correspondence assertions and whenever an error is found an intruder attacking the protocol is given.

STA implements symbolic execution of cryptographic protocols. A successful attack is reported in the form of an execution trace that violates the specified property expressed in terms of correspondence assertions.

SMC Is a prototype version of a tool for model-checking distributed systems modelled in the pi-calculus against a version of a spatial logic.

The toolkits above are integrated inside the Profundis Verification Environment, called the *Profundis WEB* or *PWeb*. The distinguished and innovative feature of the *PWeb* is the idea of viewing the environment as a distributed infrastructure exploited as a *service distributor*. In the Profundis WEB each verification toolkit has an interface which is network accessible through standard network protocols and which describes the interaction capabilities of the verification toolkit. One main idea of our approach is to make semantic-based verification toolkits available as Web services, using standards such as WSDL and SOAP. Another main idea is to establish service directories for publishing such Web services. Verification web services plus service directories thus provide a platform for distributed, deeply integrated and therefore coordinated verification activities.

In the PWeb infrastructure a verification session takes the form of *service coordination* describing the rules a set of verification services have to follow to achieve a certain goal. In other words, the coordination rules are used to specify how the sub-tasks within any verification run are to be carried out, in which order and which are the different toolkits involved. Moreover, there are mechanisms for assigning verification sub-task to the specialized toolkits that are most appropriate to solve them. Beyond the current prototype implementation,we envisage the important role that will be played by PWeb service coordination. Indeed, service coordination provides several benefits:

- *Model-based verification.* The coordination rules impose constraints on the execution flow of the verification session thus enabling a *model-based* verification methodology where several descriptions are manipulated together. Notice that there is a sound conceptual basis for model-based verification since verification toolkits provide an implementation of well understood semantic theories.
- *Modularity.* The verification of the properties of a large software system can be reduced to the verification of properties over subsystems of manageable complexity: the coordination rules reflect the semantic modularity of system specifications.
- *Flexibility.* The choice of the verification toolkits involved in the verification session may depend on the specific verification requirements.

The projects' deliverables reported on our experience in exploiting the facilities of the PWeb infrastructure in the verification of properties of distributed systems specified in some dialect of the π-calculus.

Outside the PWeb we have developed an abstract machine for a distributed process calculus (Safe Ambients), that shows that certain controls of interferences guaranteed by some of the types mentioned above are also useful at the level of the implementations of languages. This machine has then been further studied and developed. We have defined an optimised abstract machine and proves its correctness w.r.t. the original machine. The improved machine is made more efficient by adapting some standard algorithms in distributed programming, such as forwarder chains contraction using Tarjan sets.

We have also formalized in the Isabelle/HOL proof assistant a type system for a concurrent language with scheduling. This contribution represents, to the best of our knowledge, the first machine-checked account of non-interference for a concurrent language.

4.3 Specifications

A spatial logic to describe and verify properties of concurrent systems specified in the π-calculus was developed. The logic allows the specification of spatial and behavioural properties by induction and co-induction and includes freshness and hidden name quantifiers. An associated sequent calculus-based proof-system for the logic, that combines good proof-theoretic properties (e.g., cut-elimination) and direct applicability to concurrency, was proposed. The correctness of a non-trivial peer-to-peer algorithm was formalised in the logic and formally proved.

New logics, TL and SL, to describe and query semi-structured data represented by multitrees, which can embed XML Schema as a plain subset, where introduced. Decision procedures for satisfaction and model-checking the query languages logics TL and SL against formal representations of semi-structured documents were provided.

Expressiveness and minimality results have been obtained for spatial logics, showing that they can be expressed within a core, calculus-independent, spatial logic. In particular, it has been established that these spatial logics can express potentiality of interaction as they appear in the Hennessy-Milner logic. Minimality has also been addressed for the static Ambient Logic with operator of name revelation and fresh name quantification, in this context it was proved that the spatial adjunct operator can be eliminated in behalf of the remaining connectives without losing expressive power.

Tree automata were applied to obtain several complexity results related to XML schema, in a tree logic that can be seen as a form of spatial logic. This approach was also applied to derive the decidability of the quantifier-free, static fragment of ambient logic, with composition adjunct and iteration, which corresponds to a kind of regular expression language for semistructured data.

A spatial logic for systems specified in the synchronous π-calculus with recursion, based on a small set of behavioural and spatial observations, was proposed. Expressiveness results for this logic were studied and it was shown that model-checking in this logic is decidable for a useful class of processes that includes the finite-control fragment of the π-calculus. Later, extensions to the mechanisms for coping with recursive properties (both inductive and co-inductive) were devised.

A somewhat unexpected result is that the validity problem for the simplest dynamic spatial logics is undecidable, which entails the undecidability of model-checking of spatial logics with contextual operators (composition adjunct). This research contributed to raise the question of finding expressive and tractable forms of contextual reasoning inspired by the composition adjunct, extending those already provided by the decidable behavioural-spatial logics also investigated in the project.

The comparison between the spatial logics for concurrency and more standard logics has led to the definition of a spatial logic for the π-calculus that is *extensional*, in the sense that the induced logical equivalence coincides with behavioural equivalence on processes. This result shows which subset of spatial logics shares the same separative power as the Hennessy-Milner logic.

Separation logic and a classical fragment of it have been compared and shown to be equally expressive.

The quantifier-free, static fragment of ambient logic, with composition adjunct and iteration, which corresponds to a kind of regular expression language for semistructured data, was proved to be decidable.

A framework for specifying constraint problems with an open number of variables was proposed and the decidability of the satisfiability for various families of these problems has been studied.

A CCS-like calculus has been defined for which the observations have been enriched with spatial observations, giving rise to models of spatial logic.

A logical formalization of the secure composition of web services was made.

A simple spatial logic that supports specification of quality of service (QoS) properties of applications has been introduced. The evaluation of a formula in the spatial logic is a value of a suitable algebraic structure, a c-semiring, representing the QoS level of the formula and not just a boolean value expressing whether or not the formula holds. Applications of this spatial logic with quantitative information to web services and wireless systems have been considered.

4.4 Types

Some of our works tackle specific problems of distributed mobile systems, in the π-calculus or distributed versions of it, and propose solutions using type systems. These problems include: interference (classification of the possible interferences among processes, use of types to rule out certain classes of interference); control of the migration of processes and of the access to local resources that a migrated process is granted; resource allocation (specifically, the control of the number of processes that can be physically present on a given site at the same time); message-deliverability (the fact that every emitted message has a chance of being received); the specification of guarantees on the services that a process can offer, in cases where such services can evolve and change during the execution of the processes; termination (the fact that a process never reaches a divergence, that is, a point in which an infinite sequence of internal steps can be produced); access control (using types to control remote communication, process migration and channel creation, in order to guarantee compliance w.r.t. a control policy on migration of code).

The work on resource allocation, originally designed for a specific calculus of distributed processes (Mobile Ambients), has then been generalized and developed, obtaining a generalisation that handles flexible resource control policies. Other work related to this develops new methods to statically bound the resources needed for the execution of systems of concurrent, interactive threads.

A new type system has also been proposed for MetaKlaim, the distributed version of a coordination language. Here, the security guarantees on a given process are maintained even when the process interact with untrusted components. To ensure this, the type checks are partly performed statically, and are partly enforced dynamically.

All the above mentioned papers introduce novel type systems. The only exceptions are the works on termination where, to get a simpler solution, the desired property is ensured by means of a combination of known types and of syntactic conditions, and adapting the well-known technique of logical relations from functional languages to concurrency.

Types have also been used to enhance algebraic and operational techniques for untyped mobile processes.

4.5 Practicality and scalability

A substantial amount of work has gone into trying to apply types to advanced programming constructs. On this topic we have had three strands of work. One such strand has focused on information flow type systems that enforce non-interference. Here we have designed such type systems for realistic assembly languages (JVM-like), and we have introduced compilers from high-level programming languages to our low-level languages and showed that the compilers preserve information flow types. We have also studied logical formulations of non-interference, which allow a more precise analysis of programs than that allowed by type systems, and amenable to interactive or automated verification techniques.

The second strand of work has focused on session types. A session type, associated with a communication channel, can specify the state transitions of a protocol and also the data types of messages associated with transitions; thus typechecking can verify both correctness of individual messages and correctness of sequences of transitions. We have designed type systems of this kind for a multi-threaded functional language with side-effecting input/output operations. And we have showed how session types allow not only high level specifications of complex interactions, but also the definition of powerful interoperability tests at the protocol level, namely compatibility and substitutability of components.

The third strand of work has used static techniques based on control flow analysis, initially inspired by type system works. In the context of access control policies based on stack inspection and dynamic security policies, as can be found in security-aware languages such as Java or C♯, a method based on *control flow graphs* has been introduced to validate some program transformations without compromising security. The use of control flow graphs has turned out to be an appropriate alternative to more traditional type systems for the programs being analysed. On this topic, a case analysis has also been carried out. It is about a specific optimization technique, namely method inlining.

A different line of research has addressed the problem of developing models and reasoning techniques for the analysis of security protocols. Nominal calculi provide the right level of abstraction to model security protocols. However, the development of reasoning principles requires explicit handling of data to model encryption and decryption machineries. To this purpose, we introduced novel symbolic verification techniques for security protocols. Symbolic techniques reduce the search space in the verification of cryptographic protocols. We developed several methods for automatic analysis of security protocols based on a symbolic operational semantics that relies on unification and provides finite and effective protocol models. Then, we give proof techniques to carry out protocols analysis directly on the symbolic model. These techniques has been actually implemented and have lead to the development of some verification environments.

In Pisa we have established a collaboration with Telecom Italia Lab. We have exploited some of the toolkits of the PWeb to verify functional and non-functional properties of the PARLAY-X infrastructure.

5 Comparisons to original project objectives

Most of our activities have progressed according to plan. We here only mention the discrepancies.

The activities on Task 1.1. (Automata with operations and substitutions) were expected to be completed by the end of the second year. However, we modified our initial plans and we continued them till the end of the project.

In WP2 we expected to do some work on high-level extensions to the logic, but soon recognized that the topic required further foundational work on logics and some experience with case studies. Also, in view of the results obtained we decided to accord higher priority to the model-checker than to the theorem-prover.

In WP3, there has been more work than expected on experimenting with advanced programming constructs. On the other hand, we have had less work than planned on a few technical aspects, notably type declassification, spacial types, and the introduction of types into the Profundis tools. The reasons for this are both technical (unexpected technical difficulties), and decisional (in a few occasions it has been decided to give priority to other goals).

6 Relationships with other research activities

Verification of global computing systems is an important issue of other FET GC Projects. The problem of formal certification of properties has been addressed by a number of FET GC projects exploiting different techniques and methodologies, Darts, Myths, Socks, Mikado, Pepito, and Agile. Also, types is an important topic in a number of FET GC projects: Darts, Myths, MSR, Mikado, Pepito, Agile. Within PROFUNDIS we closely followed what happened in these topics. We are aware of what have been developed in the other projects and we met researchers of other GC projects in international conferences and workshops. Moreover, cross-fertilization takes advantage of the fact that at some sites (FFCT, INRIA and PISA) other research groups, closely linked to those active in PROFUNDIS, are involved in different research projects about related topics. We can mention that in Pisa four research groups are active on FET GC projects AGILE, DEGAS, PROFUNDIS and SOCKS. These collaborations have given rise to a number of works joint between PROFUNDIS researchers and researchers from other sites.

The distinguished feature of PROFUNDIS stands in its emphasis on the experimental development of verification techniques. The verification methods developed inside the projects have led, whenever possible, to the design and implementation of verification toolkits. Another distinguished feature of PROFUNDIS stands on the idea of integrating complementary proof principles. In the area of types, the unique PROFUNDIS contribution is in the emphasis on the use of types for verification, where the verification should be mechanically carried out whenever possible, and where the techniques may combine methods based on types, operational semantics and logics.

Quite relevant to the foundations of HD automata are the the work of Gabbay-Pitts, and Cattani-Sewell. Other relevant work outside GC projects is the work of Abadi, Gordon and Fournet on the exploitation of symbolic verification techniques to model web service security. Finally, on the toolkit development, the main relevant research activity outside the FET GC projects is the work at Stony Brook (Smolka, Cleaveland) on XMC/MMC (a model checker to verify properties of mobile systems specified in the π-calculus). On the toolkit integration, a different approach has been exploited by the *Electronic Tool Integration Platform* (ETI) initiative (Margaria, Steffen). ETI is a web-based infrastructure for the interactive experimentation of verification toolkits. Contacts with these groups have been already established.

On the theme of WP3 – types for mobile processes – all the main international actors are within the GC projects. Possible exceptions are Kobayashi's group in Tokyo, Honda and Yoshida in London. Other relevant work outside the GC projects, but that does not specifically target mobile processes, is that on types for control of resources; in particular, the work by Karl Crary and Stephanie Weirich in the USA. We are aware of all this work, that has indeed affected some of our research.

7 Potential Impact

A crucial issue of software technologies for global computing is the ability of formally certifying properties of applications. HD-automata provide an intermediate, syntax independent, format to represent a variety of nominal calculi equipped with mobility and distribution primitives. An important point is that for a wide class of processes and observational semantics the resulting HD-automata are finite state. Furthermore, in several interesting cases it is possible to construct for each HD-automaton its minimal realization. As a consequence, one can *re-use* both verification principles and automatic methods specifically developed HD-automata just by providing suitable translations into the HD-automata settings. At a more abstract level, the idea of looking at abstract models of nominal calculi in terms of functor categories provides general constructions to relate the different models of nominal calculi and allows one to transfer techniques and constructions from one model to the other.

Contract-signing protocols allow two or more parties to exchange signatures fairly, so that no party receives a signed contract unless all do. Proving properties of contract signing protocols is a difficult task. Only recently, few methods have been used to analyze contract-signing protocols and either find errors or suggest their absence. The symbolic techniques developed inside PROFUNDIS allow for reasoning about authentication protocols. Generalization of these techniques can be exploited to deal with contract-signing protocols.

The main idea of the PWeb approach is to make semantic-based verification toolkits available as Web services, and to establish directories for publishing such Web services. This facilitates the easy integration and maintenance of heterogeneous verification toolkits having complementary functionalities. We argue that service-based approaches have the potential to tackle the tool integration issues of the software engineering process.

There is a growing interest among the scientific community in the applications of spatial logic, and the results obtained in PROFUNDIS represent a great advance in the research directions investigated in the project: on logics, their expressiveness properties and the reasoning tool. It should be emphasized that the Spatial Logic Model-Checker is the first of its kind to fully integrate spatial and behavioural properties of processes, and since it has been made available on the web a substantial number of people has visited the site and downloaded the model-checker.

We have developed type-based techniques that can be used to ensure important properties of mobile applications, including security properties, and we have studied their impact on implementations and on realistic programming constructs. Further, since we believe that single techniques are not sufficient for realistic applications, we have studied integrations of types with other techniques, notably operational, logical and algebraic techniques. This has a potential impact since it means more realistic examples can be treated with our techniques.

8 European dimension

This project addresses the central issues of computationally feasible verification methodologies and security that proved their value in certain areas of computer science, but whose study is much more difficult and has barely begun for mobile computing. The results obtained will contribute to the design of reliable and robust systems with high quality of services and to the development of technologies that enhance trust and confidence. It is not expected that the results will be immediately used by the industry at the end of the project, but we will take steps to train young researchers and to transfer the results to industry. The European industries can gain a crucial competitive advantage by adopting verification technologies that improve the quality of their products. It is foreseeable that start-up companies can be created specifically for that purpose.

The range of problems dealt with in the project called for diverse competences that cannot be found in a single place. Models, logics and types are the different perspectives that are put together to contribute to the overall goal of developing computationally feasible verification methods. No single approach can solve this problem by itself. European universities have a long tradition and a leading position in the fields of calculi for mobility, verification, theorem provers and model checkers. All participants bring their expertise in some of the above fields (which some have pioneered as well), but none could find the complementary expertise at national level. We are confident that the added value of our collaboration in the project has allowed us to obtain results beyond what would be achieved on an individual basis.

The field of mobile distributed computing is relatively young but promises to have a strong positive impact in most aspects of our lives. To realize this promise, the field must prove itself in the commercial world, which in turn requires the prior development of appropriate theories, methodologies, techniques and tools, including those of the kind proposed in the project. An ill-founded subject can undermine the confidence of the economic agents and the public alike in the supporting technology, with undesired consequences to its future development and commercial exploitation. Our effort, directed at making mobile open systems more trustworthy, contributes to meet the social objectives of the Community of improving the quality of life and safety.

9 Future outlook

The groups intend to build on the results achieved during the PROFUNDIS project. In particular, we will continue the work on developing theories and effecting verification techniques for global computing systems, on the one hand looking for more applied targets, on the other hand trying to transfer the experience and knowledge acquired onto other forms of software systems . This will happen partly trough the participation in the GC2 program.

The results of PROFUNDIS will have an impact on teaching all levels, but mainly at the postgraduate level, in particular in the supervision of of PhD theses.

Further exploitation will improve our tools, build others with the methodologies already developed in the project and test them with new case studies, along with the necessary backing of theoretical work. Also, we are involved in several research projects where we apply the results obtained in PROFUNDIS, both at the national and at the international level

10 PROFUNDIS Project Achievements fiche

1. Scientific and technological achievements of the project (and why are they so ?)

Question 1.1. Which is the 'Breakthrough' or 'real' innovation achieved in the considered period

4	1. Verification as a web service. A way to interconnect distributed verification tools. 2. Verification techniques based on types and their integration with logical and algebraic techniques. 3. Models and Verification Techniques for Mobility and Security. 4. Logics and verification techniques for spatial and behavioural properties of distributed systems.

2. Impact on Science and Technology: Scientific Publications in scientific magazines

Question 2.1. Scientific or technical publications on reviewed journals and conferences

82	See Appendix

Question 2.2. Scientific or technical publications on non-reviewed journals and conferences

43	See Appendix

Question 2.3. Invited papers published in scientific or technical journal or conference.

4	See Appendix

3. Impact on Innovation and Micro-economy

A - Patents

Question 3.1. Patents filed and pending

0	

Question 3.2. Patents awarded

0	

Question 3.3. Patents sold

0	

B - Start-ups

Question 3.4. Creation of start-up

0	

Question 3.5. Creation of new department of research (ie: organisational change)

0	

C - Technology transfer of project's results

Question 3.6. Collaboration/ partnership with a company

2	Partner Pisa has established a collaboration with Telecom Italia Lab (TILAB). We have exploited some of the toolkits of the PWeb to verify functional and non-functional properties of the PARLAY-X infrastructure. Uppsala has established a research collaboration with IBM Research, TJ Watson Lab, on expressiveness in mobile calculi.

4. Other effects

A - Participation to Conferences/Symposium/Workshops or other dissemination events

Question 4.1. Active participation to Conferences in EU Member states, Candidate countries / NAS. (specify if one partner or "collaborative" between partners)

1	Davide Sangiorgi has organised the Symposium on Trustworthy Global Computing (TGC), Edinburgh, UK, April 7-9, 2005. The symposium was part of the European Joint Conferences on Theory and Practice of Software (ETAPS 05)

Question 4.2. Active participation to Conferences outside the above countries (specify if one partner or "collaborative" between partners)

1	F. Valencia (Uppsala) was workshop chair and organizer of COLOPS (Constraint and Logic Programming in Security), a satellite event of ICLP'03, Mumbai, India.

B - Training effect

Question 4.3. Number of PhD students hired for project's completion

12	

C - Public Visibility

Question 4.4. Media appearances and general publications (articles, press releases, etc.)

0	

Question 4.5. Web-pages created or other web-site links related to the project

9	1. http://www.it.uu.se/profundis is the project main web page 2. http://jordie.di.unipi.it:8080/pweb is a web site for the distributed PROFUNDIS tools. 3. http://www.cmi.univ-mrs.fr/ vvanacke/trust.html is a web site for the TRUST tool. 4. http://jordie.di.unipi.it:8080/mihda is the web site of the HD-Reducer 5. http://rep1.iei.pi.cnr.it/projects/JACK/hal.html is the web site for the HAL tool. 6. http://www.it.uu.se/research/group/mobility/mwb is the web site for the MWB tool. 7. http://www.dsi.unifi.it/ boreale/tool.html is the web site of the STA tool. 8. http://www-ctp.di.fct.unl.pt/SLMC is the web site of the Spatial Model Checker (SLMC) tool. 9. http://perso.ens-lyon.fr/damien.pous/gcpan is a web site for the the abstract machine for the execution of Safe Ambients.

Question 4.6. Video produced or other dissemination material

0	

Question 4.7. Key pictures of results

0	

D - Spill-over effects

Question 4.8. Any spill-over to national programs

YES	Italian Ministry of Research Project *Architetture Software ad Alta Qualità di Servizio per Global Computing su Cooperative Wide Area Networks* The French initiative ACI Nouvelles Interfaces des Mathémathiques Geocal The French initiative AS CNRS *"Méthodes formelles pour la mobilité"*, *"Sécurité informatique"*, ACI Cryptologie VERNAM, ACI Cryptologie AZURCRYPT, ACI Sécurité informatique CRISS (head), and ACI Securité informatique Rossignol. The Portuguese project MIMO, financed by Fundação para a Ciência e a Tecnologia. The Swedish projects *Semantics of Programming Languages* and *Analysis of security properties* funded by the Swedish Research Council. The Portuguese project POLY, funded by the Fundao para a Cincia e Tecnologia. The Portuguese project *Space-Time-Types: Behavioural and Spatial Type Systems*, funded by the Fundação para a Ciência e Tecnologia.

Question 4.9. Any spill-over to another part of EU IST Programme

YES	FET-GC and FET-GC2 (Project Sensoria)

Question 4.10. Are other team(s) involved in the same type of research as the one in your project?

YES	Several of the other projects involved in FET-GC University of Stony Brook, NY University of Dortmund, Germany Microsoft Research (UK) The Big Top Initiative, Microsoft (USA)

11 Publications

Reviewed Publications

[1] R. Amadio and W. Charatonik. On name generation and set-based analysis in Dolev-Yao model (extended abstract). In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

[2] R. Amadio, D. Lugiez, and V. Vanackere. On the symbolic reduction of processes with cryptographic functions. *Theoretical Computer Science*, 290(1):695–740, 2002.

[3] R. Amadio and C. Meyssonnier. On decidability of the control reachability problem in the asynchronous pi-calculus. *Nordic Journal of Computing*, 9(2):70–101, 2002.

[4] Roberto M. Amadio, Gérard Boudol, and Cédric Lhoussaine. On message deliverability and non-uniform receptivity. *Fundamenta Informaticae*, 53(2):105–129, 2002.

[5] Roberto M. Amadio and Silvano Dal Zilio. Resource Control for Synchronous Cooperative Threads. In *CONCUR 2004 – 15th International Conference on Concurrency Theory*, volume 3170 of *Lecture Notes in Computer Science*, pages 68–82. Springer-Verlag, August 2004.

[6] Michael Baldamus, Jesper Bengtson, Gianluigi Ferrari, and Roberto Raggi. Web services as a new approach to distributing and coordinating semantics-based verification toolkits. In *Proceedings of the First International Workshop on Web Services and Formal Methods (WSFM 2004)*, volume 105 of *ENTCS*, pages 11–20. Elsevier, February 2004.

[7] Michael Baldamus, Joachim Parrow, and Björn Victor. Spi calculus translated to pi-calculus preserving may-tests. In *Proceedings of LICS'04*, Turku, Finland, July 2004. IEEE, Computer Society Press.

[8] Michael Baldamus, Joachim Parrow, and Björn Victor. A fully abstract encoding of the π-calculus with data terms. In *Proceedings of ICALP'05*, LNCS. Springer, 2005.

[9] Giacomo Baldi, Andrea Bracciali, Gianluigi Ferrari, and Emilio Tuosto. A Coordination-based Methodology for Security Protocol Verification. In *WISP'04*, volume 121 of *ENCTS*. Elsevier, 2005.

[10] G. Barthe, A. Basu, and T. Rezk. Security Types Preserving Compilation. In *Proc. of VMCAI'04*, volume 2937 of *Lecture Notes in Computer Science*, pages 2–15. Springer Verlag, 2004.

[11] G. Barthe, A. Basu, and T. Rezk. Security types preserving compilation. *Journal of Computer Languages, Systems and Structures*, 2005. To appear.

[12] G. Barthe, P. D'Argenio, and T. Rezk. Secure Information Flow by Self-Composition. In R. Foccardi, editor, *Proceedings of CSFW'04*, pages 100–114. IEEE Press, 2004.

[13] G. Barthe and L. Prensa-Nieto. Formally verifying information flow type systems for concurrent and thread systems. In M. Backes, D. Basin, and M. Waidner, editors, *Proceedings of FMSE'04*, pages 13–22. ACM Press, 2004.

[14] G. Barthe and T. Rezk. Non-interference for a JVM-like language. In G. Morrisett and M. Fähndrich, editors, *Proceedings of TLDI'05*. ACM Press, 2005.

[15] M. Bartoletti, P. Degano, and G. Ferrari. Stack Inspection and Secure Program Transformations. *International Journal of Information Security*, 2(3-4):187–217, 2004.

[16] Massimo Bartoletti, Pierpaolo Degano, and Gianluigi Ferrari. Method inlining in presence of stack inspection. In *Workshop on Issues in the Theory of Security (WITS'04)*, 2004.

[17] Massimo Bartoletti, Pierpaolo Degano, and Gianluigi Ferrari. Program transformations under dynamic security policies. In *MEPHISTO Final Workshop*, volume 99 of *Electronic Notes in Computer Science*. Elsevier, 2004.

[18] Massimo Bartoletti, Pierpaolo Degano, and Gianluigi Ferrari. Enforcing secure service composition. In *IEEE Computer Security Foundation Workshop*, 2005.

[19] Massimo Bartoletti, Pierpaolo Degano, and Gianluigi Ferrari. History-based access control with local policies. In *FOSSACS 2005*, volume 3441 of *Lectures Notes in Computer Science*. Springer, 2005.

[20] Massimo Bartoletti, Pierpaolo Degano, and Gianluigi Ferrari. Policy framings for access control. In *Workshop on Issues in the Theory of Security (WITS'05)*, 2005.

[21] M. Boreale and M. Buscemi. A framework for the analysis of security protocols. In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002.

[22] M. Boreale and M. Buscemi. Symbolic analysis of crypto-protocols based on modular exponentiation. In *Proc. of MFCS 2003*, Lecture Notes in Computer Science 2747. Springer-Verlag, 2003. An extended version appears in Proc. of FCS'03.

[23] M. Boreale, M. Buscemi, and U. Montanari. D-fusion: a distinctive fusion calculus. In *Proc. of the Second Asian Symposium on Programming Languages and Systems (APLAS 2004)*, volume 3302 of *Lecture Notes in Computer Science*. Springer Verlag, 2004.

[24] Michele Boreale and Marzia Buscemi. A framework for the analysis of security protocols. *Theoretical Computer Science*, 2005. To appear.

[25] Michele Boreale, Marzia Buscemi, and Ugo Montanari. A general name binding mechanism. In *Proc. of Symposium on Trustworthy Global Computing (TGC '05)*, Lecture Notes in Computer Science. Springer Verlag, 2005. To appear.

[26] A. Bracciali, A. Brogi, G. Ferrari, and E. Tuosto. Security and dynamic composition of open systems. In *Proc. Int. Conference on Parallel and Distributed Techniques and applications*, USA, 2002. CSREA Press.

[27] A. Bracciali, A. Brogi, G. Ferrari, and E. Tuosto. Security and dynamic composition of web services. In *Proc. Int. Conference on Parallel and Distributed Techniques and applications*. CSREA Press, USA, 2002.

[28] M. Buscemi and U. Montanari. A first order coalgebraic model of pi-calculus early observational equivalence. In *Proc. CONCUR'02*, volume 2421 of *LNCS*. Springer Verlag, 2002. Full version in Technical Report TR-02-14, Dipartimento di Informatica, Universitá di Pisa, August 2002.

[29] L. Caires. Behavioral and spatial properties in a logic for the pi-calculus. In Igor Walukiwicz, editor, *Proc. of Foundations of Software Science and Computation Structures'2004*, Lecture Notes in Computer Science. Springer Verlag, 2004.

[30] L. Caires and L. Cardelli. A spatial logic for concurrency (part ii). In *CONCUR 2002: Concurrency Theory (13th International Conference)*, Berlin, 2002. Lecture Notes in Computer Science. Springer-Verlag. Also as Technical Report 3/2002/DI/PLM/FCTUNL.

[31] L. Caires and L. Cardelli. A Spatial Logic for Concurrency (Part I). *Information and Computation*, 186(2):194–235, 2003.

[32] L. Caires and L. Cardelli. A spatial logic for concurrency–II. *Theoretical Computer Science*, 322(3):517–565, 2004.

[33] L. Caires and E. Lozes. Elimination of quantifiers and undecidability in spatial logics for concurrency. In *Proc. of CONCUR'04*, number 3170 in Lecture Notes in Computer Science, pages 240–257. Springer Verlag, 2004.

[34] L. Caires and E. Lozes. Elimination of quantifiers and undecidability in spatial logics for concurrency. *Theoretical Computer Science*, 2005. Accepted for publication.

[35] Witold Charatonik, Silvano Dal Zilio, Andrew D. Gordon, Supratik Mukhopadhyay, and Jean-Marc Talbot. Model checking mobile ambients. *Theoretical Computer Science*, 308(1):277–331, November 2003.

[36] Silvano Dal Zilio and Denis Lugiez. XML schema, tree logic and sheaves automata. In *RTA 2003 – 14th International Conference on Rewriting Techniques and Applications*, volume 2706 of *Lecture Notes in Computer Science*, pages 246–263. Springer, June 2003.

[37] Silvano Dal Zilio and Denis Lugiez. A logic you can count on. In *POPL 2004 – 31st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 2004.

[38] S. Dantchev and F.D. Valencia. On infinite csp's. In *Proc. Third International CP'05 Workshop on Modelling and Reformulating CSP's*, 2005.

[39] Yuxin Deng and Catuscia Palamidessi. Axiomatizations for probabilistic finite-state behaviors. In *Proceedings of the 8th International Conference on Foundations of Software Science and Computation Structures*, volume 3441 of *Lecture Notes in Computer Science*. Springer, 2005.

[40] Yuxin Deng and Davide Sangiorgi. Ensuring termination by typability. In *Proceedings of the 3rd IFIP International Conference on Theoretical Computer Science*, pages 619–632. Kluwer, 2004.

[41] Yuxin Deng and Davide Sangiorgi. Towards an algebraic theory of typed mobile processes. In *Proceedings of the 31th International Colloquium on Automata, Languages and Programming*, volume 3142 of *Lecture Notes in Computer Science*, pages 445–456. Springer, 2004.

[42] Yuxin Deng and Davide Sangiorgi. Towards an algebraic theory of typed mobile processes. *Theoretical Computer Science*, 2005. To appear.

[43] Rocco DeNicola, Gianluigi Ferrari, Ugo Montanari, Rosario Pugliese, and Emilio Tuosto. A process calculus for qos-aware applications. In *Coordination 2005*, volume 3454 of *Lectures Notes in Computer Science*. Springer, 2005.

[44] M. Miculan F. Gadducci and U. Montanari. Some characterization results for permutation algebras. In *Workshop COMETA*, volume 104 of *ENTCS*. Elsevier, 2003.

[45] G. Ferrari, S. Gnesi, U. Montanari, and M. Pistore. A model checking verification environment for mobile processes. *ACM Transactions on Software Engineering and Methodologies (TOSEM)*, 12(4):440–473, 2004.

[46] G. Ferrari, E. Moggi, and R. Pugliese. Guardians for ambient based monitoring. In *Proc. Foundations of Wide Area Network Programming*, ENTCS. Elseviers, 2002.

[47] G. Ferrari, E. Moggi, and R. Pugliese. Metaklaim: A type safe multi-stage language for global computing. *Mathematical Structures in Computer Science*, 14(3):367–395, 2004.

[48] G. Ferrari, U. Montanari, and M. Pistore. Minimizing transition systems for name-passing calculi: A co-algbraic formulation. In *Proc. FOSSACS'02*, volume 2303 of *LNCS*. Springer Verlag, 2002.

[49] G. Ferrari, U. Montanari, R. Raggi, and E. Tuosto. From co-algebraic specifications to implementation: The mihda toolkit. In *First International Symposium on Formal Methods for Components and Objects (FMCO)*, Springer Lecture Notes in Computer Science. Springer, 2003.

[50] G. Ferrari, U. Montanari, and E. Tuosto. Coalgebraic minimization of hd-automa in a polymorphic λ-calculus. *Theoretical Computer Science*, 331(2-3):325–365, 2005.

[51] Gianluigi Ferrari, Stefania Gnesi, Ugo Montanari, Roberto Raggi, Gianluca Trentanni, and Emilio Tuosto. Verification on the WEB. In Juan Carlos Augusto and Ulrich Ultes-Nitsche, editors, *Verification and Validation of Enterprise Information Systems*, pages 72–74, Porto, Portugal, April 2004. INSTICC Press.

[52] Gianluigi Ferrari and Alberto Lluch-Lafuente. A Logic for Graphs with QoS. In *First International Workshop on Views On Designing Complex Architectures*, Electronic Notes in Computer Science, Bertinoro, Italy, September 2004. Elsevier.

[53] Gianluigi Ferrari, Ugo Montanari, and Emilio Tuosto. Verification on the web of name-passing process calculi. In *Workshop on Software Engineering Tools: Compatibility and Integration, Monterey Workshop Series*, LNCS, 2004. To appear.

[54] Gianluigi Ferrari, Ugo Montanari, and Emilio Tuosto. Model Checking for Nominal Calculi. In *FOSSACS*, volume 3441 of *LNCS*, Edinburgh, UK, 2005. Springer.

[55] Gianluigi Ferrari, Ugo Montanari, Emilio Tuosto, Björn Victor, and Kidane Yemane. Modelling fusion calculus using hd-automata. In *To appear in in First Conference on Algebra and Coalgebra in Computer Science CALCO'05*, LNCS. Springer, 2005.

[56] Neil Ghani, Kidane Yemane, and Björn Victor. Relationally staged computations in calculi of mobile processes. In *Proceedings of CMCS 2004 (7th International Workshop on Coalgebraic Methods in Computer Science)*, volume 106 of *ENTCS*, pages 105–120, Barcelona, Spain, 2004. Elsevier.

[57] P. Giambiagi, G. Schneider, and F. Valencia. On the expressiveness of ccs-like calculi. In *In Proceedings of FOSSACS'04*. LNCS, Springer-Verlag, 2004.

[58] D. Hirschkoff. An extensional spatial logic for mobile processes. In *Proc. of CONCUR'04*, volume 3170, pages 325–339. Springer Verlag, 2004.

[59] D. Hirschkoff, E. Lozes, and D. Sangiorgi. Separability, Expressiveness and Decidability in the Ambient Logic. In *Proc. of LICS'02*. IEEE Computer Society, 2002.

[60] D. Hirschkoff, E. Lozes, and D. Sangiorgi. Minimality results for the spatial logics. In *Proc. of FSTTCS'2003*, LNCS. Springer Verlag, 2003.

[61] D. Hirschkoff, D. Pous, and D. Sangiorgi. An Efficient Abstract Machine for Safe Ambients. In *Proc. of COORDINATION '05*, volume 3454 of *Lecture Notes in Computer Science*. Springer Verlag, 2005.

[62] I. Lanese and U. Montanari. Mapping fusion and synchronized hyperedge replacement into logic programming. *Theory and Practice of Logic Programming*, 2005. To appear.

[63] Francesca Levi and Davide Sangiorgi. Mobile safe ambients. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 25(1):1–69, January 2003.

[64] Alberto Lluch-Lafuente and Ugo Montanari. Quantitative μ-calculus and ctl defined over constraint semirings. In *QAPL (2004)*, volume 112 of *Electronic Notes in Computer Science*, pages 1–30. Elsevier, 2005.

[65] E. Lozes. Adjunct elimination in the static Ambient Logic. In *Proc. of EXPRESS'2003*, volume 96 of *Electronic Notes in Computer Science*, pages 51–72, 2003.

[66] E. Lozes. Separation logic preserves the expressive power of classical logic. In *Proc. Workshop Space'04*, 2004. Electronic publication.

[67] I. Scagnetto M. Miculan. A framework for typed hoas and semantics. In *Proc. of PPDP 2003, Uppsala*, Lecture Notes in Computer Science 2747. ACM, 2003.

[68] F. Martins and A. Ravara. Typing migration control in lsdpi. In *Proceedings of FCS'04*, volume 31, pages 1–12. Turku Centre for Computer Science, 2004.

[69] Marino Miculan and Kidane Yemane. A unifying model of variables and names. In *Proceedings of FOSSACS'05*, volume 3441 of *LNCS*. Springer, 2005.

[70] Ugo Montanari. Web services and models of computation. In *First International Workshop on Web Services and Formal Methods*, volume 105 of *Electronic Notes in Computer Science*. Elsevier, 2004.

[71] A. Ravara, A. G. Matos, V. T. Vasconcelos, and L. Lopes. Lexically scoped distribution: what you see is what you get. *Electronic Notes in Theoretical Computer Science*, 85(1), 2003. Presented at FGC'03.

[72] D. Sangiorgi. Types, or: Where's the difference between CCS and π? In *Proc. CONCUR '02*, volume 2421 of *LNCS*. Springer Verlag, 2002. Accompanying paper for an invited talk.

[73] D. Teller. Recovering resources in the pi-calculus. In *Proceedings of IFIP TCS 2004*. Kluwer, 2004.

[74] D. Teller, P. Zimmer, and D. Hirschkoff. Using Ambients to Control Resources. In *Proceedings of the 13th Int. Conf. in Concurrency Theory (CONCUR'02)*, volume 2421 of *LNCS*, pages 288–303. Springer Verlag, 2002.

[75] David Teller. Formalisms for mobile resource control. *Electronic Notes in Theoretical Computer Science*, 85(1), 2003. Presented at FGC'03.

[76] Emilio Tuosto. Tarzan: Communicating and Moving in Wireless Jungles. In A. Cerone and Alessandra Di Pierro, editors, *2nd Workshop on Quantitative Aspects of Programming Languages*, volume 112 of *Electronic Notes in Computer Science*, pages 77–94. Elsevier, 2004.

[77] Emilio Tuosto and Hugo T. Vieira. An Observational Model for Spatial Logics. In *First International Workshop on Views On Designing Complex Architectures*, ENTCS, Bertinoro, Italy, September 2004. Elsevier.

[78] F. Valencia. Timed concurrent constraint programming: Decidability results and their application to ltl. In *Proceedings of the Nineteenth International Conference on Logic Programming (ICLP 2003)*. LNCS, Springer-Verlag, 2003.

[79] A. Vallecillo, V. T. Vasconcelos, and A. Ravara. Typing the behavior of objects and components using session types. In Antonio Brogi and Jean-Marie Jacquet, editors, *Electronic Notes in Theoretical Computer Science*, volume 68. Elsevier Science Publishers, 2002. Presented at FOCLASA'02 - 1st International Workshop on Foundations of Coordination Languages and Software Architectures.

[80] V. Vanackère. The TRUST protocol analyser, automatic and efficient verification of cryptographic protocols. In *Verification Workshop - Verify02*, 2002.

[81] V. Vanackere. History-dependent scheduling for cryptographic processes. In *Proc. VMCAI 2004*, volume 2937 of *Lecture Notes in Computer Science*. Springer Verlag, 2004.

[82] Vasco T. Vasconcelos, António Ravara, and Simon Gay. Session types for functional multithreading. In *CONCUR'04*, volume 3170 of *Lecture Notes in Computer Science*, pages 497–511. Springer Verlag, 2004.

Invited Publications

[83] L.Bettini, V.Bono, R.De Nicola, G.Ferrari, D.Gorla, M.Loreti, E.Moggi, R.Pugliese, E.Tuosto, and B.Venneri. The klaim project: Theory and practice. In C.Priami, editor, *Global Computing: Programming Environments, Languages, Security and Analysis of Systems*, number 2874 in LNCS, 2003.

[84] Mogens Nielsen and Frank Valencia. Notes on timed ccp. In *4th Advanced Course on Petri Nets ICPN'03*. LNCS, Springer-Verlag, 2004.

[85] D. Teller, P. Zimmer, and D. Hirschkoff. Using Ambients to Control Resources (long version). *International Journal of Information Security*, 2(3-4):126–144, 2003.

[86] F. Valencia. Concurrency, time and constraints. In *Proceedings of the Nineteenth International Conference on Logic Programming (ICLP 2003)*, Lecture Notes in Computer Science. Springer Verlag, 2003.

Papers Submitted for Publication, Reports, Drafts

[87] Roberto M. Amadio, Gérard Boudol, and Cédric Lhoussaine. On message deliverability and non-uniform receptivity. Research report 05-2002, LIF, Marseille, France, May 2002.

[88] M. Baldamus. A Fully-Abstract Model for the Fusion Calculus. Preliminary Draft, 2003.

[89] M. Baldamus, R. Mayr, and G. Schneider. A Backward/Forward Strategy for Verifying Safety Properties of Infinite-State Systems. Technical Report 2003-065, Department of Information Technology, Uppsala University, Sweden, 2003.

[90] M. Baldamus, J. Parrow, and B. Victor. Spi Calculus Translated to π-Calculus Preserving May Testing. Technical Report 2003-063, Department of Information Technology, Uppsala University, Sweden, 2003.

[91] G. Baldi, A. Bracciali, G. Ferrari, and E. Tuosto. ASPASyA: an automated tool for security protocol analysis based on a symbolic approach. Submitted for publication, 2003.

[92] G. Barthe and L. Prensa Nieto. Formally Verifying Information Flow Type Systems for Concurrent and Thread Systems. manuscript.

[93] G. Barthe and T. Rezk. Secure Information Flow for the JVM. Manuscript.

[94] Jesper Bengtsson. Generic implementations of process calculi in Isabelle. In Paul Pettersson and Wang Yi, editors, *Proceedings of the 16th Nordic Workshop on Programming Theory*, number 2004-041 in IT Technical Reports, pages 74–78. Department of Information Technology, Uppsala University, October 2004.

[95] M. Boreale and M. Buscemi. *STA, a Tool for the Analysis of Cryptographic Protocols (Online version)*. Dipartimento di Sistemi ed Informatica, Università di Firenze, and Dipartimento di Informatica, Università di Pisa,, http://www.dsi.unifi.it/ boreale/tool.html, 2002.

[96] M. Buscemi. *Models and Security Verification of Mobile Systems.* PhD thesis, Dipartimento di Matematica, University of Neaples "Federico II", 2003.

[97] Marzia Buscemi and Ugo Montanari. A compositional coalgebraic model of monadic fusion calculus. Submitted for publication, 2005.

[98] Marzia Buscemi and Ugo Montanari. A congruence result for process calculi with structural axioms. Submitted for publication, 2005.

[99] Palamidessi C., Saraswat V., F. Valencia, and B. Victor. Linearity vs persistency in the pi-calculus. Manuscript.

[100] L. Caires. Model-checking of spatial properties in the pi-calculus. Research report 3, DI/FCT/UNL, December 2002.

[101] Y. Deng and D. Sangiorgi. Ensuring termination by typability. Submitted, 2005.

[102] G. Ferrari, U. Montanari, R. Raggi, and E. Tuosto. From coalgebraic specification to toolkit development. Technical Report TR-02-19, Technical Report, Dipartimento di Informatica Universita' di Pisa, 2002.

[103] G. Ferrari, U. Montanari, E. Tuosto, B. Victor, and K. Yemane. Modelling and minimising the fusion calculus using hd-automata. Technical report, University of Pisa, 2003.

[104] Fabio Gadducci, Marino Miculan, and Ugo Montari. About permutation algebras, (pre)sheaves and named sets. Submitted for publication, International Journal on Higher-Order and Symbolic Computation, 2004.

[105] S. Gay, V. T. Vasconcelos, and A. Ravara. Session types for inter-process communication. Preprint, Department of Computer Science, University of Lisbon, Campo Grande, Edifcio C5, 1749-016 Lisboa, Portugal, 2002.

[106] Simon Gay, Vasco T. Vasconcelos, and António Ravara. Session types for inter-process communication. TR 2003–133, Department of Computing, University of Glasgow, March 2003.

[107] P. Giannini, D. Sangiorgi, and A. Valente. A distributed abstract machine for Safe Ambients. Extended and refined version of a paper appeared in *ICALP'01*, 2002.

[108] E. Lozes. *Expressivité des logiques d'espaces.* PhD thesis, École Doctorale MathInf, ENS Lyon, 2004.

[109] Denis Lugiez and Silvano Dal Zilio. Multitrees Automata, Presburger's Constraints and Tree Logics. Research report 08-2002, LIF, Marseille, France, June 2002. http://www.lim.univ-mrs.fr/Rapports/08-2002-Lugiez-DalZilio.html.

[110] Denis Lugiez and Silvano Dal Zilio. XML Schema, Tree Logic and Sheaves Automata. Research report 4631, INRIA, November 2002. http://www.inria.fr/rrrt/rr-4631.html.

[111] F. Martins and A. Ravara. Controling migration in lsdpi. Preprint, Section of Computer Science, Department of Mathematics, Instituto Superior Técnico, 1049-001 Lisboa, Portugal, 2002.

[112] F. Martins and A. Ravara. Typing migration control in lsdpi. Preprint, CLC, Department of Mathematics, Instituto Superior Técnico, 1049-001 Lisboa, Portugal, 2003.

[113] L. Monteiro. Transition systems with spatial structures: A coalgebraic framework. Manuscript, 2002.

[114] L. Monteiro. A note on a noninterleaving model of concurrency based on transition systems with spatial structure. Technical Note, DI-FCT/UNL, 2003.

[115] L. Monteiro. A note on models for spatial logic based on transition systems with spatial structure. Technical Note, DI-FCT/UNL, 2003.

[116] D. Pous. GCPAN implementation. `http://perso.ens-lyon.fr/damien.pous/gcpan`, 2004.

[117] R. Raggi and E. Tuosto. *HD-Reducer (Online version)*. Dipartimento di Informatica, Universita' di Pisa, http://jordie.di.unipi.it:8080/mihda, 2002.

[118] A. Ravara, P. Resende, and V. Vasconcelos. An algebra of behavioural types. Preprint, Section of Computer Science, Department of Mathematics, Instituto Superior Técnico, 1049-001 Lisboa, Portugal, 2002.

[119] A. Ravara, P. Resende, and V. Vasconcelos. An algebra of behavioural types. Preprint, CLC, Department of Mathematics, Instituto Superior Técnico, 1049-001 Lisboa, Portugal, 2003.

[120] A. Ribeiro, L. Caires, and L. Monteiro. Verifying the Arrow Protocol in a Spatial Logic. Technical Report TR-DI/FCT/UNL-04/2004, Departamento de Informatica, FCT/UNL, 2004.

[121] C. Rueda and F. Valencia. Non-viability deductions in arc-consistency computation. Technical report, 2003.

[122] D. Teller. *Ressources limitées pour la mobilité: utilisation, réutilisation, garanties*. PhD thesis, École doctorale MathIF, ENS Lyon, 2004.

[123] E. Tuosto. *Non Functional Aspects of Wide area Network Programming*. PhD thesis, Dipartimento di Informatica, Univ. Pisa, 2003.

[124] E. Tuosto, B. Victor, and K. Yemane. Polyadic history-dependent automata for the fusion calculus. Technical Report 2003-062, Department of Information Technology, Uppsala University, December 2003.

[125] F. Valencia. On the decidability of timed CCP. Technical report, December 2003.

[126] Antonio Vallecillo, Vasco T. Vasconcelos, and António Ravara. Typing the behavior of software components using session types. Technical report, December 2004. Revised and extended version of Typing the Behavior of Objects and Components using Session Types. In Foclasa 2002, 1st International Workshop on Foundations of Coordination Languages and Software Architectures. Electronic Notes in Theoretical Computer Science, 68(3), 2002.

[127] V. Vanackère. *The TRUST protocol analyser.* Lab. Informatique de Marseille, http://www.cmi.univ-mrs.fr/ vvanacke/trust.html, 2002.

[128] Vincent Vanackere. *TRUST: un systeme de verification automatique de protocole cryptographique.* PhD thesis, Université de Provence, 2005.

[129] H. Vieira and L. Caires. Spatial Model Checker User's Manual. Technical report, Departamento de Informatica, FCT/UNL, 2003.