1 Major scientific results

1.1 Correctness of Computer Systems

Proving C or Java programs Our main activity is related to program verification. We mainly focus on the verification of behavioral specifications for programming languages such as C, Java and ML. We develop a tool “Why” which is a verification conditions generator: from an annotated program written in a small imperative language with Hoare logic-like specification, it generates conditions expressing the correctness and termination of the program. These verification conditions can be generated for several existing provers, including interactive proof assistants (Coq, PVS, HOL Light, Mizar) and automatic provers (Simplify, laRvEy, CVC Lite).

On top of this tool, we built a system called Krakatoa [3] which verifies Java source code annotated with the Java Modeling Language (JML). The main challenge was the design of a suitable model for the Java memory heap in order to tackle programs with possible aliases [6].

J.-C. Filliâtre and C. Marché designed a similar tool called Caduceus [3] for dealing with C programs. This tool was used by Th. Hubert and C. Marché [4] for proving a subtle algorithm due to Schorr & Waite for graphs traversal. J. Androuck [1] experimented on using this tool for formal verification of security properties of smart card embedded source code.

Timed automata Orsay and France Telecom R& D collaborated on the definition of a model of timed automata in Coq. It is integrated in the CALIFE platform, a general tool for specification and automatic or interactive verification of protocols. We are currently studying the quantitative analysis of behavior of protocols built on random choices.

Dependent types For his master work supervised by C. Paulin, M. Sozeau [4] designed a language with a subset type (in the spirit of the PVS language) which is convenient for programming with (a restricted class of) dependent types. He proposed a translation of a term in this language to a Coq term containing existential variables corresponding to type-checking conditions.

Case studies We developed several case studies in Coq related to correctness of computer systems. J.-F. Monin [7] from Grenoble subsite proved that the functional sprintf function of Danvy and the usual version of sprintf (with a dependent typing) are intentionally equal. Th. Hubert [3] developed libraries for certifying termination proofs using dependent pairs criteria in Coq.

1.2 Formal Mathematics and Mathematics Education

J. Duprat together with L. Vuillon from the Chambery subsite is working on formalizing discrete geometry by inductive objects.

1.3 Proof Technology

For his master work supervised by J-C. Filliâtre, N. Ayache [1] designed an interactive tactic for calling first-order automatic provers from the Coq proof assistant. The main difficulty was to derive an appropriate first-order theory from an higher-order environment.
1.4 Foundational Research

**Automatic deduction** Integrating automatic deduction into type theory is a long term research. P. Combinet [2, 2] made a significant contribution extending results in first-order intuitionistic logic with equality to the case of predicate defined by constructors. S. Conchon [1, 5] is studying decision procedures adapted to automatic resolution of proof obligations generated by checking correctness of programs.

**Extensionality** Mathematical proofs make an implicit use of extensionality which identify two objects which are provably equal. N. Oury [8] studied this rule and proposed a translation of a derivation in an extensional system into an intensional proof in a system like Coq.

2 Cooperation with industry

We are collaborating with Dassault Aviation in the area of proofs of C programs. We also have a collaboration with the Axalto company (a smartcards manufacturer) on proofs of Java and C programs, Java card applets and operating systems. Th. Hubert (Dassault), J. Andronick and N. Rouset (Axalto) are studying for their PhD part-time in the industry and part-time in our laboratory.

There is a collaboration between Orsay, Grenoble and the industrial subsite France Télécom R& D in the AVERROES national project (analysis and verification for the reliability of embedded systems) (http://www-verimag.imag.fr/AVERROES).

We also have a collaboration with César Muñoz at NIA, Hampton, USA on proof of Java programs for avionics.

We participate to the new competitiveness cluster Systematic (http://www.systematic-paris-region.org). In this cluster, the main industrial and academic research centers in the Ile-de-France Region are collaborating in the area of complex systems.

3 List of visitors from other sites

D. Wallkiewicz and J. Chrząszcz from Warsaw University visited our site in June 2005. They gave a seminar on “Consistency and Completeness of Rewriting in the Calculus of Constructions”. Patryk Czarnik a Phd student from Warsaw University visited the Orsay group 4 days in april 2005 and gave a seminar on his experiment using the Krakatoa tool for proving Java programs.

**Sites interactions** We have many exchanges (including a common seminar) with the LogiCal project which is part of the INRIA site.

**Visit to other sites** Jean-Christophe Filliâtre visited Queen Mary site in april 2005 and gave a seminar on “Verifying C and Java programs”.

3.1 URL

http://www.lri.fr/demons/introduction.en.html

4 List of publications

Refereed journal papers


Refereed conferences papers


PhD and Master thesis


Manual

Cross-references