

Workshop “analyse to compile, compile to analyse”

Sunday April 3, 2011, 14:00 – 18:00

PROGRAM

14:00 Welcome

14:00-15:00 Keynote speech by Sandrine Blazy INRIA - IRISA and University Rennes 1
<<Mechanized semantics for compiler verification>>

Compilers are complicated pieces of software that sometimes contain bugs causing wrong executable code to be silently generated from correct source programs. In turn, this possibility of compiler-introduced bugs diminishes the assurance that can be obtained by applying formal methods to source code.

The CompCert project is an ongoing experiment in developing and formally proving correct a realistic, moderately-optimizing compiler from a large subset of C to PowerPC/ARM/x86 assembly languages. The correctness proof, mechanized using the Coq proof assistant, establishes that the generated assembly code behaves exactly as prescribed by the semantic of the C source, eliminating all possibilities of compiler-introduced bugs and generating unprecedented confidence in this compiler.

The compiler relies on many intermediate languages. All the languages of the compiler are precisely defined by mechanized operational semantics and share a common memory model. This talk gives an overview of the methodologies required for defining these languages and proving the compiler correctness.

Important ! This talk will not conflict with Xavier Leroy's talk on Tuesday, 5

15:00 – 15:30 Analyze to verify compilation; Verify compilation to analyze Xavier Rival.

15:30 – 16:00 Static Loops Analysis in Concurrent Programs Lies Lakhdar-Chaouch, Bertrand Jeannot and Alain Girault.

16:00 Pause

16:30 – 17:00 Dependencies between Analyses and Transformations in the Middle-End of a Compiler François Irigoien, Fabien Coelho and Béatrice Creusillet.

17:00 – 17:30 Compiler-driven Optimization of the Worst-Case Execution Time Florian Brandner and Alain Darte.

17:30 – 18:00 Bringing IR Interpretation to Compilers for Analysis Purpose Nicolas Benoit and Stéphane Louise.