The treatment of contexts in meta-theory:
Formalize and implement a context mechanism for Abella

Adviser
Dale Miller, Parsifal team, INRIA-Saclay and LIX/Ecole Polytechnique,

Venue and duration
This internship will be conducted at the LIX lab on the Ecole Polytechnique campus in Palaiseau. This internship can last 2 or 3 month and should be taken during the months May/June/July/August.

Brief description of internship
When reasoning about formal syntactic objects that contain bindings (eg, formulas, programs, and proofs), one must almost always treat the syntax together with a context. For example, typing a program requires a typing context for all free variable. The Abella prover (a recently implemented interactive theorem prover built between the University of Minneapolis and the Parsifal team) treats contexts as list structures. The goal of this internship is to examine example Abella proofs and propose and implement a new approach to contexts since they should have a richer logical status than that of being ``just lists."

For example, context seem to be lists that generally have simple ``regular" structure. It should be possible to describe this structure and to automate the following questions:

- If one inserts certain items into a context, that context still has the same structure.
- If one selects a member from the context, automatically prove that that item satisfies certain syntactic restrictions.
- Is one context contained in another context even though they might satisfying different descriptions.

It is important to point out that these contexts arise from treating syntax that contains bound variables and that higher-order unification is part of treatment of contexts and the items they contain. The success of these automatic checks will be judged by how naturally and effectively they are used in practical theorem proving projects.

This internship should provide the basis for a PhD topic in the general area of the automation of the meta-theory of programming and specification languages.

Requirements
A successful candidate should have a strong background in computational logic generally. Particular needs include: a basic background in sequent calculus and the lambda-calculus as well as programming skills in high-level languages such as ML, OCaml, Prolog, and lambda Prolog. Since the Parsifal team is staffed with international personnel, English is the working language for the team and this internship.

References
The Abella theorem prover was developed by Andrew Gacek while he was a PhD student at the University of Minnesota and a postdoc at Parsifal. Follow the link for this

prover to find a list of relevant publications and a large list of examples. The OCaml source for this prover is available.

The details of the logic underlying Abella are described in detail in the following three papers. A full understanding of these papers is not required for this internship.

1. *A two-level logic approach to reasoning about computations*, by Andrew Gacek, Dale Miller, and Gopalan Nadathur, Submitted 16 November, 2009. [arXiv | .pdf]
2. *Nominal abstraction*, by Andrew Gacek, Dale Miller, and Gopalan Nadathur, Submitted 4 August, 2009. [arXiv | .pdf]
3. *Combining generic judgments with recursive definitions*, by Andrew Gacek, Dale Miller, and Gopalan Nadathur, Proceedings of LICS 2008 (F. Pfenning, ed.), IEEE Computer Society Press, June 2008, pp. 33-44. [slides | .pdf]

The following papers describe how Abella can be used to prove meta-theorems.

1. *The Abella interactive theorem prover (system description)*, by Andrew Gacek, Proceedings of IJCAR 2008 (A. Armando, P. Baumgartner, and G. Dowek, eds.), Lecture Notes in Artificial Intelligence, vol. 5195, Springer, August 2008, pp. 154-161. [slides | .pdf]
2. *Reasoning in Abella about structural operational semantics specifications*, by Andrew Gacek, Dale Miller, and Gopalan Nadathur, International Workshop on Logical Frameworks and Meta-Languages: Theory and Practice (LFMTP 2008) (A. Abel and C. Urban, eds.), Electronic Notes in Theoretical Computer Science, no. 228, 2008, pp. 85-100. [slides | .pdf]

Date: 1 November 2010, revised 7 Feb 2011.