Logic in Milan
an informal workshop

When: March 16 2010, 15:00 – 18:00
Where: DSI, Aula Riunioni 2 Piano
Why: why not

Speaker(s): Silvio Ghilardi & Francesco Alberti (DSI)
Title: Case study: Verification of a fault tolerant protocol with MCMT

Speaker: Marco Gaboardi (Bologna)
Title: Light Logics: a tool for polynomial time in functional programming
Abstract: We overview the Light Logics approach to Implicit Computational Complexity presenting different proposals for the class of polynomial time functions. We focus on the use of light logics in designing type systems for the Lambda Calculus considered as a core functional programming language. We conclude by suggesting some possible extensions and generalizations of their structural proof-theoretical principles.

Speaker: Ben Kavanaugh (LFCS, Edinburgh)
Title: Context Grammars
Abstract: Programming language prototyping tools depend on powerful and modular grammar specifications. Boolean grammars are an extension of context free grammars that allow with an effective recognition algorithm that allow intersection and negation. I define an extension of Boolean grammars with an unbounded left and right context operator to give a formalism that we call context grammars. These grammars are found to be very useful in practice. I give a denotational semantics for a well-behaved section of these grammars called stratified context grammars and introduce a recognition algorithm that is correct with respect to the denotational semantics. I also discuss how combinator parsers might be transformed to implement the recognition algorithm while allowing freedom to perform more flexible parsing.

Speaker: Loris Bozzato (Insubria)
Title: Constructive Semantics and Calculi for Description Logics
Abstract: In recent years, Description Logics (DLs) are emerging among logic based formalisms for knowledge representation thanks to their clearly defined semantics and relevant applications.
DLs are mostly based on fragments of classic first order logic: however, there are also some proposals for intuitionistic and constructive interpretations of DLs, often addressed as Constructive Description Logics. We present our contributions to this field by discussing two different constructive semantics for the basic description logic ALC. We also provide some hints on possible applications of the resulting logics in action formalisms and semantic web service compositions.

Speaker: Guido Fiorino (Bicocca)
Title: Simplification Rules for Intuitionistic Propositional Tableaux
Abstract: The implementation of a logic requires, besides the definition of a calculus and a decision procedure, the development of techniques to reduce the search space. In this paper we introduce some simplification rules for Intuitionistic propositional logic that try to replace a formula with a logically equivalent ``simpler'' one with the aim to reduce the search space. Our results are proved via semantical techniques based on Kripke models.

Speaker: Alessandro Avellone (Bicocca)
Title: Avoiding duplications in tableau systems for intuitionistic logic
Abstract: We present some techniques which bound the proof search space in propositional intuitionistic logic. These techniques are justified by Kripke semantics and are the backbone of a tableau based theorem prover (PITP) implemented in C++.

Speaker; Camillo Fiorentini (DSI) – time permitting
Title: Applying Answer Set Programming to UML Model Validation.
Abstract: We investigate the application of ASP to model validation in a CASE setting, where models are UML class and object diagrams with (OCL) constraints. A class diagram represent an abstraction of the problem domain; the objects populating a system state a ``snapshot" of a corresponding counterpart in the modeled world. In UML, snapshots are represented by object diagrams. The legal snapshots are those that satisfy the constraints. We present the design and implementation of a snapshot generator for UML models that employs DLV-Complex as a generator engine, the answer sets representing the legal snapshots. The more theoretical contribution consists in a specialized representation of UML into the DLV-Complex's language, which makes crucial use of external functions, but still requires a further extension with inductive types and existential quantification. This is so as to avoid the generation of useless isomorphic answer sets. `http://cooml.dsi.unimi.it/talks/lpnmr09Slides.pdf`