P.R.I.N. Project PROTOCOLLO
(from PROofs TO COmputations through Linear LOgic)

From Computations to Proofs (and back...)

Simona Ronchi Della Rocca
Dipartimento di Informatica
Università di Torino

The strong relationship between Logic and Computer Science is well established in the literature. We are particularly interested in the so called "Curry-Howard isomorphism", relating proofs with programs, formulas with types, and cut-elimination with evaluation rules. So, for reasoning about programs, both the logical and the programming settings seem to be equally suitable. But sometimes it is possible to gain something by crossing the border. We will show this through a case study. The starting point is the intersection type assignment for λ-calculus, where types (called *intersection types*) are formulas of the implicative and conjunctive fragment of the intuitionistic logic ($LJ_{\rightarrow,\wedge}$) . Despite this logical view of intersection types, there is no isomorphism between the type assignment and $LJ_{\rightarrow,\wedge}$. Looking for a logical foundation of intersection type assignment system, we define a new logic, whose proofs are equivalence classes of proofs of $LJ_{\rightarrow,\wedge}$, where the notion of equivalence expresses a synchronous introduction of some connectives. This logic, called ISL, can be interesting in itself, independently from the original motivation. Now, going back from the world of logic to the world of programming, we obtain, through a decoration of ISL, a new programming language, that turns out to be a paradigmatic language for espressing the *discrete polymorphism*, an old open problem. The relation with Linear Logic is methodological: the key point of the construction is based on the decomposition of the intuitionistic AND into two connectives, with a synchronous and asynchronous behaviour respectively.

(the talk is based on a joint work with Elaine Pimentel and Luca Roversi).