Software engineering diagramming: Improving readability with the Semiology of Graphics

**Context.** The Model Driven Engineering (MDE) refers to the use of models in Software Engineering. MDE is generally associated to the principle of modelling and code generation. This is a seducing method because it seems to score higher in terms of software reuse than library or frameworks. And it is particularly useful to manage the increasing complexity of current middleware [1].

Many modelling languages which are created from a MDE perspective are based on a graphical language (like UML). In principle, almost all software practitioners wish to get a graphical tool that help them saving time. In practice, there are barriers for wholesale adoption of such tools. The quality of graphical notations is one of these barriers: for a long time, a well-established belief was that using boxes and arrows was enough... reality on the ground shows us that it was not. Daniel L. Moody elegantly demonstrated it in [2]. He also pointed out that this is a recurrent issue in other scientific domains (cartography, statistical representations, psychology...). Our work follows from this observation.

**Approach.** One source of inspiration that is referred by Moody is the Semiology of Graphics (SoG). Created by Jacques Bertin [3], one of the major scientists in cartography, the SoG defines a set of visual variables and rules that structure any type of formalised visual constructions. For Bertin, if someone wants to efficiently communicate information through a visual construction, she/he has to make the best use of these visual variables (according their advantages and their disadvantages). Above all, she/he has to indicate clearly in a caption how these variables are employed. We implemented practically all the visual variables in our metamodelling tool ModX (http://www.lifl.fr/modx) in order to determine what can be the benefits of the SoG for Software Engineering.

**Goal.** The project mainly focus on the legends of graphical modelling languages. The student will be in charge to define the features of legends in order that they are compliant with SoG. She/he will first finalise the implementation of visual variables in ModX. The «grain» is the main variable that remains to develop. From a technical point of view, it consists in developing a library in Java which allows to draw line with any type of decoration. From a conceptual point of view, the «grains» includes and uses all the other variables and will naturally lead the student to the issue related to legend (second part of the project).

The student will collaborate with Nicolas Genon from the University of Namur and Sophie Dupuy-Chessa from the University of Grenoble.