

SCIENCES ET TECHNIQUES DU LOGICIEL SCIENCES AND TECHNIQUE OF SOFTWARE

A l'aube du 21^{ème} siècle, l'industrie du logiciel est devenue l'un des secteurs clef de l'activité économique et participe de manière très significative à la mutation de notre société. Celle-ci devient chaque jour davantage dépendante de la qualité de conception et de réalisation des centaines de milliers de logiciels utilisés quotidiennement dans le monde, et de plus en plus souvent interconnectés.

Il y a un manque de rigueur et simplicité dans les techniques de conception, de développement et de maintenance des logiciels. On pourrait arguer que ce manque de qualité vient d'une manque de professionnalisme des développeurs qui sont souvent des non professionnels de l'informatique. Mais ce ne serait pas réaliste, car non seulement les informaticiens professionnels commettent eux aussi des erreurs (dues à la complexité et à la diversité croissante des problèmes), mais aujourd'hui tout responsable de petite entreprise et nombre de particuliers doivent concevoir et réaliser eux mêmes les logiciels dont ils ont besoin.

Les techniques et outils de conception et de développement de logiciels de qualité doivent donc être utilisables par des non spécialistes et être les plus automatisées possibles.

STL regroupe sept équipes de l'I3S qui visent à simplifier et à mieux contrôler la qualité des conceptions et des développements de logiciels, le plus souvent par une approche à objets :

EP travaille principalement sur l'implantation des langages applicatifs et plus particulièrement le langage Scheme.

LANGAGES étudie les moyens de développer de manière sûre et rapide des outils pour les différentes phases du développement de logiciels. Ces outils reposent le plus souvent sur des langages, l'objectif est de définir, d'évaluer et d'implémenter de nouveaux langages, avec des techniques les plus automatiques possibles.

M@inline travaille sur l'information non linéaire et ses applications à l'enseignement à distance. L'objectif est principalement le développement des technologies et des outils appropriés pour l'enseignement à distance, appliqué à la création et à l'évaluation de supports pour l'enseignement interactif et multimedia

MECOSI travaille sur la partie amont du cycle de développement des logiciels. L'objectif est de fournir aux concepteurs de systèmes d'information des démarches guidées, basées sur des informations concrètes sur le domaine à modéliser et qui permettent d'évaluer la qualité des schémas de conception obtenus.

Oasis étudie les applications réparties. Son objectif est de proposer des principes fondamentaux, des techniques et des outils pour la construction, l'analyse, la validation, la vérification et la maintenance de systèmes fiables.

OCL travaille sur la phase de développement des logiciels par réutilisation de composants. L'objectif est de mieux contrôler la sémantique des assemblages de classes, presque toujours en perpétuelle évolution, dans un cadre persistant et réparti.

RAINBOW étudie les interdépendances comportementales entre les objets, durant la phase d'exploitation. L'objectif est de permettre une expression locale des dépendances avec un contrôle global de cohérence.

S *In this beginning of the 21st century, the software industry has become one of the key segments of the economical activity. It participates in a very significant way to the mutations of our society. More and more every day, we are becoming dependent on the quality of design and realization of hundreds of thousands of software products. They are used daily everywhere, and more and more often in an interconnected way. The techniques of design, development and maintenance of software are lacking in simplicity and rigor. One could argue that this results from a lack in professionalism for developers, who too often are simple amateurs in programming. This would not be realistic: professional programmers do make bugs, because of the increasing complexity and diversity of problems; moreover, all small enterprise managers and many private persons must design and build themselves the software products they need.*

Thus, the techniques and tools for designing and developing quality software must be usable by non specialists, and be automatized as much as possible.

EP *team works on the implementation of applicative languages, and more specifically on the Scheme language.*

LANGAGES *team works on the means of developing in a safe and fast way the tools needed for various phases of software development. Most often these tools rely on programming languages, and thus the team main goal is to define, evaluate and implement new languages, using techniques as much automatic as possible.*

M@inline *team works on non linear information, and its applications to remote teaching. The main goal is to develop techniques and appropriate tools for this purpose, and to apply them to the building and evaluation of supports for interactive and multimedia teaching.*

MECOSI *team works on the upper part in the software development cycle. Its goal is to provide the designers of information systems with guided approaches, based on concrete information about the modeled area. This allows to assess the quality of the design schema that are obtained.*

Oasis *team works on distributed applications. Its goal is to suggest fundamental principles, techniques and tools, for the construction, analysis, validation, checking and maintenance of reliable systems.*

OCL *team works on the development phase of software, focusing on components reuse. Its goal is to control better the semantics of class assemblies, most often in permanent evolution, in a persistent and distributed context.*

RAINBOW *team works on the behavioral interdependencies between objects, during the use phase. Its goal is to provide for a local expression of dependencies, together with a global integrity checking*